A Formal Approach to Computer Aided 2D Graphical Design for Blind People

Sandra Fernando

A thesis presented for the degree of

Doctor of Philosophy

Department of Computing

Goldsmiths, University of London

United Kingdom

Declaration

I hereby declare that I composed this thesis entirely myself and that it describes my own research.

Sandra A. Fernando,
Department of Computing,
Goldsmiths College, University of London,
March 2021

Dedication

This dissertation is dedicated to my mother, Padma Fernando, my father Emil Fernando, and my siblings who offered unconditional love, support and have always been there for me. God bless you all. Mum, Dad, thank you for investing in me and trusting me all along; I hope this dissertation goes some way to acknowledging your love.

Acknowledgements

I should like to acknowledge my warmest and sincere thanks to my two supervisors, Dr James Ohene-Djan (lead supervisor) and Dr Nicolas Hine. I could not have completed this research without Dr James Ohene-Djan's continuous encouragement and support. I should also like to thank both Dr Nicolas Hine and late Dr Sebastian Danicic for their prompt and detailed feedback on my work. A special thanks goes to Professor Bal Virdee for giving his time, expertise and experience so freely during the correction stage.

I should like to acknowledge my gratitude to the Department of Computing of Goldsmiths College, the University of London, for supporting me to participate at an international conference, and purchase of specialist equipment necessary to conduct the research. Goldsmith College provided me with a conducive environment that enabled me to carry out my research effectively.

The research report was initially funded by MACS charity. I therefore want to thank the board of trustees at MACS. I would like to acknowledge RNIB, BeyondSight charity and Thomas Pocklington Trust for their involvement and participation in experiments and interviews that were conducted in the research.

My chance encounter with Toby Ott and Barry Ott sparked my interest in research to help blind people. They drew my attention in the need for an effective drawing tool specifically for blind people. I would like to thank both for triggering my interest in research. I would like to thank Graham De-Smith for helping me write my first funding bid and for introducing me to blind and partially sighted people who gave me valuable insight of what was required from the drawing package I was developing.

Finally, a special thanks goes to Christopher McKiernan, Colin Haikin, and Lance Stanford. Thank you for supporting me during the research with your ideas and assistance with proofreading of my research publications.

Executive Summary

The growth of computer aided drawing systems for blind people (CADB) has long been recognised and has increased in interest within the assistive technology research area. The representation of pictorial data by blind and visually impaired (BVI) people has recently gathered momentum with research and development; however, a survey of published literature on CADB reveals that only marginal research has been focused on the use of a formal approach for on screen spatial orientation, creation and reuse of graphics artefacts. To realise the full potential of CADB, such systems should possess attributes of usability, spatial navigation and shape creation features without which blind users drawing activities are less likely to be achieved. As a result of this, usable, effective and self-reliant CADB have arisen from new assistive Technology (AT) research.

This thesis contributes a novel, abstract, formal approach that facilitates BVI users to navigate on the screen, create computer graphics/diagrams using 2D shapes and user-defined images. Moreover, the research addresses the specific issues involved with user language by formulating specific rules that make BVI user interaction with the drawing effective and easier. The formal approach proposed here is descriptive and it is specified at a level of abstraction above the concrete level of system technologies. The proposed approach is unique in problem modelling and syntheses of an abstract computer-based graphics/drawings using a formal set of user interaction commands. This technology has been applied to enable blind users to independently construct drawings to satisfy their specific needs without recourse to a specific technology and without the intervention of support workers. The specification aims to be the foundation for a system scope, investigation guidelines and user-initiated command-driven interaction. Such an approach will allow system designers and developers to proceed with greater conceptual clarity than it is possible with current technologies that is built on concrete system-driven prototypes.

In addition to the scope of the research the proposed model has been verified by various types of blind users who have independently constructed drawings to satisfy their specific needs without the intervention of support workers. The effectiveness and usability of the proposed approach has been compared against conventional non-command driven drawing systems by different types of blind users. The results confirm that the abstract formal approach proposed here using command-driven means in the context of CADB enables greater comprehension by BVI users. The innovation can be used for both educational and training purposes. The research, thereby sustaining the claim that the

abstract formal approach taken allows for the greater comprehension of the command-driven means in the context of CADB, and how the specification aid the design of such a system.

Contents

Declaration **1**

Dedication **2**

Executive Summary **4**

List of Tables **13**

List of Figures **17**

1 Introduction **20**

1.1 Motivation . . . 20

1.1.1 Background . . . 21

1.1.2 What is Missing? . . . 23

1.1.3 Initial Challenges . . . 27

1.1.4 Derived Challenges . . . 27

1.2 Technical Aims of Research . . . 27

1.2.1 The Central Question . . . 27

1.2.2 Derived Problems . . . 28

1.2.3 Research Limitations . . . 28

1.2.4 Research Goals . . . 28

1.3 Research Contribution . . . 29

1.3.1 The Research Impact . . . 29

1.4 Overview of Research Methodology . . . 29

1.4.1 Why is a Formal Approach Used to Model SETUP09? . . . 30

1.5 Thesis Structure . . . 31

1.5.1 Road Map of the Thesis Structure . . . 34

2 Assistive Technology and Design Issues **35**

2.1 Visual Impairment . . . 36

2.1.1 How Vision Works . . . 37

2.1.2 The Needs of BVI People . . . 38
2.2 Use of Assistive Technology . . . 39
2.2.1 Reading Technology . . . 40
2.2.2 Speech Technology . . . 40
2.2.3 Tactile Technology . . . 42
2.2.4 Haptic technology . . . 43
2.2.5 Issues and Design Considerations . . . 45
2.2.6 Conclusion . . . 47
2.3 System Design and Evaluation using HCI Approach . . . 47
2.3.1 Overview . . . 47
2.3.2 HCI Systems Designed For BVI and With BVI participants . . . 50
2.3.3 Representing Users in Accessibility Research . . . 52
2.3.4 Conclusion . . . 53
2.4 System Design and Evaluation using Formal Approach . . . 53
2.4.1 Systems Designed with Rule Based Grammar . . . 54
2.4.2 Formal Modelling of User Interaction . . . 59
2.5 Conclusion . . . 61

3 Modelling Issues of Computer-aided Drawing for Blind People (CADB) and Motivation 62

3.1 Issues of Drawing Techniques . . . 63
3.1.1 Issues of Analogue Drawing Tools . . . 63
3.1.2 Issues of Digital Drawing Tools . . . 64
3.1.3 Perceptual Models . . . 71
3.1.4 Blind Drawing . . . 74
3.1.5 Vision Substitution and Graphics . . . 76
3.1.6 Shape Language Considerations . . . 81
3.2 Issues of Navigation Techniques . . . 82
3.2.1 Cognitive Models . . . 84
3.2.2 Spatial Exploration Strategies . . . 84
3.2.3 Mental Models in Computer Environment . . . 87
3.2.4 Spatial Language Considerations . . . 89
3.3 Issues of Usability in Assistive Technology . . . 91
3.3.1 Usability Models . . . 92
3.3.2 Design Consideration of ISO Standards . . . 99
3.3.3 Cognitive Dimensions of Notations . . . 101
3.3.4 Design Considerations of Blind Drawing Technologies . . . 103
3.3.5 Usability Language Considerations . . . 105

3.4 Understanding Blind Users 106
3.4.1 Initial Data Gathering 106
3.4.2 Conclusion 110
3.5 Classification of a CADB model 111
3.6 Conclusion 113

4 A Formal CADB Model Design 114

4.1 Design Decisions 114
4.1.1 Drawing with Command Languages 116
4.1.2 Location Referencing and Cell Referencing 117
4.1.3 Cell Recursion and Points 119
4.1.4 Zoom Levels 120
4.2 Architecture for SETUP09 Model 121
4.2.1 Architectural Assumptions 122
4.2.2 Technical Approach 122
4.3 Introduction to Grammar formation 124
4.3.1 SPL (Space Language) Grammar Formation 125
4.3.2 SHL (Shape Language) Grammar Formation 128
4.3.3 USL (Usability Language) Grammar Formation 132
4.3.4 Context-Free Grammar for a computer-aided drawing System for Blind People . 135
4.4 Opportunities Afforded by CADB 137
4.5 What is Next? 138

5 The SETUP09 Model Implementation 139

5.1 SETUP09 Backend Class Interaction 139
5.2 SETUP09 Command Interpreter Development 141
5.2.1 Lexical Analysis 142
5.2.2 Lexical Semantic Analysis 142
5.2.3 Parsing 143
5.2.4 Semantic Validation - Abstract Syntax Tree (AST) 145
5.2.5 Target Code (Art) Generation 149
5.3 System Frontend Development 150
5.3.1 A Demonstration of SETUP09 160
5.3.2 System Architecture 160
5.4 What is Next? 162

6 Experimental Evaluation of SETUP09 Model Use 164

6.1 The Design Rationale of Methodologies 164
6.2 Evaluation Metrics 165

6.3 Study 1: Screen Navigation Technique with Blind People . . . 167
6.3.1 Hypothesis . . . 167
6.3.2 Method . . . 167
6.3.3 Procedure . . . 169
6.3.4 Results . . . 173
6.3.5 Other Observations: Memorability and Guidance . . . 177
6.3.6 Findings of Study 1 . . . 179
6.3.7 Limitation . . . 181
6.3.8 Conclusion . . . 181
6.4 Study 2: Graphics Creation Technique with Blind, Partially Sighted and Sighted Individuals . . . 182
6.4.1 Hypothesis . . . 182
6.4.2 Method . . . 183
6.4.3 Procedure . . . 185
6.4.4 Results . . . 187
6.4.5 Other Observation: Age and Previous Experience . . . 200
6.4.6 The post SETUP09 questionnaire . . . 202
6.4.7 Findings of Study 02 . . . 205
6.4.8 limitation . . . 207
6.4.9 Conclusion . . . 209

7 SETUP09 Model Revision 211
7.1 Grammar Model Revision . . . 211
7.1.1 Usability Grammar Revision . . . 212
7.1.2 Shape Grammar Revision . . . 213
7.1.3 Space Grammar Revision . . . 214
7.1.4 Revised Model . . . 215
7.2 Non-Grammar Revision . . . 217
7.2.1 Revised Architecture . . . 218
7.3 Conclusion . . . 219

8 Comparative Evaluation of SETUP09 Model Techniques 220
8.1 Study 3: SETUP09 comparison with Analogue Drawing Technique . . . 220
8.1.1 Hypothesis . . . 220
8.1.2 Procedure . . . 221
8.1.3 Results . . . 223
8.1.4 Findings . . . 234
8.1.5 Limitations . . . 235
8.1.6 Conclusion . . . 235

8.2 Study 4: Pilot study of Cell Referencing Systems . . . 236
8.2.1 Hypothesis . . . 236
8.2.2 Procedure . . . 236
8.2.3 Results . . . 237
8.3 Study 5: SETUP09 Comparison with Digital Drawing Technique . . . 238
8.3.1 Hypothesis . . . 238
8.3.2 Procedure . . . 239
8.3.3 Results . . . 241
8.3.4 Findings . . . 247
8.3.5 Limitations . . . 248
8.4 Comparing the Proposed Model with Related Work . . . 249
8.4.1 Comparison of Screen Navigation Techniques . . . 251
8.4.2 Conclusion . . . 251

9 Design Guidelines for CADB 253

9.1 Design Guidelines to Establish a Formal Specification Language for CADB . . . 253
9.1.1 Usability Guidelines . . . 254
9.1.2 Space Guidelines . . . 254
9.1.3 Shape Guidelines . . . 255
9.2 Process Flow of CADB . . . 255
9.2.1 Process Flow Diagram of CADB . . . 257
9.3 Conclusion . . . 258

10 Contribution, Further Direction and Conclusion 259

10.1 Contribution . . . 259
10.1.1 Motivation Revisited . . . 260
10.1.2 Challenges Revisited . . . 260
10.1.3 Technical Aims Revisited . . . 261
10.1.4 Summary of Contribution . . . 262
10.2 Application of Research . . . 262
10.3 Use in the Real Life . . . 262
10.4 Further Direction for Research . . . 263
10.5 Research Conclusion . . . 265

Appendix A Java Front End Lexical Phases 266

A.1 Lexical Analysis . . . 266
A.2 Lexical Semantic Analysis . . . 267

Appendix B Java Front End Parsing Phase **270**

B.1 Paring . . . 270

B.2 Parsing . . . 270

Appendix C Java Front End Target Code Generation Phase **275**

C.1 Abstract Class . . . 275

C.2 System State Object . . . 276

C.3 Concrete Classes . . . 277

C.4 Main Interface class . . . 292

Appendix D Details of the Survey, Training and Experiments **300**

D.1 Survey 1 . . . 301

D.2 Surveying 2 . . . 303

D.3 Study 1 - Evaluation of a Screen Navigation System . . . 305

D.3.1 User Guide . . . 305

D.3.2 Observer Guide Used in Study 1 . . . 307

D.4 Study 2 - Evaluation of Graphics manipulation Technique (BVI and Sighted) . . . 308

D.4.1 Testing Event Information . . . 308

D.4.2 SETUP09 Navigation System Training 2.1 . . . 309

D.4.3 Study 2 - Data and Results . . . 310

D.5 Study 3 - Comparative Evaluation of Digital System versus Analogue System . . . 312

D.5.1 Preparatory Work . . . 312

D.5.2 SETUP09 System Training 3.1 . . . 313

D.5.3 Study 3 - Results and Data . . . 315

D.5.4 Study 3 - Tasks and Commands . . . 317

D.5.5 Commands of the Cell Referencing Task . . . 317

D.5.6 Study 5 - SETUP09 Comparison with Digital Drawing Technique . . . 317

Bibliography **334**

List of Tables

2.1 Visual Impairment Classification: Presenting Distance Visual Acuity, [WHO, 2015] . . . 37
2.2 Types of Assistive Computer Technologies, [Davydov and Lozynska, 2016] 39

3.1 Navigation Strategies [Ungar, 2000] 86
3.2 Main Requirements for Educational Software Applications for Visual Impaired and Blind People [Bocconi et al., 2007] 95
3.3 Second Survey 109

4.1 Cell Names with Numerical Keys and Compass Names 118
4.2 Centre Point Cell Referencing and Multiple Points of Cell Referencing 119
4.3 Cell Recursion and Referencing 120
4.4 Zoom Levels 120
4.5 BNF Production Symbols Modified Using [Robson, 2019] 124

5.1 Hot Key Commands for the Third Version 155
5.2 Other Non Hot Key Commands of the Third Prototype 155

6.1 Number of Correct and Incorrect Recognitions Made During Activity 1-5 174
6.2 Use of System Support Commands by Participants 178
6.3 Participants Feedback 178
6.4 Participants Comments: 1 - Strongly Agree to 5 -Disagree. 179
6.5 Demography of the Participants. CL = Computer Literacy, TGC = Experience with Tactile Graphic Creation, NA = Not Applicable 186
6.6 Analysis of Instructed Drawing Performance in minutes and seconds. SD = Standard Deviation 189
6.7 T-Test values of Instructed Drawing Performance between different subgroups. 189
6.8 Analysis of Non-Instructed Drawing Performance in Minutes and Seconds. SD = Standard Deviation 193
6.9 T-Test values of Non-Instructed Drawing Performance Between Different Subgroups. . . 193
6.10 Analysis of Image Recognition Performance in minutes and seconds. SD = Standard Deviation 197

6.11 T-Test values of Image Recognition Performance between different subgroups. 198
6.12 Answers to Post Study Questions based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants 204
6.13 Personal views of the participants 205

8.1 Information on the Early and Late Blind Participants in the Experiment 222
8.2 Task 02: Table and Task 03: Flowchart 223
8.3 Tasks 2 and 3: Images Produced Using a Paper Kit by Late Blind Person 225
8.4 Tasks 2 and 3: Image in SETUP09 by Late Blind Person 225
8.5 Survey Results of the Experiment 233
8.6 SETUP09 System Images of Task 04 Created by a Participant: Lines with a Single Point Cell Reference and a Multi-points Cell Referencing Technique 237
8.7 IC2D Outputs 241
8.8 Information on Blind and Sighted Participants 241
8.9 IC2D Published Drawing and SETUP09 Drawing Outputs Produced by Blind Participants 242
8.10 Results of IC2D (n=16) and SETUP09 (n=8) testing with Blind and Sighted Participants 247
8.11 SETUP09 Participants feedback 247

D.1 Survey 01: Facts Finding 303
D.2 Survey 02: Facts Finding 305
D.3 Average Completion Percentages of BVI Individuals 311
D.4 Time Taken by BVI Individuals During Instructed Drawing 312
D.5 Errors Made by BVI during Instructed Drawing 312
D.6 Average Accuracy of Early and Late Blind Participants SETUP09 316
D.7 Average Accuracy of Early vs. Late Blind Participants with Paper toolkit 316
D.8 Sum of Errors made by Early versus Late Blind Using Participants 317

List of Figures

1.1 Motivation for a Formal Approach to a Computer Aided Command Driven Drawing System for Blind Users 25
1.2 Road Map of the Thesis Structure 34

2.1 Historical trends for primary screen reader usage [WebAIM, 2020] 40
2.2 Comparison Of The Best Speech Recognition Software, www.softwaretestinghelp.com . 41
2.3 Braille [RNIB, 2018] 42
2.4 BrailleNote [Humanware, 2018] 42
2.5 Evolution of computing haptic platforms, [WANG et al., 2019] 44
2.6 Development Approach of HCI 48
2.7 Development Approach of Formal Modelling 54
2.8 Computational Model for Integrating linguistic and Pictorial Information [Srihari, 1995] 56
2.9 Grammar Derivation Tree [Chase, 2002] 59

3.1 Tactile Graphic Production Toolkits, [Perkins School, 2019] 64
3.2 The IC2D Shape Palette, [Kamel and Landay, 2000] 66
3.3 System KEVIN, [Blenkhorn and Evans, 1998] 67
3.4 System PLUMB, [Calder et al., 2007] 68
3.5 System TUDUB, Automatic Interpretation and Presentation of Technical Diagrams for Blind People [Petrie et al., 2002] 69
3.6 Audio-Haptic Drawing Editor And Explorer for Education, [Rassmus-Grohn et al., 2007] 69
3.7 Plotter Language, [Fujiyoshi et al., 2008] 70
3.8 Plotter Language Graph, [Fujiyoshi et al., 2008] 71
3.9 Outline Drawing of a Table by a Blind Subject [Kennedy, 1997] 74
3.10 Outline Drawing of a Moving Wheel by Blind Subjects [Kennedy, 1997] 74
3.11 TDraw, A Chair and a Table [Kurze, 1996] 78
3.12 Use of a Grid System to Create a Square [Kamel and Landay, 2000] 79
3.13 Shapes 80
3.14 Classification Diagram of Shape Grammar 81

3.15 Classification Diagram of Spatial Grammar 90
3.16 Modified using CAT model [Hersh et al., 2008] 96
3.17 From the CAT model, possible communication options [Hersh et al., 2008] 97
3.18 International Classification of Functioning [ICF, 2019] 98
3.19 Classification Diagram of Usability Grammar 105
3.20 Resultant Bar Chart of the Second Survey 109
3.21 Derivation of Formal Specification : Concept Classification Diagram 112

4.1 Analogue versus Digital versus Language Driven Blind Drawing Systems 115
4.2 Components of SETUP09 Model 121

5.1 SETUP09 Class Interaction 140
5.2 SETUP09 Interaction of Classes 141
5.3 SETUP09 Class Interaction to draw a Line 141
5.4 Parser Flow 142
5.5 Abstract Syntax Trees (ASTs) 145
5.6 Image of a Line (in blue) from Prototype SETUP09 146
5.7 Abstract Syntax Trees (ASTs) 146
5.8 Image of a Circle (in blue) from Prototype SETUP09 147
5.9 Abstract Syntax Trees (ASTs) 147
5.10 Image of a 2D Fish from SETUP09 Prototype 148
5.11 Image of a 2D Fish from the Save Command 148
5.12 Abstract Syntax Trees (ASTs) 148
5.13 Image of a house from Prototype SETUP09 149
5.14 A blind Undergraduate Student Team Member Operating a Computer Keyboard with the Researcher. 151
5.15 prototype SETUP09 Version 01 152
5.16 prototype SETUP09 Version 02 153
5.17 Interface of the Third Version of SETUP09 154
5.18 SETUP09 Version 03: Flow Chart 156
5.19 SETUP09 Version 03: A scenery of a home 157
5.20 SETUP09 Version 03: Image of Data Flow Diagram 157
5.21 System Architecture 159
5.22 keyboard Input Modality 161
5.23 Swell Form Graphics Machine, [Braillo, 2019] 162

6.1 Test Design Flow to Measure Cognition of Navigation, Drawing and Comparison of Techniques. 166
6.2 System SETUP09: Screen Layout 169

6.3 System Navigation Path to East-Center-South 170
6.4 Swell Papers Used for Tactile Recognition 171
6.5 Participant Using Swell Paper 172
6.6 Participant Using SETUP09 System. 172
6.7 Number of Errors During Activity 1 - 5 175
6.8 Time and Errors During Activity 1 - 5 176
6.9 Activity 6 -Time Taken to Finding the Centre. 177
6.10 User Satisfaction - Where 1 is the Easiest and Most Satisfying Method, & 5 the Most Difficult. 177
6.11 Task 01: Triangles Presented for Instructed Drawing 183
6.12 Task 01: Diagram Presented for Instructed Drawing 183
6.13 Task 02 Non-Instructed Drawing: Stickman Produced by a Partially Sighted Participant and a 3D Chair Produced by a Sighted Participants 184
6.14 Images Presented for Image Recognition Task 185
6.15 Diagram Presented for Image Recognition Task 185
6.16 Instructed Drawing: Average Task Completion Time Based on Tasks of all participants (Blind, Partially Sighted, Sighted) 188
6.17 Instructed Drawing: Average Task Completion Time Based on Tasks and Vision with Blind (n=5), Partially Sighted (n=5) and Sighted participants (n=10) 189
6.18 Instructed Drawing: Errors made During Instructed Drawing Task based on Blind (n=5), Partially Sighted (n=5)and Sighted participants (n=10) 190
6.19 Instructed Drawing: Completion Percentage of all participants (Blind, Partially Sighted, Sighted) 191
6.20 Completion Time of Non-instructed Drawing with all participants (Blind, Partially Sighted, Sighted) 192
6.21 Completion Time of Non-instructed Drawing based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants 193
6.22 Errors of Non-instructed Drawing based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants 194
6.23 based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants 195
6.24 Image Recognition Time of All Participants (Blind, Partially Sighted, Sighted) 196
6.25 Image Recognition Time based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants 197
6.26 Image Recognition Accuracy of all Participants (Blind, Partially Sighted, Sighted) . . . 199
6.27 Average Performance Time Based on the age of participants (Blind, Partially Sighted) . 200
6.28 Average Performance Time Based on Previous (TGC) Tactile Graphic Creation Experience(Blind, Partially Sighted) 201

6.29 Average Accuracy of Diagram Drawing Task with Blind and Partially Sighted Participants with and without the Experience of Tactile Graphic Creation 202
6.30 Answers to Post Study Questions by all Participants demonstrated in a bar chart 204

7.1 Components of the revised SETUP09 Model 218

8.1 Rubber Mat, Embossing Film Papers[RNIB, 2019] 222
8.2 Overall Accuracy of Tasks by Both Groups 226
8.3 Accuracy of Paper Toolkit Tasks 227
8.4 Accuracy of SETUP09 Tasks 227
8.5 Errors in SETUP09 Tasks 228
8.6 Comparison of Errors Between Early Blind Participants with Paper Toolkit and SETUP09 System 229
8.7 Performance Time of both groups with Paper Toolkit and SETUP09 230
8.8 Performance time of Late and Early Blind Participants with Paper Toolkit 231
8.9 Performance time of Late and Early Blind Participants with SETUP09 System 231
8.10 Post Study Survey 233
8.11 Task 04: Tactile Images of Lines Used in the Experiment 236
8.12 Multiple Points of Cell versus Centre Point Cell Referencing 238
8.13 IC2D Objects Placement 240
8.14 Performance time of IC2D and SETUP09 - All tasks with both VI and Sighted Participants. IC2D n = 16, SETUP09 n = 8 243
8.15 Performance time of IC2D and SETUP09 - VI and Sighted participants for different tasks. IC2D n = 16, SETUP09 n = 8 244
8.16 Participants' Confidence of IC2D and SETUP09. IC2D n = 16, SETUP09 n = 8 245
8.17 Performance - accuracy rating of IC2D and SETUP09. IC2D n = 16, SETUP09 n = 8 . 246

9.1 Process Flow Diagram of CADB 258

A1 First Survey Questions from Focus Magazine, [MACS, 2019] 302
A2 The Flyer for Experiment 2. 309
A3 Experiment Raw Data of Blind and Sighted Individuals 311
A4 Experiment 3 Raw Data of Early and Late Blind Individuals 316
A5 Study 5 - IC2D Statistics of BVI and Sighted Individuals. 318
A6 Study 5 - SETUP09 Raw Data and Statistics of BVI and Sighted Individuals 319

Publications

1. Sandra Fernando. 2015. Blind drawing: investigation into screen location tracking for computer-aided interactive drawing. SIGACCESS Access. Comput. 111 (July 2015), 6-9. DOI:10.1145/2809904.2809906

2. J. Ohene-Djan and S. Fernando, Drawing for Blind Learners: Assistive Technology for Graphical Design, 2016 IEEE 16th International Conference on Advanced Learning Technologies (ICALT), Austin, TX, 2016, pp. 436-440. DOI: 10.1109/ICALT, 2016

3. Ohene-Djan, J. and Fernando, S. (2018), "Screen navigation system for visually impaired people", Journal of Enabling Technologies, Vol. 12 No. 3, pp. 114-128.

4. Fernando, S., and Ohene-Djan, J. (2020). An empirical evaluation of a graphics creation technique for blind and visually impaired individuals. British Journal of Visual Impairment, DOI: 10.1177/0264619620911422

5. Fernando S, Ohene-Djan J. (2021) A computer aided drawing system evaluation with early and late blind users. Assistive Technology (The Official Journal of RESNA). DOI: 10.1080/10400435.2020.1829745

Chapter 1

Introduction

This chapter gives an extensive overview of the investigation, modelling, and evaluation that was carried out in the research work presented in this thesis. The chapter starts with a brief background work of past and presents analogue and digital drawing tools, their offerings, and drawbacks. Although there are different drawing systems available for blind and visually Impaired (BVI) people (CADB) to address a variety of drawing needs, current systems are tightly coupled with input-output modalities that do not facilitate design changes, effective and independent screen navigation, and graphics creation with appropriate support functionalities. Currently, there is no formal method with a set of guidelines that overcome these challenges. The aforementioned analysis reveals a need for a new drawing technology, which is the subject of this thesis. Proposed in the thesis is a novel formal model-based approach. The proposed model has a solution based on a formal approach that can specify images without recourse to a specific technology solution in the design and development of a computer-aided drawing system, an approach that is not taken by current systems. The initial challenges of the proposed model were exposed by questioning blind and visually impaired (BVI) users' fundamental behaviours. Other specific challenges were revealed by questioning BVI users' on the design, development, and evaluation of a formal model. This approach uncovers pertinent research questions on the characteristic of the proposed formal model. The chapter discusses various research methodologies along with the appropriateness of the selected formal approach. A summary is provided on the research goals, research contribution, the wider social impact of the research outcome, and limitations. The introductory chapter ends with an outline of the rest of the thesis.

1.1 Motivation

This section presents an account of the motivation of the research, which is stemmed from (i) the needs of the wider BVI community; (ii) the gap in existing computer-aided drawing (CAD) drawing technologies; (iii) the need for sustainable design guidelines; and (iv) the challenges faced by assistive technology researchers of CAD systems.

Ceri Morris in his study [Morris, 2014] highlights the need for blind people to have efficient diagrammatic tools to put them on a level footing with sighted people in their day-to-day lives and for more effective and inclusive education. He discovered that the presence of special education support workers can negatively affect the social integration of students with visual impairment. Furthermore, success of education is influenced by factors such as the interaction between the student and the tutor, the specific needs of students, and the subject tutor's knowledge of the material. Morris' research reveals that visually impaired students have drifted away from selecting STEM subjects to more practical courses on feasibility grounds. This is because BVI pupils can lack in the understanding of how to interact with the alternative schema for diagrammatic based tasks that has long term disadvantaged for employment. Some BVI students prefer to select special education centres where the adaptive learning takes place. This is only possible where specialist equipment is available and BVI pupils are supported by suitably qualified teachers and tutors. However these students will have to eventually readjust to living in the sighted world once again, therefore specialist tools and knowledge from mainstream education providers are needed to promote inclusion in society. These findings are also supported by (Royal National Institute of Blind People) RNIB [SSMR, 2009] research on Blind and VI people's needs. RNIB's findings reveal that BVI students educationally benefit from slower paced delivery of teaching material and the use of specialised equipment. This means that mainstream schools can be inadvertently more damaging for blind and partially sighted people.

Research finding in the above two reports, i.e. [Morris, 2014] and [SSMR, 2009] reveal the importance for mainstream education providers investing in simple but self-reliant technologies that enables BVI people access to mainstream education that will facilitate better societal integration and employment opportunities, thus allowing them to work side by side with sighted people. The research in this thesis is on developing a digital drawing technique for BVI users based on a novel formal model-based approach. The proposed technology addresses fundamental requirements not only needed to the understanding of what drawing means for blind people, but it also promotes essential characteristics to increase the effectiveness of user interaction which is unique to CADB technology.

1.1.1 Background

There are many analogue systems used in education establishments for drawing shapes and creating artwork for blind people. For example, the InTACT SketchPad uses [inTACT, 2014] a stylus on the tactile film to create raised line that appears under the person's fingertips, so that they can feel the drawing. Sewell EZ Write N Draw Raise Line Drawing Kit [MaxiAids, 2019] is a rubber surface, foil sheet, and stylus on which one can write and draw. This equipment creates raised tactile results that are readable on the reverse side of the sheet. On the other hand, QuickDraw paper [APH, 2014b] is popular with children as can be used to create geometric shapes by simply drawing with a felt tip pen on a sponge paper. Another medium commonly used by BVI people is the sensational Blackboard [Sensational Books, 2012] which is a hard plastic board with a rubber surface on one side of it. This allows the user to use any A4 paper and a pen to draw on the rubber side of the board to feel the drawing as they do it. The advantage of the Sensational Blackboard is that it allows the user to feel the drawing as they do it. Also commonly used is the Swail l Dot Inverter [APH, 2014c]. This works by placing a braille paper on a rubber pad. The special stylus allows dots to be embossed upwards by puncturing the paper on the down stroke and then pulling the dot upwards on the up stroke. Another example is the the drawing tool Draftsman [Perkins School, 2019] that produces simple raised-line graphics that is useful for demonstrating maths and science concepts, practice handwriting skills, playing games such as tic-tac-toe and for creating art drawings.

The above technologies have numerous deficiencies which include none or very limited storage facility for future retrieval, limited usability features and functionality for supporting art production such as error correction, erase, redo, undo, group, and reuse. Furthermore, the technologies do not facilitate BVI users to work alone without having to rely on support workers. In the past, only marginal research work has been carried out on digital drawings for blind people which has scarcely addressed the issues just mentioned. This is the main motivation for the research which is the subject of this thesis.

Digital drawing work has gathered momentum with speech technologies [Zhang et al., 2017], haptics, Hyper-braille [Leo et al., 2016], 3D printing [Williams et al., 2014], and with non-speech audio information (Sonification) [Walker and Mauney, 2010]. Most blind learners often seek the help of a support worker to draw pictures or diagrams, or they avoid drawing. This is because they find it difficult to believe that they would be able to create pictures or diagrams without the guidance from a sighted person and are therefore reluctant to try it. Hence, expressing pictorial thinking for blind users through computers is limited. Most BVI students and practitioners are in the habit of using tactile maps to recognise highlight-raised line art or objects [Takagi, 2009]. However there are limitations to the information that tactile graphics can convey. Since Bach-y-Rita [Bach-y-Rita, 2004] presented the idea of tactile-vision sensory substitution in 1969, similar technology applications have seen rapid growth. From tactile-vision perception and understanding to voice-vision substitution, this has been incorporated in various ways to help BVI people in their daily lives as well as education and careers. Even though tactile images and 3D printing exists, this technology needs further improvement for complex and dynamic art production according to Williams [Williams et al., 2014].

Early versions of digital drawing systems for blind people can be viewed as modality driven (closely coupled with input output modes) and support worker dependent with tight coupling of the user and user-interface components. Most systems previously designed for blind drawing [Kamel and Landay, 2002], [Petrie et al., 2002], [Blenkhorn and Evans, 1998], [Calder et al., 2007], extract information and display it in objects and associations with sequential style communication using built-in command buttons. According to Hersh [Hersh et al., 2008] a drawing should enable the user to memorise a floor plan, be easy to navigate, be able to relocate and backtrack to the original point, and memorise the object arrangement. This minimises the processing of information and the mitigates redundancy effect (repetition of content in different formats), making it easy to learn and use, and take less time to complete a task. The challenge therefore is to develop technologies that are effective and easy to use in producing graphics for the different abilities of BVI users.

Described now are several well-known digital drawing systems including their current offerings and problems in a nutshell. The first one is System KEVIN [Blenkhorn and Evans, 1998]. This was developed to read, edit and create the information of data flow diagrams in a N^2 chart matrix style grid with a tactile layer. This system however, fails to give information on the diagram layout. System PLUMP [Calder et al., 2007] uses Heaps algorithms to store data where information is fed in a sequential manner. The user can then operate algorithms using synthesised speech and observe the data links of a data structure. The system is operated using a fixed set of interface commands by the user, however this system does not keep track of layout information or the artwork created. System IC2D [Kamel and Landay, 2002] does keep track of the layout information and provides screen navigation information but it still has a limited fixed set of system commands to operate it. System AHEAD [Rassmus-Grohn et al., 2007] uses an external device other than the computer to haptically draw images with the aid of a human assistant. The drawback of this system is that it takes a long time to draw anything with it, and as it is haptic modality bound it needs the support of another person. DocExplorer [Ishihara et al., 2006] system uses visual analysis for creating diagram structures. It creates metadata that describes the relationships of objects and converts them into a readable tree structure for blind people. However, DocExplorer is not based on a graphics creation technique, rather a graphics analysis and conversion method that does not retain the layout information. BPLOT3 [Fujiyoshi et al., 2014] uses a command user interface (CUI) that requires sighted people to work with blind people. It has a tablet user interface for blind people to produce haptic drawings. The CUI uses a grammar-controlled language that relies on coordinates to track the graphs locations. The drawback of this system is that the user must work out a specific point on the screen from many available x, y coordinates in the absence of object labels. In reality the command user interface is mainly designed for sighted users. The pin matrix system [Bornschein et al., 2018] is another haptic system that uses external drawing kits. It has different mode settings for drawing objects which can take BVI users much longer time to use than grammar controlled systems.

The above digital systems with haptic modality require another person's support and external devices that ultimately means that it can take BVI users a much longer time to complete any drawing tasks. Other systems that retain information on the art/diagram but not the layout information. Even though the meaning is conveyed, the layout information is lost, and users are not involved in 2D space exploration when producing artwork. This means it is not possible to determine how the unused 2D drawing space can be exploited. These systems do not directly introduce modality free drawing techniques for blind users, nor a system design that is adaptable for any number of modalities or technologies.

1.1.2 What is Missing?

Although there are different drawing systems available for BVI users to address a variety of drawing needs, there is still no formal approach which should potentially facilitate research, in the space of

CADB.

The above mentioned modality driven systems define the space for CADB. As yet the question of multimodal, interoperable, technology which promotes self-reliance is not addressed as past systems only limit themselves to their input and output mechanisms. In addition, the computer aided drawing systems designed for blind people are censured for their inability to serve their purpose at educational establishments and work environments. CADB systems have been criticised by [Bornschein and Weber, 2017] for the tightly coupled approach often taken towards their interaction and the absence of assistive, usability features. A lack of widely accepted non-abstract guidelines and models for assistive drawing tools for BVI users has directed the research in this thesis to inquire into the theoretical basis for producing effective design guidelines .

Although command-line text to diagram conversion has been an experiment in the past, there is still no agreement regarding the guidelines that are unique to CADB apart from an experiment outcome presented by the leading researcher of computer-aided drawing tools [Bornschein et al., 2018]. According to him, the majority of tools are lacking in drawing support and error correction. Therefore the focus of the research presented in this thesis is on modelling a computer-aided drawing technique with an effective support mechanism based on a set of defined guidelines.

There are many more details which need to be understood about the representation of drawing for blind people. This investigation is not only limited to recognising reoccurring common errors presented in previous drawing systems but also the essential characteristics needed to improve efficiency and usability of navigation and graphics manipulation techniques for BVI users. The context of the mental images by blind users is explored throughout. With the understanding gained from the past and current-state-of-the-art research work, a design methodology for a formal specification is proposed in this thesis. The developed system is based on a formal specification which is referred to here as SETUP09, which constitutes the main contribution of this research work.

In this thesis, an approach is proposed with a set of guidelines that aims to address the issue of how to formally model a CADB which can effectively support user interaction and for drawing artwork. The aim of the proposed formal approach is to introduce a formal specification that can work with many modalities and thereby increase the interactive use of a drawing support system using a new navigation and art creation technique based on command language interaction. It will have a well-defined syntax, semantics, and rules and grammar. The formal approach is developed to analyse drawing problems, synthesise solutions, constructs language sentences, resolves language sentences and supports the use of language by a tool. Such a system should be useful in many applications and areas where BVI users have different drawing goals. The development of the guidelines in the formal approach will be tailored and refined according to user behaviour that will be obtained through experimental observations. In later chapters in this thesis, it will be evident from experimental studies that the proposed system facilitates easy and quick screen navigation by BVI users.

The main contribution of this thesis is the development of a novel formal model-based approach for the design and development of computer-aided drawing system that can specify images without recourse to specific solutions. The approach is intended to identify and formalise the properties of usability, space and shape properties for drawing different types of 2D images. The proposed approach is intended to overcome issues plagued by existing drawing tools which are restricted to follow a certain modality or generation of a specific type of art/drawing. Although assistive technology is the common factor between the research presented here and existing drawing tools however the focus of this research is on developing a formal approach and a set of guidelines guided by experimental evaluation rather than the testing/evaluating of alternative modalities.

The proposed technique is visualised against existing techniques in the picture below.

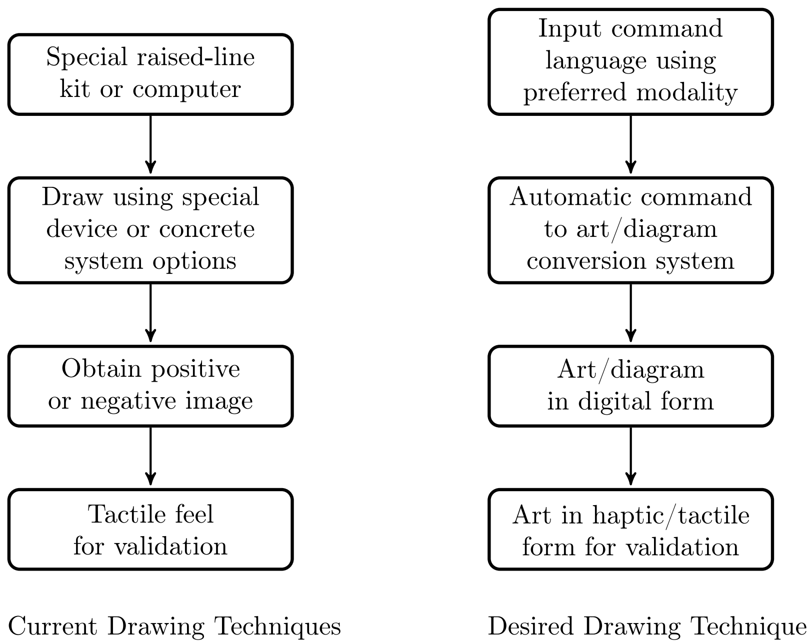


Figure 1.1: Motivation for a Formal Approach to a Computer Aided Command Driven Drawing System for Blind Users

Figure 1.1 shows two flowcharts for a command driven drawing system. The figure has two columns. The first flowchart depicts the general process of current techniques. The second flowchart depicts the process of the flow for the desired technique. In the current techniques, the sequence of events is as follows: the BVI user draws an image using a manual drawing kit and associated drawing tools, obtains a positive or negative image and then checks the image for accuracy by touch only. Some users may also use concrete systems to draw images. The desired technique operates with command language using a preferred modality, where computer commands are converted to diagrams, which are then displayed in digital format and presented to the user in tactile form.

As illustrated in Figure 1.1 the desired approach differs in the way blind users interact with the system by choosing a preferred modality. The technique of navigation and 2D shapes creation and grouping takes a new approach to accommodate effective art production. All user actions are translated through a command language translator. The surveys presented in Chapter 4 confirms that pace, mode, and level of system usage significantly improve interaction with a drawing system.

The user interactions of some existing drawing systems [Blenkhorn and Evans, 1998], [Calder et al., 2007] are determined by the system designer's decision. Some of these design decisions are one-sided and cannot be altered. It can be said that the designer owns the creation. Thus, such a system lacks personalised features. Although users can produce artwork, but screen navigation is not allowed. Moreover, system help alteration and storage for future use are not promote.

The research presented in this thesis is motivated by merely building a systems with a fixed set of functionalities that do not accommodate the drawing artwork and create diagram with greater autonomy. The research in this thesis proposes a way to overcome issues of prior assistive technologies for drawing with a novel formal approach that defines a command language to implement and extend the benefits of drawing systems for blind people according to their specific drawing goals. The list below summarises the main motivations for the research.

1. There is no universally accepted specification for screen navigation graphic manipulation for BVI users.

2. Most systems that were made in the past assume that users are required to produce one particular type of graphics or diagram (e.g.data flow diagram, graphs etc.) and do not enable art production in general [Blenkhorn and Evans, 1998], [Calder et al., 2007].

3. Should there be system accommodation for different levels blindness?

4. There is no existing abstract approach to support assistive technology with alternative input and output mechanisms.

The research investigation has been conducted because of the lack of an abstract formal approach for a drawing tool that characterises the properties that enable principles of design decisions. It will contribute to the space of possibilities, which other assistive technologists will able to test.

The thesis explains what precise abstract features can be made a part of such a drawing system, so that issues related to technologies, are not restricted by current technological limitations. Such a featurization paves the way towards overcoming specific issues in technology-driven systems, that will also rise to specific research areas in human-computer interaction and interface design.

The concepts formed in this thesis address, not only to an understanding of what drawing systems mean for blind people, but how some essential features might be beneficial to increase their effectiveness in drawing tasks. Even though there is a lack of very large scale, empirical and long-term studies into the effectiveness of such systems, guidelines introduced in this study are backed up by experimental evaluation that are more appropriate at this stage of formalisation. The expressiveness of the model introduced enables the space of possibilities of such system design.

This research also addresses the cognitive overload of screen navigation and drawing tasks. Less cognitive load means, less effort in screen navigation and more time spent on actual drawing tasks. Some of the difficulties that fall in to this category include completing a simple task quickly, navigating on the screen in relation to start and end points, backtracking to the original point or remembering the start point, relating the size of the art on the screen to other images on the screen, and re-using previously created art that employs repeated drawing. Although, Swell [Sweller, 1994] mentioned that too many computer aided drawing features may add a high level of cognitive overload.

1.1.3 Initial Challenges

The research needs a strong basis to prove that a computer-aided, command driven drawing system for blind people is an effective and efficient drawing mechanism. For the research to provide this evidence, the following questions need to be addressed, which are discussed fully in Chapter 2:

- What are the main concerns of computer usage among BVI users?
- What are the different uses of assistive technology and their impact?
- What are the different design approaches to assistive technology?

1.1.4 Derived Challenges

The questions below were derived as a result of the initial facts finding that was conducted in the research which are discussed in Chapter 3:

- What are the characteristics of a drawing system for blind people?
- What sort of object layout a blind person would easily conceive and convey?
- What sort of a navigation system would easily build a sense of object layout?
- What sort of a space arrangement could allow easy navigation?
- What usability features are best suitable to design CADB?

1.2 Technical Aims of Research

This section addresses the central question of the research and the associated problems, followed by the approach taken to solve these problems and the finally technical development.

1.2.1 The Central Question

The central question which this dissertation aims to answer is: What is the scope of a computer-aided command driven drawing technique for blind users and what interaction could be made available to blind users?

A formal approach is taken in presenting the solution that builds upon the theory and practise of drawing systems for blind users. The end goal is to develop a formal approach that enables abstract characterisation using such a system.

1.2.2 Derived Problems

The following derived problem are presented in answering the central question.

The primary research questions

1. What constitutes a formal drawing language for the blind user?
2. What is the best way to model and implement a drawing language?
3. How can such a model be evaluated with blind people and other against other existing systems?

1.2.3 Research Limitations

The responses to these questions evoke the following limitations:

1. The main approach has been to identify and implement a formal approach, which may be difficult to generalise in terms of its grammar.
2. Often it is difficult to measure the distinct, independent effect of command driven language with diverse user ability and disability.
3. The design decisions made in the process of implementation are driven by number of factors such as what available digital technology solutions, and what modality functions they support, which are difficult to personalise.
4. Observations being analysed from the researcher's point of view and biased answers may led to biased results that impact the analysis and interpretation of results.

1.2.4 Research Goals

The research aims to respond to these limitations via the following goals:

1. Identify, classify and synthesis problem solution by design/grammar rules.
2. Use formal modelling techniques to express problem characterisations and/or problem solution with defined syntax and semantics to make guidelines clear and therefore revisable.
3. Model a command-to-diagram conversion language for blind users, in the way the language can support multiple modalities if needed.
4. Develop a proof system based on the model that can be used a basis to evaluate the syntactically well-formed sentence of the language.
5. Evaluate of the navigation and art creation technique by experimenting with many different subsets of users (blind, partially sighted, late blind and sighted users).

1.3 Research Contribution

The main contribution is the formal approach to a computer aided drawing technique; the importance of such a technique is that drawing systems for blind people can be studied with greater conceptual clarity than other digital drawing experimentation. Another attraction is that the proposed solution is presented at a level of abstraction above concrete technologies, that enable us to explore the specific issues relating to drawing by blind users. The contribution made in this thesis is unique, as far as the author is aware, in that it formally models user-interaction to bring user satisfaction. Furthermore, the expressive user language introduced in this thesis and the experimental evaluation trailed by different end users may help other researchers in the future to innovate and advance the technology.

System architecture and user interactions are essential in developing a concrete system. This thesis contributes towards this realisation by presenting system implementation procedure and an interpreter based on the proposed formal approach.

1.3.1 The Research Impact

- Contributes to enhancing the imaginative thinking of blind people through 2D drawing.
- Provides a framework to implement a formal specification for graphics manipulation.
- Provides a foundation for a formal specification of screen navigation and art creation language that can be used by system designers.
- The proposed technique and prototype method can be embedded in curriculum design, such as diagrammatic drawing for blind students.
- The proposed navigation techniques can be implemented as an alternative navigation mechanism by sighted individuals in the absences of a mouse or other input device.
- The proposed navigation techniques can be implemented as an alternative navigation mechanism to navigate hidden surfaces.
- The proposed navigation techniques can be implemented as an alternative navigation mechanism by game designers.

1.4 Overview of Research Methodology

Several methodologies such as the formal approach of computer science and the qualitative/quantitative analysis approach of social science were deemed appropriate for this study in order to consolidate the claims in this thesis.

A formal method in software development is a kind of mathematical technique for writing specifications, development, and verification [Chase, 2002][Cook, 1992]. It is used when mathematical analysis contributes to the reliability and effectiveness of the design. There are many formal methods using a broad range of software techniques for tackling various problems: formal languages, logic calculi and automata theory are some of them. In this thesis formal language is used by introducing formal sets of abstract grammar rules. The rules are analysed, designed using empirical research and fine-tuned using experimental methodologies in this thesis. A language was introduced to create a CAD systems for blind people.

Users' and system designers' views are subjective in language introduction, thus an objective approach for language creation was followed. Writing a formal language is the only valid method in computer science to introduce a language that leads to parse and compiler. Some of the advantages of formal language are: (i) a grammar model that is conducive for creating a language based on rules and logic, (ii) it can process semantics, (iii) the ability to make quick changes to the model, (iv) it has an alternative structural design based on theory, and (v) it allows better control for system development and exploration of different design pathways. Formal language approach was taken in the thesis because it is written using a formal grammar, namely context-free grammar, and a set of production rules that describes all possible strings in a computer language. The popular notation Bakus-Naur form was used to express such a grammar model.

A user-involved, software development methodology was used as the building methodology in Chapter 5. Even though HCI (Human-Computer Interaction) methodology is highly popular for system development, it is not used for grammar model formation because language implementation in computer science is mainly accomplished by the formal method approach, not a user-centred approach. Hence, this thesis introduces a formal approach that is not an HCI approach.

The rationale for selecting the method selection and design decisions are presented throughout the thesis. Special attention was given to the representation of end users in accessibility research, discussed in Chapter 2. Design rationales for experimental evaluations, blind people's cognition, and user survey are presented in Chapter 3. Users were involved during grammar model, system design and evaluation stages in Chapters 5, 6 and 8. System evaluation is based on ISO standard, [ISO, 2011] using criteria of effectiveness, efficiency and user satisfaction as the main measures.

The findings of empirical data resulting from this study proposes a set of design guidelines to establish a formal specification language for computer-aided drawing for blind people which is presented in Chapter 9.

1.4.1 Why is a Formal Approach Used to Model SETUP09?

The research here is targeted at some of the difficulties encountered in existing assistive technologies for creating artwork by BVI users. Language development based on the formal approach was deemed suitable for the proposed technology. This is because the formal approach is consistent and expandable that facilitates production of abstract artwork and navigation that allows the user to envisage where the objects are located on the screen. These attributes contribute toward overcoming the limitation of drawing by BVI users due to the absence of visual modality; it helps them to draw scientific diagrams during classroom activities; it accommodates graphics presentation and visualisation of imaginative thinking such as art in general; it contributes to the image processing; and sighted people could also benefit using the techniques when creating images on non-visual surfaces.

Another reason to produce a grammar-controlled navigation and object tracking language is in the gaming technology for blind game players. Game technologies such as Interactive Fiction (IF)[1] are purely based on text and command languages allowing user input tracking, voice feedback and sonification for interaction and a screen compass for object search. IF gaming technology has been accessible to blind game players for decades. Blind game players are fond of interactive fiction games/-text adventure games where players use text-based language to control the environment[2]. This study, through the development of the proposed SETUP09 system, introduces a compass-based location tracking approach together with interactive fiction gaming communication styles, to develop the user's concept of drawing by using SETUP09 system. This method of drawing does not need the help of a support worker, does not take an inordinate amount of time to complete a task, and is not expensive to use. The method is intuitive interactive and easy to learn, and has provision for error detection according to BVI participant testing, discussed in later chapters.The approach taken here to navigate on the screen is a compass-based interactive fiction style. Several studies [Kamel and Landay, 2002]; [Zhu and Feng, 2010] have also introduced grids for easy navigation similar to compass-based navigation and finding places on maps accurately in an unknown environment that support the design decision of the navigation technique proposed by this research.

In summary, the motivation for a command driven grammar controlled language is derived from the gaming behaviour of blind users and the research outcome is for a platform-independent method to construct an image by a text-based interface language using a formal approach. A language for the blind user is modelled using a formal approach that has been defined specifically for this purpose. The model incapsulates the space, shape and usability functionalities within the proposed language specification. The work consists of a model that gives a greater understanding of the abstract level of features required. It also gives an understanding of how the formal approach can aid the design and development of such a system. Furthermore, the implementation of the introduced language uses concrete syntax. Therefore, the model is implemented using a software programme with an interface to display 2D drawings and a parser that resolves the syntax and semantics of a grammar-controlled user language.

1.5 Thesis Structure

This document is structured as follows.

1. **Chapter 2** opens with an explanation of blindness, vision and the needs of blind people. It then explores the various existing assistive technologies and design issues. The chapter also elaborates

[1] IF games are computer software environment, where gamers use text commands to manipulate objects of the game, The natural language processing is an essential part of interactive fiction development.

[2] https://www.game-accessibility.com/documentation/visually-impaired-gamers-where-to-go-what-to-play/

on HCI system design and evaluation approach, and formal specification system design and evaluation approach to build a solution.

2. **Chapter 3** explores modelling issues of CAD for blind people by exploring three main areas; (i) current technologies and associated issues and theories of blind drawing; (ii) current navigation technologies and associated issues and navigation models for blind people; and (iii) issues of usability of CADB systems and associated theories and recommendations. The chapter highlights the main building blocks that form the basis of a formal specification for the proposed solution for navigation and drawing; which consists of usability language, spatial language and shape language. It presents a classification diagram that visualises the foundation of the language.

3. **Chapter 4** presents the design choices of the SETUP09 model concepts such as location, points, and sizes. It then formally defines the language that consists of space, shape and usability, and ends by introducing a context-free grammar model for CADB.

4. **Chapter 5** describes a formal language implementation with front-end interpreter, parser and semantic validation, with examples of user commands and system output. This chapter also discusses the front-end concrete system development.

5. **Chapter 6** contains empirical evaluations of the proposed model; Study 01 tests the navigation with blind participants and Study 02 tests the drawing commands with different subsets of users (blind, partially sighted and sighted). The chapter opens with motivation for the studies, design rationale and evaluation metrics. Each study is followed by the stated hypothesis, method, procedure, findings, limitation and conclusion.

6. **Chapter 7** presents the revised SETUP09 model and recommendation for future work based on empirical findings and changes made to the specification.

7. **Chapter 8** compares the digital drawing tool SETUP09 with analogue drawing methods. The effectiveness of using German film paper and a digital text-to-diagram conversion technique are compared in Study 1. This chapter also compares the SETUP09 digital tool with another digital tool (IC2D system) in Study 2.

8. **Chapter 9** provides a refined set of design guidelines when designing computer aided drawing tools for blind people. Experimental studies with end users testify the effectiveness of the proposed drawing guidelines. The proposed guidelines should enable software designers and developers to improve and innovate future assistive technology related 2D drawing tools.

9. **Chapter 10** concludes this thesis by outlining the research contributions made, possible future developments and directions for extending this research. It lists the outcomes that are drawn from the approach used to design and develop a computer-aided command driven drawing system.

Figure 1.2 shows the road map of the thesis and outcomes that led the direction of the research. In particular, Chapters 2 and 3 findings of AT design and CAD for blind modelling issues led to the need for a formal approach in the development of the proposed SETUP09; The formal approach in Chapter 4 led to a need to implement a language; Specification development in Chapter 5 led to a need to test the model; In chapter 6 and 8, findings affirm efficient navigation and drawing of SETUP09 and high accuracy compared to previously published results and other systems; Chapter 7 introduces model revision and finally in Chapter 9 the research reveals a set of CAD design guidelines for blind people.

1.5.1 Road Map of the Thesis Structure

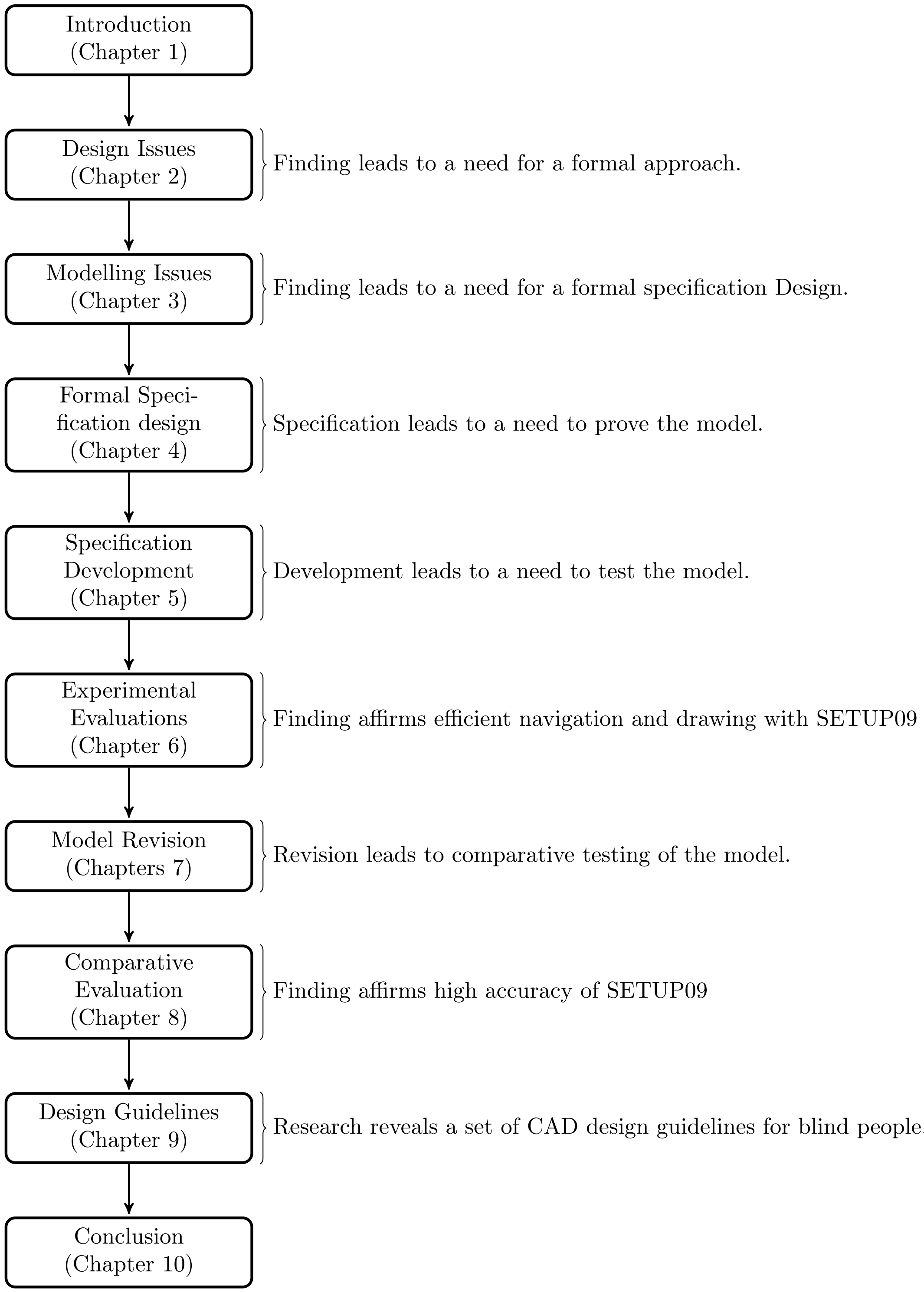


Figure 1.2: Road Map of the Thesis Structure

Chapter 2

Assistive Technology and Design Issues

To place this thesis in its proper context, this chapter provides a detailed analysis of the main lines along which the research was conducted into assistive technology including design issues and development approaches. This analysis highlights the background study, the first stage of problem identification, as well as the related areas that need investigation. The previous chapter laid out an overview of the research whereas this chapter discusses the initial questions of the research. What is visual impairment? What are the different uses of assistive technology and their impact? What are the different design issues of assistive technology? Studies are presented of blindness and its broader recognition, a different subset of blindness and their needs, recommendations of RNIB, and services of NHS in the UK. Examined in detail are current recommended and well-known assistive technologies, and their pros and cons. These assistive technologies are addressed, specifically to assist with the development of CADB model for BVI people. Consideration of the current technological issues clearly reveals the need for the development of new assistive technology design. It is recognised that not only a set of design considerations determines the success factor of a good piece of technology but also the approach to the design, development, and evaluation. The chapter therefore answers further questions, What are the different design approaches to assistive technology? What is HCI approach? What is formal approach? How is a formal language being designed, modelled and evaluated? Hence, the chapter ends with an exploration into design and evaluation of HCI approach with examples of technologies for BVI people and another exploration into system design and evaluation by the means of formal approach modelling of user interaction, grammar, and user language with examples technologies, and discusses its suitability for a CADB design. This then led to answering more specific questions on the formal approach: What constitutes of a formal language for CAD for blind people? the primary motivation that is presented in Chapter 3.

2.1 Visual Impairment

The World Health Organization [WHO, 2018] describes “Disability” as the limitation of activity and restriction of participation due to a person’s particular health situation, personal limited factor and environment, as opposed to “functioning” which means having full ability without limited of participation and is a positive reference to their health situation. According to the Center for Disease Control and Prevention [CDC, 2109], blindness is a severe vision impairment, that is not correctable by contact lenses or other medical interventions. It makes a person unable to perform their day to day activities or even people with low vision experience difficulty in their day to day activity. However, any human being can experience disability due to a specific health condition at some point in their life. For example a person using a crutch or in a wheelchair experiences the same inability to access a multi-storey building. In other words, good assistive tools and systems should not limit a person’s activity because of their health situation or a specific condition.

Disability is also defined as what makes someone disabled is not their medical condition, but the attitudes and cultures of society according to the social model of disability [Mental-Health-Foundation, 2020]. "Impairments" prevent individuals from doing day-to-day activities, whereas "disability" is an additional unfavourable condition used as a label by people that treat "impairments" as abnormal thus excluding people from complete participation. For example, prejudice, labelling, ignorance, lack of financial independence, overprotectiveness, and not having information in an accessible format are some of the ways impaired people become “disabled” . The social model definition of disability tells us that target usability features are not an add-on to product development but an essential part of day-to-day inclusive systems development.

Nearly 1.3 billion people live with some form of visual impairment, 39 million are blind and 246 million have low vision, plus an estimated 19 million children are visually impaired, according to the World Health Organisation[WHO, 2018]. Blindness and visual impairment are caused by many reasons or factors: absence of certain genes, deficiencies, or infections. Blindness can mean that a person has reduced vision or no vision. Blindness of some late blind individuals is caused by cataracts which can be removed, and while some people gain sight and develop vision over time, in other cases it is difficult to get normal vision back or they revert to their original state, and they find it difficult to make much functional use of vision [Editor, The National Federation of the Blind, 2002]. Therefore, it is a significant challenge to develop one software that accommodates the differing quality of life of various subsets of BVI individuals by the means of assistive technology.

There are many legislative acts and standards to improve information access and communication technology such as UK’s, Disability Discrimination Act 1995 (DDA) [Crown, 2019] and Equality Act 2010 [OGL, 2019], that prohibit discrimination due to disability and promote equal access to education, employment and other social services.This research is particularly targeted at people who are fully blind. However many recommendation and finding are suggested for visually impaired computer users.

2.1.1 How Vision Works

According to [Editor of Texas School for the Blind and Visually Impaired, 2019] the perfect vision functions through a combination of the optical system (eye, eye muscles, optics nerve) and the perceptual system (brain). When the optical system gathers and sends information to the brain, it then identifies, classifies and stores it in the memory. The brain then processes visual images to understand the world which are eventually used for human reasoning (to manipulate mental images) to initiate actions such as walking, speaking, looking. A number of conditions must be met for image processing, such as alignment of the eyes, correct adjustment of pupil size for the lighting, the shape of the eyeball and cornea, lens adjustment, a functional retina, and the capability of the optical nerve in transmitting an image to the visual cortex in the brain that is responsible for image processing. Brain malfunctioning, such as an inability to monitor and adjust eye functions, can also cause blindness.

Table 2.1 gives a breakdown of visual impairment by the the International Council of Ophthalmology (2002) and the recommendations of the WHO consultation on "Development of Standards for Characterisation of Vision Loss and Visual Functioning" (Sept 2003). It classifies mild to severe blindness by giving the visual acuity related to each category. For example, the category of "Blindness" could mean no light perception or one-sixtieth of worse visual acuity. The visual acuity means the quotient of the distance at which an average person sees an object at which the distance at the same object is seen by a visually impaired person. For example, 20/200 means that where a sighted individual can see an object 200 metres away, a visually impaired individual can only see that same object at a distance of 20 metres. A person be described as visually impaired or blind if the visual field is no greater than 10° around the central point whereas a sighted person is able to see 180° with their visual field [Eyehelp, 2019].

Category	**visual acuity worse than:**	**visual acuity equal to or better than:**
0 Mild or no visual impairment		6/18, 3/10 (0.3), 20/70
1 Moderate visual impairment	6/18, 3/10 (0.3), 20/7	6/60, 1/10 (0.1), 20/200
2 Severe visual impairment	6/60, 1/10 (0.1), 20/200	3/60, 1/20 (0.05), 20/400
3 Blindness	3/60, 1/20 (0.05), 20/400	1/60*, 1/50 (0.02), 5/300 (20/1200)
4 Blindness	1/60*, 1/50 (0.02), 5/300 (20/1200)	Light perception
5 Blindness	No light perception	
9	Undetermined or unspecified	
	* or counts fingers (CF) at 1 metre.	

Table 2.1: Visual Impairment Classification: Presenting Distance Visual Acuity, [WHO, 2015]

Human reasoning works well with mental images that fundamentally comes from the optical system and perceptual system in sighted individuals. But a visually impaired person may have a healthy or heightened perceptual system as a result of information absorption from other senses even if their optical system is presented with certain conditions. This phenomenon is further explained in Chapter

3, under perceptual models of blind people. Table 2.1 also informs us that blindness can take many forms, from mild to severe and total blindness, hence the software design should consider different interaction needs of blind users to improve social inclusion.

2.1.2 The Needs of BVI People

The National Health Service in the UK provides some services and facilities for those who obtain a Certificate of Vision Impairment and register as a visually impaired person. These include modifications to the home and access to help and special services, some examples of which follow [NHS, 2019] .

- Changes to the home environment: big-button telephones or computer keyboards, screen display software and text readers, wearable alarms, and powerful light bulbs for the partially sighted.
- Facilities for reading and writing: magnifying devices, large print publications, e-readers and changes to computer settings, talking newspapers and magazines, Braille, talking book services, voice recognition facilities.
- Help in obtaining Braille publications, a long cane to aid getting around independently, and regular sight tests.
- Guidance on employment and access to work schema.
- Guide dogs, GPS and driving advice can be provided by the Royal National Institute for Blind People, UK.

RNIB and [SSMR, 2009] published a report on Understanding the Needs of Blind and Partially Sighted People, their experiences, perspectives, and expectations. It identifies some key areas where support is needed. They are as follows: help to obtain employment, a one-stop resource centre to access information, better social inclusion, independent mobility, information and clarification on living allowance, signposting towards extra support and resources, information on different support organisations, transport and mobility training, assistance with household tasks, counselling and emotional support, availability of specialist equipment at lower cost, specialised software to read websites, support for those considering starting a business, access to holidays, support to play sport, activities with sighted individuals to enhance the perception of disability, support with hospital visiting, and assistance for parents with young children to deal with practical issues. Further suggestions included more audio descriptions in GP surgeries and cinemas, an agency to help young people at home, support for carrying out small tasks such as changing light bulbs, better support of the availability to go through Internet access, awareness raising and training for employers.

The RNIB recommendations and NHS services now promote different forms of tactile support and speech support features for BVI people to obtain information and live independently. Some recommendations are to enlarge text sizes to make reading easier for partially sighted people, and other times it facilitates with screen reading for independent digital reading and writing.

In summary, the definition and different forms of blindness varies according to which organisations you are referring to, i.e. WHO, CDC, and NHS. It is apparent that various measures need to be taken for different subsets of blind users when designing software. This necessitates an understanding of how blind people process information perceptually. The following section further elaborates on current types of speech and tactile technologies and their practicality in day-to-day life.

2.2 Use of Assistive Technology

It has been identified that the visual cortex is not purely visual [Banf et al., 2016]. Different senses influence each other. When one sense is weaker you seek the help of a stronger sense. Most blind people have some degree of useful vision, which could be considered in designing software. The distinctive aspect of sight is to obtain an overview of a situation or see and perceive 3-dimensional objects, which could be still or moving. Therefore, sight with the combination of brain functionality is used to obtain, select, interpret, and organise information. This constantly changes based on changes in the external environment and in the body.

Speech, tactile and haptic-related technologies are used to assisted individuals with visual impairment. They could be any item, piece of equipment, or product that is used to increase, maintain, or improve the capabilities of BVI people with or without the use of multiple senses. Some of these commonly used assistive technology types are listed below in Table 2.2 with examples. The technologies are categorised into input means, output means, sensory substitution means, and correction means, and contain examples such as special keyboards, voice synthesisers, sensors, and context grammar corrections tools.

Type	**Scope**	**Examples of ACT means**
Control means	Human-Computer	special computer keyboards, Braille keyboards, input device view, special microphones, adapted API's and other
Output means	Computer-Human	braille display, braille printer, voice synthesiser, gestures, animator, adapted APIs
Sensory-substitution means	Human-Environment	voice, sensor, visual substitution or augmentation
Translation and correction means	Computer-Human, Human-Human, Human-Environment	Transformation tools of information, dictionaries, syntax rules, semantics, context grammar correction tools, tools for computer vision and speech recognition

Table 2.2: Types of Assistive Computer Technologies, [Davydov and Lozynska, 2016]

Other technology-based categorisation can be presented as: reading technology, speech technology, tactile technology and haptic technology. The above example are further elaborated in technology-based categorisations below.

2.2.1 Reading Technology

Jim Thatcher [Thatcher, 1994] developed the first screen reader in 1986 for the BVI community and since then screen reading software for BVI users has become commonplace. Screen readers can read any text on the interface using voice synthesisers or Braille outputs. According to WEBAIM [WebAIM, 2020] survey 2019 of preference of screen reader users' preference, NVDA (Non-Visual Desktop Access) [NV-Access, 2020] is gaining popularity over JAWS [Freedom Scientific, 2018] and is commonly used with Chrome browser [Google, 2020a]. Since 2009 JAWS had been the most common primary screen reader, however, the popularity of NVDA and VoiceOver usage is now increasing. WEBAIM's latest survey also records that VoiceOver [Apple, 2020] mobile screen readers are commonly used on mobile phones. Participants found that headings, heading levels, and landmarks are useful in reading digital pages, but also believed that some systems are not designed with accessibility guidelines due to a lack of awareness among developers. Partially sighted individuals can also see the benefit of products such as Dolphin's Supernova [Su et al., 1999] that has integrated magnifiers. The graph in Figure 2.1 shows the historical trends for primary screen reader usage taken from WebAIM.

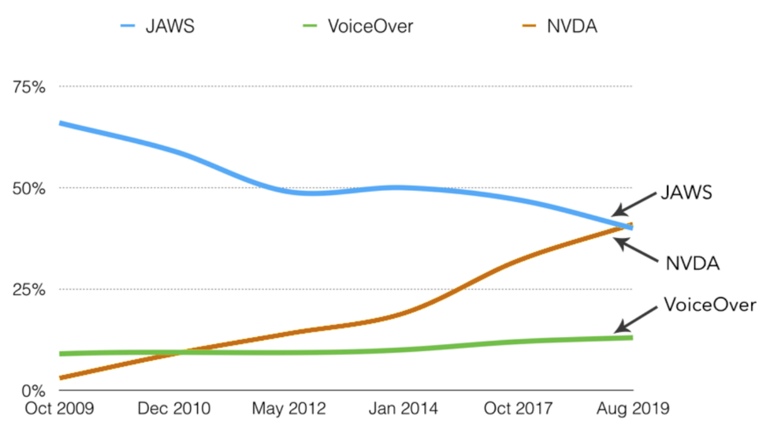


Figure 2.1: Historical trends for primary screen reader usage [WebAIM, 2020]

In summary, Screen-reading software is one of the key methods of computer-to-human information delivery and verification that allows BVI people to interact with computers. Such software is used by many blind people when writing documents on word processing applications, reading articles on the internet, writing emails, social networking, and computer programming according to the American Foundation for the Blind. However, screen reading facilities for BVI users has to be compatible with operating systems, applications, and Braille display. It is important that any new software builds upon work done by existing screen readers.

2.2.2 Speech Technology

The gramophone was the first device to record and reproduce sound in 1877. Since then there have been many developments, including speech synthesisers/TTS (Text To Speech) developed at Bell's

Lab that take material in the form of text or other signals , and imparts that received information to the listener using artificially generated speech.

The Microsoft Corporation released Windows Vista in 2007 with its first speech recognition software that takes human speech and decodes it into text. In 2011, Apple made a major breakthrough in mobile speech recognisers, with the introduction of Siri [Apple, 2019] that understood the meaning of users' speech and and undertook appropriate actions: such as sending emails, making calls, arranging meetings. The recent development in voice recognition is Echo, a voice controlled speaker introduced by Amazon in 2014 which is supported by the personal assistant service Alexa [Amazon, 2020], which is always connected to the Internet and provides information such as music, news, weather and much more. Products such as Google Home [Google, 2020b] are particularly helpful for the BVI community as they initiate operations by speech rather than depending on vision to navigate on the screen. Figure 2.2 is a comparison of speech recognition software comparison by software-testing-help.com [1] . Dragon Naturally Speaking for PC is at the top of the chart - although not free to use, it can be used personally or professionally for dictating homework, sending emails, web surfing, creating documents, and can be customised work. In second place is Google Now for Android devices which can be used to perform any function on the phone or device. Then comes Siri for iOS devices, Cortana for Windows and Amazon Lex for applications, although that is not free of charge.

Comparison Of The Best Speech Recognition Software

Speech Recognition Software	Best For	Platform	Free Trial	Price
Dragon Naturally Speaking (DRAGON NATURALLYSPEAKING)	Overall dictation and voice recognition	Windows OS	No	Dragon- Home is $150, Professional Individual is $300, Legal Individual is $500.
Google Now	Android Mobile Devices	Android & iOS devices.	-	Free
Siri	iOS devices	iOS devices.	-	Free
Cortana (Cortana)	Windows devices.	Windows 10, iOS, Android, and Windows phone devices	-	Free
Amazon Lex (Amazon Lex)	Creating Chatbot.	Used in the applications.	No	Based on the no. of speech requests processed.

Figure 2.2: Comparison Of The Best Speech Recognition Software, www.softwaretestinghelp.com

[1] https://www.softwaretestinghelp.com/voice-recognition-software/

It is evident from above that speech technologies enable not only BVI users but also people with a physical disability to dictate and compose documents completely hands-free using voice recognition technology. The systems work most efficiently with a Bluetooth headset and by muting the microphone when not in use. There is also a learning period associated with speech software which is necessary to achieve optimum efficiency when using it. Speech recognition is a promising input method for BVI users in assistive technology software creation. The mechanism behind screen navigation using speech technology and its inaccuracies and delays is discussed in Chapter 4 .

2.2.3 Tactile Technology

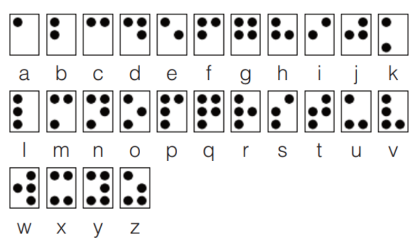


Figure 2.3: Braille [RNIB, 2018]

Figure 2.4: BrailleNote [Humanware, 2018]

Figure 2.3 shows Braille representations of the letters in the English alphabet and Figure 2.4 shows demonstrates the BrailleNote device introduced by HumanWare in 2018. Tactile technologies can also be categorised as haptic but are felt by fingers. Braille is a tactile reading and writing system used by BVI people who are unable to access printed materials. It uses 6 or 8 dot codes to symbolise regular and special characters of the printed alphabet, including mathematical, scientific, musical and computer notation, and foreign languages. Frenchman Louis Braille developed the model in the early 19th century. It has evolved since then and become successful as a relational method of raised dots 'read' by the fingertips and used solely by the BVI community.

BVI users can directly or indirectly communicate with computers and other electronic supporting devices using the Braille system. Leading market producer HumanWare have introduced many solutions for BVI users based on Braille technology that display letters in a tactile way. BrailleNote, Braille displays, Braille printers and talking book players are some examples. They all enable BVI users to input work on a computer and read digital text output by means of the display pin matrix.

Another widely accessible tactile method of communication is a swell paper, a special kind of paper upon which images and art can be printed or sketched and turned into tactile images or letters. It can be used with a marker pen, printer, or photocopier to create images. When the paper is subjected to thermal treatment the dark areas create raised relief lines. This technique enables users to feel the image on the printed paper and has been widely used by BVI individuals to recognise images, letters, and paper direction [RNIB, 2018].

In summary, tactile technologies enable blind users to feel information using special papers, digital interfaces, or pin-based devices as means of digital communication. Not everyone can afford HumanWare products, which are newer to the market than braille printers. However, raised line printed papers are commonly available and widely used by the BVI community and commonly found in special education establishments. This type of technology opens opportunities for BVI users to effectively participate in educational establishments and have a career on a level playing field with the sighted community.

2.2.4 Haptic technology

The word 'haptic' originates from the Greek word 'haptikos' pertaining to the sense of touch, and haptic technology refers to the use of force, vibration or motion by a computer user to input data or capture output. Users can create virtual objects and manipulate computer functions by means of sensory technology. The conversion of visual information into 2D drawings gets lost in translation with Braille where only text encoding is supported. But a tactile graphical user interface [Schiewe et al., 2009] enables tactile icons that can translate non-text dialogue, where touch screens such as electro-tactile displays provide the hand's touch receptors with a textured surface to better feel the 2D world of object placement and layout information rather than just using audio communication to relay it. Haptic gloves and haptic maps enable users to touch 3D objects in virtual space with vibrational feedback. Haptic shoes for blind people send vibrations to the user when they use Google's Lechal app to navigate to their destination - a vibration at the front of the shoe indicates go ahead, to the left means turn left, and so on [Medgizmo, 2018]. Parsee glasses enable blind people to explore the world as sighted people see, to help them recognise text, faces, shapes, colours and items using wireless communication and a mobile app Parsee [1] .

[1] http://www.parsee.org/

Paradigm / Features	Desktop haptics	Surface haptics	Wearable haptics
Computing platform			
Interface metahorp	Handheld stylus	Bare finger	Whole hand
Controlled avatar	Rigid tool	Virtual finger	Virtual hand
Typical device	 Phantom Premium	 T-Pad	 CyberGrasp

Figure 2.5: Evolution of computing haptic platforms, [WANG et al., 2019]

Figure 2.1 shows the evolution of haptic HCI model platforms in three stages: desktop, surface, and wearable haptics. The use of a stylus to handle motion, force or dimension is presented in desktop haptics, whereas fingertips on a touchscreen allowing a user to make gestures such as selecting, moving, rotating are presented in surface haptics. In wearable haptics, the user wears a haptic glove, glasses or shoes for diverse gestures such as grasping, moving, lifting and changing direction.

In summary, there are many more advantages to the use of haptic technologies than just a means to improve the experience of digital communication. Devices such as iPad enable individuals to learn new technologies in the field of haptics for an optimal experience even though haptic technology is in its infancy. But It is still difficult to create devices to reproduce a piece of visual information in a haptic form where the ideal interface combines both text and graphical information. According to Corrigan [Corrigan, 2015], targeted research and development funding to the various haptic initiatives should help with the production of advanced sensors for haptic devices.

Partially sighted people can also benefit from assistive devices that maximise their remaining vision and help with independent living, such as screen magnifiers or text simplification. Assistive technologies developed for people with no vision have also proven to be of benefit for people with low vision or even people with vision, for instance when there is a need to convert audio into text in a loud surrounding, for language translation, or navigation in unknown places. Even though some products are still at an experimental or developmental stage, such as OMoby [Zhong et al., 2013]; OCR (Optical Character Recognition) [ABBYY, 2019]; RIA [Crandall et al., 2001]; RFID (Radio-frequency identification) [Saaid et al., 2009], their features have helped the BVI community and contributed to new innovations.

This section presented some examples of assistive technology products, their benefits, and areas for improvement such as speech, reading, tactile and haptic technology. These technologies either assist information conversion into another modality or facilitate the perception of information as it exists without changing its modalities. The features of existing haptic technologies can be experimented with in designing a new assistive products, i.e.: free screen readers, speech technology, braille input methods and braille printers. These are some of the findings focused towards the design decisions that will be discussed in Chapter 4. The product design for BVI communities needs special attention with regard to the ways in which BVI people operate, perceive and model information. The usability and learnability attributes of software are key measures when designing assistive technologies. Design considerations and issues related to blind user's computer needs require particular attention in order to design systems with high usability, discussed in the next section.

2.2.5 Issues and Design Considerations

Some of the software design considerations of assistive technology are listed below.

- Cognitive load theory about learning difficulty and instructional design, [Sweller, 1994] suggests that effective instructional material promotes learning by directing cognitive resources towards activities that are relevant to learning rather than to processes that are adjunct to learning. The main problem, which occurs for blind learners using a screen reader, may be the large amount of information that has to be stored in the working memory to navigate to a webpage. The conceptual framework identifies that learning consists of accessing, using, and doing. This is pointed out by the fact that, in an experiment conducted on evaluating the quality of learning experience by [Evans, 2009], blind learners only spent 66% of their time on 'doing' an activity whereas sighted learners used 90% of their time actually 'doing' an activity. More time spent on accessing and using the software tool makes learning harder for blind users, therefore mental effort is not equal to time spent on a task. Minimising the processing load, disorientation and redundancy effect are important quality factors for better engagement or learning. Poor design of learning objects, lack of key information, and non-existent or difficult navigation are some of the barriers to learning in general. However, BVI students enjoy learning as much sighted learners, even though BVI students are dependent on others in some educational tasks; hence, they access classroom assistance. Therefore, learning materials should be designed in a way that BVI individuals are not disadvantaged by graphical information or components that they cannot access. Multimodal human communications for an able-bodied person can use images, sign languages, text, gestures, oral speech, touch, mime, body language, and facial expression. However due to limited sight, only some of this could be considered, such as text and oral speech, for BVI individuals if other haptic technologies are not available.

- Units of speech can be divided into syllables, words, phrases and sentences, and there are rules for combining all of these. But sometimes it is sufficient to search for a set of relevant words as described by Hersh [Hersh et al., 2008]. She mentioned that "Speech recognition is likely to remain an intellectual frontier as long as we search for a deeper understanding of ourselves in general and intelligent behaviour". Therefore, if speech is the input medium, then the final product or output must be evaluated by its naturalness, multilingualism, speaker characteristics, computing power and also cost, coverage and labelling. Speech synthesisers could provide feedback to user actions and conformation when completing an action. However, both blind and sighted users are not in favour of complex technologies, therefore it is important that there are easily accessible technological solutions and that information is available in a number of different formats. Colours could also improve human performance in searching for targets, according to research [Hersh et al., 2008] carried out on partially sighted computer users. A yellow highlighted background enhances search time performances. Dual output modes and low picture density also revealed better performance. People with low vision can benefit from screen magnifiers and from colour change and visual clues. Not only BVI people but also the older generation could benefit from assistive technology products.

- Screen readers produce voice outputs that come through a speech synthesiser or Braille outputs. When a pop-up dialog appears, the screen reader is actually aware that it is on the screen, as it is inserted into the off-screen model (Windows) or via API calls (Mac), however the screen reader does not always alter the behaviour of the user based on the object with focus. For example: the keystroke to read the current line will (almost always) do so. It is apparent in magnifiers. Also, a magnifier as higher magnification does not provide undistorted clear text. Predictors are helpful in predicting the word or phrase as the user is typing but the predictive text can be erroneous and off message. User typed word or phrase could be installed as a part of word processing so as the user words, predicted word completions could appear. Also, Braille keyword characters are not always converted correctly to computer inputs/ASCII.

- Multiple frames and large numbers of links make the interface system difficult to extract information from. OLE (object linking and embedding) allows an application to both obtain information and control another application [Petrie et al., 2002]. Most assistive technology architecture uses similar mechanisms to interact with applications. Therefore, there is a need for a new assistive technology with an intelligent interface that makes greater use of contextual information to provide BVI users with more meaningful experience.

- Working memory capacity of an individual is another important factor when system messages are conveyed by voice. It is advised that messages be short and easily understood. According to cross-modality theory, when the amount of information is large, it is not suitable to present that information using the voice. Multimode information presentation has proven to be better than unimode information presentation on digital devices. Therefore according to Zhao, distribution of the information load across multiple channels is advantageous [Zhao et al., 2009]. However if materials are very simple then there would be less benefit from multiple modalities.

2.2.6 Conclusion

Issues associated with existing assistive technologies have been considered above. Design considerations of next generation of such technologies need to consider new standards and consumer legal requirements, which will be discussed in Chapter 4. The next section details different approaches to assistive technology design. There are reasons why some products fail to meet customer expectations while other products demonstrate longevity with high user satisfaction. Some products don't adapt to changing or unrealised customer needs, where other products show greater flexibility. The next section addresses the question of suitable design approaches to the development of AT and the formulation of a design specification that gives greater conceptual clarity and flexibility.

2.3 System Design and Evaluation using HCI Approach

This section describes the research and efforts made in the HCI approach of system design and development. It looks at the general principles of HCI system design, the process of design, and system evaluation criteria. Evidence is provided of specific BVI experimental evaluations and highlighted is the representation of users in accessibility research.

2.3.1 Overview

Some designers use their intuition and hope for the best, whereas others go with more logical choices to understand their users, such as taking account of what people are good at, finding out what methods are practically beneficial for users, thinking about the quality of user experience, listening to what users want, getting them involved, and testing the build with users during its design. Interactive design is "designing interactive products to support the way people communicate and interact in their everyday and working lives" as explained in [Preece et al., 2015]. One of the most promising areas of HCI design is the application of supporting and intensifying the sensing capability of people with disabilities, according to leading HCI researchers Jenny Preece and Helen Sharp [Preece et al., 2015]. The HCI perspective can offer the widest possible fulfilment in assisting BVI users with many assistive devices and technologies, as discussed earlier in this chapter, such as speech software and Braille. Figure 2.6 shows the development approach of HCI, where the user programme is written with a concrete syntax language based on a set of user requirements. The iterative process is not included in the image.

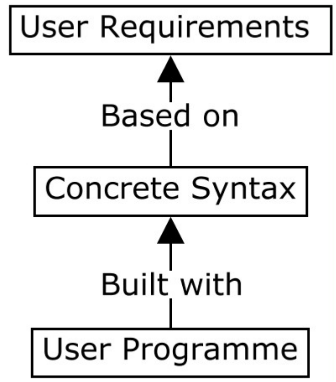


Figure 2.6: Development Approach of HCI

System design, interface design and feedback design are three important aspect of user experience design. The application interface can itself be designed with features such as alternative text, namely audio clues with sequential style information layouts for BVI individuals as discussed earlier. The HCI approach following system design should create a good fit with human interactions, computer interactions, and task completion. The designer should have a clear grasp of user obstacles to accomplish the tasks with a clear user scenario, usability considerations, and ease of use [1].

The interface is the way most users interact with the operating system of a computer/device. Therefore, interfaces should help users to complete tasks efficiently with appropriate feedback. The user interface is formed of two elements - computer to human communication, and human to computer communication. Graphical user interface, form-fill interface, menu interface, natural-language interface, and command-language interface are some of the commonly found digital interface types that cover human to computer communication and vice versa. Other modern interfaces such as touch-sensitive screens, speech recognition, and stylus-based tools are suitable for BVI users.

As discussed previously, mental models of BVI people work well with sequential style communication, therefore command-line interfaces can be a good fit for BVI users. Such command-line interfaces also have guidelines and user experience recommendations. CLI (Command-line interface) designers should always provide help for all the commands including sub-commands, use familiar names and have a clear naming philosophy for commands and sub-commands, and provide helpful, readable, user-friendly output and tab-completion (the typing can be completed with computer predictive commands) as suggested by Ubuntu [2] .

Feedback to the user is another essential part of interface design, where too much information at the wrong time is obstructive. Feedback, such as notification that the input is incorrect or correct, notification of an ongoing process, and acknowledgement of completed and non-completed requests, is useful in many contexts.

The process of interactive design starts with establishing requirements, designing alternatives, prototyping, and evaluating. The goal of the interactive design process is to achieve systems with certain

[1] https://www.w3computing.com/systemsanalysis/guidelines-dialog-design-hci/
[2] https://ubuntu.com/blog/command-line-usability-a-terminal-users-thought-process

qualities commonly used. The system usability evaluation criteria are: the time to complete a task (efficiency), time to learn a task (learnability) and the number of errors made over time (memorability) according to Jenny Preece and Helen Sharp [Preece et al., 2015]. For example, memorability can be measured by the user's ability to remember a sequence of operations or commands. Memorability can be included in the system by meaningful icons, commands names, menu option, etc. There are many more design principles that need to be looked into when designing an HCI system; as mentioned earlier, feedback methods, visibility, constraints with user interactions, consistency, and affordance (allows the user to know how to use a system) are some of them.

Blackwell [Blackwell, 2011] explains different HCI design and experimental techniques. One of the famous user-oriented design approaches is prototyping which has been applied to the study of the system interface for usability. HCI research emphasises the development of a large number of prototypes to explore different usability alternatives. Rapid prototyping, iterative prototyping, and deep prototyping are some of the prototyping techniques applied in user-oriented interfaces.

The most common empirical method of interface evaluation technique is the controlled experiment where observations and recordings are made while participants are using an interface. According to Blackwell [Blackwell, 2011], typical system measures are the time taken and number of errors where a broad range of alternative measure are possible, including performance based on age, prior skills, literacy, and heart rate.

Experiments should enable designers to generalise from the observed effect, identify psychological theories, replicate and repeat. Hence most HCI experiments contain more than one trial, more than one participant, and people from different age groups to prove environmental validity and avoid serious concerns. These types of controlled experiments need a significance test to disapprove that the observed variation is not a simple occurrence of random variation. However, it is also important to notice that human differences and limitations in the HCI field, such as subject motivation, accidental hits, individual IQ difference, distraction during trials or any other random factors, can cause variation in statistical measurements as explained by Blackwell.

When it is not easy to statistically measure systems, user feedback about the interface can be analysed. There are various techniques to capture user feedback in different stages of user-oriented design. Surveys come in different forms, with open and closed questions. Closed questions use a Likert scale for recording answers and open questions require questionnaires and thinking out loud studies.

Kasper [Hornbæk, 2011] explains the philosophy of experiments in the field of research. There are different reasons to undertake HCI experiments. Engineering research provides validation of modelling technique and HCI research is very much solution building or extending previous work as explained by Kasper. Experiments form an essential part of the chosen methodology, but it is important not to be simply method-oriented, rather problem-oriented to move towards the solutions.

There is an instance where an alternative to experiments can be used, and that is where a product development contribution is along technical or engineering lines. According to Kasper [Hornbæk, 2011], it is sufficient in such cases to run a usability study with a few users to study some system conditions that give insight into how to improve the system or run further studies. Alternatively, the researcher can engage with a long-lasting iterative development process where the evaluations drive further development needs. Another analogy is that, when creating a building, there is a clear difference in the way in which architects and engineers face the problem. Architects think about people and their interaction with the building, whereas engineers are concerned with issues to do with structural stability, durability, and other aspects. Similarly, there is a distinction between designing a human-computer interactive product and engineering a software product, discussed in section 2.4. The next section discusses the methodologies of the past HCI assistive technology system experiments.

2.3.2 HCI Systems Designed For BVI and With BVI participants

The case study of a game editor for orientation and mobility training by Mattheiss [Mattheiss et al., 2017] is an example of user-centered design working with visually impaired pupils. Pupils, teachers, and mobility trainers were involved. They used methods such as interviews, focus groups, workshops, game storming, digital surveys, behavioural observation, self-experience, and early-stage prototype testing. A group of 25 blind and partially sighted teenagers were design partners in the process. Design methods were discussed with the partners (i.e. the teenagers), as well as sample sizes, the aims of the study and a description of the method. The outcome of this design process emphasises the need to talk to the people who work with the participants prior to the experiment to understand the participants' behaviour, likes and dislikes of the environment. The researchers learned that they had to prepare for heterogeneous target groups with incompatible needs that could change the original ideas and design needs. Participants' liking of playful and brainstorming methods was understood. Researchers had to be aware of a group setting and previous experience of the group as the difference of opinion influenced the outcome. The limitations of the standard classification of digital survey scale options were observed where questions were left answered. The research needed many more prototype iterations than expected and prototyping did not provide answers to all the questions when the problem was too abstract and when observed inconsistencies were presented in the results.

Sahib [Sahib et al., 2013] conducted a participatory design process with blind users in a scenario-based approach. A dialog-simulated interaction between BVI participants and designers evoked a new web search interface that was yet to be constructed. One technology-savvy blind participant was in the design team and four others were involved in prototyping sessions. After each walk-through of the scenario the evaluator noted participants' feedback and then the feedback was analysed by proposing design decisions. The research reports benefits such that the scenario-based approach allowed blind users to envision the proposed interface and form imaginative thinking as to how they would interact with it, which was helpful to communicate design ideas rapidly and correctly. The involvement of the blind users in the design process meant it was important to use the right vocabulary in the absence of visual communication aids. On the other hand, the researchers were mindful about using a standard script to ensure that no variation happened during the conceptualisation stage and gave significant details to ensure that users could visualise the picture.

Vaza [Vaza et al., 2018] published a study that was designed with an interactive and iterative prototype construction and refinement methodology to build an exhibitor for assisting blind and visually impaired visitors in the tactile exploration of original museum pieces. He used a code processing programme to read user action and audio descriptions of the museum displays. However, there was not much information about iterative methodology. A questionnaire was used as the main research method for data collection to understand pragmatic and hedonic qualities about interaction, such as clarity of the audio, the similarity with the geological samples and improvement suggestions. The experiment had no time restrictions for exploration. Thirteen participants were asked about their use of technologies, the use of voice commands and screen reader software, in order to have a better understanding of the evaluation group. A five-point Likert Scale was used to analyse the participants' answers to questions. The answers were summarised using descriptive statistics and vertical Likert line charts and also analysis of direct quotes from the participants.

Kamel [Kamel and Landay, 2000] conducted a study of blind drawing practice to explore the limitation of tactile drawing and to understand what types of feedback a drawing programme for blind users must include. They used a swell raised line kit that has a rubber surface and a sheet of plastic where participants draw on the plastic using a stylus, creating marks on the plastic paper. Five participants took part in the experiment; three were partially sighted (blindfolded) and two were totally blind. They were asked to draw the letter "D", a ">" sign and two squares. There was no quantitative analysis of the experiment but participants were asked to compare what they intended to draw against what they actually produced. The results revealed that tactile freehand drawing provided no feedback to target an abstract point in relation to the drawing. Performances of every user were discussed with individual experience. Kamel also conducted another experiment with a grid-based IC2D drawing tool [Kamel and Landay, 2002] with eight sighted (blindfolded), five partially sighted (blindfolded) and three congenitally blind individuals. Participants were asked to draw circles in different grid positions, a triangle in the bottom left of the grid and a line connecting the triangle and a top rectangle to evaluate the participants' ability to carry out drawing instructions. They were also given system-produced images to explore to see if participants could visualise different spatial representations. Performance was measured with task completion time, a self-assessed level of confidence and the quality of their drawings with a score. There was no significant difference between VI participants and sighted participants on all three tasks.

The TeDUB system, introduced by [Petrie et al., 2002] make technical diagrams accessible to blind and visually impaired people. The system provides a mechanism to navigate and annotate diagrams. A user evaluation was carried out in four different countries with 35 participants to evaluate the effectiveness of the approach. The participants were IT literate and could analyse UML diagrams. Participants were asked to follow the instructions given to explore the UML diagrams at different levels of complexity. After the evaluation, the users were asked questions on satisfaction based on the point scale and provide written feedback. However, [Petrie et al., 2002] provides no statistical analysis on the accuracy of the task, time, or errors made.

The system KEVIN [Blenkhorn and Evans, 1998] that uses speech and touch to enable blind people to access schematic diagrams was not designed with users but evaluated using formal and informal evaluation techniques. However, the design is unique, logical and practical as observed at the evaluation. The KEVIN system represents the data flow diagram in N^2 charts by storing data inflows and outflows with processes. In the informal analysis, two blind users carried out a task to investigate and modify existing models, one using Braille and the other a keyboard. The Braille user managed to complete the task more efficiently and the keyboard user struggled to complete the task at that time. A formal evaluation was carried out to investigate many things including KEVIN versus tactile diagrams. Five blind participants were asked to answer questions about data flow connections on a given data-flow diagram using both KEVIN and tactile paper. There was no significant difference on the time spent in answering the questions using either method of exploration, even though the tactile DFD approach cannot be used to provide an interface to an interactive CASE (Computer-aided software engineering) tool. It is noted that there is no equivalent system to compare the results against, users only spent a short time using the tool and the small sample size may have impacted the results. As evidenced in system KEVIN some original, abstract ideas were not initially originated with a user-centred approach but such ideas were evaluated iteratively using a visually impaired audience.

2.3.3 Representing Users in Accessibility Research

[Sears and Hanson, 2011] point out the importance of relevant demographics for users in accessibility study research and their experience of similar technology. In most cases this would be the description of a participant by age and gender, as well as a description of their experience with technologies, in order for others to make generalisations in a new situation.

It is difficult to find candidates with unique forms of disability for studies in accessibility, for example an experiment group with blind participants, deaf participants, people with a specific injury, or motor impairment. The use of non-representative users has been the practice of some studies that have led to inaccurate conclusions. For example, a non-visual speech-based computer interaction for physically disabled students but without interactive devices such as a keyboard or mouse. Another example is a non-visual speech-based computer interaction for blind people's studies designed with blind and blindfolded individuals. Results analysis such as error rate and user satisfaction may not be very accurate if the process of user activity is not recorded or considered in discussion of the results. Blind users may easily spend more time getting familiar with speech commands and feel more satisfied than sighted individuals and demonstrate fewer errors than sighted individuals, as will be observed in experiments discussed in later chapters of the thesis.Therefore, traditional statistical methods with design assumptions may not be applicable for studies in the area of accessibility.

Alternative approaches can be designed, such as repeated measurement with single subject experiments, and use of non-parametric analysis may be appropriate. However, use of a standard HCI control group is beneficial to compare similarities and differences, although it shouldn't be a justification to use traditional statistical methods. Alternative statistical methods should be utilised if the study contains a limited number of participants or variability within the group is high. When recording the participant's capability, it is important to record medical or other formal assessment data rather than mere observation or self-report which gives other researchers the ability to replicate in their studies. If a study is based on a representative group, then the results should not generalise the outcome of the study for non-representative groups. For example, results from early blind individuals cannot be generally applied to late blind individuals. There is a difference in the performance and mental models of early and late blind individuals [Heller, 1989].

The conclusion drawn from the results should only address the clear visible outcome drawn from the study group. Data from separate groups (control and experiment) should be analysed separately, and a more comprehensive description of participants should be recorded for experiment replication purposes. Claims based on results should not be explained beyond what raw data evidence is there in order to lead to the development of more effective solutions in the field of accessibility.

2.3.4 Conclusion

In summary an HCI approach to system design and evaluation is commonly used and established among system developers. It starts with the designer taking into account what people are good at, what might help people, what provides a quality user experience, and trying and testing the designs. Users from different backgrounds bring diverse system needs, which can make progress difficult. Evaluation of systems for BVI users should accept user feedback and surveys as an important part of system evaluation measures, aside from data on task time completion, errors, and memorability. Such data informs the success factor of BVI user interface interaction, and it is most important in the improvement of assistive technology products. However, HCI system design and evaluation alone cannot guarantee the structural accuracy and completeness of the engineering process for assistive technology products. The next section elaborates on the engineering aspect of systems.

2.4 System Design and Evaluation using Formal Approach

This section discusses the research and efforts made in the past into developing different linguistic and grammar models when designing systems using a formal approach. It looks at linguistic importance in design, pictorial representation, and assistive technologies. Then the focus is given to grammar formation, past experiments to shape grammar and language models.

A formal language defines the technical specification when engineering a system, and HCI design defines the user level requirements of a system. A technical specification validates the consistency and accuracy of a system specification, whereas the user requirement specification is not designed to prove syntax and semantic accuracy but the validation of ergonomic properties. Therefore, formal specifications are not built by following a user-centred approach but rather a theoretical approach using description logic of mathematical constructors. Thus, in order to validate and refine the initial specification, several experiments need to be carried out.

Use of formal models in assistive system design can provide a number of advantages. It ensures consistency across multiple platforms, reachability, completeness, and incorporation of the user interface design process into a large, formally based software development process [Bowen and Reeves, 2007] .

Linguistics is the first step towards grammar formation. It is a methodological study of language. It analyses language for meaning in a specific context. Studies at the University at Buffalo explains that linguistics is a framework that is widely active in language description. Grammar is a set of structural rules that is defined once the language (linguistics of the domain) is recognised.

[Lukach, 2013] explains a linguistics model as artificially created using a device that reproduces by imitating the behaviour of its original or a sample that stands as a benchmark to reproduce the same type, whereas modelling is a scientific method to reproduce an object that cannot be directly analysed. Building a model requires the recording of facts, formulation of hypotheses to explain the facts, implementation of hypotheses in the form of a model with initial and new facts, and experiment verification. There are several types of linguistic models: research, analytical, synthetic, and generative models. Most models mentioned above take kinds of input information, mainly text, and produce text structure, grammar, or vocabulary to perform its intended task. Figure 2.7 represents stages of formal modelling using generative rule-based grammar. Firstly, the user language is fed into concrete syntax, then it parses with concrete syntax using a grammar specification which causes abstract syntax generation.

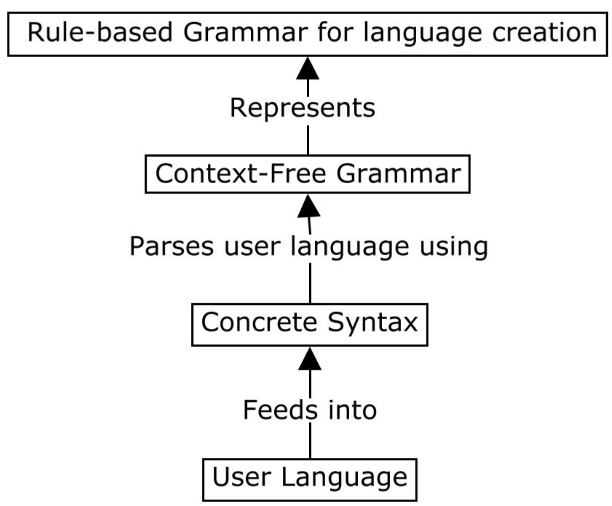


Figure 2.7: Development Approach of Formal Modelling

2.4.1 Systems Designed with Rule Based Grammar

Generative rule-based grammar was introduced by Chomsky's Syntactic Structure in 1957. It refers to a particular method of studying syntax. By applying the method of generative grammar, the designer gives a set of rules that will correctly predict which combination of words will form grammatical sentences; it goes beyond a specific language but conveys a universal model, that has properties common to all languages that are hidden by the outer structure [Amin, 2019].

Computational Models for Integrating Linguistic and Visual Information [Srihari, 1995] mentioned that it is necessary to have the ability to visualise in order to understand pictorial information about language. Practical applications such as knowledge-based systems, avatars finding a way with verbal description, or use of language to access graphics are examples of language-vision understanding. However computer vision and natural language processing are artificial problems too difficult to address.

When there are many input modalities, they need consolidating to a single unified presentation. Hence there is a need for a language model and a visual model. A language specification should consist of lexicons, grammar, object recognition, line drawing, image sequencing, primitive shapes and location, and space tracking techniques. Figure 2.8 shows demonstrates a proposed architecture to link visual and linguistic models by [Srihari, 1995]. The system accepts either visual or language input, but not both. The system does deal with both linguistic and pictorial inputs; however, they rely on the symbolic representation of objects to compare to the physical world. But the main challenge was for the model to understand spatial language, generate coherent text describing required objects, and association between objects.

Research by [Srihari, 1995] mentions that primitive shapes such as lines and curves, text and icons give a degree of understanding to produce an integrated meaning, like commonly accepted conventions in map drawing for example. The notion of meaning in picture maps of picture representations are somewhat ambiguous similar to those found in physical textbooks. Even though high-level natural language is involved in describing pictures in the above scenario, there needs to be language to explain pictures at basic levels of cognition.

Figure 2.8 shows demonstrates a knowledge-based architecture of image interpretation by text meanings, where a hypothesis is generated and consolidates information to provide output such as text information, pictorial information, and action in the physical world.

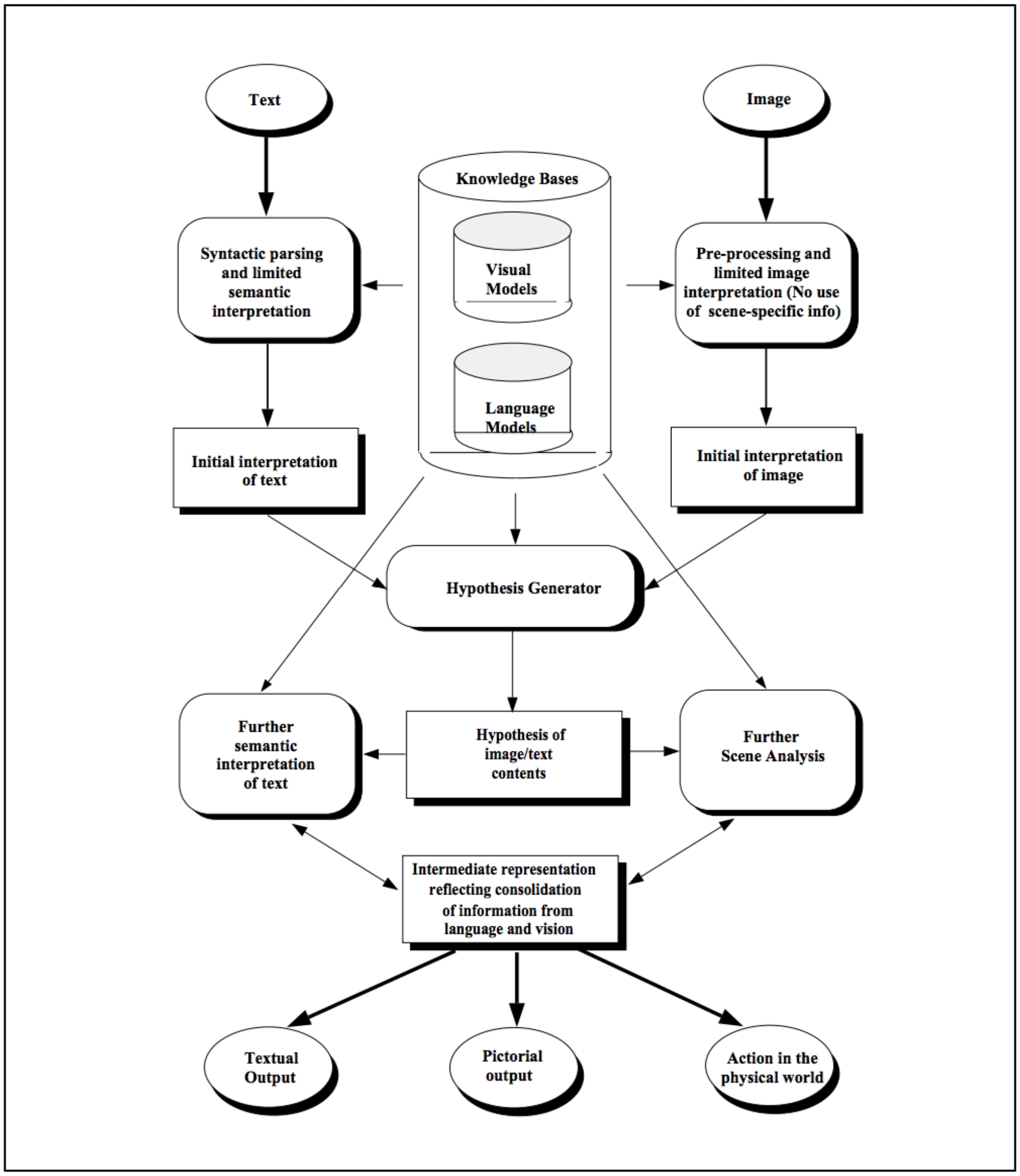


Figure 2.8: Computational Model for Integrating linguistic and Pictorial Information [Srihari, 1995]

When drawing a simple line or curve using language with one or many modalities, consideration should be given to start position, end position, orientation, width, patterns, and connection between segments. A better understanding can be assured if groups of primitive shapes are used to produce integrated meaning. Further mapping between symbolic description and visual properties is important when analysing images to give them meaning and accurate visualisation. Pictures with captions help to identify information about the pictures in an unknown environment, or for intelligent information retrieval tasks.

The user's interaction with their computer is linear by its nature, for example getting focus on a button, selecting an interface button, entering text from a keyboard. Creating a linguistic model for assistive technologies requires thinking beyond concepts of language, as it involves other details of presentation, language and transformation. The quality of these methods is based on underlying algorithms, linguistic dictionaries and comprehensive understanding of the target audience. Assistive technology deals with both natural and artificial language. Most artificial languages tree model structures are based on an abstract syntax tree.

[Davydov and Lozynska, 2016] have developed an extended mathematical model approach to assistive computer technologies based on linguistic models of machine translation. A model that consists of words W, concepts C, dictionary D, sentence structure metric space T (grammatically and semantically important relations in a set of possible word positions), semantics S, and grammar G (L=W,C,D,S,G). The introduced model has taken the order in which words can be counted with numbers similar to words in a syntax tree where nodes are counted with numbers. Events are passed as words and the position of words are passed as event sequences, such as a click of a button to save, enter text, etc. which are practically linear by nature in that they represent a number of events and an event sequence. There are other types of models that are based on language recognition and language transformation that use different types of modality as inputs. It is important to use low level language recognition and language transformation methods to protect the likely miscommunication in assistive software development, for example text to Braille or optical character recognition.

[Stiny, 1980] has introduced grammar for shapes using set theory and Boolean algebra. Operations such as union, intersection and difference in set theory can also be applied to a shape and its subshapes' formation. Shapes are made of a finite set of lines or a finite set of initial shapes and points (symbols). Labelled shapes can also be produced by shape grammar and can be used with languages of labelled shapes in architectural or mathematical diagrams.

[Tnpia, 1999] discusses the limitation of shape grammar by its application and implementation. Shape grammar contains lines either orthogonal or parallel to existing lines of the fist shape. All transformation and scaling is a 90 degree change along x, y coordinates. Tapia introduces a new paradigm for shape grammar where the designer creates the rule by changing the spatial and logical constraints. The designer explores the grammar language, while changing it to internal forms, explores the language, generates design imposing additional constraints, backtracking to a previous design, saving the current state, and interpreting the design as a basis for their design. The author mentions that when considering developing a drawing tool for the development of shape the tool must allow designers to determine spatial relations. In order to solve this problem, the author suggests the design needs an appropriate vocabulary to express special relations and a method for designers to express those relations in their own way.

Systems Designed with Context Free Grammar

Context-Free Grammar is a subset of rule-based grammar. Grammar that governs the production of a language can be a collection of words, phrase, or clauses. Theories that deal with grammar, such as generative grammar or context-free grammar, provide a framework to produce a combination of purposeful sentences using a set of production rules and replacement as discussed in Noam Chomsky's grammar.

Formal grammar [Cook, 1992] enables blind people to use a technique that is understood by the formality of a language, that is understood empirically, which has natural words, and the application of rules. This is difficult to achieve by writing a simple programme because of the limitation or inability to process semantics (grammatical correctness), its permutations and recursive nature. The model's (grammar) expansion or changes to the model are unable to be accommodated by a computer programme because computer programmes are unable to easily accommodate changes to grammar or rules. A model brings other software designers a formality of how systems work and should be modelled without defining design details, input details or output interaction. A model gives understanding for the designers about the philosophy of language creation. It enables structural design based on theory and experiments.

According to Chase [Chase, 2002] formal grammar-based systems provide better, automated control for system designers to be innovative, and to consider alternatives based on user-defined constraints. Grammar-based systems are considered to be active constructing tools, whereas systems such as computer-aided design (CAD) are considered passive. CAD systems are based on users' design decisions, whereas grammar-based systems gradually refine by each new construction. Grammar can also support new, emerging shapes/user requirements during design that enable designers to explore different design paths. However, grammar-based tools are still evolving, and their interface design is still primarily compared to systems like CAD. [Chase, 2002] introduced a user interface model that can work with the changing nature of user interaction with a grammar-based system.

[Chase, 2002] introduces different models for user interaction in grammar-based design, and different user control scenarios for the development and application of grammar. Users are able to select rules, sub-shapes, and matching conditions (transformation of rule shape to design shape is determined) that are interdependent. Greater flexibility and interaction of grammar application can be achieved if grammar invocation is controlled not only by the developer but also by end users, this can be either end with the user's or machine's choice of rules, shapes, or matching conditions. For example, the user is responsible for rule innovation as well as grammar development (full control); the user is responsible for some aspects of rule innovation as well as grammar development (partial control); the user has no control over invocation or grammar, the system automatically generates designs of objects/shapes (no control). A grammar generally consist of the following components, as explained by [Chase, 2002] and shown in Figure 2.9 .

- A vocabulary or vocabularies of objects
- Rules of the form A $\rightarrow$ B, where A and B are vocabulary elements
- An initial state

The rules of the form A $\rightarrow$ B apply to state "g" when a matching condition that maps A to "g" in a decided manner by navigating through a sequence of states and choice of rules that can also be

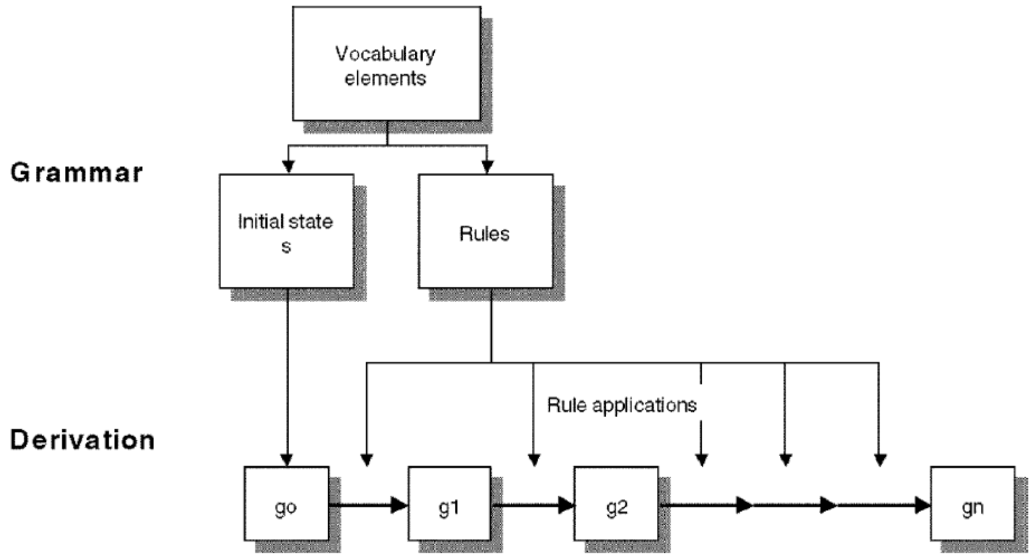


Figure 2.9: Grammar Derivation Tree [Chase, 2002]

explained with a derivation tree, as shown in 2.9. Decisions are made on stages in the design process where grammar is involved, different entities interact, and the user controls possibilities. Grammar development is responsible for the creation of objects/art representation, control mechanisms and rules.

Formal System Validation

As explained by Ait-Ameur and Baron, [Ait-Ameur and Baron, 2006] systems can be tested using formal and experimental HCI validation methods in cooperation. Generally, validation can take place either by automatic proving of a model using tree logic and/or a proof system abstraction, where the model is described by variables, operations, events, and other properties (using Z notation based on set theory). Formal testing can verify any level of abstraction, whereas HCI experimental testing is based on the GUI level of application objects. In a formal approach, logical expressions are used to demonstrate a transition from the initial state to a target state of a system task, defining a trace, to check any missing steps, using design and architectural validation. HCI experimental techniques demonstrate validation of user behaviour or ergonomic properties, where these kinds of properties cannot be checked by formal validation. Hence grammar-based system specifications were also evaluated with HCI user experiments in the past [D'Ulizia et al., 2007], [Dabek and Caban, 2017] by tracking commonly accepted measures such as accomplished tasks/actions, application errors and user satisfaction.

2.4.2 Formal Modelling of User Interaction

A language model predicts the best likely continuation of user interactions in which word sequences occur. Some systems rely on domain vocabulary gathered from surveys and others rely on existing literature analysis [Hsiao and Chen, 1997]. In either case, the efficiency of a model has to be tested and trained on a large combination of inputs from target domains. Creating a linguistic model for blind individuals has restrictions to do with domain vocabulary. For example, the concepts of far, near, big and small are relative perceptual matters that need addressing with relation to allocentric parameters [Struiksma et al., 2009]. [Vanlierde et al., 2003] have described strategies used to explore space by the early blind who adopted a "coordinate XY" strategy for spatial exploration tasks while the sighted cohort adopted a visual-spatial strategy which informed pointers to spatial language creation. The concept of a "language game" introduced by Ludwig Wittgenstein [Biletzki and Matar, 2018] suggests that language could be separate to reality, and concepts do not need clarity for meaning. Forms of language simpler than the whole language itself and speaking language can form part of an activity or way of life that gives meaning to the language suitable for its intended purpose. Some modelling languages such as Logico-linguistic modelling use a "language game" to create language during mode|ling for the creation of grammar rules.

Concrete Syntax

Concrete syntax consists of a set of rules that is familiar to the programmer, whereas abstract syntax is a set of trees used to represent a program to the compiler. When a user inputs some text, it is read and checked against the rules of concrete syntax and transferred to a tree defined by the abstract syntax [Appel, 2002].

Context-free grammar (CFG) is used in a compiler/interpreter to decide how to interpret a string as a programme, to represent the concrete syntax of a programming language. Rules are applied - the left side of the rule has one single non-terminal, but the right side may contain one or many terminals or non-terminals. CFG is widely used because of its support of recursive concepts to describe a programme language. Therefore, computer languages can be easily implemented in CFG. Backus-Naur form and syntax diagrams are a communication protocol and notation technique that can be used to design context-free grammar (CFG) [Norvell, 2002].

User Language

Various types of input and output devices and sensors can be used to receive and present information. In assistive language creation the user commands can be identified as input statements with text, speech, or Braille characters. Standard input mechanisms like keyboard, speech or Braille have interchanging technology to communicate to and from the computer. For example, speech recognising software converts analogue waves to digital data that the computer can understand and runs a statistical model that compares them to a large database of words, phrases, and sentences. Braille outputs sends the text line to be displayed through Braille's refreshable display, where dots in each cell (representing a letter) are raised so that the user can read the text by touch. Information can be conveyed with different statements. A statement can be read, checked, and processed for output production, such as graphics on the interface, with speech output and print outputs. A user input statement can contain terminals or non-terminals and can be recursively used to produce multiple graphics. For example, a user input statement could be “draw line, north to south point”, “draw curve east to west” or “call circle ”.

It is understood from the above that rule-based grammar enables system designers to generate a user language by first forming a set of rules in a syntactically and semantically accurate structure. It can be done with set theory, CFG, calculus, or any other type of formal design technique. There could also be a different degree of user involvement in rule formation in a grammar-based system, where a user could even be a grammar developer. It has been observed that systems designed with a formal approach to show pictorial data specify the need for a spatial language and spatial relationship language including language techniques to specify objects, shapes, locations, sequences, space tracking techniques and the association between objects.

2.5 Conclusion

This chapter has provided the context on the subject of visual impairment, the needs of BVI people, different uses of assistive technology and their advantages and disadvantages, and different design issues surrounding assistive technology from different point of views and analysis. This chapter further elaborates on the HCI design approach along with the formal design approach, highlighting the strengths and weakness of both approaches. As the research in this area moves towards "engineering" system design more than interactive system design, this thesis then focuses on answering a further question, namely, What constitutes a formal language for CAD for blind people?, by exploring the system modelling issues in Chapter 3.

Chapter 3

Modelling Issues of Computer-aided Drawing for Blind People (CADB) and Motivation

The previous chapter discussed answers to the initial challenges posed on design consideration and approaches of assistive technology. This chapter places in a logical context the first stage of CADB problem identification and classification using the formal model approach, which forms the bases of the major research contribution carried out in this thesis. The criteria used for inclusion of technologies was based on how clearly the model reflected important trends or model characteristics that were considered important in the context of this thesis. The chapter focuses on the answers to the research questions listed in the introduction: What mental models do BVI people use to imagine the shape of the world around them, and how could that be used to provide a drawing tool that they can use? What sort of an object layout a blind person would easily conceive and convey?, What sort of a navigation system would easily build a sense of object layout?, What sort of a space arrangement could allow easy navigation? and What usability features are best suitable to design CADB? The answers to above questions then led to answering this question: What constitutes a formal drawing language for the blind user? Presented in the chapter are modelling issues of CAD for blind people and the theories of blind drawing models (perceptual models). These are used to propose suitable drawing guidelines. Next, the chapter establishes the modelling issues of navigation techniques and presents theories of navigation models (cognitive models) from which suitable navigation guidelines are proposed. Also presented are usability models and recommendations for usability guidelines. The chapter ends with presenting a set of modelling guidelines for a shape language, spatial language, and a usability language for the proposed model with clear associations to theories through a classification diagram that is unique in nature and that has not used to classify assistive technology domain problems before. Moreover, formal constitution of a drawing language was introduced that brought out a new challenge, with the question, How can such formal drawing guidelines be modelled? These guidelines are then used to create a formal model specification in Chapter 4.

3.1 Issues of Drawing Techniques

This section discusses the issues relating to analogue and digital drawing tools. It then focuses on perceptual models of blind people's mental imagery, the short and long term memory capacity of imagery, blind drawing, and the substitution of vision with other senses in art representation and production. Finally, the section presents its findings on considerations required for shape drawing.

3.1.1 Issues of Analogue Drawing Tools

Blind individuals commonly use raised-line kits or handicraft drawing tools, [Perkins School, 2019] which are the most widely used art production methods in education establishments worldwide. These tools can produce either positive or reverse images. A positive image is produced when raised from a special kind of paper that is drawn on, for example German Film [RNIB, 2018] which can be used with any kind of pen. There are other heat sensitive materials or pens that give a raised effect to any drawing or predictable structures. It is important that these tools are portable, affordable, and versatile.

Draftsman from APH [APH, 2014a] allows the user to produce simple raised-line graphics on film paper. A stylus is used to trace the drawing which produces a raised line. Draftsman is used for demonstrating maths and science concepts, practicing handwriting skills, playing games such as tic-tac-toe and creating art drawings. A board encases a double-layered rubber surface that includes drawing film and can be used with a stylus or any pen to draw an image. One of the limitations of this tool is that the drawing is not digitally stored, therefore there is no functionality to erase, reproduce or reuse it. Also the stylus tends to fall out therefore it is helpful to use a separate rubber mat.

InTACT SketchPad [inTACT, 2014] is made with a stylus and tactile film. A raised line is generated as the person draws, enabling the person to feel the drawing. The device is portable and secure and holds the papers still while drawing.

Some other examples of analogue drawing tools are TactiPad [SightandSound-Technology, 2017], Sensational Blackboard [Sensational Books, 2012], Sewell EZ Write N Draw Raise Line Drawing Kit [MaxiAids, 2019], Swai Dot Inverter [APH, 2014c], and Quick Draw Paper [APH, 2014b].

All of these tactile drawing tools have some features in common. They use special kinds of foil or film sheets to get the raised effect to give the feel of the drawing, which can be positive or reverse. They are portable, lightweight, inexpensive, and secure to hold in many cases where a rubber mat is used. However products of this nature cannot erase the created image, the reproduction of images is difficult, there is a limited amount of drawing support, and the resultant drawing cannot be easily inserted into a digital document [Bornschein and Weber, 2017]. Figure 3.1 shows tactile graphic production toolkits taken from Perkins School. It includes products such as Draftsman, Intact Sketchpad, Sensational Blackboard, and Sewell EZ Write N Draw Raise Line Drawing Kit.

Figure 3.1: Tactile Graphic Production Toolkits, [Perkins School, 2019]

3.1.2 Issues of Digital Drawing Tools

Computer-based drawing mechanisms can overcome the disadvantages of the above-mentioned tactile drawing products. Digital drawing allows easy reproduction and sighted users can benefit from these facilities and share a common platform for drawing and understanding images using universal symbols. However a properly effective drawing application is difficult to find and produce due to the complexity of what is required. Drawing a picture is a hard task if one is not trained to transfer the perceived image to paper, even with a simple geometric shape or a parallel line let alone a complex object such as a face or the human form. From connecting two points without help to converting a three-dimensional object to a two-dimensional representation, these remain difficult tasks as they require a high mental load and understanding for a blind individual [Bornschein and Weber, 2017].

According to Kurze, blind perception is based on the process of fitting sensory perceived information from the environment to an existing internal idea, the mental image. He mentions that "The most important characteristic of a picture is not it's channel of perception (visual ,or non visual) but the methods of arranging information in space" [Kurze, 1996]. The following list of drawing tools are based on arranging information in space to perceive that information using different modalities.

The IC2D System

Kamel and Landay [Kamel and Landay, 2000],[Kamel et al., 2001], [Kamel and Landay, 2002] introduced a product called IC2D, see Figure 3.2. IC2D has palettes for users to select shape types and colours. It divides the screen into nine navigable smaller workspaces. In order to make more precise selections, each of the nine cells is then divided into a further nine cells, which in turn are divided into nine more cells. Annotations are given for meaningful semantics so that the user can build an accurate mental image. The IC2D interface relies on keyboard inputs and provides auditory feedback using a voice synthesiser. The main focus of IC2D drawing is to enable blind users to pick a cursor point on the screen to enable navigation from one point to the next point on the screen.

The formulation of the 3x3 grid-based system works well with intended blind users' intuition as Kamel's findings suggest. The 3x3 arrangement is based on the numbers of a telephone keypad, with which blind users are familiar, and enables them to move to and from any of the nine cells as follows. Keyboard keys act as directional keys such as up, down, left, and right. Also the numbers act as cell locations. The unique point of reference of a cell is the centre point, which is then selected to perform drawing. The main functionalities of drawing are supported by different palettes such as shape type, colour and file commands. Each palette provides nine options to which users navigate using the same grid-based system. Hence keyboard keys can be used to operate these palettes. Using recursive grid schema, it allows users to make more precise selections. Users can find their positions by further dividing into nine more cells and it goes up to two additional levels. IC2D also helps blind users to label their drawing by their shape and hierarchy in order to recognise them later.

IC2D enables blind users to navigate and find screen and palette information without the help of a sighted user. The simplicity of keyboard input and synthetic voice output is helpful in learning and developing this technology. However, the system limits the shape and colour choices to its own palettes, it only uses centre point cell referencing as opposed to multipoint cell referencing which could be more efficient, and the system modality is tightly coupled with keyboard input and menus. However, once created the artwork cannot be reused. The tool was not based on a universal set of design guidelines but built with concrete system functionalities, therefore it is difficult to replicate or modify. Figure 3.2 an interface showing 2D shapes from IC2D. It includes shapes such as different sizes of rectangle, a triangle, different types of circles, line shapes, and a pentagon.

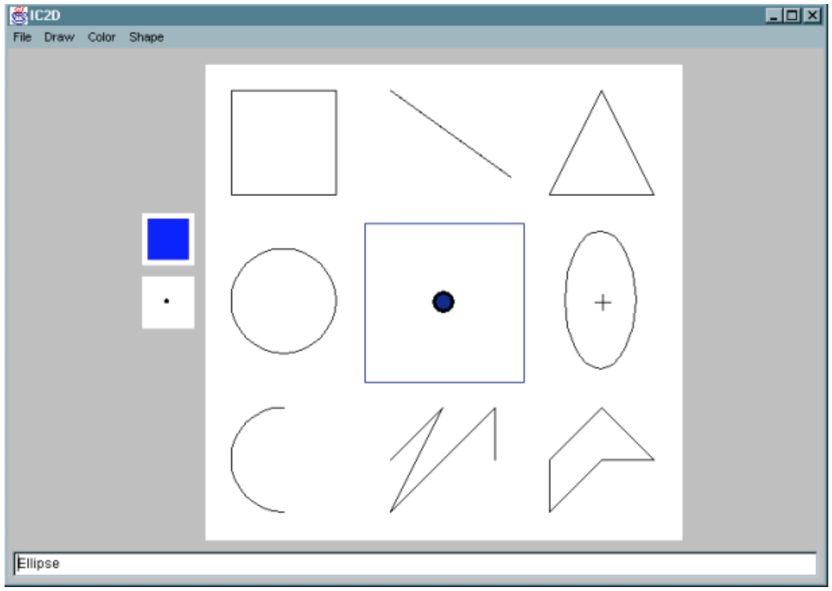


Figure 3.2: The IC2D Shape Palette, [Kamel and Landay, 2000]

The KEVIN System

The KEVIN system in Figure 3.3 by [Blenkhorn and Evans, 1998] enables users to read, edit and create diagrams using an N^2 chart (a matrix representation of interfaces between system elements). This system does not keep track of screen layout information but retrains the mandatory information needed for a simple data flow diagram (DFD). It has all the facilities of a typical CASE tool. For example, transformations of a DFD are located on the main diagonal of an N^2 chart, leaving connections are written in the same row and entering connections are written in the same column. This is a talking tactile diagram and it has a tactile layer. Different speech messages are delivered as the user moves around the DFD. KEVIN has a data exchange facility through the standard CASE data exchange format, which means the system can read diagrams created by other CASE tools.

The KEVIN system only supports DFD diagrams, but is not adaptable in educational settings where users need to create other diagrams associated with STEM subjects. One of the main disadvantages of the system is that it does not keep track of layout information, therefore the user has no sense of object placement. Also, when other diagrams are imported into KEVIN the transformations and connections of a DFD must be removed.

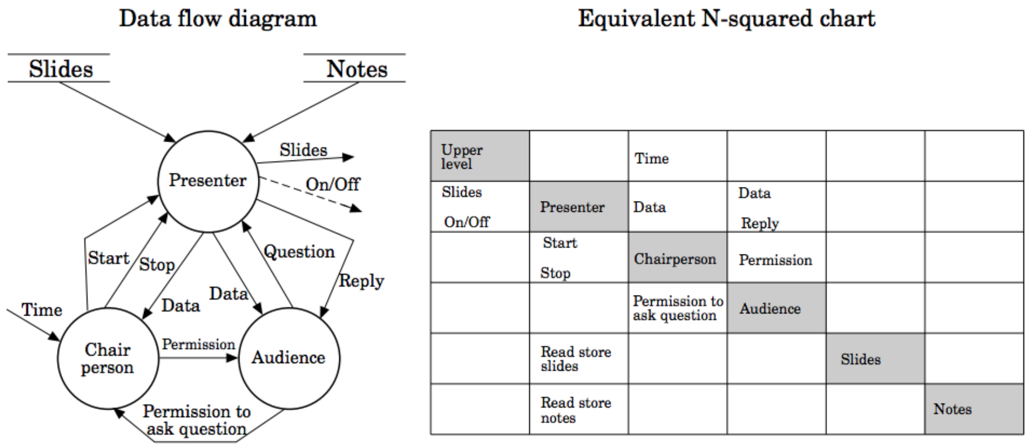


Figure 3.3: System KEVIN, [Blenkhorn and Evans, 1998]

The PLUMB System

The PLUMB system [Calder et al., 2007] uses linked lists and Heap's algorithms to store data in a data structure and to access it in a sequential manner. It displays diagrams, graphs and relational information to users who are blind. Given a drawn graph, the system uses auditory cues to help a blind user traverse the graph. The user can select the data structure to animate its operations, as the user steps through the algorithm for the operation and observes how each step affects the data structure. The system provides both graphical interfaces for sighted users and the PLUMB interface for blind users. It uses synthesised speech to announce the selections and operations, for instance when the user selects "insert node" a new node is inserted into the existing link list, which is then audibly acknowledged to the user. Users can navigate forward and backward to explore the state change. The system uses GXL XML files to read and save information and this portable format allows the modification and creation of data structures in a variety of different editors. The system explores vertex and edge at a given time. The intensity and the tone of musical sounds dictate the distance to vertices and edges. With the keyboard activation system more details of vertex and edge are given. The PLUMB system conveys no information about the shape of the layout of the graph. The tablet is pen-based and requires students to explore graphs which can be time consuming and depend on the precision of the user's hand movement. The application generates audio clues when the user moves the stylus over an element of the graph. But the system has no clear indication of layout information or shapes. The original PLUMB system uses the modality of a tablet and pen and a later version employs a keyboard.

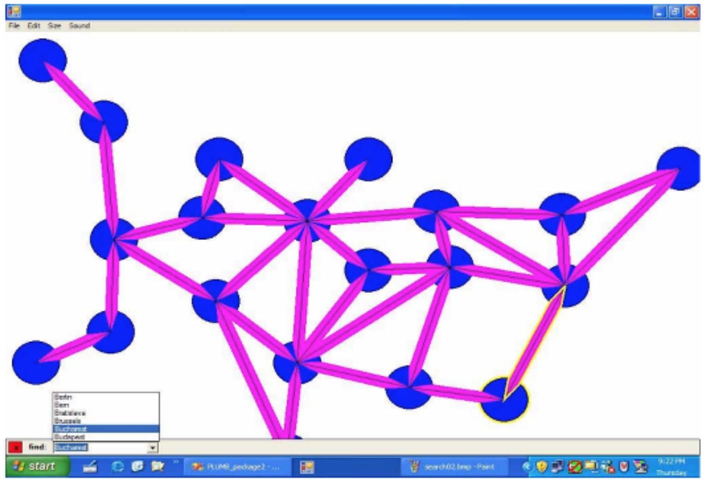


Figure 3.4: System PLUMB, [Calder et al., 2007]

The TeDUB System

The main focus of the TeDUB system, introduced by [Petrie et al., 2002] and presented in figure 3.5 is on importing images or diagrams in their visual form in a way that blind people understand. It provides an automated interpretation and accessible presentation of technical diagrams and was developed according to COTIS (Confederation of Transcribe Information Services) guidelines taking non-visual navigation of information space and verbal descriptions of diagrams into consideration. The images in formats such as BMP, PNG, JPEG and GIF go through an image analysis process, identifying their related text. Then the system recognises basic geometric primitives such as lines, arcs, and text regions. The second stage involves knowledge processing which interprets the image by network hypotheses. This stage requires involvement of a sighted person, as the stage is error-prone when identifying important image components. Some UML images can be encoded and fed into XML format to TeDUB, which makes them easier to be read by the system without error. The system also keeps track of the knowledge of its previous images to interpret new SVG (scalable vector graphics). The system uses synthesised speech and a Braille display to communicate the image to the user, and keyboards and joysticks also facilitate system functions. The TeDUB system suggests that it could be improved with a numerical keypad for navigation.

The TeDUB system is not a graphics drawing tool but an image exploration tool. The tool is discussed here because its detailed layout information is not found in many other drawing tools. It presents a verbal description of diagrams by analysing primitive geometric shapes and giving them a label based on a knowledge-based hypothesis which could be error prone. The system also keeps track of previously analysed images for further use.

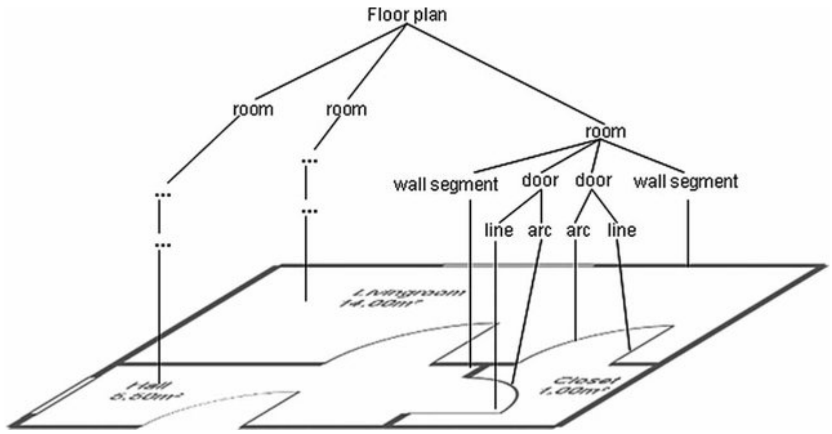


Figure 3.5: System TUDUB, Automatic Interpretation and Presentation of Technical Diagrams for Blind People [Petrie et al., 2002]

The AHEAD System

Figures 3.6 shows the AHEAD system, an audio-haptic drawing editor and explorer for education. The virtual environment consists of different modes. The user selects drawing objects by touching them with a PHANToM pen or computer mouse. PHANToM users can feel the lines while drawing. The drawing can be changed by resizing, copying, deleting, pasting, moving, and saving for later.

Figure 3.6: Audio-Haptic Drawing Editor And Explorer for Education, [Rassmus-Grohn et al., 2007]

The AHEAD system is a haptic drawing/exploration tool that needs to be used with a sighted person or an assistant. A mouse user can draw by pressing the left mouse button while a PHANToM user can feel the drawing in their PHANToM device. It can take more than 30 minutes for a user to explore the systems map. AHEAD is a haptic tool, that is tightly coupled with haptic modality, not easily portable and needs a second person.

The BPLOT System

There have been recent developments in command-line drawing products for blind and also sighted people. Recently introduced systems BPLOT, [Fujiyoshi et al., 2014] and BPLOT2, [Fujiyoshi et al., 2008] employ system dialogue to create a drawing using command language. But the command language is mostly for sighted users of BPLOT when using graph points. BPLOT2 [Fujiyoshi et al., 2008], introduced a universal design tactile graphics production system that uses a plotter control language. Figures 3.8, 3.7 show plotter commands from the BPLOT2 system and a graph output of the commands. Plotter commands are calling the numbers of the x and y-axis to draw lines and curvature. It uses universal plotting language co-ordinates on the sheet to produce a user command language. Plotter commands consist of the name of the command and the parameter, such as circle or straight line, which can be described as single commands that are imported from a vector-graphics printing device. Automatic text-to-diagram conversion introduced by [Mukherjee et al., 2014], is also referenced by coordinates with the numerical values of the end point of the four component lines.

The BPLOT system has evolved through many versions and is currently on version BPLOT4 that has gone through interface and language changes. The system has multiple modes of input apart from its unique plotter control language that is fundamentally created for sighted users. The blind user is given a tablet input with a stylus to directly draw the image to be stored on the computer. The system initiates the plotter control language and the resulting pictorial image is displayed in another canvas window. The plotter control language is not created for the interaction of the blind user but is used by sighted people to change the created graphic. The language directly addresses x and y coordinates of a canvas that can go up to the maximum size of screen width and height. Hence it is difficult to make accurate sense of the location if an image is not presented.

```
// Labor force participation rate
aux
window 1 41 0 24
    braille 5 23 ,Lab—r =ce "picip,n rate
    braille 8 22 ( ,Japanese wom5

window -10.4 10.4 -12.5 12.5
    origin -6.5 -8

dot 0
    xaxis 0 15 3 2 2
    yaxis 0 15 1.5 2 2

    braille -2.3 16.2 7p}c5t7
    braille -2.3 15.2 #ajj
    braille -1.8 0.2 #j

    braille 1.5 -0.5 #be-#bi
    braille 7.5 -0.5 #de-#di
    braille 7.5 -1.5 7age7

dot 1
    spline 0 15 0 15 2 0 0 2
        0 3 // point 1
        1.8 11 // point 2
        4 7 // point 3
        9 10 // point 4
        12 6 // point 5
        15 2.5 // point 6
        9999 0 // end of data

ff // formfeed
```

Figure 3.7: Plotter Language, [Fujiyoshi et al., 2008]

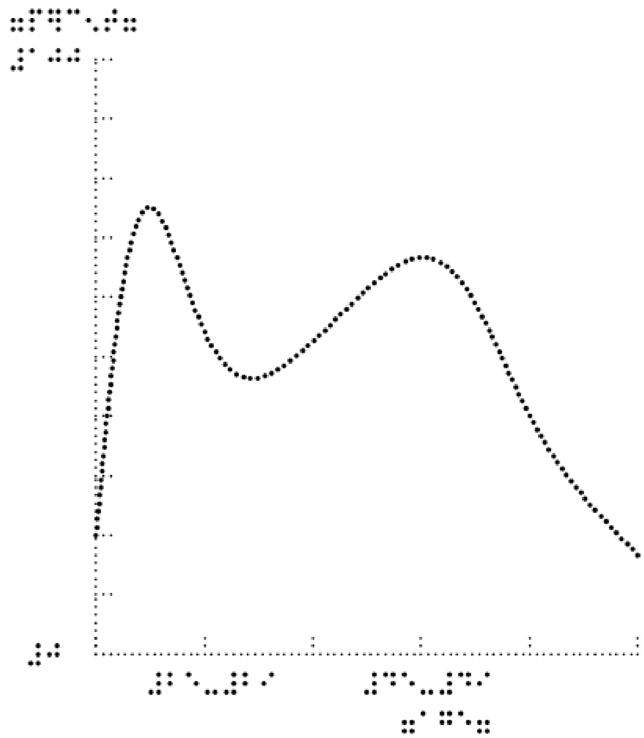


Figure 3.8: Plotter Language Graph, [Fujiyoshi et al., 2008]

According to [Bornschein and Weber, 2017], a good drawing system for blind people enables independent creation, is intuitive and easy to use, provides a graphical user interface, gives direct and continuous feedback, has keyboard-based interaction, allows change and modify functions, enables connecting lines, labelling elements and grouping, provides help and error correction, has zooming functions, and many more usability commands. However, the above-mentioned systems have their own merits but are not always viable nor do they necessarily have all the above desirable qualities. The next section looks into the mental imagery capabilities of blind people and how that supports actual drawing and vision substitution mechanisms.

3.1.3 Perceptual Models

This section covers the mental representation of information among different group of people and includes mental imagery and blindness. Mental imagery is defined as the human ability to visualise (or construct mental pictures of) various concepts, where the concept can be a simple object or as complex as an entire sentence [Pinker, 1984]. It is argued that mental imagery has two properties - the ability to know the properties of an object and the ability to think and reason in the absence of preconceived perception [Kosslyn and et al, 1990]. It has been identified that short term memory (STM) and long-term memory (LTM) are dedicated to mental imagery. STM is known as a visual buffer that has an array-like structure, where shapes are displayed in an object-centred representation, and LTM contains information about appearance, which is viewer-centred information. STM and LTM work together in processing new combinations of previously viewed objects.

Withsgen [Withagen et al., 2013] experimented on the short-term memory (STM) and working memory (WM) of blind and sighted children using tasks such as name learning, word span, listening span and backwards digit span. It was found that blind children outperformed sighted children in both STM and WM tasks. The blind group outperformed the sighted especially in verbal tasks. It also explains that visually impaired people have superior memory abilities and use serial strategies to outperform the non-existence of visual information, as their mind become more adapted to spatial, sequential, and verbal information.

Miller and Johnson-Laird [Miller and Johnson-Laird, 1976] introduced an algorithmic description of processes and verification as a fundamental source of evidence for determining linguistic meaning. The two main components of the understanding are perception conceptualization (identifying an object) and understanding the lexical concept (the label associated with that object), and these can be taken together when integrating language and vision.

One of the important questions that needs to be addressed is "What type of internal representation is used by BVI people in daily decision making?" The study of [Kerr, 1983] shows that congenitally blind individuals use spatial representation where as sighted individuals use mental images when reasoning. However both groups have proven to perform equally well if a task required individuals to work with spatial representation.

Knauff [Knauff and May, 2006] hypothesizes that visual mental imagery is not necessary in reasoning. They experimented with the mental image reasoning of different groups, namely sighted, blindfolded and blind, and revealed that clear verbal descriptions affect the process of representation. It was easier for people who were blind from birth to construct a correct spatial representation when the information was conveyed correctly. However it took longer for a sighted participant to reason when the representation had visual images that had irrelevant information. The congenitally blind individuals were not perceptive to such negative interference since they did not use irrelevant visual details. [Chen, 2011] published experiment results comparing blind and sighted participants in their ability to use audio tones and signals to identify graphical objects and draw pictures and shapes with his programme which was named SoundVision. He used similar technology and techniques to scan a given graphics image in the way that blind people use a white cane to scan obstacles in order to generate a mental model of their surroundings. Sighted participants were involved and blindfolded as a standard baseline to understand the difference in perceiving sound from different groups. The results confirmed another alternative method, the use of audio tones to access graphics information and express objects graphically.

Even though much research was done such as [Millar, 1994] on acquiring spatial information via non-visual modalities, the important question to address here is whether the spatial information obtained by non-visual modalities (by BVI people) can generate the same properties of external representation in their mind's eye, or does visual imagery depend on previous experience of life? Kerr [Kerr, 1983] experimented with the difference in time taken to scan visual imagery between sighted and BVI individuals. The results explain that spatial identification is achievable by blind individuals, even though they took longer than their sighted counterparts, and was not significantly different. A similar point is confirmed by [Iachini and Giusberti, 2004] whose results show that vision and visual mechanisms enhance the speed and accuracy of spatial performance. Apart from the support of different modalities, a set of vocabulary is proven to stand as a representation of spatial information in BVI image and space exploration according to Bloom [Bloom et al., 1996].

Struiksma [Struiksma et al., 2009] published a study with sighted and blind individuals about the nature of spatial images and the role of spatial language. It revealed that blind individuals were able to perform tasks associated with spatial imagery, such as mental scanning, mental pathway completion and mental clock time comparison, even though congenitally blind individuals had never experienced visual inputs, but it was not always in a similar manner to sighted counterparts. The author argues that spatial imagery perception goes beyond input modalities but maintains the connection with verbal and perceptual inputs. People perceive information with multiple modalities. Some information is extracted as perceptual symbols. These symbols can be combined into concepts. Therefore words are linguistic symbols that are related to perceptual concepts, and they provide a set of processing instructions to create a mental representation of a scenario. When spatial information is provided by verbal communication, the analysis takes place in working memory. The study revealed that when memory load increases people adopt a pictorial strategy and a combination of modalities can strengthen the image. However, a study conducted by [Cattaneo et al., 2008] revealed that language and motor inputs are the next most important sources of information, after visual input, which contribute to spatial images.

The visual cortex of blind and sighted individuals has been investigated during the formation of mental images [Lambert et al., 2004]. It was found that blind individuals activated dorsal pathways (haptic pathways) when generating mental images from verbal instruction which led to research on spatial language modelling. Further studies by [Vanlierde et al., 2003] have described strategies used to explore space by early blind, late blind, and sighted individuals. They reported similar performance for early blind, late blind and sighted individuals with regards to a mental representation of a verbal 2D pattern in a grid. While accuracy was the same for both blind and sighted individuals and performance was highly similar even though they used different strategies, the sighted cohort adopted a visio-spatial strategy while the blind adopted a "coordinate XY" strategy.

The above research on perceptual models reveals that mental imagery is not simply a product of vision, therefore BVI people can and should work with activities related to art and imagery using the appropriate information with other senses. Correct, sequential style verbal instruction is proven to activate haptic pathways in generating mental images in blind individuals. BVI and sighted people do not necessarily use the same method of decoding strategies, but their performance of image exploration is not significantly different. The section below further elaborates on blind drawing in blind people's terms.

3.1.4 Blind Drawing

Arnheim [Arnheim, 1990] introduces blind perception not as what sightless people are missing, but the way in which blind people conceive their haptic world in its own terms. Receptor organs of touch are located all over the body from the top of the head to the bottom of the feet. Touch perception is also connected to a person's awareness of their body movements, whereas eyes are limited to a single projected view of an image. In haptic drawing, blind people show a preference for symmetry and other simple forms of relation, hence there are different styles of art that meet these conditions that model structural features of art rather than figural representation. Blind people find it easier to perform line drawing as the perceptual cognition is related to understanding structural features .This method is evidenced in raised-line drawing that has been used for many decades . Rudolf Arnheim's emphasis is that it is important to see the contribution of blind people to the culture to which they belong, rather than urge them to make up for their lack of sight.

Outline Drawing

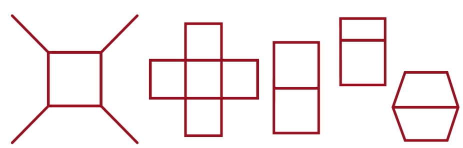


Figure 3.9: Outline Drawing of a Table by a Blind Subject [Kennedy, 1997]

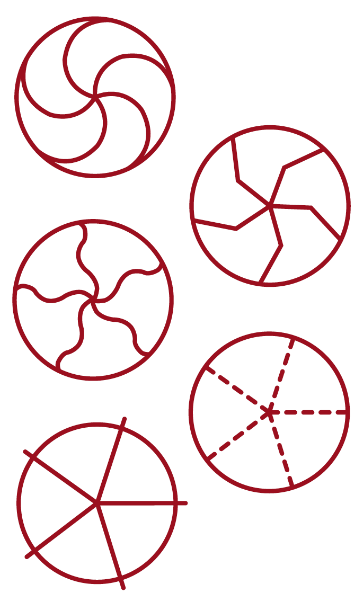


Figure 3.10: Outline Drawing of a Moving Wheel by Blind Subjects [Kennedy, 1997]

The images in Figure 3.9 were drawn by blind subjects to represent tables. The first view shows the table represented by a star shape to represent its appearance from underneath. The second is like an opened-out box, showing the front face of the table in the centre surrounded by top, bottom, left and right faces. In the next, two squares represent the front and the top of the table. Another shows the front of the table as a square and the top as a shortened rectangle, as it is further away from the observer. The fourth one features two trapeziums joined together, the extra length denoting the edge nearest to the observer.

Figure 3.10 shows drawings of moving wheels by blind subjects, with curved spokes for a spinning wheel, wavy spokes for a wobbly wheel , bent spokes for a jerky wheel, extending spokes for a braking wheel and dashed spokes for a quickly moving wheel.

Pictures are made to communicate, not just for representation. Kennedy [Kennedy, 1988] mentioned that perspective is not entirely a matter of convention, it has an intuitive basis in picture making, hence the observer's ability to understand the intention of the artist, and that plays a part in understanding art. Based on the studies conducted by [Kennedy, 1988] it is reported that blind people's ability to produce outline drawing contains universal elements, irrespective of their origin or sight limitation, and that they contain similarities to sighted people's drawing. Results similar to ancient cave and rock art were produced by blind people through their drawing, even in those who had no previous exposure to such pictures. Line drawing has been successful even with no prior experience, where the lines can denote surfaces, edges, corners, boundaries, and parallel features. Line drawings do not show what is purely visual but depict features that are understood by touch, which suggests line drawing lies in the perceptual pathway common to vision and touch. Practical applications, such as embossing techniques to show raised line illustrations, have been helpful for blind people's day-to-day lives.

When asked to draw the movement of an object (e.g. wind, hand in pain, rolling wheel) rather than abstract art, blind subjects depicted the art metaphorically, literally or diagrammatically with meaningful context. The blind subjects clearly showed the capacity to devise metaphoric ways to show abstract art, quite similar to the way a child uses metaphor and expects to be understood [Kennedy, 1997]. He also found out some common pictorial shorthand used by both blind and sighted subjects, such as lines to represent edges and irregular lines to convey motion.

Kennedy conducted an experiment to find out the symbolic meaning of abstract primitive shapes. He used word pairs where each word pair went with a shape. For example, one soft-hard word pair was found to be describing circle-square symbolics and it seems to have been agreed upon by a high majority of subjects. Blind subjects are able to identify surface and edge function the same way sighted people receive information about a line drawing by vision and touch. Hence blind people appreciate line drawing and other graphic symbols when drawing art. It was discovered that blind people use outline drawing to describe object arrangement and other surfaces in space.

The section below explains the benefits of other senses in the absence of vision, and their advantages and disadvantages in mental imagery.

3.1.5 Vision Substitution and Graphics

Converting information acquired by one sense to another modality is called sensory substitution. For example, information acquired from cameras can be converted to audio or tactile outputs so as to "see with the brain not with the eyes" [Bach-y-Rita et al., 2003]. There are systems that convert visual information into sound, [Meijer, 2018]. These may require extensive training. The tactile video substitution system by Bach-Y-Rita has experimented with turning images into tactile stimulation in the form of vibration or direct electrical stimulation of the tongue. Even though the results were promising, they were of limited practical value as a result of the complexity of interpreting real world objects, user concentration and user expectation, [Collins, 1984].

The evidence of Munhall [Munhall et al., 1996] states that different senses influence each other. Tactile or auditory senses could compensate for visual pictures according to the McGurk effect, which says that multiple senses give a better perception of the world to a blind person than simply using one sense at a time.

Research conducted in Italy by Ricciardi and Bonisno [Ricciardi and Bonisno, 2009] related to the vision issues experienced by blind people illustrates that activities can be learned and performed in the absence of sight; the brains of congenitally blind individuals demonstrated the activation of learning and visual processing which had not been learned through visual modality. The visual experience is not a necessary prerequisite for blind people to interact effectively with the external world. Individuals with no visual experience acquire knowledge of shapes, movement, and placing of objects through non-visual sensory modalities. Therefore, visually based imagery is not necessary for an abstract representation of the object and spatial features in the visual cortex of the left hemisphere in congenitally blind individuals. Brain plasticity research conducted in France by [Lambert et al., 2004], shows that the primary visual cortex (PVC) appears to be persistently active in a mental imagery task when only verbal instructions were given to subjects who were blind from birth. This activation of the PVC has a connection with cognitive functions when performing cognitive tasks such as mental imagery in blind people. Therefore this finding is helpful in developing sensory substitute devices such as blind drawing tools. People with no vision find it difficult to convert 3D mental images to 2D drawings. This task is harder when using hand drawing or touch screen manipulation. The positioning of the focus point and knowledge of the ending and starting points of a drawing are unclear, [Ishihara et al., 2006]. Peripheral localisation becomes more difficult for older adults and blind people. If a message is conveyed by voice, it has to be short and easy to understand due to the possibility of exceeding working memory capacity. However, if the information is too simple there is less benefit from multiple modalities. Thus processing load, disorientation and redundancy effect have to be removed for good engagement or learning, and distributing the information load across multiple channels is advantageous, [Hersh et al., 2008].

Pictures and diagrams can be constructed using low-tech manual solutions, such as cardboard, pins, rubber strings, Swell Papers and other physical items to build tactile images. But there could be complexity associated with these techniques. Some pictures need more paper for printing, tactile objects could become messy with information and the process could be expensive. Therefore other sensory channels, such as audio and haptic, have been experimented with for unimodal or multimodal information presentation, [Cohen, 2004].

Haptic technology is still in its infancy when compared with speech technology. Common problems include the greater time it takes to complete a drawing compared with an audio drawing tool, and difficulties in finding exact locations, navigating on a digital screen or a drawing sheet, finding the starting and ending points, and finding relative positioning. Nevertheless, speech-based solutions are relatively well developed, low cost and fairly easy to use, and take less time to complete in a learning environment, [Zhu and Feng, 2010].

Maps can be produced using tactile drawing although voice can play a role in it. It needs training to grasp the differentiation between 2D and 3D senses. It needs thorough understanding of geographical compass locations, distance, scale, relative position and so forth. Research work has been undertaken into the standardisation of tactile maps, such as environmental symbol features and audio feedback for augmented tactile maps [Collins, 1984]. These systems are useful but difficult to produce because of the cost associated with tactile overlays. Devices such as joysticks, gamepads and trackballs are capable of providing haptic feedback in map exploration.

Graphics production for blind users has its own complexity. Visual arrangement in tactile drawing needs extensive training and touch itself cannot resolve fine detail in the way vision does. Hence visual pictures must be carefully adapted, and even then identifying tactile drawing is very difficult for early blind people [Hollins, 1989].

Kurze [Kurze, 1996] in TDraw expresses the characteristics of blind drawing. In Kurze's early version of TDraw, a raised line was used to draw on swell papers and a digitizer pen to comment on each object, and the computer retained the object details and vector graphic files as shown in Figure 3.11. Talking Blaster is used as the output from the audio element of the system. One of the developments introduced by TDraw is the importance of object arrangement. The user not only wants to simply draw objects, but arrange the objects so that they make sense. According to Kurze one way to do that is if the user recognises 2D-screen arrangement and order. It is necessary for immediate feedback to be made available during the drawing activity, as well as smooth finishing, a functionality to attach tags /comments, an ability to change the image and general ease of use of the system. Object arrangement, object connection and order are all recognised as important facets of graphics manipulation. Figure 3.11 shows an actual drawing of a chair and table by a blind person presented in Kurze, 1996.

Presenting information using graphs plays a crucial role in both education and the workplace. A graph generation tool was experimented with by Yu Wai and developed using haptic technology including a WingMan Force Feedback mouse, keyboard, swell papers, and audio files [Wai and Stephen., 2003]. This tool is web-based and uses Java SDK features such as applets and MIDI files to render the animation and to produce varied voice pitches to differentiate data in bar and pie charts. A grid-based layout is used and screen starting points are announced with coordinates. The evaluation of the system

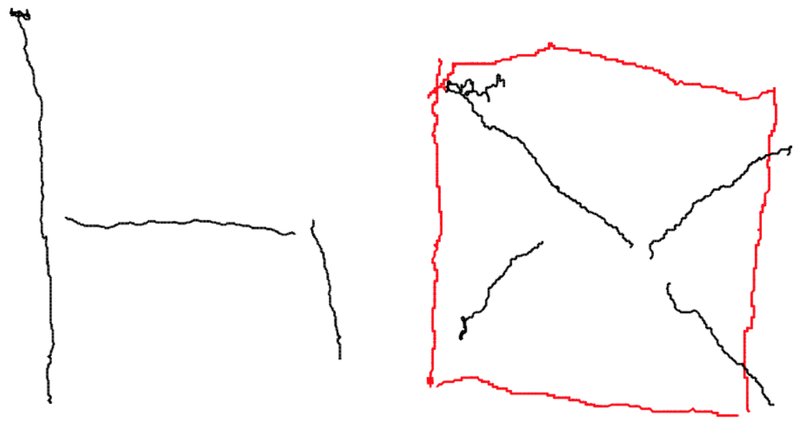


Figure 3.11: TDraw, A Chair and a Table [Kurze, 1996]

shows that there is easy navigation and clear data entry facilities are available. However feedback on the starting of a graph and the edges of bar charts were not announced enough to the user. Findings conclude that it is important to test BVI tools with different screen readers (such as JAWS, Window-Eyes or Supernova) to make sure that keyboard key conflicts are eliminated. Facilities such as start and stop audio are important for different levels of system users and their comfort.

The image in Figure 3.12 shows that visually impaired users can produce meaningful drawing when given adequate technological support. The project aim was not to check if blind people can produce the same quality pictures as sighted people, but if they could draw what was intended. As discussed earlier, Kamel [Kamel and Landay, 2002] points out that blind art software should provide facilities to relocate and backtrack to the original points. The screen plan /floor plan should be easily memorised by the user or blind person, therefore the floor plan and the areas of division have to be simple and should be one single-digit number division to retain the floor plan in the working memory. He emphasises the relative and absolute position of an object so that it can relate to other objects on the interface.

Sense substitution is key when designing software for BVI users. Even though many senses exist and have shown promise, above research recognises that hearing is the more natural substitute for vision as it is also the best substitution for BVI users. One of the main goals of Kamel's software is to allow a BVI person to select a desired point on the screen, move the cursor to perform another task at a new location and later relocate the cursor back to the original position through a sequential set of audio instructions and actions. Figure 3.12 shows a picture of a square populated with an IC2D drawing connecting center grid points.

Creation of tactile graphics and pictures by BVI users need further exploration. Hearing has been revealed to be more of a suitable substitution for blind people in the absence of visual modality. Some of the important features of graphics production that were presented above are based on 2D-screen arrangements, give immediate feedback, have tagging, are compatible for screen reading, allow image change, and are easy to use.

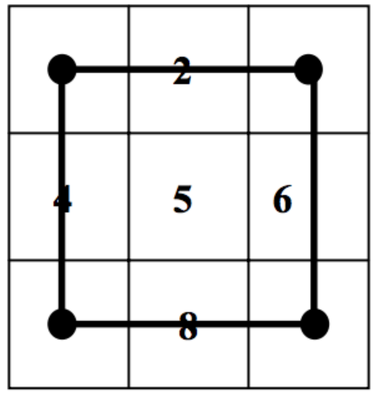


Figure 3.12: Use of a Grid System to Create a Square [Kamel and Landay, 2000]

Primitive Shapes as a Technique for Image Cognition

Kennedy [Kennedy, 1997] reported that individuals who have been blind since the age of three use many of the same techniques as sighted artists when it comes to outline drawing to produce blind art. Lines are used to represent surfaces such as in 3.9 to represent different views of a table. Blind people can show objects such as a house from different angles and use shapes to convey abstract messages, like a heart around a child to convey love surrounding that child. They also use foreshortening to show the placement of an object, so a closer object is larger in size and a further away object is smaller.

Primitive shapes such as points denote a unique location or an element of space. Lines extends equally between its points. Points and lines are the two most essential elements of outline drawing used among blind and sighted artists to convey the abstract shape of a real object. There are many ways to represent curves, a line or outline which gradually deviates from being straight, such as a closed circle, ellipse or a regular oval shape. A square (a plane figure with four straight sides and four right angles, especially one with unequal adjacent sides) and a polygon (a plane figure with at least three straight sides and angles) are some of the shapes which can be used as an abstract representation of real world objects. Shapes are also used in an academic context for diagrammatic drawings that have specific meaning that is used and defined with its context. Figure 3.13 shows some 2D and flowchart shapes used in everyday life.

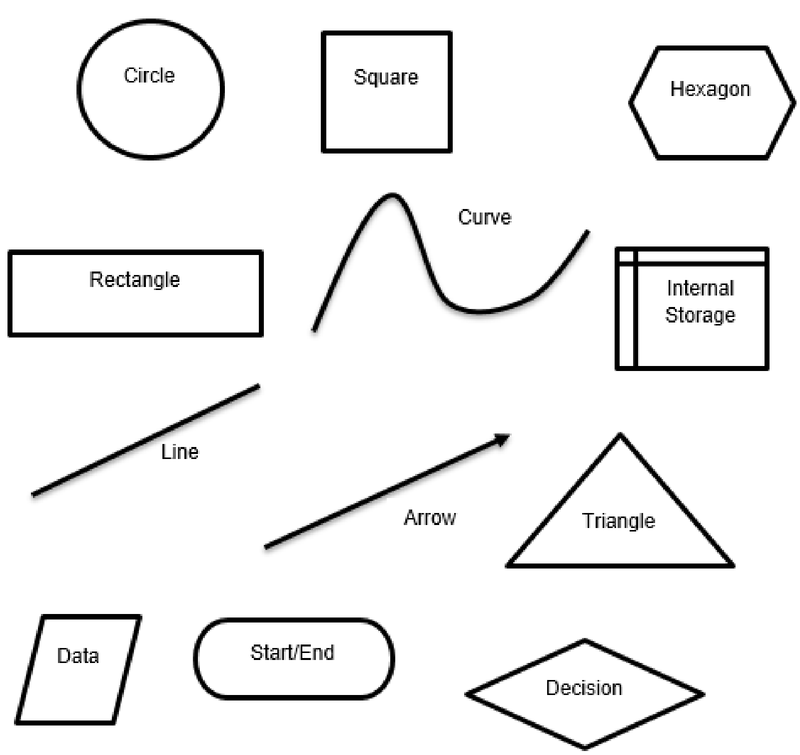


Figure 3.13: Shapes

Interpretation needs appropriate words to describe a phenomenon or concept, which could be subject to its context. Art in general is open to several interpretations without any of those being necessarily incorrect.

The best method to describe a picture as introduced by Yvonne Eriksson, from The Swedish Library of Talking Books and Braille [Eriksson, 2010] as listed below.

- Present the picture and give a general view of it before going into detail.
- Locate each element of the picture first before starting to describe it. This means explaining where the object depicted is to be found on the picture surface.
- Describe the picture in detail, once it has been presented and the different parts located. The information given should be extensive enough to give the pupil a notion of the appearance and/or function of the different parts.
- Make a summary of the picture. Ending up with a short summary is a good way of returning to consider the picture as a whole.

The next section presents a classification diagram for shape language that resulted from the investigation into mental models of blind people, blind drawing, evaluation of the current CADB tools, and benefits of sensory substitution.

3.1.6 Shape Language Considerations

Figure 3.14 shows the classification and direction of shape language consideration in a diagrammatic form. Perceptual models of blind people lead to outline drawing that can be created with CADB tools using primitive shapes. These shapes should be created, labelled, grouped, and defined in a language for shape production.

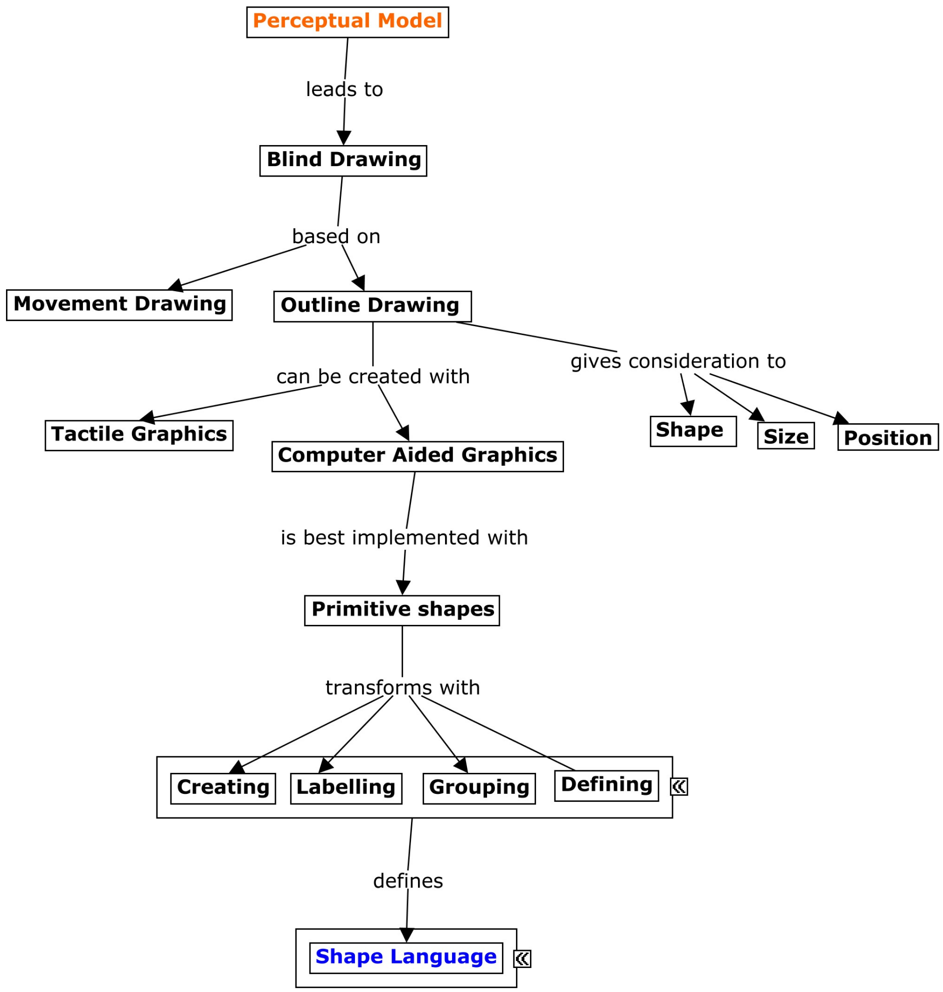


Figure 3.14: Classification Diagram of Shape Grammar

In summary, studies that have more weighting in art production are based on: (i) J.M. Kennedy's research findings mentioned above on line drawing, its application in producing art and to communicate ideas; (ii) G.A.Miller, P.N.Johnson-Laird, Marijin E. Struiksma work with language vision on to understanding objects, and linguistics to produce a better realisation of art and associated concepts using language; (iii) H. M. Kamel's experimentation with placing art on 2D space grid arrangement systems; and (iv) Mamoru Fujiyoshi's BPLOT tool with its command language introduction to diagram production. The above empirical studies and experiments have led to the introduction of a set of shape consideration when placing 2D shapes in a given space to construct diagrammatic drawings for blind people as listed below.

- **A method to recognise a point in space**: An element/notion in space that primitive shapes are built that have x and y coordinates in 2D space.
- **A method to create a line/lines between points**: A line has two endpoints where as line segments have at least two end points but can be extended indefinitely.
- **A method to create a curve/curves between points**: A quadratic curve is defined by two endpoints and one control point and a cubic curve is defined by two endpoints and two control points.
- **A method to produce primitive or arbitrary shapes**: Create two-dimensional primitive shapes made of points, lines, rectangles, arcs, ellipses, and curves, define a collection of shapes with a name/label to convey a meaning.
- **A method to group a set of shapes**: To assign a name/label to one or many shapes in order to recognise them later.
- **A method to reuse a new shape**: Call to use a created shape by its name/label.

In this section explored were issues relating to current blind drawing systems and the mental imagery mechanisms of blind people. As a result, shape guidelines are introduced. The next section explores blind people's navigation techniques and space exploration strategies and introduces spatial guidelines for a CADB system.

3.2 Issues of Navigation Techniques

Screen navigation can be a time-consuming activity for BVI individuals. Multimodal techniques are proven to be more effective when using a keyboard, stylus or speech [Suhm et al., 2001], but this is not always viable. Different screen navigation techniques [de Mauro et al., 2001], [Mcnair and Waibel, 1994], [Sears et al., 2001], [Kamel and Landay, 2000]] have been introduced in the past and some of them were based on route and map strategies that can compensate for vision when using computer screen navigation.

Mauro [de Mauro et al., 2001] introduced voice-controlled mouse navigation. Subsequently direction-based navigation resulted in discrete or continuous cursor movement that allowed faster and more robust experiences than traditional keyboards [Sears et al., 2001]. In discrete direction-based cursor movements a "move left two words" causes the cursor to move to the new location by jumping two words left. Discrete direction-based formulas can also be set as inches, centimetres or letters on the screen. Continuous direction-based navigation is mainly situated in speech input for start and stop commands. For example, "move right" moves the cursor to the right until the "stop" command is given.

The drawback of the above method is that the cursor moves continuously either in horizontal, vertical, or diagonal directions until a stop command is issued. If the stop command is delayed, then a higher degree of discrepancy results between the intended drawing and the actual artwork graphic realised. It is also difficult to judge distance or unit length using direction-based navigation.

Mcnair and Waibel [Mcnair and Waibel, 1994] investigated and introduced an early version of target-based speech navigation, as well as immersion speech-based navigation on web documents. A user says a word that serves as a link to other web pages or menus or by saying a number that eventually takes them to another link. With target-based navigation, a speech command "Select Friday" in the text document can directly navigate to the word Friday. It also can be set to navigate to icons, menus, and regions on the screen.

The drawback of target-based speech navigation is that there can be voice recognition errors a lot of the time and the cursor can end up in the wrong location. It has a longer task completion time, and there are also difficulties when there are a higher number of similar targets and the user needing to know the required targets.

Grid-based navigation methods for drawing were first studied by Kamel [Kamel and Landay, 2002] and tuning and magnification research was studied by Feng and Zhu [Zhu and Feng, 2010]. The grid-based navigation system manoeuvres to an area on the screen without contextual information, using 3x3 matrix system navigation techniques. The formation of a nine cell system works well with the intuition of BVI individuals as it is organised in a similar way to a telephone keypad. The cell numbers work from left to right and top to bottom. The system enables users to move to and from any of the nine cells. The unique point of a cell is the centre point which is then selected to perform drawing. A recursive grid schema allows users to make more precise selections. Users can find a unique position by further dividing into nine more cells and increasing zoom levels.

Grid-based navigation methods have proven to bring easy and quick navigation to the screen. This method has gone through several experimental modifications that include grid-based speech cursor control and magnifications. When the speech vocabulary is simple to follow the grid-based speech cursor works with ease, although a higher rate of success and accuracy was recorded with keyboard grid navigation.

Screen readers such as JAWS, NVDA, VoiceOver present content linearly to computer users as discussed in [WebAIM, 2019]. This is different from most people's use of visual content. Sighted users scan the content immediately, with information about layout, graphics, styles, and other finer level details of digital content. But a screen reader makes linear progression through content from start to finish, such as an automated menu option, in a step-wise style without disclosing all of the information at once. However, screen readers provide a linear style audio interface for users to steer through the content in different ways, such as TAB keys to navigate to different TAB content and heading to navigate to different content. Landmarks, page sections, paragraphs, skip navigation, tables, lists, buttons, links and images are all methods used as navigation pointers by screen readers.

The drawback of screen reader navigation methods is that webpages and applications are not always produced with linked content, headings, meaningful headings, tags, or standard element mark-ups.

3.2.1 Cognitive Models

This section describes mental models, conceptual level mapping and perceptual level mapping, where different substitution mechanisms and strategies are used by blind and vision-impaired individuals to compensate for situations where a lack of vision may pose difficulties.

Holst [Holst, 1999] speculated two alternative mental models, the first where users do not know what the screen looks like and they memorise a large set of options to achieve a given effect, and a second where the user has a pretty good cognitive or mental map of the screen. A study by [Kurniawan et al., 2003] indicates that users' mental models are a combination of these models speculated by Holst. According to Kurniawan, [Kurniawan et al., 2003] cognitive models are users' internal abstract models that have information about where things are located and how things work or behave. He points out that blind computer users' mental models are based on the use of day-to-day technology such as screen readers, like JAWS or NVDA, other devices such as Braille, or speaking software like Siri, [Apple, 2019]. However there are differences in mental models based on level of vision when performing spatial tasks. The section below further explores the navigation strategies of early blind, late blind, and partially sighted individuals.

3.2.2 Spatial Exploration Strategies

Ungar [Ungar, 2000] in his writing elaborates on different coping strategies used by sighted, blind and late blind individuals during different spatial tasks. [Millar, 1994] pointed out that vision has advantage over body-centred and external reference points when travelling. However, in the absence of vision people use egocentric references (comparing things relative to their own body) that work well with small space exploration but are still challenging in large space exploration as the body has to translate change and location. Late blind individuals perform spatial tasks more similarly to sighted people if the late blind people have had sight from three months to three years of age. Millar [Millar, 1994] explains different coding in haptic exploration. In large space exploration, early blind people tend to use body reference or movement reference points for accuracy. The drawback of haptic strategy is that it proves less effective when tasks are associated with mental reorganisation, mental rotation and spatial inference.

Sighted people tend to work with logic and reasoning to make decisions in simple and complex tasks but may not necessarily use calculation according to Hill [Hill et al., 1993]. They make decisions based on prior experience of their minds and representation, however individual decisions change when they are faced with new tasks to which they cannot relate with prior experience. "Response latencies" (time span between a stimulus and a response or reaction) of early blind individuals are proven to be higher than sighted individuals as early blind individuals tend to use calculation strategy (object to object, perimeter to object, or home base to object) when given a task which might suggest a trade off in speed as listed in Table 3.1.

Imaginative thinking has proven to be more superior in blind/ early blind individuals than in sighted or late blind individuals when doing tasks such as spatial memory calculation or way finding. This suggests that mental presentation is not directly based on vision but other factors, such as calculation strategies used by individual blind people's egocentric understanding and not yet proven factors. The study [Ungar, 2000] conducted on tactile maps suggested that tactile maps are a good representation of an external framework in the environment that cannot be understood by direct access alone.

Albert Postma [Postma et al., 2008] conducted a study to understand the contribution of visual experience in haptic and verbal modality task performance among early blind, late blind and sighted individuals. The results suggest that visual experience helps by using relevant stored visual images or by increasing the tendency or ability to employ visual imagery in certain aspects of haptic task processing where late blind individuals performed better than early blind individuals. However, Thinus-Blanc and Gaunet offered a view that in verbal modality tasks, the late and early blind individuals performed worse than sighted individuals due to the difficulty of complex computation with mental images [Thinus-Blanc and Gaunet, 1997]. This suggests that the coping strategies and imagery abilities of different blind individuals varies based on previous memory. Therefore, understanding the mental ability of different subsets of blind people is essential in building digital space exploration and screen navigation tasks.

Table 3.1 shows different navigational strategies observed in blind individuals when exploring space extracted from [Ungar, 2000] based on [Hill et al., 1993] and [Gaunet and Thinus-Blanc, 1996]. Exploring the boundaries of an area to identify the area's shape, size, and key features around its perimeter, by walking along the edge of the layout, is called perimeter strategy. Investigating the internal elements of an area to learn their spatial relationships, by taking straight-line paths from one side of the layout to the other, is called grid strategy. Moving repeatedly from one object to another or feeling the relationship between objects using the hand or a cane, is called object to object strategy. Moving repeatedly between an object and the perimeter, or moving repeatedly between the home base (origin point for exploration) and all the others in turn is called home base to object strategy. Each of the four objects visited in turn, and then returning to the first object, is cyclic strategy. And moving repeatedly between two named objects is called back and forth strategy.

Strategy	Description	Studies
Perimeter	Exploring the boundaries of an area to identify the area's shape, size and key features around its perimeter, by walking along the edge of the layout	Hill et al.
Grid	Investigating the internal elements of an area to learn their spatial relationships, by taking straight-line paths from one side of the layout to the other	Hill et al.
Object to object	Moving repeatedly from one object to another, or feeling the relationship between objects using hand or cane.	Hill et al.
Perimeter to object	Moving repeatedly between an object and the perimeter	Hill et al.
Home base to object	Moving repeatedly between the home base (origin point for exploration) and all the others in turn	Hill et al.
Cyclic	Each of the four objects visited in turn, and then returning to the first object	Gaunet and Thinus-Blanc
Back-and-forth	Moving repeatedly between two objects	Gaunet and Thinus-Blanc

Table 3.1: Navigation Strategies [Ungar, 2000]

Hill's [Hill et al., 1993] experiment also introduced features of three reference point strategies of space exploration as discussed earlier, such as object to object, object to wall, or object to start. They found out that blind people who used one of those were more accurate in direction estimates irrespective of their blindness. However further studies revealed that blind people also prefer to select cyclic search patterns, such as walking to the object sequentially in Gaunet's study [Gaunet and Thinus-Blanc, 1996].

Lahav [Lahav and Mioduser, 2008] explores on a conceptual level mapping to recognise and support different navigation paths followed by blind individuals in order to develop exploration strategies for the surrounding environment. Jacobson [Jacobson, 2013] explains two different techniques of indoor environment spatial strategies demonstrated by blind individuals. Perimeter recognition is where a blind individual walks along a wall and explores objects attached to the wall and grid-scanning is a strategy to explore the interior of the space as first introduced by [Hill et al., 1993] and [Gaunet and Thinus-Blanc, 1996]. His experiment introduced two basic strategies, namely route strategy for linear recognition and map strategy for multiple perspectives of the space. This confirms Ungar's basic mode of navigation strategies but with different naming categories [Ungar, 2000].

Lahav [Lahav and Mioduser, 2002] also explored spatial mapping of blind people. When mapping through a perceptual level the visual channel is compensated for other sensory substitutions e.g. hearing, smell or touch. Haptic (sense of touch) information has proven to be an essential means for spatial analysis and object recognition. Cane, palms, fingers, legs, all provide haptic information for blind individuals about their immediate surroundings, allowing texture-based information, object recognition, the presence of other people, materials that objects are made of, and estimates of distance within space [Hill et al., 1993]. Auditory-based tactile recognition is based on spatial landmarks, clues and information from the surroundings that act as compensatory sensorial channels, as an alternative to the visual channel. This suggests that people who are blind can benefit from information technology devices that provide spatial information such as verbal description, tactile maps, physical models, talking signs, and personal location guidance based on satellite communication.

Zagler [Zagler et al., 1994] introduced a concept, method prototype and audio framework to inform interaction between objects and the environment to support blind computer users. He introduced two approaches to produce non-visual interface mapping by conceptual (using an audio room concept to generate special sound) and perceptual (using other sensory modalities) structure when using applications such as Windows GUI interfaces. Non-speech audio clues are designed and produced to give variations to colour, saturation, luminescence, and texture by using audio qualities such as timbre, pitch, amplitude, reverberance, locality, phase modulation and other sounds. Just as graphical objects resemble everyday objects in a Windows user interface, so sound icons are used with association to everyday sounds. Such sounds include tapping on a glass pane to represent touching a window, a menu search creating a series of shutter sounds, push button sounds used for radio buttons and the sound of a typewriter to represent touching text. Zagler proposed and experimented with a primitive sound class on a highly semantic level that needed a low rate of storage. The sound generated acts as synthesised sounds for parameterised auditory icons for blind computer users' interfaces. This suggests that perceptual level mapping can take many forms such as audio, haptic and olfactory. Amongst these, audio has proven to sit well with the early blind individual in comparison to late blind and sighted people. The next section explores the mental models of the blind user in a Windows environment and their screen familiarity, perceptual familiarity, and structural properties.

3.2.3 Mental Models in Computer Environment

[Kurniawan et al., 2003] investigated blind users' mental/cognitive models in the Windows environment and their coping mechanisms. It was found that there is a clear relationship between their adaptability to new systems and preconceived mental models. Kurniawan categorises three different mental models based on experiment and observation. Blind users with a structural mental model perceive the desktop environment as strict columns and rows. Users with a functional mental model identify the Windows environment as a set of functions and commands, and users in this category are not concerned with interface layout. Some blind users associate functional commands and structure in the Windows environment, and this is identified as a hybrid mental model. Blind users explore, take action and make configurations during their interaction with new software as their coping mechanism. The research reports that blind users were not receptive to unfamiliar audio feedback and the operational style of the software model. This reveals that the interface of the models need to be framed around styles familiar to the user. Saei [Saei et al., 2010] extends the mental model analysis further by introducing other contributing factors for system designers such as skills-based, knowledge-based, domain-user expert and system help to improve BVI users' satisfaction.

Fakrudeen [Fakrudeen et al., 2013] proposes a mental model with enhanced usability features by taking touch screen users into consideration. The research puts forward eight new HCI features that add on to already existing features, covering the areas of feedback, error correction, undo and help functions. These newly suggested usability features are audio quality and user characteristics that impact on effective listening and interaction. Environmental factors such as noise, sweat, hand and finger movement during interaction, minimal categories of content and sub-content, orientation, gesture and device are the other important mental model usability considerations for haptic system design for BVI users. Their report suggests that the touch screen usability features for the blind should be largely similar to the design of touch interfaces for sighted users in terms of usability.

An investigation was carried out to compare three groups of non-visual sources by Afonso [Afonso et al., 2010], to investigate structural properties of spatial mental images created from non-visual sources. One of the experiments checks if different groups can create metrically accurate spatial representation under verbal and haptic learning conditions among all three groups, and another experiment checks the ability to generate accurate spatial mental images after both verbal and locomotor learning created by an audio virtual reality system. The experiments evidenced that blind and blindfolded participants created metrically accurate spatial representations of a small-scale special configuration. Late and congenitally blind participants generated accurate spatial mental images with audio virtual reality technology, whereas blindfolded sighted participants were impaired in their ability to generate precise spatial representations from locomotor experience. Their results reveal that motor system of users is crucial for building metrically accurate representations of the spatial environment. The report also highlights the potential of audio-rendering technology for exploring the spatial representations of visually impaired participants.

Mynatt [Mynatt, 1997] introduces tree structure style screen navigation when transferring graphical information into auditory information for blind users. Her recommendations are for access to functions where (i) spatial and structural interfaces are mapped, (ii) iconic representations of interface objects where sounds are based on concepts in graphical interfaces, (iv) structural organisation where grouping of objects is conveyed with clues by object group, (v) direct manipulation where users interact directly with objects, (vi) spatial arrangement where layout information defines the order of the auditory interface, and (vii) relative size indicated by auditory icons and persistent presentation with parts of the auditory cues that help to convey portions and location of the interface in later stages. These recommendations give useful pointers to audio clue generation during hierarchical screen navigation.

Grid-Based Cardinal Direction and Points

Since the first introduction to spatial navigation of blind people by [Hill et al., 1993] and later by [Lahav and Mioduser, 2008], grid-based techniques were employed by scientists in screen navigation. Grid-based screen navigation systems have been experimented with and are well recognised as the natural navigation mechanism for people who are blind as the technique is related to telephone keypad style communication. [Kamel and Landay, 2002] introduced a numeric grid-based screen layout, identifying screen areas with numbers for navigation, [Meliones and Sampson, 2018] developed an indoor navigation system using a floor grid and cardinal direction-based turning points, Fitzpatrick

[Fitzpatrick and McMullen, 2008] proposed a row-dot numeric identification system for audio-tactile production to improve readability of graphics as part of the AHVITED project, [Silva and Wimalaratne, 2017] used grid-based obstacle localisation employing sensors, and [Kassim et al., 2016] invented positioning and localisation with a digital compass and direction guiding through voice commands.

In grid-based approaches, a grid overlays the space/screen to fix the number of cells at the beginning based on screen resolution. The user can recursively drill down using this technique through each cell until the mouse pointer is in a desired location. The cursor is placed on the centre of the cell and a particular reference point can be obtained by x and y axis.

However, one cursor point may not be sufficient when producing artwork where multiple reference points are needed within a cell to avoid recursion. Hence, a mechanism is needed for multiple reference points in a cell. Such a technique would enhance outline drawing of primitive shapes as explained by [Kennedy, 1997] in the section Blind Drawing.

The more reference points a cell has the better, so the user can draw different shapes. Multiple points in the cell increase the cell reference than a single cell point reference in grid-utilisation. Multiple points enable further enhancements such as magnification and fine-tuning to the grid- based approach to obtain clarity on recursive screen cells as recommended by [Zhu and Feng, 2010].

A meaningful name can be given to the reference points from concepts familiar to blind people in their everyday use. Previously introduced AT technologies, and AT devices incorporate cardinal directions (North, South, East, West) with GPS (global positioning systems) to guide blind people in different indoor and outdoor environments. Hence cardinal names can be natural and familiar concept to blind people.

3.2.4 Spatial Language Considerations

Figure 3.15 shows the classification diagram of spatial language. It starts with cognitive models of early blind individuals. Then the identification goes into spatial mapping techniques that have perceptual and conceptual levels. Conceptual level mapping needs exploration strategies identified as perimeter, grid and object-to-object. The grid strategy is then explored with the possibility of screen navigation with the recognition of points, locations and cardinal directions that are established in above research and lay the foundation for spatial language in this thesis.

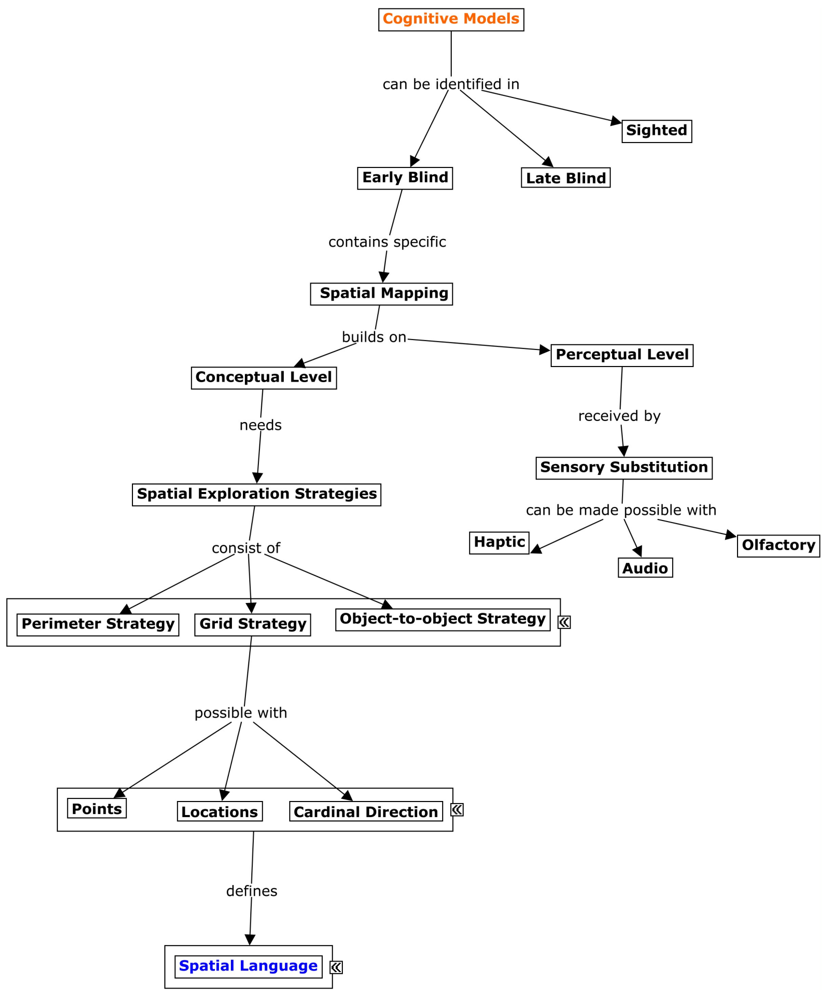


Figure 3.15: Classification Diagram of Spatial Grammar

A navigation language consideration needs a technique to navigate, reference, backtrack and remember a position. The reference is identified as a naming system to locations and points during system/user dialogue. Below are the spatial language considerations for a CADB system.

- **Reference to a navigation route**: Ability to inform user position with reference to the starting location; a position on the screen with an effective referencing system of the route.
- **Reference to a location** : A location on the screen with an effective referencing system to identify the location.
- **Reference to a point**: A point on the screen with an effective referencing system to identify different points in a cell.
- **Technique to navigate to a location**: Effective linguistics to convey the action of transition.
- **Technique to navigate out of a location**: Effective linguistics to convey the action of relocation.

This section explores the issues relating to current screen navigation techniques and mental models in the computer environment and different spatial exploration strategies of blind people. An investigation was conducted to find out the navigation of blind people's spatial exploration strategies. It was found that the grid strategy has multiple benefits. Investigation into points, location, cardinal directions were needed to navigate using a grid strategy. As a result, suitable spatial guidelines were introduced that provides a reference to a point, location, route on a grid interface. This guideline links to the candidate's own work of space language creation in section 4.3.1. The next section explores issues and recommendations of usability in assistive technology design.

3.3 Issues of Usability in Assistive Technology

This thesis has covered a range of usability issues in technologies for blind people, both in Chapter 2 and so far in Chapter 3. In Chapter 2 usability issues were mainly found in reading technologies, tactile technologies, speech technologies and haptic technologies, whereas more specific issues are addressed in Chapter 3 related to drawing and navigation technologies. One of the main concerns is the non-inclusion of accessibility in usability design, i.e. software that is not designed to be compatible with screen readers, or features that are heavily based on vision. Some blind user-targeted systems are useful and widely in use, but they are not digitally portable, data is not stored or reused, and they are missing many usable auditory or haptic features. This thesis has also explored systems that are designed for blind people but it is found that these systems are not efficient and effective even with the inclusion of auditory, speech or haptic technology due to insufficient user understanding and poor technical and design decisions in their development.

One of the main reasons for the failure of digital software for blind people is that not enough design consideration is given to user ability and disability as a result of not following mixed design approaches that are flexible and formal. This section further explores some usability-specific issues found and then discusses international guidelines, recommended frameworks, and taxonomies for accessible software design.

According to the experiment conducted by Charlotte Magnusson [Magnusson et al., 2009], the pedagogical barriers in teaching BVI children to draw is limited by the special knowledge of teachers and assistants. They found out that teachers were very focused on teaching and transferring knowledge in graphical form and would often record that maps and mathematical figures were one of the biggest problems in knowledge transfer. The HIPP drawing application [Rassmus-Gröhn et al., 2013] is written in C++ and Python with H3D API and Cairo graphics that uses haptic and sound feedback to draw and display information. The system enables pupils to do doodle drawing as far as they can hold a pen. Then the drawing is discussed with their teacher and they explore shapes, scaling and perspective. This type of drawing exploration and discussion develops knowledge and an interest in art with a playful approach that has proven to work well with school pupils. Ungar [Ungar, 2000] points out that the inability of educators to keep up to date with new research stands as a barrier to understanding external frameworks of reference. Ungar has discovered that tactile maps are not utilised extensively in classrooms and the outside world, and there are only a few standards for software designers to refer to. This section leads to a need for haptic and sound feedback for activities such as drawings.

Charlotte Magnusson [Magnusson et al., 2009] also conducted a survey of non-visual mobile navigation systems focusing on blind and partially sighted users. The experiment leads to a need for landmarks, that is information about the objects in the environment, position accuracy, speech feedback and conformation. Correct user orientation and map content were some of the recommended features that are integrated in to this research SETUP09. The experiment was mainly conducted for mobile navigation, hence some of the main suggestions are hands-free and eye-free solutions that need the designer to adopt auditory description techniques to navigate.

Bocconi [Bocconi et al., 2007] writes about variety of obstacles in accessibility and usability in the process of ICT education and law in force in the European and non-European countries. Such as online based E learning tools, computer based learning and M-learning tools that are available through mobile devices tablets and, personal digital assistance, blind students need screen readers and visually impaired students need optical aids and special customisations to access information. The customisation may come in many forms or shapes based on individual diagnosis. It can include larger fonts, different layout of information, different menu items, larger graphics or tagged graphics and keyboard driven access. It is found that web-based systems often follows [W3C, 2018] guidelines hence they are less difficult. But some standalone applications may not take screen positioning into consideration and hence lose part of the learning or increase the cognitive load of the activity. Some systems are designed with crowded screens, difficult screen partition, and rigid text based pages that makes it difficult to use.

The issues of accessibility are not limited to the aforementioned. Accessibility issues can be attributed to a number of factors including: (i) limited special knowledge of support workers or teachers, (ii) software that is not compatible with screen readers or a lack of personalisation, and (iii) facilities that are designed without exploring the full functional requirements of BVI people or which are modality bound. Some of the common suggestions are that navigation software should contain speech feedback and give consideration to screen position and objects in the environment. Apart from these common recommendations for BVI software development, there are specific usability models that guide assistive technology designers as explained below.

3.3.1 Usability Models

The ATOMS project conducted by the University of [Wisconsin-Milwaukee, 2015] and the literature presented by [Erdem, 2017] various types of assistive technology models and taxonomies relating to assistive technology resources. The ATOMS project provides over 50 various theoretical taxonomic support models for future modelling in the area of assistive technology development, whereas [Erdem, 2017] discusses the contribution of various frameworks of assistive technology that are used in special education.

The HAAT (Human Activity Assistive Technology) Model [Cook and Polgar, 2015] was developed to demonstrate basic factors that affect assistive technologies. Individual performance is identified by four main areas which are human (senses, processing, motor skills), activity (performance in a certain environment), assistive technology (technological or environmental features) and context (environment, social and cultural models). When designing a curriculum using the HAAT model, the designers can question the content, task, and assistive technologies, already established methods, knowledge of practitioners, and the length of the implementation addressing the changes to the environment, technology, and process.

The Wisconsin Assistive Technology Initiative (WATI) process, from Assistive Technology Consideration to Assistive Technology Assessment, [WATI, 2018] makes training and technical assistance available for students to access services, the curriculum, school and community activities through the use of assistive technologies and is for children with disabilities up to the age of 21. They provide provisional, regional and local training, local technical assistance, create access to resources, and develop collegial support and technical assistance across the state.

SETT Framework, [SETT, 2002] obtains information to do with the student, their environment, the task they are undertaking and tools they will need, in order to make better decisions for people with disabilities when determining assistive technologies. The following questions are answered when making effective decisions on assistive technologies: (i) What are the special needs and skills of students, (ii) What activities will students carry out, (iii) What are the educational and physical arrangements, (iv) What particular equipment is used in the education environment, (v) What support is required for students and support workers, (vi) What activities and elements of activities will students have to do, (vii) What kind of assistive technology is needed to perform the task, and (viii) What strategies are needed to improve the performance of the student.

Quality Indicators for Assistive Technology, [QIAT, 2012] include quality indicators and indicator matrices. Quality indicators are namely: assistive technology needs, assessment needs, assistive technology in IEP (individualised educational programmes), documentation, implementation, evaluation of effectiveness and professional development. The matrices are developed for all of the indicators so that service providers can assess their teaching and delivery practice and plan for improvement. Example criteria for an AT needs indicator matrix are below.

- Assistive technology (AT) devices and services are considered for all students with disabilities regardless of type or severity of disability.

- During the development of the individualised educational programme (IEP), the IEP team consistently uses a collaborative decision-making process that supports systematic consideration of each student's possible need for AT devices and services.

- IEP team members have the collective knowledge and skills needed to make informed AT decisions and seek assistance when needed.

- Decisions regarding the need for AT devices and services are based on the student's IEP goals and objectives, access to curricular and extracurricular activities, and progress in the general education curriculum.
- The IEP(individualised educational program) team gathers and analyses data about the student, customary environments, educational goals, and tasks when considering a student's need for AT devices and services.
- When AT is needed, the IEP(individualised educational programme) team explores a range of AT devices, services, and other supports that address identified needs.
- The AT consideration process and results are documented in the IEP and include a rationale for the decision and supporting evidence.

Mentioned below in Table 3.2 is the great majority of educational software application needs covered by Italian law 4/2004 [Italy, 2019] for educational applications with regards to the two main categories of visual impairments. Table 3.2 breaks down main requirements for educational software applications and their basic correlation with the two main categories of visual impairments. Requirements impacting blind computer users specifically are: The functions provided by the user interface must be able to be activated by means of keyboard commands in cases where a description of the function or the result of executing it can be provided; Commands and functionalities of the user interface must not limit or disable the accessibility functionalities of the operating system, made available by the manufacturer of the operating system; The application must provide sufficient information, such as identification information, operations possible and status, on objects contained in the user interface so that the assistive technology can identify them and interpret their functionalities; Textual information must be provided using the functionalities of the operating system provided to display text; Textual content, the location of the insertion point and the text attributed must be available; The active "focus" element of a user interface must be clearly identifiable; Identification and variation of the focus must be indicated at the application programming interface (API) level, so that the assistive technology can manage them. Other elements that require action by the user must also be adequately indicated, and supporting documentation for the product and accessibility characteristics must also be made available in accessible electronic form.

The Italian law 4/2004 [Italy, 2019] leads to a need for keyboard commands whose software is (i) not tightly coupled with the operating system, (ii) sufficient information provided in the user interface, (ii) the cursor is a point of active focus, and (iv) location of object given. These requirements have been integrated within the proposed SETUP09 guidelines and software, refer to chapter 4.

Italian Law 4/2004 requirements, Agency [Italy, 2019]	**Blind**	**Low Vision**
Req. 1.The functions provided by the user interface must be able to be activated by means of keyboard commands in cases where a description of the function or the result of executing it can be provided.	yes	yes
Req. 2. Commands and functionalities of the user interface must not limit or disable the accessibility functionalities of the operating system, made available by the manufacturer of the operating system.	yes	yes
Req. 3. The application must provide sufficient information, such as identification information, operations possible and status, on objects contained in the user interface so that the assistive technology can identify them and interpret their functionalities.	yes	yes
Req. 4. Where graphic symbols are used to identify controls, status indicators or other programming elements, the meaning assigned to these symbols must be consistent within the framework of the whole application, including the user interface.		
Req 5. Textual information must be provided using the functionalities of the operating system provided to display text. Textual content, the location of the insertion point and the text attributed must be available.	yes	yes
Req 6. Applications that use audio signals must provide an equivalent visual functionality, in accordance with any conventions of the operating system.		
Req 8. Applications must not overlap selections made by the user with regard to contrast or colour levels and other display attributes.		yes
Req 9. The user interface must not contain flickering text elements, objects or other elements with a intermittent frequency greater than 2 Hz and lower than 55 Hz.		yes
Req 10. The active "focus" element of a user interface must be clearly identifiable. Identification and variation of the focus must be indicated at the application programming interface (API) level, so that the assistive technology can manage them. Other elements that require action by the user must also be adequately indicated.	yes	yes
Req 11. Supporting documentation for the product and accessibility characteristics must also be made available in accessible electronic form.	yes	yes

Table 3.2: Main Requirements for Educational Software Applications for Visual Impaired and Blind People [Bocconi et al., 2007]

CAT Model

The "CAT " (Comprehensive Assistive Technology) model was introduced by Marian Hersh in 2007 [Hersh et al., 2008] and provides an emphasis on designing assistive technologies to change the quality of life of deafblind people. The proposed model contains many layers and this research falls into the category of the cognitive and affective activities and skills required by a blind person in transmitting creative and imaginative thinking and scientific diagrams. Figure 3.16 demonstrates a section of the comprehensive assistive technology model by Hersh that is associated with activities such as scientific-technical drawing and expressing imaginative thinking through computers, which are some of the assistive technology needs in education and employment that are supported with this piece of research.

The context aspect of the model takes a wider view of all social and cultural factors, including language. The user's manual and speech output should be considered using the end user's local language. State legislation regarding accessibility and social inclusion, availability and reliability of

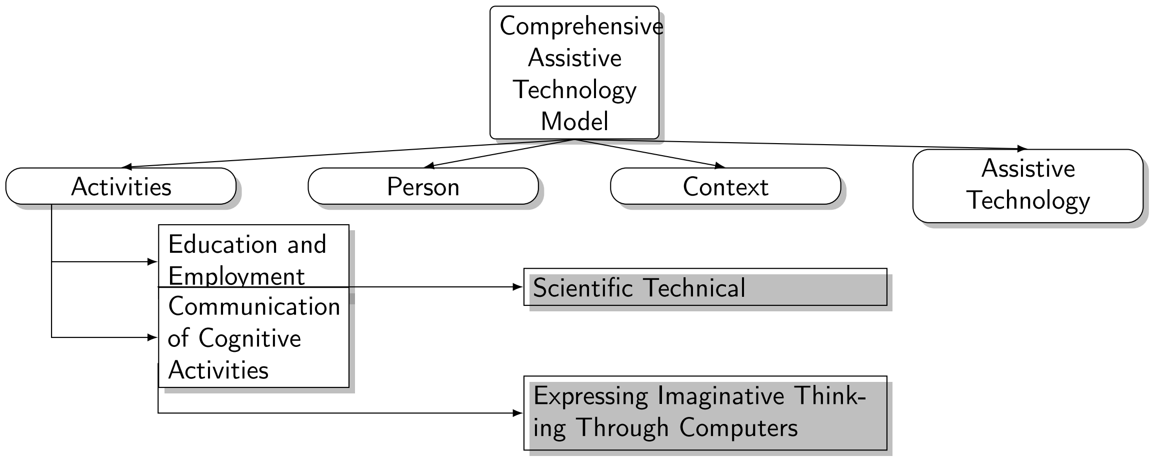


Figure 3.16: Modified using CAT model [Hersh et al., 2008]

technology, location and the environment and its accessibility by blind people should also be taken into consideration.

The person aspect of the model includes age, gender, ethnicity, origin and physical and mental health should all be considered when designing a system. AT design should not let users lose their self-confidence or self-esteem, which is important for the success of assistive technology. Considerations such as the user's attitude, their experience with technology, how willing they are to try new technology, their self-identity and degree of perseverance are important when designing an AT tool.

The activities aspect of the model is to support visually impaired people with activities such as education and communication of imaginative thinking. Imaginative thinking is a cognitive activity. UK charities and other publications show that blind people have low self-esteem about drawing and their natural instinct is that art/drawing is impossible. But research shows [Ungar, 2000] that blind people can visualise the world and they can build their mental maps similarly to their sighted counterparts. There shouldn't be any prejudices about mental imagination or the model of the world which one perceives. And the very fact that they have the ability to visualise can help.

Some assistive technology systems related to scientific drawing or art production software are not yet available so blind people find it difficult to carry out related activities which can lead to low self-esteem as found in our survey mentioned in section 3.4 When designing an AT tool one should consider the new system's sub-tasks, the user's physical and sensory requirements, and cognitive demands made on the user such as the availability of a sequence of instructions for a limited period of time, and memory capacity.

The design approach should consider the target audience, maintainability, and expandability for wider communities. In the case of technology selection the system should look into technology availability, reliability and cost. The system specification should have multiple modalities such as speech, braille text and deafblind manuals as shown in Figure 3.17. Ease of use and attractiveness, with a user-friendly interface and intuitive use of the system, and learnability always improve a system's popularity and success. Figure 3.17 shows possible communication options that can be input and

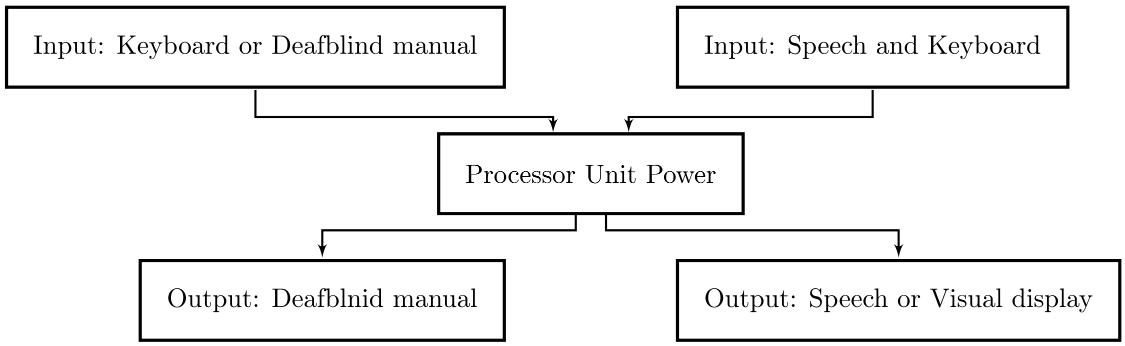


Figure 3.17: From the CAT model, possible communication options [Hersh et al., 2008]

output by Hersh's model. Inputs are a keyboard with deafblind manual or speech with keyboard, and output include deafblind manual or speech with visual display.

ICF Model [WHO, 2018]

A Classification Based on ICF (International Classification of Functioning) for Modelling Human Computer Interaction, [ICF, 2019] describes a need for an international classification for functioning, described in Figure 3.18 for disability and health that helps to understand a person's specific needs. The aim of ICF is to provide a scientific basis for understanding health, provide a common language for health and related descriptions, enable data comparison, and provide a systematic coding scheme. One of its goals is to improve communication across sectors to facilitate the exchange of information, to avoid duplication or contradictory mechanisms in service delivery that take functioning and disability into consideration. Figure 3.18 shows human-computer interaction based on the international classification of functioning. When performing activities such as accessing educational materials, orientation, mobility and household activity, people with health conditions to do with visual acuity, visual fields and light sensitivity will have to be provided with a solution with environmental factors and personal factors. These could be optical and electronic solutions, specialised hardware and software tools and assistive technology software.

Billi [Billi et al., 2006] explains an extension to the human ability component in human-computer interaction. It explains the changing nature of user ability based on the actual environment and context of use rather than static ability and disability status. Hence the focus on the user action. Two areas are identified in improving access to and production of information. Comprehension is the area that groups the receiving and understanding of information and communication in every possible form, and expression covers producing information or communication in every possible way. Coding of the interaction characteristic of devices has also been introduced which enables the characteristics of devices to be questioned by their intended users rather than by intended service operators. It has features such as composition capabilities, preference profiles, media features, media queries, and user agent profile-based vocabulary. The classification of device/health status is designed to work in parallel with user characteristics by using a simple coding set. The extension is helpful in situations such as

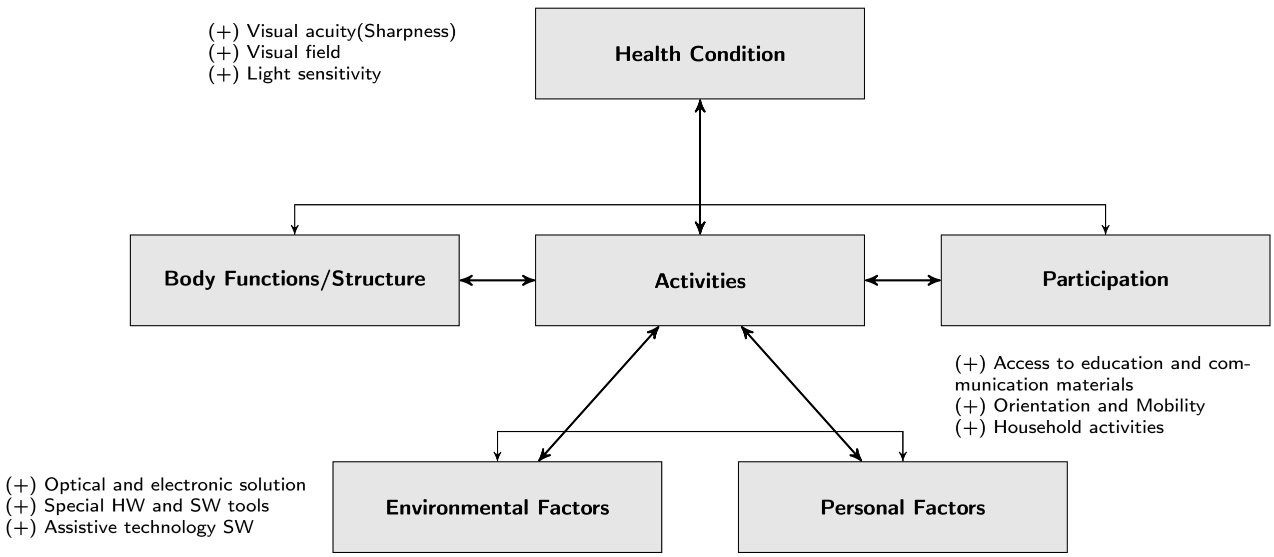


Figure 3.18: International Classification of Functioning [ICF, 2019]

the use of modelling human-computer interaction, taking actual user capabilities into consideration with environmental and situational constraints and the health status of the device determined by the environment and by availability of facilitating modifications. These parameters give service providers the best way to accommodate the requested service effectively under all possible conditions available.

non-functional requirements of Usability

Some of the non-functional requirements identified by Ferrera [Ferreira et al., 2012] align usability requirements with the accessibility guidelines focusing on the visually impaired. Some of the general accessibility features relating to information presentation /data entry include:

- Consistency , Feedback , Different ability level and human behaviour , Human perception , Metaphor , Minimise Memorisation , Efficiency on dialogue, movement and thoughts , Command functional sort , Direct manipulation , Exhibit only information essential to the context , Meaningful labels , Resolution independency , Helpful facilities , Minimise errors , Error recovery

These non-functional guidelines are not only applicable for AT systems for BVI but also appropriate for CADB system design in terms of its feature integration.

The above-mentioned usability model explores the design guidelines necessary to achieve high quality AT software. The HAAT, WATI, SETT and QIAT models inform various standards to be followed in educational AT software design. The CAT model addresses different areas of AT design categorisation which cover overall quality of life, whereas the ICF model classifies AT needs based on a person's disability and health. All of these models provide an understanding of where AT fits into the broader spectrum of a BVI person's life and give broader guidelines to design different forms of AT. The section below elaborates on various standards set by ISO with regards to assistive technology design.

3.3.2 Design Consideration of ISO Standards

The list below sets out ISO standards for usability, design, and measures that are taken from [Bevan et al., 2016]. These standards inform regulatory requirements for AT software design and usable product quality measurements. They also reflect the previously discussed recommendations of the different usability models.

- ISO 9999: Assistive Products for Persons with Disability
- ISO 9241-11: Usability: Definitions and Concepts
- ISO 9241-220: Processes for Enabling, Executing and Assessing Human-Centred Design within Organisations
- ISO/IEC 25066: Common Industry Format for Usability
- ISO/IEC 25022: Measurement of Quality in Use
- ISO/IEC 25023: Measurement of System and Software Product Quality

The scope of standardisation for assistive products for persons with a disability, ISO standards [ISO, 2011], means that those products which require the assistance of another person are classified in a way that is recognised by manufacturers and developers. It classifies assistive technology for drawing and writing within the classification hierarchy of class 22 (level one) and subclass 12 (level two). The standard ensures that assistive products are safe and usable for the target audience. The CE (European Conformity) marking process takes users' needs and regulatory requirements into consideration when following the production process. The CE marking process and the ISO confirm the standards and rules for products sold in the European economic area and they define the health, safety and environmental requirements which ensure consumer safety. Manufacturers must follow these guidelines during product design, risk analysis, technical tests, clinical evaluation, documentation, product registration, affixing CE marking, marketing, surveillance, and vigilance. As a result users are protected from harmful or ineffective-products that are not ISO certified [Borg, 2015].

ISO 9241-11, Usability: Definitions and Concepts, explains usability as "the extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use". The standard has been extended with a few more criteria. These include protection from the negative consequence of system use, such as incurring a high cost for the accidental selection of a feature/option, like the roaming facility on your phone, or the unnecessary use of energy that might harm health or the environment. Also protection from an incorrectly perceived outcome, such as booking an online service but not getting a conformation and therefore booking it twice, or thinking you submitted a form but it was not successful and you didn't realise so didn't re-try. User satisfaction has been reintroduced into the standard as it affects user experience along the lines of the cognitive, affective, and psychomotor responses of an individual that result in positive attitude, emotions, and feelings of security. Wide applicability is another extension that has been added to the standard, and this can include learnability or increasing efficiency with regular use, which gives accessibility for users with a wide range of capabilities, and maintainability.

Processes for Enabling, Executing and Assessing Human-Centred Design within Organisations, ISO 9241-220 elaborates on an approach to systems design and development that aims to make interactive systems more usable by focusing on the use of the system and applying human factors/ergonomics and usability knowledge and techniques to the design and development process . A human-centred design process and its categories are discussed in three main sections: ensuring enterprise focus of human-centred quality at strategic level, enabling human-centred design across projects and systems as an organisational infrastructure, and executing human-centred design with projects when introducing, operating and retiring a system [ISO, 2019] .

Common Industry Format for Usability, ISO/IEC 25066: This Usability Evaluation Report focuses on the content that should be included in a wide range of approaches to usability evaluation, such as examination of the facilities in the physical environment, an evaluation of the technical environment and administration tools under the observation of user behaviour, and users' performance measurements.

ISO/IEC 25022: Measurement of Quality in Use suggests the ability to measure for freedom from risk apart from standard measurements. Risks can be economic risks (on return on investments, benefits and revenue), related to health and safety, or be environmental. Flexibility measures take account of circumstance, opportunities, and individual preference. The software or product must have the capability to support or adapt for new types of users, tasks, and environment with stability and individualisation.

Measurement of System and Software Product Quality, ISO/IEC 25023: This standard measures the system and software product quality. It contains usability measurements that go beyond standard system measurements by including other parameters such as appropriateness, recognisability, learnability, operability, user error protection, user interface aesthetics and accessibility feature types discussed below [ISO, 2016].

- Appropriateness - Description completeness, Demonstration coverage, Entry point self descriptiveness
- Learnability - User guidance completeness, Entry fields defaults, Error message understandability, Self-explanatory user interface
- Operability - Operational consistency, Message clarity, Functional customisability, User interface customisability, Monitoring capability, Undo capability, Understandable categorisation of information, Appearance consistency, Input device support
- User error protection - Avoidance of user operation error, User entry error correction, User error recoverability
- User interface aesthetics - Appearance aesthetics of user interfaces

- Accessibility - Accessibility for users with disability, Supported languages adequacy

The aforementioned ISO standards ensure that the product is safe and usable considering ergonomic, health and measurement of software quality factors. The section below further explores the faculty of programming in software that gives consideration to structural representation and dimensional bases for effective, usable programmes that are useful in language creation.

3.3.3 Cognitive Dimensions of Notations

The Cognitive Dimension of Notations Framework, [Green, 1989] is widely used by researchers and product designers to analyse difficult areas of human-computer interaction (HCI). [Green, 1989] explains a system as a collection of notations and environments in the context of cognitive dimensions. A notation is explained as the programming language space that needs to be tied into user activities very closely, and the word environment explains other interactions or interactive programmes that are coupled with the main programmes. Cognitive dimensions can be applied to declarative and generative languages. It defines how the preferred cognitive style can be applied to user interface design. It encourages opportunist planning (where actions can be thought in any order) with notation rather than top-down design strategy. However, hidden dependencies between objects or structures limit HCI design as listed below.

- Rebuilding the programme in a different structure
- Disable the recreation of appropriate mental chuck
- Non-identification of hard mental operations
- Other dimensions such as inconsistency
- Lack of perceptual clues leading to rigid forms and structures.

Thus maintaining one dimension may hinder another, hence a careful user model analysis is needed in structural and environmental designing, having provision to the growth of unrecognised dimensions. Charles [Wood, 1993] conducted an investigation into external representation of cognitive activity. The recommendations of his findings include the following features of an interface that has less cognitive demand.

- Production takes less effort
- Grasp of the structural representation take place without much effort.
- Ability to change the structure when required without system rigidity.
- Use of graphical marks giving meaning with ease.
- Ability for the user to access system information with ease

- Ability to explore the full environment before committing to the order of the system.
- Less resistance to edit, display of history and current state.
- Representation to carry meaning without being heavily dependent on context for clarification.
- Have great power of meaning of the presentation that improves the above-mentioned characteristics.

As a result Charles [Wood, 1993] organised data into cognitive relevant properties such as:

- Delay gratification
- Terseness
- Perceptual clues
- Accessibility
- Premature commitment
- Viscosity and formalness

Cognitive Dimension (CD's) of notation framework [Blackwell and Green, 2003] also emphasises the need to go beyond external representation and consider vocabulary and design criteria based on the community of developers for the purpose of the activity in different disciplines. Notational dimension introduces four main areas of components to be considered when designing a system as stated below.

- Interaction language/notation (letters, music notes, graphic elements),
- Environment to edit notation(features of different editors),
- Medium of interaction(visual, auditory display)
- Kinds of sub-devices (helper systems with different notation dimensions)

Notational dimension clearly complements cognitive dimension by introducing the following features.

- Flexibility in resistance to change.
- Ability to view components.
- Constraints on the order of doing things.
- Transparency links between entities.
- Invitation to mistakes.

- Role of the system structure.

- Abstraction mechanisms.

The above mentioned secondary notation of the notation framework explains dimensions such as how close is the notation to the purpose for which it is created, expression of similar semantics in one form, non-consumption of the working area, cognitive demand, commitment to the order of system navigation or art production and progressive evaluation. Cognitive dimension, notational dimension and secondary notation dimension inform the need to design systems with particular attention paid to explaining these aspects. Hence cognitive friendliness of a system plays a vital part for BVI users and should be considered in the development of a new language or system design.

3.3.4 Design Considerations of Blind Drawing Technologies

Standards have been introduced and established by ISO as mentioned at the start of this section, such as ISO 9241-110 and ISO 9241 210, which relate to the ergonomics of human-system interaction. The list below by [Bornschein and Weber, 2017] explains very specific HCI features for non-visual drawing applications according to ISO-9241:110. The list has been devised based on empirical studies, experience and ISO standards and it is divided into four main parts. The current non-visual drawing systems are then given a rating based on the listing. The rating was undertaken by blind individuals. It was found that some of these features could also be changed based on the nature of application, such as some applications use free hand drawing and others use computer-aided graphics manipulation, because requirements may place high importance on some features and other features may not be so important. Hence the priority changes are based on personal experience, familiarity, use and purpose.

1. General feature
    - Independent creation
    - Allow for collaboration with sighted users
    - Intuitiveness and easy to learn
    - Direct and continuous feedback
    - Ability to individualise different input and output modalities
    - Compatibility with other assistive technologies
2. Interaction and Input
    - Support to draw accurate shapes
    - Availability to predefined primitives
    - Orientation aids
    - Keyboard-based interaction

- Stepwise interaction (property changing ability)
- Protection of critical function (deletion of erase all)
- Support with bimanual interaction(draw with one hand and feel with the other)

3. Functions and Features

- Allow for free hand drawings
- Allow to change and modify objects (resize, transform)
- Allow to edit small grid values
- Allow to erase element
- Support for connecting lines
- Labelling elements
- Error correction (undo/redo)
- Allow to copy and paste
- Provide zooming functionality
- Provide simple saving
- Export or share tactile image
- Export and share visual image

4. Presentation and Output

- Multimedia access to draw visual, tactile, textual verbal and auditory
- Provide tactile feedback
- Provide graphical user interface
- Support of awareness of object state
- Filtering of presentation information (to avoid complexity)
- No perceivable rasterisation of the drawing
- Indication of tactons (buttons) functions
- Grouping

Out of all the above-mentioned models, frameworks, and standards, this list is most closely and appropriately based on CADB design and development in the domain of AT. Created by Bornschein Weber and following ISO standards, it categorises the sets of features into logical units of general, interaction, functional, and presentation areas specifically for BVI users' art creation. All of the models, frameworks and standards have repeated patterns of guidelines and new focus areas that lead to the formation of usability guidelines in this thesis as listed below.

3.3.5 Usability Language Considerations

Figure 3.19 shows the classification diagram of usability language. It is derived from the above-discussed standards, frameworks, theories, and taxonomies. These models commonly describe the environment, person, activity, and support, which can be identified by performance in a certain environment, technology, or environmental features, senses, and processing skills. The recommendations from those areas leads to further pointers towards the ideal features of AT technologies, i.e. they should be independent, personalisable, easily recoverable and intuitive products, which in turn lead to the definition of a usability language introduced in this thesis.

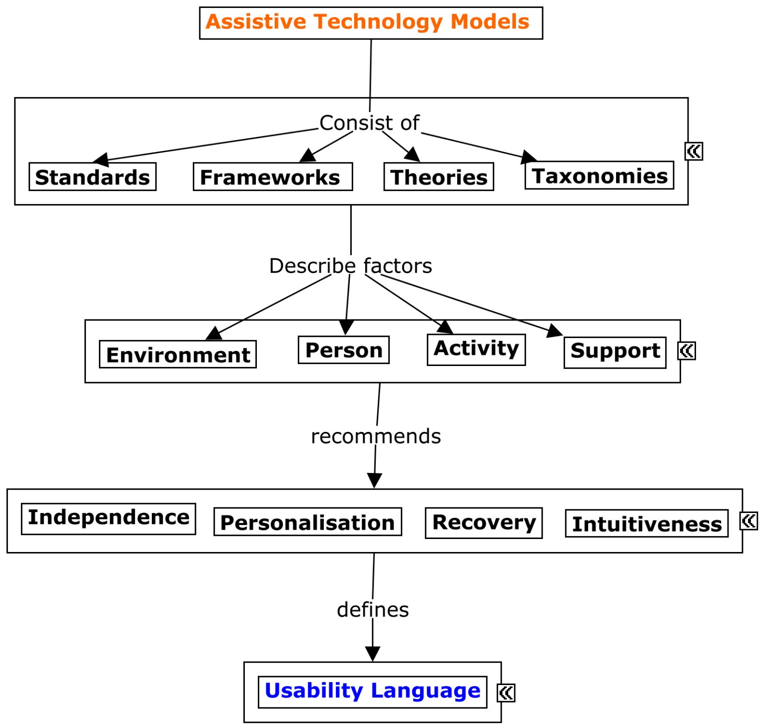


Figure 3.19: Classification Diagram of Usability Grammar

The set of usability language considerations laid out below for a drawing and navigation model is drawn from the literature presented above, such as the standard presented in ISO 9241-110, ISO 9241 210 and other standards, frameworks, taxonomies, perspectives and dimensions.

1. **Ability to personalise -** System features to pick different levels of interaction pace and interaction method. For example little or more voice feedback, change of system input and output modes from keyboard to braille to voice, change of interface items and sizes.

2. **Ability to work without relying on help -** Self initialisation and exiting the application. Inbuilt shortcuts and quick access to open, exit, help and screen reading features. Ability to put work in progress on hold and return to it.

3. **Ability to learn easily -** Access to help and different modes of help (voice, braille) and different levels of "how to use" tutorials.

4. **Support for intended activity -** Availability of primitive shapes, ability to draw non-primitive shapes, ability to use previously designed art, ability to group and define art.

5. **Ability to recover -** Features such as save, delete, erase, and ability to retrieve previous images.

6. **Ease of use -** Complete an intended task without much delay and effort. Ability to interact with the system without much effort, and some more features such as redo, undo, copy and paste.

7. **Ability to manage cognitive load -** Simple, short direct linguistics interaction to complement the working memory of the intended user.

The above sections have explored three areas that drive this thesis: shape considerations, spatial considerations and usability considerations from existing literature. The next section explores the broad field study conducted in understanding the blind user.

3.4 Understanding Blind Users

Despite the developments in technology, there are still aspect of work and life for blind people where a support worker can be helpful according to RNIB [RNIB, 2020] that blind learners often seek the help of a support worker to draw pictures or diagrams, or they avoid drawing because they find it difficult to believe that they would be able to create pictures and diagrams without guidance from a sighted person and therefore would not even make an attempt. Hence, expressing imaginative thinking through computers is limited for blind people. Therefore, the need for self-reliant blind drawing techniques and technology is highly valued among blind communities. The question is not whether it is useful for a blind person to draw at all because they cannot see. Instead we focused on finding the answers to the following research questions:

- What are the main concerns of computer usage among BVI users?
- What screen layout arrangement do BVI users prefer?
- What sort of a navigation system would easily build a sense of object layout?
- What sort of a space arrangement could allow easy navigation?
- What feedback mechanisms are effective and efficient?

3.4.1 Initial Data Gathering

The first survey was conducted with people from institutes, such as RNIB[1] and MACS[2] charity members during their family weekend. A presentation, followed by a system demonstration were conducted at an annual family meeting and the survey answers were collected.

[1]https://www.rnib.org.uk/
[2]https://macs.org.uk/

A survey was also sent out to MACS members which was published in Focus Magazine Issue 1 in December 2013 in the appendix. We asked the following questions:

- Do you or does your child use technology and if so, what do you use and for what purpose? For example, what games do you play? Do you use software like JAWS or other applications?
- Do you or does your child like to create pictures, images or diagrams and how do you / they do this? Would you or your child like to be able to use technology to create images and designs?
- What particular problems or barriers do you face in using technology generally?
- If you could imagine a tool which would allow you to express yourself by getting the image in your mind onto a computer screen, what would it be like? What would work best? Would you prefer to use speech, touch, a keyboard or other methods to direct your ideas?

We met up with several MACS members during their annual family weekends on many occasions for data gathering and later for evaluation of the prototype.

We found out that many blind people use software in their day-to-day lives which may not be specially designed for blind people. Even though screen readers are still in the early stages of reading diagrammatic drawings, it is about knowing the inbuilt accessibility features of software. We received the following answers for technology use and purpose:

- *"I use technology - but it's nothing specially designed for VI people. I use an Android phone with Android software and apps (no special apps for VI). I use an Apple Mac (and a PC) with the inbuilt accessibility software. I do use a video magnifier from Freedom Scientific sometimes"*
- *"My son uses JAWS based on a Windows application and also voice over when using an Apple Mac both give access to many areas of software but both have limitations in software access"*

Another finding was that some members didn't think drawing was necessary for blind people as they would not pursue a career in graphics designing, and they would rather be writers stating answers such as *I'm a writer, not a drawer.* Some blind and partially sighted people did not think that there was a need for a diagrammatic drawing tool and we received the following answers:

- *"I see a chicken and egg problem. How does a blind person know what a triangle is if they've never seen one? They can feel it, but how can you be sure that the image they're drawing in their head is a triangle? I suppose the ultimate answer would be an iPad where you can draw onto it, with some kind of tactile aid. However, I don't see this as being a massive problem for VI people. They're never going to get jobs as graphic designers, so why the need? Braille is still the most adaptive form of writing for completely blind people - learning by audio leads to mistakes, being "hands on" is the only way to learn how to write properly. Therefore, being 'hands on' - with tactile materials - is the only way to learn to draw properly!"*
- *"This is not an option in relation to seeing or reading images in any format via technology."*

- *"I would like to have access to a screen reader that could read images and have software that would help me to draw images using speech recognition. You should be able to speak to create an image or object; so if I say circle, or top left circle, this should appear in the top left of the screen. It should also explain images with the size and colour."*

One of the findings repeatedly made was that the barrier to the use of technology dependent on the instructor's or teacher's knowledge of the technology. The following answers were received by participants when asking about particular problems or barriers in using technology:

- *"I believe the biggest barrier to a blind person using normal technology is the knowledge of the teacher or instructor. Most 'normal' technology is accessible to blind people, it's just perceived as inaccessible. I don't believe there's a great need for over-complication of technology for the VI - there's just a need for greater awareness amongst trainers."*
- *"Access to various forms of software that do not provide the relevant access controls."*

To conclude, half of the blind participants overlooked the idea of a drawing technology, whereas other participants showed excitement at the idea of self-reliant, digital drawing technology and shared the possible interface scenarios. Some participants were less hopeful of drawing technologies as they find limitations in existing screen readers that do not explore images, while others liked the idea of diagrammatic drawing, especially in the STEM related subjects, in a classroom setting. A higher number of participants relied on screen readers, therefore they were keen on screen reading technology for any possible CADB systems. Even though some people believed that drawing is not their field in which to excel, no one denied the need for a drawing tool in education but said they would be happy to seek the help of a support worker. Very few participants managed to come up with blue sky thinking for a drawing tool with computer support facilities for quick and easy drawing that could have multiple modalities.

The first set of questions were more generic in finding BVI people's perceptions and current state of technology use, their general views on AT use and thinking about barriers and ideas for a new drawing tool. Another survey was designed in Table 3.3 that had more questions focusing on participants' previous drawing experience. The second set of survey questions were designed to obtain answers about equipment used for drawing, the type of art that was produced, and the difficulty of production. The survey was extended to, other charities and different subsets of blind people (early blind, late blind and VI people).

Section 3.4 gives details on some previous seminal studies carried out with BVI participants. The above studies pre-date the emergence of apps like Seeing AI, Envision AI, and the inclusion of image descriptions as part of the Voiceover screen reader. These new technologies are not dismissed nor forgotten. A clear description of these technologies are presented in section 2.2

The questions below were asked and recorded in one-to-one interview sessions with five different participants from [RNIB, 2018] and charities such as [Beyond Sightloss, 2018] and [1]. The results are

[1]https://macs.org.uk/

shown in table 3.20

Question	Yes	Partially	No
Do you like to be able to create images or diagrams?	5	-	0
Have you ever produced art?	4	-	1
Was it easy?	3	1	1
Did you complete the drawing task?	3	1	1
Did you enjoy creating art?	3	-	2
Was it a digital/computer tool?	1	-	4

Table 3.3: Second Survey

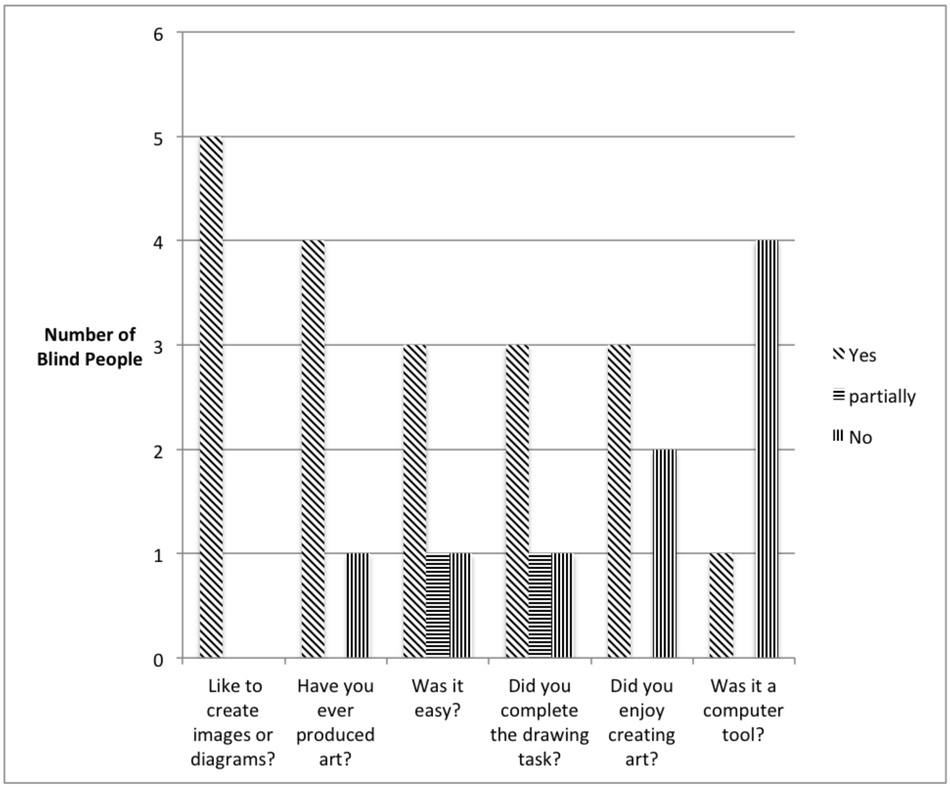


Figure 3.20: Resultant Bar Chart of the Second Survey

Figure 3.20 is a bar chart that shows demonstrates that all five participants were excited to be able to create images or diagrams and most of them attempted it with the intention to succeed. Some people found it easy to create simple images using manual paper kits, however some of them didn't enjoy it due to the mental effort, output inaccuracy and difficulty of the process. The survey data informs us that 100% of the participants wanted to be able to create images if a mechanism could be provided that supported the whole process.

There were some open-ended questions asked during the interviews to understand participants' drawing experience:

- What was the image produced? How did you create it? Did it take time?

  - *An outline of a 2D house, 10 minutes, Swell paper*
  - *An elephant, A squirrel, Mat and pens*
  - *shapes, Bonham Geometry devices, diagram boards, rollers made it emboss*
  - *2D shapes, triangles, German film, TactileView*

- How did you do the correction and validation?
  - *This is how I saw it, touch the paper for feel*
  - *Touch paper, perkins braille*
  - *Using new papers*
  - *By feel*

- If you imagine a digital tool that could help you to draw, how would you like it to be? And why?
  - *Touchpad, Tablets such as Kanut*
  - *Clear labels, visible lines*
  - *Verbal explanation is enough for me, I can't understand /grasp things by touch, e.g I can imagine an elephant* **This is mentioned by a late blind individual who lost sight in their late teens.**
  - *Touch by reading models, toys, I can understand the concept.*
  - *Get correct shapes with the help of a computer, feel for validation, Speech synthesiser as standard screen reader do not read maths symbols or art*

From the second survey, we found out that all of the participants are in favour of creating images and diagrams for education and other purposes. The majority of them have attempted some sort of drawing and enjoyed the process, while few didn't like or enjoy it. Four out of five of the participants used tactile drawing tools as they were the tools most available from education providers, hence an efficient digital tool would be in high demand. Participants had produced outline drawings of simple images and used a tactile mechanism to verify their drawings using swell papers. A late blind individual preferred a speech explanation method of image exploration, while early blind participants preferred the tactile method. Some of them suggested digital drawing features such as shape correction with the help of a computer and inbuilt speech synthesisers.

3.4.2 Conclusion

The initial studies were carried out to understand blind users' technology use, views, and experience of drawing. These studies provide useful information including the equipment and AT software used by BVI people in general and during drawing activities, barriers to AT use, symbolic representations of art, methods of verification, methods of understanding drawing concepts, subtle variations between different subsets of blind users, and suggestions for an ideal drawing tool as imagined by BVI people. The intention of the study was not to build the hypothesis for screen design based on the views, but merely to understand the user in order to steer the technical specification in the right direction for the research in this thesis.

3.5 Classification of a CADB model

I have created the classification diagram presented in the form of a concept map in Figure 3.21 which is based on previous studies in literature discussed in this chapter. The map shows how the classifications split into different branches that have been discussed in this chapter. This map is used in the final design of the specification in this research. The research direction started from a position of exploring cognitive models, perceptual models, and assistive technology models in order to formulate a formal specification.

The first branch on cognitive models can be identified in early blind, late blind and sighted individuals. People who acquired their blindness at an early age have a certain perception of spatial mapping that builds on the conceptual and perceptual level. The conceptual level builds on spatial models, and the perceptual level is built with a sensory substitution that can be made possible by haptic, audio and olfactory means. The spatial model needs some exploration strategies which could be perimeter, grid and object-to-object that were investigated by [Ungar, 2000]. Grid strategy can be easily conceived by a navigation technique moving back and forth in a screen grid system. Such a navigation technique is possible with screen locations, screen points and cardinal directions. The above investigation into empirical information is used to define spatial grammar in this thesis.

This research is looking into outline drawing that can be created with tactile graphics and computer-aided graphics. The second branching of the classification diagram on perceptual model leads to blind drawing, which is based on outline drawing or movement drawing. Such computer-aided graphics are best implemented with primitive shapes. A transformation of these primitive shapes (such as the creation, labelling, grouping, and defining of shapes), is used to the defining the shape grammar of the research in this thesis.

The third branch of the classification diagram is on assistive technology models consisting of standards, frameworks, theories, and taxonomies. The classification diagram serves to recommend standards for products and services needed in the daily life of a blind person needing environmental support, personal and general activity support and other needs. The usability grammar proposed in this thesis is based on such standards and recommendations The formal specification in this thesis describes the usability, shape and space considerations mentioned above and stands as the basis for SETUP09 grammar rules.

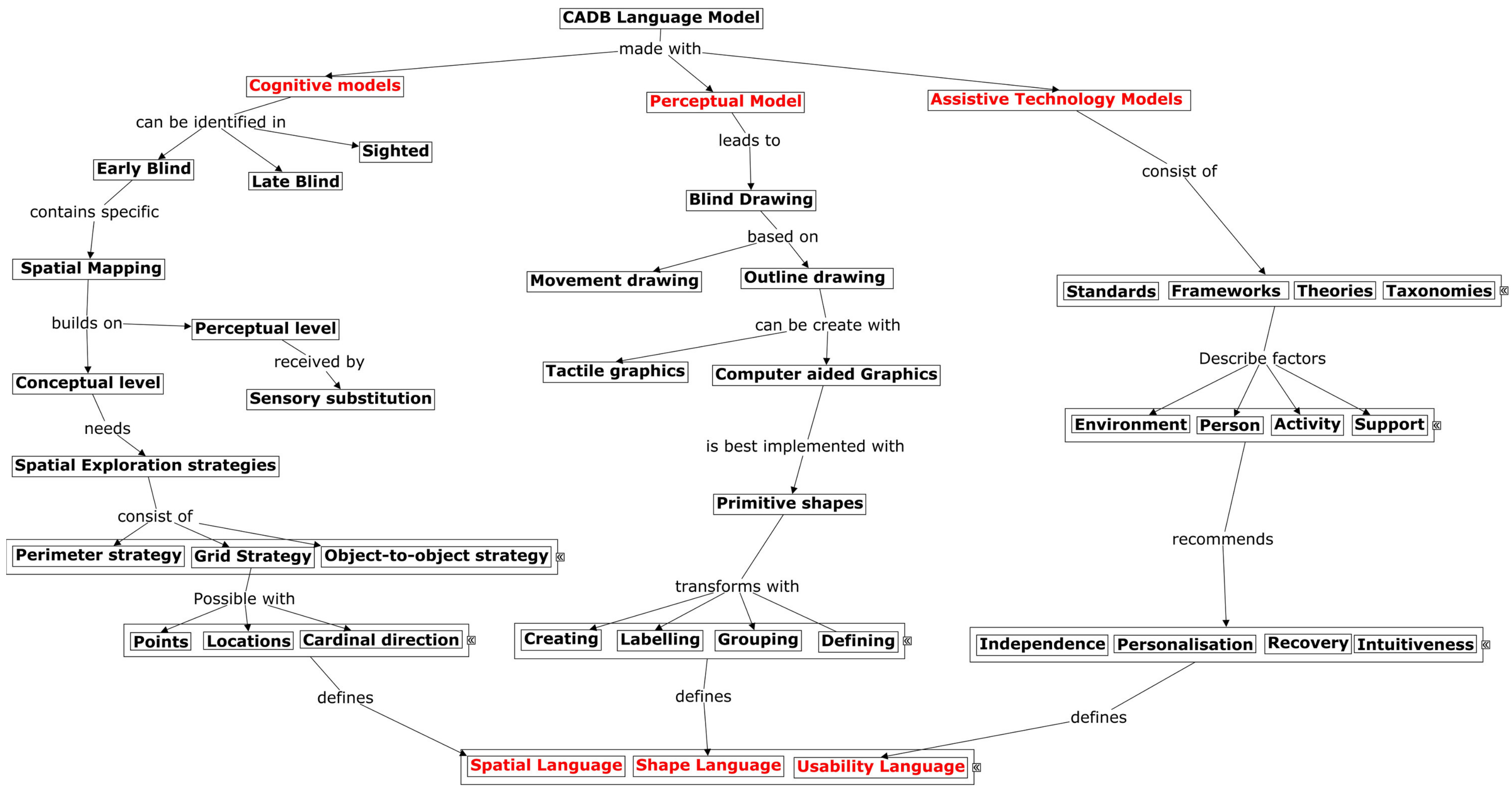


Figure 3.21: Derivation of Formal Specification : Concept Classification Diagram

3.6 Conclusion

This chapter has explored the modelling issues of CADB, discussed issues of drawing techniques, navigation techniques, and the usability of assistive technologies. Then the chapter answered questions such as: What constitutes a formal drawing language for the blind user?, What sort of object layout would a blind person easily conceive and convey?, What sort of navigation system would easily create a sense of object layout?, What sort of space arrangement could allow easy navigation? and What usability features are most suitable for designing CADB? The answers to these questions then inform guidelines to form a CADB language in the areas of space, shape, and usability domains. This model then leads to the question of "How is such a CADB model to be formally designed?" which is discussed in Chapter 4.

Chapter 4

A Formal CADB Model Design

This chapter presents the next logical stage in the development of the formal approach that employs a formal modelling technique. This approach is a mathematically proven basis for language creation through guidelines (introduced in chapter 3) with a set of production rules. The work in this chapter forms the major contribution of the research carried out in this thesis. The previous chapter introduced modelling guidelines for a shape language, a spatial language, and a usability language whereas this chapter presents the guidelines in a formal modelling form. In order to bring the totality of the model, the reader is first presented with language design decisions that reason out certain concepts (locations, points) based on literature findings of cognitive models covered in Chapter 3. The chapter then presents the component diagram and grammar formation of each component of the proposed SETUP09 model. Space Grammar, Shape Grammar, Usability Grammar are presented with Backus-Naur form and explanation based on the guidelines introduced in the previous chapter, which were derived from literature findings. Finally, the chapter presents the first version of the Context-Free Grammar model for the CAD system for blind people use, which is the main contribution of this thesis. The section ends with opportunities of such a formal model for system designers and users. As the specified model is for a language implementation, it is an implementation of a compiler/interpreter, that will need to be validated. This leads to the next question, which is, How is such a specification be implemented and validated? The answer to the question leads to compiler/interpreter implementation in chapter 5.

4.1 Design Decisions

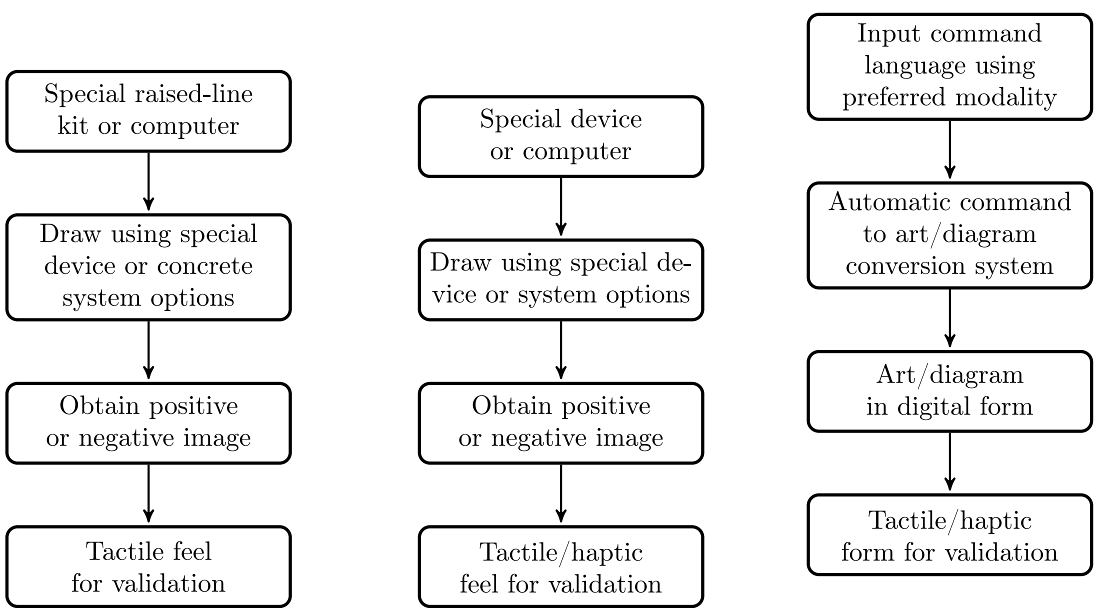
Blind Drawing System Types

Analogue drawing systems [1] established in the past are still commonly used but do not provide the required accessibility features recommended by Ergonomics of human-system interaction, Part 110:

[1] Manual Drawing Paper Toolkits such as raised line papers, plastic film papers, Dot Inverter, Quick draw paper explained in chapter 02

Dialogue principles, [ISO, 2006]. This specifies the usability of special features, including functionalities such as redo, undo, edit, delete and timely feedback, when designing a computer software product for BVI users.

Current digital drawing systems have overcome previously mentioned impediments to some extent by including usability features. Some systems come with special external kits that pay specific attention to usability (e.g. AHEAD, [Rassmus-Grohn et al., 2007]) and other systems do not pay attention to layout information but convey message and meaning about the intended diagram (e.g. System KEVIN, [Blenkhorn and Evans, 1998] PLUMB, [Calder et al., 2007]). Recently introduced systems BPLOT, [Fujiyoshi et al., 2014] and BPLOT2, [Fujiyoshi et al., 2008] employ system dialogue to create drawings using a command language but only for sighted users. The plotter language of BPLOT is created mainly for sighted individuals to produce accurate graphs, whereas blind individuals have a tablet interface. The plotter control language operates mainly with coordinate systems and this can require a high level of mental acuity to complete a simple task, hence it is more appropriate for sighted users to work with it. Even though some systems pay attention to layout information and screen navigation, such as IC2D, these systems are operated by a defined set of system shapes and the drawing functionalities are not necessarily easy to use in practical situations due to the limited screen referencing functionalities. The SETUP09 model is introduced to overcome the above limitations by building a technical specification that enables user-initiated drawing and support features by employing a drawing language.



Analogue Drawing Systems | Digital Drawing Systems | Language Driven Drawing Systems

Figure 4.1: Analogue versus Digital versus Language Driven Blind Drawing Systems

A comparison of digital and analogue systems was presented in Chapter 1. This chapter furthers the comparison of analogue versus digital versus command-driven systems by highlighting the technical differences in concrete CADB systems and the SETUP09 CADB system. Figure 4.1 shows the main differences between analogue, digital and language-driven drawing systems. In the figure there are three flow charts. The first chart shows the general flow of current analogue systems that are in use. The second chart shows the flow of the digital systems that are in use. The third chart shows the flow of the command-driven systems that are in use.

Figure 4.1 shows three flow charts. The first flow chart shows a user using a current analogue system to draw an image using a manual drawing kit tools to produce a positive or negative image and then checking the image for accuracy by touch alone. The second flow chart shows a user using a concrete system to draw an image. The user is able to manipulate the image using a computer and generate a positive or negative of the image. Once again, the user can check the accuracy of the image generated by touch. The third flow chart shows another modality based on inputting a specified command language to draw the image using a computer. The user?s voice commands are converted by the computer to draw an image and display it on the computer screen in tactile format.

In most of the cases the existing analogue systems do not allow for user correction, expansion of space and addition of useful tools such as rulers or stencils. However the drawing can be faster as there is no system learning or operation time. Digital drawing system activities can take longer as operational knowledge of the system and learning time needs to be factored in. Language-driven systems are not common but promising. They take away the need for additional hardware devices to perform the drawing. The computer takes care of command-to-diagram conversion once the LS (language systems) commands are correctly learnt.

4.1.1 Drawing with Command Languages

This section explains the existing command-driven languages for screen navigation and art production, the use of those commands with examples, and then presents the example of SETUP09 as a command-driven language.

BPLOT2 introduced by [Fujiyoshi et al., 2008] is a universal design tactile graphics production system using a plotter control language. For example, the commands to draw a box in plotter control language are fed in to the system as: *box 0,0,2,8,3.* "box" is to draw a box where the first two parameters "0,0" are for the coordinates of the centre point centre, "2" is the width, "8" is the height and "3" is the length of interval between dots to create a box in BPLOT2 language.

Automatic text-to-diagram conversion introduced by [Mukherjee et al., 2014] is referenced by coordinates with the numerical values of the endpoint of the four component lines. A parallelogram ABCD is created with four coordinates:*AB: line (100,200),(50,100), BC: line (50,100), (250,100), CD: line (250,100), (300,200), DA: line (300,200), (100,200)* four co-ordinates.

Document production system such as Latex support packages such as like TikZ to produce command line drawing. A simple rectangle in Latex can be produced by *draw (0,0) rectangle (4,4);* where the first two parameters "0,0" means the coordinates of the the starting point, then followed by the co-ordinate of the corner diagonally opposite to draw a rectangle in Latex, [Overleaf, 2019].

Turtle Graphics introduced Wally Feurzig and Seymour Papert in 1966, and use in the LOGO programming language, later owned by [Python, 2019], is a computer graphics system that uses a relative cursor the *turtle* upon a cartesian plane. The movement of the turtle happens with commands relative to its own position. An example command to draw a square using turtle is *turtle.forward(50), turtle.left(90), turtle.forward(50), turtle.left(90), turtle.forward(50), turtle.left(90), turtle.forward(50), turtle.left(90)* such as "meaning move forward 50 spaces" and "turn left 90 degrees" to draw a square.

Most of the aforementioned systems work by calling shapes and screen locations using screen coordinates that employ allocentric, object-to-object screen navigation, whereas Turtle Graphics uses egocentric, self-to-object screen navigation. Some languages also feed in the size of the shape, whereas other languages have vocabulary for repetitive drawing. Similarly, the SETUP09 system introduced in this thesis uses a command line language calling compass-based points to draw shapes and art. For example the command to draw a square in SETUP09 language explained in this chapter is fed in to the system in multiple ways. e.g. *lines NW,NE,SE,SW,NW* or *square NW* or *line NW NE, line NE SE, lines SE SW, lines SW NW* and *call myline* by calling points or previously created lines. First the user has to navigate to the intended screen location as explained below in section 4.1.2, Location Referencing. The experiment findings in the results from study 2, which is presented in Chapter 6, suggest that the SETUP09 method is easy to use. The next section explains the referencing technique used in SETUP09.

4.1.2 Location Referencing and Cell Referencing

Screen locations can be identified in many ways. Identifying screen location by coordination does not lead to accurate results as the x, y coordinates refer to a point not to an area. Screen points are beneficial for point referencing however they are difficult to remember when many coordinates are presented to create art. For example, the value of an x coordinate on the screen of an Apple MacBook Pro 15-inch can go up to 2880 any Y can go up to 1800 pixels [1], , which is much more difficult to remember and refer to as opposed to a screen with a grid of 9 squares with names. A blind researcher [Kamel et al., 2001] introduced a telephone keypad style screen division, with a numbering system and centre only point cell referencing. His findings suggest a 3x3 style layout is the most efficient for screen navigation.Table 4.1 illustrates numeric style screen location recognition versus compass-based naming style screen location recognition that is introduced in this thesis.

This paragraph presents the reasons for the choice of compass naming for locations and points. BVI users mentioned that screen area recognition with compass direction naming is easy to remember for screen navigation than a numbering system during our interactions with them. Furthermore, BVI users are familiar with screen navigation as directional compass names are very well integrated into mobility training applications on cell phones (eg: BlindSquare, talking compasses, braille compasses [2]) as well as on accessible GPS devices which are used on a day-to-day basis by blind people. Compass directional navigation is also evidenced in game technologies such as command-line interac-

[1] http://screensiz.es/

[2] https://familyconnect.org/browse-by-age/grade-schoolers/transition-to-independence-grade-schoolers/how-to-master-cardinal-directions/1235/

tive fiction [3] games that are played by blind gamers', and navigation research conducted in the past, [Meliones and Sampson, 2018], [Silva and Wimalaratne, 2017] and [Kassim et al., 2016] also used cardinal names in its applications. In table 4.1 screen locations are represented by nine grid cells. The left screen has numbered cells (1 to 9), and the right image has named cells. The names are compass directions written on grid cells.

1	2	3
4	5	6
7	8	9

North-West	North	North-East
West	Centre	East
South-West	South	South-East

Table 4.1: Cell Names with Numerical Keys and Compass Names

Most screen location tracking systems use coordinates, whereas few systems are grid-based. A single cell reference system can be time consuming and memory intensive for a blind or sighted person compared to a solution such as the multiple point referencing system that is introduced in this thesis. Multi-cell reference system uses a compass-based naming convention because the points represent north, south, east, west, north-west, north-east, south-west, south-east direction points in space, hence the perfect name to call those points were the same as location names. There was no confusion of the naming convention recorded during the system experiments presented in Chapter 6 even though compass naming convention was used for location referencing and also for point referencing, as the concept was clear and logical.

Table 4.2 shows the distinction between single point cell referencing and multi-points cell referencing to draw shapes referenced by screen points. Table 4.2 shows two interfaces, the left interface is marked with the centre point of each cell and the right interface is marked with 9 different locations on each cell to make drawing easier by referencing the screen locations. A multi-point cell referencing system has 19 different points as shown in Table 4.2, as opposed to two points in a single point cell referencing system. Some points overlap others when changing from one cell to the next and the relative positioning of campus direction changes. For example, the East point of the West cell becomes the West point of the centre cell.

[3] http://if-games.com/

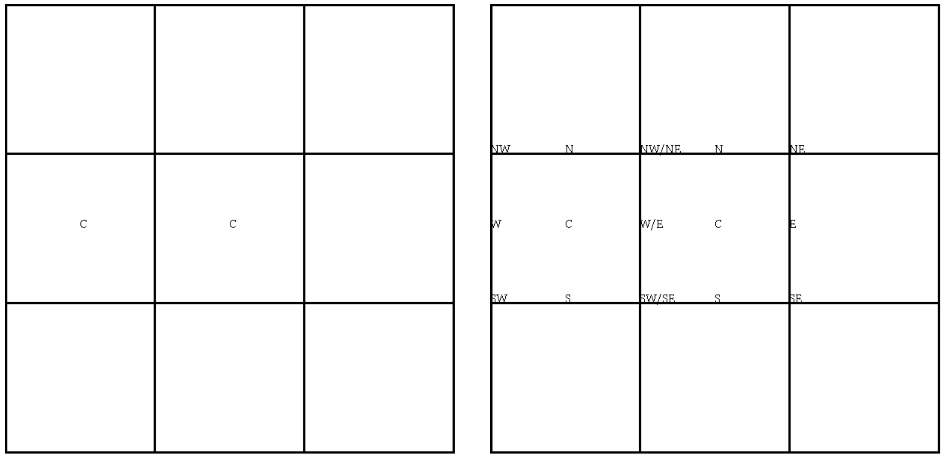


Table 4.2: Centre Point Cell Referencing and Multiple Points of Cell Referencing

4.1.3 Cell Recursion and Points

Table 4.3 3 illustrates the recursiveness of the grid system to enable cursor movement to any desired location on the screen. The cell recursion also allows a user to refer a point in one location to points in other locations. Table 4.3 shows a triangular shape referring three different points in three different screen locations and zoom levels. The system keeps track of zoom levels (the size of the current location) as the cursor moves to a smaller location. The system is on zoom level 2 in Table 4.3 whereas level 0 refers to the startup screen/whole area of the screen without any cell recursion. By default the cursor is set at the NW point on a given location. The size and the start point of the shapes can also be based on a default zoom level and a default cursor point.

Screen Navigation

The navigation technique of SETUP09 is based on a grid system that contains nine cells. The grid has three columns and three rows (3x3). The cells are referred to with names (rather than numbers): North, South, East, West, North East, North West, South East, South West and Centre. Arrow keys and the space bar invoke SETUP09 navigation using the keyboard input mode. The size of the user screen determines the size of these cell areas. On any given screen a cell has 1/9 of the total screen area allocated to it. These screen sizes are provided only as guidance and are not fixed. The user has to navigate out of a screen cell before they navigate to a new screen area, once a cursor is in a particular cell/location. However, the user can decide to navigate to smaller screen areas within an area without navigating out of that area to find a specific screen location. A given area also has multiple points to which users can refer when starting a shape in the same way that a cell is referred to by names rather than numbers; the points in a cell can also be referred to by names: North, South, East, West, North East, North West, South East, South West and Centre. For example a line drawing has a start point (North) and end point(South); so a user can easily refer to two different points on the screen to draw a line.

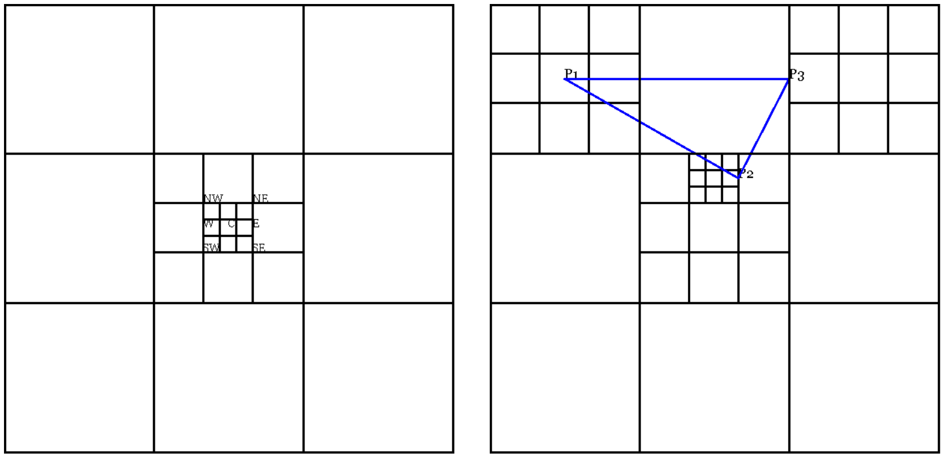


Table 4.3: Cell Recursion and Referencing

4.1.4 Zoom Levels

Table 4.4 illustrates some circles with different zoom level numbers: 0, 1 and 2. Zoom level is used as a technique to navigate to a smaller granularity on the screen. Zoom size can also be set as the default size of a shape if a shape is called on to a location, rather than creating a shape with lines or curves. In Table 4.4, the first circle is created on the screen at zoom level *0* and with no reference to a location, the second circle shape is created on the screen in zoom level *1* and location *center* , and the third circle is created on the screen at zoom level *2* and location *center, center*.

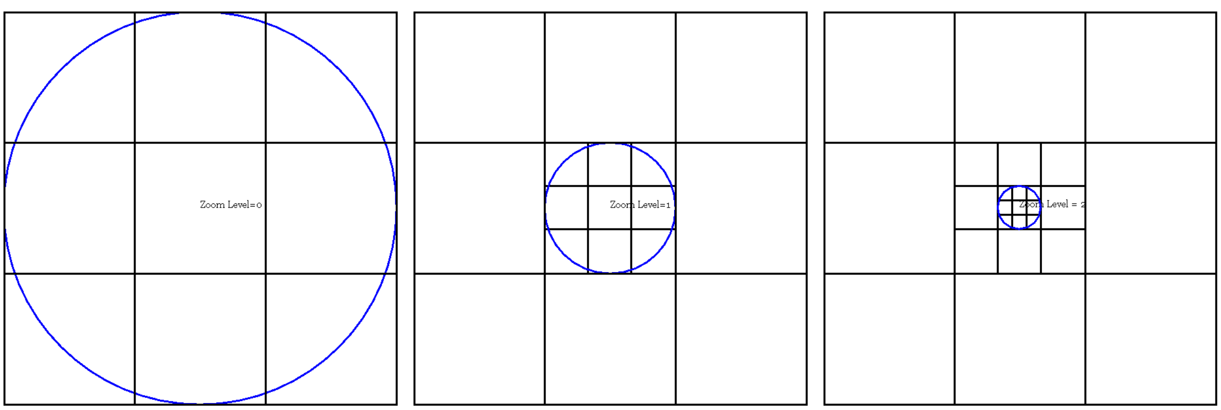


Table 4.4: Zoom Levels

This section gave an explanation of the design decisions around important language constructs, the rationale of the proposed command language in comparison with existing systems and cell referencing, cell recursion, and zoom levels of spatial language. Next, the architectural components for the SETUP09 model and its assumptions will be addressed in the following section.

4.2 Architecture for SETUP09 Model

As depicted in Figure 4.2 it is assumed that the context-free grammar model is coupled with input interfaces such as text, speech or Braille and output with art/diagram in tactile form. It is established in the literature review that context-free grammar consists of shape language, space language and usability language. The below language components are designed based on requirements found and discussed in Chapter 3.

Formal Language→ Space Language+Shape Language+Usability Language.

In short, shape language consists of requirements to identify points, draw lines or curves between points, produce primitive or arbitrary shapes, group and reuse them. Space language consists of requirements to navigate to and from a screen area, reference a screen area, a point or a route, and describe art. Usability language consists of requirements to allow the user to work without help, learn quickly, get support during drawing, recover work, to work with ease, and have a minimal cognitive load. Linguistics are then passed through a lexical analyser to retrieve tokens and semantic analysis to identify the type and meaning of tokens at the input recognition stage. Parser functions perform semantic matching, including error tackling, and art generation is the last order of component links to get the desired representation of diagrammatic images.

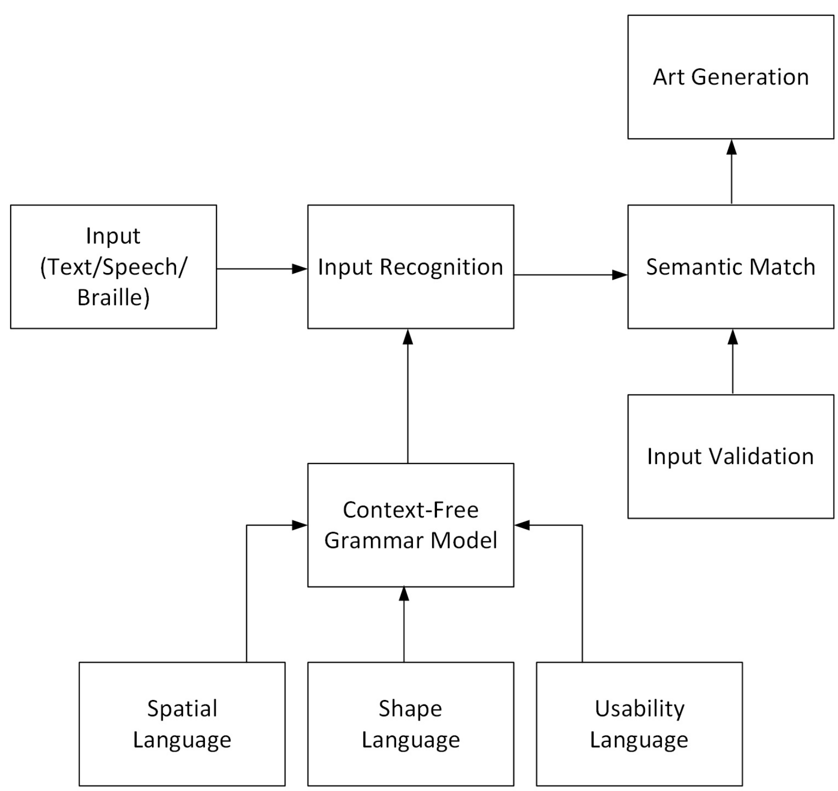


Figure 4.2: Components of SETUP09 Model

The next section explains the architectural assumptions of SETUP09 Model.

4.2.1 Architectural Assumptions

A general, open architecture for CADB is depicted in Figure 4.2 as a simplified data flow diagram. The model contributes in this thesis propose formal specification view of the SETUP09 architecture while there could be other engineering and ergonomic views.

As shown in Figure 4.2, it is assumed that CADB functionality is loosely coupled with a user interface and command interpreter technology. An example of a user modality is the keyboard and a Java canvas interface for screen art. The interpreter can render formal language.

The steps associated with the model architectural are as follows. The screen interface captures user commands for the desired drawing. If the request is for drawing a 2D shape, then the interpreter first checks for the screen location, starting position and zoom size. Then the interpreter checks for the semantic accuracy of the command and produces the requested art.

It is assumed that CADB may be added with other APIs (Application Programme Interfaces) input commands. In that case, the client input is received by speech or Braille and input validation has to take place for input accuracy.

Note that the main contribution of this model is not invalidated if the assumption is falsified. However, the assumptions make it easier to value the new CADB properties and to argue that the contribution does not clash with tightly-coupled CADB systems.

The architecture is notwithstanding the fact that the user may want to personalise the user-interface; it can be strongly argued that a broad spectrum of user action in CADB should be accommodated by the architecture.

To model a personalisable CADB, it is proposed that user interaction is taken from any modalities and transformed through an input recognition process, "an interpreter". The SETUP09 architecture is therefore modelled with additional semantic matching, parsing and input validation components.

4.2.2 Technical Approach

The technical approach taken in this thesis is to formally model core CADB functionality from the specification. The user does not place actual art on the canvas, but rather a specification of how to build the art upon input.

The semantic specification is given with reference to the AST (abstract syntax tree) by specifying user commands. An AST is used to illustrate the execution of the input command with production of the associated artwork.

SETUP09 is modelled as a grammar-controlled art production process. Such a method can store the command history which can resurrected many times for reproduction in different sizes onto different canvasses. The SETUP09 method can easily be coupled with various modalities that allow customisation. The method is free from actual drawing on the canvas, hence it is fast and less error-prone.

In summary, the model is an abstract one, built on context-free grammar, used to represent the semantics of command language. So far, CADB has been viewed as a concrete system-driven implementation, in particular it is modality coupled with systems. It allows personalisation by giving ownership of the process of building shapes from designers to users. To ensure the user language is consistent, its elements are introduced from the formal definition. Therefore all design decisions are within the scope of user-driven commands. The next section explains these user-driven commands with BNF notation, expounding a proven mathematical basis for language creation.

4.3 Introduction to Grammar formation

Table 4.5 illustrates the symbols used to write the formal specification in BNF environment.

Operation	Outcome
Production Operation	$\models$
Production Name/Non-terminal	$\langle$name$\rangle$
OR operator	$\mid$
AND operator	,
Empty String	λ
Terminal description	*description*
Terminal String	'string'

Table 4.5: BNF Production Symbols Modified Using [Robson, 2019]

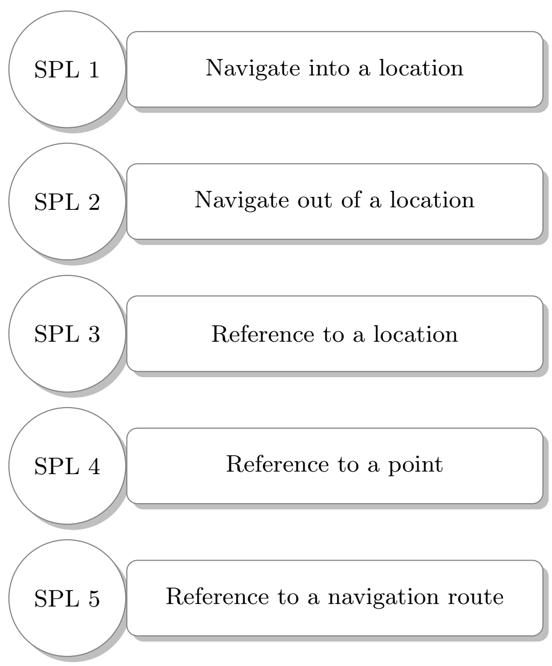
The space, shape and usability guidelines presented next are written in the BNF notation presented in Table 4.5, which is formed based on spatial awareness of blind people, blind imagery and usability standards that was explained and mapped in the classification diagram in Chapter 3.

4.3.1 SPL (Space Language) Grammar Formation

The first subcomponent of the context-free grammar (CFG) model is space language. Spatial orientation and recognition are important to blind users when navigating on the screen, to know screen locations in relation to the current screen area and current cursor point.

Space language consideration needs a technique to identify a point in space, the current point, the change of a point, the area in which the point exists, a technique to change from one point to the next, and forward and backward movement from point to point. The space design guidelines are formed as a result of work introduced by Hill [Hill et al., 1993] on spatial navigation techniques, Lahav's [Lahav and Mioduser, 2008] introduction to grid-based technique, Kamal's [Kamel and Landay, 2002] telephone keypad communication style, Meliones [Silva and Wimalaratne, 2017] floor grid indoor navigation and cardinal direction based turning, and Kassim's [Kassim et al., 2016] digital compass and direction guiding system as discussed in Chapter 3.

SPL(Space Language) Considerations



- SPL 1 - Technique to navigate into a location: Effective linguistic to convey the action of transition

$$\langle\text{zoomin}\rangle \models \text{'zoomin'}, \langle\text{location}\rangle$$

The non-terminal grammar *zoomin* is the way finder on the screen. The grammar is called with the intended location name, for example *zoomin centre* where *centre* is the location name. The user can call any one of nine terminals explained in the *points* rule. It takes the focus from a bigger screen area to a smaller screen area (e.g. 1/9 of the current focus area in SETUP09 system). The grammar *zoomin* in SETUP09 is used as a navigation command rather than a magnification. The system also keeps track of the zoom level based on *zoomin*. The zoom level determines the size of shapes. At the start the zoom level is 0 and the shape covers the whole screen area. For example, if command *zoomin center* is called, the cursor moves to the centre area of the screen and the zoom size increases to 1. *Zoomin* user grammar is the most important grammar that moves the cursor in to screen compass grid blocks.

- SPL 2 - Technique to navigate out of a location: Effective linguistic to convey the action of relocation

$$\langle \text{zoomout} \rangle \models \text{'zoomout'}$$

The non-terminal *zoomout* is the opposite of *zoomin*. It takes the focus from a smaller area to a bigger screen area. (e.g. nine times bigger than the area in focus). The command *zoomin* has no other terminals attached to it. To navigate to a new screen location, first the user has to exit the current location. The grammar *zoomout* automatically reduces the zoom size when it is invoked; images get larger.

- SPL 3 - Reference to a location: A location on the screen with an effective referencing system to identify the location

$$\langle \text{location} \rangle \models \text{N} \mid \text{S} \mid \text{E} \mid \text{W} \mid \text{NE} \mid \text{NW} \mid \text{SE} \mid \text{SW} \mid \text{C}$$

The terminal location has nine non-terminals represented by compass names: North, South, East, West, North West, North East, South East, South West and Centre. The screen grids 3x3 are named with screen locations. This has proved to be memorable and sensible to blind people in way-finding activities, as discussed in the literature review. Locations are used with a *zoomin* terminal for example *zoomin centre* to navigate to the centre area of the screen.

- SPL 4 - Reference to a point: A point on the screen with an effective referencing system to identify different points in a cell

$$\begin{aligned} \langle \text{points} \rangle &\models \langle \text{point} \rangle \mid \langle \text{point} \rangle \langle \text{points} \rangle \\ \langle \text{point} \rangle &\models \text{N} \mid \text{S} \mid \text{E} \mid \text{W} \mid \text{NE} \mid \text{NW} \mid \text{SE} \mid \text{SW} \mid \text{C} \\ \langle \text{assign} \rangle &\models \langle \text{variable} \rangle \text{ '=' } \langle \text{point} \rangle \end{aligned}$$

The Terminal point/points have nine non-terminals represented by compass names similar to location names but instead the points refer to a particular point on the focus area: North, South, East, West, North West, North East, South East, South West and Centre. A screen location has nine points and they are named with compass directions. Location terminal and point terminal both have the same non-terminal but the meaning and use of these terminals are completely different. Points are used to produce shapes such as lines *line N S* generates a line from north point to south point.

The terminal *assign* is used with two Non-Terminals, a variable and a point. It assigns a specific point on the screen area to a variable. This variable name can be referenced as a point mainly in drawing shapes. For example the user marks a point on the screen in one zoom level and later gives reference to it from a different zoom level. The user language *assign X to C* means that point C can be called as X from a different screen location.

- SPL 5 - Reference to a navigation route: Ability to inform user position with reference to the starting location; a position on the screen with an effective referencing system of the route

$$\langle \text{route} \rangle \models \text{'route'}$$

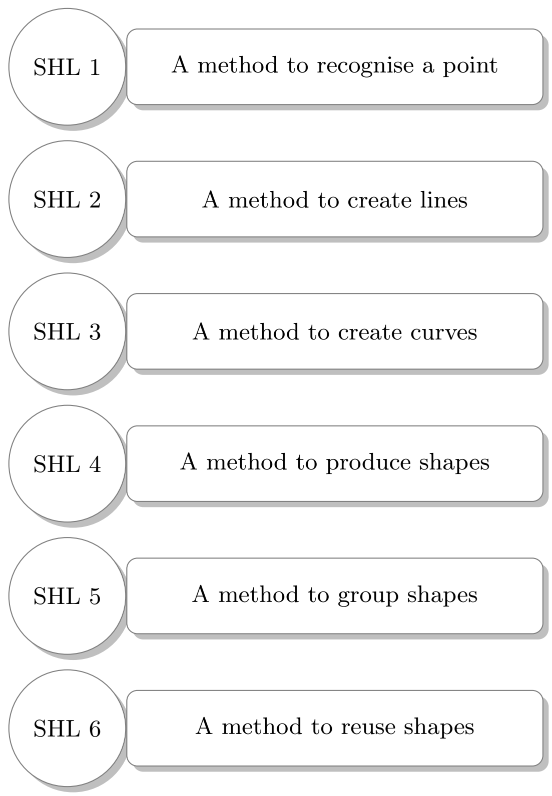
$$\langle \text{position} \rangle \models \text{'position'}$$

The non-terminal *route* gives the name of navigated screen locations. There is no other terminal or non-terminal attached to this grammar specification. The non-terminal *position* gives the current screen location and zoom level.

4.3.2 SHL (Shape Language) Grammar Formation

The shape grammar consideration for the production of diagrammatic drawing techniques for blind people mentioned below is an outcome of the work carried out by many authors that were mentioned in Chapter 3. They are [Kennedy, 1997], [Miller and Johnson-Laird, 1976], [Struiksma et al., 2009], [Kamel et al., 2001] , [Fujiyoshi et al., 2014] and [Fujiyoshi et al., 2008] . These authors introduced empirical work such as line drawing, linguistics production for art, 2D space grid arrangement and command language introduction that led to the production of shape language.

Shape Language(SHL) Considerations



1. SHL 1- A method to recognise a point in space: An element/notion in space that primitive shapes are built that have x and y coordinates in 2D space.

$$\begin{aligned} \langle \text{points} \rangle &\models \langle \text{point} \rangle \mid \langle \text{point} \rangle \langle \text{points} \rangle \\ \langle \text{point} \rangle &\models \text{N} \mid \text{S} \mid \text{E} \mid \text{W} \mid \text{NE} \mid \text{NW} \mid \text{SE} \mid \text{SW} \mid \text{C} \end{aligned}$$

SHL 1 is already introduced in space grammar, however this is equally important with shape production. The terminal *point/points* are already covered in the space language consideration. However *point/points* are important non-terminals for line or curve drawing that are the basis of any other complex 2D shapes.

2. SHL 2 - A method to create a line/lines between points: A line has two end points whereas line segments have at least two end points but can be extended indefinitely.

$$\langle\text{lines}\rangle \models \text{'line'},\langle\text{point}\rangle \mid \langle\text{variable}\rangle,\langle\text{points}\rangle \mid \langle\text{variables}\rangle$$
$$\langle\text{line}\rangle \models \text{'line'},\langle\text{point}\rangle \mid \langle\text{variable}\rangle,\langle\text{point}\rangle \mid \langle\text{variable}\rangle$$

The terminal *line/lines* forms a line between two or multiple points. There have to be a minimum of two points to form a line. The non-terminals can be a finite set of points or user-defined variables. A user can produce many types of 2D shapes using line commands.

3. SHL 3 - A method to create a curve/curves between points: A quadratic curve is defined by two end points and one control point and a cubic curve is defined by two end points and two control points.

$$\langle\text{curves}\rangle \models \text{'curves'}, \langle\text{curve}\rangle \mid \langle\text{curve}\rangle\langle\text{curves}\rangle$$
$$\langle\text{curve}\rangle \models \text{'curve'}, \langle\text{point}\rangle \mid \langle\text{variable}\rangle, \langle\text{point}\rangle \mid \langle\text{variable}\rangle, \langle\text{point}\rangle \mid \langle\text{variable}\rangle$$
$$\langle\text{wave}\rangle \models \text{'wave'}, \langle\text{point}\rangle \mid \langle\text{variable}\rangle, \langle\text{point}\rangle \mid \langle\text{variable}\rangle, \langle\text{point}\rangle \mid \langle\text{variable}\rangle, \langle\text{point}\rangle \mid \langle\text{variable}\rangle$$

The terminal *curve* has one point as mentioned above to produce a curvature and two end points, whereas terminal *curves* can resolve one or more curves. The terminal *wave* has two points to control the curvature and two end points. The points are compass-based points or user-defined variable points.

4. SHL 4 - A method to produce primitive or arbitrary shapes: Create two-dimensional primitive shapes made of points, lines, rectangles, arcs, ellipses, and curves, define a collection of shapes with a name/label to convey a meaning.

$\langle$triangle$\rangle$ $\models$ 'triangle', $\langle$point$\rangle$ | $\langle$variable$\rangle$, $\langle$point$\rangle$ | $\langle$variable$\rangle$, $\langle$point$\rangle$ | $\langle$variable$\rangle$, $\langle$point$\rangle$ | $\langle$variable$\rangle$

$\langle$square$\rangle$ $\models$ 'square', $\langle$point$\rangle$ | $\langle$variable$\rangle$

$\langle$circle$\rangle$ $\models$ 'circle', $\langle$point$\rangle$ | $\langle$variable$\rangle$

$\langle$oval$\rangle$ $\models$ 'oval', $\langle$point$\rangle$ | $\langle$variable$\rangle$

$\langle$process$\rangle$ $\models$ 'process', $\langle$point$\rangle$ | $\langle$variable$\rangle$

$\langle$rec$\rangle$ $\models$ 'rec', $\langle$point$\rangle$ | $\langle$variable$\rangle$

$\langle$arrow$\rangle$ $\models$ 'arrow', $\langle$point$\rangle$ | $\langle$variable$\rangle$, $\langle$point$\rangle$ | $\langle$variable$\rangle$

$\langle$write$\rangle$ $\models$ 'write', $\langle$point$\rangle$ | $\langle$variable$\rangle$, $\langle$string$\rangle$

The terminals *triangle, square, circle, oval process, rec, arrow, write* are some of the predefined shapes. These can be resolved by calling the name and starting point/variable on the screen. The art is then placed in the current cell location matching the current zoom size (zoom size is 1/9 of the existing location). The terminal *arrow* has a starting point and ending point whereas the *write* terminal enables the placing of Braille letters when a starting point is given.

5. SHL 5 - A method to group a set of shapes: To assign a name/label to one or many shapes in order to recognise them later.

$\langle$define$\rangle$ $\models$ 'define', $\langle$name$\rangle$, $\langle$Statementlist$\rangle$

The terminal *define* has a non-terminal name as a string and any number of statements. These statements can be lines, curves or shapes. A user can call multiple shapes and construct an image by giving a name. e.g. *define mytable, rec NW, line W SW, line E SE*

6. SHL 6 - A method to reuse a new shape: Call a non-terminal is used to bring an already created shape by its name/label.

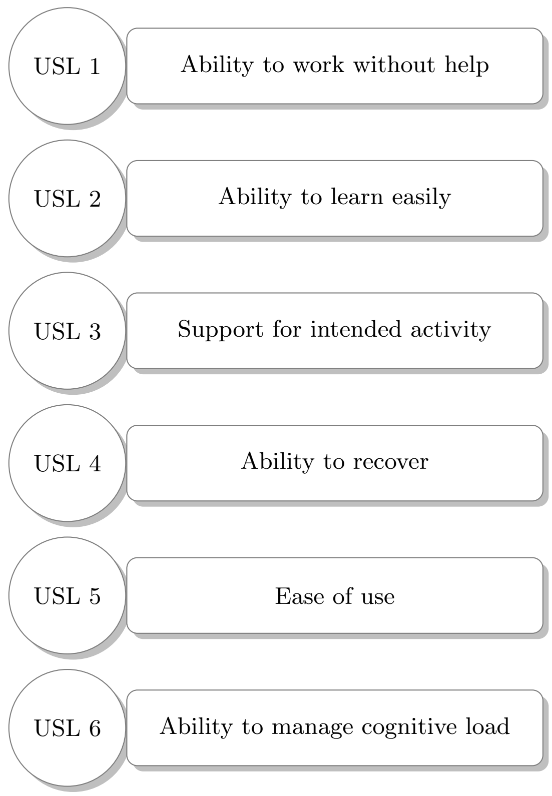
$\langle$call$\rangle$ $\models$ 'call', $\langle$name$\rangle$

The terminal *call* has one non-terminal *name*. The terminal places a previously defined object on the screen by adding the image to the canvas when the constructed image's name is called e.g. *call mytable* The image *mytable* changes its size based on the current zoom level of the system.

4.3.3 USL (Usability Language) Grammar Formation

Usability is critical when designing products for blind people. Different standards can be found in the following articles and websites on standards, frameworks, taxonomies, perspectives and dimensions that are covered in the literature review. Some of the relevant information can be found in [Cook and Polgar, 2015], [Ferreira et al., 2012], [WHO, 2018], [ISO, 2019] , [Hersh et al., 2008], [RNIB, 2018], [Italy, 2019]. The usability language is an analysis of the work of the above researchers and bodies.

Usability Language(USL) Considerations



1. USL 1 - Ability to work without relying on help: Self-initialisation and exiting the application, inbuilt shortcuts and quick access to open, exit and screen reading features.

$$\begin{aligned} \langle \text{open} \rangle &\models \text{'open'}, \langle \text{string} \rangle \\ \langle \text{exit} \rangle &\models \text{'exit'} \\ \langle \text{des} \rangle &\models \text{'des'} \end{aligned}$$

The terminal *open* opens a pre-saved image from a specific destination when a file name is provided. It also also gives the ability to put work in progress on hold and return to it. The terminal *exit* closes the application and it has no additional non-terminals whereas the terminal *des* describes shapes and labelled images on the interface locations as presented in the screen without the need of naming a location.

2. USL 2 - Ability to learn easily: Access to help and different modes of help (voice, Braille) and different levels of "how to use" tutorials.

$$\langle \text{help} \rangle \models \text{'help'}, \langle \text{string} \rangle$$

   The help non-terminal has one non-terminal, which is a string that recognises the area the user needs help with. Suppose the help is needed in drawing a line, the user language *help line* should read how to draw a line on the screen. The system level can be set to low, medium or high. System mode can be changed to Braille.

3. USL 3 - Support for intended activity: Availability of primitive shapes, availability to draw non-primitive shapes, ability to use previously designed art, ability to group and define art.

   This is previously covered by shape grammar. These can be resolved by calling the name of the shape and starting point/variable on the screen.

4. USL 4 - Ability to recover: Features such as save, erase, ability to retrieve previous images and put work in progress on hold and return to it.

$$\begin{aligned}
\langle \text{erase} \rangle &\models \text{'erase'} \\
\langle \text{save} \rangle &\models \text{'save'}, \langle \text{string} \rangle \\
\langle \text{savescript} \rangle &\models \text{'savescript'}, \langle \text{string} \rangle \\
\langle \text{print} \rangle &\models \text{'print'}
\end{aligned}$$

   The above terminal *erase* clears the drawing of current location by clearing the image buffer of object state. The terminal *save* with a non-terminal name saves the image in a designated place with a given name whereas terminal *savescript* saves the written user commands in a text file for later use. The terminal *print* sends the existing canvas image to a 2D printer.

5. USL 5 - Ease of use: Complete an intended task without much delay and effort. Ability to interact with the system without much effort, and some more features such as redo, undo, copy, paste.

$$\langle \text{redo} \rangle \models \text{'redo'}$$

$$\langle \text{undo} \rangle \models \text{'undo'}$$

$$\langle \text{copy} \rangle \models \text{'copy'}, \langle \text{name} \rangle$$

$$\langle \text{paste} \rangle \models \text{'paste'}, \langle \text{name} \rangle$$

The terminal *undo* reverses the previously taken action whereas *redo* repeats the previous action. The terminal *copy* copies a labeled image and places it in a new location using *paste* grammar. This works is a similar way to *call* grammar introduced previously.

6. USL 6 - Ability to manage cognitive load: Simple, short, direct linguistic interaction to complement the working memory of the intended user.

   There is no grammar to demonstrate this consideration, however the commands are designed with short, simple language and the actions can be performed in a shorter space of time, which is evidenced in the thesis experiment chapter.

7. USL 7 - Ability to personalise: System features to pick different levels of interaction pace, interaction method. For example little or more voice feedback, change of system input and output modes from keyboard to Braille to voice, change of interface items and sizes.

$$\langle \text{speed} \rangle \models \text{'speed'}, \langle \text{speed} \rangle$$

$$\langle \text{mode} \rangle \models \text{'mode'}, \langle \text{mode} \rangle$$

$$\langle \text{level} \rangle \models \text{'level'}, \langle \text{level} \rangle$$

$$\langle \text{speed} \rangle \models \text{L} \mid \text{M} \mid \text{H}$$

$$\langle \text{mode} \rangle \models \text{B} \mid \text{S} \mid \text{K}$$

$$\langle \text{level} \rangle \models 1 \mid 2 \mid 3$$

The terminal *speed* has three non-terminals (medium, low, high) for voice speed operation and the terminal *mode* has three non-terminals (keyboard, Braille, speech) for input or even in output mode operations. The terminal *level* has three non-terminals (1, 2, 3) where advanced users can change the level to 3 to avoid unnecessary system details. These non-terminals can be expanded to accommodate new technologies.

Some of the other terminals that are not mentioned here are the facility for mag-

nification which is a future development area for VI users and the integration of commercial screen readers that are not yet evolved to read images.

4.3.4 Context-Free Grammar for a computer-aided drawing System for Blind People

This sections demonstrates the earlier introduced context-free grammar of the space, shape and usability language; its terminals and non-terminals written in Backus-Naur form.

⟨statementlist⟩ ⊨ ⟨abstractstatement⟩ | ⟨abstractstatement⟩⟨statementlist⟩

⟨abstractstatement⟩ ⊨ ⟨call⟩ | ⟨define⟩ | ⟨zoomin⟩ | ⟨zoomout⟩ | ⟨assign⟩ | ⟨write⟩ | ⟨line⟩ | ⟨lines⟩ | ⟨curve⟩ | ⟨curves⟩ | ⟨triangle⟩ | ⟨square⟩ | ⟨circle⟩ | ⟨oval⟩ | ⟨process⟩ | ⟨arrow⟩ | ⟨rec⟩ | ⟨save⟩ | ⟨open⟩ | ⟨print⟩ | ⟨erase⟩ | ⟨route⟩ | ⟨position⟩ | ⟨describe⟩ | ⟨help⟩ | ⟨undo⟩ | ⟨redo⟩ | ⟨copy⟩ | ⟨paste⟩ | ⟨speed⟩ | ⟨mode⟩ | ⟨nullstatement⟩

⟨define⟩ ⊨ 'define', ⟨name⟩, ⟨Statementlist⟩

⟨call⟩ ⊨ 'call', ⟨name⟩

⟨zoomin⟩ ⊨ 'zoomin', ⟨location⟩

⟨zoomout⟩ ⊨ 'zoomout'

⟨assign⟩ ⊨ ⟨variable⟩ '=' ⟨point⟩

⟨route⟩ ⊨ 'route'

⟨position⟩ ⊨ 'position'

⟨write⟩ ⊨ 'write', ⟨point⟩⟨variable⟩, ⟨string⟩

⟨line⟩ ⊨ 'line', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩

⟨lines⟩ ⊨ 'line',⟨point⟩ | ⟨variable⟩,⟨points⟩ | ⟨variables⟩

⟨curve⟩ ⊨ 'curve', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩

⟨curves⟩ ⊨ 'curves', ⟨curve⟩ | ⟨curve⟩⟨curves⟩

⟨wave⟩ ⊨ 'wave', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩

⟨triangle⟩	⊨	'triangle', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩
⟨square⟩	⊨	'square', ⟨point⟩ | ⟨variable⟩
⟨circle⟩	⊨	'circle', ⟨point⟩ | ⟨variable⟩
⟨oval⟩	⊨	'oval', ⟨point⟩ | ⟨variable⟩
⟨process⟩	⊨	'process', ⟨point⟩ | ⟨variable⟩
⟨rec⟩	⊨	'rec', ⟨point⟩ | ⟨variable⟩
⟨arrow⟩	⊨	'arrow', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩

⟨erase⟩	⊨	'erase'
⟨save⟩	⊨	'save', ⟨string⟩
⟨savescript⟩	⊨	'savescript', ⟨string⟩
⟨print⟩	⊨	'print'
⟨open⟩	⊨	'open', ⟨string⟩
⟨exit⟩	⊨	'exit'
⟨des⟩	⊨	'des'
⟨help⟩	⊨	'help'
⟨redo⟩	⊨	'redo'
⟨undo⟩	⊨	'undo'
⟨copy⟩	⊨	'copy', ⟨name⟩
⟨paste⟩	⊨	'paste', ⟨name⟩
⟨speed⟩	⊨	'speed', ⟨speed⟩
⟨level⟩	⊨	'level', ⟨level⟩
⟨mode⟩	⊨	'mode', ⟨mode⟩

$$
\begin{aligned}
\langle \text{variables} \rangle &\models \langle \text{variable} \rangle \mid \langle \text{variable} \rangle \langle \text{variables} \rangle \\
\langle \text{variable} \rangle &\models \langle \text{string} \rangle \\
\langle \text{name} \rangle &\models \langle \text{string} \rangle \\
\langle \text{string} \rangle &\models \langle \text{characters} \rangle \\
\langle \text{characters} \rangle &\models \langle \text{character} \rangle \mid \langle \text{character} \rangle \langle \text{characters} \rangle \\
\langle \text{character} \rangle &\models \textit{any ASCII character} \\
\langle \text{speed} \rangle &\models \text{L} \mid \text{M} \mid \text{H} \\
\langle \text{mode} \rangle &\models \text{B} \mid \text{S} \mid \text{K} \\
\langle \text{level} \rangle &\models 1 \mid 2 \mid 3 \\
\langle \text{location} \rangle &\models \text{N} \mid \text{S} \mid \text{E} \mid \text{W} \mid \text{NE} \mid \text{NW} \mid \text{SE} \mid \text{SW} \mid \text{C} \\
\langle \text{points} \rangle &\models \langle \text{point} \rangle \mid \langle \text{point} \rangle \langle \text{points} \rangle \\
\langle \text{point} \rangle &\models \text{N} \mid \text{S} \mid \text{E} \mid \text{W} \mid \text{NE} \mid \text{NW} \mid \text{SE} \mid \text{SW} \mid \text{C} \\
\langle \text{nullstatement} \rangle &\models \lambda
\end{aligned}
$$

4.4 Opportunities Afforded by CADB

The open model of CADB interaction described in this chapter answers the question: How is a CADB model be formally designed? It qualifies with one such answer as it does not model any capability that can be coupled with modality or concrete functionality. A different conceptual framework would introduce a different view of CADB, therefore it could give a different answer to the question. The model does not claim to be the only possible characterisation of CADB. Furthermore, both non-grammar based systems and grammar-based systems provide alternative characterisations. The contribution of this thesis lies in providing one such characterisation.

In summary, the abstract model of CADB functionality contributed by this chapter possesses the following beneficial characteristics:

- Modality independence: Different forms of input and output are not restricted by the solution and can be coupled and used with a formal grammar.
- Content and presentation independence: Grammar allows independent creation of art and independent access to art.
- Freely define art: The model allows links between shapes and forms a defined shape.
- Dynamically generate art: The art is created automatically with the input of user language.

- Freely navigate the screen: Screen movement is allowed using a grid-based navigation method.
- A core drawing functionality: The model delineates the scope of CADB actions and core functionalities as a result of user interaction.

Explained above are opportunities for a grammar-based system. Grammar-based systems may not be suitable for all software applications. For examples, if

- The development is an upgrade or an extension of an existing system.
- The system has limited time, budget and capacity.
- The development team is not competent with grammar-based models in the process of software engineering.

4.5 What is Next?

This chapter covers the design decisions of the SETUP09 formal model, its architecture and formation of space grammar, shape grammar and usability grammar. In Chapter 5 the formal model is implemented and formally validated using an abstract syntax tree with a set of user language and production of art. This shows the system functionality needed for a user language action. Chapter 5 also shows the transformation of the SETUP09 model into an example implementation of the model by explaining each stage of the implementation process. The process elaborates with parse components to resolve the user language with lexical and semantic analysis.

Chapter 5

The SETUP09 Model Implementation

The previous chapter introduced a Context-Free Grammar model specification for the CADB system whereas this chapter implements the proposed model (SETUP09) as a grammar-controlled language by means of an interpreter explaining each stage of the implementation process. The next logical step in this thesis is achieved by model implementation, creating a drawing language, validation and setting the foundation for testing with users. This chapter shows how the model specification is transformed into a drawing language that enables validation of the model, its syntax and semantics. The approach of Chapter 5 is that the implementation uses java to implement grammar-controlled interpreter. The chapter examines the different stages of an interpreter, firstly through the lexical stage of reading and resolving inputs, secondly through lexical-semantic analysis for the meaning resolution, and finally the parser stage where the commands get resolved by the production the output art. The model is then validated using abstract syntax tree structures testing several language commands and their outputs. The chapter also presents user interface development activities took place alongside the specification development. The SETUP09 grammar model now has the backend interpreter that resolves the specification and front-end user-interface that interacts with the user. This led to the next phase of the research was to test the utility and usability of the "service" or "assistive technology" enabled by SETUP09 model with target users which is described in Chapter 6.

5.1 SETUP09 Backend Class Interaction

SETUP09's automatic command to art works convert with the interaction of classes shown in Figure 5.1 - abstract statement super-class, other concrete sub-classes, parser, interface and state object. One of the essential classes in the prototype development is *state class* which keeps track of cursor location, co-ordinates, zoom level, image buffer and also a list of user commands and user-defined points. *State object* works as the temporary data structure of the running programme, working in the computer's RAM. The interpreter communicates with the state object at the start or during the target art generation. It also communicates at the end of target art generation to update the new values of temporary variables such as cursor location, co-ordinates, zoom level and image buffer. Interface classes provide access to speech, Braille, keyboard and raised line printer, image synthesiser, external files and tactile interfaces. Although these classes are not fully developed in the SETUP09 prototype but it is designed to accommodate different inputs and output classes. Another important class is *Parser* class that processes lexical analysis, parsing and semantic analysis when a command is input into the prototype before the target art is generated. Other abstract and concrete classes are designed with an interpret method to generate target art once the parser is resolved. Collectively these classes form the work of a simple interpreter and the interaction flow is shown in Figure 5.2.

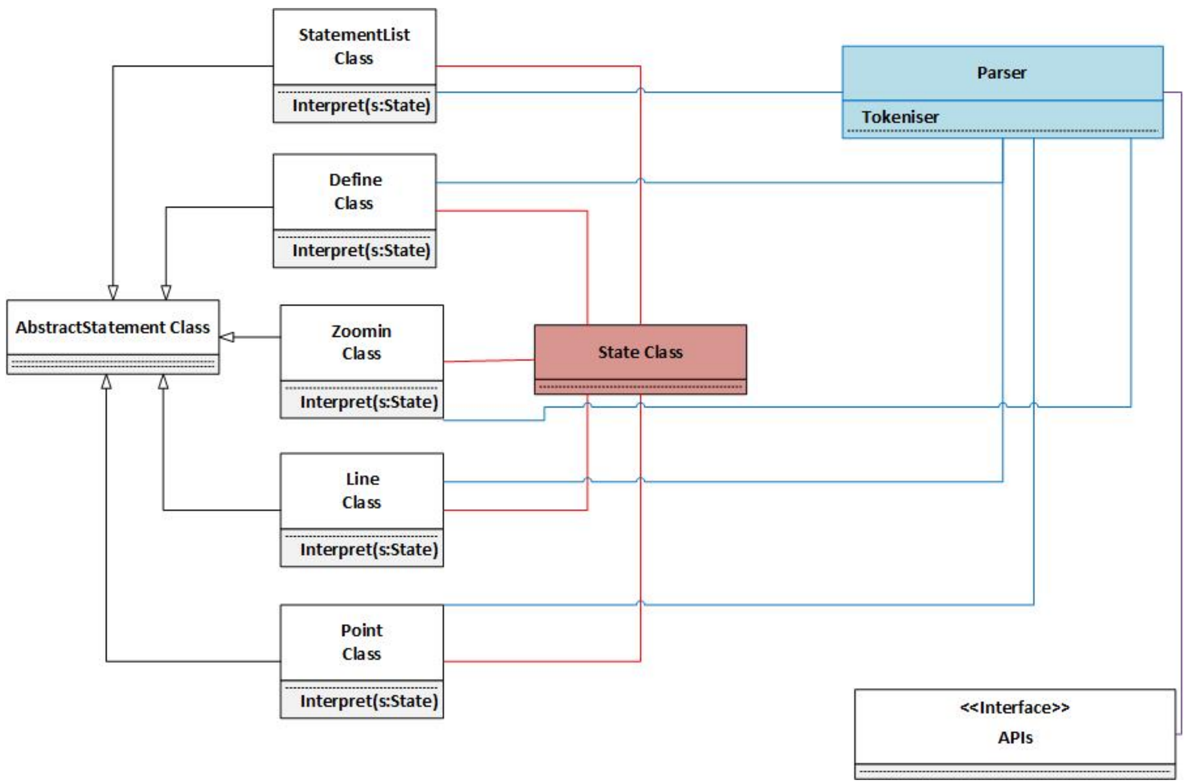


Figure 5.1: SETUP09 Class Interaction

Figure 5.3 shows class interactions needed to draw a line. The main interface *TestGame2019Experiment* class obtains user input, through a default set to keyboard modality. The *interface* class imports *TokenClass* and access methods such as *Myparser*, *Token*, *Tokenizer* to covert the user language to a *StatementList* which is a kind of a *AbstractStatement*. Then the user command finds its specific *AbstractStatement* to execute the target code (art) generation. In Figure 5.3, the *Line* class a type of a concrete class also uses *Point* class another type of a concrete class to find the start point and end point of a line. The output changes the system *state*. Then the *TestGame2019Experiment* interface class obtains a new *state* that contains a line shape in the interface as a result of user command execution.

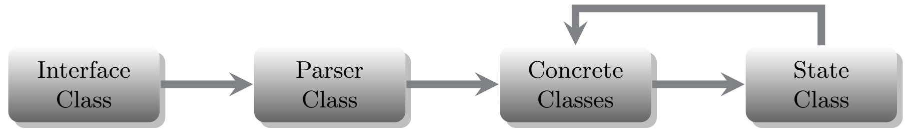


Figure 5.2: SETUP09 Interaction of Classes

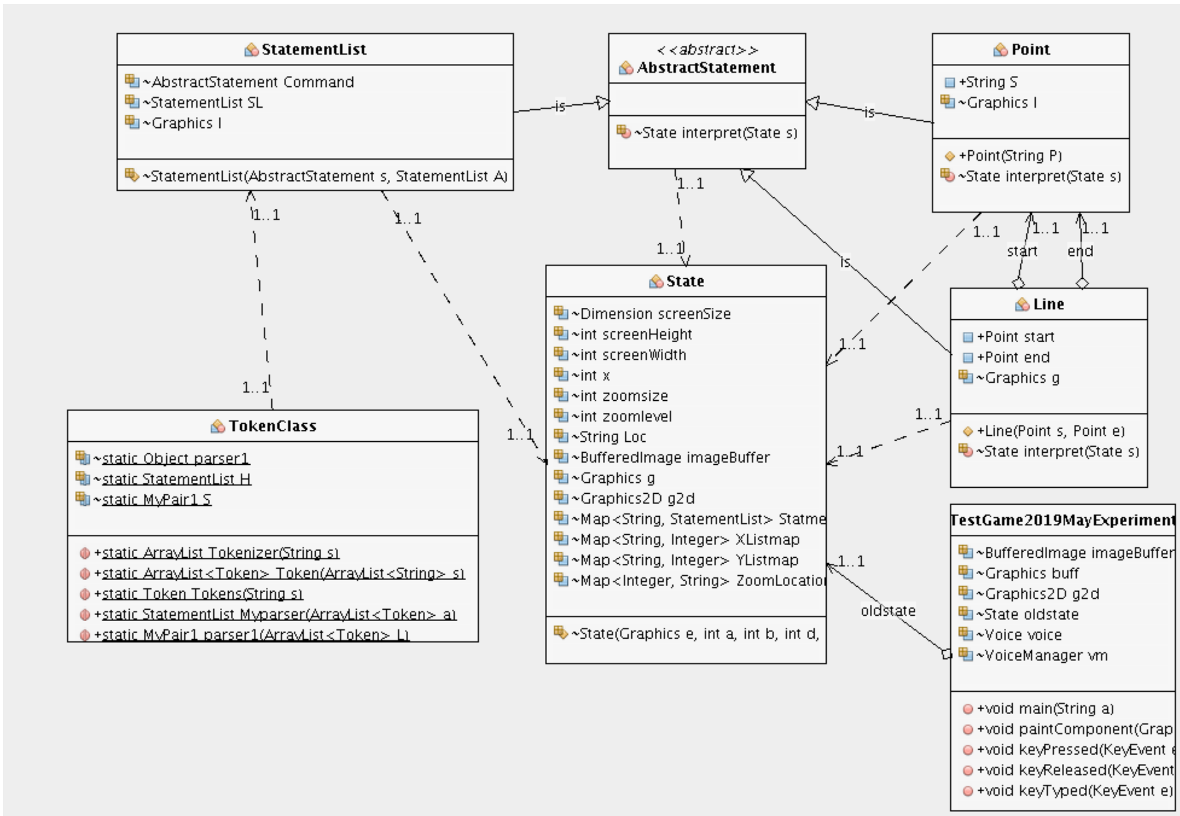


Figure 5.3: SETUP09 Class Interaction to draw a Line

The above class interactions and work is for a line drawing. The same process takes place for other 2D shapes, but with different concrete classes: circle, square, polygon, arrow, write, and call. The section below expands the compilation of user commands that are resolved with lexical analysis, semantic analysis, parsing, and also abstract syntax tree that test the model execution accuracy of SETUP09.

5.2 SETUP09 Command Interpreter Development

The front end parser diagram shown in Figure 5.4 consists of several stages: user specification, lexical analyser, parsing, semantic analysis and interpretation [Appel, 2002]. Input and output classes are not discussed at this stage but assume that the user commands (text, speech or Braille) are read and identified as tokens (set of lexicals). User specifications can be one or many command line arguments. They are then passed through a lexical analyser, resolving input into tokens and then to the parser including semantic analysis. Tokens are resolved one at a time until a meaning is found and until the end of command line arguments. Finally, user specification generates diagrammatic art with the use of JDK: *java.awt.Graphics2D* library.

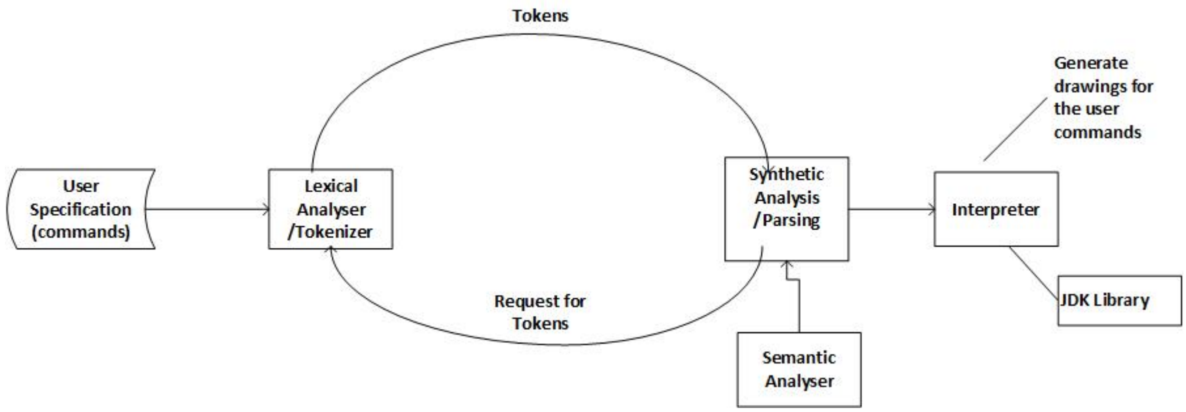


Figure 5.4: Parser Flow

5.2.1 Lexical Analysis

The first stage of the interpreter is to read various input, such as names, numbers, space, marks, and resolve the input into words or tokens. This stage does not deal with logic problem solving. For example the command "line W E " returns three tokens namely **line, W, E** identifying tokens separated with the space character.

```
public static ArrayList Tokenizer(String s) {

    ArrayList items = new ArrayList();

    StringTokenizer st = new StringTokenizer(s, " ", true);

    while (st.hasMoreTokens()) {
        items.add(st.nextToken());
    }

    return items;
}
```

The Java method name *Tokenizer* in the above code acts as a lexical analyser returning tokens as items.

5.2.2 Lexical Semantic Analysis

This stage is dedicated to looking at every input expression, checking for the correct type, translating and assigning it into an intended representation, resolving the meaning of the tokens. It returns a token type and value validating its meaning. The task of the semantic analyser is to check the types for correct assignment. The SETUP09 command language uses integers to keep type attribute and string type to keep its value.

```
static class Token {

        int type;
        String Value;

        Token(int t, String v) {
            type = t;
            Value = v;

        }
    }
```

The above *Token class* has two attributes and one method. *Type and value* are attributes where *token* is the method. The purpose of the class is to looks at the parsed token, see if any meaning can be generated, and if so, a type and a value are produced.

Suppose the command language passes input **line W E**, the command is broken into tokens and produce token types: [type(1), type(2), type(3)] and values: [value("line"), value("W"), value("E")]

```
if (s.equalsIgnoreCase("line")) {
            return new Token(1, "line");
            }
if (s.equalsIgnoreCase("W")) {
            return new Token(2, "W");
            }
if (s.equalsIgnoreCase("E")) {
            return new Token(3, "E");
            }
```

5.2.3 Parsing

Parsing puts together a phrase of the source programme and resolves the issue of expressive power of the language of context-free grammar. There are many ways of applying transformation to the grammar such as LR(k) technique (Left-to-right parse, right most derivation, k-token lookahead). The parser takes token generated by the lexical analysis stage, to find a pattern. It will then call the required actions by defining the syntax of the language.

Class **Mypair** enables the user to input more than one drawing command in the command prompt. **Mypair** separates first **AbstractStatement** from the rest of the tokens and assign it to the **first** part of the pair and resolves one statement at a time. The remaining input commands automatically get assigned to the **second** part which is an Arraylist of tokens.

```
static class MyPair {
    AbstractStatement first;
    ArrayList<Token> second;
    public MyPair(AbstractStatement f, ArrayList<Token> s) {
        first = f;
        second = s;
    }
}
```

The method **Myparser** checks the user commands' first part and second parts. Commands are then passed to the parser function that resolves the syntax of the language until the user commands are not empty.

```
public static StatementList Myparser(ArrayList<Token> a) {
    StatementList b = null;
    MyPair p = parser(a);
    if (!p.second.isEmpty()) {
    return new StatementList(p.first, Myparser(p.second));
    }
    return new StatementList(p.first, b);
}
```

The method **parser** checks the token type. It calls the required function if a predefined token type is found. In the example code below, the parser identifies a valid token type which is **1** for the user command **line W E**, arranges tokens into a syntactically correct sentence and then parses the *AbstractStatement* to the next stage of art production. The syntax of the parser function is based on context-free grammar introduced in the previous section.

```
public static MyPair parser(ArrayList<Token> L) {
    if (L.get(0).type == 1) {
            String Y = L.get(2).Value;
            String Z = L.get(4).Value;
            S = new MyPair(new Line(new Point(Y), new Point(Z)), L);
            }
}
```

In the method above, the input token *arraylist* is checked for type *1* which is *line* token. If it is found, then subsequent tokens are read, *W, E.* This is inline with Line grammar introduced in Chapter

4.

$$\langle \text{line} \rangle \models \text{'line'},\langle \text{point} \rangle,\langle \text{point} \rangle$$

The above sections demonstrate an example of a command resolution with front-stages of interpreter implementation in Java. In practice, interpreters/compilers need a compiler application to resolve the front and back stages of such grammar, examples of which are Yacc, Lex, OCaml, Nim [Appel, 2002]. The SETUP09 interpreter has been successfully used to demonstrate the front stages of lexical analysis, semantic analysis, and parsing by resolving context-free-grammar introduced in Chapter 4. It was found that it is not sufficient to check if a command sentence belongs to the SETUP09 grammar model, but it must also be evidenced in the production of art. The next section demonstrates accuracy of the SETUP09 grammar model by displaying if semantic actions of the parser also produce useful art.

5.2.4 Semantic Validation - Abstract Syntax Tree (AST)

The abstract syntax tree in Figures 5.5 shows the phrase structure of the programme, with all parsing problems rectified without resolving the meaning of the input programme block. A given user command can be visualised in the following format, using an abstract syntax tree. Error free context-free grammar should only produce one possible parsing tree structure for a given user command. The AST portrays the visualisation of the lexical processing flow therefore input/out mechanisms are not discussed at this stage. See the examples below in Figures 5.5 and 5.7.

- Example 01 - Drawing of a line for the user command:
  - *line E W*

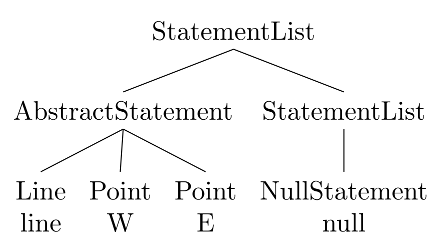


Figure 5.5: Abstract Syntax Trees (ASTs)

new StatementList(new AbstractStatement(new Line("line", new point("E"), new point("W"),null)));

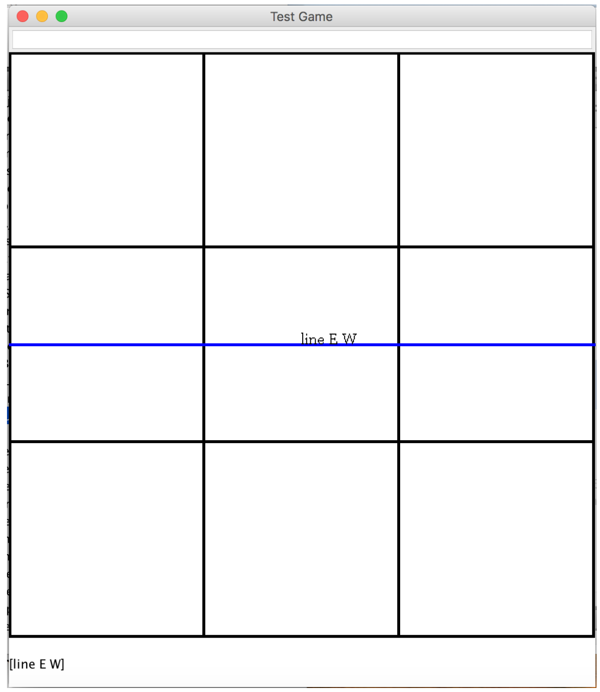


Figure 5.6: Image of a Line (in blue) from Prototype SETUP09

- Example 02 - Drawing of a circle for the user command:
  - *Circle NW*

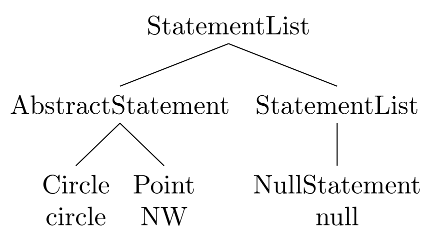


Figure 5.7: Abstract Syntax Trees (ASTs)

new StatementList(new AbstractStatement(new Circle("circle", new point("NW"),null)));

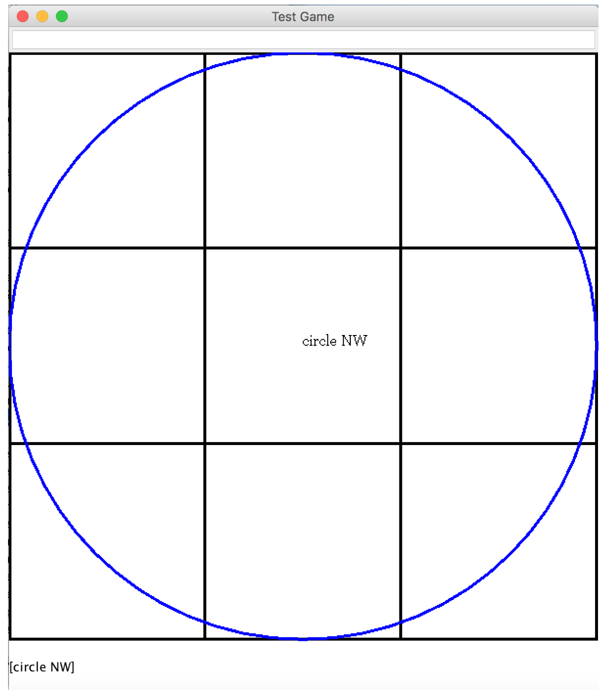


Figure 5.8: Image of a Circle (in blue) from Prototype SETUP09

- Example 03 - Save command:
  - *Save Image-Fish*

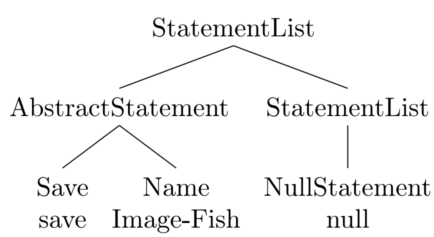


Figure 5.9: Abstract Syntax Trees (ASTs)

new StatementList(new AbstractStatement(new Save("save", new string(Image-Fish),null)));

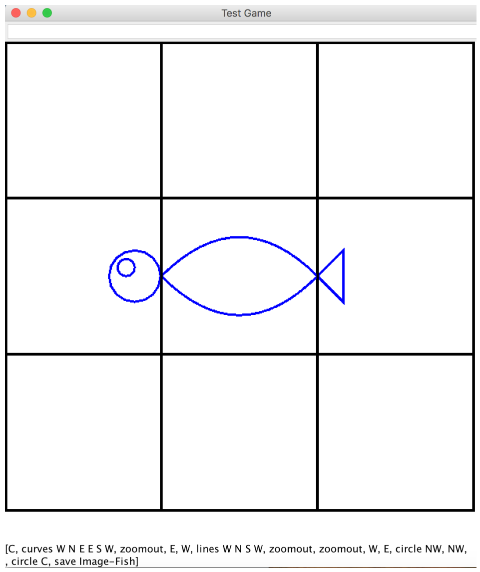


Figure 5.10: Image of a 2D Fish from SETUP09 Prototype

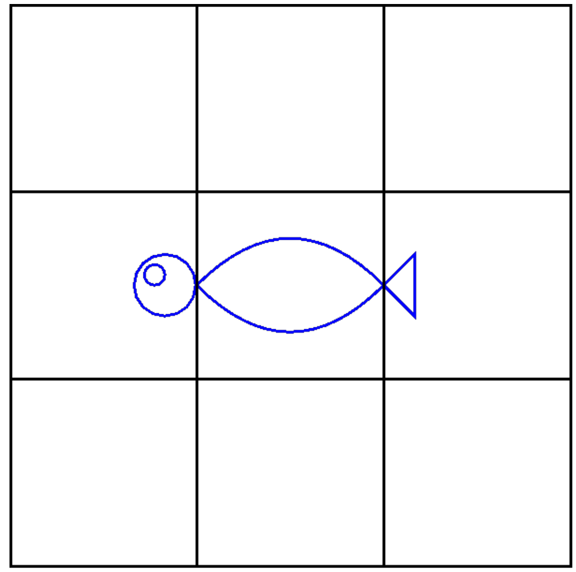


Figure 5.11: Image of a 2D Fish from the Save Command

- Example 04 - Drawing of a house by labeling it with *Myhouse* in order to be used again by calling the name *Myhouse*. Note that command *lines* can have multiple points where *line* can only have two points. The user command as below:
  - *define Myhouse, lines E SE SW W E N W, call Myhouse*

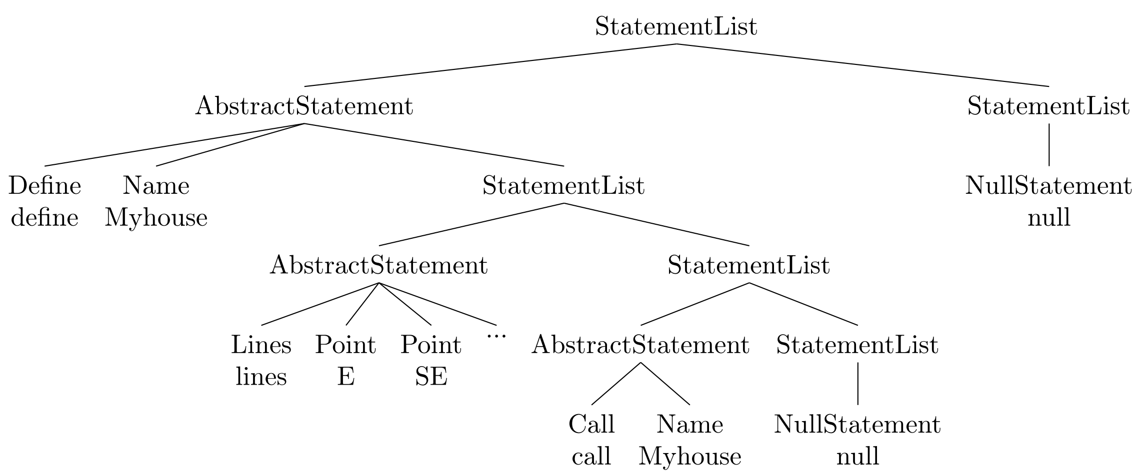


Figure 5.12: Abstract Syntax Trees (ASTs)

new StatementList(new AbstractStatement(new Define("define", new name(Myhouse), new StatementList (new AbstractStatement(new Lines("lines", new point("E"),new point("SE"),new point("SW"), new point("W"), new point("E"), new point("N"), new point("W"))) new StatementList (new AbstractStatement(new call("call", "Myhouse",null)),null)))));

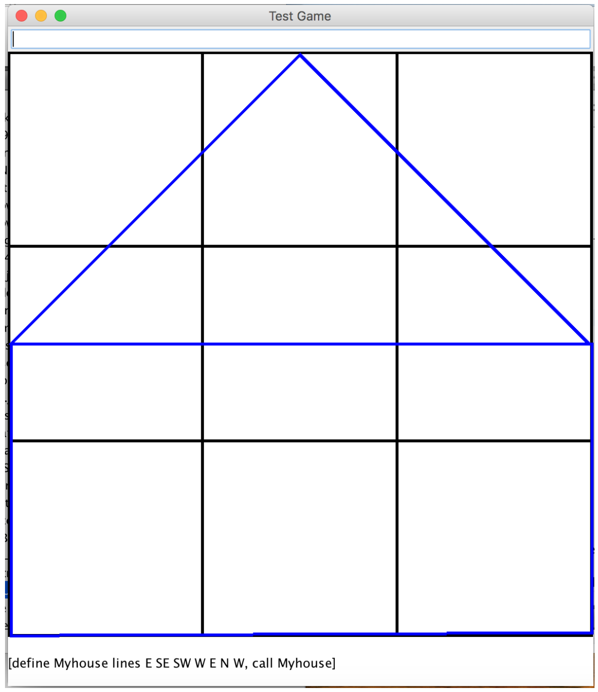


Figure 5.13: Image of a house from Prototype SETUP09

5.2.5 Target Code (Art) Generation

The parser calls the relevant classes to initiate the intended presentation of the drawing. The codes below represent the parsing of the "Line W E" command by calling Line Statement. The *line* class interprets the drawing of a line based on the user input points. The *line* class also returns a new state to the system with new parameters.

```
public  class Line extends AbstractStatement  {
 public Point start;
 public Point end;
 Graphics g;
     public Line(Point s, Point e)
     {
      start=s; end=e;
     }
 @Override
  State interpret(State s)
       {
         start.interpret(s); end.interpret(s);
         drawLine(startx,starty,endx,endy);
         return (new State);
        }  }
```

Line concrete class is a child class of *AbstractStatement*. Line class uses *point concrete* class to resolve *W E* start and endpoints. They get resolved by overriding the interpret method in the *AbstractStatement* class. First, it resolves start and end points of a line calling *start.interpret(s) and end.interpret(s); methods* and then it calls for the DrawLine method of 2D Java library.

Provided above were the visualisation and evidence of the source code execution of SETUP09 language using AST and its resultant art production. AST validates the accuracy of the SETUP09 model presented in Chapter 4. The next section discusses the front-end development of the SETUP09 interface that evolved during the research with feedback from different people.

5.3 System Frontend Development

According to Jenny Preece, [Preece et al., 2015] a user-centered approach does not always require much potential user involvement at the design stage. Where there is no user involvement it is important to gather preliminary information on the design aspects using surveys. However, user involvement in system testing is vital to validate the design. In some cases limited participants or a closely-involved potential user may be suitable due to practical difficulties, project time restrictions, limited user experience and diverse user interests. In such cases initial surveys and system testing can be expanded with wider audiences.

In order to built useful and ease computer systems, the design process needs to understand the user, their interaction with simulations, and manage iterative design. This section brings the early finding discussed in Chapter 3, user interaction, and SETUP09 development iterations into the attention. The design of the front-end didn't take place as a result of the back-end (interpreter) development, but started in the early stages of the research and evolved when the research direction changed. Hence there are several versions of the system development presented in this section.

The interface development of the system in this project is based on four phases. The design team included a sighted person (researcher and designer), a blind undergraduate student who was closely involved throughout the development process, and ten other blind participants who were loosely involved.

The blind student had knowledge of game interfaces Figure 5.14 shows the blind person operating the computer keyboard with the researcher. This approach provided quick feedback in the development of navigation interface. The student was able to help with determining the labels for the screen locations and the appropriate vocabulary.

Figure 5.14: A blind Undergraduate Student Team Member Operating a Computer Keyboard with the Researcher.

The navigation interface was verified by using a scenario conveyed as a textual narrative and printed on tactile paper. Additional features were added to the navigation interface that were identified from observations. The detailed descriptions of interfaces were created, descriptions of prototype features were refined, the overall scenario was constructed and target users were invited to interact with it. A similar process was introduced in [Sahib et al., 2013].

Step 01 - Identified features from observations: When identifying a set of interface features the blind team member contributed significantly from his experience of using computer game such as interactive fiction games and auditory interfaces and their navigation.

Step 02 - Created detailed descriptions of interfaces: Detailed descriptions of all interface features were created in order to build an accurate scenario and explain each system functionality. The blind team member's involvement was significant to ensure that system commands were appropriate and that the correct language was used to describe the commands.

Step 03 - Refinement of system features: Several informal conversations took place to ensure that the blind user had the same understanding as the researcher. The blind user came up with interaction options that are used in most screen readers. For example, the shortcut keys of save, copy, paste and open file commands.

Step 04 - Construction of the overall scenario: The overall system functionalities were created like a story whereby a user draws an image for the first time after hearing it from a friend in a classroom environment.

Step 05 - Dialogue-simulated interaction with target users : Formative evaluations were conducted with potential BVI users based on an overall scenario. The designer engaged dialogue with potential users to produce interaction with the system. Therefore the designer would explain system functionalities and ask for feedback at the end of every step.

Three different versions of the system were developed, the first with the initial idea of nine screen divisions, the second with knowledge-based rules for drawing commands, and the third version was with the implementation of SETUP09 interpreter language.

First Development

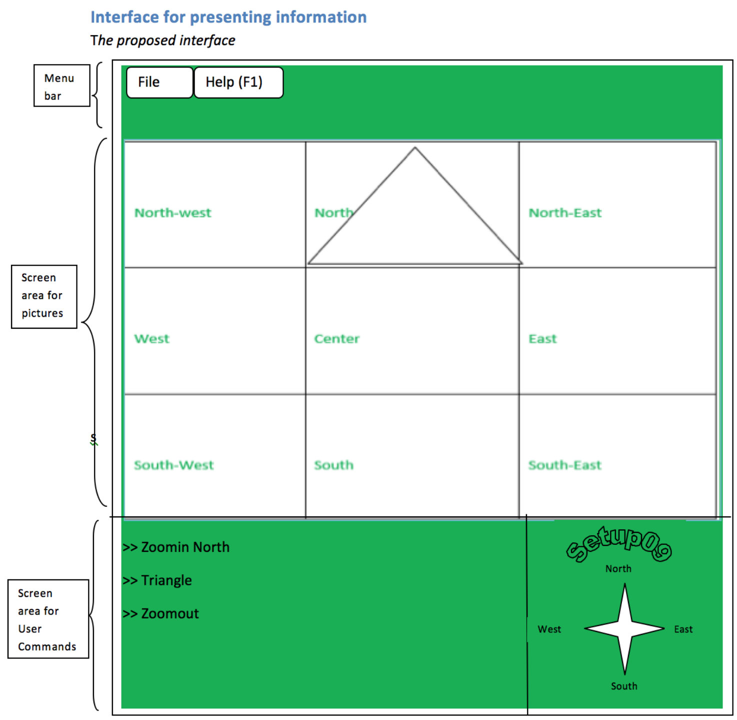


Figure 5.15: prototype SETUP09 Version 01

A scratch version of the user interface was designed and printed in swell paper with the consultation of blind participants. The screen interface in Figure 5.15 of the screen interface demonstrates the initial SETUP09 design screen with three sections: an area for user input commands, a menu for usability functions such as file, save, and help (with assigned shortcut keys), and a main body, drawing area divided in to nine screen locations as proposed. At this stage the idea of a point system was not fully developed but the concept of the automatic conversion of the command language to a drawing was formed. Many blind people mentioned the need for computer support with art generation even though they wanted to manage the drawing process in the initial surveys. The second iteration introduced computer-supported 2D shape drawings.

Second Development

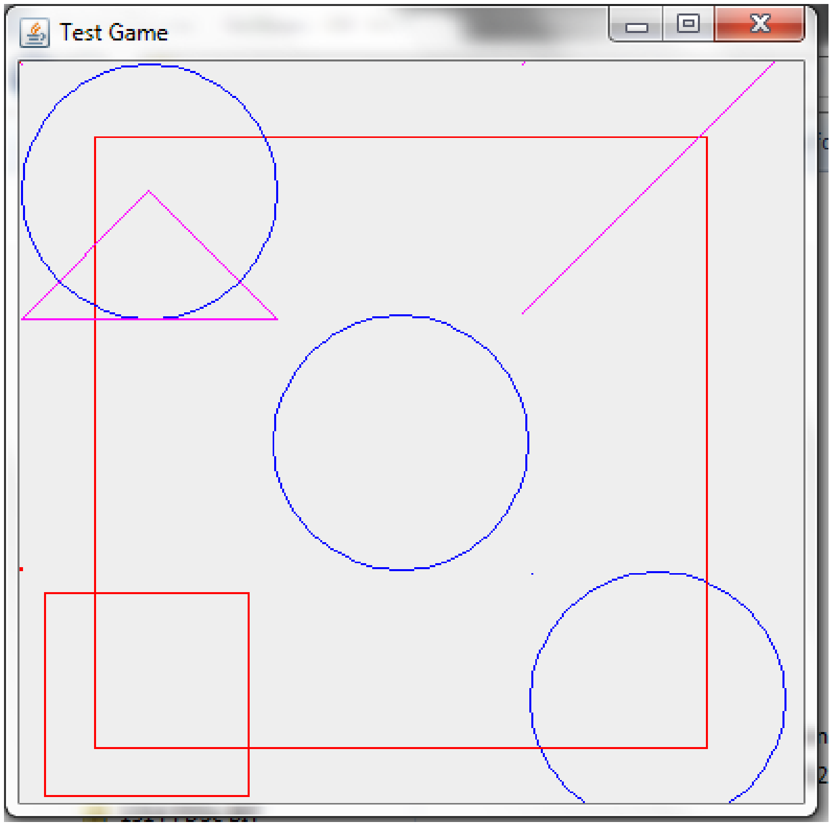


Figure 5.16: prototype SETUP09 Version 02

Figure Figure 5.16 shows the interface of the second stage of software development with a few 2D shapes on the screen. At this stage, the idea of a multiple points system as opposed to a centre point cell reference was identified to be able to draw and connect more shapes quickly and efficiently. The system was only used to draw shapes with points. Only a few primary shapes were tested and no actual system evaluation took place. At the end of the second build, it was possible to use the system to produce a few shapes in nine cells and nine points in a cell system.

In the development of the second iteration, several programming languages were tested like Java programming and Prolog, [1]. System tests identified that normal operations, such as selection and iteration, were not sufficient for a user command language. There was an obvious need for an expressive language, and the introduction of a grammar-based user command.

However we first tested the logic in Prolog during the second iteration. Prolog is a rule-based logical language that can retrieve logic from databases. The SETUP09 problem was easily solved by a set of grammar rules rather than declarative logic. The need was more for an expressive user language which has drawing capabilities and to be able to process the source user commands to target art without translating into machine language. Designing an interpreter was a solution to achieve the drawing outcome. The user language was simple enough, with no arithmetic expression only a set of

[1] http://www.swi-prolog.org/

string tokens. The use of programming data structures to design and implement interpreter classes and use of its 2D libraries was a suitable solution compared to other interpret language design such as Python, [Foundation, 2019]. The important outcome of the second iteration was the system parser design, allowing the system with lexical analysis, semantic analysis, parsing and target generation.

Third Development

Figure 5.17 illustrates the startup screen for SETUP09, where the screen divided into nine cells in a grid structure. It has a text area to capture keyboard inputs at the top, a central panel to display graphics and another display area for listing the history of input commands. The cursor navigation paths are demonstrated in Chapter 6. The cursor can move to different cells and can also move within a given cell location.

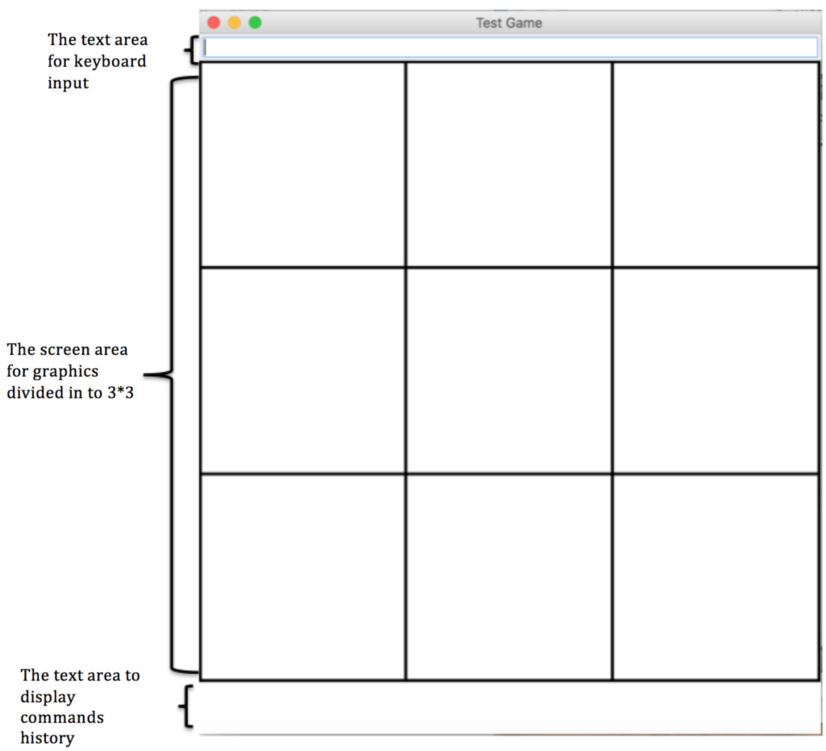


Figure 5.17: Interface of the Third Version of SETUP09

The third prototype was operated by inputting keyboard commands. The keyboard arrow keys were used for navigation, and for zoomin and zoomout. Other hot keys were designated for commands such as *des, position, help and exit.* The third version hot keys and commands are demonstrated in Tables 5.1 and 5.2.

Command	**Hot Keys**
Zoomin C	alt
Zoomin W	left arrow
Zoomin E	right arrow
Zoomin N	up arrow
Zoomin S	down arrow
Zoomin NW	space + left arrow
Zoomin NE	space + up arrow
Zoomin SE	space + right arrow
Zoomin SW	space+ down arrow
Position	numeric key 8
Description	numeric key 9
Zoomout	black slash
Help	esc

Table 5.1: Hot Key Commands for the Third Version

Arrow keys are used for basic navigation for North, South, East and West screen areas. They were chosen based on participants' feedback. In fact, many other keyboard keys were considered for NW, NE, SW and SE screen areas. It was found that arrow keys with space bar were less confusing and easier for a blind person to use and remember. Different commands can be used to produce the same image using arrow keys. For an example , a square can be produced by *lines NW NE SE SW NW* or *square NW* . Even *line NW NE, line NE SE, SE SW, SW NW* commands can also produce a square. A user can also execute the command *call mysquare* to use a previously created and labeled square.

Other Non Hot Key Commands of the Third Prototype Iteration	
line [point point]	lines [point point*]
curve [point point point]	curves [point point point*]
waves [point point point point], write [point text]	assign [variable point]
circle [startpoint], triangle [startpoint]	square [startpoint], diamond [startpoint]
rectangle [startpoint], long rectangle [startpoint]	process [startpoint], arrow [point point]
define [commands*]	call [name[
save [name], open [name]	exit

Table 5.2: Other Non Hot Key Commands of the Third Prototype

Table 5.2 shows a list of the fully developed commands. Note the asterisk denotes the multiple occurrence of the same non-terminal. Usability enhancement is a never-ending process; the system needs testing and iterative development with target users. The system is not fully developed with all the grammar terminals introduced in Chapter 4, but it is developed sufficiently to carry out user experiments to prove the concept of SETUP09. Some of the system-produced images are shown below.

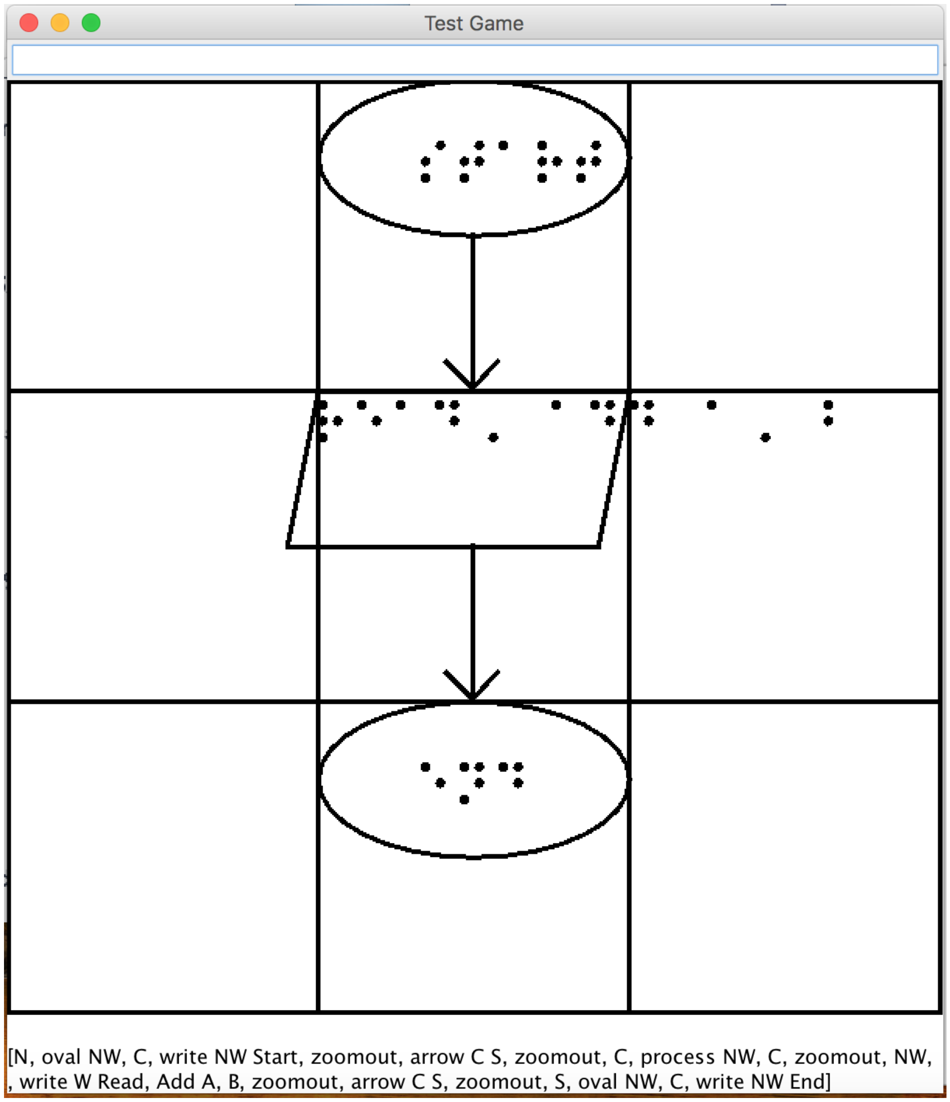


Figure 5.18: SETUP09 Version 03: Flow Chart

Figure 5.18 illustrates an image of a flow chart with start, process and end symbols; the text is in braille letters. The text font is set to a large size in order to get a better tactile quality when printed. The images are created with black colour to improve the tactile quality of lines when embossed.

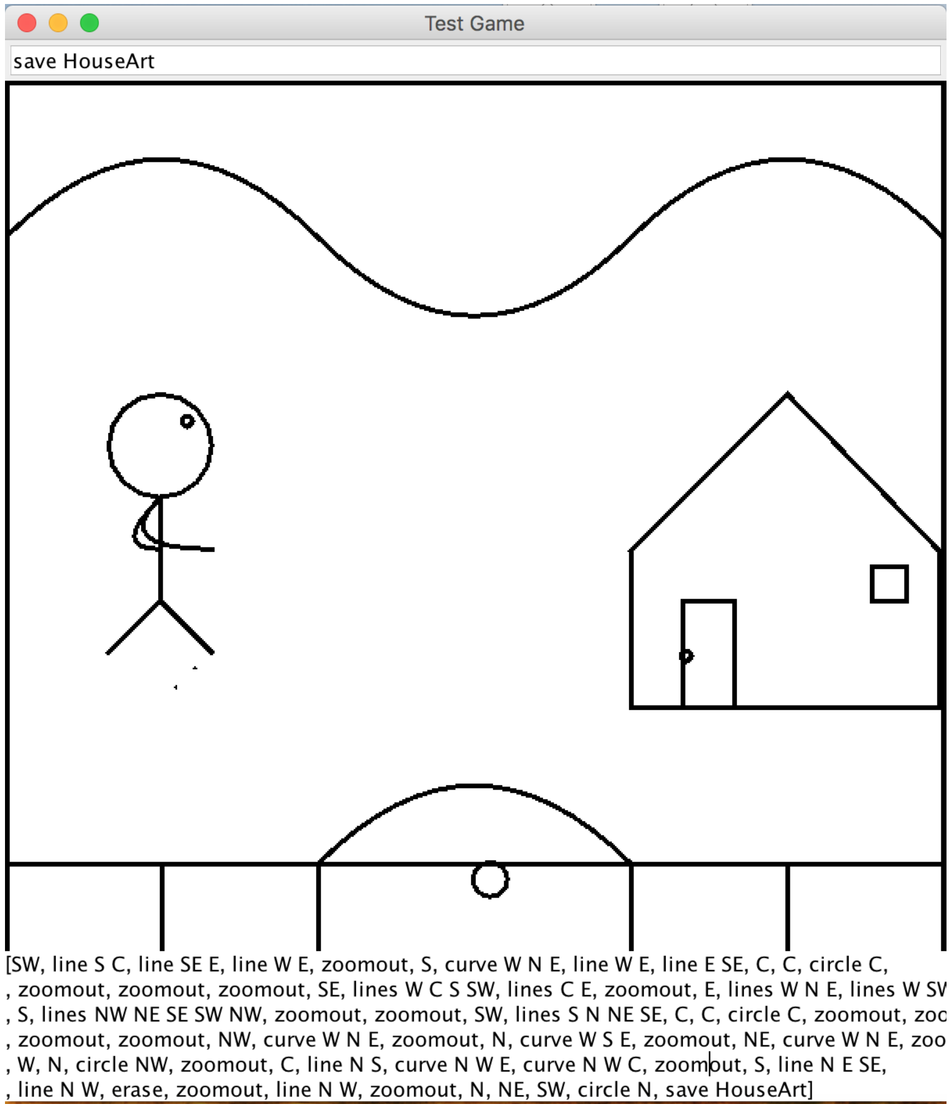


Figure 5.19: SETUP09 Version 03: A scenery of a home

Figure 5.19 illustrates an image of a house with a fence, the sky and a person. 2D shapes such as lines, curves and circles are used to construct the images. The user commands are visible at the bottom of the panel.

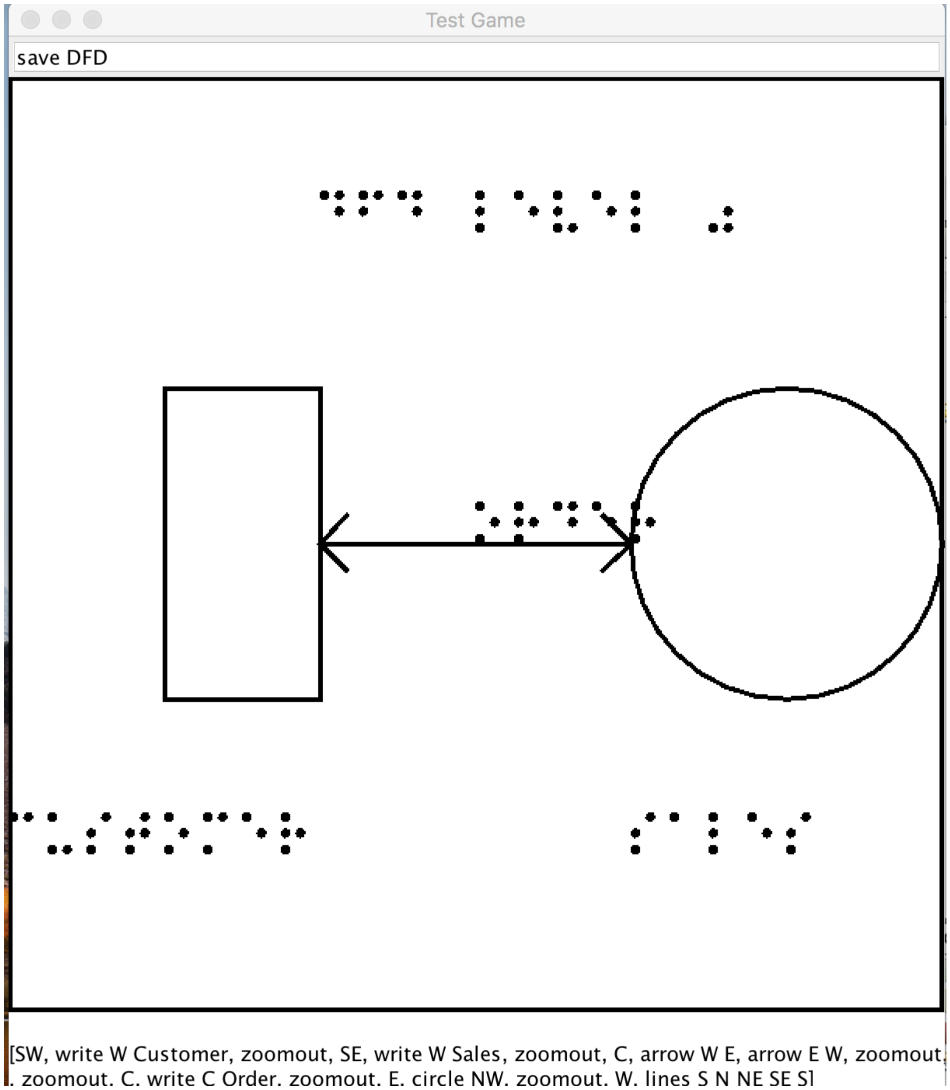


Figure 5.20: SETUP09 Version 03: Image of Data Flow Diagram

Figure 5.20 illustrates an image of a data flow diagram (DFD) level 0, showing symbolically a customer and order system that are represented by a long rectangle and a circle.

The development of the third version of SETUP09 included the implementation of a parser and the user commands to generate target code. Some functionality were not fully implemented in the prototype due to time constraints but can be implemented in future iterations. As can be seen in Figure 5.20 , the interface captures input commands at the top of the screen with a text box, and the main body of the screen below contains the created drawing. The bottom part of the screen displays the history of the input commands. System usability commands are invoked by input commands. This approach avoids the use of a menu on the screen. The developed programme (SETUP09) once selected on a computer initiates itself and then , asking the user to input commands via a speech synthesiser.

In the third stage, the system and grammar were expanded from user feedback. As a result keyboard shortcut keys and many more usability functions, such as erase, redo, undo, help and route, were implemented. The system is purely driven by a command language hence users need to get familiar with it.

In the third iteration, the SETUP09 navigation and graphics creation method was tested several times, first to evaluate the navigation model, then to evaluate its drawing capabilities. The evaluation was done with several subsets of users, first with blind users and then with with visually impaired users and finally with sighted users. The system was also tested against traditional special paper-based drawing kits used in blind schools and with another digital system. More about system evaluation and its results are discussed in the next chapter.

Explained in the next section are the proposed system input-output layers and the system architecture. It discusses different input-output modalities and the possibility of modality integration with SETUP09.

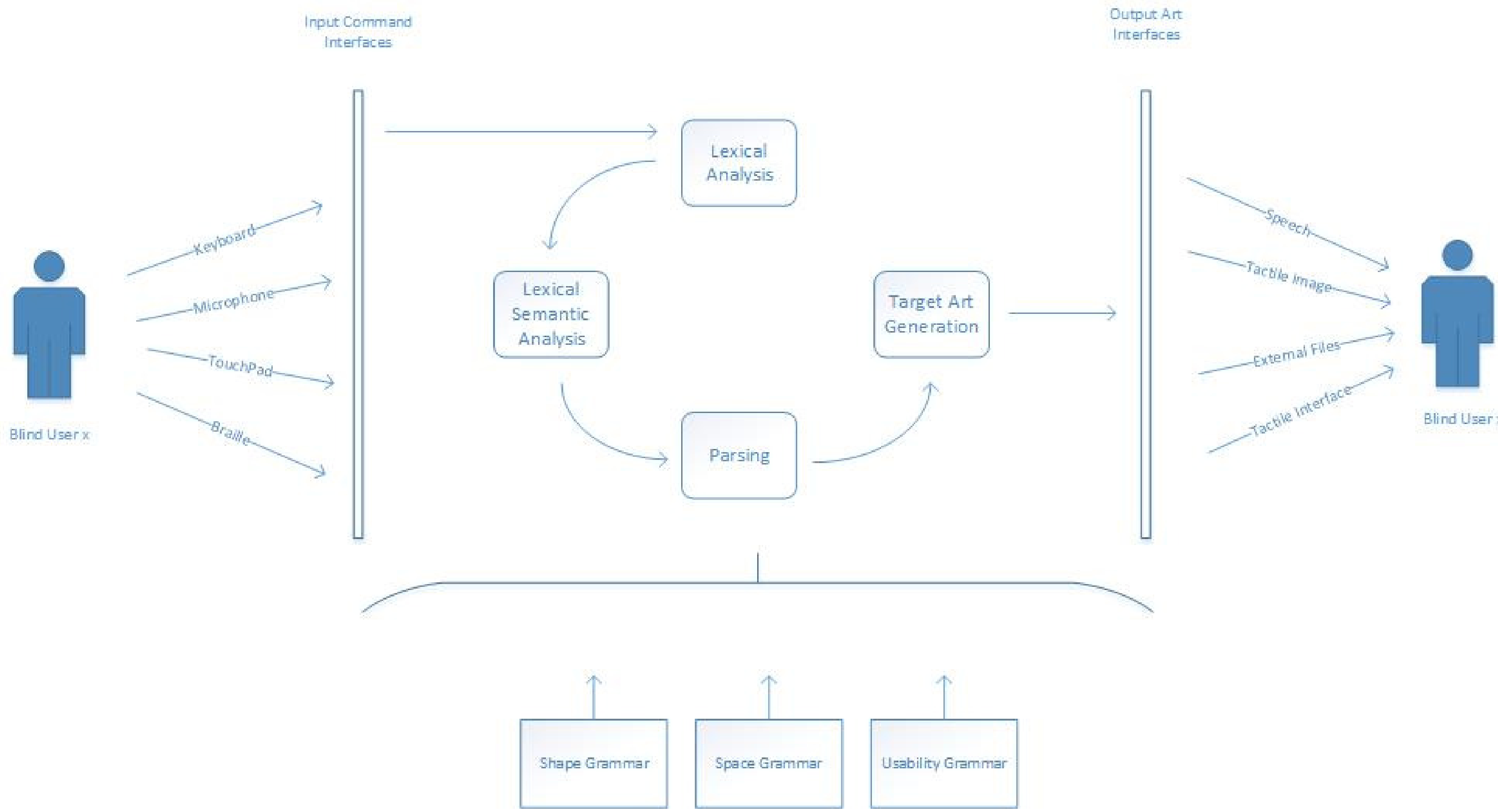


Figure 5.21: System Architecture

5.3.1 A Demonstration of SETUP09

Follow the link for a short (5.38 m:ss) demonstration of SETUP09 system:

www.youtube.com/watch?v=vVgliLBBw58

5.3.2 System Architecture

Figure 5.21 shows different input modalities and output modalities with front-end parser stages, namely: lexical analysis, lexical semantic analysis, parsing and target code generation. The possible input modalities are keyboard, microphone, touchpad and Braille, whereas some output modalities are speech, tactile images, external files and tactile interface. It also visualises possible grammar syntax in three components: space, shape and usability, as discussed in the previous chapter.

Interface Class: Input handling

The interface class of the third prototype is operated by keyboard keys. Once the text is read, the input is sent through the *tokenizer* and *parser* methods in the token class to identify the syntax and semantic accuracy. It then calls the graphics on to the panel by calling a new system state. The new system state has properties of the system, such as cursor coordinates x and y, current zoom size, level, image buffer object, and other data collection objects to store changing variables. An extended interface class is included in the Appendix.

Input Output Modalities

The only input modality evaluated in this thesis is the input mechanism i.e. the keyboard. Figure 5.22 shows a picture of a participant using a Mac computer keyboard to operate the proposed SETUP09 system. Other modalities were not tested due to the limitation of the scope and time of this thesis. Both Windows and Mac computer keyboards were used during the evaluation and development stages. The system was evaluated with many participants who were competent with keyboard use.

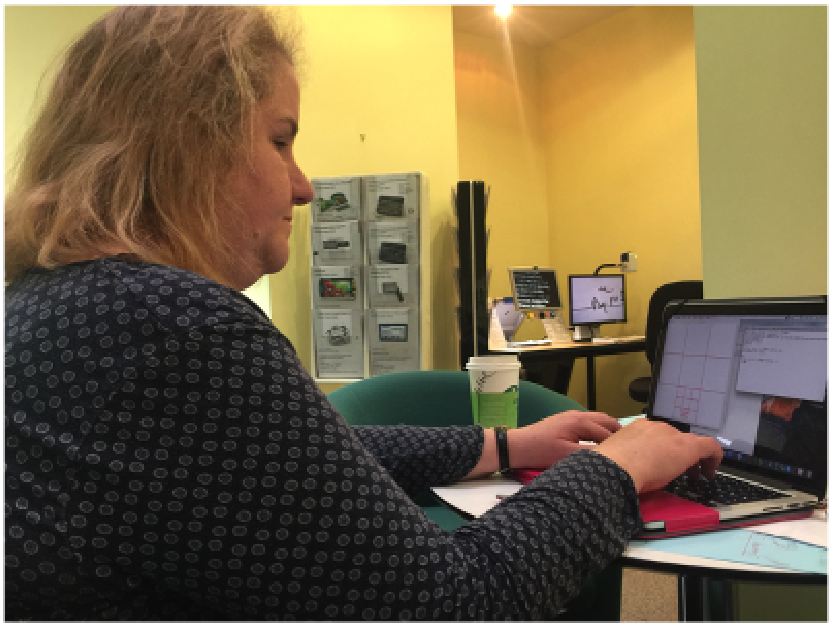


Figure 5.22: keyboard Input Modality

One of the input modalities proposed in the proposed architecture was the use of speech recognition software which has recently gained momentum with the introduction of Speech Recognition Grammar Specification (SRGS) by [W3C, 2019] and technologies such as Sphinx, [1]. The SRGS context-free grammar is introduced both in XML and BNF format which have expressive power. Grammar defines what humans can or should say. It could be a collection of words, phrases or sentence patterns that can be expected from a blind person. Java has an interface to support such speech recognition grammar rules: the interface is *javax.speech.recognition.RuleGrammar*, [Sun, 1998]. The rules can be defined in a data structure or Java Speech Grammar Format (JSGF) and managed via a recogniser interface. The rules can be defined in a data structure or Java Speech Grammar Format (JSGF) and managed via a recogniser interface. The opportunities offered by speech recognition grammar are out of the scope of this thesis, however system SETUP09 can be integrated with such technology as it is built on grammar-based input recognition.

Cloud Speech-to-Text API Client Library for Java, [Google, 2019] is another library that can be used for speech recognition for SETUP09, and works with Google speech recognition technologies in the SETUP09 application. The text transcription of the audio file is returned when speech audio file is sent to the 'speech-to-text' API. A spoken lecture processing system was introduced for online visual recording of academic lectures, [Glass et al., 2007]. It experimented with speech processing, transcription, indexing, segmenting and summarising with pre-fed vocabulary-based recognition into spoken documents.

Another input modality mentioned in the architecture Braille input to computer. This can happen via two different methods. In the first method the Braille is typed directly into the computer using through Humanware products such as Brailliant BI 40 which is a Braille notetaker where a Braille

[1]CMUSphinx is an open source speech recognition system. https://cmusphinx.github.io/

display is connected with a special cable to a computer in order to send product signals as input and output, [Humanware, 2018]. The second method to input text into the computer is using a computer keypad. Translation software is then used to translate ASCII keys into Braille. Tactile overlays of Braille symbols and other important keys can facilitate tactile typing, enabling easy recognition of keys and guiding blind users to their location.

Recently introduced TouchPad Pro can be used for input commands. Designed for BVI users, it is portable and has a special electronic stylus and a large tactile and visual display. It also has the capability of read-write and many more features, [TouchPad Pro, 2019].

The SETUP09 system was evaluated using tactile images, [Braillo, 2019] and speech synthesiser outputs, [Microsystems, 2005]. Tactile images were printed using a Swell Form Graphics Machine (fuser) using swell touch paper as shown in Figure 5.23. Once the image is printed on swell paper using a standard printer, it is then run through a Swell Form Graphics Machine. The black ink responds to the heater and the inked areas puff up creating a tactile image.

Figure 5.23: Swell Form Graphics Machine, [Braillo, 2019]

The SETUP09 build produces images that can also be saved in a picture format for later retrieval or import to other graphics software packages. The format in this prototype is kept as JPG but is not limited to that format. The save commands can also write the user-typed commands to a text file to be loaded later.

5.4 What is Next?

In this chapter the proposed SETUP09 model was implemented in three stages, i.e. a lexical stage, semantic analysis stage, and finally the parser stage. The model accuracy was validated using an abstract syntax tree structure to generate artwork, with output art examples. This was then followed the implementation of the system's front-end. The implementation of SETUP09 leads to the need to test the system model. The next chapter opens with establishing the groundwork for system evaluation. It addresses the question : Does a SETUP09-based service meet the needs of BVI people? Does it improve the quality of art creation by them? This can be evaluated by studies carried out using the SETUP09 system.

Chapter 6

Experimental Evaluation of SETUP09 Model Use

The previous chapter detailed the implementation of SETUP09 model, with a grammar-controlled language and user-interface implementation whereas this chapter consists of two field studies of experimental evaluation of SETUP09 model that is used as a method of validation. This chapter hypothesises, rationalises, and brings findings of the user validation of SETUP09 model to the questions: Does a SETUP09 based service meet the needs of BVI people? Specifically, Can different subsets of blind users, make high quality outputs accurately and efficiently? It was therefore necessary to test the model implementation with different subsets of users. Study one tests for the efficiency and effectiveness of SETUP09 navigation language whereas study two tests for the efficiency and effectiveness of graphics creation techniques with multiple subsets of users. Both studies are presented with hypotheses, rational, and evaluation matrix. This is followed by a study method, procedure, finding, and conclusion. The chapter elaborates on the limitations that may impact with the generalisation of studies and assumptions made. From study one, it was found that participants managed to correctly use navigation tool and comprehend the screen location of SETUP09 far better than any tactile grid paper format. Also, no difference in time was recorded in using the two methods. Finding of study two revealed that the users successfully completed their tasks with considerably less time than recorded similar systems. Collected from user observation and user feedback during the studies, the chapter concludes by presenting changes required to the SETUP09 model and its front-end implementation. This led to the need of grammar refinement, which form the bases of Chapter 7.

6.1 The Design Rationale of Methodologies

SETUP09 prototype required the assessment for two main cognitions: (i) to reach a screen location and (ii) to be able to produce a drawing. Additionally, the prototype was tested to assess usability: specifically ease of use, efficiency, supportiveness. Established drawing systems such as *IC2D*

[Kamel and Landay, 2002], BPLOT [Fujiyoshi et al., 2014], AHEAD [Rassmus-Grohn et al., 2007], and KEVIN [Blenkhorn and Evans, 1998] were tested with blind participants, visually impaired participants and also with sighted participants. The usability and accessibility assessment was made through observation and feedback during and after the testing sessions. Blind participants were asked to create a pre-produced image or actual 3D shape during the IC2D product test by [Kamel and Landay, 2002], whereas during the test stage of AHEAD [Rassmus-Grohn et al., 2007], participants were required to produce images prepared by an instructor, and during the product test of TDraw introduced by [Kurze, 1996], participants were required to produce art based on the participants comprehension of the world with out any instruction. The system KEVIN introduced by [Blenkhorn and Evans, 1998] used formal and informal methods to evaluate the concept using initial feedback on the tool developed; the formal evaluation of KEVIN was against standard swell paper tactile diagrams. It was suggested that tactile maps are found to have a good frame of reference to the external world according to the studies conducted by Ungar in cognitive mapping without visual experience[Ungar, 2000].

The SETUP09 prototype evaluation studies were designed with special care, in the knowledge that there is no equivalent established digital system to compare against. Different tasks were carried out to find cognition of navigation, cognition of drawing and image recognition using swell papers and analogue drawing kits as explained in Chapter 3. As shown in Figure 6.1 three types of basic testing were carried out to evaluate the proposed system. Firstly the navigation of the SETUP09 system was by checking if a user could navigate to an intended location, secondly the drawing of the SETUP09 system was tested by designing instructed tasks, non-instructed tasks, and image recognition tasks. Analysis of the results and the claims made were done for different groups: blind, partially sighted, and sighted people. The results were displayed for five blind, five partially sighted and ten sighted participants in the bar charts presented in Study 2. The results presented compare the blind and partially sighted groups, and all visually impaired participants with all sighted participants, in order to highlight the difference in performance between people with different types of vision. Moreover, a comparison of results from early blind and late blind individuals is discussed to evaluate the proposed system.

6.2 Evaluation Metrics

Bevan [Bevan et al., 2016] discusses [ISO, 2011] standards of usability and measures that are based on the Ergonomics Subcommittee of ISO 9241 and ISO 25000 Software Engineering Committee. ISO 9241-11 explains the measures for usability under three headings: effectiveness, efficiency, and satisfaction. So does [Cook and Polgar, 2007]. Effectiveness is measured by errors in the tasks and task completion. Efficiency is measured by task time for user actions. Satisfaction is measured by users' overall satisfaction, user trust, and feedback. The prototype SETUP09 is compared with three measures of ISO standards, namely effectiveness, efficiency and satisfaction with different sub-groups (blind, partially blind, late blind and sighted counter parts), comparing analogue with digital methods of diagrammatic drawing. The *efficiency, accuracy, and user satisfaction* are taken as evaluation metrics in study one and study two.

- **Efficiency** is calculated by the time taken to complete an activity using the prototype SETUP09 system.
- **Effectiveness/Accuracy** is calculated on the number of correct attempts during a task compared with incorrect attempts.
- **User satisfaction** is measured with a set of post-experiment questionnaires. These questions were designed to access how real users found the accuracy, ease of use and cognition of the concept.

Usability and Accessibility Survey

John [John, 1996] introduced a System Usability Scale, a questionnaire designed to assess users' satisfaction with ten questions on a five point Likert scale. Kirakowski [Kirakowski and Corbett, 2006] introduced SUMI (Software Usability Measurement Inventory) composed of 50 questions that was commercially popular. Lewis [Lewis, 1993] from IBM also introduced different types of scenario questionnaires, namely ASQ, PSQ, PSSUQ, and CSUQ that use situation and scenario based questions. However, ASQ (After-Scenario Questionnaire) questions are better suited for the cohort in this research study as the feedback is captured after the system is used. The SETUP09 prototype was evaluated for satisfaction against the concept of usability measures, and this stands as another reason to select the After-Scenario Questionnaire. The SETUP09 questionnaire was designed to include five items based on the hypothesis tested: (1) SETUP09 builds a layout model in the participant's mind; (2) SETUP09 builds a navigation model in the participant's mind; (3) SETUP09 is easy to use; (4) SETUP09 is efficient; and (5) SETUP09 is supportive. These questions were designed using a Likert scale to obtain participants' feedback.

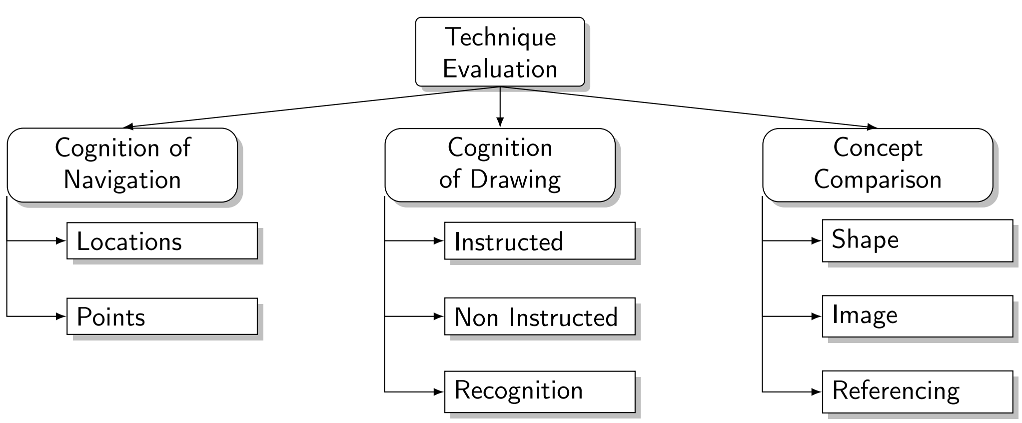


Figure 6.1: Test Design Flow to Measure Cognition of Navigation, Drawing and Comparison of Techniques.

Figure 6.1 shows the different SETUP09 tests conducted with BVI and sighted users. The SETUP09 technique was evaluated to find out the cognition of navigation and cognition of drawing, and the concept of SETUP09 was compared with other systems in Chapter 8.

6.3 Study 1: Screen Navigation Technique with Blind People

Blind computer users were tested using different screen navigation tasks to assess the accuracy, efficiency and satisfaction of the compass-based navigation technique by using the proposed SETUP09 prototype and tactile paper grid maps. Study 1 tested to find out participants' ability to navigate and the ability to find out specific locations on the screen as discussed below.

6.3.1 Hypothesis

- **Hypothesis (H1):** Going through a virtual environment generates a navigation route in a blind person's mind.

  Rationale: From literature findings it is established that only a very few blind drawing tools experimented with in the past paid attention to screen layout without necessarily needing haptic modality (IC2D [Kamel and Landay, 2000], BPLOT [Fujiyoshi et al., 2008]). While SETUP09 is compared with IC2D in Chapter 8, this Study 1 - H1 is designed with the need to know if a blind person can visualise a screen location with the compass-based technique, using command driven navigation routes of SETUP09 before any art is produced. The digital screen was reproduced on swell paper and participants were asked to show the target screen area of the given tasks using both the swell paper and the SETUP09 system to cross-check their understanding.

- **Hypothesis (H2):** SETUP09 navigation method is an efficient method to find a specific location on the screen compared to tactile maps.

  Rationale: From literature findings it is found that existing mainstream establishments use embossed drawing as the main technique to construct drawings. This is achieved using swell papers, film papers or special toolkits [2] that provide good frames of reference for blind people as explained by [Ungar, 2000] and [Takagi, 2009]. Hence embossed printed papers was used here to compare SETUP09 screen grids.

6.3.2 Method

The tasks sought to identify the accuracy and efficiency of the screen navigation technique using a compass-based navigation system (SETUP09) involved giving participants six navigation tasks. In fact, participants were given tactile swell papers with grids that presented the SETUP09 screen layout. They used tactile swell Papers to recognise a specific location given by the researcher.

[2] Different tactile production methods: https://www.perkinselearning.org/scout/blog/tactile-graphics-students-who-are-blind-or-visually-impaired, $https://en.wikipedia.org/wiki/Tactile_graphic$

Participants were asked to identify different raised areas on the tactile paper. They were then asked to input navigation commands using the SETUP09 system to bring the focus point on the SETUP09 screen to the required location on the tactile paper. This was done to check if they could correctly translate their tactile understanding onto a non-tactile command driven digital interface. Participants were also asked to navigate directly to a given screen location on the system without any tactile papers. The purpose of the activity was to find out if participants were able to navigate to a specific location on the computer screen simply by visualising the computer screen layout without tactile paper guidance. Their understanding on the target locations was then checked on by asking them to show the corresponding tactile area on the swell paper.

Participants were then asked to type the commands to go back to the starting point. The purpose of the activity was to find out if participants were able to navigate to a specific location on the computer screen that was given to them and trace their way back purely by visualising the navigation route.

Some target locations were smaller than others, with activities designed to target different zoom levels (2, 3). Every person had five attempts at Study 1 and the task completion time and accuracy were recorded. The system commands used for Study 1 are listed below.

1. zoomin <location >
   - Locations can be North, South, East, West, North-east, North-west, South-east, South-West, or Centre. The command is used to navigate to a specific and smaller screen location.
2. zoomout
   - This command is used to come out from a specific area of the screen.
3. position
   - This is a helper command to trace the location of the screen in its current position.

Participants were also asked to find the centre of the grid layout by drawing a circle shape. For this experiment, swell paper was used with a grid layout. Participants were asked to draw a circle on the centre of the given resource and the time taken and accuracy of the drawing were recorded. The purpose of the activity was to assess the efficiency of different techniques to find a specific location. An easy target was given for the task to limit possible confusions. Each participant had one attempt at this task.

At the start, the cursor or the point of focus sits at the top left corner position of the screen layout and when a new command is executed the cursor moves accordingly. To draw a line or write a label, the user can call a specific point on the screen. This then sets its new cursor position to the left most position of the screen layout area. In the case of incorrect entry, a user can abort input commands. Figure 6.2 shows the layout of the SETUP09 screen, showing the focus area of the command "zoomin N"

- Zoomin <N >

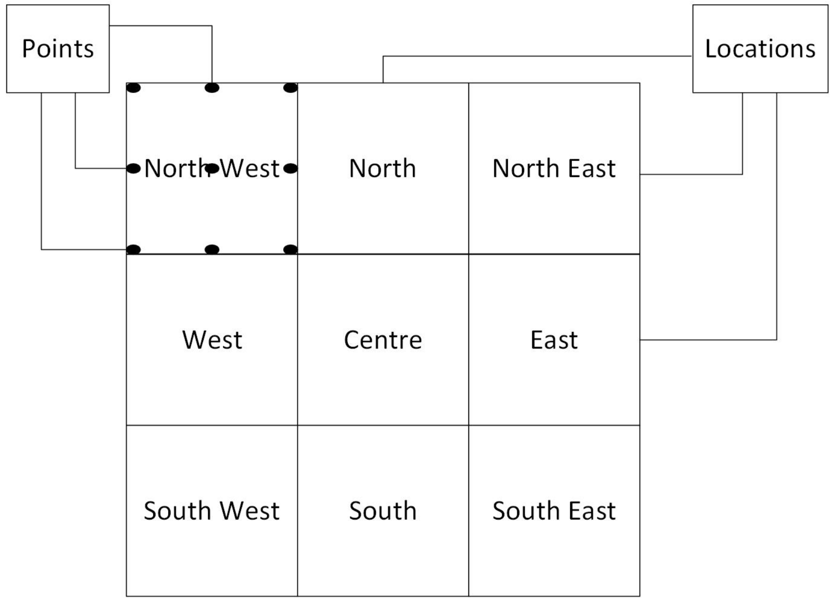


Figure 6.2: System SETUP09: Screen Layout

Figure 6.3 shows four interface changes that happen as a result of cursor navigation to East-Centre-South of the screen. The first screen is the initial screen with no focus on any particular grid but the top left corner of the screen. The second interface displays further grids in the East area. The third interface displays further grids in the East-Centre area. The fourth interface displays further grids in the East-Centre-South area. For example, if the intended navigation location is the South Centre of the East location, the navigation path is East to Centre to South, as illustrated in Figure 6.3.

6.3.3 Procedure

Experiments were conducted on a group of BVI people to find out the efficacy of compass-based navigation solutions used by individuals who are totally and severely blind. A group of five participants of different age groups, different levels of blindness and different levels of computer literacy were used. The age of the participants ranged from 27 to 52 years old. Computer literacy ranged from moderate to high. Three participants were totally blind and two severely visually impaired where their sight is limited to lines and shapes. The SETUP09 compass-based navigation system is proposed as a solution to navigate in a virtual environment. The computer uses a speech-based feedback mechanism. Given the low number of participants available, it was understood that the most productive method to collect data was by one-to-one training, experiment observation and recording.

To select a target location, a user must enter/input the zoomin command in the area which the user intended to travel. The size of the target is influenced by the zoom level as it divides the screen into nine sections. Each time the zoom is used, the zoom level increases by one. This study only discusses the navigation efficacy of compass-based navigation, whereas in future studies the efficacy of the drawing commands will be investigated. There are five target locations in Study 1. Each time

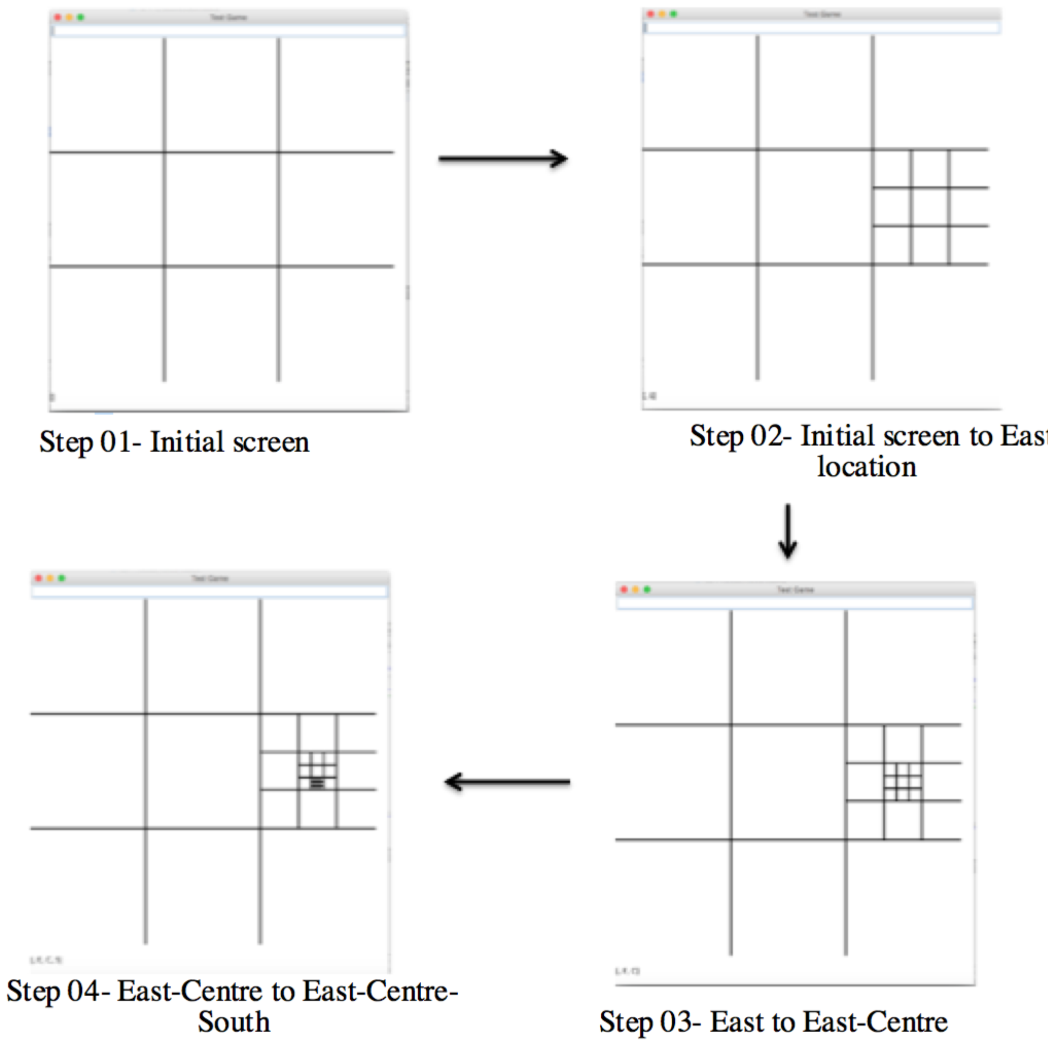


Figure 6.3: System Navigation Path to East-Center-South

the screen starts with the default start position, which is with a grid of nine memorable locations, e.g. North, South, East, West, Centre, North-east, North-west, South-east and South-west. Screen location size is determined by the “zoomin ” command that takes the focus to different granularities on the screen.

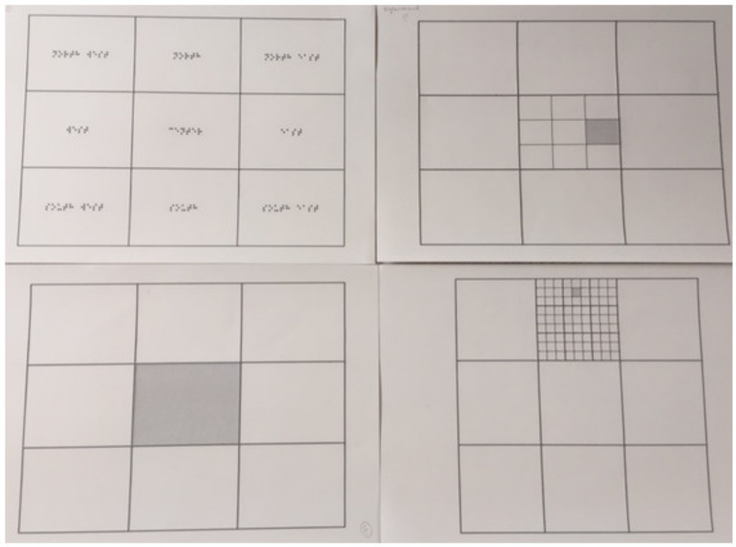


Figure 6.4: Swell Papers Used for Tactile Recognition

A MacBook Pro (Retina, 13-inch) with Mac and Windows operating systems, Intel Iris Graphics 6100 1536 MB and FreeTTS (speech synthesis system) was used for speech interaction that was implemented in the SETUP09 system. Before completing the task, participants were given a user manual and a training session. During the training period, participants used the available commands to test different screen target locations of their choice. No time limit was set for the training session and they could repeat it if desired. Assistance was available during training. After the training six different tasks were completed. Each participant spent around 30 minutes for the training and for the experiment. Training materials for this study are given in Appendix E3. After the experiment they completed a questionnaire and gave feedback on the experiment.

Figures 6.4 shows swell papers with grids that were used during the experiment, Figures 6.5 and 6.6 contain two images of participants during the experiment. The participant in Figure 6.5 recognises the highlighted areas on the swell/tactile paper, and the participant in Figure 6.6 demonstrates keyboard input operation using the SETUP09 system.

The first study investigated whether going through a virtual environment generates a navigation route in a blind person’s mind. As mentioned in the literature review, screen navigation does not always lead to accurate focus location of where BVI users intend to travel. The user must know, and remember, the various commands in order to select the required commands. The system must then recognise the words typed. Prolonged processing delay in the compass-based navigation can cause errors or incorrect cognition of the screen by the user. As a solution, the system could provide the

Figure 6.5: Participant Using Swell Paper

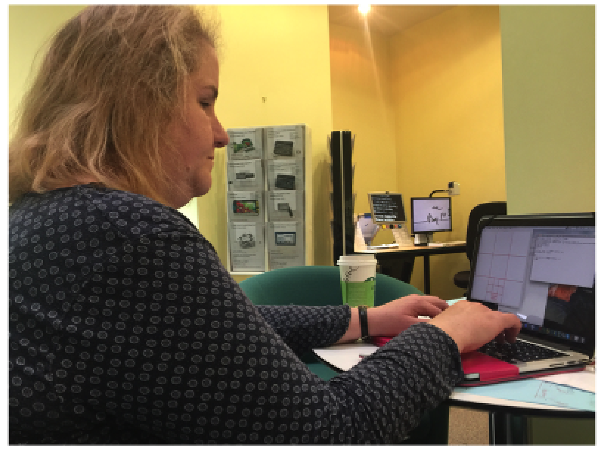


Figure 6.6: Participant Using SETUP09 System.

user some assistance/feedback to prevent such errors. More specifically, this study investigated the potential benefits of compass-based navigation, as well as investigating the accuracy of compass-based navigation compared with a tactile map, and efficiency based on completion time. The list of activities below was used in the experiment to record the findings of Study 1.

- **Activity-1:** Feel tactile image given to you. Type the commands using the system to get to the raised area.
- **Activity-2:** Type the commands to go back to the starting point.
- **Activity-3:** Type commands: zoomin N, zoomin C. Show the focus area using the tactile paper given to you.
- **Activity-4:** Type commands: zoomin N, zoomin N, zoomin C . Show the focus area using tactile paper given to you.
- **Activity-5:** Feel the tactile paper image given to you. Type the commands to get to the highlighted area.
- **Activity-6:** Find the middle of the screen/ swell paper to draw a circle shape given to you.
- **Activity-7:** Now use the tactile grid and find the middle of the tactile paper. Mark the area with a pen/pencil.
- **Activity-8:** Now input the commands to the SETUP09 system to navigate to that area.

The observer noted the difficulty of the activity at each stage, the method and the time taken for each method and asked the following questions:

- How was the actual experiment? Explain your answer.
- What were the difficulties you found? How did you overcome them?
- In what ways can the system be improved?

6.3.4 Results

The results are produced based on completion time of the experiment, errors made and accuracy.

Time Taken for Activity 1-5

All participants were able to complete the first five activities with an average time of 33 seconds. Some participants were quick to learn and other participants took time to find the location on the swell paper. All of them found grids on swell paper to be harder to use than the SETUP09 system as they had to navigate by remembering what they had done and counting grid squares on the swell paper. Activities 4 and 5 were more time consuming, especially where the zoom level extends beyond two levels. Severely BVI participants were able to identity the raised-line printing much quicker than totally blind participants. They all enjoyed the activities and were keen to get them right. Blind participants spent no longer than 1.13 minutes with the standard deviation S = 22 seconds.

Participants on average were able to reach target locations on the SETUP09 system much faster than when using tactile maps. Participants spent an average time of 33 seconds on the SETUP09 system as opposed to 42 seconds on tactile paper related to activities 2 and 3. It is evident from these results that a system-generated cognitive map enabled the users to find a location much faster than a tactile map. One participant made a mistake typing the command and hence took a longer time in an activity on the SETUP09 system, and another participant made a mistake looking for a location on the tactile paper with the wrong orientation, and hence took more time using the tactile paper. Overall, participants found it easier to reach a location with the SETUP09 system than with grids on tactile papers.

Challenges: Users found that feeling the tactile images was difficult to comprehend especially where there were more than two zoom levels. Some participants had to make several attempts due to the need to count smaller areas on swell paper in order to recognise the exact map location, or to map the given commands on swell papers. But they all seemed to be comfortable with the SETUP09 location finder. They all mentioned that the navigation route can be visualised accurately; however, relating it onto the grids on tactile paper was difficult.

Accuracy: H1

Table 6.1 shows that 72% of attempts made in activities 1-5 using the compass-based navigation were successful, and participants were able to show the exact same location on a tactile swell paper, while 28% of attempts were unsuccessful at the first effort. However, the users managed to self-correct by

Participants	Number of Correct recognitions	Number of incorrect recognitions
Participant 1	4	1
Participant 2	3	2
Participant 3	3	2
Participant 4	4	1
Participant 5	4	1

Table 6.1: Number of Correct and Incorrect Recognitions Made During Activity 1-5

probing the system, using commands such as “My position” to complete the activity.

In total seven errors were made out of 25 input commands. Each participant had five input commands and there were five participants. On average they made at least one error and at most two errors. Some of them were confused with raised-line grids as some activities were designed to identify the target location highlighted on swell paper. The confusion was mainly due to closely printed grids on the tactile paper, and some participants found it difficult to identify exact locations on anything requiring zoom level 2. However, they were all confident about their mind maps and trusted the SETUP09 system to navigate to the correct destination. The errors were mainly made during activities 4 and 5 when recognising the exact location highlighted on swell papers that was located on zoom level 3.

Incorrect Understanding of Activity 1-5: H1

The total errors made are illustrated using a bar chart in Figure 6.7. The seven errors made were mostly on zoom level 3 related activities. This was due to the fact that participants had to count and remember the grids on the tactile paper, which generated a high degree of mental work and some confusion.

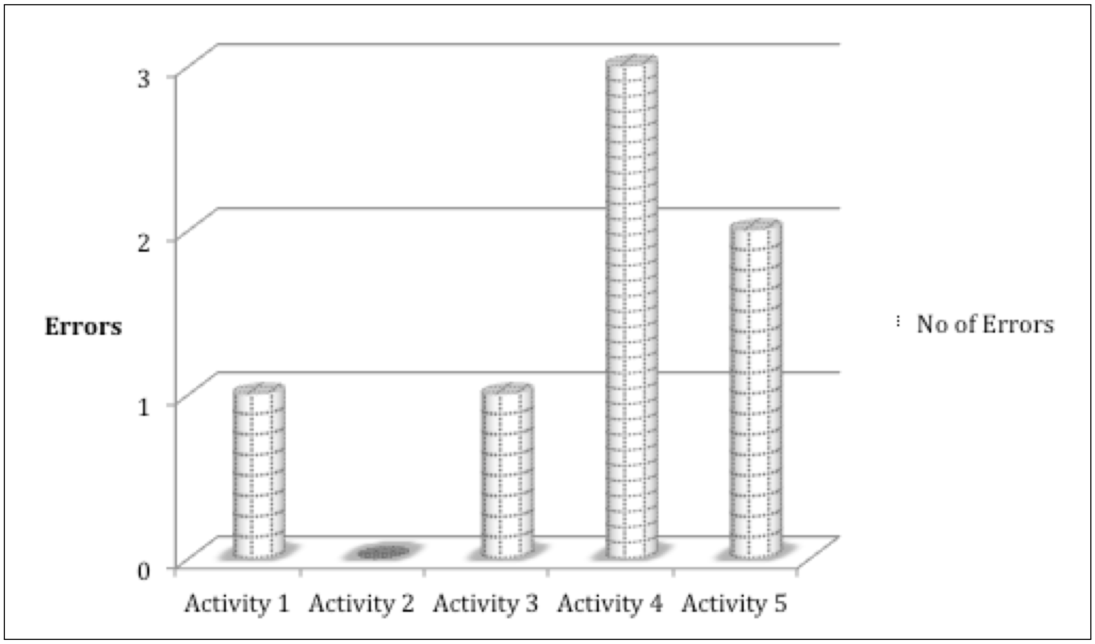


Figure 6.7: Number of Errors During Activity 1 - 5

Accuracy of the mind Map of Different Zoom Levels.: H1

Figure 6.8 shows the nature of errors made during activities 1 to 5. Participants made two errors on zoom level 2 activities and five errors on level 3 activities. More errors were made on activities related to zoom level 3 compared with zoom level 2.

Longer time was taken to complete the activities where errors were made when completing them. More errors were made on activities, 4 and 5, which had three, zoom levels to recognise screen areas. Errors are labeled E1 to E7.

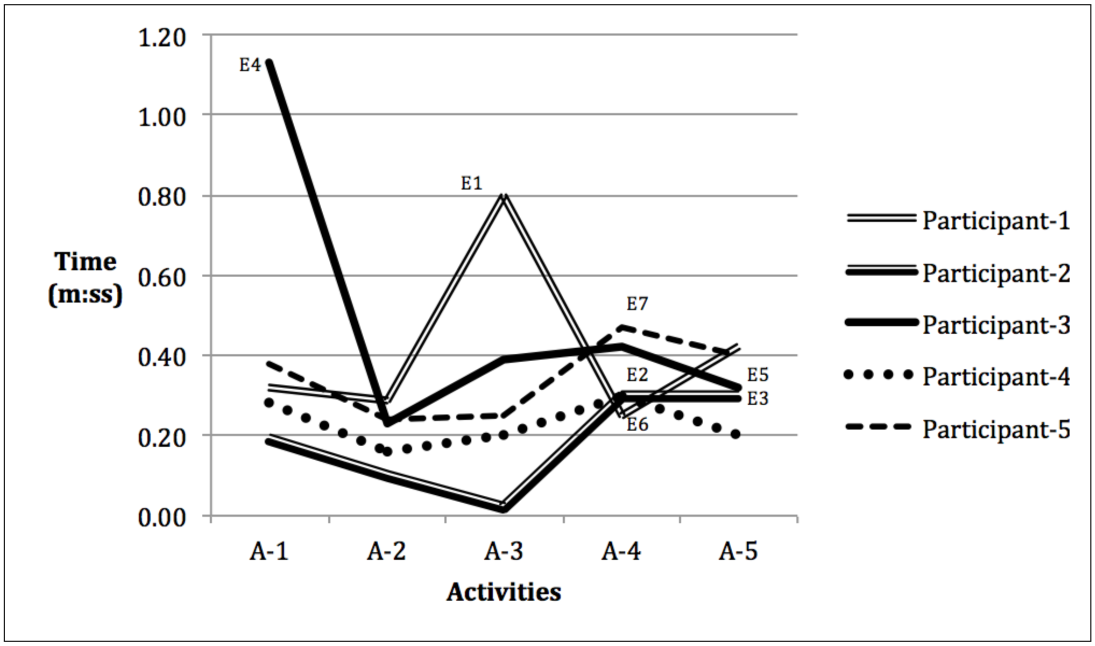


Figure 6.8: Time and Errors During Activity 1 - 5

Efficiency: H2

Time taken for Activity 6 : Finding the Centre Area of a Swell Paper and SETUP09 System.

Finding the centre region on swell papers and SETUP09 was relatively easy. The participants were all able to recognise the centre area on the swell paper and SETUP09 system fairly quickly. The swell paper used here was designed to the first zoom level with nine areas. The chart in 6.9 indicates that all participants managed to find the centre area on the SETUP09 system and the swell paper method with ease and they were quite confident of their accuracy as discussed in the post-study survey in section 6.3.5. On average, finding the centre area on both the swell paper and the SETUP09 system took the participants 5 seconds.

Activity 6 - Comparison: H2

Finding the centre area of a swell paper and on the SETUP09 system. This was quantified numerically, where entry 1 is the easiest and 5 represents the most difficult. Participants found it easier to use the SETUP09 system than swell papers to find the location C (Centre). Participants were highly satisfied and more confident with the accuracy of the SETUP09 system than when using the swell paper, according to their feedback. The level of difficulty is presented using a line graph in Figure 6.10. Participants were also better satisfied with SETUP09 than swell paper during the post-survey questioning shown in the next section.

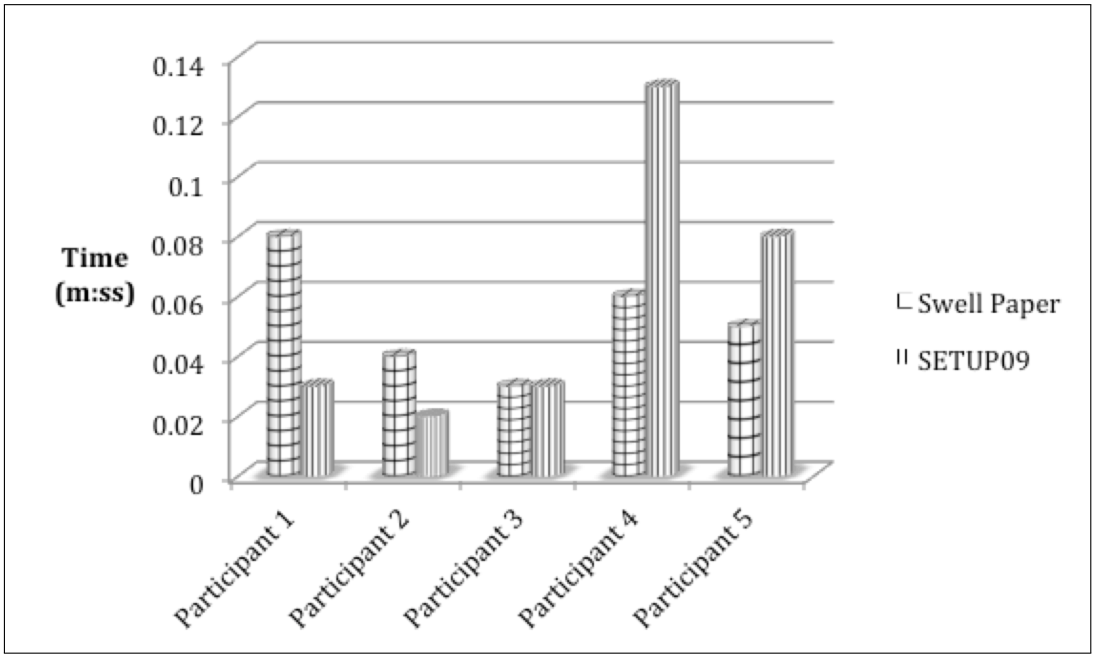


Figure 6.9: Activity 6 -Time Taken to Finding the Centre.

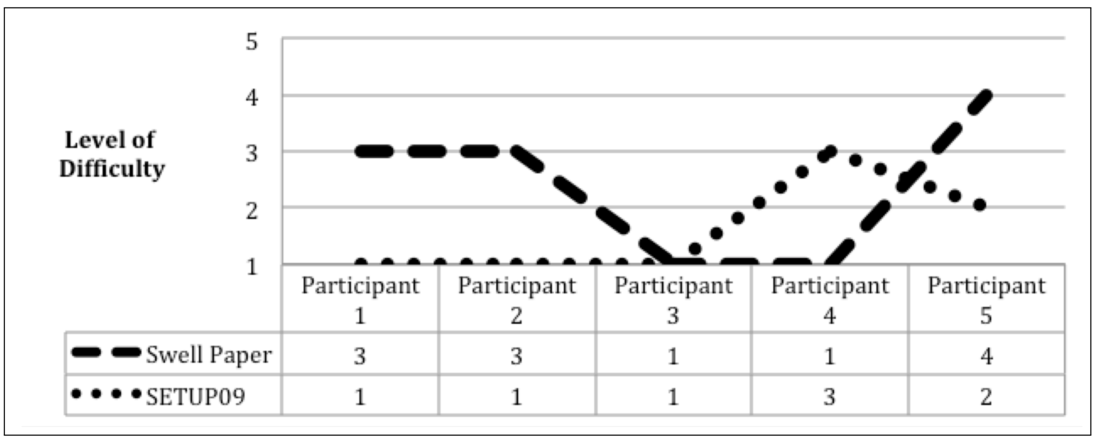


Figure 6.10: User Satisfaction - Where 1 is the Easiest and Most Satisfying Method, & 5 the Most Difficult.

6.3.5 Other Observations: Memorability and Guidance

This first study was based on using three commands to test navigation: zoomin <location> zoomout, position. Out of 25 attempts made by the participants, seven attempts were incorrect. The incorrect attempts were not due to the difficulty of memorising the commands, but as a result of the confusion the participants found with the raised line grids and visualising locations of higher zoom levels. However, it was observed that four attempts were incorrectly typed using capitalisation and one attempt was typed with incorrect spelling. Consequently participants had to input those commands twice. Although they did not find any difficulty with memorising the system commands, but they did have a typing error ratio of 20%. See below for the overall error rate of the first experiment:

$$\text{Error ratio} = (\text{errors/number of attempts}) \text{ x100}$$

$$(7/25) \text{ x } 100 = 28\%$$

It was also noted that participants sought guidance on ten occasions as recorded in 6.2, especially with navigation to higher zoom levels. System guidance asked was mainly on using "position".

Activity	P1	P2	P3	P4	P5
1	0	0	0	0	0
2	0	0	0	1	0
3	0	0	1	0	0
4	0	1	1	1	1
5	1	0	1	1	1

Table 6.2: Use of System Support Commands by Participants

The guidance ratio was calculated thus,

$$\text{Guidance ratio} = (\text{number of times system support was used/number of attempts}) \text{ x100}$$

$$(10/25) \text{ x } 100 = 40\%$$

The guidance ratio does not indicate any fault with the system design but indicates the autonomy of movement with available system usability functions.

Post Study Survey

Presented in Table 6.3 are participants comments and feedback analysis.

User Feedback	Participants' Feedback Analysis
"SETUP09 is a clever method, I do know where I am on the screen every time I navigate."	4/5 participants strongly agreed that the SETUP09 technique enables then to understand the target grid location /cell. This indicates that the SETUP09 technique is highly effective among participants.
" My position" helped me to memorise the navigation better"	Some participants commented on having the confident to trust the computer without feeling the need to confirm the commands. They suggested touch pads, voice input and hotkeys as input mechanisms.
" The prototype needs practising several times before you really learn to use it ".	Different user levels could be an option at the start-up screen with different levels of system feedback, where more feedback is provided to new users and less system feedback for advanced users.

Table 6.3: Participants Feedback

The feedback made by participants in Table 6.3 were used to improve SETUP09 system. Some participant commented on being confident to trust the computer without feeling the need to confirm the commands. Touch pads, voice input and hotkeys were suggested as input mechanisms. Some participants needed more time to get used to the system and commands, therefore different user levels could be an option at the start-up screen. Many of the participants commented on the difficulty of recognising the swell paper grids, especially beyond the second zoom level. They found it easier when the swell paper lines were thicker, with fewer grid cells , and presented with highlight variation and having enough spacing. It is therefore important that the haptic (2D) images be carefully planned before printing the constructed images on the SETUP09 system.

Participant's also agreed and commented on the statements given in Table 6.4 (1) "Compass-based navigation (SETUP09) system helped me to understand the target location", and (2) "I needed to be certain about my target location by touch or feel of the tactile map". The participants said that by asking themselves the question where is my position when navigating the system helped them to memorise the navigation better. Another common commented was (3) "I need to practice on the prototype several times before I can really learn to use the system properly".

Statements	P1	P2	P3	P4	P5
SETUP09 compass-based navigation helped me to understand the target location better than swell papers.	1	1	1	2	1
The prototype needs practising several times before I use it with confidence.	1	3	1	2	2
I think my cognitive map is correctly built when using the SETUP09 system .	1	1	1	1	1

Table 6.4: Participants Comments: 1 - Strongly Agree to 5 -Disagree.

The next section details the outcome of study 1, answering the following questions. What has been learned from testing these systems? What needs to be changed as a result? What worked well, and what did not?

6.3.6 Findings of Study 1

- Participants identified target locations on the tactile maps incorrectly when it went beyond three zoom levels. They used a strategy of counting grids when working with tactile grid papers, whereas in SETUP09 participants visualised the screen area without needing to count. Participants also used perimeter strategy, whereby they explored the boundaries of an area to identify the area's shape and size, and grid strategy to investigate the internal elements of the screen area to learn spatial relationships by taking straight-line finger paths from one side of the tactile grid layout to the other as experimented and discussed by [Ungar, 2000] and [Lahav and Mioduser, 2008]. Their attempts to use the virtual map for smaller points of identification and determining area sizes were difficult. They were able to tell the number of zoom levels from memory and were able to find the route back to the root within three zoom levels.

- Comparing the results between the SETUP09 prototype and tactile paper grid maps, it was observed that blind participants showed high levels of accuracy and efficiency using SETUP09 to reach a target location compared to the tactile grid map. Participants found it difficult to find smaller locations on the tactile paper even when presented with a raised-line grid system. This was because of the small size of the grid cells. Work by [Afonso et al., 2010] also explains that congenitally blind participants generate accurate spatial mental images with audio virtual reality technology also verifies the ability to build structural properties of spatial mental images via audio clips.

- During the execution of the first set of commands when finding the target locations, the participant were successful in getting to the location with an error rate of 28% and in a short amount of time (33s), without the help of a support worker.

- There is no significant difference in the completion time between finding a location using the SETUP09 navigation system as opposed to using raised-line tactile maps. However SETUP09 is recorded as being more efficient than swell papers because it has a slightly shorter completion time, less mental load and better accuracy, based on the participants' feedback. The feedback is captured in Table 6.4.

- There is a significant difference $(t = 0.0001, p - val = 0.0002 < 0.05 = \alpha)$ between correctly identifying a location using SETUP09 and incorrectly identifying a location using SETUP09. This demonstrates that for 72% of the time the navigation map in participants' minds was correct. The main reason for the difference in the time (incorrect recognition of target locations - Activities 4 and 5) is due to the complexity of using higher zoom levels. Another reason for the difference in the time is due to the unfamiliarity of the SETUP09 system which can eventually be rectified with greater us of the system.

- Many of the participants were interested in the nature of navigation movement. For example, whether specific commands would move to the target location as desired. Participants were also able to construct navigation routes when several zoomin commands took place at the same command line execution. This demonstrated participants' capacity to remember and visualise several navigation commands and zoom levels.

- All participants agreed that they enjoyed the simple command vocabulary to navigate the SETUP09 system and that they would like it as a screen navigation technique.

- The observation shows that participants were faster and more accurate when finding smaller target areas with SETUP09 than tactile grids. Although experiment results will need to be extended to a larger group of participants, however the initial results for the SETUP09 system are very positive as it enables BVI users to move from one location to another location with confidence, without the help of a support worker.

6.3.7 Limitation

Qualification of digital system and tactile paper comparison: It may be questioned why in the study the tactile map is compared with command driven digital system. The study does not equate the ability to locate a point using navigation commands with the ability to locate an equivalent point on a two-dimensional tactile sheet of paper. One requires intensive use of the tactile sense, whereas the other simply requires the application of a series of (essentially linear) commands. The differences in the various modalities are not explored here, as it is not part of the hypothesis of this study. However, two-dimensional tactile sheet of paper provides good frames of reference for blind people to feel 2D images as explained by [Ungar, 2000] and [Takagi, 2009]. Therefore tactile sheet of paper with grids is currently the best tool to check the blind person's comprehension of a virtual 2D screen grid layout to verify location accuracy. Even though sequential commands were used with SETUP09 or finger-exploration strategies were used with a tactile sheet, the representation of the mind-made image remains the same. The study checked for location accuracy using the mind-made image not the difference in modality or strategies. Previous research work of system KEVIN [Blenkhorn and Evans, 1998] and IC2D [Kamel and Landay, 2000] both have used tactile paper against digital systems to evaluate those systems where as other systems such as [Bornschein et al., 2018] and [Fujiyoshi et al., 2018] were not evaluated against any other analogue or digital systems, but on its own with potential blind and sighted users.

6.3.8 Conclusion

In this fist study, the experimental results of the compass-based screen navigation system (SETUP09) was discussed in the creation of computer-aided drawing by BVI individuals. The navigation system is operated by inputting compiler commands. The compass on the system interface takes the user into the intended location where the screen is divided into 3x3 memorable locations. The granularity can be changed by zoomin and zoomout commands. BVI participants tested the system against the conventional tactile grid paper with different screen navigation tasks to determine the accuracy and efficiency of the proposed compass-based navigation technique. The efficiency of the system was determined from the completion time, and error rate was used to determine the accuracy. User surveys have been taken into consideration in assessing user satisfaction of the navigation technique. The general observation made from the study of the compass-based navigation system that it requires very little time and effort to learn and use it, and it does not require additional resources such as assistance from support workers. The next study is mainly focused on 2D drawing mechanisms using the parser commands of the SETUP09 system, which requires the use of the compass-based navigation technique to navigate on the screen. The study will examine the ability of the SETUP09 system to draw a diagram, a flow chart and an image using compass-based navigation. Drawing commands will be invoked for user-defined shapes, primitive geometric shapes, grouping of objects and labelling. The efficiency, accuracy, ease of use and user satisfaction will be evaluated.

6.4 Study 2: Graphics Creation Technique with Blind, Partially Sighted and Sighted Individuals

Study 1 focused on the validity of screen navigation technique, and Study 2 now focuses on the SETUP09 drawing technique. This study is based on an understanding of the difference and similarities between blind, partially sighted, and sighted individuals as was discussed in Chapter 3.

6.4.1 Hypothesis

- H1: SETUP09 2D drawing technique is an efficient drawing technique to draw images for blind, partially sighted and sighted individuals.

  Rationale: Literature findings in Chapter 3 established that different mental models and coping strategies are used by blind and sighted individuals for an external frame of reference [Hill et al., 1993], whereby blind individuals use calculation strategies for spatial understanding and sighted individuals use previous memory. Therefore mental representation is not directly based on vision. Hypothesis H1 is postulated to established the similarities and differences in performance between different subsets of individuals using the SETUP09 system when tactile images are given to participants before performing the task (instructed drawing).

- H2: The SETUP09 2D drawing technique can accurately reproduce mental images of blind, partially sighted, and sighted individuals.

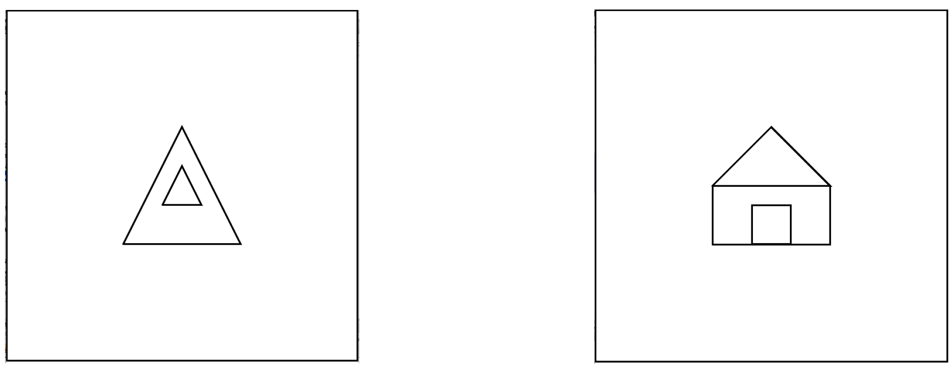
  Rationale: The literature findings in Chapter 3 established that image visualisation in BVI people is not a direct result of vision, but rather multiple senses such as touch and sound as well as preconceived ideas of the world [Lahav and Mioduser, 2002]. Blind, partially sighted, and sighted individuals were asked to imagine 2D images on a nine-cell grid interface and then draw it using the SETUP09 system. Hypothesis H2 is postulated to established the similarities and differences in performance between different subsets of individuals when tactile images are not given before the task is performed (non-instructed drawing).

- H3: SETUP09 2D system-produced tactile images are accurately perceived by blind, partially sighted and sighted individuals.

  Rationale: The literature findings established in Chapter 3 that tactile maps provide a good frame of reference to the external world when understanding spatial representation [Ungar, 2000]. SETUP09 system produced, 2D tactile images was used to assess the understanding of 2D spatial awareness from blind, partially sighted and sighted individuals and recorded their understanding. Hypothesis H3 is postulated to establish the similarities and differences between different subsets of individuals using SETUP09 for image recognition tasks.

6.4.2 Method

- **Instructed Drawing H1:** Task 01 was divided into three parts. BVI participants were given printed swell papers containing shapes of triangles on multiple screen locations, one inside the other and at multiple zoom sizes. The participants were asked to draw the given shapes using SETUP09 exactly as they were on the swell papers. Their drawings were printed on swell paper to compare with the original shapes. Upon completion they were asked to describe the whole picture. The second part of this task was to give the participants an image (i.e. a house) printed on swell paper. The participants were asked to draw the given image exactly as the original image on the swell paper using SETUP09. In the third part of this task they were given a system produced DFD (Data Flow Diagram) printed on swell paper. Participants were asked to draw the image exactly as they felt it on swell paper using SETUP09. The purpose of Task 01 was to find out the participants' ability to reproduce system images with accurate recognition of shapes, locations, and sizes.



(a) A Square (b) A Table

Figure 6.11: Task 01: Triangles Presented for Instructed Drawing

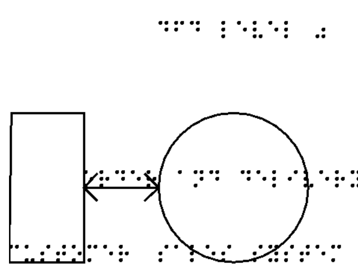


Figure 6.12: Task 01: Diagram Presented for Instructed Drawing

- **non-Instructed Drawing H2:** Task 02 was divided into two parts. The participants were asked to imagine a shape of their choice in a selected screen location with a selected zoom size.

They were asked to verbally describe the shape, which was recorded. They were then asked to produce the imagined shape using SETUP09. There drawing was printed at the end of the task on a swell paper and compared with the original recorded idea. At the end of the task the accuracy of the user-produced image was noted. The second part of this task was for the participants to produce a simple image of their choice using different shapes. Their verbal description of the mental image of this shape was recorded prior to the commencement of the task and this description was used to compare with their system-produced drawing. The purpose of Task 02 was to find out the participants' ability to produce non-guided shapes and images with accuracy and efficiency.

(a) A Stickman

(b) A Chair

Figure 6.13: Task 02 Non-Instructed Drawing: Stickman Produced by a Partially Sighted Participant and a 3D Chair Produced by a Sighted Participants

- **Image Recognition H3:** In the final Task 03 the participants were presented with three system-produced images and diagrams, i.e. a square, a table, a flowchart. The images were printed on swell paper, and gave participants one image at a time. They were asked to explore the images and verbally describe them. The results were categorised into 2 different groups:

  "100%" accuracy means the participants' ability to recognise all shapes, whereas "50%" accuracy means participants' ability to recognise some shapes presented in the image. The purpose of Task 03 was to find out if system-produced art and spatial representation of screen images are accurately perceived by BVI individuals.

(a) A Square (b) A Table

Figure 6.14: Images Presented for Image Recognition Task

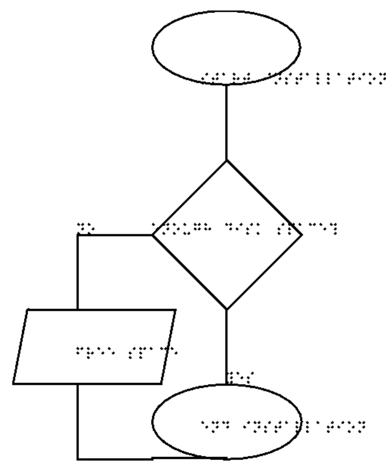


Figure 6.15: Diagram Presented for Image Recognition Task

6.4.3 Procedure

Twenty people were used in the experiment. The experiment was designed to analyse sighted and BVI user's performance variation and similarities. Ten participants with total blindness or partial visual impairment were recruited via the MACS charity. The BVI participants selected, five male and five female, were from different age groups and with different levels of visual impairment. In fact, five participants were completely blind and the other five were predominantly diagnosed with Microphthalmia and Coloboma. All ten BVI participants were registered blind and with an average age of 22 years and a standard deviation of 12.81, their ages ranging from 7 to 49 years old. All 10 BVI participants did have prior tactile exploration experience and 6 BVI participants had tactile graphics creation experience. Two BVI participants were children who had a moderate level of computer literacy. The other ten participants were sighted people, four male and six female, with an average age of 29.4 years and a standard deviation of 11.03. Their ages ranged from 12 to 40 years old.

Table 6.5 Table 6.5 shows the participants' details, in particular, type of vision, gender, age, the reason for blindness, age at which they became blind, computer literacy (CL), prior tactile graphics creation (TGC) experience. Their computer literacy ranged from moderate to high, which was based on the participant's use at work of computers, keyboard and word processing applications.

Person	Vision	Gender	Age	Reason for Blind	Age at Blindness	Education	CL	TGC
P1	None	M	26	Microphthalmia	Birth	Graduate	High	Yes
P2	None	F	21	Microphthalmia	Birth	School leaver	High	Yes
P5	None	M	37	Microphthalmia and Coloboma	Birth	School leaver	High	No
P6	None	F	20	Microphthalmia and Anophthalmia	Birth	School leaver	Low	Yes
P7	None	M	14	Microphthalmia and Coloboma	Birth	Student	High	Yes
P3	Some Vision	M	15	Microphthalmia and Coloboma	Birth	Student	High	Yes
P4	Some Vision	M	10	Microphthalmia and Coloboma	Birth	Student	Moderate	No
P8	Some Vision	F	7	Microphthalmia and Coloboma	Birth	Student	Moderate	No
P9	Some Vision	F	49	Glaucoma and Microphthalmia	Birth	School leaver	Low	No
P10	Some Vision	F	27	Cortical Visual Impairment	Birth	School leaver	High	Yes
P11	Sighted	F	40	NA	NA	School leaver	Low	NA
P12	Sighted	F	14	NA	NA	Student	Moderate	NA
P13	Sighted	F	17	NA	NA	Student	High	NA
P14	Sighted	F	29	NA	NA	Graduate	High	NA
P15	Sighted	M	31	NA	NA	School leaver	Low	NA
P16	Sighted	M	40	NA	NA	School leaver	High	NA
P17	Sighted	M	12	NA	NA	Student	Moderate	NA
P18	Sighted	M	38	NA	NA	Graduate	High	NA
P19	Sighted	F	36	NA	NA	Graduate	High	NA
P20	Sighted	F	37	NA	NA	Graduate	High	NA

Table 6.5: Demography of the Participants. CL = Computer Literacy, TGC = Experience with Tactile Graphic Creation, NA = Not Applicable

Each participant was presented with three tasks to complete and their performance was measured. Each participant was given roughly 30 minutes of training on the SETUP09 system. The training was split across three experimental tasks and included: (i) introduction to SETUP09 and its drawing and navigation language; (ii) familiarity with SETUP09 hotkeys and help keys; (iii) hands-on practice using the prototype and different drawing commands; (iv) steps to draw simple shapes and images, and to draw in different zoom levels and locations; (v) and labelling and defining an image. The user interface was hidden from sighted participants during the training. Sighted participants were blindfolded during image recognition activities and given 10 minutes on the introduction of the system. If participants needed assistance during the experiment they were referred to the help menu. It was observed that participants were able to read the screen labels and seek system help with orientations, to memorise and build a layout map of the interface, and complete instructed and non-instructed drawing task. Audio feedback was provided with help commands. Very few tasks were misunderstood however participant were allowed to redo them.

The experimental system consists of the SETUP09 prototype that operates on Mac and Windows operating systems with TTS (text to speech system), and Zyfuse Heater with swell paper to produce the tactile images. To ensure higher accuracy in tactile image recognition, simple images were created with thick lines and plenty of space between shapes. Participants who preferred the Windows keyboard were provided with Windows computers during the actual experiment so that, there was no disadvantage due to unfamiliarity and therefore time delay. For this experiment, the prototype functions were as follows:

(1) use hot key and keyboard input; (2) navigate to a specific location on the screen; (3) display shapes when manipulated; (4) save and label images; and (5) send images to printer and heater output for tactile images. Swell paper was used during the experiment to print haptic images for image recognition tasks. This is a special kind of paper upon which images and art can be printed or sketched and turned into tactile representations. It can be used with a marker pen, printer or photocopier and when the paper is subsequently subjected to thermal treatment the dark areas creates raised-relief lines.

Time was recorded for all three tasks using a stopwatch. Data was collected from the 20 participants. The first task had three activities, the second task had two activities, and the third task had three activities. Two blocks of (3*20) trials and one block of (2*20) trials were collected from the 20 participants. In total there were 160 trials done. The trials were recorded for accuracy, errors, and efficiency. Eight attempts had to be redone by the participants due to confusion over instructions; also, five erroneously recorded trials needed repeating. There was no difference in performance by gender, however observations based on age and prior experience are discussed later in this chapter.

It was observed that blind individuals needed more guidance and explanation than the partially sighted individuals. However, blind individuals performed with confidence when the concept and system commands were clearly understood, explained the next section with statistics. Participants' performance, task feedback and usability questionnaire were used to decide on the suitability, ease of use and effectiveness of the SETUP09 system.

6.4.4 Results

The performance for all three tasks, i.e. instructed drawing, non-instructed drawing and image recognition, was considered by examining the variations between different groups, phases (activities) and visual acuity (blind, partially sighted and sighted individuals).

The total length of time it took for participants to perform the tasks was recorded including the time taken for errors. Time was recorded from the point at which the participant started to use the system until the end of the given task. It was observed that this measurement was impacted by degree of familiarity with the keypad, keyboard typing speed, recollection from memory of the system commands and the speed at which the system gave feedback messages.

The next section presents the results of the first hypothesis: The SETUP09 2D drawing technique is an efficient method to draw images for blind, partially sighted and sighted individuals.

- **Instructed Drawing: H1**

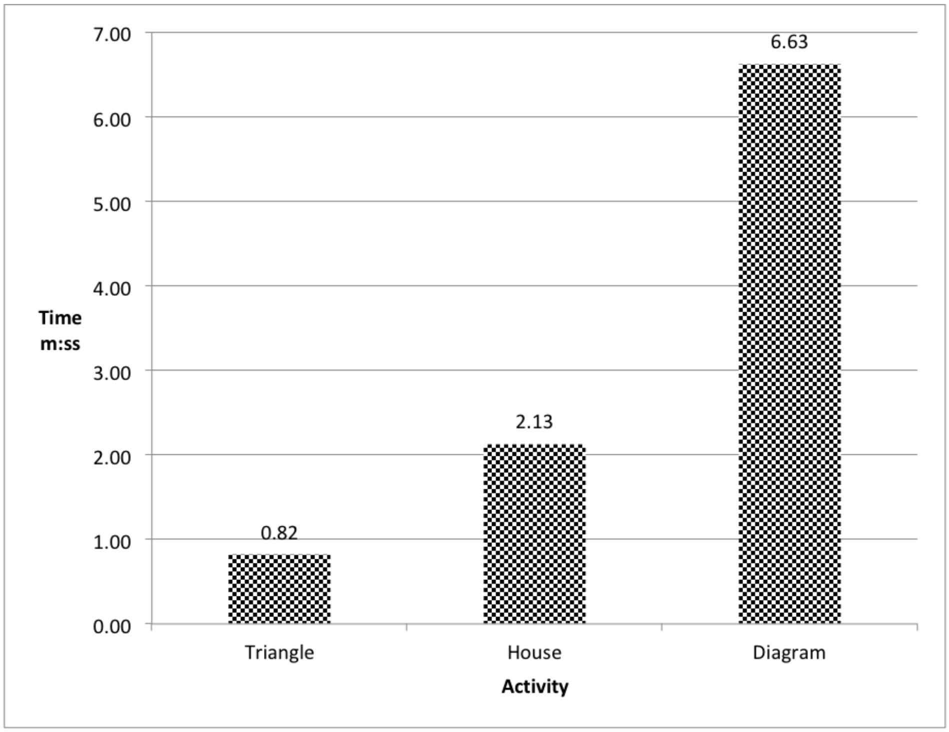


Figure 6.16: Instructed Drawing: Average Task Completion Time Based on Tasks of all participants (Blind, Partially Sighted, Sighted)

Performance was measured on time taken, errors made and output accuracy for the participants to examine the shapes, their locations and sizes. The bar chart in Figure 6.16 shows the average time taken to complete the task on the instructed drawing by all participants. The mean time taken to draw a diagram in minutes and seconds (m:ss) = 6.63, SD = 2.24 m:ss; the mean time taken to draw an image (house) = 2.13, SD = 1.19 m:ss; and the mean time taken to draw a shape (triangle) = 0.82, SD = 0.69 m:ss. Participants took more time to draw complex images such as the diagram and less time to draw simple shapes such as triangles. However the time spent was still lower than the time reported by other similar systems such as IC2D by [Kamel and Landay, 2002] which took participants around 7 minutes to complete without any diagram drawing, and it took participants 19 minutes to complete the pin-matrix haptic system experiment by [Bornschein et al., 2018].

Participant Type	Size	Shapes Average	Image Average	Diagram Average	Shape SD	Image SD	Diagram SD
Blind	5	0.77	2.24	7.56	0.64	1.20	2.37
Partially Sighted	5	0.87	2.01	5.69	0.81	1.37	1.87
Sighted	10	0.31	0.92	3.76	0.11	0.58	1.71

Table 6.6: Analysis of Instructed Drawing Performance in minutes and seconds. SD = Standard Deviation

Participant Type	Shapes	Image	Diagram
Blind vs Partially Sighted	$t = 0.41, p = 0.83 > 0.05 = \alpha$	$t = 0.38, p = 0.77 > 0.05 = \alpha$	$t = 0.10, p = 0.20 > 0.05 = \alpha$
BVI (Blind and Partially Sighted) vs Sighted	$t = 0.022, p = 0.044 < 0.05 = \alpha$	$t = 0.006, p = 0.013 < 0.05 = \alpha$	$t = 0.002, p = 0.004 < 0.05 = \alpha$

Table 6.7: T-Test values of Instructed Drawing Performance between different subgroups.

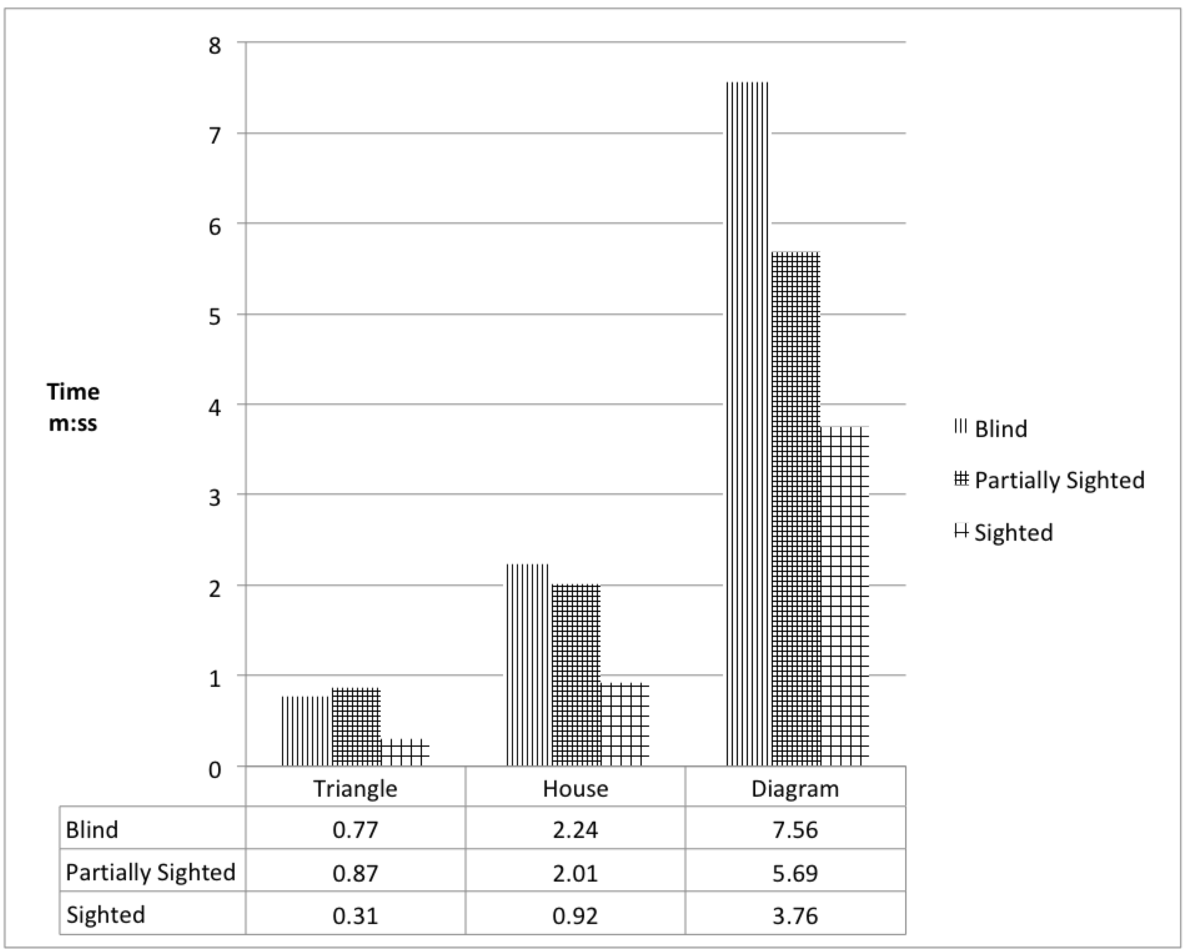
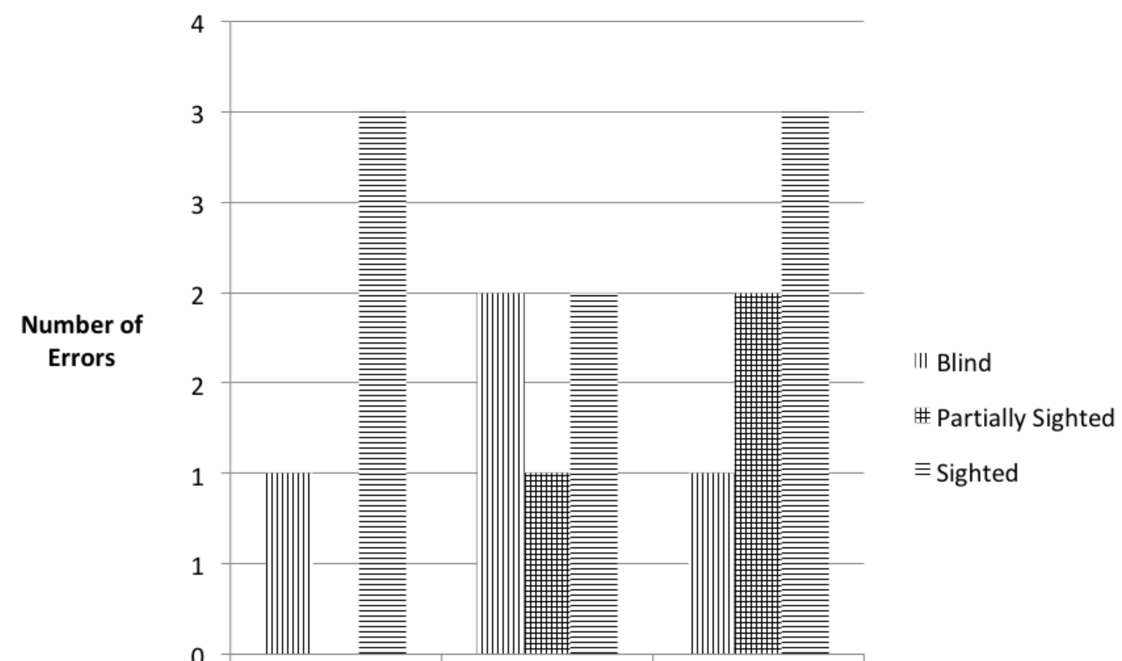


Figure 6.17: Instructed Drawing: Average Task Completion Time Based on Tasks and Vision with Blind (n=5), Partially Sighted (n=5) and Sighted participants (n=10)

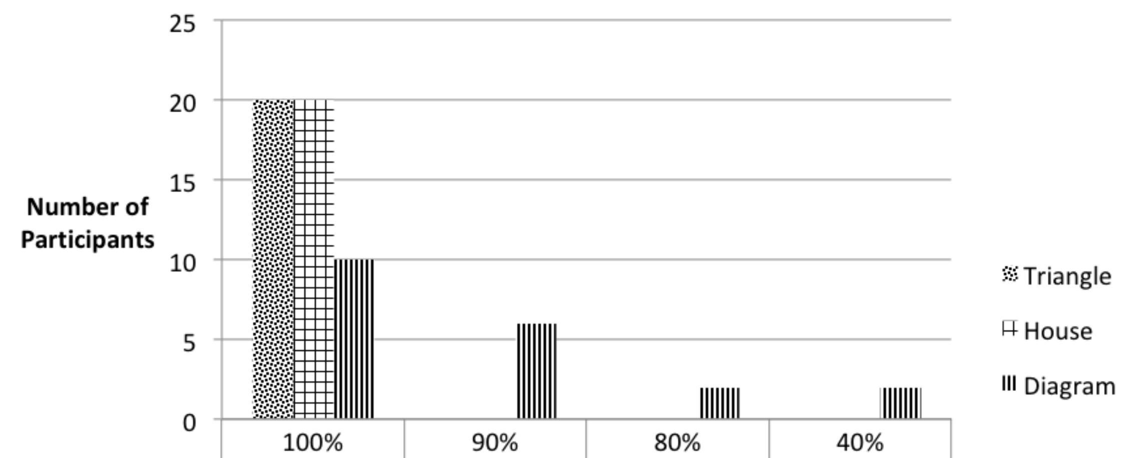
Figure 6.17 bar chart illustrates the variation of data among different cohorts. Table 6.6 gives the descriptive statistics of instructed drawing performance and Table 6.7 gives inferential statistics of the instructed drawing task. Even though the completion time difference between blind and partially sighted participants was not statistically significant where the P value is not less than 0.05, the completion time difference between sighted and BVI groups was statistically significant as recorded $P < 0.05$ in all three instructed drawing tasks. Further discussion of these results is presented in the section on "Findings of Study 2".



	Triangle	House	Diagram
Blind	1	2	1
Partially Sighted	0	1	2
Sighted	3	2	3

Figure 6.18: Instructed Drawing: Errors made During Instructed Drawing Task based on Blind (n=5), Partially Sighted (n=5)and Sighted participants (n=10)

Figure 6.18 shows the errors of Task 01 on instructed drawing. The blind and partially sighted group made seven errors and the sighted group made eight errors overall. The BVI group made fewer errors than the sighted group.



	100%	90%	80%	40%
Triangle	20	0	0	0
House	20	0	0	0
Diagram	10	6	2	2

Completion Percentage of All Participants

Figure 6.19: Instructed Drawing: Completion Percentage of all participants (Blind, Partially Sighted, Sighted)

Figure 6.19 shows the accuracy of Task 01 on instructed drawing. Accuracy is defined by the absolute difference between the shapes produced by participants and the given shape. We recorded accuracy in four categories. We recorded 100% accuracy for producing a given image with no difference; 90% accuracy for producing a given image with 1-2 changes to the original image; 80% accuracy for producing the given image with 3-4 changes to the original image; and 40% for producing 5-10 changes to the original image during image creation, e.g. navigation to incorrect location, referring to incorrect points, calling incorrect shapes, incorrect input commands and input command errors which added extra time for completion.

Errors were mainly due to forgetting commands and lack of prior system use. Figure 6.19 shows that the participants completed simple tasks, but found it challenging to complete more complex tasks such as diagrams. The bar chart shows that all of them completed the shape (triangles) and image (house) activity with 100% accuracy, but only 10 out of 20 participants managed to get 100% or more of the diagram completed. Completion was highly correlated with the difficultly of the task.

The blind and partially sighted groups accurately produced the given shapes (shape, diagram and image) with 77% accuracy. They took less than a minute on average to produce a given shape. Activity 03, the diagram drawing activity, took more time (mean = 6.72 m:ss) as it was challenging due to multiple navigation steps across the screen, manipulation of multiple shapes at different zoom levels and from different starting points. Not only did the challenges of the activity impact on the time, but as mentioned previously the short period of system practice also added to the difficulty.

Hypothesis H1 is supported by the above results on errors made, accuracy and completion time as indicators of the efficiency of the system among blind, partially sighted and sighted participants. The above data demonstrates that compared with partially sighted participants, the blind participants performed better in the shape activities, but not with image or diagram drawing. However, on overall the sighted participants performed much better than blind and partially sighted participants. The performance of blind and partially sighted groups is not statistically significant, but the performance of sighted groups is statistically significant compared to the BVI group as a whole. Some of the BVI users reported they took a longer time to complete tasks due to several factors: needing more time for system practice, to get used to the system. The next section presents hypothesis two H2, i.e. the SETUP09 2D drawing technique can accurately reproduce mental images of blind, partially sighted, and sighted individuals.

- **Non-Instructed Drawing: H2**

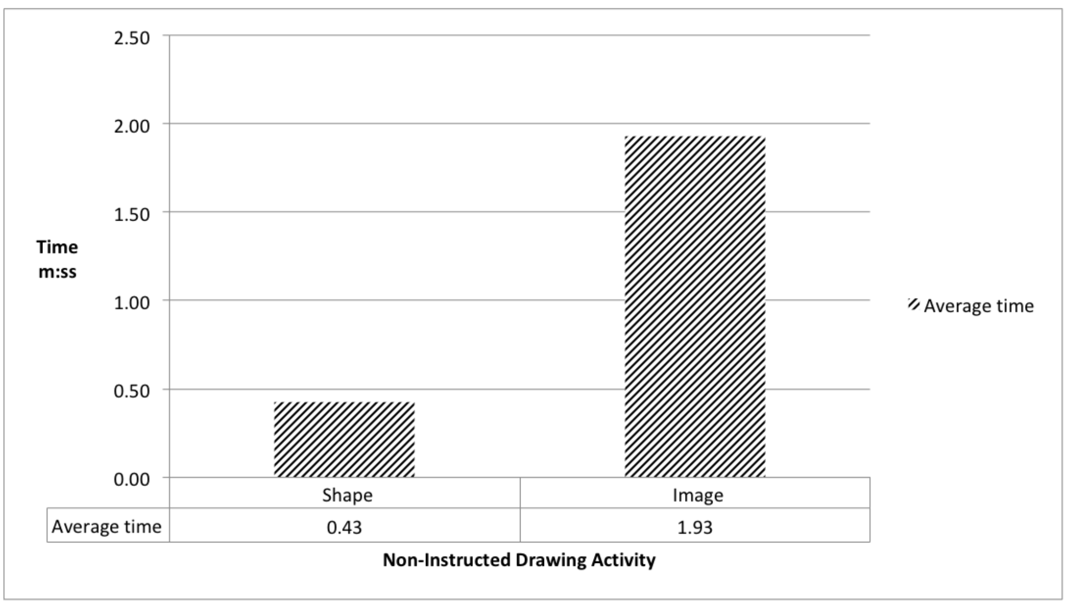


Figure 6.20: Completion Time of Non-instructed Drawing with all participants (Blind, Partially Sighted, Sighted)

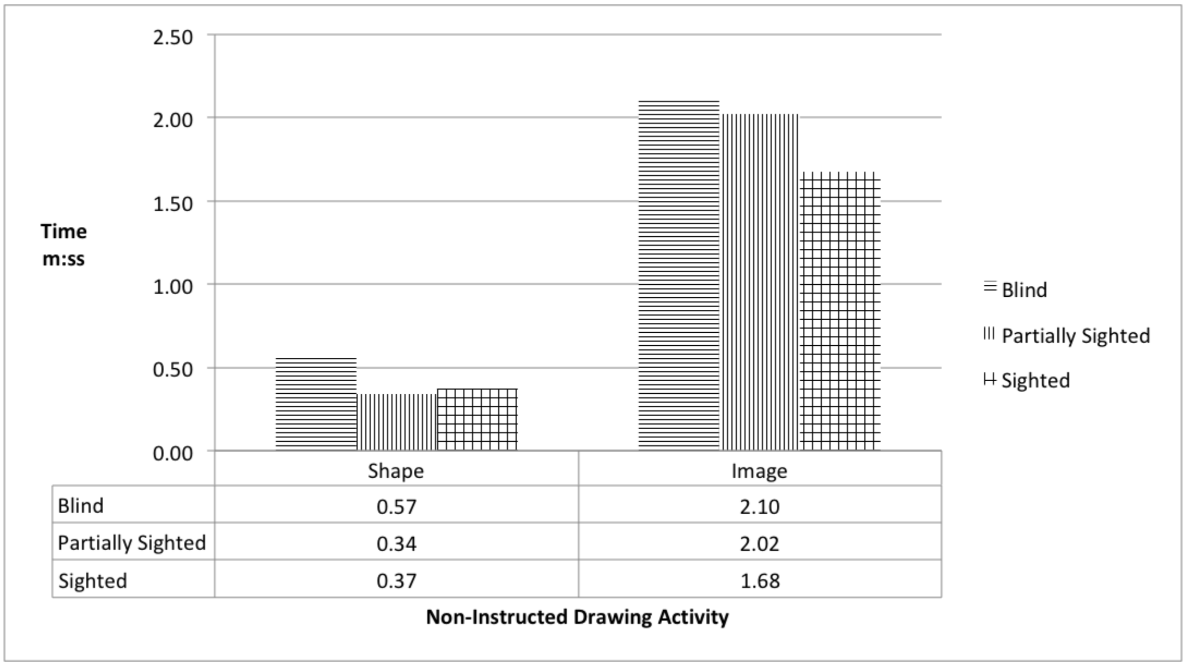


Figure 6.21: Completion Time of Non-instructed Drawing based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants

Participant Type	Size	Shapes Average	Image Average	Shape SD	Image SD
Blind	5	0.57	2.10	0.53	0.68
Partially Sighted	5	0.34	2.02	0.11	0.70
Sighted	10	0.37	1.68	0.38	0.70

Table 6.8: Analysis of Non-Instructed Drawing Performance in Minutes and Seconds. SD = Standard Deviation

Participant Type	Shapes	Image
Blind vs Partially Sighted	$t = 0.19, p = 0.39 > 0.05 = \alpha$	$t = 0.42, p = 0.85 > 0.05 = \alpha$
BVI (Blind and Partially Sighted) vs Sighted	$t = 0.31, p = 0.62 > 0.05 = \alpha$	$t = 0.11, p = 0.22 > 0.05 = \alpha$

Table 6.9: T-Test values of Non-Instructed Drawing Performance Between Different Subgroups.

The strip chart in Figure 6.21 strip chart demonstrate the range of times taken for non-instructed drawings. Table 6.8 gives the descriptive statistics of non-instructed drawing performance, and Table 6.9 gives inferential statistics for the non-instructed drawing task. The difference between the blind and partially sighted groups was not statistically significant during the non-instructed shape drawing and image drawing tasks. Also the difference between the sighted and BVI groups was not statistically significant during the non-instructed shape drawing and image drawing tasks where P value is not less than 0.05. Overall, sighted and BVI participants performed statistically closer to each other when instructions were not given.

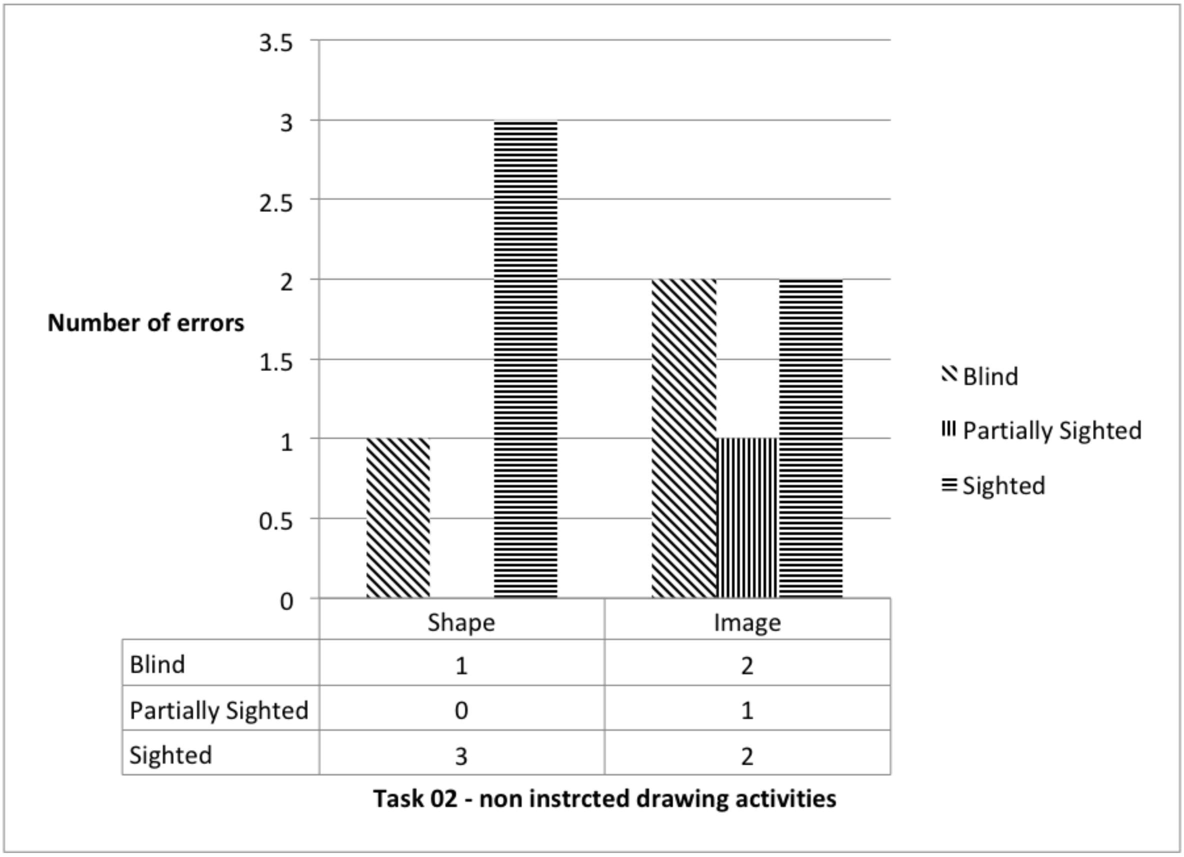


Figure 6.22: Errors of Non-instructed Drawing based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants

Figure 6.22 gives the errors made during Task 02, non-instructed drawing. Overall, nine errors were made, three errors were from the blind group, one error was from the partially sighted group, and five errors were from the sighted group where the number of participants was 10. Errors were mainly due to forgetting commands and lack of prior system knowledge. Blind participants completed 70% of trials without errors and partially sighted participants completed 90% of trials without errors, whereas sighted participants made 75% trials without errors.

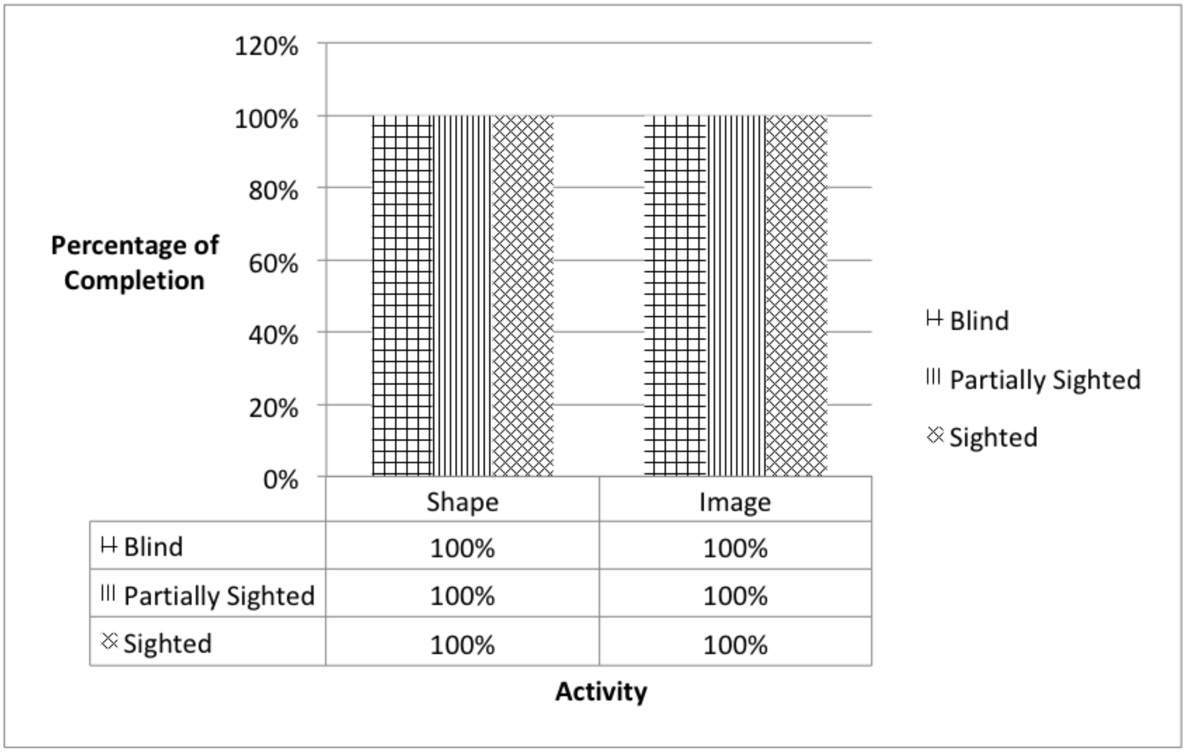


Figure 6.23: based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants

Figure 6.23 shows that both blind and BVI participants completed the planned/recorded drawing and stayed with the task until completion. They also had more practice of the system from Task 01. They produced shapes and images with 100% accuracy taking an average time of 1.26 (m:ss), whereas in the case of sighted participants the mean completion time recorded was 1.02 (m:ss) Accuracy was defined by the absolute difference between the shape produced by participants and their recorded shape at the start of the activity. Participants clearly enjoyed more drawing shapes and images of their choice than by following instructions for a prescribed shape.

Hypothesis H2 is supported by the above results on accuracy and errors made as indicators of the SETUP09 drawing technique's ability to accurately reproduce mental images of BVI individuals. The above data demonstrates that partially sighted participants were the leading performers in terms of speed of completion compared to the blind group. However, both groups reported 100% accuracy in creating their preconceived mental images using SETUP09. The partially sighted group made less errors compared to their blind counterparts. The above data demonstrates that sighted participants were the leading performers when it came to speed of completion. However both groups reported 100% accuracy in manipulating their conceived mental images using SETUP09. In fact, BVI participants (20%) made fewer errors compared to their sighted counterparts (25%).

The next section discusses the results of Hypothesis 3: SETUP09 2D system-produced tactile images are accurately perceived by blind, partially sighted and sighted individuals.

- **Image Recognition: H3**

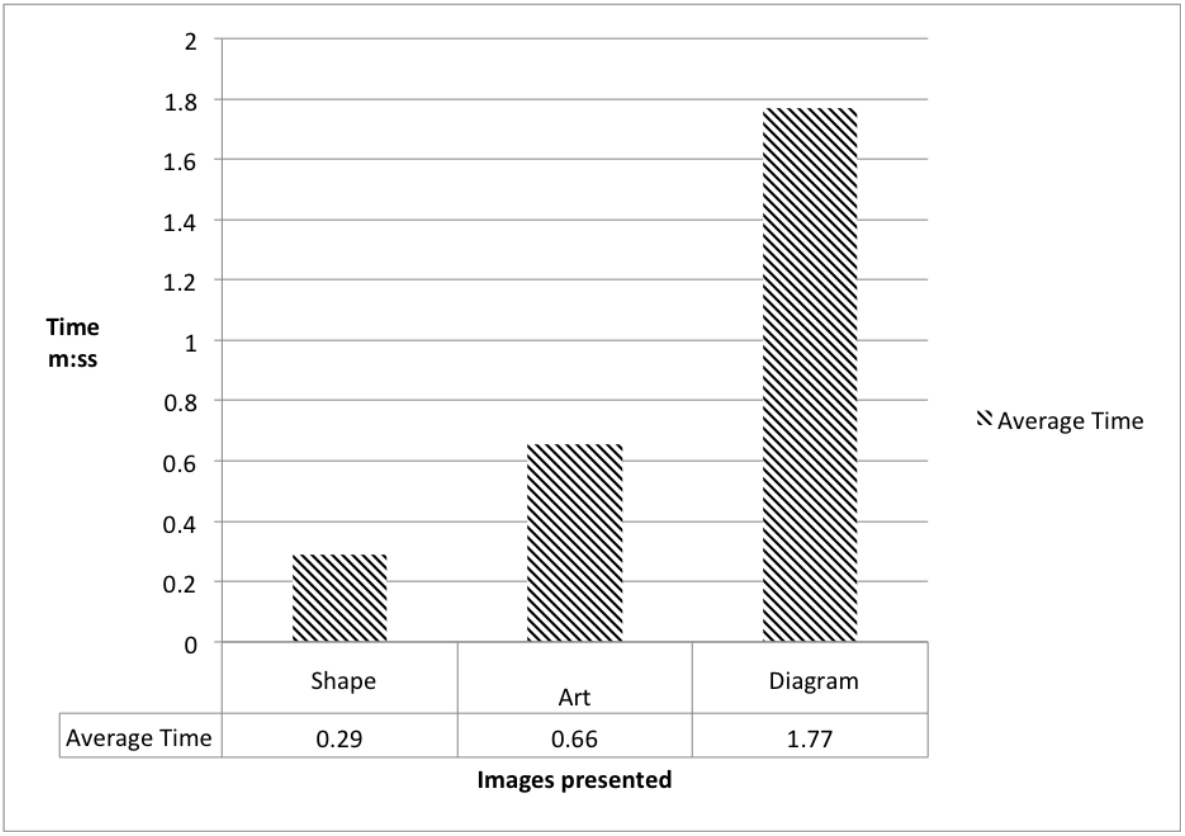


Figure 6.24: Image Recognition Time of All Participants (Blind, Partially Sighted, Sighted)

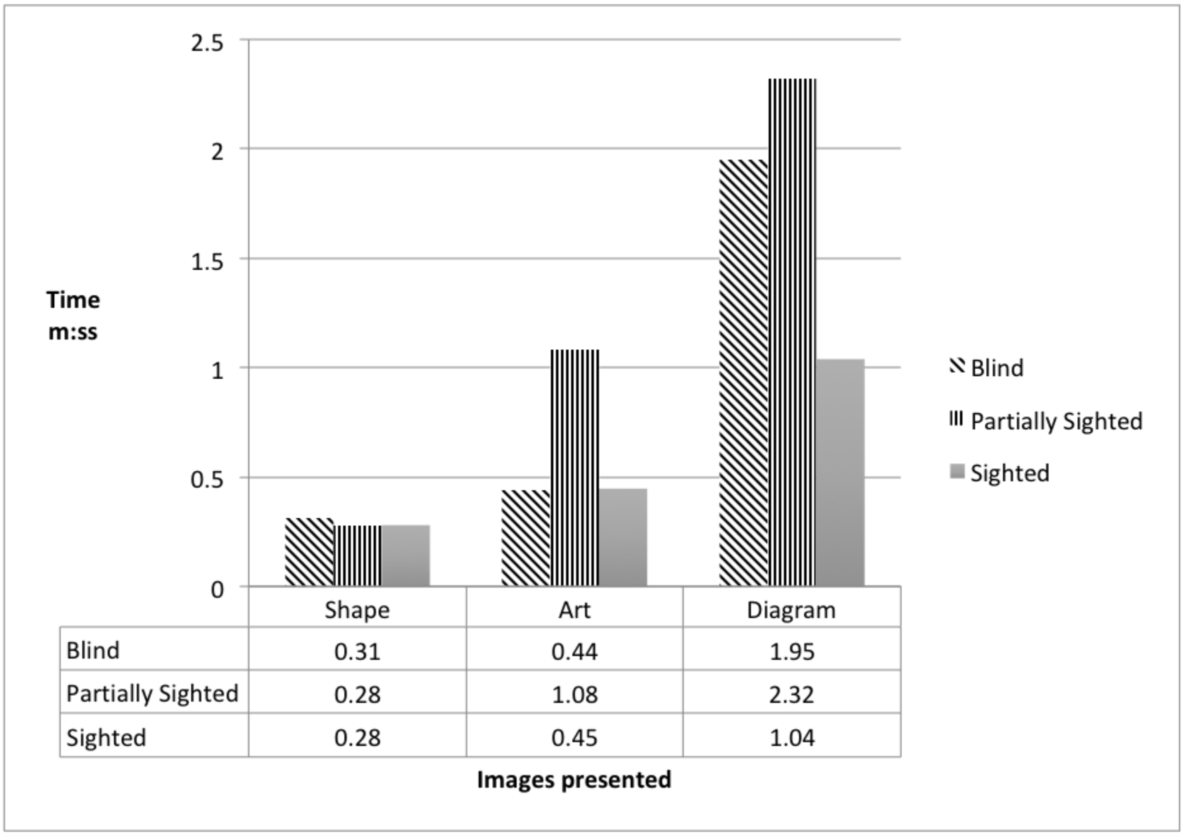


Figure 6.25: Image Recognition Time based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants

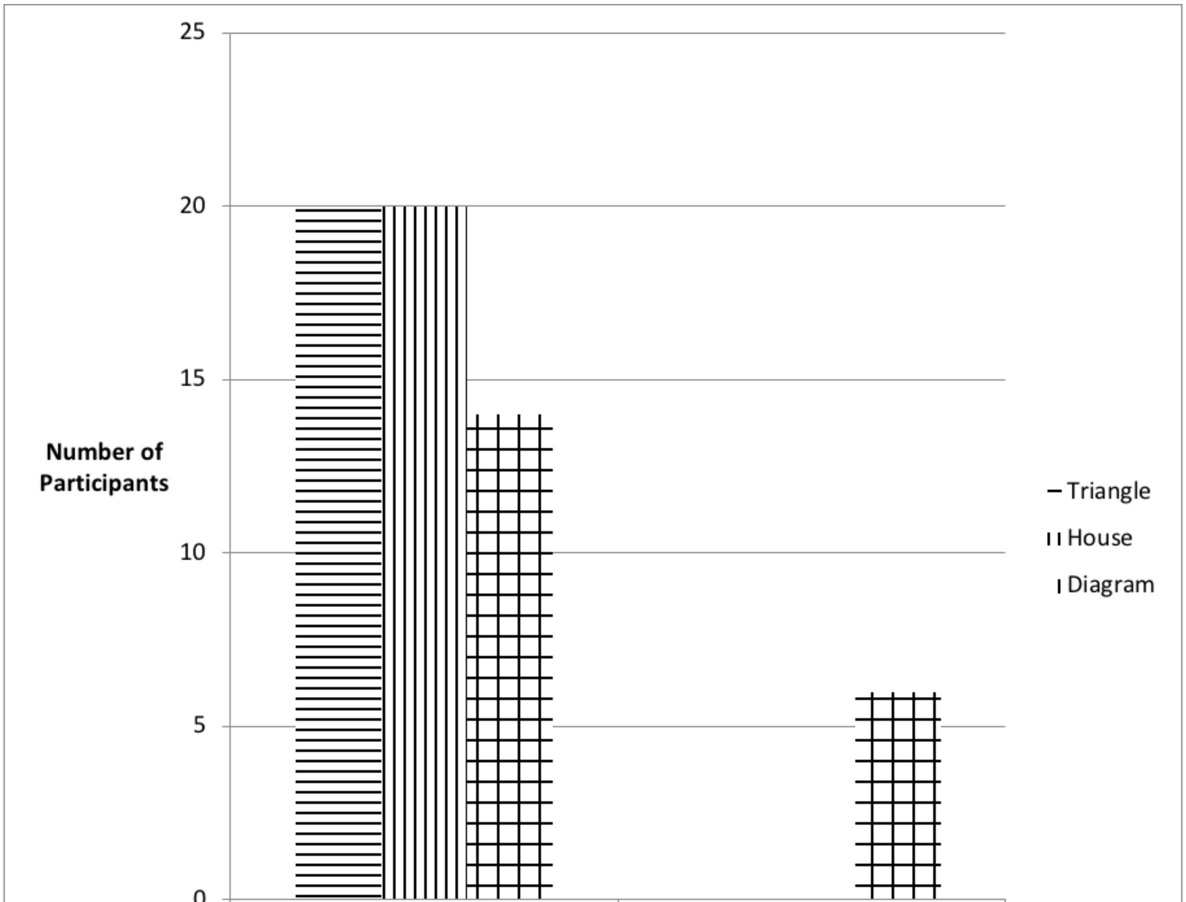
Participant Type	Size	Shapes Average	Image Average	Diagram Average	Shape SD	Image SD	Diagram SD
Blind	5	0.31	0.44	1.95	0.12	0.33	1.13
Partially Sighted	5	0.28	1.08	2.32	1.01	0.52	0.58
Sighted	10	0.28	0.45	1.04	0.08	0.56	0.88

Table 6.10: Analysis of Image Recognition Performance in minutes and seconds. SD = Standard Deviation

Participant Type	Shapes	Image	Diagram
Blind vs Partially Sighted	$t = 0.11, p = 0.23 > 0.05 = \alpha$	$t = 0.027, p = 0.05 = 0.05 = \alpha$	$t = 0.27, p = 0.54 > 0.05 = \alpha$
BVI (Blind and Partially Sighted) vs Sighted	$t = 0.31, p = 0.62 > 0.05 = \alpha$	$t = 0.11, p = 0.22 > 0.05 = \alpha$	$t = 0.006, p = 0.012 < 0.05 = \alpha$

Table 6.11: T-Test values of Image Recognition Performance between different subgroups.

The Figure 6.25 bar chart shows the variation in the data among different cohorts. Table 6.10 gives the descriptive statistics of image recognition performance and Table 6.11 gives inferential statistics for the image recognition task. The results reveal that sighted participants were more efficient in identifying shapes, images and diagrams produced by the SETUP09 system. The difference between the blind and the partially sighted participants was not statistically significant, nor among BVI and sighted participants except in the diagram recognition task. Sighted participants were ahead compared with blind or partially sighted participants in terms of the time taken in identifying shapes/images and diagrams produced by the system. Recognition of the diagram was difficult by all participants, the main reason being unfamiliarity with flowcharts and the complexity of multiple shapes. Most blind and partially sighted participants have never attempted to study or draw flowcharts due to the complexity of visual details. However, most blind and partially sighted participants managed to recognise individual shapes with high accuracy.



	100 % Recognition	50% Recognition
Triangle	20	0
House	20	0
Diagram	14	6

Figure 6.26: Image Recognition Accuracy of all Participants (Blind, Partially Sighted, Sighted)

Figure 6.26: Image Recognition Accuracy for all Participants (Blind, Partially Sighted, and Sighted)

The bar chart in Figure 6.26 categorises completion of the recognition task by participants into 100%, and 50%. "100%" means the participants' ability to recognise all shapes, whereas "50%" means participants' ability to recognise some shapes presented in the image. An accuracy of 87% is reported with full image recognition among BVI participants, whereas sighted participants had an accuracy 93%.

Both groups recognised most of the art and shapes presented. This conforms that no one had great difficulty. BVI and sighted participants fully recognised system-produced shapes, images and diagrams with 87% accuracy. A few attempts were made with partial recognition but that was mainly in the activity involving recognising the diagram. Overall, the participants took less than a minute on average to recognise a given shape and slightly longer to recognise a diagram. Recognition of the diagram was challenging due to the need for having knowledge on flowcharts and multiple shapes on the screen. This knowledge can easily be gained by learning about flowcharts and with the use of tactile diagrams.

Hypothesis H3 is supported by the above results on accuracy and the number of errors as indicators to measure the ability to perceive system-produced tactile images. The above data demonstrates that partially sighted participants performed better than blind participants, but sighted participants were the leading performers when it came to time taken. However, both groups reported over 90% full recognition of system-produced images. In fact, all BVI and sighted participants consistently recognised images, shapes and diagrams (87%) of the presented drawing.

6.4.5 Other Observation: Age and Previous Experience

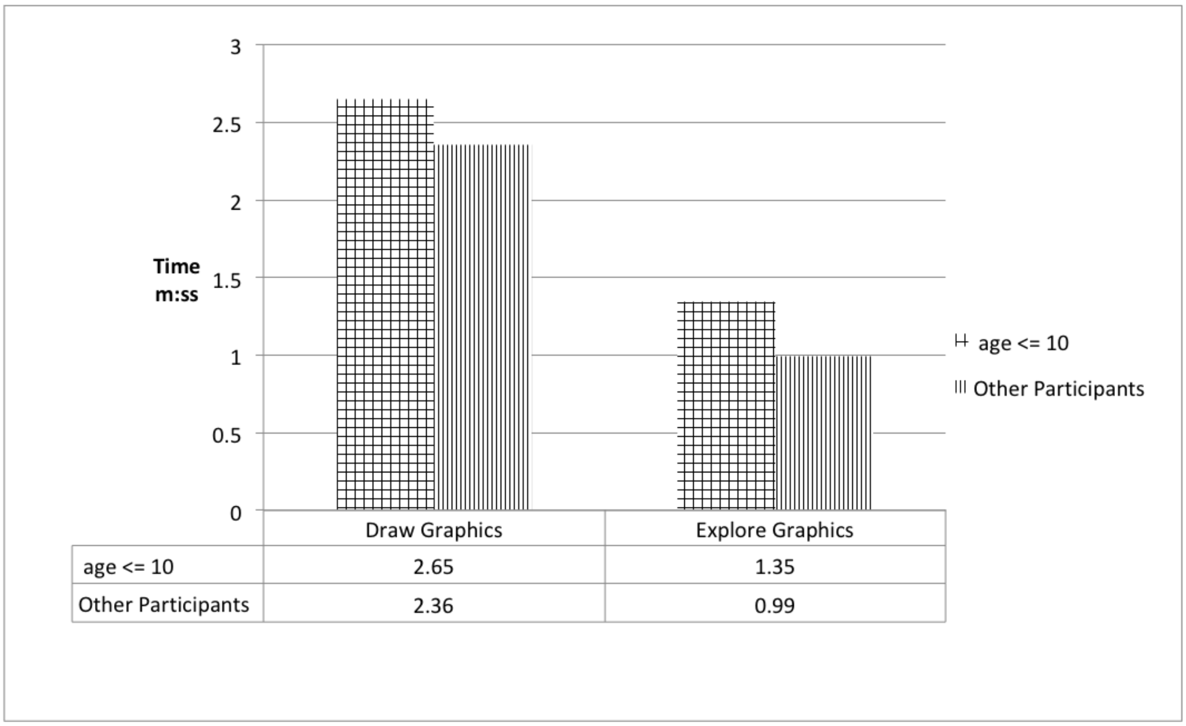


Figure 6.27: Average Performance Time Based on the age of participants (Blind, Partially Sighted)

Figure 6.27 shows the average time taken by children are less than or equal to the age of 10 versus other participants. Child participants took an average time of 2.65 m:ss to create graphics and 1.35 m:ss to explore graphics, where other participants took 2.36 m:ss to create graphics and 0.99 m:ss to explore graphics. Child participants took a longer time than other participants to create and explore graphics according to the results obtained.

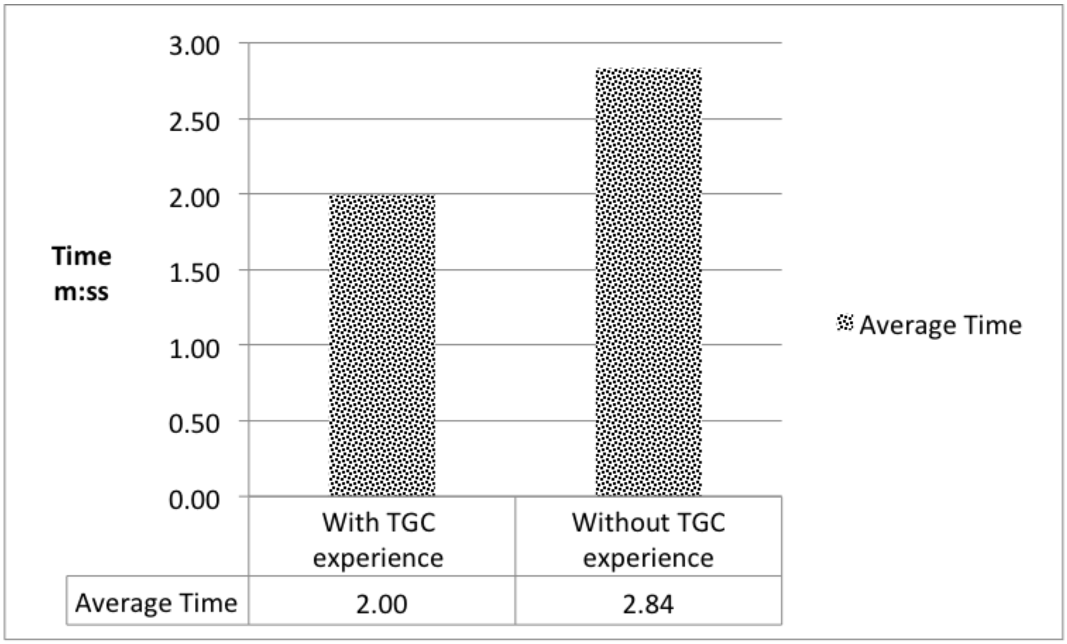


Figure 6.28: Average Performance Time Based on Previous (TGC) Tactile Graphic Creation Experience(Blind, Partially Sighted)

Figure 6.28 shows the average times taken by blind and partially sighted participants who have or have not experience in tactile graphics creation. Participants with tactile graphics creation experience completed the drawing tasks with an average time of 2.00 m:ss and participants without tactile graphics creation experience complete it in 2.84 m:ss. These results demonstrate that prior experience of tactile graphics creation is important in determining the amount of time taken.

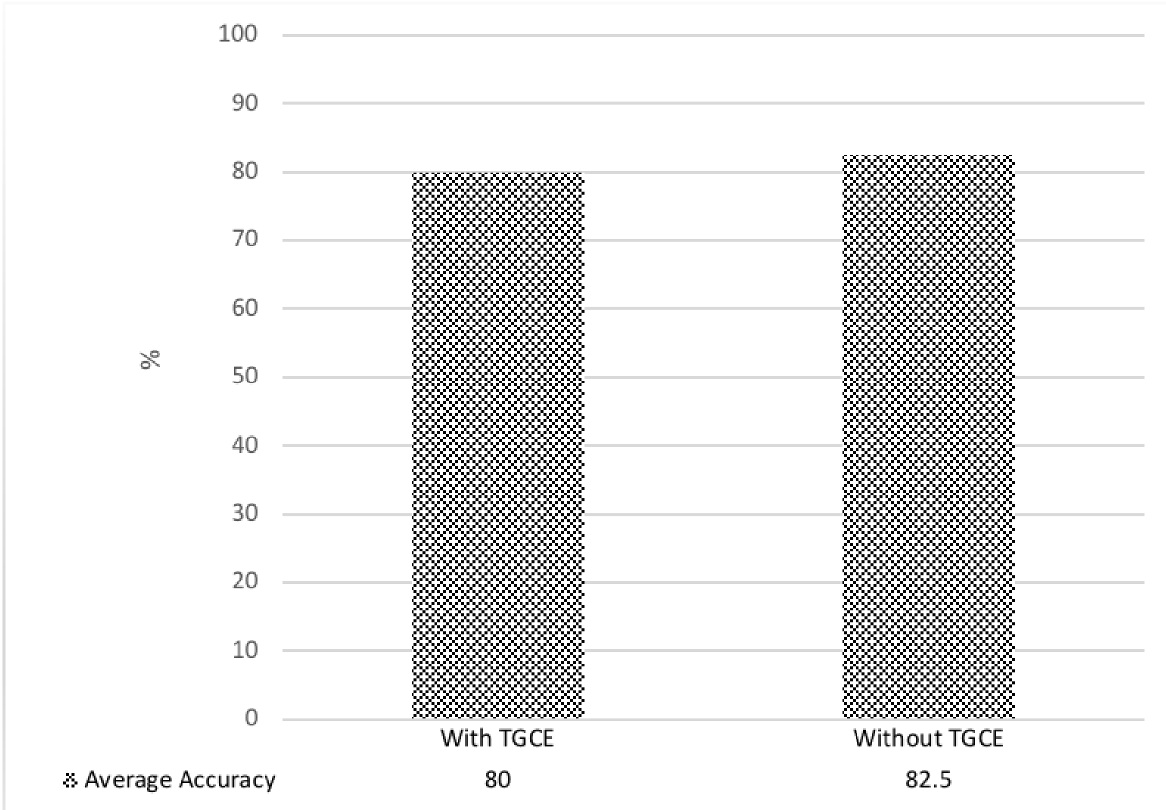


Figure 6.29: Average Accuracy of Diagram Drawing Task with Blind and Partially Sighted Participants with and without the Experience of Tactile Graphic Creation

Figure 6.29 shows the average accuracy of blind and partially sighted participants who have or have not experience in tactile graphics creation. Participants with tactile graphics creation experience completed the drawing tasks with an average accuracy of 80% and participants without tactile graphics creation experience complete it with 82.5%. The result demonstrate that prior experience of tactile graphics creation has not had much impact with accuracy in this experiment as participants with and without prior experience performed equally well with accuracy with minor variation of 2.5%. But the above results informs the impact of prior experience with time.

6.4.6 The post SETUP09 questionnaire

The post-SETUP09 questionnaire reports the alpha value of 0.74 for scale reliability with acceptable internal consistency, coefficient of reliability= -0.741, mean= 2.8 and SD=3.6. All participants who completed all three tasks completed the post-SETUP09 task questions. Five questions were posed at the end of three tasks. We asked all participants to give ratings for the five different aspect of the SETUP09 system. Level 1 signifies agreeing strongly and Level 7 signifies disagreeing strongly to the questions asked.

1. (Q1) SETUP09 is efficient.

2. (Q2) SETUP09 is easy to use.

3. (Q3) SETUP09 is supportive.

4. (Q4) SETUP09 builds a navigation model in participant's mind.

5. (Q5) SETUP09 builds a layout model in the participant's mind.

Level 1 and 2 were repeatedly selected by the majority of the participants throughout the questionnaire for all the questions except the one about the supportiveness of the system. Participants suggested features such as auto text correction, redo and undo options for some potential improvements, as well as better system feedback for error correction. Both BVI and sighted participants picked similar score rating on the above-mentioned questions. The results of this questionnaire reveals a slightly higher number of BVI users agreeing with system efficiency and ease of use than sighted participants, whereas sighted participants were more keen on system supportive functionalities. However, both user groups agreed on the system's ability to create a navigation and layout model.

Seventeen out of 20 participants agreed that SETUP09's technique is efficient and the technique facilitates in building a navigational model in participants' minds (Levels 1 and 2 of the Likert scale); 14 out of 20 participants agreed that the SETUP09 technique is easy to use; and 12 out of 20 strongly agreed that SETUP09 technique facilitates in building a navigational model in participants' mind. However, the majority of participants thought that the system could be more supportive in terms of error detection, correction, input and output functionalities. The data in Table 6.12 and bar chart 6.30 show that participants with no vision predominantly selected Levels 1 and 2 of the Likert scale, whereas sighted and partially sighted participants thought the prototype could be further improved. Two-way ANOVA gives a P-value of 0.0045, which is less than 0.05. This demonstrates that there is a difference between the post-experiment feedback among blind and partially sighted participants. Table 6.13 shows participants' post-experiment feedback on the system, which is discussed in detail in the next section.

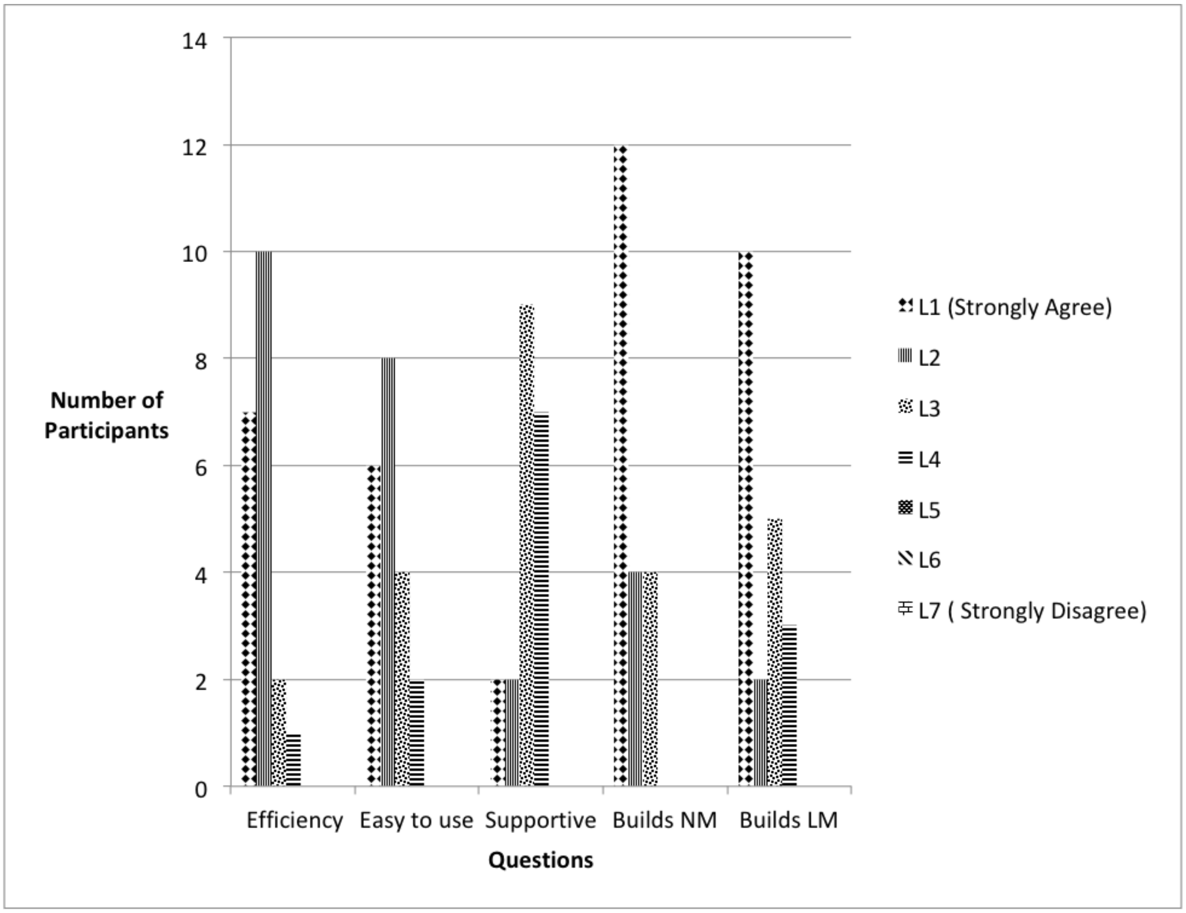


Figure 6.30: Answers to Post Study Questions by all Participants demonstrated in a bar chart

Person	Vision	Efficiency	Easy to use	Supportive	Builds NM	Builds LM
P1	No Vision	1	1	3	1	1
P2	No Vision	2	1	1	1	1
P5	No Vision	1	2	2	1	3
P6	No Vision	2	2	3	1	2
P7	No Vision	2	2	3	2	1
P3	Some Vision	4	3	4	1	3
P4	Some Vision	2	2	3	2	3
P8	Some Vision	1	2	3	2	1
P9	Some Vision	2	4	4	3	4
P10	Some Vision	1	1	4	1	1
P11	Sighted	1	1	1	1	1
P12	Sighted	4	3	4	3	4
P13	Sighted	2	3	4	3	3
P14	Sighted	1	2	3	1	1
P15	Sighted	2	2	3	1	2
P16	Sighted	2	4	4	2	3
P17	Sighted	4	3	4	3	4
P18	Sighted	2	1	3	1	1
P19	Sighted	1	1	2	1	1
P20	Sighted	2	2	3	1	1

Table 6.12: Answers to Post Study Questions based on Blind (n=5), Partially Sighted (n=5), Sighted (n=10) participants

Vision	Views of the participants
No Vision	The compass system is very intuitive, I can visualise the art I produce, This is helpful for schools with VI students, Waiting time for the system feedback seems to be obstructive, Many grids on printed paper is difficult to read, Different levels of system feedback is needed, More time is required to get used to the system, Tactile gridpad for such a system is a useful tool.
Some Vision	The system is effective and interesting, I like if the software corrects typing errors, The bigger text is required for people with low vision and lines with different thickness, System interaction could be faster, Ability to change system voice and auto text correction facility is needed.
Sighted	I need more focus to visualise images with my eyes folded, Great if system corrects my typing errors, I like the help facility, Takes time to get used to the system, System response is slow, I can understand how this benefits the drawing of blind children.

Table 6.13: Personal views of the participants

Table 6.13 shows participants' feedback, categorised based on the vision (No vision, some vision, sighted).

The next section details the outcome of study 2, answering the following questions. What has been learned from testing these systems? What needs to be changed as a result? What worked well, and what did not?

6.4.7 Findings of Study 02

1. Blind and partially sighted participants showed statistically non-significant results in instructed tasks of shape, image and diagram drawing which is similar to the claims made in a previous study by Kamel. He [Kamel and Landay, 2002] also confirms the results obtained in this study of no significant difference among Blind and partially sighted participants in the evaluation of the IC2D drawings but do not provide the reason for such difference. Blind and partially sighted participants managed to complete the drawing tasks without the help of a support worker with an average time of 2.22 (m:ss) which is lower than the time recorded in previous blind drawing research, for instance compared with an experiment conducted by [Kamel and Landay, 2002] where the time recorded was around 7 minutes, and an experiment by [Bornschein et al., 2018] where the time recorded was 19 minutes.

2. BVI and sighted participants did show statistically significant results in instructed tasks of shape, image and diagram drawing. This result demonstrates that the system is not only perceived well by BVI participants but also by sighted participants. This also informs us that sighted participants managed to complete screen navigation and graphics manipulation much faster than

the BVI group. We also learnt that sighted participants' feedback was more on the grounds of the higher mental load of not seeing the interface and were quite different from the BVI group's feedback. The evaluation of IC2D [Kamel and Landay, 2002], BPLOT4 [Fujiyoshi et al., 2018] and Pin-matrix system [Bornschein et al., 2018] also revealed different views and ranking of blind and sighted users similar to what is revealed in the this study. A previous study conducted by Kerr's [Kerr, 1983] with sighted and VI people's scanning times of visual imagery confirmed that spatial identification is achievable by blind individuals even though it takes longer than for their sighted counterparts as found in this studies without giving any particular reason. Hill [Hill et al., 1993] attributes this to retained memory of previous similar experience.

3. Blind, partially sighted and sighted participants showed statistically non-significant results in non-instructed tasks of the shape and image non-instructed drawing tasks. This informs us that the BVI group performed similarly to the sighted group when instructions were not given, demonstrating that better visualisation skills of the BVI group when the tasks are designed and led by themselves. This also demonstrates that highly imaginative skills of the BVI group as confirmed by [Lahav and Mioduser, 2002].

4. Blind and partially sighted participants showed statistically non-significant results in the shape and image recognition tasks. These results reveal that less complex, system-produced images are recognised without much effort, and that BVI users are fast with tactile image recognition as confirmed by Ungar [Ungar, 2000]. However, the tasks involving diagrams demonstrate statistically significant results among sighted and BVI groups, which informs us that the diagrams need much care with direction, letter placement, and overlapping lines. Therefore we learnt that the clarity of images is important in complex diagram presentation. It is discussed by Postma [Postma et al., 2007] that blind and sighted people could adopt different spatial coding strategies to explore different type of tasks that can impact on the time of the task.

5. Many blind and partially sighted participants agreed that enough time is needed for them to get familiar with the system so that they can operate the system more efficiently. Some participants didn't like the system's voice feedback feature as it slowed the operation of the system resulting from the processing commands. Hence some suggested including shorter audio feedback clips and auto corrective text.

6. Image description is not a feature of SETUP09 even though the system has the ability to label images. Some participants wanted to add a better description to an image rather than just a label. This idea will help in the review of the system grammar.

7. Some participants wanted an option that allowed them to choose the system's operating voice from a selection. However, voice production is out of the scope of this thesis. even though we find it useful to hear in order to review system grammar.

8. The system's speech feedback facility on user actions was received well by some participants, not so much by others. Some participants explained that they didn't really want conformation of all of their actions, whereas others needed more speech feedback. This feedback was used to review the system grammar.

9. Some of the participants were partially sighted, and therefore would have liked to have bigger lines and shapes. Hence, this feature was added into the system by incorporating magnification and line's outline thickness. This feature is addressed in the grammar review presented in the next chapter.

10. Some of the errors were caused as a result of limited input validation in the prototype, e.g. capitalisation of tokens. This type of correction needed front-end system validation rather than grammar change.

The above findings have led to a grammar and prototype revision of the system, which is discussed in Chapter 7.

6.4.8 limitation

This study aims to achieve it's hypotheses, (1) SETUP09 2D drawing technique is an efficient drawing technique to draw images for blind, partially sighted and sighted individuals (2) The SETUP09 2D drawing technique can accurately reproduce mental images of blind, partially sighted, and sighted individuals.

These two hypotheses are not designed to answer the question as to why differences exist among the blind, partially sighted and sighted individuals' performance ability and strategy. Such explanation warrants a biological investigation among cohorts which is not the purpose of this study. The above hypotheses are designed to test the system's ability among different groups of people. The focus is on the system, not the people. Thus further explanation is presented below for the factors that are too broad to control (time, accuracy, previous knowledge, age) but presented in this study that alines with other blind drawing studies in the field with reference and qualification of comparison and assertions made.

Qualification of the age difference - It may be questioned the impact on the time of two younger participants who were partially sighted. The performance-based on age is discussed in Section 6.4.5. The chart shows that participants who are age <= 10 took only 29 seconds extra on average to draw compared with other participants whereas 36 seconds extra on average to explore given images. Even if that differences are added to the partially sighted statistics the claims made are not changed, partially sighted group still performed better than blind group, remains as the second-best group, whereas with image recognition, partially sighted group took the longest. Therefore the age has insignificant baring in this study.

Qualification of blind and sighted people's prior experience of diagram recognition comparison - One shortfall of this study was blind participants' inexperience of diagram recognition.

Although the diagrams were designed using 2D shapes, participants did not have the same level of prior experience in diagram recognition. A task was needed to check if diagrams produced by SETUP09 is fit to be recognised in activities related to education. Section 6.4.4, image recognition data is re-examined to include the people who did not have prior knowledge of flowcharts, therefore who did not name the diagram type, but identified everything else correctly, now fall into 100% category. Therefore the new analysis contains a higher number of blind and partially sighted people (6 out of 10) recognising the flowchart diagram shapes with 100% accuracy without needing to name it. Even though participants were not able to identify the name of the image, they all did manage to identify the shape of the flowchart and acquired a high level of accuracy. Therefore it is clear that the name of the diagram had the least bearing in the performance. Further, there is no reason to believe that the results of diagram recognition activity are invalid because of the failure to recognise the diagram type. The speed (time) and accuracy (errors) recorded by the image recognition task using 2D shapes demonstrates a high success rate of recognition of 2D system-produced tactile images, confirming the H3 hypothesis.

Qualification of the time as a measure - Above paragraphs discuss the bearing of the age, and the prior experience on the measure of time. However, prior experience has more bearing than the age on time measure. The performance-based on prior experience of graphics creation is discussed in Section 6.4.5. The chart shows that participants who did not have prior experience with graphics performed 84 seconds slower than people without prior experience during the tasks. Nevertheless, not only among blind and partially sighted groups but there were participants in the sighted group who did not know the concept of diagrams. And the sighted cohort was blind-folded during image recognition activity even though some had preconceived memory of diagrams. The age and prior-experience do not invalidate the claim made on inferential statistics. The performance speed still confirms better performance with sighted group than partially sighted or blind group. Another time-on-task measurement was the type of keyboard used. Mac and both Windows keyboard were available during the experiment. Hence participants were not disadvantaged with getting used to new keyboards that would have impacted on the time-on-task measurement. But, we observed the variation of the speed of some participants in their keyboard use than the others. This data is not collected as it was not part of the study 2 hypotheses.

Qualification of accuracy as a measure - The absolute difference in images was easy to capture in the instructed drawing task, but it was far more challenging to capture in the non-instructed drawing task because no pre-drawn images were given. But recorded were blind users' verbal descriptions of their mental images, which were cross-checked after the experiment with the system-produced tactile images. Even though verbal descriptions were difficult to compare with tactile images in order to record absolute differences, participants' confirmation on the accuracy of their imaginary drawing was relied on. Blind participants verbal description is such where the shape or image are located, in what zoom level and what it supposed to look like. We checked if their tactile output picture matches the description they provided prior to the task and asked them to verify their system-produced tactile output. Participants were honest during the verification stage, and informed any deviation to the original idea. Even though some sighted people has the drawings recorded in a paper, there is no reason to believe that comparison accuracy is incorrectly observed with blind people, although there is a subjective element to it.

Qualification of blind and sighted people's recording method comparison of non-instructed drawing activity - It may be questioned why in the study that some sighted participants used paper drawings to inform their mental images as preparation for non-instructed drawing tasks, while blind participants had to verbally describe their mental images. Although the preparatory work was recorded with suitable and preferred methods, during the actual non-instructed drawing activity the sighted group had no access to those drawings. Therefore there is no reason to believe that the comparison between sighted and BVI groups is invalid, because of difference of their preparatory work, that was simply used for the purpose to record participants' mental images as verification for later. The speed (time) and accuracy (errors) recorded during the non-instructed drawing activity demonstrates the system' s ability to reproduce participants' mental images which is the hypothesis of this study, thus maintaining the system's claims.

Further testing of the system by participants who are not just blind or visually impaired but late blind with different input modalities would be beneficial, using Braille, speech and touch pad with participants. It was discussed in the literature review chapter that people with different mental models seek usability and satisfaction in many ways. The measures on ability, such as completion time and errors made in tasks, may therefore vary as a result. This means the experiment needs to be run with larger cohort of people before the results can be generalised.

The prototype needs further expansion, with much more grammar and a specific interface class to accommodate the usability aspect of the product. Grammar expansion and grammar implementation are some aspects to be considered for future work.

6.4.9 Conclusion

Computer-aided drawing techniques for blind people can meet the need for artwork and diagram drawing both in an educational setting and in life in general, with benefits that include non-reliance on a support worker, reduced interaction time and fewer errors. A unique approach for CADB systems was taken with the grammar controlled language used in the design of SETUP09.

Studies in this chapter revealed that system modification is needed for different groups of visually impaired people. The partially sighted group noted that they would benefit from thick line drawings and magnification, whereas participants with no vision needed additional usability features that could customise the speed, speech and feedback levels of the system. Even though participants with no vision and partial vision performed similarly in drawing and recognition tasks, participants with vision performed better in terms of time taken and were ahead of the BVI groups. It may be that the sighted participants used their prior life memories of 2D shapes, thus they needed better system features to correct system commands.

Studies in this chapter used an understanding of the activities found in education to evaluate system use with given hypotheses and thus find out areas for future improvements. The experiment results demonstrate that the technique gives an improved performance with time in art production compared with existing techniques [Kamel and Landay, 2002], [Bornschein et al., 2018] . The next chapter presents revisions to the SETUP09 model and non-grammar changes based on user feedback presented during the above experiments.

Chapter 7

SETUP09 Model Revision

The previous chapter provided details of the field studies to validate the SETUP09 model whereas this chapter proposes revisions to the grammar model and user-interface implementation. In this chapter, the grammar of the proposed SETUP09 model is refined based on user feedback from the previous stage and user testing. The original SETUP09 model now includes new and revised grammar, and revised is the Backus-Naur grammar which is presented in chapter 4. The changes in the grammar are categorised into space, shape, and usability revisions. Some users paid particular attention to ergonomic properties of the system interface as it determined their overall user experience during testing. Therefore not all changes are grammar-based. This chapter also presents non-grammar-based user-interface and system architecture changes. The refinement of the proposed SETUP09 model necessitated further comparison with other technologies (analogue and digital) which is discussed in chapter 8.

7.1 Grammar Model Revision

The revision of the grammar model is categorised into three sections, which are the components that build the SETUP09 model, i.e. space grammar revision, shape grammar revision, and usability grammar revision. Studies of the system under experimental conditions do not give full exposure to the model's capability. Hence long-term testing and wider audience testing with repeated measures are required to place certain usability measures into proper context, such as memorability, ease of use, learnability, and many other usability matrices.

ISO standard 9241-11 [ISO, 2011] explains the measures of usability under three headings: efficiency, effectiveness, and satisfaction. According to Jenny Preece and Helen Sharp [Preece et al., 2015], commonly used usability evaluation criteria are: time taken to complete a task (efficiency), time taken to learn a task over time (learnability), and the number of errors made over time (memorability). Unless the experiment is repeated with regular users, a few times, it is difficult to accurately measure memorability and learnability.

In this thesis, effectiveness is demonstrated as the number of errors made over the number of trials (completion rate); efficiency is measured by time taken to complete a task, and also in comparison with other systems and methods; and data relating to user satisfaction and feedback is presented. The experiment findings conducted under the above matrix lead to the revision of the grammar model presented below and other non-grammar based changes. A higher number of usability grammar changes were encountered than simply space or shape grammar revisions as a result of system testing.

7.1.1 Usability Grammar Revision

1. Some participants suggested the system should have a facility to add an alternative text attribute to images, so that the system remembers and informs the full description of an image rather than simply the name of the shape. This has led to a new grammar rule for alternative text introduction as stated below.

   - An alternative text to describe a drawing:

$$\begin{aligned} \langle \text{text} \rangle &\models \text{'text'}, \langle \text{name} \rangle, \langle \text{string} \rangle \\ \langle \text{string} \rangle &\models \langle \text{characters} \rangle \end{aligned}$$

2. There was a suggestion to improve the system by changing the system voice, whereby the user could pick one from a selection available or even supply a voice they like. However, voice production is out of the scope of this thesis, even though we find it useful to hear the feedback. The grammar below introduces a new voice command. The available voice commands are part of the help command function.

   - A grammar rule to select the preferred voice for system interaction:

$$\begin{aligned} \langle \text{voice} \rangle &\models \text{'voice'}, \langle \text{voices} \rangle \\ \langle \text{voices} \rangle &\models \text{manUK} \mid \text{womenUK} \mid \text{manUSA} \mid \text{womenUSA} \end{aligned}$$

3. There was a suggestion to implement different levels of system feedback. The grammar below is a new feature, a level that limits the automatic system feedback on user action. In level L the system's default feedback confirms every user action. Level M is where only the drawing is confirmed, not navigation actions. Level H is where the drawing is confirmed when asked using the "des" grammar command rather than any form of ongoing drawing conformation.

   - A grammar rule to select the level of system interaction:

$$\begin{aligned} \langle \text{level} \rangle &\models \text{'level'}, \langle \text{feedback-level} \rangle \\ \langle \text{feedback-level} \rangle &\models \text{L} \mid \text{M} \mid \text{H} \end{aligned}$$

4. There was a suggestion to implement bigger text and lines with different thicknesses for people with low vision. Hence, a grammar was introduced where a user selects user-type (B - blind, PS - partially sighted, S - sighted) at the start of the system and the system changes the size of text and thickness of lines accordingly.

   - A grammar rule to select the user type for interaction:

$$
\begin{aligned}
\langle \text{user} \rangle &\models \text{'user', } \langle \text{users} \rangle \\
\langle \text{users} \rangle &\models \text{B} \;|\; \text{PS} \;|\; \text{S}
\end{aligned}
$$

5. Interface magnification was requested by partially sighted individuals. Hence, a grammar was incorporated that enables magnification of the screen based on its location and zoom level. A location non-terminal was introduced in Chapter 4, that shows screen compass locations. A plus sign with the location name is introduced to magnify locations at a fixed rate.

   - A grammar rule to magnify a screen area based on the location and level:

$$
\langle \text{mag} \rangle \models \text{'+', } \langle \text{location} \rangle
$$

6. Another piece of feedback was to remove grids before printing the image for a tactile feel. This change can take place in the "print" concrete class in the interpreter. In terms of grammar, the "print" class was revised by removing the grids on a picture before printing. Therefore no changes to the actual command were needed.

   - A command to remove grids on the image is same as print grammar:

$$
\langle \text{print} \rangle \models \text{'print'}
$$

7.1.2 Shape Grammar Revision

1. As an alternative to the "define" command, introduced was a "group" command where a user can draw individual objects first and then group them by giving them a name label. As the terminal "name" was introduced in Chapter 4 only presented below are "group" and non-terminals "names".

   - An alternative text to define:

$$
\begin{aligned}
\langle \text{group} \rangle &\models \text{'group', } \langle \text{names} \rangle \\
\langle \text{names} \rangle &\models \langle \text{name} \rangle \;|\; \langle \text{name} \rangle \langle \text{names} \rangle
\end{aligned}
$$

2. Further shapes can be added to the grammar as required by potential system users. But for this thesis, the prototype is limited to 2D shapes as presented in the revised model below.

7.1.3 Space Grammar Revision

1. For speed of screen navigation, system hotkeys were suggested as an alternative to the system grammar "zoomin" and "zoomout" as shown below. The grammar is introduced as an alternative to the original, as user suggestions were based on keyboard input modality. The original grammar rules are retained to accommodate other modalities, such as speech and Braille.

   New non-terminals are introduced to produce alternative keys to the grammar model. The "zoomin-key" non-terminal /arrow keys are for *north, south, east, west* locations whereas "alt-key" for *centre* location. The "space-key" and "zoomin-key" used together denote *north-west, north-east, south-east, south-west* locations. The command "zoomout" can also be represented by *backslash character.*

   - An alternative text to zoomin:

$$\begin{aligned}\langle\text{zoomin-alt}\rangle &\models \langle\text{zoomin-key}\rangle \mid \langle\text{alt-key}\rangle \mid \langle\text{space-key}\rangle, \langle\text{zoomin-key}\rangle\\ \langle\text{zoomin-key}\rangle &\models \textit{left arrow} \mid \textit{right arrow} \mid \textit{up arrow} \mid \textit{dowm arrow} \mid\\ \langle\text{alt-key}\rangle &\models \textit{alter key}\\ \langle\text{space-key}\rangle &\models \textit{space key}\end{aligned}$$

   - An alternative text to zoomout:

$$\begin{aligned}\langle\text{zoomout-alt}\rangle &\models \langle\text{zoomout-key}\rangle\\ \langle\text{zoomout-key}\rangle &\models \textit{backslash character}\end{aligned}$$

7.1.4 Revised Model

This section demonstrates the revised context-free grammar of the space, shape and usability language; its terminals and non-terminals written in Backus-Naur form. The above discussion on space, shape and usability grammar changes are now integrated with the SETUP09 model (see Chapter 4) and highlighted in red colour. More terminals were introduced to accommodate non-terminals in the new context-free grammar model which are shown below.

⟨statementlist⟩	⊨	⟨abstractstatement⟩ | ⟨abstractstatement⟩⟨statementlist⟩
⟨abstractstatement⟩	⊨	⟨call⟩ | ⟨define⟩ | ⟨zoomin⟩ | ⟨zoomout⟩ | ⟨assign⟩ | ⟨write⟩ | ⟨line⟩ | ⟨lines⟩ | ⟨curve⟩ | ⟨curves⟩ | ⟨triangle⟩ | ⟨square⟩ | ⟨circle⟩ | ⟨oval⟩ | ⟨process⟩ | ⟨arrow⟩ | ⟨rec⟩ | ⟨save⟩ | ⟨open⟩ | ⟨print⟩ | ⟨erase⟩ | ⟨route⟩ | ⟨position⟩ | ⟨describe⟩ | ⟨help⟩ | ⟨undo⟩ | ⟨redo⟩ | ⟨copy⟩ | ⟨paste⟩ | ⟨speed⟩ | ⟨mode⟩ | ⟨nullstatement⟩
⟨define⟩	⊨	'define', ⟨name⟩, ⟨Statementlist⟩
⟨call⟩	⊨	'call', ⟨name⟩
⟨group⟩	⊨	'group', ⟨names⟩
⟨text⟩	⊨	'text', ⟨name⟩, ⟨string⟩
⟨zoomin⟩	⊨	'zoomin', ⟨location⟩
⟨zoomin-alt⟩	⊨	⟨zoomin-key⟩ | ⟨alt-key⟩ | ⟨space-key⟩, ⟨zoomin-key⟩
⟨zoomout⟩	⊨	'zoomout'
⟨zoomout-alt⟩	⊨	⟨zoomout-key⟩
⟨assign⟩	⊨	⟨variable⟩ '=' ⟨point⟩
⟨write⟩	⊨	'write', ⟨point⟩ | ⟨variable⟩, ⟨string⟩
⟨line⟩	⊨	'line', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩
⟨lines⟩	⊨	'line',⟨point⟩ | ⟨variable⟩,⟨points⟩ | ⟨variables⟩
⟨curve⟩	⊨	'curve', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩
⟨curves⟩	⊨	'curves', ⟨curve⟩ | ⟨curve⟩⟨curves⟩
⟨wave⟩	⊨	'wave', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩

⟨triangle⟩ ⊨ 'triangle', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩

⟨square⟩ ⊨ 'square', ⟨point⟩ | ⟨variable⟩

⟨circle⟩ ⊨ 'circle', ⟨point⟩ | ⟨variable⟩

⟨oval⟩ ⊨ 'oval', ⟨point⟩ | ⟨variable⟩

⟨process⟩ ⊨ 'process', ⟨point⟩ | ⟨variable⟩

⟨rec⟩ ⊨ 'rec', ⟨point⟩ | ⟨variable⟩

⟨arrow⟩ ⊨ 'arrow', ⟨point⟩ | ⟨variable⟩, ⟨point⟩ | ⟨variable⟩

⟨erase⟩ ⊨ 'erase'

⟨save⟩ ⊨ 'save', ⟨string⟩

⟨savescript⟩ ⊨ 'savescript', ⟨string⟩

⟨print⟩ ⊨ 'print'

⟨open⟩ ⊨ 'open', ⟨string⟩

⟨exit⟩ ⊨ 'exit'

⟨des⟩ ⊨ 'des'

⟨help⟩ ⊨ 'help'

⟨redo⟩ ⊨ 'redo'

⟨undo⟩ ⊨ 'undo'

⟨copy⟩ ⊨ 'copy', ⟨name⟩

⟨paste⟩ ⊨ 'paste', ⟨name⟩

⟨mag⟩ ⊨ '+', ⟨location⟩

⟨speed⟩ ⊨ 'speed', ⟨speeds⟩

⟨level⟩ ⊨ 'level', ⟨feedback-level⟩

⟨mode⟩ ⊨ 'mode', ⟨modes⟩

⟨voice⟩ ⊨ 'voice', ⟨voices⟩

⟨user⟩ ⊨ 'user', ⟨users⟩

$\langle$variables$\rangle$ $\models$ $\langle$variable$\rangle$ | $\langle$variable$\rangle\langle$variables$\rangle$

$\langle$variable$\rangle$ $\models$ $\langle$string$\rangle$

$\langle$names$\rangle$ $\models$ $\langle$name$\rangle$ | $\langle$name$\rangle\langle$names$\rangle$

$\langle$name$\rangle$ $\models$ $\langle$string$\rangle$

$\langle$string$\rangle$ $\models$ $\langle$characters$\rangle$

$\langle$characters$\rangle$ $\models$ $\langle$character$\rangle$ | $\langle$character$\rangle\langle$characters$\rangle$

$\langle$character$\rangle$ $\models$ *any ASCII character*

$\langle$speeds$\rangle$ $\models$ L | M | H

$\langle$modes$\rangle$ $\models$ B | S | K

$\langle$voices$\rangle$ $\models$ manUK | womenUK | manUSA | womenUSA

$\langle$users$\rangle$ $\models$ B | PS | S

$\langle$feedback-level$\rangle$ $\models$ L | M | H

$\langle$zoomout-key$\rangle$ $\models$ *backslash character*

$\langle$location$\rangle$ $\models$ N | S | E | W | NE | NW | SE | SW | C

$\langle$points$\rangle$ $\models$ $\langle$point$\rangle$ | $\langle$point$\rangle\langle$points$\rangle$

$\langle$point$\rangle$ $\models$ N | S | E | W | NE | NW | SE | SW | C

$\langle$zoomin-key$\rangle$ $\models$ *left arrow* | *right arrow* | *up arrow* | *down arrow* |

$\langle$alt-key$\rangle$ $\models$ *alter key*

$\langle$space-key$\rangle$ $\models$ *space key*

$\langle$nullstatement$\rangle$ $\models$ λ

7.2 Non-Grammar Revision

Some user suggestions are based on interface features, system input and output preferences that are not necessarily grammar-based modifications. This section discusses non-grammar based changes proposed or observed during system testing.

1. Some of the errors were due to use of capitalisation when it is not required. Therefore, a functionality equalsIgnoreCase() was used to change the interpreter to parse non-case sensitive tokens.

```
if (s.equalsIgnoreCase("zoomin")) {
return new Token(16, "zoomin");
```

2. Some participants suggested that letter (Braille/alphabet) placement in the tactile image should be kept away from the actual image lines and curves. The system command "write" needs to take place closer to the shape but not obstruct the shape. For such a revision, the system needs to check the surrounding object placement and place the writing without obstructing lines. This check needs to take place in the "write" concrete class.

3. An external tactile grid-pad for real time feedback was suggested. This can represent the image on the screen for feedback conformation rather than the voice feedback mechanism. A touch sensitive tactile display on a grid layout is suitable for such a development.

4. Error correction was repeatedly asked for various participants. The system provides error detection but not correction. Keyboard is only one input mechanism for the proposed architecture. If speech input is used, the correction is based on a speech input signal that is different from the method of keyboard error correction. One method of serving the request is by introducing an error correction class into the system architecture, which corrects user input with the system accepted syntax and also provides speech feedback with correction.

7.2.1 Revised Architecture

In order to implement an automatic error correction facility, as mentioned above, a new component named "error correction" was introduced into the SETUP09 component model, that is closely associated with the input validation sub-component as shown in Figure 7.1.

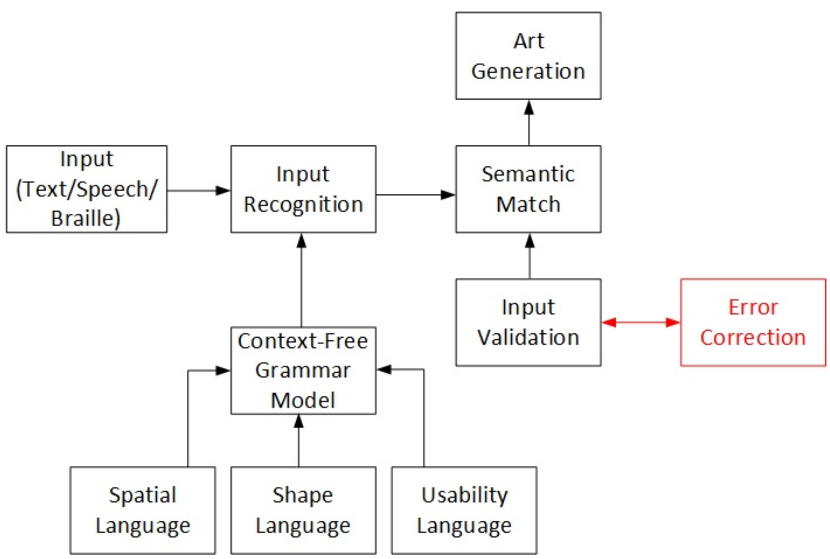


Figure 7.1: Components of the revised SETUP09 Model

7.3 Conclusion

This chapter presented revisions to the space, shape and usability grammar of the SETUP09 model, along with non-grammar and architectural changes. These revisions are presented as a result of observations and feedback from the user experiments presented in Chapter 6. Some revisions are presented as an alternative to the original grammar, and other rules are newly introduced. This thesis has presented the initial evaluation of the SETUP09 model in Chapter 6 and then its revision in this chapter whereas the next chapter presents the SETUP09 system in comparison with other analogue and digital systems and their techniques.

Chapter 8

Comparative Evaluation of SETUP09 Model Techniques

The proposed SETUP09 model has already gone through system user validation and grammar refinement. The previous chapter presented a revised SETUP09 grammar model. Provided is a comparison of actual experimental data with a statistical analysis as a method of validation. The uniqueness of this chapter is that there were no comparative evaluations of two digital systems done previously in literature even though there were systems evaluated on its own. Study one presents a detailed comparison of the effectiveness of two methods for blind people with German film papers and the SETUP09 system model. Study two compares SETUP09 with the IC2D digital system. In both cases the experimental conditions were identical. From study one it was found that SETUP09 performed with higher quality outputs accurately compared to German film paper. Whereas from study two it was discovered that SETUP09 performed with higher efficiency in terms of time and better accuracy compared to IC2D digital system. The chapter ends by concluding the findings of comparative studies which completes the work in this thesis. Any further exhaustive system testing of usability measures are necessary but beyond the scope of this thesis. The next chapter presents the guidelines of the proposed SETUP09 model, its design and development stages to facilitate extension of the work by other researchers and designers in the assistive technology domain.

8.1 Study 3: SETUP09 comparison with Analogue Drawing Technique

8.1.1 Hypothesis

- Hypothesis (H1): A compass-based graphics creation method is an effective drawing method for creating shapes compared with a tactile graphics creation method.

Rationale: The literature findings established that raised-line kit is a popular analogue method for creating drawings by blind users, which uses a special kind of paper to draw on (such as drawing film) that becomes raised when pressure is applied to it by pen or stylus. This technique is widely used in schools for blind art production. It is therefore important to assess the effectiveness of the proposed digital drawing system (SETUP09) against a tactile graphics creation tool such as a Raised-line kit.

- Hypothesis (H2): A compass-based graphics creation method is an effective drawing method among both early and late blind individuals

Rationale: The literature findings established that early and late blind individuals perceive images and spatial information differently. Late blind individuals work with their pre-blindness memories using mixed methods, while early blind individuals (who were blind before three years of age) use different calculation techniques. It is therefore important to assess the effectiveness of the compass-based graphics creation method of SETUP09 against early and late blind individuals.

8.1.2 Procedure

Eight blind participants were employed in the experiment described here. Four of them were late blind individuals, who lost their sight at ages ranging from 17 to 56 years, and the other four participants were blind from birth, details of which are given in Table 8.1. There were three male participants and five female participants. Six participants identified themselves as highly computer literate. Six of them were graduates, one a college leaver and one a manager. Each participant was presented with three tasks to complete and their performance was measured. Each participant was given about 30 minutes of training on the system. The training was split across three experimental tasks and included: an introduction to SETUP09 and its drawing and navigation language; familiarity with SETUP09; hot keys and help keys; hands-on practice using the prototype and different drawing commands; steps to draw simple shapes and images, to draw in different zoom levels, and locations; and labelling and defining an image.

Participants were able to read the screen labels and seek system help with orientations if required. Audio feedback was provided with help commands. Very few tasks were misunderstood they could retake them.

Suitability of the command-driven drawing technique and virtual navigation system was evaluated with early and late blind individuals. An earlier study of SETUP09's navigation revealed that blind computer users were able to successfully navigate to screen locations and draw without the help of a support worker [Fernando and Ohene-Djan, 2020]. This study experiments with both drawing and navigation methods with early and late blind individuals. Participants were recruited via contacting different charities supporting both early and late blind people. Participants were selected from different age groups and different education levels.

This experiment presents a systematic evaluation of analogue methods that are commonly used in special needs educational establishments versus the proposed digital method (SETUP09). As discussed

Person	Gender	Age	Reason	Residual Vision	Age of Blindness	Education	Computer Literacy
1	F	37	Microphthalmia	None	Birth	Manager	High
2	M	70	Microphthalmia	None	Birth	Graduate	High
3	F	56	Optic nerve damage	None	Birth	Graduate	Moderate
4	M	29	Retinopathy	None	Birth	Graduate	High
5	F	65	Retinitis Coloboma	None	17	Graduate	High
6	F	65	Retinitis	None	17	Graduate	High
7	F	68	Glaucoma	None	22	Graduate	Low
8	M	57	Ocular Atrophy	None	56	College Leaver	High

Table 8.1: Information on the Early and Late Blind Participants in the Experiment

in Chapter 3, SETUP09 is not just a computer drawing system but also a command-to-diagram conversion technique that has compass-based grid naming and multi-cell referencing. The focus of this work does not explore modality mechanisms but a system that is adaptable to different modality mechanisms if expansion is needed.

Introduction to an Analogue and Digital System

Most BVI students and practitioners are in the habit of using tactile maps to recognise raised-line art or objects [Takagi, 2009]. The analogue toolkit consists of a rubber mat as a backing sheet for making the raised-line drawing, embossing film papers and a pen instead of an embossing tool. A light inked pen for the experiment was used here to get a visual effect for readers. The toolkit gives a negative raised image. Figure 8.1 is a picture of a rubber mat, embossing film paper and a drawing tool.

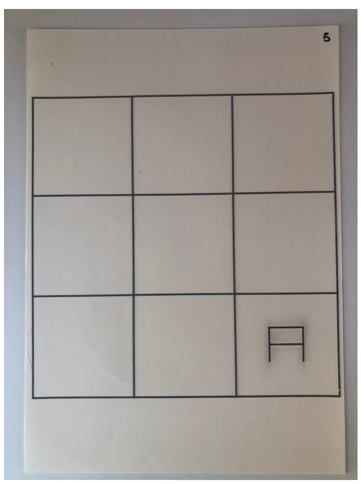
Figure 8.1: Rubber Mat, Embossing Film Papers[RNIB, 2019]

Some adjustments were made to the analogue toolkit to enhance the sense of location and size experienced by participants when drawing on the film paper. To achieve this the film papers were embossed with 3 x 3 grids to provide guidance on the location and size. Finding locations and sizes otherwise would have been a time-consuming task using the analogue toolkit. Some participants needed further grid marks and point marks on the film paper to replicate an exact image for the tasks given below. It was found that too many grids can confuse blind participants' conception of the tactile view.

The SETUP09 system prototype was used on both Mac and Windows operating systems. A Zyfuse Heater with swell paper was used to produce tactile images. To ensure higher accuracy in tactile image recognition, simple images were created with thick lines and sufficient space between shapes. The text was produced in Braille letters and the English alphabet as required. Swell paper was used during the experiment to print haptic images for image recognition tasks.

Tasks

Participants were tested with three designed tasks. No image was given during Task 1 but they were asked to draw a four-sided shape on the screen using any nine cardinal/compass reference points. Table 8.2 illustrates the images given to participants for their second and third drawing tasks. It shows an outline picture of a 2D table and a flowchart with Braille text.



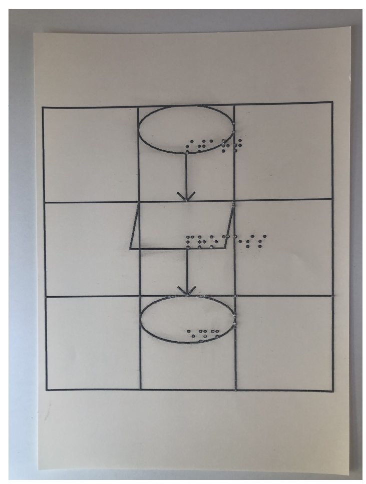


Table 8.2: Task 02: Table and Task 03: Flowchart

8.1.3 Results

Participants were tested individually, and experiments were conducted in different locations at different times. The results of the experiment tasks were evaluated in terms of accuracy, errors made and observations. The achievement of a given task was used to measure the accuracy by the participants. The errors were counted which was based on the number of times a mistake was made on the interface as a result of forgetting commands or incorrect mental perception to what the task was required.

The time spent on tasks was also recorded. It was realised that time was an imperfect indication of achievement as not all attempts by participants were correct.

A percentage score was given for the actual drawings to measure accuracy. For example, 90% accuracy was given when participants produced the given/intended image with minor errors such as drawing with distorted lines, incomplete shapes, or unwritten letters. 80% was given for incorrect sizes, shapes and moderated levels of the incompleteness of lines and shapes.

During Task 1, the non-instructed production of a 2D shape, individuals spoke about the intended shape, location and size of their mental model prior to the activity being recorded. The same shape was reproduced using the SETUP09 system. The IBM usability questionnaire [Lewis, 1993] with a seven-point scale was used to assess participants' perception of system suitability, ease of use and cognition.

Ungar [Ungar, 2000], in his writing, elaborates on different coping strategies used by sighted, blind and late blind individuals during different spatial tasks, therefore we expected to observe different strategies during this experiment . [Postma et al., 2007] verifies that late blind individuals' strength with the verbal response and the role of visual experience depends on the task aspect of the experiment, therefore we expected better performance by late blinds individuals. We also believe that late blind participants were advantaged against completely blind participants due to their pre-blindness memory of shapes, which blind individuals did not have. Not all participants were familiar with the diagram recognition task due to not having prior knowledge of scientific diagrams (e.g. flowcharts, DFDs).

The time was recorded for all three tasks carried out by the participants using a stopwatch. Data was collected from eight blind participants. Three blocks of two trials (paper drawing and system drawing) from eight participants (3 x 2 x 8) resulted in a total of 48 trials, which were recorded with accuracy, number of errors and other observations. Two attempts had to be redone due to confusion in interpreting the instructions, and these were requested by the participants themselves. Another attempt had to be repeated as an incorrect time was recorded. There was no difference in performance by gender. It was observed that some participants were more thorough when following instructions than others.

Output Comparison: H1 and H2

Table 8.3 shows the images produced by a late blind participant of a table and a flowchart. The images produced by the late blind participant are not as clear as the images produced by the early blind individual. The shapes were not maintained, and the lines were considerably distorted. The images were produced on a paper toolkit for task two and three. The same images produced using SETUP09 by late blind individuals shown in Table 8.4 have minor variations that have an impact on the accuracy, which is discussed in the next section.

The quality of output shown in Table 8.3 (paper kit drawing) and Table 8.4 (SETUP09 drawing) supports hypothesis H2 (a compass-based graphics creation method of SETUP09 is an effective drawing method among early and late blind individuals). This is discussed in the next section with statistical data. Overall, there is no significant difference in the output produced by early and late blind individuals, except that early blind individuals took slightly longer than late blind individuals. However the effectiveness of the systems were tested with accuracy rather than time. The images shown in 8.4 demonstrates that output is produced in line with H1 (a compass-based graphics creation method of SETUP09 is an effective drawing method compared with a tactile graphics creation method for creating shapes) which is further discussed with statistics in the following sections.

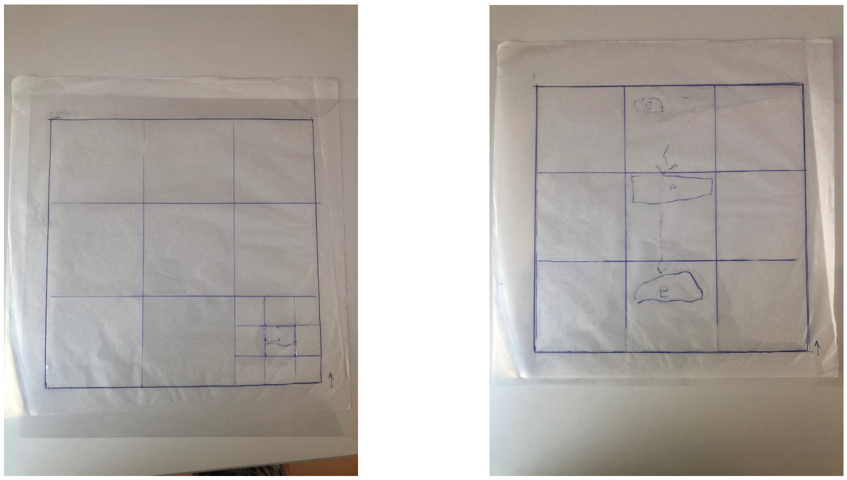


Table 8.3: Tasks 2 and 3: Images Produced Using a Paper Kit by Late Blind Person

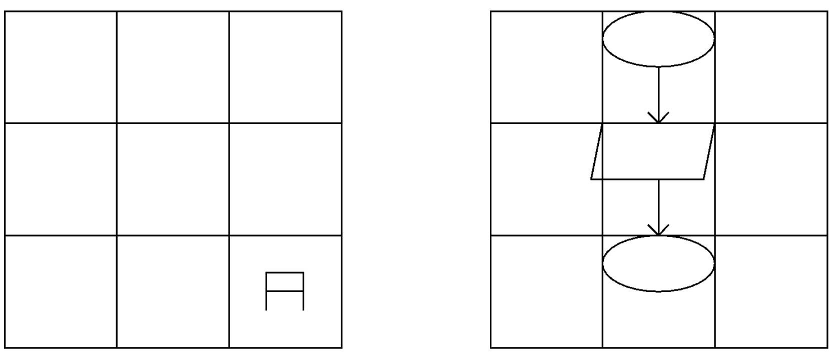
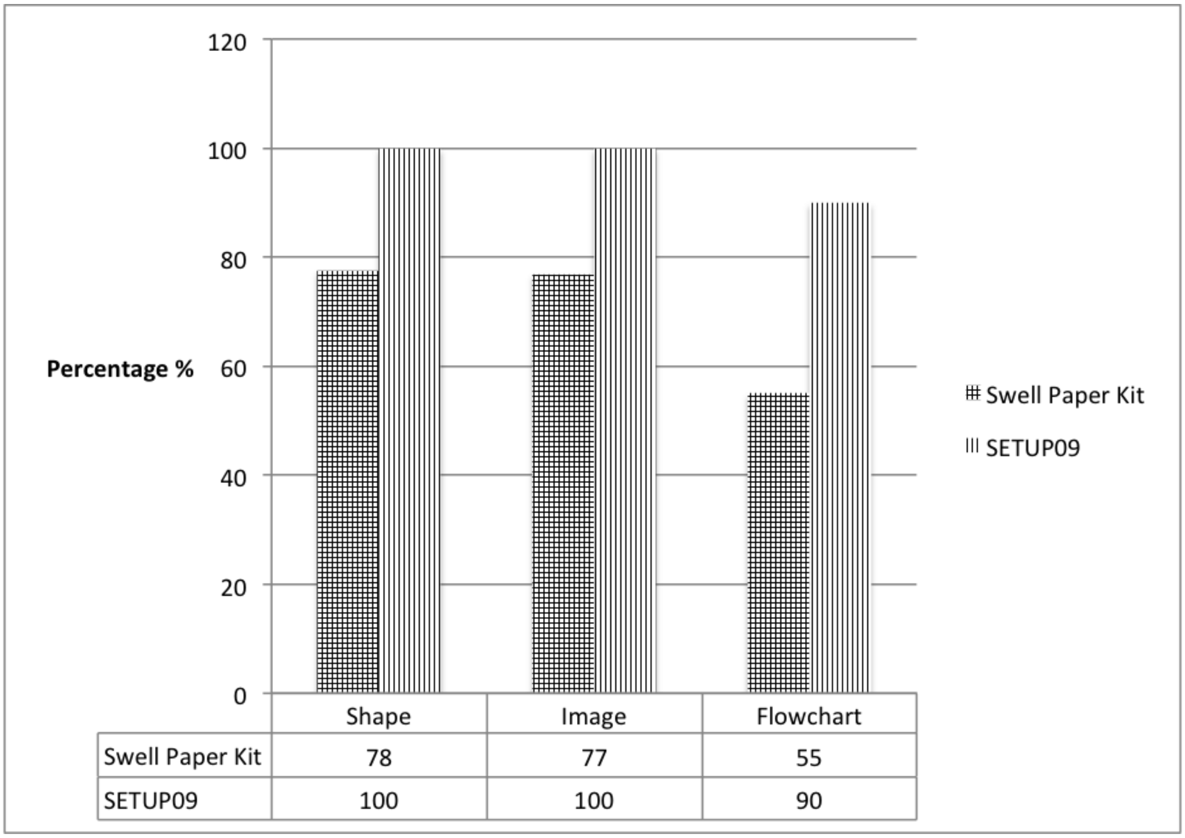


Table 8.4: Tasks 2 and 3: Image in SETUP09 by Late Blind Person

Accuracy

Participants were given a score percentage based on output produced. 90% for minor accuracy errors such as drawing with distorted lines, non-completed shapes, or unwritten letters. 80% for incorrect sizes, shapes and moderated levels of the incompleteness of lines and shapes. Poor scores such as 20% were given when the image did not make any sense or was placed in the wrong location but contained some shapes.

The results of the tasks presented in Figure 8.2 shows that blind individuals managed to complete



	Shape	Image	Flowchart
Swell Paper Kit	78	77	55
SETUP09	100	100	90

Figure 8.2: Overall Accuracy of Tasks by Both Groups

non-instructed shape drawing, image (table) drawing and flowchart drawing in the SETUP09 system with an average accuracy of 97% and with the standard deviation of six. The bar charts in Figures 8.3 and 8.4 shows the accuracy between early and late blind individuals in both paper kit drawing and SETUP09 drawing. The accuracy with SETUP09 system is 100% for shape and image production, but the accuracy is reduced to 90% for chart production by both early and late blind individuals. However, a low level of accuracy was recorded with paper toolkit drawing where early blind participants had an average accuracy of 62% and late blind participants were 78% accurate. The results confirm that the SETUP09 technique can be successfully utilised by both early and late blind groups, which validates hypothesis H2, that a compass-based graphics creation method is an effective drawing method among early and late blind individuals.

A few minor issues were scoring slightly low than the above results. However, with the swell paper kit there was a considerable difference in average accuracy, which was 70%, and standard deviation was 13. As shown in the bar chart in Figure 8.4 the early and late blind people performed equally well with the SETUP09 system tasks (shape and image), recording an accuracy of 100%. However, the late blind group performed slightly better than the early blind group in the paper toolkit drawing as shown in 8.3, but this was not statistically significant ($P = 0.214$).

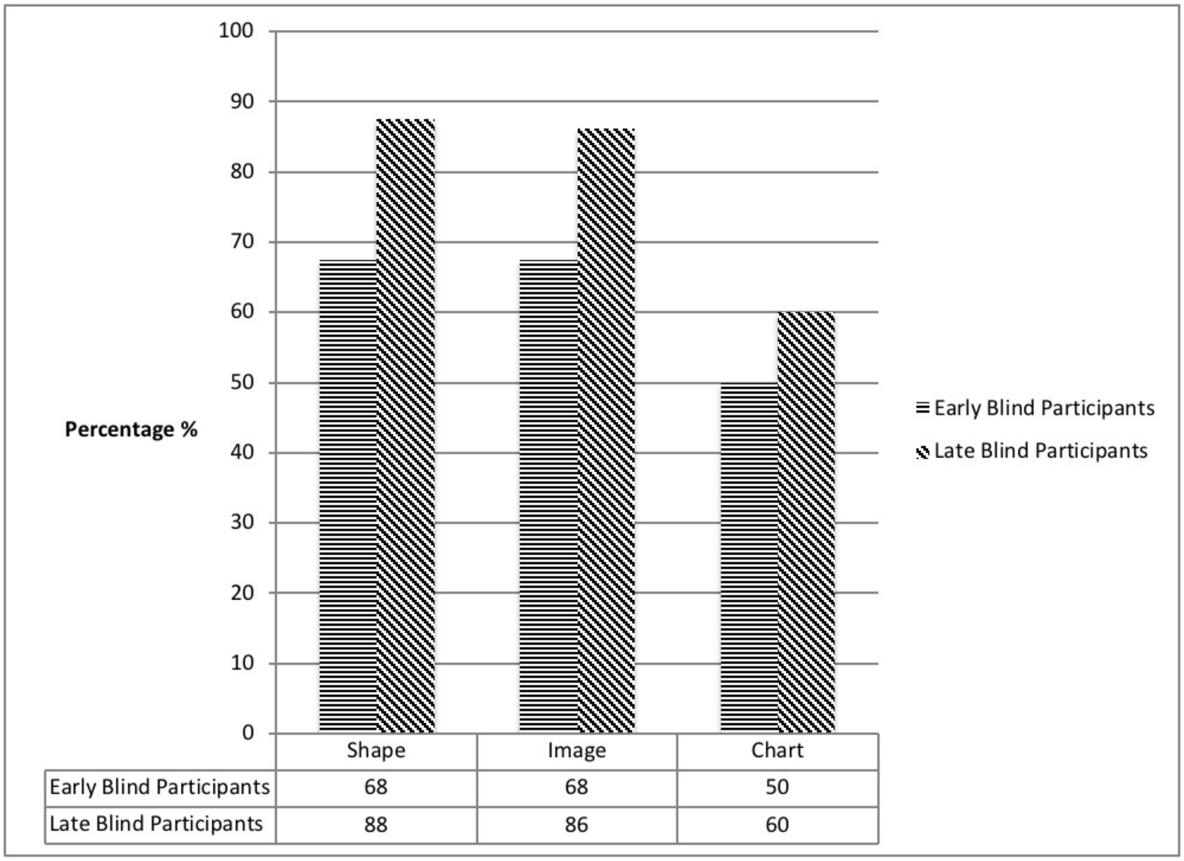


Figure 8.3: Accuracy of Paper Toolkit Tasks

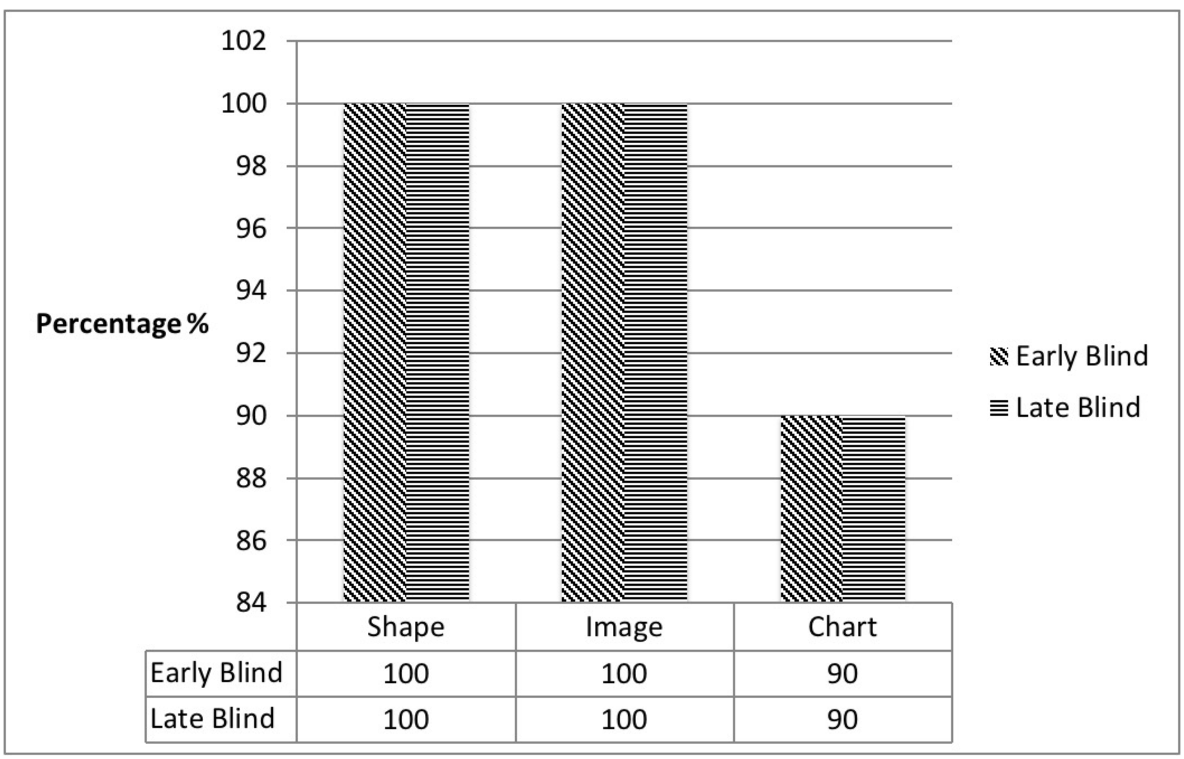


Figure 8.4: Accuracy of SETUP09 Tasks

The above results highlight the difference between the paper kit and the SETUP09 system with early and late blind groups. The paper images might not have been successful at all if paper grids were not provided to support participants with landmarks and sizes. The difference in image accuracy between the two methods was statistically significant, with the P-value of 0.050 proving hypothesis H1 that a compass-based graphics creation method is an effective drawing method compared with a tactile graphics creation method for creating shapes.

Errors

The errors during the experiments were recorded, such as incorrect shape, wrong location, wrong size and incomplete images. The bar chart in Figure 8.5 shows errors made using SETUP09 drawing tasks. Two errors were recorded with shape drawing; five errors were recorded with image drawing; and 12 errors were recorded with flowchart drawing. As shown in the Figure 8.6 bar chart, 17 errors overall were recorded when using the SETUP09 system and 54 errors with paper drawing. The late blind individuals made 35 errors overall during the experiment, and early blind participants made 36 errors. The SETUP09 system errors were corrected, but errors made using the paper kits were all visible and couldn't be corrected, which had an impact on the accuracy data.

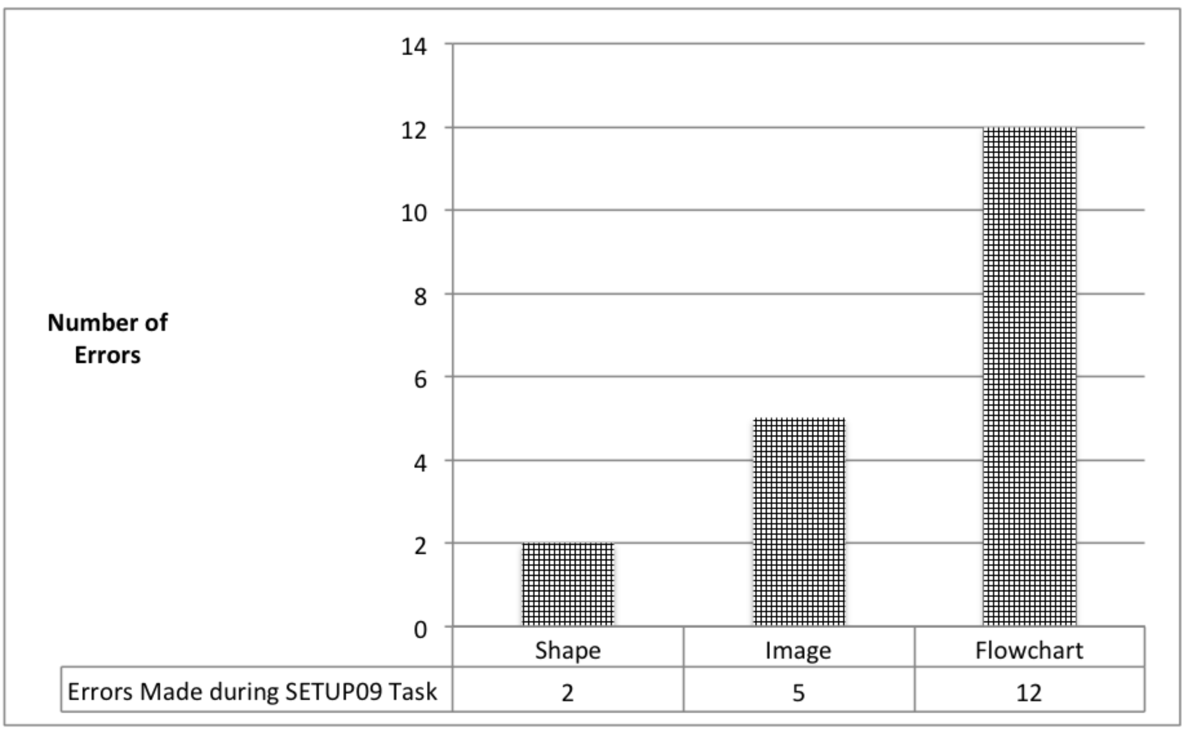


Figure 8.5: Errors in SETUP09 Tasks

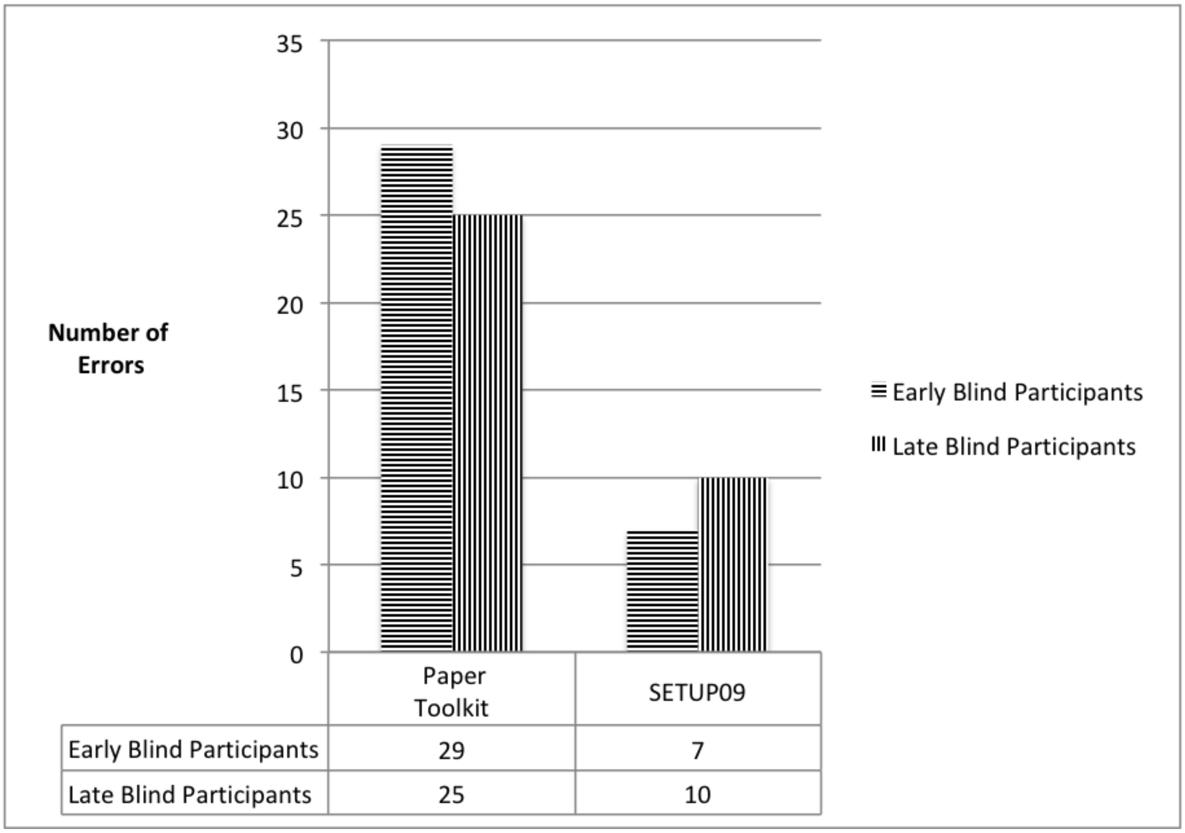


Figure 8.6: Comparison of Errors Between Early Blind Participants with Paper Toolkit and SETUP09 System

Participants were only given 30 minutes of system training before the commencement of the test. Therefore, the errors made were mainly due to forgetting commands and lack of prior system knowledge. However, errors could be rectified using system commands such as *erase*, *help*, *position*, which allowed participants to achieve a successful outcome (97%) as discussed in the above Accuracy section.

Errors made during paper drawing activities were clearly visible on film papers, but unfortunately the participants were not aware of their errors because the nature of errors was different from SETUP09 errors. It was not about forgetting system commands but about the incorrect reproduction of the intended drawing. The shapes, images and flowcharts produced on the paper were mostly incomplete with distorted lines, incorrect shapes, incorrect sizes and variation in locations. Some participants were lost without sufficient landmarks such as start and endpoints of lines and found drawing flowcharts very difficult. Some participants knew that their drawings were not correct but did not know how to correct them. They clearly needed help from a sighted person. Participants who produced incorrect images and mentioned “This is what I think the image is in my mind" The mistakes were obvious to the observer which were clearly correlated with the accuracy of the task as the error correction mechanism was not available with paper toolkit system, so errors were left uncorrected.

Time

Performance was measured on time taken by participants on examining shapes, locations and sizes. The bar chart in Figure 8.7 shows the average time taken by both groups to produce drawings in the tasks using SETUP09 and paper. Bar charts in Figures 8.8 and 8.9 show the average time taken to produce drawings using SETUP09 and paper drawings by early and late blind individuals.

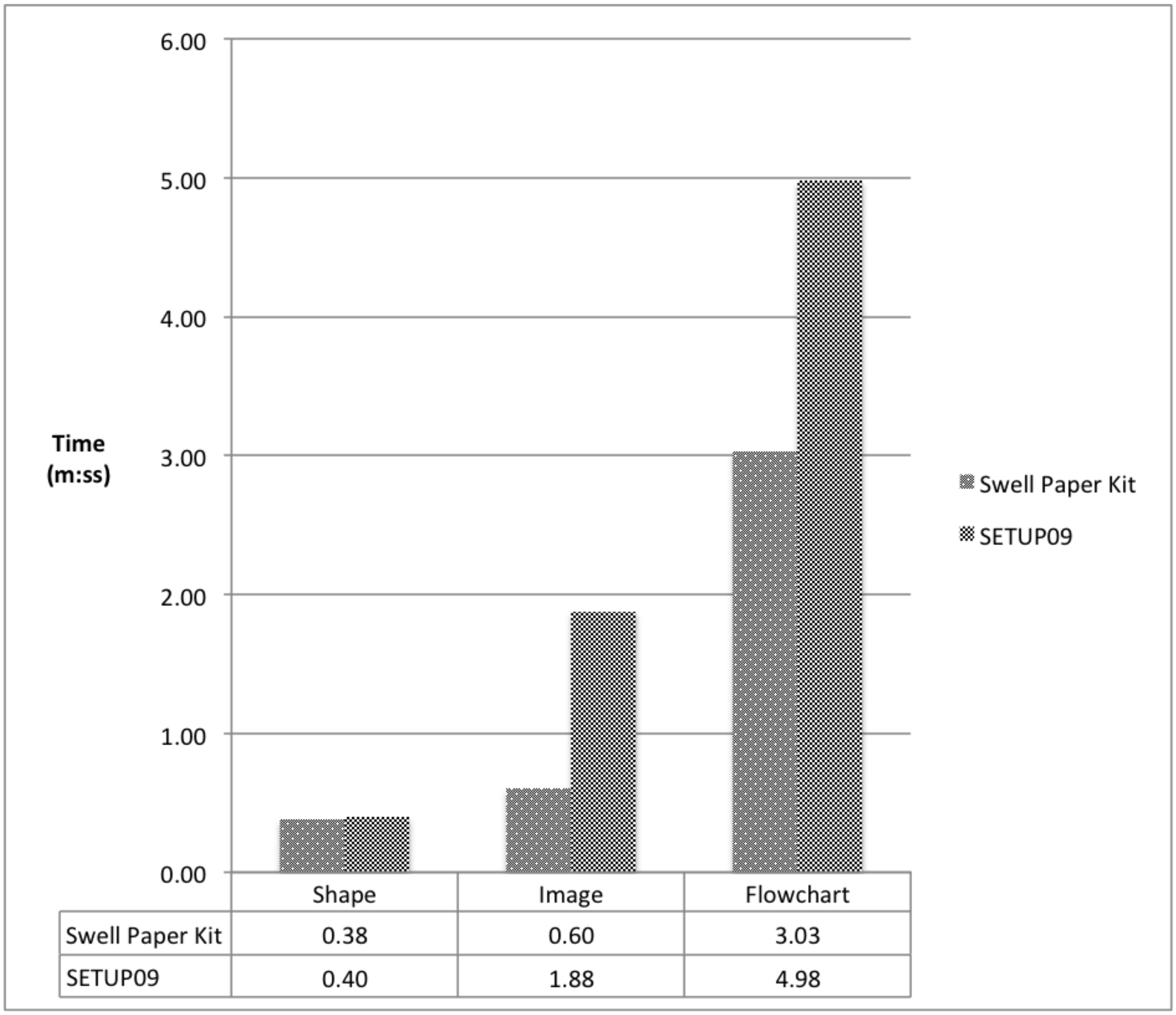
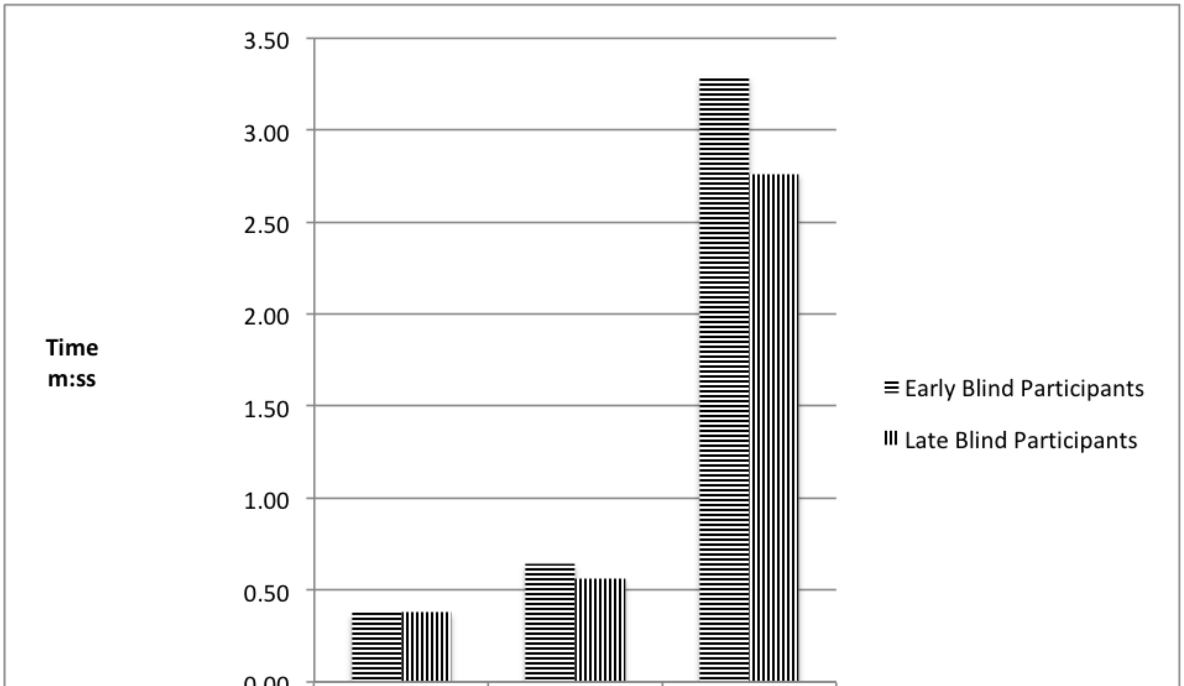
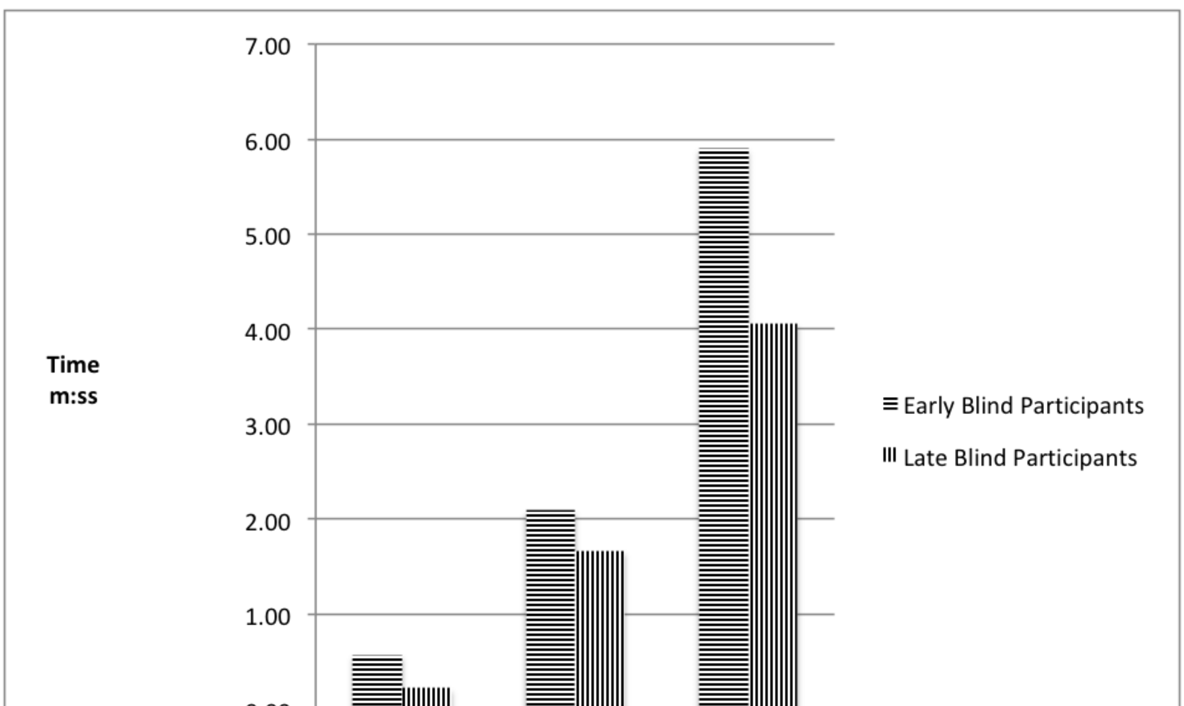


Figure 8.7: Performance Time of both groups with Paper Toolkit and SETUP09



	Shape	Image	Chart
Early Blind Participants	0.38	0.64	3.29
Late Blind Participants	0.38	0.56	2.76

Figure 8.8: Performance time of Late and Early Blind Participants with Paper Toolkit



	Shape	Image	Chart
Early Blind Participants	0.57	2.09	5.90
Late Blind Participants	0.22	1.67	4.06

Figure 8.9: Performance time of Late and Early Blind Participants with SETUP09 System

The SETUP09 system recorded a more extended time in all three drawing tasks than the swell paper kit. For both groups of participants the mean time taken to draw a shape, an image and a flowchart using SETUP09 was 2.42 (m:ss). The mean time of the same activities using paper drawing was recorded as 1.34 (m:ss). The P-value recorded was P = 0.541, and the difference was not statistically significant even though participants took a shorter amount of time in drawing with the paper kit. Comparison of the results presented in this section reveal that the late blind participants completed both the paper kit drawing tasks with P = 0.875 and the SETUP09 drawing tasks with P= 0.678, which was faster than early blind individuals even though there is no significant difference.

It is clear that both groups of participants completed the paper drawing activity faster when using paper toolkit than SETUP09. Not only had the new system's operational time and learning memory impacted the outcome in terms of time taken to complete a task, but also extra time was added to account for error correction and system help. The valid question to ask is whether the extra time with SETUP09 is acceptable in the successful completion of tasks. Participants' views were collected at the end to analyse the validity of extra time when using SETUP09.

Even though the paper drawing activities were completed faster, their accuracy was poor. It was found that the paper activity was incorrectly completed, and in some cases the participants acknowledged the errors made but were unable to correct them, and in other cases participants were not aware of their mistakes.

Post-experiment Survey

The post-task questionnaire for SETUP09 reports Cronbach's alpha value as 0.90 of scale reliability with acceptable internal consistency, mean = 2 and SD = 0.99. All participants who completed all three tasks successfully also completed post-task questionnaire. Level 1 signifies agreeing strongly, and Level 7 signifies disagreeing strongly with questions asked. Five questions were posed at the end of the three tasks. All participants were asked to give ratings for five different qualitative measures of the SETUP09 system as set out in Figure 8.10. Six out of eight participants agreed or strongly agreed that the SETUP09 technique is more effective, easier to use and builds a better navigation model in the participant's mind than the film paper kit.

Participants believed that the SETUP09 system is more effective as they managed to produce more accurate images (similar to the original images) using the SETUP09 system than using a film paper kit. Seven out of eight participants agreed or strongly agreed that the SETUP09 technique is more supportive than a film paper kit and the SETUP09 technique builds a better layout model in the participant's mind than a film paper kit.

1. (Q1) The SETUP09 technique is more effective than film paper kit.
2. (Q2) The SETUP09 technique is more easy to use than a film paper kit.
3. (Q3) The SETUP09 technique is supportive than film paper kit.

4. (Q4) The SETUP09 technique builds a better navigation model in the participant's mind than a film paper kit.

5. (Q5) The SETUP09 technique builds a better layout model in the participant's mind than a film paper kit.

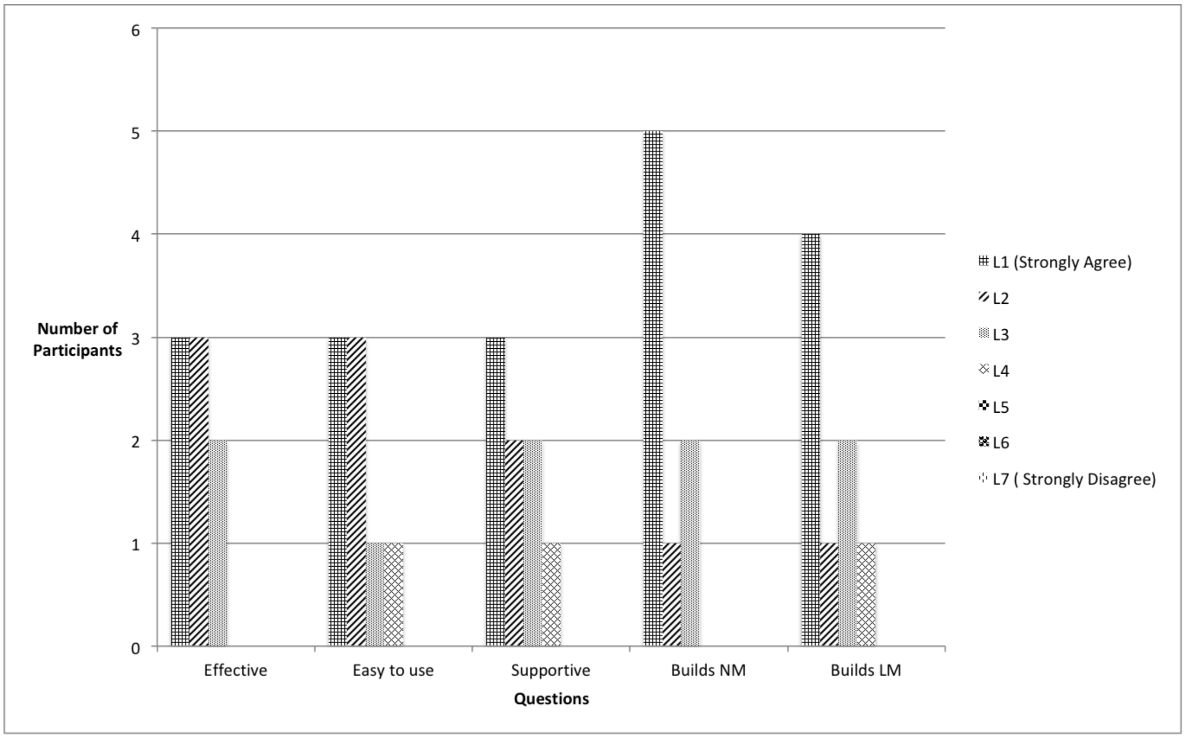


Figure 8.10: Post Study Survey

Participant	vision	Efficiency	Easy to Use	Supportive	Builds Navigation Model	Builds Layout Model
P1	Early Blind	2	1	2	1	1
P2	Early Blind	3	2	2	3	3
P3	Early Blind	1	1	1	1	1
P4	Early Blind	1	1	1	1	2
P5	Late Blind	3	3	4	2	4
P6	Late Blind	1	2	1	1	1
P7	Late Blind	2	4	3	3	3
P8	Late Blind	2	2	3	1	1

Table 8.5: Survey Results of the Experiment

As shown in the bar chart in Figure 8.10, levels one and two were predominantly selected by early blind participants. Participants suggested features such as auto text correction, redo and undo options and better system feedback for error correction as some potential improvements. Five out of eight participants strongly agreed that the SETUP09 technique builds a good navigational model in participants' minds whereas five out of eight participants agreed that the technique builds a good layout model in participants' minds by picking Levels 1 and 2 of the Likert scale. Participants repeatedly selected 1 to 4 of the Likert scale showing high positivity and suggestions for improvement. However, the majority of participants thought that the system could be more supportive in terms of error detection, correction, input and output functionalities.

The data in Table 8.5 demonstrates that participants who were early blind predominantly selected Levels 1 and 2 of the Likert scale, whereas late blind participant thought the prototype could be further improved to accommodate late blindness, therefore they predominantly picked Levels 2, 3 and 4. The P-value of early and late blind individual groups is 0.015, which is $P<0.05$ and demonstrates that there is a difference between the post-experiment feedbacks of the two groups (early blind versus late blind participants). Some participants didn't like smaller text, text with less space and Braille text, and the multitude of grid lines confused them during the recognition of shapes stage. Many participants agreed that enough time is needed for system familiarisation, which is the key to using SETUP09 with confidence and efficiency.

8.1.4 Findings

- There was no significant difference found in artwork production among early and late blind individuals, except early blind individuals took a slightly longer time than late blind individuals ($0.214 = P>0.05$). However, participants (incorrectly) completed most of their paper kit drawing much faster than their SETUP09 drawing (Paper kit = 1.34, SETUP09 = 2.42 m:ss). This finding informs that the time measurement in this experiment has little value in proving the hypothesis because participants incorrectly completed the paper kit activity faster than SETUP09 system commands.

- There was no difference found between accuracy (completion of given shape) among late and early blind individuals, but there was a considerable difference in the accuracy of SETUP09 completion, which recorded 97% accuracy compared with the paper tool kit which had an accuracy of 70%. This finding informs that early and late blind participants performed equally well despite the retain memory of late blind individuals.

- The errors made by late and early blind individuals were similar (late blind individuals = 35, early blind individuals = 36), whereas errors made using the SETUP09 system and paper drawing kit show a considerable difference (SETUP09 = 17, paper drawing kit = 54). This finding informs that even though it is impossible to make perfect shapes using a drawing kit, it is much better to use computer-aided drawing functionalities to obtain accurate shapes and completeness of images with less number of inaccuracies.

- It was noticed that participants were unable to correct their errors during paper toolkit drawing, and in some cases they didn't even realise they were making mistakes, whereas SETUP09 errors were related to forgetting commands and using incorrect command which was attributed to limited system training time. However, with SETUP09 participants were able to correct these errors due to the inbuilt feedback and error detection features. This finding informs that features such as system feedback, help and error detection elements play a significant role in the accuracy of digital drawing tasks.

8.1.5 Limitations

- H1 and H2 hypotheses were tested by comparing SETUP09 with a well-established paper drawing toolkit. There is a clear difference between the modalities of paper drawing and SETUP09, where paper drawing involves hand drawing and SETUP09 involves command insertion. Digital drawing experiments conducted in the literature have always been compared with manual drawing kits because of the absence of similar digital tools. The comparison done here is therefore consistent with previous studies and does not invalidate the results presented here that show the strength of the SETUP09 system as an efficient tool among available and established manual tools. However, SETUP09 will be tested with another digital drawing tool in the next Section.

- Measurement of the accuracy of paper toolkit drawing against SETUP09 system drawing is difficult as the nature of errors were different (as discussed above). Some of these errors can also be made by sighted individuals using a paper drawing kit. Although H1 and H2 are not testing blind people with sighted people's ability to draw using a manual mechanism, but the hypotheses test the strength of SETUP09 as a technique to produce artwork amongst other available well-known techniques.

8.1.6 Conclusion

An experiment was conducted to compare and evaluate the effectiveness of the SETUP09 drawing method with a tactile graphics creation method for creating shapes by early blind and late blind participants. The comparison was done between the two different technologies because of the absence of an established digital drawing and navigation tool for BVI users. Compared to the tactile graphics creation method the SETUP09 system with its navigation and drawing techniques was evaluated to be highly effective among early and late blind participants. Although the analogue system was faster to use, but SETUP09 was far more accurate. Also studied was the confidence and familiarity with using technology among different individuals. Overall, the results confirmed that the SETUP09 2D drawing and navigation technique is reliable and effective and facilitates the reproduction of a given artwork task better than a film paper drawing kit. The next section presents a pilot study of different, digital cell referencing techniques followed by a comparison of such digital tools.

8.2 Study 4: Pilot study of Cell Referencing Systems

8.2.1 Hypothesis

- H3: The multiple (cardinal) point cell referencing technique used in SETUP09 is a more efficient method than centre point cell referencing.

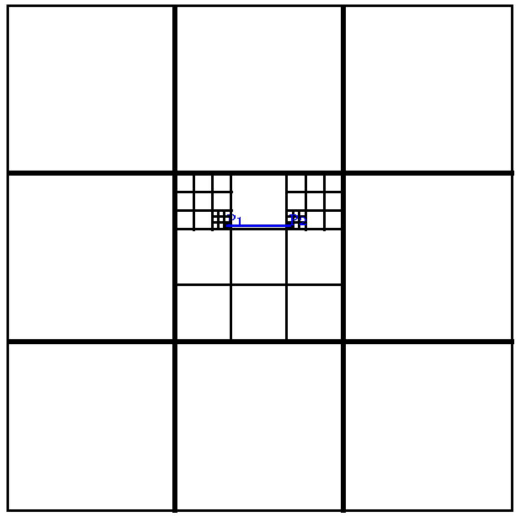
Rationale: The literature findings established that current grid-based systems are designed with centre point cell referencing or using x, y coordinates referenced to recognise a single point in two-dimensional space. However, compass-based graphics creation methods have nine points of reference in a cell. The hypothesis tests whether drawing using centre point cell referencing has a different effort level than using multiple points of reference in a cell.

8.2.2 Procedure

The hypothesis H3, tests whether drawing using centre point cell referencing has a different effort level than using multiple points of reference in a cell. In study 4, participants were asked to draw a line between two points, as illustrated in Figure 8.11. Both output lines in Figure 8.11 look the same, but there is a considerable difference in the effort level based on the technique/referencing system as illustrated in Table 8.6.

Figure 8.11: Task 04: Tactile Images of Lines Used in the Experiment

8.2.3 Results



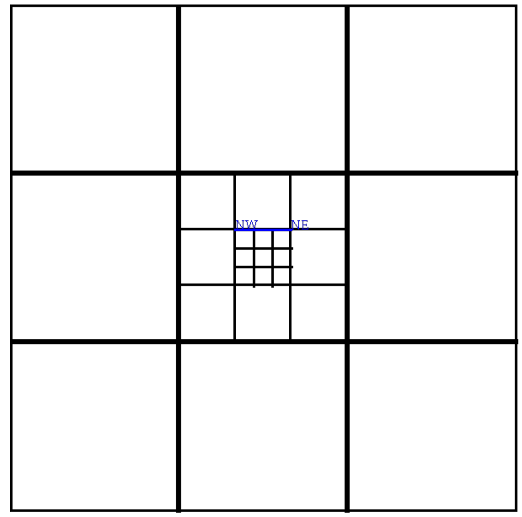


Table 8.6: SETUP09 System Images of Task 04 Created by a Participant: Lines with a Single Point Cell Reference and a Multi-points Cell Referencing Technique

Figure Table 8.6 demonstrates two different user interfaces created by a participants; A line with a single point cell reference needs many commands: *C, NW, SE, SE, assign P1 C, zoomout, zoomout, zoomout, NE, SW, SW, assign P2 C, line P1 P2* whereas multi-point cell referencing, the referencing mechanism of the SETUP09 technique, uses far fewer commands, *C, C, line NW NE* . Task 4 (line drawing) was only tested with two participants, one early and one late blind individual. The reason for fewer participants was that they had great difficulty in finding and visualising the exact location to reproduce the same line using the centre only referencing system. This is clearly evidenced in the results, without many trials needing to be run.

The participants were asked to draw a line on the screen as illustrated in Figure 8.11. One line at the centre of the cell with a start and end points, and another line using any of nine points by using SETUP09 commands: *E*, W, S, N, NW, NE, SE, SW, C. Table 8.6 shows the navigation path of the centre only cell reference. The user had to visualise the possible centre point and its associated cell to start and end the line. The user also had to go through multiple *zoomin* and *zoomout* commands to find a specific centre point, which was less effective than multi-point cells. Finding the centre cell reference point was time-consuming and cognitively demanding according to observation and feedback from the participants. There was a considerable time gap between the two referencing techniques in doing Task 4.

- Commands for centre point cell referencing: *C, NW, SE, SE, assign P1 C, zoomout, zoomout, zoomout, NE, SW, SW, assign P2 C, line P1 P2*
- Commands for multi-point cell referencing: *C, C, line NW NE*

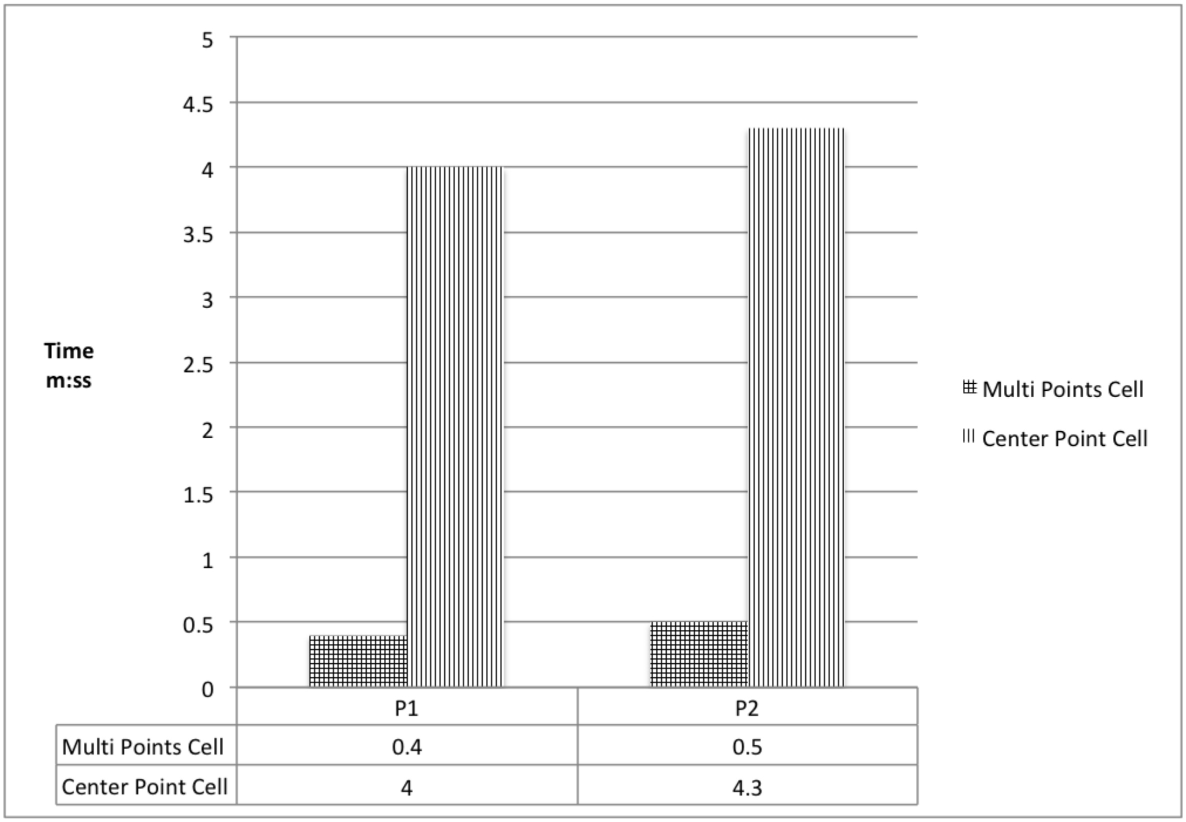


Figure 8.12: Multiple Points of Cell versus Centre Point Cell Referencing

The bar chart in Figure 8.12 shows the difference in the two main referencing styles. The average time of the multiple (cardinal) point cell referencing technique in SETUP09 is 0.45 (m:ss) and centre point cell referencing is 4.15 (m:ss). Hence this pilot study confirms H3, that the multiple (cardinal) point cell referencing technique in SETUP09 is a more efficient method than centre point cell referencing. The next study takes the digital system comparison further and presents actual drawing tasks and findings by comparing previously presented IC2D results with SETUP09 drawing system results.

8.3 Study 5: SETUP09 Comparison with Digital Drawing Technique

8.3.1 Hypothesis

- H4: The SETUP09 technique is a more efficient method to create artwork than the existing IC2D technique.

Rationale: It was found from literature finding that IC2D [Kamel and Landay, 2002] is a well-established digital technique using a cell referencing system that enabling users to create artwork with 2D shapes based on a keyboard input modality. Several digital systems have been introduced recently, such as BPLOT3 [Fujiyoshi et al., 2014], Multimodal Pin-matrix system [Bornschein et al., 2018]. These digital systems use haptic drawing with the support of computer functionalities and are not keyboard or command driven and the purpose and modality of those systems are different. Therefore most valid and closest comparison that can be made to benchmark SETUP09 is with the IC2D art production technique.

The usefulness of the IC2D [1] product [Kamel and Landay, 2002] was originally investigated with blind and sighted participants for: instructed drawing, non-instructed drawing, graphics exploration and 3D image production. H4 hypothesis compares the drawing capability of the well-established technology, i.e. IC2D, with the new technology, i.e. SETUP09. In this study for the purpose of comparing the two techniques under identical experimental conditions I have asked the participants (blind and sighted) to create a specified image based on instructed and non-instructed mode. The purpose of this comparison is to mimic the functionality of another digital system and compare them under experimental conditions.

8.3.2 Procedure

The experiment in the study presented here was designed and conducted during the social isolation period in the year 2020 due to Covid-19 global pandemic [2] . This was done with the approval of both the university's research ethics committee and the participants. To minimise the risk of catching Covid-19, the experiment was conducted on an individual basis with the permission of the participants at their residence. The participant observed social distancing, wore personal protection equipment (face mask), used hand sanitiser whenever appropriate, and all surfaces we were in contact with were cleaned with antibacterial wipes including the computer/laptop. At the location of the experiment the only people present were the participant and the observer.

Kamel tested the IC2D system with blind and sight participants; hence the proposed SETUP09 software was tested by 4 blind and 4 sighted people. The experiment involved the participant in drawing a simple piece of artwork/diagram on a computer/laptop using SETUP09 software with the given instructions. The observer took several HCI measurements for analysis. The parameters recorded during the study included:

- Task time
- Participants confidence
- Performance rating

Blind participants for the study were recruited through the Beyond Sight Loss Charity. The blind participants were predominantly diagnosed with Microphthalmia, Coloboma and Optic nerve damage. All participants (blind and sighted) had an average age of 50 years and standard deviation of 15.04. Their ages ranged between 30 to 71 years stated in Table 8.8. Instructions for the experiment were

[1] https://www.slideserve.com/dillian/constructing-images-eyes-free-a-grid-based-dynamic-drawing-tool-for-the-blind
[2] https://www.gov.uk/government/publications/covid-19-guidance-on-social-distancing-and-for-vulnerable-people/guidance-on-social-distancing-for-everyone-in-the-uk-and-protecting-older-people-and-vulnerable-adults

circulated prior to the study so that they were familiar with what was required from them. Participants were explained how to take control of their computer/laptop screen, and the keyboard commands they would be using in the study. Prior to the study commencing each participant was given roughly 30 minutes to practice using the keyboard commands for SETUP09 and get familiar with the software. In the study the eight participants were given at least three trials. In fact, there were in total 24 (8 x 3 = 24) trials recorded. At the end of the experiment, participants' feedback was taken for further analysis.

Tasks 1 and 2 given in Table 8.7 were carried out by the participants using object placement technique of SETUP09. This involved the placing the objects on top of gridlines. For task 1 participants were asked to draw three circles in three different grid positions of their own choosing and then draw the smallest triangle possible in the bottom right cell of the grid. They were then asked to draw a rectangle inside the top left cell, and finally draw a line connecting the top vertex of the triangle with the top left vertex of the rectangle in task 2. For comparison purposes, data of this task using IC2D was extracted from the results reported in [Kamel and Landay, 2002].

There is a difference in 2D shape placement in IC2D and SETUP09. IC2D has a palette of 2D objects for users to select shapes such as circles, squares and triangles. In IC2D the objects are placed at the centre point of the grid whereas SETUP09 uses the grid outline, as shown in 8.13. Also, SETUP09 has object placement functionality for grid outlines and centre point. In one task the participants had to reproduce the square inside the gridline, and another task involved the square on top of gridlines. In SETUP90 there are more commands for finding grid points inside the cell than on the gridlines.

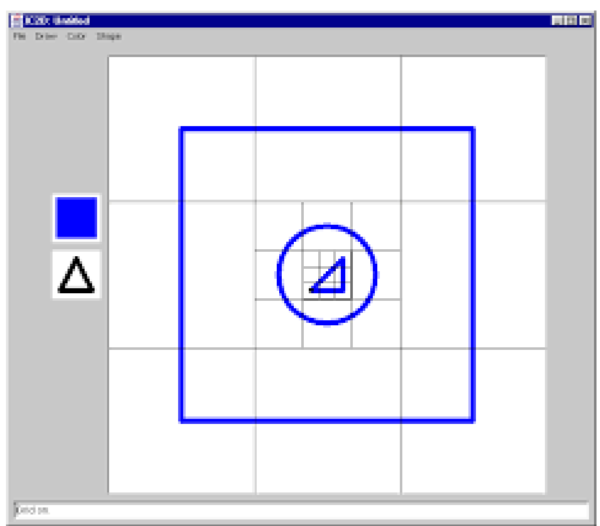


Figure 8.13: IC2D Objects Placement

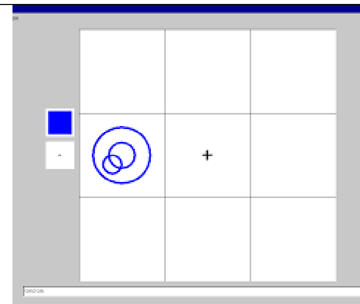
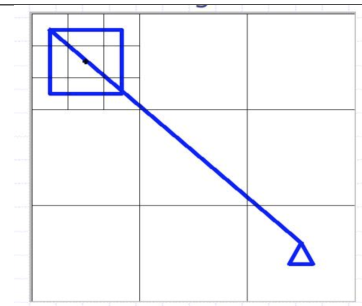
Task	Output of IC2D
Task 1 - Participants were asked to draw three circles in three different grid positions of their own choosing. After drawing each circle, we asked them to verbally describe the location they had attempted to draw it for later comparison with the actual drawing. The goal of this part of the task was to see if participants could describe their own non-instructed drawings	
Task 2 - Participants were asked to draw the smallest triangle possible in the bottom right cell of the grid, then draw a rectangle inside the top left cell, and finally draw a line connecting the top vertex of the triangle with the top left vertex of the rectangle. The goal of this part was to see if participants could carry out instructed drawing with precision.	

Table 8.7: IC2D Outputs

Person	Gender	Age	Reason	Residual Vision	Age of Blindness	Education	Computer Literacy
1	M	71	Microphthalmia	None	Birth	Graduate	High
2	F	66	Coloboma	None	Birth	Graduate	High
3	F	38	Microphthalmia	None	Birth	Manager	High
4	F	57	Optic nerve damage	None	Birth	Graduate	Moderate
5	F	37	-	Sighted	-	Graduate	High
6	F	36	-	Sighted	-	Graduate	High
7	F	52	-	Sighted	-	Graduate	Low
8	M	56	-	Sighted	-	Graduate	High

Table 8.8: Information on Blind and Sighted Participants

8.3.3 Results

The outcome of all three study tasks, namely the non-instructed drawing, and the two instructed drawings were analyzed by examining the variation in the participant performance in the two groups (blind and sighted). The time recorded to complete the task by the participants included time taken for correcting any errors. Participant's confidence was assessed by asking them specific questions at the end of the task. The questions were designed to elicit their ability to accomplish the tasks, judgment of accuracy, or likelihood of success. Their performance was also assessed by the observer on the accuracy of the output produced. Because the study was conducted during a global pandemic in 2020, the participant size was limited to 8. Although the sample size for the IC2D study was double that for SETUP09 this does not invalidate the results of the study as the purpose of the study was to determine trend.

Table 8.9 on 8.9 show the images produced by a blind participant using IC2D in the left column as reported in [Kamel and Landay, 2002] and on the right column the output produced using SETUP09. Two different methods and instructions are employed for creating the drawing using IC2D. One method uses object placement on the edge of the grid line and the other uses gridline. The reason for this is because IC2D object placement is designed to place the object away from the gridline whereas SETUP09 object is designed to place objects on the grid line.

Compared to IC2D the proposed SETUP09 software uses more navigational command to precisely place the object in the required location. As a consequence, task 2 (method 2) has taken the participants more time to complete than task 2 (method 1) as is evident in Figures 8.14 and 8.15.

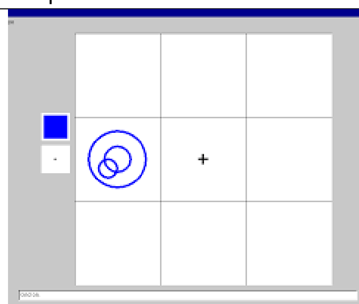
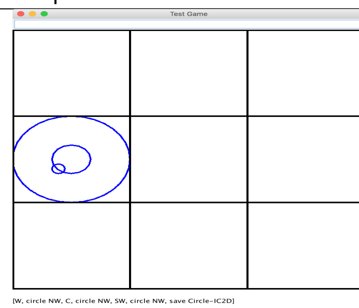
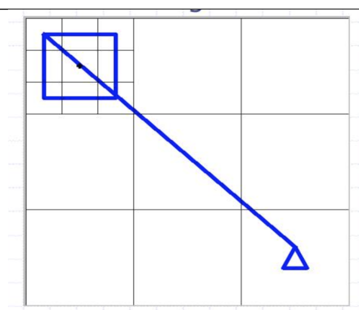
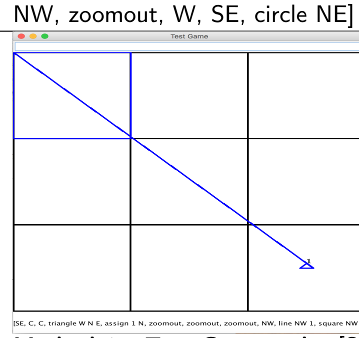
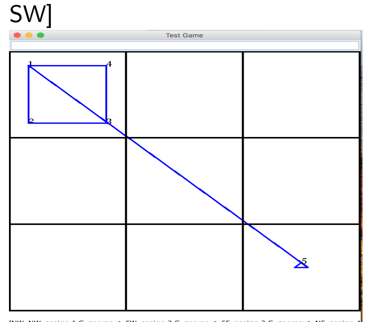
Task	Output of IC2D	Output of SETUP09
Task 1	 [Using a palette of shapes]	 **Text Commands:** [W, circle NW, C, circle NW, zoomout, W, SE, circle NE]
Task 2	 [Using a palette of shapes]	 **Method 1 - Text Commands:** [SE, C, C, triangle W N E, assign 1 N, zoomout, zoomout, zoomout, NW, line 1 NW, square NW NE SE SW]  **Method 2 - Text Commands:** [NW, NW, assign 1 C, zoomout, NE, assign 2 C, zoomout, SE, assign 3 C, zoomout, SW, assign 4 C, , square 1 2 3 4, zoomout, zoomout, SE, C, C, triangle W N E, assign 5 N, line 1 5]

Table 8.9: IC2D Published Drawing and SETUP09 Drawing Outputs Produced by Blind Participants

This part presents the results of the study on hypothesis H4: The SETUP09 technique is a more efficient method to create artwork than the existing IC2D technique. Each participant's performance was measured based on the time taken to complete the given task and accuracy of output. Also recorded was the participant's self-confidence level at completing the task using SETUP09.

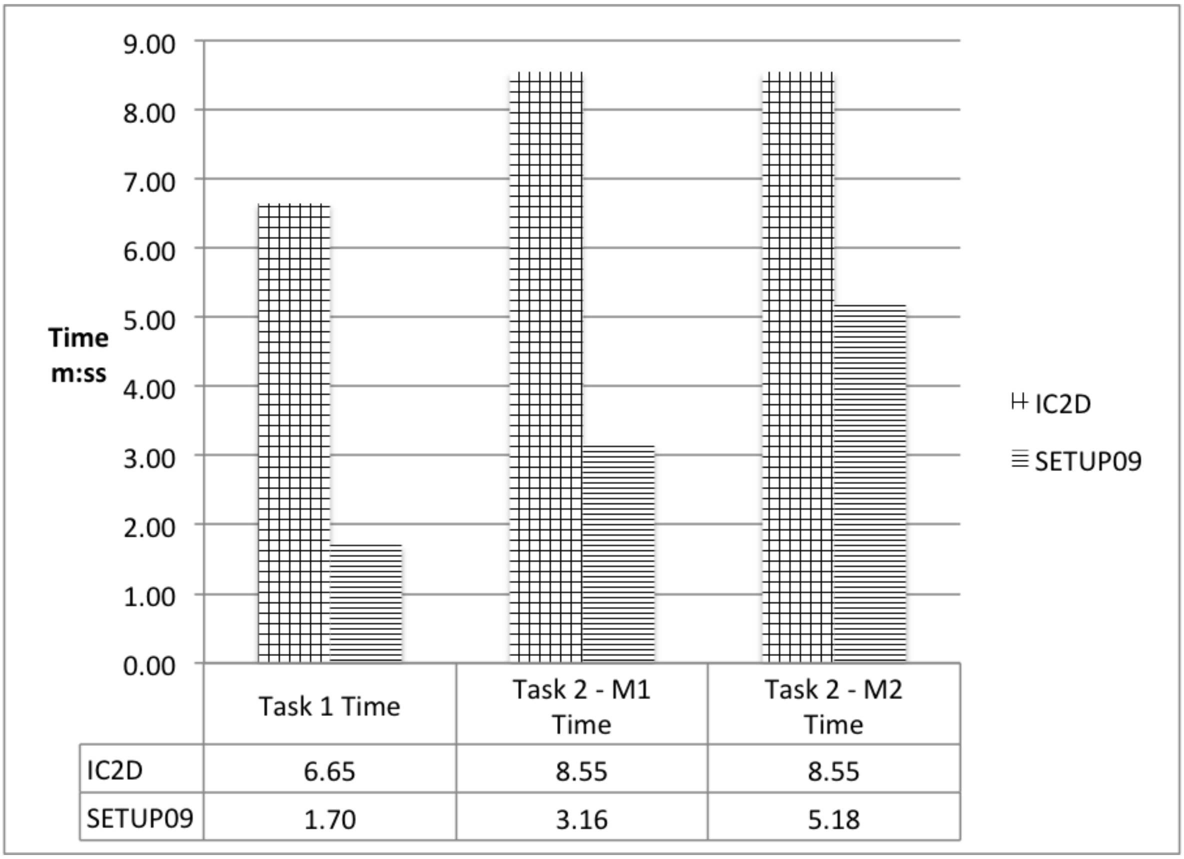


Figure 8.14: Performance time of IC2D and SETUP09 - All tasks with both VI and Sighted Participants. IC2D n = 16, SETUP09 n = 8

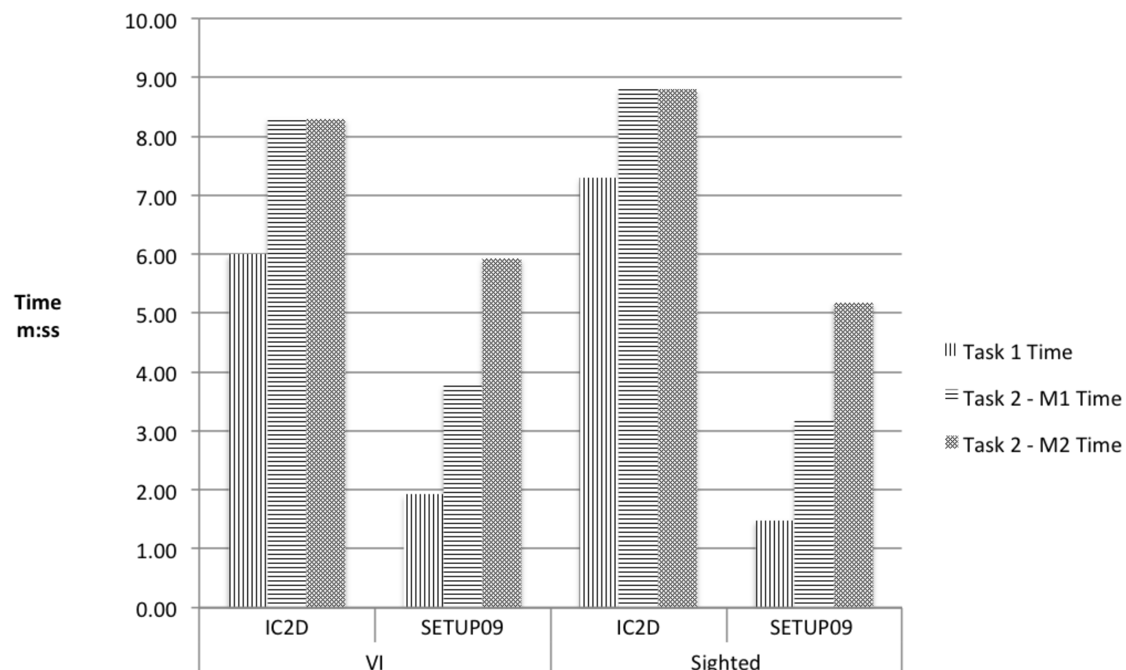
The participant's performance was assessed by the observer on the accuracy of the output produced. The bar chart in Figure 8.14 shows the average time taken to complete different tasks using IC2D and SETUP09 by both BVI and sighted participants. The mean time taken to complete task 1 by the participants using the SETUP09 system was 1.70 minutes whereas in the case of IC2D system it took participants 6.65 minutes. The mean time taken to complete task 2 - method 1 in the SETUP09 system is 3.16 minutes whereas the IC2D system has recorded 8.55 minutes. The mean time taken to complete task 2 - method 2 in the SETUP09 system is 5.18 minutes whereas the IC2D system has recorded 8.55 minutes. These results show that compared to IC2D the proposed SETUP09 took significantly less time to complete the tasks. In fact, the improvement in completing task 1 was by 74%, task 2 - method 1 by 63% and task 2 - method 2 by 39%.

The study was conducted using the same approach and conditions as that set in [Kamel and Landay, 2002]. The results of the on IC2D is presented in the website listed in the footnote [1] and also an image containing the IC2D result's averages are presented in the appendix E.5.6 that were used for the purpose

[1] https://www.slideserve.com/dillian/constructing-images-eyes-free-a-grid-based-dynamic-drawing-tool-for-the-blind

of the comparison with SETUP09 presented here.

The time measured to complete the tasks using SETUP09 in this study are consistent with the study described Chapter 6 where the participants had to draw simple shapes and navigate to multiple locations on the screen. In Task 2, to realise the exact output shape as in the study with IC2D reported in [Kamel and Landay, 2002], the participants had to use two methods. Despite using additional instructions, the participants completed the task in significantly less time using SETUP09 than with IC2D.



	VI		Sighted	
	IC2D	SETUP09	IC2D	SETUP09
Task 1 Time	6.00	1.92	7.30	1.48
Task 2 - M1 Time	8.30	3.77	8.80	3.16
Task 2 - M2 Time	8.30	5.93	8.80	5.18

Figure 8.15: Performance time of IC2D and SETUP09 - VI and Sighted participants for different tasks. IC2D n = 16, SETUP09 n = 8

The bar chart in Figure 8.15 compares the time taken by BVI and sighted participants to complete the various tasks using IC2D and SETUP09. BVI participants have taken slightly less time, i.e. 9.2%, than sighted participants in completing activities using IC2D, whereas with SETUP09, BVI participants took slightly more time, i.e. 18.3%.

For the task 1and 2, IC2D has recorded 6.00 and 7.30 m:ss with BVI participants and 7.30 and 8.80 m:ss for sighted participants. With SETUP09 tasks 1, 2 the average time is recorded as 1.92 and 3.77 m:ss with BVI participants and 1.48 and 3.16 m:ss for sighted participants. It should be noted that the difference in the completion time is not statistically significant between the BVI and sighted participants using IC2D and SETUP09 systems.

There are several studies in the past that recorded a better performance with sighted than blind participants. This was not the case with IC2D. It must be emphasised that the same conditions were used as in [Kamel and Landay, 2002]. We can conclude that BVI people performed similarly to sighted participants because of various reasons including greater exposure and therefore greater competence in using computers/laptop, as well as the ability to quickly processes non-visual information and instructions similar to people with normal vision people. It must be emphasised that the scope of the thesis is limited to developing SETUP09 software and establishing its pros and cons, and not on neuroscience research to determine the underline reasons why some people perform better at certain tasks. Such research requires a background in psychology and neuroscience and the use of specialist equipment like MRI scanner. The next two charts show the results on the participants confidence of using SETUP09 and performance or accuracy rating.

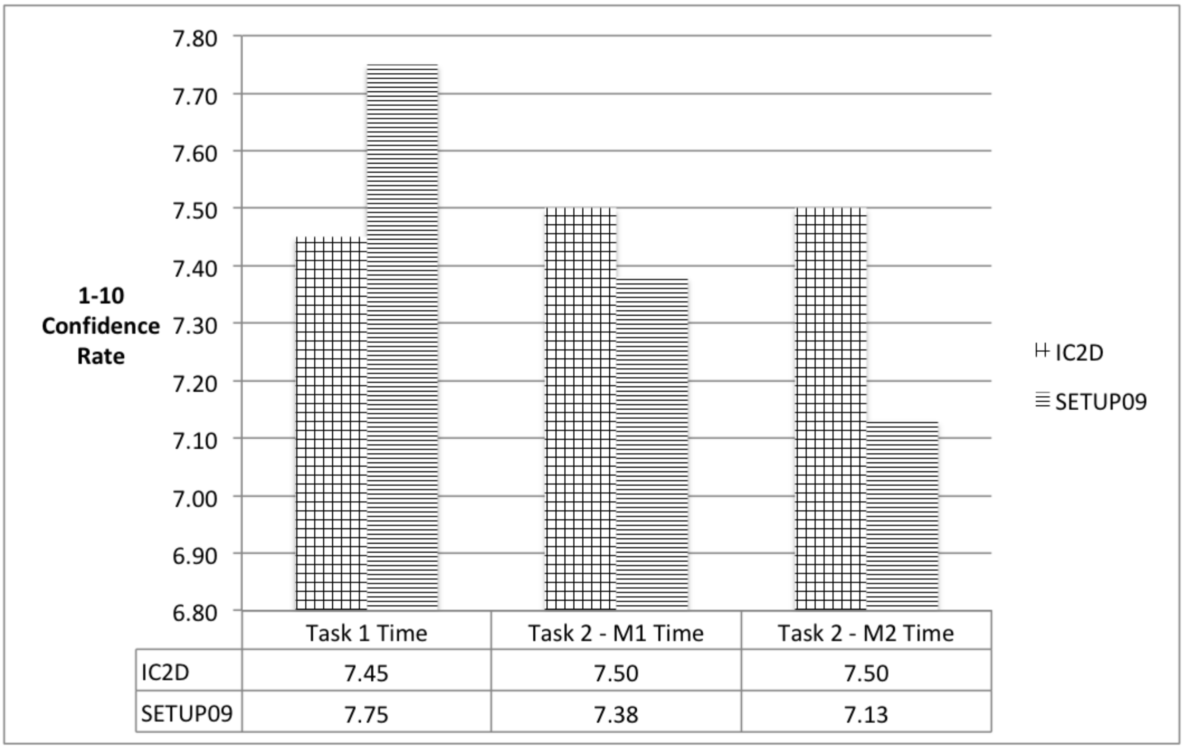
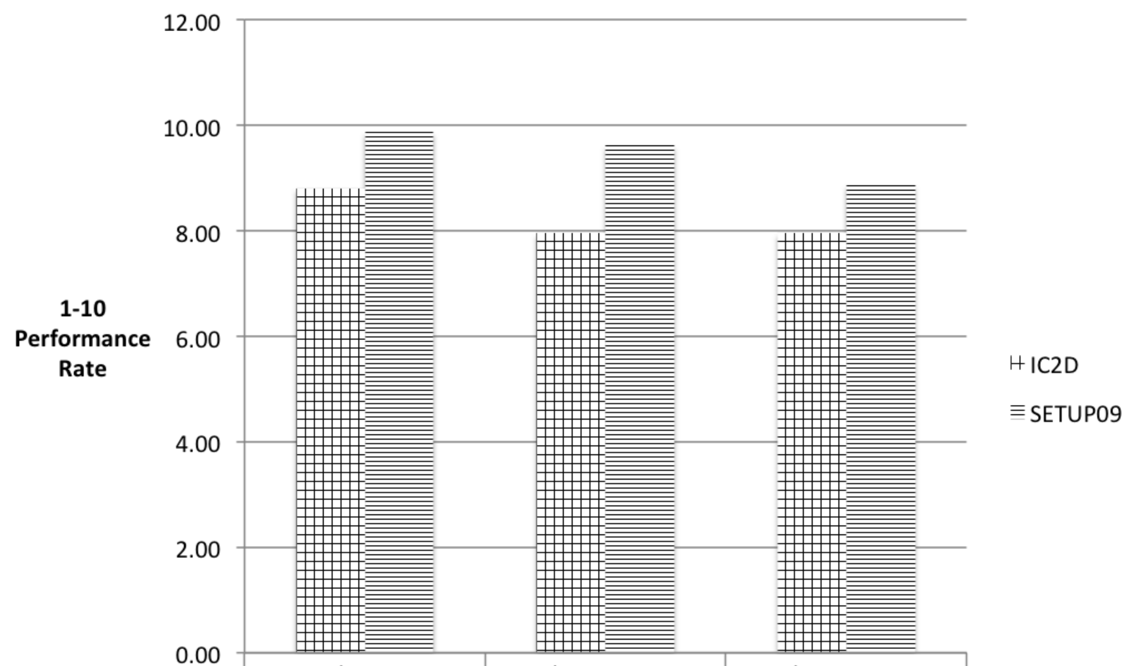


Figure 8.16: Participants' Confidence of IC2D and SETUP09. IC2D n = 16, SETUP09 n = 8

The bar chart in Figure 8.16 compares the confidence rate of participants at using IC2D and SETUP09. IC2D data was extracted from the report in [Kamel and Landay, 2002] and the SETUP09 data was collected from the study. SETUP09 shows a slightly higher performance average, 7.75 than IC2D 7.45 in task 1. For both task 2, methods 1 and 2 shows 7.50 with IC2D and 7.38, 7.13 with SETUP09. For task 1, participants were marginally more confident in using SETUP09 than IC2D by 4%. For both methods in task 2, the participants were marginally more confident in using IC2D than SETUP09 by 1.65% for method 1 and 5.2% for method 2. However, the difference is not statistically significant.



	Task 1 Time	Task 2 - M1 Time	Task 2 - M2 Time
IC2D	8.80	7.95	7.95
SETUP09	9.87	9.62	8.87

Figure 8.17: Performance - accuracy rating of IC2D and SETUP09. IC2D n = 16, SETUP09 n = 8

The bar chart in Figure 8.17 compares the performance or accuracy rate of output by the observer. The performance was a measure on the percentage of completion. For example, an incorrect location or point was given 9 out of 10, and three multiple incorrect locations were given 7 out of 10 by the observer. In all tasks, the performance rating using SETUP09 was higher than that reported for IC2D, in particular, for task 1 it was higher by 12%, for task 2 (method 1) is was higher by 21%, and for task 2 (method 2) it was higher by 11.5%. IC2D task 1 has 8.80 average rating whereas SETUP09 shows 9.87. For task 2, methods 1 and 2 shows 7.95 whereas SETUP09 shows 9.62 and 8.87. The above data demonstrate that the end product of SETUP09 was correctly produced both in task 1 and task 2 than in IC2D. The above data demonstrates that the effectiveness of SETUP09 compared to the prior IC2D technology.

The table below 8.10 summarises the average time, confidence and performance rating. Table 8.11 shows the participants' feedback of the SETUP09 system discussed in the finding section.

Task	IC2D Average Task Time	SETUP09 Average Task Time	User Confidence IC2D 1 - 10(very high)	User Confidence SETUP09	Evaluator Rating IC2D 1 - 10(very high)	Evaluator Rating SETUP09
Task 1	6.65	1.70	7.45	7.75	8.8	9.87
Task 2 - method 1	8.55	3.16	7.5	7.38	7.95	9.62
Task 2 - method 2	8.55	5.18	7.5	7.13	7.95	8.87

Table 8.10: Results of IC2D (n=16) and SETUP09 (n=8) testing with Blind and Sighted Participants

Participant type	Feedback
VI	I enjoy the actual metal challenge Easy and clear, able to visualise, I forgot what keys to navigate You have to carefully listen to system commands The more I use, the more I know, Can the system auto correct my commands
Sighted	Custom voice is needed, Can system tell, "your drawing is completed" Last one was hard, you need to know and remember the commands

Table 8.11: SETUP09 Participants feedback

8.3.4 Findings

- BVI and sighted participants took on average less time (3.35 m:ss) to complete the tasks using the SETUP09 system than the IC2D system (7.91 m:ss). This is attributed to the command-driven input mechanism of SETUP09 system which is faster than the menu-driven input mechanism of IC2D. When the commands are known, the drawing can be accomplished with few keystrokes. Another reason is because SETUP09 system has a multi-point cell referencing system as opposed to a single cell referencing system of IC2D. This enables a fewer number of navigation steps than the IC2D system.

- The confidence level of participants was recorded. This was done by questioning the participants, 10 being the highest level of confidence. With SETUP09 system the confidence level recorded was 7.42 and with the IC2D system it was 7.48. The main reason that impacted the confidence of using SETUP09 was the fact that the participants had to remember the user commands whereas this was not the case with IC2D. The feedback of the participants indicated that with greater use and therefore familiarity of the system their performance and confidence of using SETUP09 would improve immensely. This was very encouraging.

- Participant performance was rated (1-10) based on the accuracy of the task completed. For all three tasks the results show that SETUP09 was found to be more accurate than IC2D. The average accuracy rate with SETUP09 was 9.4 whereas with IC2D it was 8.2. The second task with SETUP09 was more challenging than the first task which impacted on the performance. Some participants made few errors and showed less accuracy in during the second task. It is evident from the results the participants made fewer errors using SETUP09 than IC2D.

- Results reported reveal that BVI participants performed better with IC2D system whereas sighted participants performed better with SETUP09 system. But there is no significant statistical difference in the results between BVI and sighted participants. This does not mean that IC2D is a better tool for BVI users than SETUP09. The results are very much dependent on how experiments were conducted. For example, the study conducted in [Kerr, 1983] and [Hill et al., 1993] revealed that sighted individuals took less time than blind individuals in visual imagery scanning and spatial identification experiments. In the IC2D study reported in [Kamel and Landay, 2002] sighted people were blindfolded. This begs the questions of the accuracy of sighted participants' speed of keyboard use as sighted people do not normally use the keyboard blindfolded and finding the keyboard key must have been difficult. Hence, it is very likely blindfolded sighted people may have spent more time carrying out the task. In the SETUP09 study the interface was hidden from the sighted user rather than blindfolding them.

8.3.5 Limitations

- The sample size of SETUP09 was smaller than that reported for IC2D. This was because the study was conducted during the global pandemic and the sample size was limited as per the recommendation by the university's research ethics committee. However, the comparison analysis reveals the trend in the results obtained. It should be noted that the results of the study on SETUP09 in Chapter 6 was conducted with a bigger sample size of participants, where similar results are reported. This gives the above study credence.

- IC2D results were extracted from [Kamel and Landay, 2002]. The actual experimental conditions for IC2D study conducted in 2002 are likely to have been different to that of SETUP09, for example, the lower speed of the computer peripherals, computer literacy skills of the participant, degree of keyboard use etc. This must have influenced the time taken by IC2D system. However, this does not invalidate the IC2D tool as a technique for blind drawing as much as the SETUP09 system.

- It could be questioned as to why sighted people were made the control group in this study to compare with blind participants. This was because of their prior experience of recognising different shapes. The researcher has conducted multiple studies under controlled conditions where blind people were compared with groups of people who were partially sighted, early blind, late blind, and sighted. It was surprising to learn from the findings of the study that there was no significant difference between the performances among these groups, other than sighted people were marginally faster. The study demonstrated that prior visual experience of sighted people to recognise shapes did not give them a greater competitive advantage over partially sighted, early and late blind people. Despite the sample size being relatively small nonetheless these findings are encouraging and confirm that the SETUP09 software enables BVI people to generate artwork/drawing with confidence as sighted people. The experiment did not evaluate the groups skills in visualisation of shapes, memory and computer usage. It should be noted that the only claim made by this hypothesis is that Blind people perform better in using SETUP09 than IC2D.

- It is important to point out that this experiment was not conducted by the same individuals. Therefore, an exact comparison cannot be made. However, the application of the tool in the studies carried out here used the same experimental conditions (insofar as was feasible) as was used in a previous study IC2D. Hence, the effect of this on the results may vary somewhat based on the user's ability, technical competency, and hardware familiarity and hardware models. Therefore, the effect on experimental results, e.g. completion time, errors, user satisfaction, needs to be understood with this in mind. Impact on experimental results of above factors was also observed in the experiment presented by [Bornschein et al., 2018].

In the above section SETUP09 is compared with similar analogue and digital systems using measures such as time, accuracy, error count and user feedback. In the next section a comparison of the SETUP09 model was provided with prior related work in terms of navigation and image production. The research reported in Chapter 3 is not repeated here but aims to give the reader an understanding of how the model is proposed. The enhancements made to the proposed SETUP09 is critically appraised with existing approaches.

8.4 Comparing the Proposed Model with Related Work

There have been many applications introduced in the past to plot graphs, draw lines, and create geometry and DFD (Data Flow Diagrams) for blind users. Although a detailed discussion of these tools was conducted in Section 3.1.2 of this thesis, but they are not compared with the proposed SETUP09. In this Section the prior related techniques are compared with SETUP09.

The Kevin system [Blenkhorn and Evans, 1998] uses a tabular tactile over layer with a fixed set of buttons to navigate and edit DFDs without retaining layout information, whereas SETUP09 is designed specifically for retaining layout information through techniques such as zoomin, navigation and grouping, by marking absolute and relative positions.

There are similarities between IC2D [Kamel and Landay, 2002] and the SETUP09 system. One of the main similarities is the grid-based navigation system. Keyboard and voice are the main input and output sources of both systems. SETUP09 however uses compass direction-based navigation, where IC2D uses telephone keypad-based navigation. They are both represented by 3x3 grids on the screen. When using IC2D the user recognises the top left cell as grid number one, and in SETUP09 the user recognises the top left cell as the north-west cell. SETUP09 has eliminated the limitation of a palette system and designed a command language for drawing. Using the SETUP09 drawing language the blind user can do the following (i) create new drawings, (ii) reuse them in different locations and zoom levels, (iii) use primitive shapes and make connections to and from different zoom levels, and (iv) label and link objects as they wish. The SETUP09 system is not limited to one grid reference point, but multiple (i.e. nine) compass points on each grid-cell. IC2D only has centre point cell referencing, as opposed to the multi-point cell referencing system as in SETUP09. Compass cursor points enable the production of more sophisticated drawings by linking points on the same zoom level or different zoom levels. Limited cell referencing in IC2D leads to limitation in screen utilisation and therefore extended time is taken to produce artwork. Furthermore, the system does not follow an expandable formal abstract approach but is based on concrete system development with limited support for different modalities.

The AHEAD system [Rassmus-Grohn et al., 2007] is an audio-haptic drawing editor and explorer for education. There is an external PHANToM device for different modes of drawing, using a PHANToM pen or computer mouse. The blind user of the AHEAD tool will require assistance from a sighted person otherwise the user will take a considerably longer time ($>$ 30 minutes) compared with keyboard-driven drawing system.

A pin-matrix based tactile drawing tool [Bornschein et al., 2018] needs a drawing workstation comprising a laptop computer, monitor, BrailleDis 6240 tactile pin-matrix device, ToF 3D camera system with wooden blocks for capturing 3D movement and a wireless digitiser stylus and takes more than more than 18 minutes. Compared with the above-mentioned tools SETUP09 is not driven by haptic input modality but by command-driven modality. Therefore, the time taken to draw an image in SETUP09 is considerably shorter than the recorded times for the AHEAD system and pin-matrix based tactile drawing tool.

The TeDUB system [Petrie et al., 2002] is mainly focused on importing and analysing existing diagrams, and it TEDUB does not convey information about the shape of the layout or artwork drawn. Whereas SETUP09 facilitates the implementation of drawings from scratch, however SETUP09 is not designed to analyse images from other systems. SETUP09 retrieves its own images by executing a set of concrete commands. Therefore, the information analysis stage of an image would not be necessary. Furthermore, finding the semantic meaning of diagrams or images is not necessary too. In fact, blind users could interpret the image by 3D printing, image verbalisers or via modern Braille machines. The TeDUB system suggests a keyboard-based navigation technique as a future improvement, whereas SETUP09 already uses a compass-based keypad for navigation.

System PLUMP [Calder et al., 2007] uses linked lists and Heap's algorithms to store data in a data structure, which is accessed in a sequential manner. Unlike SETUP09 no information is conveyed about the shape of the drawing layout.

The BPLOT system [Fujiyoshi et al., 2014] is a universal tactile graphics production system that uses plotter control language for sighted people to work alongside blind people with a text command editor. However blind people are generally provided with a tactile input tablet interface when using BPLOT. Plotter control language is mainly used to control a vector graphics printing device, which is only useful for sighted users as there is a need to remember screen coordinates of vector graphs and is not a user-friendly technique to identify points on the screen. Unlike BPLOT the proposed SETUP09 is fundamentally operated by a command-driven language designed and developed specifically for blind people to support orientation, navigation and art production. Results of the evaluation on SETUP09 in Chapter 6 confirms blind people find it easy to use without the support of a sighted person, and the users do not need to remember screen orientation and screen coordinates.

The next section addresses screen navigation techniques by comparing them with SETUP09's navigation technique.

8.4.1 Comparison of Screen Navigation Techniques

Several alternative screen navigation techniques to SETUP09 were discussed in Chapter 3.2. It includes (i) [de Mauro et al., 2001] voice-controlled navigation , (ii) [Mcnair and Waibel, 1994] target-based navigation, (iii) direction-based navigation [Sears et al., 2001], and (iv) [Zhu and Feng, 2010] grid-based magnification navigation with linear style audio interface.

Voice-controlled and target based navigation methods can take a longer time to process, as a result the cursor location can be missed and causing a high level of errors according to Sear [Sears et al., 2001], whereas grid-based keyboard controlled magnification navigation [Zhu and Feng, 2010] has proven to be efficient while magnification accommodates people with a low level of sight. This thesis introduced a compass-based navigation technique in Chapter 4.1.3 which is a form of grid navigation using compass direction names that are familiar to blind people.

The advantage of SETUP09 navigation is that the blind user can compartmentalise the screen areas to navigate and place objects on the screen. This helps the blind user easily process the mental images they are trying to create as well as make any corrections to the image. In SETUP09 not only do keyboard arrow keys facilitate the initiation of such navigation, but voice and Braille input interfaces can also initiate this type of navigation because SETUP09 is developed using an abstract set of rules that can easily be integrated into different external input modalities.

8.4.2 Conclusion

This chapter presents the results of comparative studies of two blind drawing methods analogue and digital (SETUP09) and two software techniques for creating computer artwork and drawing based on IC2D system and SETUP09 system. The studies were conducted with early-blind, late-blind and sighted participants. The results reveal compared to the analogue manual toolkit and IC2D the proposed SETUP09 technique is an effective drawing tool for blind participants. BVI participants found that SETUP09's navigation functionality was relatively easy to use for accurately producing artwork and drawing. In an ideal situation, further testing would have been desirable as explained by Jenny Preece and Helen Sharp in [Preece et al., 2015], but such system testing is beyond the scope of this thesis. In the thesis, the testing was conducted to confirm the outset hypothesis. Presented next in Chapter 9 is a set of guidelines to devise CAD command-driven drawing systems and the process of breaking down a set of requirements to produce a proof of concept prototype using a formal approach.

Chapter 9

Design Guidelines for CADB

This chapter introduces an added outcome of this thesis that the formal abstract approach of modelling CADB technology presented in this thesis not only allows for greater understanding of what the model means but also aids in the design of assistive technology system. This chapter highlights a summary of a refined set of space, shape and usability design criteria for the proposed assistive technology for blind people. It lists the step by step actions (process flow) needed to the development of rule-based systems. These guidelines should facilitate an effective design of CADB with dissemination of methodological information that is presented in user-friendly fashion that has not introduced before. It can also be used to improve similar systems by changing/adding features for efficiency, ease of use, and to assist with the system evaluation. This chapter also emphases the main contribution of the proposed innovation in much broader terms. It highlights the benefits of the proposed guidelines for software designers to recognise a technology requirement, problem classification, formal model application, development of a proof system and model evaluation in a formal and methodological manner leading to the final recap of the research contribution, research questions and goals in, Chapter 10.

9.1 Design Guidelines to Establish a Formal Specification Language for CADB

The design guidelines presented for computer-aided drawing systems in this Chapter can be categorised into three subsets. The first subset relates to system help, namely usability guidelines; the second relates to system navigation, namely space guidelines; and the final subset relates to art creation, namely shape guidelines. These guidelines were explained in Chapter 4 using formal language specifications and given were implementation examples.

It became apparent from the experimental evaluation with BVI users that some guidelines were more popular than others. After consultation with the BVI community during and after the experiments, that involved five participants from RNIB and MACS, the guidelines are presented here in order of importance. The participants were asked to prioritise guidelines based on their understanding. Four out of five users thought usability language to be the most important set of guidelines and space language the second most important. Participants said that this is because they liked to be able to work with the software independently without relying on help from sighted people. They liked the easy navigation facility offered by the grid system which gave them a sense of screen orientation, hence they picked space language as the second most important set of guidelines. One participant thought that priority should be given to shape creation, as the software is built specifically for the purpose of drawing, hence he prioritised shape language and space language over usability language. The following sections present guidelines according its popularity.

9.1.1 Usability Guidelines

When designing a product for blind individuals, usability is the most important factor identified by BVI participants as it enables them to work independently without relying on help/assistance from a support worker. Usability control features include:

- **Ability to work without relying on help** - Self-initialisation and exiting the application. Inbuilt shortcuts and quick access to open, exit, help, and screen reading features. Ability to put work in progress on hold and return to it.
- **Ability to learn easily** - Access to help and different modes of help (voice, Braille) and different levels of "how to use" tutorials.
- **Support for intended activity** - Ability to create primitive shapes, draw non-primitive shapes, use previously designed artwork, group and define art.
- **Ability to recover** - Features such as save, delete, erase, and ability to retrieve previous images.
- **Ease of use** - Complete an intended task without much delay and effort. Ability to interact with the system without much effort, and have access to features such as redo, undo, copy, paste.
- **Ability to manage cognitive load** - Simple, short, direct linguistic interaction to complement the working memory of the intended user.
- **Ability to personalise** - System features to select different levels of interaction pace, interaction method and user type. For example, control little or more voice feedback, change of system input and output modes from keyboard to Braille to voice, change of interface items and sizes.

9.1.2 Space Guidelines

Spatial orientation and spatial recognition are the second most important features identified by BVI participants when navigating on the screen - to know the screen focus position in relation to a known point, to know the screen location, and to navigate in and out to a screen location or point. Spatial control features include:

- **Reference to a location**: A location on the screen with an effective referencing system to identify the location.
- **Reference to a point**: A point on the screen with an effective referencing system to identify different points on a cell/screen.
- **Technique to navigate to a location**: Effective technique and linguistics to convey the action of transition.
- **Technique to navigate out of a location**: Effective technique and linguistics to convey the action of relocation.
- **Reference to a navigation route**: Ability to inform user position with reference to the starting location; a position on the screen with an effective referencing system of the route.

9.1.3 Shape Guidelines

BVI participants were keen to obtain system support with graphics creation, thus be co-creators of images by making decisions such as start points, direction, end points, sizes, and shapes. Shape control features include:

- **A technique to create a line/lines between points**: A line has two endpoints, whereas line segments have at least two endpoints but can be extended indefinitely.
- **A technique to create a curve/curves between points**: Two endpoints and one control point define a quadratic curve, and two endpoints and two control points define a cubic curve.
- **A technique to produce primitive or arbitrary shapes**: Create two-dimensional primitive shapes made of points, lines, rectangles, arcs, ellipses, and curves; define a collection of shapes with a name/label to convey a meaning.
- **A technique to group a set of shapes**: Assign a name/label to one or many shapes in order to recognise them later.
- **A technique to reuse a new shape**: Call and use a created shape by its name/label.

9.2 Process Flow of CADB

Previous research by[Mitchell et al., 2015], [Wilkinson, 2011], [Luck, 2003] has introduced user-centred participatory design processes in the context of HCI system design. But there is no universally accepted design process in the context of computer-aided drawing systems for blind users. Hence, developed in this research with the involvement of BVI users is a CADB design process using a formal approach. This section describes in detail the process flow of CADB design and development that can be used when choosing a formal approach to system design. These steps are not just specific to SETUP09 but suitable for the pursuance of grammar-based system implementation. Process flow steps for CADB design are:

1. Classify requirements based on the guidelines.

   - Requirements of a system can be user-based and/or research-based. When the requirements are collected, they are first categorised into predefined guidelines (SETUP09 has usability, space and shape guidelines) by rewriting them with technical details for system designers. This enables designers to compartmentalise system components, prioritise requirement subcategories, rewrite missing and misunderstood requirements.

2. Convert requirements into abstract grammar rules (repeat).

   - A formal approach in computer science uses an appropriate mathematical technique to write a formal specification. A formal specification starts with grammar formation based on collected requirements and designer details. This stage enables the conversion of high-level user requirements into mathematical notations, by eliminating repetition and inaccuracies by making decisions on language terminals and non-terminals.

3. Refine grammar rules to form a connection between rules.

   - Grammar rules of a formal specification are well connected and do not stand in isolation. A terminal is described by another terminal or non-terminal and there could be multiple methods to execute grammar. Some rules refer to other rules before a primary rule is resolved. The rules can be resolved in a complex or simple manner. Grammar refinement enables the elimination of sophisticated grammar parsings, maintaining the specification accuracy that can be checked using a parsing tree.

4. Implement grammar using concrete syntax (one at a time).

   - Specification grammar needs a high-level programming language to be implemented and initiate user interaction through a system. When a high-level programme language is picked based on the complexity of the specification, rules are then expressed using the chosen language data structures to generate system actions for user inputs. Implementing grammar also requires lexical analysis, semantic analysis and parsing that work together with complete concrete syntax implementation.

5. Check if grammar works in conjunction.

   - A language should be able to make a meaningful output/artwork by resolving multiple grammar rules in conjunction. After the concrete system implementation, the language can be checked using a system interface, with multiple commands, for expected outputs. Complex language commands can also be tested, with an abstract syntax tree to find any inconsistencies within the language.

6. Test requirements.

   - A requirement test is conducted with potential users to check if the specification meets user needs, while the grammar test is conducted to verify mathematical and structural accuracy. The ultimate purpose of the specification is that it delivers user action through a system in an efficient and user-friendly manner. This can be achieved by a series of system evaluations and available comparative system evaluations.

7. Modify implementation.

   - A formal analysis of the specification is conducted with the revision and updating of grammar rules as a result of user testing, to ensure the model meets user needs and improve design consistency. This stage can be achieved by introducing new grammar (terminals, non-terminals, or tokens), or changing the previously formed grammar. This stage can take place many times with new test findings. Some changes may not impact specification alteration but require system updates or interface updates that can be dealt with simultaneously.

9.2.1 Process Flow Diagram of CADB

Figure 9.1 shows an architectural model of the process flow guidelines, how they can be implemented, improved and integrated when designing computer-aided command-driven blind drawing systems. The first stage is to identify a set of user requirements, classify them and produce a refined set of abstract rules. The second stage is to implement the grammar using abstract/concrete tools. The third stage is to test the requirements and modify implementation, finally resulting in a computer-aided drawing system for blind people.

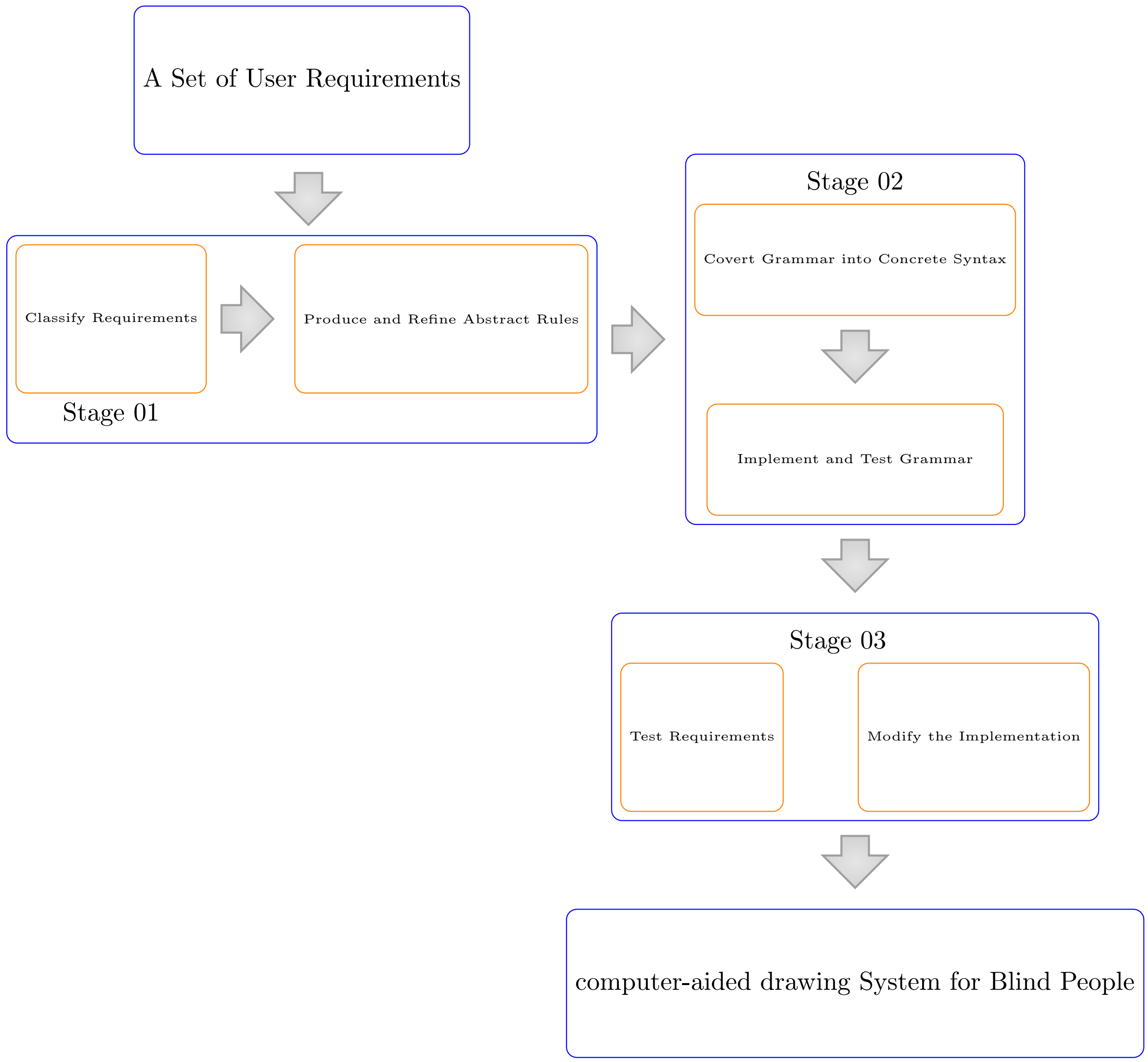


Figure 9.1: Process Flow Diagram of CADB

9.3 Conclusion

The guidelines provide a framework through which content is turned into the features of a computer-aided drawing system. It provides a clear understanding of what is needed by designers and developers when developing a drawing system for the intended user group. The chapter also gives a process overview of CADB system design and development, i.e. the guidelines and their implementation. Provided are step by step actions needed to convert the guideline requirements to a formally driven CADB system. This then leads to the final chapter, which revisits research questions to expound the contribution and outcome of this thesis.

Chapter 10

Contribution, Further Direction and Conclusion

The final chapter concludes the overall work presented in the thesis. It outlines the main research contribution, the formal model based innovative computer-aided command-driven drawing technique and initial research questions on formalisation along with motivation, challenges and the aim. The main aim of the research was to investigate and realise a formal model approach to a computer-aided drawing technique for blind people whereas some added outcomes of this work are a set of well-refined CADB design features presented as design guidelines and the findings of user testing that revealed information of BVI user's CADB system-interaction. This was achieved systematically through successive stages in the research, that is, (i) background study in Chapter 2; (ii) problem identification and solution classification in Chapter 3; (iii) create a formal specification in Chapter 4; (iv) use a formal model to generate syntax and semantics to development a proof system in Chapter 5; and (v) a set of experimental evaluations and revisions in Chapters 6 to 9. The proposed technique has been designed and developed by a formal set of grammar with the knowledge from empirical studies that has not tried before in the domain of assistive technology, CADB system design. Chapter 10 ends with an explanation on the use of the grammar-driven application in the production of artwork without the help/assistance of support work, and its wider social impact.

10.1 Contribution

The most significant contribution of this thesis in the field of HCI and computing is the formal approach to a computer-aided command-driven drawing technique for BVI users. The technique is innovative and introduces compass-based screen locations and points for creating drawing and artwork. It has been designed and developed using a formal set of grammar paradigm together with knowledge from experimental studies.The command language-driven grammar has been modelled in a manner which allows users to be more expressive and use their own initiative. In particular, the aim was to provide a formal method for a command-driven drawing technique for blind people. System architecture and a formal implementation of front-end parsing structure have been proposed here that add adaptive capabilities to the SETUP09 model. The approach comprised:

1. Identification and classification of characteristics of computer-aided drawing language for blind people.
2. Use of a formal modelling technique to add adaptivity.
3. Modelling a command-driven interpreter to enable the proposed drawing language to be used by blind people.
4. Development of a prototype (SETUP09) based on the proposed model that can be used to practically evaluate the innovation.
5. Experimental evaluations of the navigation and artwork creation technique with a subset of different users (blind, partially sighted, late blind and sighted users).

10.1.1 Motivation Revisited

The research was inspired by the need of the wider blind and visually impaired community to improve assistive technologies of computer-aided drawing systems. This would enable BVI people to be involved in activities such as scientific diagram drawing and art production, as well enable them to engage in education and the creative arts. To achieve this, (i) it was necessary to develop sustainable design guidelines that would overcome the challenges faced by assistive technology developers/researchers of computer-aided drawing systems; and (ii) introduced was a formal approach to computer-aided command driven drawing. This approach is shown to enable BVI users to produce artwork using 2D shapes according to a given specification. Other motivating factors for the research are:

1. Currently, there is no universally accepted specification for screen navigation and graphics manipulation for BVI users.
2. Most systems that have been developed in the past assume that users are required to produce one particular type of graphic or diagram and do not enable artwork production in general.
3. The system should accommodate for different levels blindness.
4. There is no existing abstract approach to support technology with alternative input and output mechanisms.

10.1.2 Challenges Revisited

Presented here are the initial research questions/challenges that were brought to light in the early stages of the research work that had to be tackled, and the research questions/challenges resulting during the research work.

Initial Challenges

- What are the main concerns of computer usage among BVI users?
- What are the different uses of assistive technology and their impact?
- What are the different design approaches to assistive technology?

Derived Challenges

- What are the characteristics of a drawing system for blind people?
- What sort of object layout a blind person would easily conceive and convey?
- What sort of a navigation system would easily build a sense of object layout?
- What sort of a space arrangement could allow easy navigation?
- What usability features are best suitable to design CADB?

10.1.3 Technical Aims Revisited

The central question the research answered was: What is the scope of computer-aided command-driven drawing technique for blind users, and what interaction could be made available to blind users?

The thesis complements existing theories and systems of computer-aided drawing technique with an abstract formal set of guidelines and grammar. The research also addressed these derived questions:

The primary research questions

1. What constitutes a formal drawing language for the blind user?
2. What is the best way to model and implement a drawing language?
3. How can such a model be evaluated with blind people and other systems?

To date, the primary approach to providing answers to these questions has been software implementation with alternative input-output mechanisms. Although such an approach has given a good understanding of how to implement computer-aided drawing techniques for blind people, such an approach is often difficult to extrapolate.

Furthermore, the design decisions made in such systems and their implementation are difficult to trace and revise; hence, none of these systems are in mainstream use. The systems are time-consuming or require specialist equipment to use them. This has resulted in the drawback of making them difficult for blind people to use in real-world activities.

10.1.4 Summary of Contribution

During the research work, it has come to light that there is limited availability of computer-aided drawing technologies, and there is a limited understanding by developers of such technologies of the design process that is needed for usage by blind people. The main contribution of this thesis is a formal approach to CADB system design, and additional benefits of this research are a drawing technique with screen navigation, a set of experimental, comparative findings of the technique, and a set of guidelines to design computer- aided command-driven drawing languages for blind people. Chapter 3 discusses the modelling issues, Chapter 4 explains the specification rules, and Chapter 5 explains the implementation of a command-driven drawing language and its facets. The other vital contributions were the experimental evaluations in Chapter 6 and a comparative study in Chapter 8. The research findings and contribution to the guidelines are presented in Chapter 9. From the experimental findings, it could be said that even though the prototype is not thoroughly refined, some essential rules are clearly presented in the SETUP09 system and the system is capable of screen navigation and 2D image production. The validation results of the SETUP09 technique are presented in the various chapters and provides a basic outline for the future direction of research.

10.2 Application of Research

The following sections will outline the application of research and use in real life.

The design guidelines are predominantly for computer-aided drawing systems. However, the usability guidelines can be used to support other software designs for BVI people such as writing, coding, and reading applications. The primary emphasis of the research was on the usability design guidelines that were introduced earlier, such ability to work without relying on help, ability to learn easily, support for intended activity, ability to recover, ease of use, ability to manage cognitive load, and ability to personalise. The guidelines can be adapted to work with more universal design strategies. The formal approach is not by itself a complete solution to the main issue concerned with design and implementation of a computer-aided drawing system but adds significant value to many application areas that seek a formal approach to produce designs with input signals. At present, there are many efforts to design and implement drawing tools for blind people using different methods (e.g. 2D Braille, Phantom Pens, tablets). In particular, there is a trend towards the design of the computer-based input-driven system with user-friendly interfaces. The proposed design and implementation model may also be applied to the design of such drawing systems for blind people. Such systems would need guidelines to design and a formal approach to implement.

10.3 Use in the Real Life

The proposed technique can be employed in many real-life situations related to the design and development of assistive technologies.

Blind students in educational establishments such as primary and secondary schools can use the software tool to learn to draw simple and complex 2D shapes, construct images using screen points and compass directions, or programme a sequence of commands to get the desired picture output without the need of a teacher or support worker. Participant feedback from the user study conducted in this research suggests that independent drawing skills learnt from early years should alleviate anxieties surrounding drawing, increase screen coordination skills and thus promote the confidence of BVI users. In the STEM field the software can enable blind students to create scientific drawings such as flowcharts, data flow diagrams, wireframes, structure charts and many more technical diagrams.

Developing the spatial skills of blind people is essential for them to independently orient and navigate in space. This requires conceptualised thinking of space with some referencing system such as a reference to a location, spatial points, shapes or distance. During the experiments conducted in this research, a few BVI participants commented that the technique improved their memory skill. It was also found from the experiments that independent exploration of screen navigation and tactile map outputs can facilitate the users understanding of screen spatial/grid structure.

Blind people who are interested in art production for leisure, work or community purposes can benefit by using the proposed software to produce artwork as well as create art for leaflets or banners, or visualise artwork by others. Those who own swell papers, a printer and fuser can benefit from tactile printed images.

Sighted people can also benefit from the innovation, not just as a navigation technique for drawing but also as an easy way to navigate on screen in everyday applications. For example, the user can utilise a compass grid over-layer combined with keyboard or speech commands to move the cursor to a specific location in a word processing application, instead of using a mouse or tracker pad. Another example would be the ability to move the cursor to a specific location on a webpage in a software such as screen readers. A third example is to find a location on a map using a 3x3 compass-based grid over-layer. The technique traces the navigation path and remembers to navigate back and forth. Compass-based navigation takes the cursor into the intended screen location without having to rely on a tracker pad, mouse or visual perception.

10.4 Further Direction for Research

It has become apparent that there are several possible directions for the future expansion of computer-aided drawing systems for blind people.

Although the development of SETUP09 system has made significant contributions to the drawing guidelines and their formal implementation, the individual rules must be evaluated with different input modalities such as speech, Braille and tablet. According to the findings in this thesis, speech instruction and speech commands worked well with individuals who became blind at a later stage of life than other subsets of users (blind, visually impaired or sighted) that were tested in the research as discussed in Chapter 8.

Although Braille has been extensively used by blind and partially sighted people it has limitations of producing drawings and artwork. Based on the studies carried out in this research with BVI participants it is envisaged that alternative modalities based on keyboard-based computer-aided assistive technologies will become popular. To this end participants have suggested the use of on-screen mobile vibration for validating the onscreen drawings, which can be achieved with the use of haptic technology.

In Chapters 6 and 8 a short-term, small-scale user study of the system is provided. Although the short-term studies evaluate the system, a long-term evaluation on a large scale is necessary to better evaluate system's effectiveness. The proposed prototype provides some facilities for error detection and manual error correction. However, the usability of the system could be further improved if implemented with a more robust error detection and correction classes or API. It was also realised that features such as magnification are important, to accommodate size of the image, size of letters and thickness, which could significantly improve the use of the proposed SETUP09 system.

Currently, the system is available only through prototype distribution. An open-source, web-based front-end interface could tremendously improve the accessibility of the software among targeted users. Even though the prototype adequately provided facilities to carry out an experimental evaluation, the proposed rules must be fully implemented before the system could be distributed.

The system was mainly designed for blind people who have an interest in drawing general or scientific artwork. However, the commands used may not always accommodate the production of different types of diagrams. Therefore, end-users must have a mechanism to add new shapes to the shape library using specific symbols.

In answer to the question as to what is the functionality provided by SETUP09 itself? Well it is an art creation technique through command interpretation. It is not integrated with different input or output modalities. The proposed approach addresses the issues of how to formally specify and implement a set of rules which can effectively support drawing by blind people. Although this approach for rule implementation and system interaction is presented, research and testing should be carried out into alternative applications of these guidelines to compare what these alternative applications might look like and how they might work. Therefore, the research of this thesis promotes further investigation in the areas of:

1. Different modes of input modalities
2. Web interface
3. Error detection and correction
4. Alternative method of command interpretation
5. Tablet tactile over-layer
6. Personalisation: method to add library shapes
7. Output exchange facilities to similar drawing tools

8. Import facilities from other drawing tools

9. Empirical testing of proposed facilities for its effectiveness

10.5 Research Conclusion

The primary application of this research for blind people is how drawing system design, development and implementation can be undertaken. This thesis provides an initial investigation and examination of spatial skills, image presentation ability, and usability required to design an art production technique based on a formal set of guidelines. It was essential to seek the collaboration of blind people in the development of this bespoke technology to gain insight, eradicate superfluous commands and for the technology to be useful and relevant to them.

As much as it is necessary to accommodate different modalities, it is more advantageous to identify the basic components of a blind drawing language, and design, develop and test a suitable, self-reliant, easy to use, efficient technique to navigate on the screen that facilities drawing by BVI users.

It is hoped that the formal approach taken for modelling the system, at a level of abstraction above that of a concrete system, may clarify the abstract guidelines and rules that should be subject to the action of drawing and other possibilities. Such abstraction should also give clarification to evaluate different drawing systems in different contexts. It also provides answers to the primary thesis questions: What constitutes a formal drawing language for the blind user, and how to model, implement and evaluate it. Finally, it is hoped this thesis has provided the foundation for devising a set of CADB guidelines, rules and example of implementation and empirical testings that can be used in a broader context.

Appendix A

Java Front End Lexical Phases

This appendix details the code presentation of the frontend parser phases. The section demonstrate formal syntax and semantics of SETUP09 built using Java programming construct. This is provided as an alternative for the reader to better understand SETUP09 formal grammar implementation.

A.1 Lexical Analysis

The Tokenizer function reads various input such as names, numbers, space, marks and then resolve the input in to words or tokens.

```
/*Lexical Analysis function.*/
 public static ArrayList Tokenizer(String s) {

        ArrayList items = new ArrayList();

        StringTokenizer st = new StringTokenizer(s, " ", true);

        while (st.hasMoreTokens()) {
            items.add(st.nextToken());
        }

        return items;
    }
```

A.2 Lexical Semantic Analysis

The purpose of the class is to looks at the parsed token, see if any meaning can be generated, and if so, a type and a value is produced.

```
/*Token Class for Lexical Semantic Analysis function. */
 static class Token {

        int type;
        String Value;

        Token(int t, String v) {
            type = t;
            Value = v;

        }
    }

/***************************************************/


/*Lexical Semantic Analysis function using the Token Class*/

  public static Token Tokens(String s) {
        if (s.equals(",")) {
            return new Token(1, ",");
        } else if (s.equalsIgnoreCase("N")) {
            return new Token(2, "N");
        } else if (s.equalsIgnoreCase("C")) {
            return new Token(3, "C");
        } else if (s.equalsIgnoreCase("S")) {
            return new Token(4, "S");
        } else if (s.equalsIgnoreCase("W")) {
            return new Token(5, "W");
        } else if (s.equalsIgnoreCase("NE")) {
            return new Token(6, "NE");
        } else if (s.equalsIgnoreCase("SE")) {
            return new Token(7, "SE");
        } else if (s.equalsIgnoreCase("SW")) {
```

```
        return new Token(8, "SW");
    } else if (s.equalsIgnoreCase("NW")) {
        return new Token(9, "NW");
    } else if (s.equalsIgnoreCase("zoomin")) {
        return new Token(16, "zoomin");
    } else if (s.equalsIgnoreCase("zoomout")) {
        return new Token(10, "zoomout");
    } else if (s.equalsIgnoreCase("circle")) {
        return new Token(11, "circle");
    } else if (s.equalsIgnoreCase("line")) {
        return new Token(12, "line");
    } else if (s.equalsIgnoreCase("define")) {
        return new Token(13, "define");
    } else if (s.equalsIgnoreCase("call")) {
        return new Token(17, "call");
    } else if (s.equalsIgnoreCase("square")) {
        return new Token(18, "square");
    } else if (s.equalsIgnoreCase("E")) {
        return new Token(14, "E");
    } else if (s.equalsIgnoreCase("assign")) {
        return new Token(19, "assign");
    } else if (s.equalsIgnoreCase("write")) {
        return new Token(20, "write");
    } else if (s.equalsIgnoreCase("triangle")) {
        return new Token(21, "triangle");
    }else if (s.equalsIgnoreCase("diamond")) {
        return new Token(22, "diamond");
    } else if (s.equalsIgnoreCase(".")) {
        return new Token(23, ".");
    }else if (s.equalsIgnoreCase("oval")) {
        return new Token(24, "oval");
    }else if (s.equalsIgnoreCase("curve")) {
        return new Token(25, "curve");
    }else if (s.equalsIgnoreCase("wave")) {
        return new Token(26, "wave");
    }else if (s.equalsIgnoreCase("lines")) {
        return new Token(27, "lines");
    }else if (s.equalsIgnoreCase("erase")) {
```

```
            return new Token(28, "erase");
        }
        else if (s.equalsIgnoreCase("save")) {
            return new Token(29, "save");
        }
        else if (s.equalsIgnoreCase("open")) {
            return new Token(30, "open");
        } else if (s.equalsIgnoreCase("rec")) {
            return new Token(31, "rec");
        }else if (s.equalsIgnoreCase("process")) {
            return new Token(32, "process");
        }
        else if (s.equals("arrow")) {
            return new Token(33, "arrow");
        }else if (s.equals("curves")) {
            return new Token(34, "curves");
        }else {
            return new Token(15, s);
             return new Token(15, "error");
        }
    }
```

Appendix B

Java Front End Parsing Phase

Parsing puts together a phrase of the source programme and resolves the issue of expressive power of the language of context-free grammar. The parser takes token generated by the lexical stages, to find a pattern. It will then call the required actions by defining the syntax of the language.

B.1 Paring

Class **Mypair** enables the user to input more than one drawing command in the command prompt. MyParser class resolves user inputs input syntax of the language until the user commands are not empty.

```
/* MyPair class check the whole input and
 pick the first statement to resolve .*/

   static class MyPair {
        AbstractStatement first;
        ArrayList<Token> second;
        public MyPair(AbstractStatement f, ArrayList<Token> s) {
            first = f;
            second = s;
        }
    }
```

B.2 Parsing

MyParser class resolves user inputs arraylist calling the parser function recursively. Parser function checks for correct syntax and semantic and call the output/art to be generated.

```
/* MyParser function*/
   public static StatementList Myparser(ArrayList<Token> a) {
        StatementList b = null;
        MyPair p = parser(a);
        if (!p.second.isEmpty()) {
        return new StatementList(p.first , Myparser(p.second));
        }
        return new StatementList(p.first , b);
    }

/************************************************/

/* parser function*/
public static Mypair parser(ArrayList<Token> L) {
        if (!L.isEmpty()) {
            if (L.get(0).type == 16) {
                String y = L.get(2).Value;
                S = new Mypair(new Zoomin(new Location(y)), L);
            } else if (L.get(0).type == 14 || L.get(0).type == 2 ||
                L.get(0).type == 3 || L.get(0).type == 4
                || L.get(0).type == 5
                || L.get(0).type == 6 || L.get(0).type == 7
                || L.get(0).type == 8
                || L.get(0).type == 9) {
                String n = L.get(0).Value;
                S = new Mypair(new Zoomin(new Location(n)), L);
            } else if (L.get(0).type == 11) {
                String y = L.get(2).Value;
                S = new Mypair(new Circle(new Point(y)), L);
            } else if (L.get(0).type == 24) {
                String y = L.get(2).Value;
                S = new Mypair(new Ellipse(new Point(y)), L);
            } else if (L.get(0).type == 22) {
                String y = L.get(2).Value;
                S = new Mypair(new Diamond(new Point(y)), L);
            } else if (L.get(0).type == 21) {
                String x = L.get(2).Value;
                String y = L.get(4).Value;
```

```
      String z = L.get(6).Value;
      S = new Mypair(new Triangle(new Point(x),
    new Point(y), new Point(z)), L);
  } else if (L.get(0).type == 10) {
     S = new Mypair(new Zoomout(), L);
  } else if (L.get(0).type == 18) {
      String w = L.get(2).Value;
      String x = L.get(4).Value;
      String y = L.get(6).Value;
      String z = L.get(8).Value;
      S = new Mypair(new Rectriangle(new Point(w),
      new Point(x),
     new Point(y), new Point(z)), L);
  } else if (L.get(0).type == 12) {
      String y = L.get(2).Value;
      String Z = L.get(4).Value;
      S = new Mypair(new Line(new Point(y),
      new Point(Z)), L);
  } else if (L.get(0).type == 27) {
      ArrayList<Point> p = new ArrayList<>();
      for (int i = 0; i < L.size(); i++) {
       p.add(new Point(L.get(i).Value));
      }
      S = new Mypair(new Lines(
      (new ArrayList(p))), L);
  } else if (L.get(0).type == 25) {
      String x = L.get(2).Value;
      String y = L.get(4).Value;
      String z = L.get(6).Value;
      S = new Mypair(new curve(new Point(x),
     new Point(y), new Point(z)), L);
  } else if (L.get(0).type == 26) {
      String a = L.get(2).Value;
      String b = L.get(4).Value;
      String c = L.get(6).Value;
      String d = L.get(8).Value;
      S = new Mypair(new wave(new Point(a),
      new Point(b),
```

```
    new Point(c), new Point(d)), L);
} else if (L.get(0).type == 34) {
    L.remove(null);
    ArrayList<Point> p = new ArrayList<>();
    for (int i = 0; i < L.size(); i++) {
        p.add(new Point(L.get(i).Value));
        L.remove(0);
    }
    S = new Mypair(new curves((new ArrayList(p))), L);
} else if (L.get(0).type == 19) {
    String y = L.get(2).Value;
    String Z = L.get(4).Value;
    S = new Mypair(new Assign(y, new Point(Z)), L);
} else if (L.get(0).type == 13) {
    String y = L.get(2).Value;
    H = Myparser(L);
    L.clear();
    S = new Mypair(new Define(y, H), L);
} else if (L.get(0).type == 20) {
    String Z = L.get(2).Value;
    L.remove(null);
    String listString = "";
    for (int i = 0; i <= L.size() - 1; i++) {
        listString += L.get(i).Value;
    }
    L.clear();
    S = new Mypair(new Write(new Point(Z),
     listString), L);
} else if (L.get(0).type == 28) {
    L.clear();
    S = new Mypair(new Erase(), L);
} else if (L.get(0).type == 29) {
    String y = L.get(2).Value;
    S = new Mypair(new Save(y), L);
} else if (L.get(0).type == 30) {
    String y = L.get(2).Value;
    S = new Mypair(new Open(y), L);
} else if (L.get(0).type == 31) {
```

```
            String y = L.get(2).Value;
            S = new Mypair(new RecLong(new Point(y)), L);
        } else if (L.get(0).type == 32) {
            String y = L.get(2).Value;
            S = new Mypair(new Process(new Point(y)), L);
        } else if (L.get(0).type == 33) {
            String y = L.get(2).Value;
            String Z = L.get(4).Value;
            S = new Mypair(new Arrow(new Point(y),
            new Point(Z)), L);
        } else if (L.get(0).type == 17) {
            String y = L.get(2).Value;
            S = new Mypair(new Call(y), L);
        } else if (L.get(0).type == 15) {
            L.remove(0);
            S = new Mypair(new NullStatement(), L);
        }

    }
    return S;
}
```

Appendix C

Java Front End Target Code Generation Phase

The codes below represents the classes written in Java for the target art generation. The main class is the AbstractStatement class where all the other classes inherit the *system state* and interpret method. A state class contains a given state of the system. For that, it keeps track for many parameters: cursor position, zoom size, buffer images, statement lists, zoom location maps and other data structures.

C.1 Abstract Class

```
/* @author sandra.fernando*/
/* The main AbstractStatement class,
the parent class of all the other classes */

public abstract class AbstractStatement {
    abstract State interpret(State s);
}
/*************************************************/

/* @author sandra.fernando */
/* The StatementList class is a type of AbstractStatement,
enables multiple input statements execution */

public class StatementList extends AbstractStatement {
    AbstractStatement Command;
    StatementList SL;
    Graphics l;
```

```
    StatementList(AbstractStatement s, StatementList A) {
        Command = s;
        SL = A;
    }

    State interpret(State s)
    {
        State t = Command.interpret(s);
        if (SL == null) {
            return t;
        } else {
            return (SL.interpret(t));
        }

    }
}
/**************************************************/
```

C.2 System State Object

```
/*@author sandra.fernando*/
/* State class keeps track of system properties */

public class State {
    Dimension screenSize =
    Toolkit.getDefaultToolkit().getScreenSize();
    int screenHeight = (int) Math.round(screenSize.getHeight());
    int screenWidth = (int) Math.round(screenSize.getWidth());
    int x = 0, y = 0;
    double zoomsize = screenWidth;
    int zoomlevel = 0;
    String Loc = "NW";
    BufferedImage imageBuffer = new
    BufferedImage(screenWidth, screenHeight,
    BufferedImage.TYPE_INT_RGB);
    Graphics g = imageBuffer.createGraphics();
    Graphics2D g2d = (Graphics2D) g;
```

```
    Map<String, StatementList> StatmentListmap =
     new TreeMap<String, StatementList>();
    Map<String, Integer> XListmap = new TreeMap<String, Integer>();
    Map<String, Integer> YListmap = new TreeMap<String, Integer>();
    Map<Integer, String> ZoomLocationListmap = new
    TreeMap<Integer, String>();

    State(Graphics e, int a, int b, double d, int c, BufferedImage
    myimageBuffer, Graphics2D myg2d, Map<String, StatementList> f,
    String l, Map<String, Integer> XX, Map<String,
     Integer> YY, Map<Integer, String> ZoomLoc) {
        g = e;
        x = a;
        y = b;
        zoomsize = d;
        zoomlevel = c;
        imageBuffer = myimageBuffer;
        g2d = myg2d;
        StatmentListmap = f;
        Loc = l;
        XListmap = XX;
        YListmap = YY;
        ZoomLocationListmap = ZoomLoc;

    }

}
```

C.3 Concrete Classes

Several concrete classes are presented below. The *Point class* resolves points of graphic such as start, end and control points by associating other concrete classes. The *Location class* resolves locations by associating *zoomin class*. Other concrete classes are for various shapes creation and usability functionalities such as *save, help*.

```
 /* @author sandra.fernando */
  /* Point class resolves a given input point. */

public class Point extends AbstractStatement {
```

```java
public String S;

public Point(String P) {
    S = P;
}

Graphics l;

@Override
State interpret(State s) {

    int a = (int) s.zoomsize;

    if (S.equals("C")) {
        s.x = s.x + a / 2;
        s.y = s.y + a / 2;
    }

    if (S.equals("NW")) {
        s.x = s.x;
        s.y = s.y;
    }

    if (S.equals("SW")) {
        s.x = s.x;
        s.y = s.y + a;
    }

    if (S.equals("W")) {
        s.x = s.x;
        s.y = s.y + a / 2;
    }

    if (S.equals("E")) {
        s.x = s.x + a;
        s.y = s.y + a / 2;
    }
```

```
        if (S.equals("S")) {
            s.x = s.x + a / 2;
            s.y = s.y + a;
        }

        if (S.equals("N")) {
            s.x = s.x + a / 2;
            s.y = s.y;
        }

        if (S.equals("SE")) {
            s.x = s.x + a;
            s.y = s.y + a;
        }

        if (S.equals("NE")) {
            s.x = s.x + a;
            s.y = s.y;

        }

        if (s.XListmap.containsKey(S)) {
            s.x = s.XListmap.get(S);
            s.y = s.YListmap.get(S);
        }

        return (new State(s.g, s.x, s.y, s.zoomsize, s.zoomlevel,
        s.imageBuffer, s.g2d, s.StatmentListmap,
        s.Loc, s.XListmap, s.YListmap, s.ZoomLocationListmap));

    }
}
/**************************************************/
 /* @author sandra.fernando */
  /* Location class resolves a given input location
  on the screen mainly with zoomin class */
```

```
public class Location extends AbstractStatement {

  public String S;

  public Location(String P) {
      S = P;
  }

  Graphics l;

  @Override
  State interpret(State s) {
      if (S.equals("C")) {
          s.x = (int) Math.round(s.x + s.zoomsize);
          s.y = (int) Math.round(s.y + s.zoomsize);
          s.Loc = "C";
      }

      if (S.equals("NW")) {
          s.x = (int) Math.round(s.x);
          s.y = (int) Math.round(s.y);
          s.Loc = "NW";
      }
      if (S.equals("SW")) {
          s.x = (int) Math.round(s.x);
          s.y = (int) Math.round(s.y + s.zoomsize * 2);
          s.Loc = "SW";
      }

      if (S.equals("W")) {
          s.x = (int) Math.round(s.x);
          s.y = (int) Math.round(s.y + s.zoomsize);
          s.Loc = "W";
      }

      if (S.equals("E")) {
          s.x = (int) Math.round(s.x + s.zoomsize * 2);
          s.y = (int) Math.round(s.y + s.zoomsize);
```

```
            s.Loc = "E";
        }

        if (S.equals("S")) {
            s.x = (int) Math.round(s.x + s.zoomsize);
            s.y = (int) Math.round(s.y + s.zoomsize * 2);
            s.Loc = "S";
        }

        if (S.equals("N")) {
            s.x = (int) Math.round(s.x + s.zoomsize);
            s.y = (int) Math.round(s.y);
            s.Loc = "N";
        }
        if (S.equals("SE")) {
            s.x = (int) Math.round(s.x + s.zoomsize * 2);
            s.y = (int) Math.round(s.y + s.zoomsize * 2);
            s.Loc = "SE";
        }

        if (S.equals("NE")) {
            s.x = (int) Math.round(s.x + s.zoomsize * 2);
            s.y = (int) Math.round(s.y);
            s.Loc = "NE";
        }

        if (s.XListmap.containsKey(S)) {
            s.x = (int) Math.round(s.XListmap.get(S));
            s.y = (int) Math.round(s.YListmap.get(S));
        }

        return (new State(s.g, s.x, s.y, s.zoomsize, s.zoomlevel,
        s.imageBuffer, s.g2d, s.StatmentListmap,
        s.Loc, s.XListmap, s.YListmap, s.ZoomLocationListmap));

    }

}
```

```
/**************************************************/
```

Line class resolves line command such as *Line W E*. The line class interact with point class to resolve its start and end points. The line class interprets the drawing of a line based on the user input points. The line class also returns a new state to the system with changed system statues.

```
/*@author sandra.fernando*/
/* Line class creates a target line */

public class Line extends AbstractStatement {
    public Point start;
    public Point end;
    Graphics g;
    public Line(Point s, Point e) {
        start = s;
        end = e;
    }
    @Override
    State interpret(State s) {
        int startx = 0, starty = 0, endx = 0, endy = 0;
        int myx = s.x;
        int myy = s.y;
        start.interpret(s);
        startx = s.x;
        starty = s.y;
        s.x = myx;
        s.y = myy;
        end.interpret(s);
        endx = s.x;
        endy = s.y;
        Graphics2D g2d = (Graphics2D) s.g;
        g2d.setStroke(new BasicStroke(3));
        s.g.setColor(Color.BLACK);

        s.g.drawLine(startx, starty, endx, endy);

        return (new State(s.g, myx, myy, s.zoomsize,
        s.zoomlevel, s.imageBuffer, s.g2d, s.StatmentListmap, s.Loc,
```

```
        s.XListmap, s.YListmap, s.ZoomLocationListmap));

    }
}

/*************************************************/

/* @author sandra.fernando*/
/* Lines class creates one or many lines*/

public class Lines extends AbstractStatement {

    public ArrayList<Point> points = new ArrayList<>();
    Graphics g;
    public Lines(ArrayList<Point> p) {
        points = p;
    }

    @Override
    State interpret(State s) {
        int i = 0;
        ArrayList<Integer> x = new ArrayList<>();
        ArrayList<Integer> y = new ArrayList<>();

        Graphics2D g = s.imageBuffer.createGraphics();
        g.setColor(Color.BLACK);
        g.setStroke(new BasicStroke(3));

        int myx = s.x;
        int myy = s.y;

        for (Point p : points) {
            p.interpret(s);
            x.add(s.x);
            y.add(s.y);
            s.x = myx;
            s.y = myy;
        }
```

```
        GeneralPath polygon = new GeneralPath
        (GeneralPath.WIND_EVEN_ODD, x.size());
        polygon.moveTo(x.get(0), y.get(0));

        for (int index = 1; index < x.size(); index++) {
            if (x.get(index) == 0) {
                polygon.lineTo(x.get(index) + 3, y.get(index));
            } else if (y.get(index) == 600) {

                polygon.lineTo(x.get(index), y.get(index) - 3);
            } else if (x.get(index) == 600) {

                polygon.lineTo(x.get(index) - 3, y.get(index));
            } else if (y.get(index) == 0) {

                polygon.lineTo(x.get(index), y.get(index) + 3);
            } else {
                polygon.lineTo(x.get(index), y.get(index));
            }
        };
        g.draw(polygon);
        return (new State(s.g, myx, myy, s.zoomsize,
        s.zoomlevel, s.imageBuffer,
         s.g2d, s.StatmentListmap, s.Loc,
         s.XListmap, s.YListmap, s.ZoomLocationListmap));

    }

}

/***************************************************/

    /* @author sandra.fernando */
 /* Define class  enables a user to create and
  define an image using multiple commands*/

public class Define extends AbstractStatement {
```

```
    static String name;
    static StatementList S;

    Define(String s, StatementList A) {
        name = s;
        S = A;
    }
    Graphics l;
    State interpret(State s) {
        Map<String, StatementList> StatmentListmapcopy = new
        TreeMap<String, StatementList>();

        StatmentListmapcopy = s.StatmentListmap;
        StatmentListmapcopy.put(name, S);

        return (new State(s.g, s.x, s.y, s.zoomsize,
        s.zoomlevel, s.imageBuffer, s.g2d,
        s.StatmentListmap, s.Loc, s.XListmap,
        s.YListmap, s.ZoomLocationListmap));

    }
}

/*************************************************/

 /* @author Sandra.fernando*/
 / * Call class returns user defined images
 from the state object list. */

public class Call extends AbstractStatement {

    public String S;
    public Call(String s) {
        S = s;
    }

    Graphics l;
```

```
    @Override
    State interpret(State s) {
        Graphics2D g2d = (Graphics2D) s.g;
        g2d.setStroke(new BasicStroke(3));
        s.g.setColor(Color.BLACK);
        if ((s.StatmentListmap.containsKey(S))) {
            s.StatmentListmap.get(S).interpret(s);
            s.x = 0;
            s.y = 0;
            s.zoomlevel = 0;
            s.zoomsize = 600;
            s.Loc = "NW";
        }
        return (new State(s.g, s.x, s.y, s.zoomsize, s.zoomlevel,
        s.imageBuffer, s.g2d, s.StatmentListmap, s.Loc, s.XListmap,
        s.YListmap, s.ZoomLocationListmap));

    }

}
/*************************************************/

/* @author sandra.fernando*/
/* Curve class creates a curve*/

public class curve extends AbstractStatement {
    public Point start;
    public Point mid;
    public Point end;
    Graphics g;
    public curve(Point s, Point m, Point e) {
        start = s;
        mid = m;
        end = e;
    }


    @Override
```

```
    State interpret(State s) {
        Graphics2D g = s.imageBuffer.createGraphics();
        g.setStroke(new BasicStroke(3));
        g.setColor(Color.BLACK);
        int startx = 0, starty = 0, midx = 0, midy = 0,
         endx = 0, endy = 0;
        int myx = s.x;
        int myy = s.y;
        start.interpret(s);
        startx = s.x;
        starty = s.y;
        s.x = myx;
        s.y = myy;
        mid.interpret(s);
        midx = s.x;
        midy = s.y;
        s.x = myx;
        s.y = myy;
        end.interpret(s);
        endx = s.x;
        endy = s.y;
        s.x = myx;
        s.y = myy;

        QuadCurve2D q = new QuadCurve2D.Float();
        q.setCurve(startx, starty, midx, midy, endx, endy);
        g.draw(q);
        return (new State(s.g, myx, myy, s.zoomsize,
        s.zoomlevel, s.imageBuffer, s.g2d, s.StatmentListmap,
        s.Loc, s.XListmap, s.YListmap, s.ZoomLocationListmap));
    }
}

/**************************************************/


  /* @author sandra.fernando */
   /* Curves class creates multiple curves as call by the user.  */
```

```java
public class curves extends AbstractStatement {

  public ArrayList<Point> points = new ArrayList<>();
  Graphics g;
  public curves(ArrayList<Point> p) {
      points = p;
  }

  @Override
  State interpret(State s) {

      ArrayList<Integer> x = new ArrayList<>();
      ArrayList<Integer> y = new ArrayList<>();

      Graphics2D g = s.imageBuffer.createGraphics();
      g.setColor(Color.BLACK);
      g.setStroke(new BasicStroke(3));
      int myx = s.x;
      int myy = s.y;

      for (Point p : points) {
          p.interpret(s);
          x.add(s.x);
          y.add(s.y);
          s.x = myx;
          s.y = myy;
      }

      QuadCurve2D q = new QuadCurve2D.Float();

      for (int i = 0; i < x.size() - 2; i += 3) {
          q.setCurve(x.get(i), y.get(i), x.get(i + 1),
          y.get(i + 1), x.get(i + 2), y.get(i + 2));
          g.draw(q);

      }
```

```
        return (new State(s.g, myx, myy, s.zoomsize, s.zoomlevel,
        s.imageBuffer, s.g2d, s.StatmentListmap,
        s.Loc, s.XListmap, s.YListmap, s.ZoomLocationListmap));

    }

}


  /*************************************************/

/* @author sandra.fernando*/
/* Zoomin class takes the cursor in to a new cell
and change zoom size*/


public class Zoomin extends AbstractStatement {
    public Location P;
    public Zoomin(Location p) {
        P = p;
    }
    Graphics l;
    @Override
    State interpret(State s) {
        Graphics2D g2d = (Graphics2D) s.g;
        g2d.setStroke(new BasicStroke(6));
        s.g.setColor(Color.BLACK);
        s.g2d.setBackground(Color.WHITE);
        int x = 0;
        x = (int) Math.round(s.zoomsize / 3);
        s.zoomsize = s.zoomsize / 3;
        s.zoomlevel = s.zoomlevel + 1;
        P.interpret(s);
        s.ZoomLocationListmap.put(s.zoomlevel, s.Loc);
        g2d.setStroke(new BasicStroke(3));
        s.g.setColor(Color.BLACK);

        s.g.drawLine(s.x, (int) Math.round(s.y + (x / 3)), (int)
```

```
        Math.round(s.x + x), (int) Math.round(s.y + (x / 3)));
        s.g.drawLine(s.x, (int) Math.round(s.y + (x / 3) * 2), (int)
        Math.round(s.x + x), (int) Math.round(s.y + (x / 3) * 2));
        s.g.drawLine((int) Math.round(s.x + (x/ 3)), s.y, (int)
        Math.round(s.x + (x / 3)), (int) Math.round(s.y + x));
        s.g.drawLine((int) Math.round(s.x + (x / 3) * 2), s.y, (int)
        Math.round(s.x + (x / 3) * 2), (int) Math.round(s.y + x));



        return (new State(s.g, s.x, s.y, s.zoomsize, s.zoomlevel,
        s.imageBuffer,s.g2d, s.StatmentListmap,
        s.Loc, s.XListmap, s.YListmap, s.ZoomLocationListmap));

    }
    }

/**************************************************/
/* @author Sandra.fernando */
/* Zoomout class changes the cursor position
and changes the zoom level */

public class Zoomout extends AbstractStatement {
    public Zoomout() {
    }

    Graphics l;

    @Override
    State interpret(State s) {
        Voice voice;
        VoiceManager vm = VoiceManager.getInstance();
        voice = vm.getVoice(VOICENAME);
        voice.allocate();

        s.Loc = s.ZoomLocationListmap.get(s.zoomlevel);

        if (s.Loc.equals("C")) {
```

```
        s.x = (int) Math.round(s.x - s.zoomsize);
        s.y = (int) Math.round(s.y - s.zoomsize);
    }

    if (s.Loc.equals("NW")) {
        s.x = (int) Math.round(s.x);
        s.y = (int) Math.round(s.y);
    }
    if (s.Loc.equals("SW")) {
        s.x = (int) Math.round(s.x);
        s.y = (int) Math.round(s.y - s.zoomsize * 2);
    }

    if (s.Loc.equals("W")) {
        s.x = (int) Math.round(s.x);
        s.y = (int) Math.round(s.y - s.zoomsize);
    }

    if (s.Loc.equals("E")) {
        s.x = (int) Math.round(s.x - s.zoomsize * 2);
        s.y = (int) Math.round(s.y - s.zoomsize);
    }

    if (s.Loc.equals("S")) {
        s.x = (int) Math.round(s.x - s.zoomsize);
        s.y = (int) Math.round(s.y - s.zoomsize * 2);
    }

    if (s.Loc.equals("N")) {
        s.x = (int) Math.round(s.x - s.zoomsize);
        s.y = (int) Math.round(s.y);
    }
    if (s.Loc.equals("SE")) {
        s.x = (int) Math.round(s.x - s.zoomsize * 2);
        s.y = (int) Math.round(s.y - s.zoomsize * 2);
    }
    if (s.Loc.equals("NE")) {
        s.x = (int) Math.round(s.x - s.zoomsize * 2);
```

```
            s.y = (int) Math.round(s.y);
        }

        int x = 0;
        x = (int) Math.round(s.zoomsize * 3);
        s.zoomsize = s.zoomsize * 3;
        if (s.zoomlevel > 0) {
            s.ZoomLocationListmap.remove(s.zoomlevel);
        }
        s.zoomlevel = s.zoomlevel - 1;

        if (s.zoomlevel == 0) {
            s.x = 0;
            s.y = 0;
            s.Loc = "NW";
            s.zoomsize = 600;
            // s.zoomlevel = s.zoomlevel + 1;
            voice.speak(" You cannot zoomout further");
        }
        voice.speak(" Zoomout Level is " + s.zoomlevel);
        voice.speak(" Zoomout Location is " +
        s.ZoomLocationListmap.get(s.zoomlevel));
        s.g.setColor(Color.BLACK);

        return (new State(s.g, s.x, s.y, s.zoomsize, s.zoomlevel,
        s.imageBuffer, s.g2d, s.StatmentListmap,
       s.Loc, s.XListmap, s.YListmap, s.ZoomLocationListmap));

    }
}

/***************************************************/
```

C.4 Main Interface class

The main interface class runs the prototype connecting its associate concrete classes and parsing classes. It is built with Java *awt and swing* packages. A free *text to speech, graphics, java io* packages are used to operate the keyboard modality.

```
/* @author Sandra.Fernando */
/* Main Interface class for keyboard input modality */

public class SETUP09FinalYear extends JPanel implements KeyListener {

    BufferedImage imageBuffer = new BufferedImage(630, 630,
    BufferedImage.TYPE_INT_ARGB);
    Graphics buff = imageBuffer.createGraphics();
    Graphics2D g = (Graphics2D) imageBuffer.getGraphics();
    Graphics2D g2d = (Graphics2D) buff;
    State oldstate = new State(g, 0, 0, 600, 0, imageBuffer, g2d, new
    TreeMap<String, StatementList>(),
    "N", new TreeMap<String, Integer>(), new TreeMap<String,
    Integer>(), new TreeMap<Integer, String>());
    String pretext = "";
    String text = "";
    Voice voice;
    VoiceManager vm = VoiceManager.getInstance();
    private static final String VOICENAME = "kevin16";
    String starttext = "Your command is  ";
    static JTextArea inputCommands = new JTextArea(2, 20);
    static JTextField inputtext = new JTextField(10);
    ArrayList<String> textinputArray = new ArrayList<>();
    boolean up = false, down = false, left = false,
    right = false, space = false, center = false;



    public static void main(String a[]) {
        JFrame frame = new JFrame("Test Game");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        SETUP09FinalYear gamePanel = new SETUP09FinalYear();
        frame.getContentPane().add(gamePanel,
        BorderLayout.CENTER);

        inputtext.requestFocus();
        frame.getContentPane().add(inputtext,
        BorderLayout.NORTH);
```

```java
    frame.getContentPane().add(inputCommands,
    BorderLayout.SOUTH);
    frame.setSize(602, 700);

    gamePanel.setBackground(Color.white);
    frame.setLocationRelativeTo(null);
    frame.setVisible(true);
    frame.setAlwaysOnTop(true);

    inputtext.addKeyListener(gamePanel);

}

public void paintComponent(Graphics g) {

    super.paintComponent(g);
    Graphics2D g2d = (Graphics2D) g;
    g2d.setStroke(new BasicStroke(3));
    g.setColor(Color.BLACK);

    VoiceManager vm = VoiceManager.getInstance();
    voice = vm.getVoice(VOICENAME);
    voice.allocate();
    inputtext.setText("");
    voice.speak("Input your command?");
     g.drawImage(imageBuffer, 0, 0, this);

}

public void keyPressed(KeyEvent e) {
    startTime = System.nanoTime();
    switch (e.getKeyCode()) {
        case KeyEvent.VK_BACK_SLASH:
            text = "zoomout";
            pretext = "";
            input();
            break;
```

```
        case KeyEvent.VK_ENTER:
            text = inputtext.getText();
            pretext = "";
            input();
            break;
        case KeyEvent.VK_8:
            position();
            break;
        case KeyEvent.VK_9:
            layoutshapes();
            break;
        case KeyEvent.VK_ESCAPE:
            new Help();
            break;
        case KeyEvent.VK_UP:
            up = true;
            break;
        case KeyEvent.VK_DOWN:
            down = true;
            break;
        case KeyEvent.VK_LEFT:
            left = true;
            break;
        case KeyEvent.VK_RIGHT:
            right = true;
            break;
        case KeyEvent.VK_SPACE:
            space = true;
            break;
        case KeyEvent.VK_ALT:
            center = true;
            break;
    }

    if (left == true && space == true) {
        pretext = "zoomin location to ";
        text = "NW";
        input();
```

```
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
    } else if (left == true && space == false) {
        pretext = "zoomin location to ";
        text = "W";
        input();
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
    } else if (up == true && space == true) {
        pretext = "zoomin location to ";
        text = "NE";
        input();
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
    } else if (up == true && space == false) {
         pretext = "zoomin location to ";
        text = "N";
        input();
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
    } else if (right == true && space == true) {
        pretext = "zoomin location to ";
```

```
        text = "SE";
        input();
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
    } else if (right == true && space == false) {
        pretext = "zoomin location to ";
        text = "E";
        input();
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
    } else if (down == true && space == true) {
        pretext = "zoomin location to ";
        text = "SW";
        input();
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
    } else if (down == true && space == false) {
        pretext = "zoomin location to ";
        text = "S";
        input();
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
```

```
    } else if (center == true) {
        pretext = "zoomin location to ";
        text = "C";
        input();
        up = false;
        down = false;
        left = false;
        right = false;
        space = false;
        center = false;
    }

}

void input() {
    voice.allocate();
    inputtext.setText("");
    voice.speak(starttext + pretext + text);
    text2 = inputtext.getText();

    oldstate = Myparser(Token(Tokenizer(text))).interpret(new
    State(g, oldstate.x, oldstate.y, oldstate.zoomsize,
    oldstate.zoomlevel, oldstate.imageBuffer,
     oldstate.g2d, oldstate.StatmentListmap,
     oldstate.Loc, oldstate.XListmap, oldstate.YListmap,
     oldstate.ZoomLocationListmap));

    textinputArray.add(text);
    if (textinputArray.size()
     textinputArray.add("\n");
    }
    inputCommands.setText(textinputArray.toString());

    up = false;
    down = false;
    left = false;
    right = false;
    space = false;
```

```
        center = false;

        repaint();

    }


}

/*************************************************/
```

There are many more concrete classes that are not displayed in the document. For exampe, *save class, help class, open class, arrow class, triangle, square, oval class, Diamond class and process class*

Appendix D

Details of the Survey, Training and Experiments

The Research project started in October 2013. The researcher worked with a few charities and a trust during the research period, to collect initial data and subsequent experiment results from the members, namely: RNIB[1] MACS[2] Beyond Sightloss charity[3] and Thomas Pocklington Trust[4]. The participants were contacted through their respective board of trustees and management.

The main participants of the research project were blind people (early blind, late blind, and partially sighted) and some sighted people. A few participants were under the age of eighteen and their parents accompanied them during the experiments, and some other blind participants were accompanied by their carers. Participants' involvement, time and effort were all voluntarily given.

The research initiated two surveys. The first survey was a set of questionnaires published in the Focus Magazine of MACS charity and some further data was collected at a family weekend in the form of verbal communication and in writing. The research details and benefits were passed through the charity contact point to participants prior to data collection. The data was collected through the charity's contact person therefore no personal data was submitted. The second survey was conducted during one-to-one interviews and participants were informed of the project details and benefits in advance. Consent from participants was verbally obtained before the start of the survey and published results were passed to the charities.

Many different sets of experiments took place after the surveys. The first experiment involved five blind people, the second experiment involved ten blind and ten sighted participants, and another experiment involved eight early and late blind participants and the final experiment had four blind and four sighted participants. The experiments were conducted on a one-on-one basis in different locations. The first step was to explain the purpose, details and potential benefits of the experiment. The second

[1] https://www.rnib.org.uk/
[2] https://macs.org.uk/
[3] https://www.sightlinedirectory.org.uk/Listings/Details/4308/beyond-sightloss
[4] https://www.pocklington-trust.org.uk

stage was to obtain the participants' verbal consent. The actual experiments were conducted thereafter, with system training, system testing and question sessions, before and after the actual system testing had taken place. Observations such as time duration, errors, behaviours and feedback were recorded with the consent of the participants. Publications were sent to some participants and the charity contact persons were informed in other situations.

There were no ethical issues or adverse effects identified during this research phase. No conflict of interest or negative effects such as personal distress were observed.

D.1 Survey 1

The survey 1 was conducted with people from MACS charity members during their family weekend. A presentation followed by a system demonstration was conducted at the annual family meeting. And then the survey answers were collected. The Survey 01 questions are presented below in Table D.1 and Figure A1

Figure A1 demonstrates the flyer of the first survey questions from Focus magazine published by MACS charity in 2013. The image has an introduction to the researchers and their rationale for the research - the need for assistive technology for blind drawing. These are the questions were posted to find out blind and visually impaired people's experience with technology. The questions presented in the flayer are: Do you or does your child use technology and if so, what do you use and for what purpose? For example, what games do you play? Do you use software like JAWS or other applications? Do you or does your child likes to create pictures, images or diagrams, and how do you/they do this? Would you or your child like to be able to use technology to create images and designs? What particular problems or barriers do you face in using technology generally? If you could imagine a tool, which would allow you to express yourself by getting the image in your mind onto a computer, screen, what would it be like? What would work best? Would you prefer to use speech, touch, a keyboard or other methods to direct your ideas?

Technology; a vision for the future?

MACS has recently joined forces with researchers from Goldsmiths, University of London on a very exciting project. Our aim is to explore the ways in which blind and visually impaired people can access technology to express their thoughts for use in art, design and other forms of communication.

Just because a blind person cannot see, it doesn't mean that they don't visualise or imagine designs, diagrams and pictures in a similar way to a sighted person. It does mean, however, that they will need specific tools to help them transfer what they see in their mind's eye onto a screen so that they can share it with others.

Not having such a tool at their disposal, means that blind and visually impaired people's access to learning, and developing certain skills, may be reduced.

As MACS members (parents, adult members, young people and children alike), you could be instrumental in developing this project and we hope that you will share your thoughts and ideas with us. Sandra, the key researcher on this project, will be coming along to our annual Family Weekend in May next year to present information about her progress and to give you the opportunity to participate in the process.

In the meantime, we welcome your thoughts on the following questions. Please have a look and send your answers to **victoria@macs.org.uk** as soon as you can.

Remember, we want to hear any ideas that you have. Be imaginative if you can; anything goes!

Q1. Do you or does your child use technology and if so, what do you use and for what purpose? For example, what games do you play? Do you use software like JAWS or other applications?

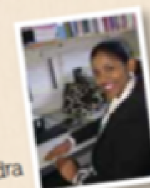
Q2. Do you or does your child like to create picture, images or diagrams and how do you / they do this? Would you or your child like to be able to use technology to create images and designs?

Q3. What particular problems or barriers do you face in using technology generally?

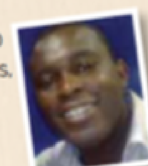
Q4. If you could imagine a tool which would allow you to express yourself by getting the image in your mind onto a computer screen, what would it be like? What would work best? Would you prefer to use speech, touch, a keyboard or other methods to direct your ideas?

We asked Toby his thoughts on some of these questions. This is what he said...

'I'd like to have access to a screen reader that could read images and have software that would help me to draw images using speech recognition. You should be able to speak to create an image or object; so if I say circle, or top left circle, this should appear in the top left of the screen. It should also explain images with the size and colour.'



Sandra Fernando has ten years' experience in the software industry and as a teacher of IT and is now beginning a PhD investigating the development of a computer aided drawing tool for blind learners. Sandra was inspired by MACS member Toby Ott who she came across while teaching at Lewisham and Southwark College.

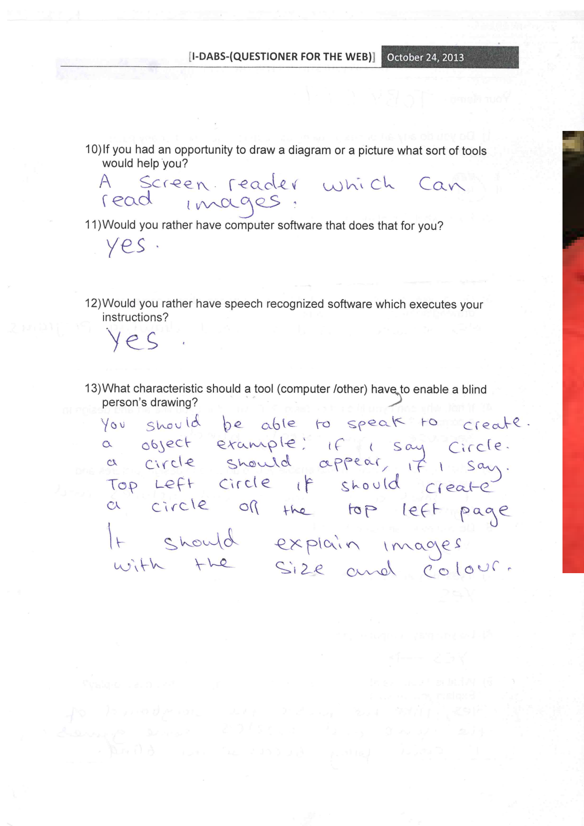


Dr James Ohene-Djan gained a PhD in computer science from Goldsmiths, University of London and is a senior lecturer in Assistive Technologies, Computing and Social Media. He founded the Assistive Technologies Research Group at Goldsmiths which specialises in designing tools and techniques for disabled people.

5

Figure A1: First Survey Questions from Focus Magazine, [MACS, 2019]

Table D.1 shows two images. The left column is a feedback from a participant, and on the right image is an image of a blind participant using a computer device. The image has some questions and answers from a questionnaire presented at the family weekend. The first question was: What characteristic should a computer tool have to enable a blind person's drawing? The answers was "You should be able to speak to create an object. An example: If I say a circle, a circle should appear. If I say top left circle, it should create a circle on the top left page. It should explain images with the size and colour": When asked another question If you have an opportunity to draw a diagram or a picture of what sort of a tool would help you? And the answer was "A screen reader, which can read images?". The participant also answered, "yes" to questions such as: Would you rather have computer software that draws images for you? And would you rather have speech recogniser software, which executes your instructions?


[I-DABS-(QUESTIONER FOR THE WEB)] October 24, 2013

10) If you had an opportunity to draw a diagram or a picture what sort of tools would help you?
A screen reader which can read images.

11) Would you rather have computer software that does that for you?
yes.

12) Would you rather have speech recognized software which executes your instructions?
yes.

13) What characteristic should a tool (computer /other) have to enable a blind person's drawing?
You should be able to speak to create. a object example: if I say circle. a circle should appear, if I say. Top Left circle it should create a circle on the top left page
It should explain images with the size and colour.

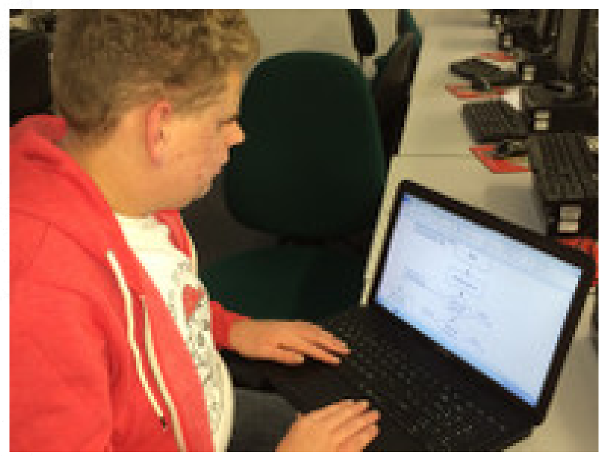


Table D.1: Survey 01: Facts Finding

D.2 Surveying 2

Survey 02 was conducted with people from various charities such as MACS, RNIB, Beyond Sightloss Charity.

Questions

Name:

Age:

Gender:

Type of visual impairment:

- Do you like to create images or diagrams?
- Have you ever produced art?
- Was it easy?
- Did you complete the task of drawing?
- Did you enjoy creating them?
- Was it a digital/computer tool?
- What was the image produced? How did you create them? Did it take time?
- How did you do the correction and validation?
- If you imagine a digital tool that could help you to draw, how would you like it to be? And why?

Table D.2 shows two images. The left and right columns have feedback from two participants during Survey 2. The first participant answered "yes" to questions such as: Do you like to create images or diagrams? Have you ever produced art? Do you complete the task of drawing? Did you enjoy creating them? And the participant answered "no" to questions such as: Was drawing easy? Was it a digital/computer tool? Another open-ended question asked was, what was the image produced? How did you create them? Did it take time? The answer was "drawings of houses with children as I was a child protection social worker. Some people don't say what is going on, but they draw" The second participant answered "yes" to questions such as do you like to create images or diagrams? Have you ever produced art? Did you complete the task of drawing? Did you enjoy creating them? Was drawing easy? And the participant answered "no" to questions: Is it a digital/computer tool? Another open-ended question asked was, What was the image produced? How did you create them? Did it take time? The answer was "Bonham Geometry device, boards, diagrams boards 8 to 16 inches."

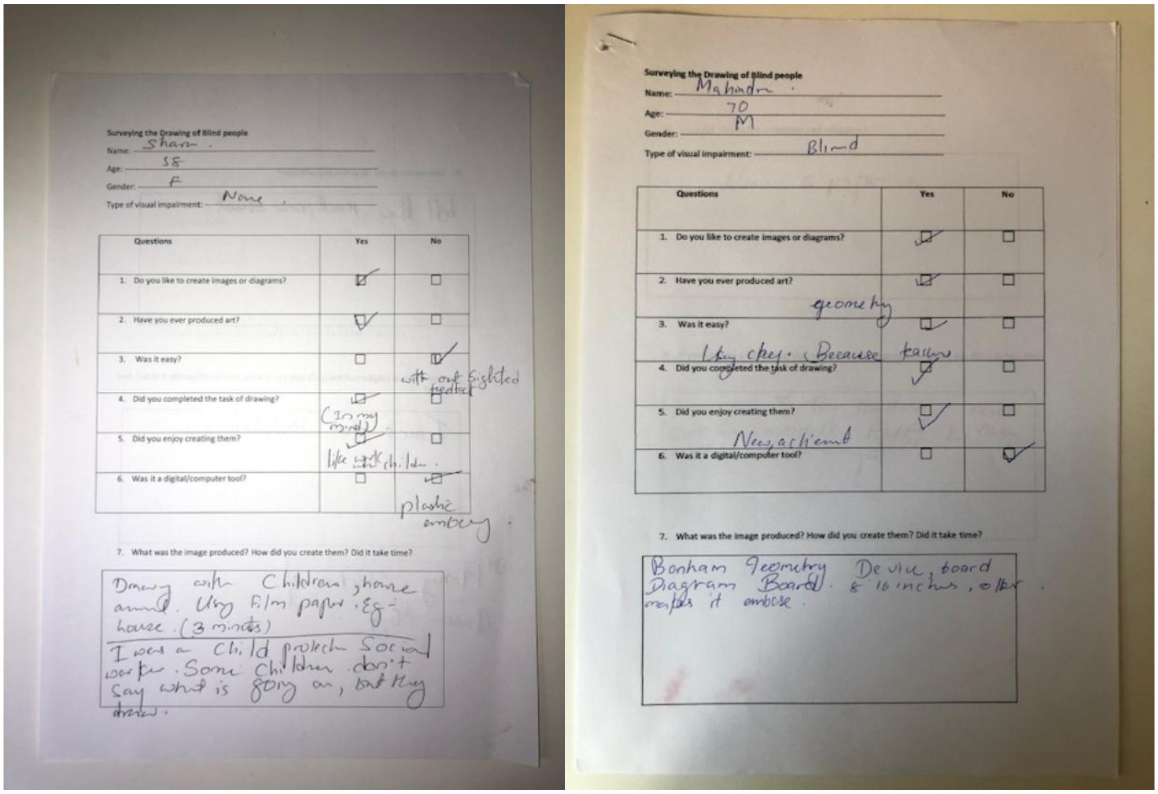


Table D.2: Survey 02: Facts Finding

D.3 Study 1 - Evaluation of a Screen Navigation System

D.3.1 User Guide

This experiment was designed to find out the blind people's understanding of a virtual navigation system of a computer-aided drawing tool. There was a need to find out whether going through a virtual environment generates a navigation route in a blind person's mind; specifically, its efficiency and accuracy. Efficiency was measured by task completion time; accuracy was measure by error rate and tactile maps measure accuracy.

Notes given to the Participants - Understanding the system SETUP09

SETUP09 is software that runs on a computer. When the program starts, it divides the screen in a 3*3 grid system using symmetric locations. It is based on compass directions such as north, south, east, west, northeast, northwest, southeast, and southwest. It also has the centre region named centre. Each region/location has nine points such as north, south, east, west, northwest, northeast, southeast, and southwest. Each location can be further subdivided a into a 3*3 grid system according to compass based directions. This division can continue as many times as the user requires to find a specific smaller location on the screen. Users can enter one or many commands at the user prompt to manipulate an image. For the purposes of this experiment only some commands are observed to answer the question that going through a virtual environment generates a cognitive map in a blind person's mind.

How is the experiment designed - to be taken place?

- Participants will enter some commands into the system to find target locations as given by the observer. Observations on accuracy, time taken and errors made are recorded.
- Participants will then show the target locations of each of those commands on a tactile grid paper.
- The observer will compare the accuracy of SETUP09 target locations against tactile grid target locations for the given commands.
- Finally the observer evaluates the accuracy of SETUP09 commands with the task given.

Commands that are used for the experiment are:

- To get the focus of the screen: Zoomin [name of the area]
- To extract the focus out of an area: Zoomout ?

Equipment for experiment 1

- A computer
- A Keyboard
- Tactile grid paper

Example Commands

- To get the focus on the Center region: zoomin C
- To get the focus on the North East region: zoomin NE
- To get the focus on the Center region and again the focus of its Center region: zoomin C, zoomin C (one command at the prompt or all at once separated by a comma)
- To get the focus on the North region and again the focus of its West region and again the focus of its Center region: zoomin N, zoomin W, zoomin C (one command at the prompt or all at once separated by a comma)

Training Activity

Practice the above commands and familiarise yourself with the system for the next 30 minutes. Input these commands to the system

- User Command1:

- User Command 2:
- User Command 3:
- User Command 4:

These commands will be revealed at the experiment. Show your finding to the organiser. Thank you!

D.3.2 Observer Guide Used in Study 1

Instruction and user manual (recorded and written instruction is available)

Level 01

- Feel tactile image 1. How many areas on image 01? What are they?
- Do you have any other observations? How can you best describe the image?
- Feel tactile image 2.What are the similarities and difference between image 1 and image 2? Do you have any other Observations?
- What can you tell about the size of those areas?

Instruction explanation

Those sizes determine the size of the shape or image, and it is named zoom size. At the start of the programme, the zoom size is equal to the output interface size and hence zoom level is 0. The zoom level increases as you go to smaller areas on the screen. The screen measurements are starting from the top left corner. The left most starting point has the x coordinate and y coordinates set to 0. As you travel through x-axis and y-axis the coordinates increases.

- What is the area highlighted in image 3?
- What is the area highlighted in image 4?

Level 02

- What do you understand by zoomin?
- What do you understand by zoomout?
- What is meant by zoomin Center?
- Feel image 1. What is the focus area of zoomin Center?
- Now you are focused on Center area. What will be the focus area of zoomout command?

Level 03

- What would be the command to draw an image on the area highlighted on image 5?
- You are now focused on South West. What is the command to go to the Center area of South West?
- What is the value of your zoom level?
- List the commands to go back to zoom level 0? /start point.
- What is the area highlighted in image 6?
- What is the area highlighted in image 7?

System commands

- Open system SETUP09 and type using a keyboard
- Type: zoomin C
- Type: yes
- Type: zoomin N
- Type: yes
- What area is in focus? Show using a tactile map.
- Type: zoomout
- What area is in focus? Show using tactile image.
- Type the commands on the keyboard to get the focus of the area on image 6?

D.4 Study 2 - Evaluation of Graphics manipulation Technique (BVI and Sighted)

D.4.1 Testing Event Information

An event was organised at a MACS[1] family weekend for system testing. Find the event flyer below in figure A2 shows the experiment event details created on Eventbrite during a family weekend.

[1]https://macs.org.uk/

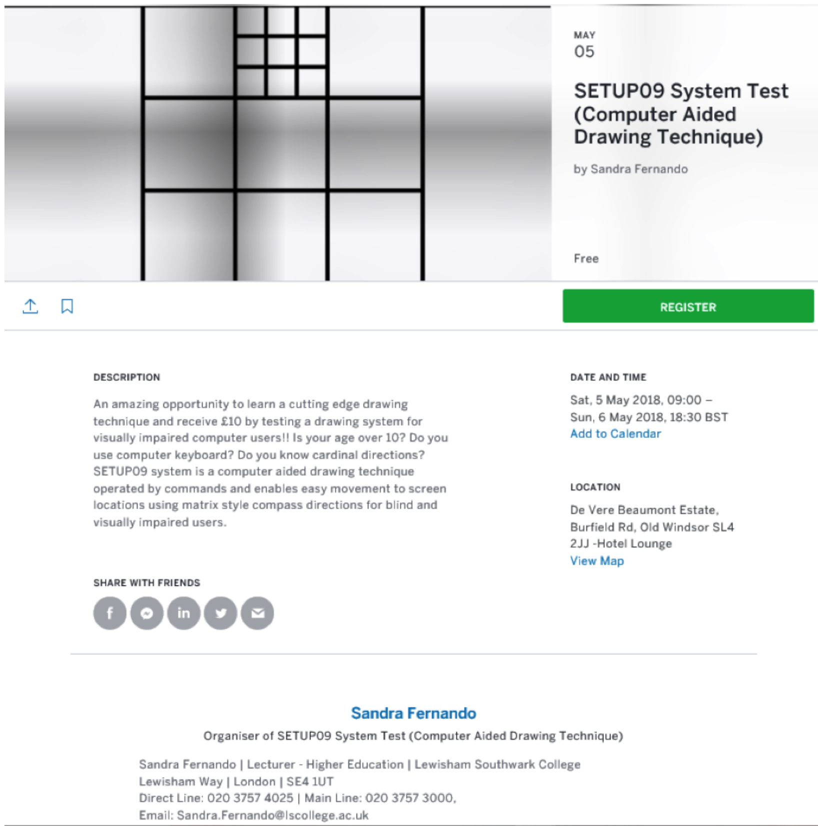


Figure A2: The Flyer for Experiment 2.

D.4.2 SETUP09 Navigation System Training 2.1

Level 01

- Feel tactile image 1, 2. How many areas on image 01 and 02? What are they? Do you have any other observations? What can you tell about the size of those areas?
- What do you understand by zoomin?
- What do you understand by zoomout?
- Feel image 1. What is the focus area when you zoomin to Center/ alt key?
- Now you are focused on Center area. What will be the focus area of zoomout command/backslash key?
- What is the area highlighted area in image 2?
- What is the value of your zoom level?
- List the commands you would type to go back to zoom level 0? /start point.

Level 02

- What is the command to draw a circle in the north area?

- What is the command to draw a square in the centre area?
- What are the commands to draw a triangle on the south area?
- How do you draw a triangle with lines command?

Here is an image of a an actor, feel it in the swell paper. Now lets draw it using the SETUP09 system

- Define Actor1
- Labelling the drawing to use it later
- Up arrow, circle NW,backslash key, Draw the head and come out of the north area
- Alt key, Go into the centre area
- line N S, Draw a stick to represent the body with a line
- line N W, Draw left arm
- line N E, Draw right arm
- backslash key, Come out of center area
- backslash key, Go into south area, line N SW
- Draw left leg in a line, line N SE
- Draw right leg with a line

Learning the System

Open system SETUP09 and type the commands below using a keyboard.

- Type: alt key
- Type: up arrow key
- What area is in focus? Show using tactile image 3.
- Type the commands on the keyboard to get to the focus of the area on image 4?

D.4.3 Study 2 - Data and Results

Figure A3 shows a picture of a table that has timings taken by both control and experiment groups of 20 blind and sighted people to draw shapes, images and flowcharts and also timings taken to recognise images.

Group	Drawing Task 1	Time taken (task 01) Instructed	Time taken (task 02) Non-Instructed	Time taken (task 03) Recognition
Experiment	Triangle	0.22	0.29	0.31
	Triangle	0.43	0.17	0.21
	Triangle	0.30	0.20	0.45
	Triangle	0.37	0.30	0.20
	Triangle	1.46	0.57	0.45
	Triangle	0.26	0.35	0.30
	Triangle	1.47	1.49	0.30
	Triangle	2.17	0.45	0.25
	Triangle	1.17	0.46	0.22
	Triangle	0.33	0.30	0.30
Avg		0.82	0.46	0.30
Experiment	House	1.04	2.30	1.00
	House	1.10	1.46	0.43
	House	0.28	1.50	0.30
	House	1.04	2.03	0.33
	House	2.56	1.45	0.12
	House	3.52	2.17	1.02
	House	3.00	3.11	1.56
	House	3.47	3.10	1.30
	House	3.24	2.15	1.33
	House	2.00	1.30	0.20
Avg		2.13	2.06	0.76
Experiment	DFD	5.06		3.00
	DFD	5.06		1.46
	DFD	3.50		0.30
	DFD	5.07		3.00
	DFD	8.48		2.01
	DFD	10.20		2.09
	DFD	9.00		2.51
	DFD	8.50		3.00
	DFD	6.38		2.55
	DFD	5.00		1.45
Avg		6.63		2.14

Group	Drawing Task 1	Time taken (task 01) Instructed	Time taken (task 02) Non-Instructed	Time taken (task 03) Recognition
Control	Triangle	0.28	1.15	0.20
	Triangle	0.43	0.19	0.22
	Triangle	0.37	0.15	0.33
	Triangle	0.27	0.19	0.34
	Triangle	0.24	0.19	0.25
	Triangle	0.40	0.40	0.40
	Triangle	0.50	1.00	0.30
	Triangle	0.10	0.10	0.30
	Triangle	0.20	0.20	0.34
	Triangle	0.30	0.15	0.12
Avg		0.31	0.37	0.28
Control	House	0.48	0.40	2.00
	House	1.35	2.00	0.17
	House	1.18	2.18	0.10
	House	1.22	2.07	0.10
	House	0.48	3.00	0.40
	House	1.40	1.50	0.54
	House	2.00	1.50	0.50
	House	0.30	1.00	0.16
	House	0.33	1.50	0.20
	House	0.45	1.60	0.30
Avg		0.92	1.68	0.45
control	DFD	3.14		0.33
	DFD	6.00		2.37
	DFD	3.54		0.30
	DFD	2.26		0.50
	DFD	3.50		2.30
	DFD	4.23		1.45
	DFD	7.41		2.00
	DFD	2.00		0.30
	DFD	2.50		0.46
	DFD	3.00		0.38
Avg		3.76		1.04

Figure A3: Experiment Raw Data of Blind and Sighted Individuals

Table D.3 illustrates the the blind and visually impaired people's completion percentage. House and triangle drawings were completed with 100% accuracy, but diagram drawing had six people out of ten completing above 90%.

Completed Percentage	Triangle	House	Diagram
100%	10	10	3
90%	0	0	3
80%	0	0	2
40%	0	0	2

Table D.3: Average Completion Percentages of BVI Individuals

Table D.4 illustrates the timings taken, average timings, P-value of blind group of people and visually impaired a group of people to complete drawing activities.

Participants	Triangle	House	Diagram
P1- No Vision	0.22	1.04	5.06
P2- No Vision	0.43	1.10	5.06
P5- No Vision	1.46	2.56	8.48
P6- No Vision	0.26	3.52	10.20
P7- No Vision	1.47	3.00	9.00
Avg	0.77	2.24	7.56
P3- Some Vision	0.30	0.28	3.50
P4- Some Vision	0.37	1.04	5.07
P8- Some Vision	2.17	3.47	8.50
P9- Some Vision	1.17	3.24	6.38
P10- Some Vision	0.33	2.00	5.00
Avg	0.87	2.01	5.69
P Val	0.8347	0.7725	0.2051

Table D.4: Time Taken by BVI Individuals During Instructed Drawing

Table D.5 illustrates the number of errors made by the blind group of people and the visually impaired group of people in completing drawing activities.

Participants	Triangle	House	Diagram
P1- No Vision	0	0	0
P2- No Vision	0	1	0
P5- No Vision	0	0	0
P6- No Vision	0	0	0
P7- No Vision	1	1	1
P3- Some Vision	0	0	1
P4- Some Vision	0	0	0
P8- Some Vision	0	0	0
P9- Some Vision	0	1	1
P10- Some Vision	0	0	0

Table D.5: Errors Made by BVI during Instructed Drawing

D.5 Study 3 - Comparative Evaluation of Digital System versus Analogue System

D.5.1 Preparatory Work

This emails was sent to various charities, organisations and people in order to recruit participants.

sandra fernando <saanfdo@yahoo.com>

To: info@Thomas Thomas Pocklington Trust.org.uk

Jul 4 at 10:35 AM

To whom it may concern,

I am a researcher at Goldsmiths, University of London. I have been researching on computer-aided scientific drawing systems for blind people for the last six years. I am looking forward to meet some late blind participants that can take part in an experiment. The experiment will take roughly 60 minutes, first the training and then the experiment. I am happy to come to your location to talk about it and conduct the experiment at your convenience.

Rational of the experiment: Art creation is proven to be enhancing the perception, mental ability and life of blind people. The purpose of this experiment is to analyse a digital navigational and drawing technique with different types of blind individuals and their mental load capacity required for art creation. As a result I will introduce a digital technique. Our publication will help other scientists and developers to make correct decisions in designing digital and art production tools for blind people. We will acknowledge you in our publication and your charity will be known to an international audience. The tasks of the experiment include:

- Answer survey questions about use of ICT and drawing
- Draw few 2D shapes using a German embossing film papers and a software system
- Answer feedback questions for the experiment
- I'd like to offer a small incentive to participants at the end of the experiment for their time and participation.

Looking forward to hearing from you !
Kind regards,
Sandra Fernando

D.5.2 SETUP09 System Training 3.1

The participants in study 3 were given a training, on a one-to-one basis on the system commands mentioned below.

Basic Commands

- Help menu for commands: esc key
- To Navigate/zoomin to North: up arrow key
- To Navigate/zoomin to South: down arrow key
- To Navigate/zoomin to West: left arrow key
- To Navigate/zoomin to East: right arrow key
- To Navigate/zoomin to Center: alt key
- To Navigate/zoomin to Northeast: space key + up arrow key

- To Navigate/zoomin to Northwest: space key + right key
- To Navigate/zoomin to Southeast: space key + down arrow key
- To Navigate/zoomin to Southwest: space key + left arrow key
- Find the location of the focus/positioned: number 8 key
- Find the shapes of the focus position: number 9 key
- Zoomout: backslash key
- To save the image: save[filename]
- To open a saved image: open[filename]
- To erase the art of area on focus: erase

Drawing Commands

- To label a point on the screen: assign [labelname] [N, W, S, E, C, NE, NW, SW, SE]
- To draw shape circle: circle [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape oval: oval [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape square: square [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape diamond: diamond [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape curve: curve [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape triangle: triangle [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape rectangle: rec [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape process: process [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape line: line [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint] [any ending point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]

- To draw shape arrow: arrow [any starting point out of N, W, S, E, C,] [any starting point out of N, W, S, E]
- To draw shape of many lines such as a polygon: lines [any number of points out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape of 3-point curve: curve [any 3 of points out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape of 4-point curve: curves [any 4 of points out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To draw shape of many curves: wave [any number of 3 points curves N, W, S, E, C, NE, NW, SW, SE, labelledpoint]
- To write a string on the interface: write [any starting point out of N, W, S, E, C, NE, NW, SW, SE, labelledpoint] yourstring
- To define an object: define [anynamelabel] any number of system commands. Note zoomout and zoomin has to be typed. e.g.: zoomin [location]
- To call a defined object: call: [[namelabel]

D.5.3 Study 3 - Results and Data

Figure A6 shows a picture of a table with time, accuracy and errors of every individual who took part in drawing of a shape, an image and a flowchart in SETUP09 and using a swell paper kit. It displays the time taken to produce drawing, completion percentage, errors of both SETUP09 and paper-kit.

Task	Participants	Time (Swell Paper kit)	Time (SETUP09)	Accuracy % (Swell Paper kit)	Accuracy % (SETUP09)	Errors Made (swell Paper Kit)	Errors Made (SETUP09)
Shape	P1	0.40	0.39	70	100	1	1
	P2	0.27	0.46	90	100	1	0
	P3	0.37	0.43	90	100	0	0
	P4	0.47	1.00	20	100	2	0
	P5	0.30	0.38	90	100	1	0
	P6	0.37	0.13	90	100	1	0
	P7	0.4	0.13	80	100	1	1
	P8	0.45	0.24	90	100	1	0
Image	P1	0.26	1.30	80	100	1	2
	P2	0.30	2.30	80	100	2	0
	P3	1.00	2.40	80	100	1	0
	P4	1.00	2.36	30	100	2	0
	P5	0.28	2.00	95	100	1	1
	P6	1.27	1.33	90	100	1	0
	P7	0.24	1	70	100	2	1
	P8	0.46	2.33	90	100	1	1
FlowChart	P1	3.41	6.50	50	80	5	1
	P2	2.00	5.00	70	100	5	1
	P3	2.25	7.10	70	100	4	1
	P4	5.50	5.00	10	80	5	2
	P5	0.38	5.20	70	80	5	1
	P6	3.43	4.52	70	100	4	1
	P7	3.23	2.45	10	80	5	3
	P8	4.01	4.06	90	100	2	2

Figure A4: Experiment 3 Raw Data of Early and Late Blind Individuals

Table A6 illustrates average accuracy of early and late blind participants when using SETUP09. Both early and late groups recorded a hundred per cent accuracy when shape and Image creation, but 90% was recorded with chart drawing.

Group	Shape	Image	Chart
Early Blind Group	100	100	90
Late Blind Group	100	100	90

Table D.6: Average Accuracy of Early and Late Blind Participants SETUP09

Table D.6 illustrates the average accuracy of early and late blind participants when using SETUP09. Both early and late groups recorded a 100% accuracy with shape and image creation, but 90% was recorded with chart drawing.

Group	Shape	Image	Chart
Early Blind Group	68	68	50
Late Blind Group	88	86	60

Table D.7: Average Accuracy of Early vs. Late Blind Participants with Paper toolkit

Table D.7 illustrates average accuracy of early and late blind participants with the paper toolkit. The early blind group achieved 68% when drawing shape and image but 50%when drawing the chart. The late blind group achieved 88% when drawing the shape and 86% when drawing the image but 60% when of accuracy drawing the chart.

Table D.8 illustrate the sum of errors made by early and late blind participants. The early blind group made 29, and the late blind group made 25 errors when drawing on the paper toolkit. However,

Group	Paper toolkit	SETUP09
Early Blind Group	29	7
Late Blind Group	25	10

Table D.8: Sum of Errors made by Early versus Late Blind Using Participants

both groups made a low number of errors when using SETUP09, where early blind group made seven errors, and the late blind group made ten errors.

D.5.4 Study 3 - Tasks and Commands

4-Sided Shape Command (for a diamond shape art)

- C, line W N,E,S,W

Table Command

- SE, C, lines W, NW, NE, E, W
- line W SW
- line E SE

Flowchart Commands

- N, oval NW, arrow C S, C, write NW Start, zoomout, zoomout
- C, Process NW, arrow C S, C, write NW Process, zoomout, zoomout
- S, oval NW, arrow C S, C, write NW End, zoomout, zoomout

D.5.5 Commands of the Cell Referencing Task

SETUP09 Line Commands

- C,C, line NW NE

CenterReference Line Commands

- C, NW, SE, SE, assign C1 C, zoomout, zoomout, zoomout, NE,
- SW, SW, assign C2 C, line C1 C2, save lineCenterReference,

D.5.6 Study 5 - SETUP09 Comparison with Digital Drawing Technique

Study 5 is on the use of bespoke software (SETUP09) that had 4 blind and 4 sighted people. To minimise the risk of catching Covid-19 the experiment was conducted on an individual basis with the permission of the participants at their residence. The participant and I observed social distancing, wear protection gear (face mask and gloves) and wipe any surfaces we contacted with including laptops. They have all agreed to meet at the location of the experiment; there was only be the participant and me. The experiment involved the participant in drawing simple art/diagram using the software with the given instructions on a computer/laptop. Several HCI measurements were taken by the observer for future analysis.

An email was sent to all participants detailing what the experiment involves and that to minimise any risk of catching Covid-19 that we follow PHE's protocol, i.e. (1) participants and I to observe social distancing by staying 2 meters apart, and (2) wear protective equipment (face mask) and clean our hands and laptop/computer with antibacterial gel/wipes before and after the use of experiments.

My personal security plan has been arranged with my PhD supervisor (Dr. James Ohene-Djan). I informed him via WhatsApp before I enter the participant's premises and after I leave the premises.

Next section presents images of data collected and IC2D data referenced in study 5. The users were trained using commands discussed in Study 2 above. IC2D experiment Statistics of BVI and sighted Individuals are extracted from the URL below :

```
https://www.slideserve.com/dillian/constructing-images-eyes-free
-a-grid-based-dynamic-drawing-tool-for-the-blind
```

(n=16)	Visually Impaired	Sighted(blindfolded)
Task1a time	6 (average)	7.3 (average)
Task1a confidence	8.1	6.8
Task1a performance	8.8	8.8
Task1b time	8.3	8.8
Task1b confidence	8.1	6.9
Task1b performance	8.3	7.6

Figure A5: Study 5 - IC2D Statistics of BVI and Sighted Individuals.

The image below represents the excel raw data collected at the time of the study. Other statistical data is presented in Chapter 8.

Activity	vision	time	P Confidence (1-10)	Performance Rate (1-10)	Accuracy	Errors	Memorability (1-10)
Task A	Blind	1.54	8.00	10	100	0	8
	Blind	2.09	7.00	10	100	1	8
	Blind	2.14	7.00	9	90	1	9
	Blind	1.92	8.00	10	100	0	7
B Avg		1.92	7.50	9.75	97.50	2.00	8.00
	Sighted	1.34	9.00	10	100	0	8
	Sighted	1.43	7.00	10	100	0	7
	Sighted	2.00	8.00	10	100	0	8
	Sighted	1.13	8.00	10	100	0	9
S Avg		1.48	8.00	10.00	100.00	0.00	8.00
All Avg		1.70	7.75	9.875	98.75	2	8
Task B - M1	Blind	3.30	8.00	10	100	1	6
	Blind	4.00	7.00	9	90	1	8
	Blind	4.00	7.00	8	80	2	9
	Blind	3.77	8.00	10	100	1	8
B Avg		3.77	7.50	9.25	92.50	5.00	7.75
	Sighted	2.14	8.00	10	100	1	8
	Sighted	2.40	6.00	10	100	1	8
	Sighted	3.10	8.00	10	100	0	8
	Sighted	2.56	7.00	10	100	0	9
S Avg		2.55	7.25	10.00	100.00	2.00	8.25
All Avg		3.16	7.38	9.625	96.25	7	8
Task B - M2	Blind	5.51	7.00	9	90	1	6
	Blind	6.10	7.00	7	70	2	8
	Blind	6.01	7.00	7	70	3	7
	Blind	6.11	8.00	9	90	1	8
B Avg		5.93	7.25	8.00	80.00	7.00	7.25
	Sighted	4.10	8.00	10	90	2	7
	Sighted	4.22	6.00	9	90	2	7
	Sighted	5.00	7.00	10	100	1	7
	Sighted	4.40	7.00	10	100	1	9
S Avg		4.43	7.00	9.75	95.00	6.00	7.50
Avg		5.18	7.13	8.875	87.5	13	7.375

Figure A6: Study 5 - SETUP09 Raw Data and Statistics of BVI and Sighted Individuals

Bibliography

[ABBYY, 2019] ABBYY (2019). What is OCR and OCR technology. https://www.abbyy.com/en-gb/finereader/what-is-ocr/. Last checked on Sep 02, 2019.

[Afonso et al., 2010] Afonso, A., Blum, A., Katz, B. F. G., Arroux, P., Borst, G., and Denis, M. (2010). Structural properties of spatial representations in blind people: Scanning images constructed from haptic exploration or from locomotion in a 3-D audio virtual environment. *Memory and Cognition*, 38(5):591–604.

[Ait-Ameur and Baron, 2006] Ait-Ameur, Y. and Baron, M. (2006). Formal and experimental validation approaches in HCI systems design based on a shared event B model. *International Journal of Software Tools Technololy Transfer*, 8(547–563).

[Amazon, 2020] Amazon (2020). Meet alexa. https://www.amazon.co.uk.

[Amin, 2019] Amin, P. W. O. (2019). Models of Generative Grammar. Technical report, Ibn Rushd College of Education University of Baghdad.

[APH, 2014a] APH (2014a). Draftsman tactile drawing board and film. http://shop.aph.org. Last checked on Aug 19, 2019.

[APH, 2014b] APH (2014b). Quick-draw paper. https://shop.aph.org. Last checked on Aug 16, 2019.

[APH, 2014c] APH (2014c). Swail dot inverter. http://shop.aph.org. Last checked on Aug 19, 2019.

[Appel, 2002] Appel, A. W. (2002). *Modern Compiler Implementation in Java*. Cambridge University Press, New York, NY, USA, 2nd edition.

[Apple, 2019] Apple (2019). Siri. https://www.apple.com/uk/siri/. Last checked on Sep 02, 2019.

[Apple, 2020] Apple (2020). Accessibility.

[Arnheim, 1990] Arnheim, R. (1990). Perceptual aspects of art for the blind. *The Journal of Aesthetic Education*, 24(3):57–65.

[Bach-y-Rita, 2004] Bach-y-Rita, P. (2004). Tactile sensory substitution studies. In *Annals of the New York Academy of Sciences.*, volume 1013, pages 83–91.

[Bach-y-Rita et al., 2003] Bach-y-Rita, P., Tyler, M., and Kaczmarek, K. (2003). Seeing with the brain. *International Journal of Human Computer Interaction*, 15(2):285–295.

[Banf et al., 2016] Banf, M., Mikalay, R., Watzke, B., and Blanz, V. (2016). PictureSensation – a mobile application to help the blind explore the visual world through touch and sound. *Journal of Rehabilitation and Assistive Technologies Engineering*, 3.

[Bevan et al., 2016] Bevan, N., Carter, J., Earthy, J., Geis, T., and Harker, S. (2016). New ISO standards for usability, usability reports and usability measures. *HCI LNCS Part I*, (9731):268–278.

[Beyond Sightloss, 2018] Beyond Sightloss (2018). Beyondsightloss, the directory for services aimed at helping blind or partially sighted people. https://www.sightlinedirectory.org.uk. Last checked on Jun 04, 2019.

[Biletzki and Matar, 2018] Biletzki, A. and Matar, A. (2018). Ludwig Wittgenstein. In 2018, S., editor, *The Stanford Encyclopedia of Philosophy*.

[Billi et al., 2006] Billi, M., Burzagli, L., Emiliani, P., Gabbanini, F., and Graziani, P. (2006). A Classification Based on ICF for Modelling Human Computer Interaction. *ICCHP*, (4061):407 – 414.

[Blackwell, 2011] Blackwell, A. (2009-2011). Human Computer Interaction. Technical report, University of Cambridge Computer Laboratory, Computer Laboratory, Williams Gates Building, J J Thomson Avenue, Cambridge CB3 0FD.

[Blackwell and Green, 2003] Blackwell, A. F. and Green, T. R. (2003). Notational systems – the cognitive dimensions of notations framework. *HCI Models, Theories and Frameworks: Toward a multidisciplinary science*, pages 103–134.

[Blenkhorn and Evans, 1998] Blenkhorn, P. and Evans, D. G. (1998). Using speech and touch to enable blind people to access schematic diagrams. *Journal of Network and Computer Applications*, 21(1):17–29.

[Bloom et al., 1996] Bloom, P., Peterson, M. A., Nadel, L., and Garrett, M. F. (1996). *Language and space*. Cambridge, MA: MIT Press.

[Bocconi et al., 2007] Bocconi, S., Dini, S., Ferlino, L., Martinoli, C., and Ott, M. (2007). ICT Educational Tools and Visually Impaired Students: Different Answers to Different Accessibility Needs. In *Universal Access in HCI, Part III, HCII*, volume 4556, pages 491–500.

[Borg, 2015] Borg, J. (2015). Assistive product classifications and regulations. In UNICEF, editor, *Assistive Products for Children with Disabilities Forum*, pages 6–7.

[Bornschein et al., 2018] Bornschein, J., Bornschein, D., and Weber, G. (2018). Comparing computer-based drawing methods for blind people with real-time tactile feedback. In *CHI*.

[Bornschein and Weber, 2017] Bornschein, J. and Weber, G. (2017). Digital drawing tools for blind users: A state-of-the-art and requirement analysis. In *Proceedings of the 10th International Conference on Pervasive Technologies Related to Assistive Environments*, pages 21–28, Island of Rhodes, Greece. ACM.

[Bowen and Reeves, 2007] Bowen, J. and Reeves, S. (2007). Using formal models to design user interfaces a case study. In *People and Computers XXI – HCI*. British Computer Society.

[Braillo, 2019] Braillo (2019). Swell form graphics machine. https://braillo.com/swell-form-graphics-machine/. Last checked on May 31, 2019.

[Calder et al., 2007] Calder, M., Cohen, R. F., Landry, J. L. N., and Skaff, J. (2007). Teaching Data Structures to Students who are Blind. In *12th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education*, volume 39, Dundee, Scotland, UK. ACM, SIGCSE.

[Cattaneo et al., 2008] Cattaneo, Z., Vecchi, T., Cornoldi, C., Mammarella, I., Bonino, D., and Ricciardi, E. (2008). Imagery and spatial processes in blindness and visual impairment. *Neuroscience and Biobehavioral Reviews*, 32(8):1346–1360.

[CDC, 2109] CDC (2109). Gateway to health communication and social marketing practice. https://www.cdc.gov/healthcommunication/ToolsTemplates/EntertainmentEd/Tips/Blindness.html. Last checked on May 31, 2019.

[Chase, 2002] Chase, S. C. (2002). A model for user interaction in grammar-based design systems. *Automation in Construction*, 11:161–172.

[Chen, 2011] Chen, C. (2011). SoundVision: Graphics Communication Methods for Blind Users. *CHI*, (7-12).

[Cohen, 2004] Cohen, M. H. (2004). *Voice user interface design.* Addison-Wesley, Boston, Mass, Harlow.

[Collins, 1984] Collins, C. (1984). On mobility aids for the blind. In Warren, I. E. R. S. D. H., editor, *Proc. of the NATO Advanced Research Workshop on Visual Spatial Prostheses for the Blind.*

[Cook and Polgar, 2007] Cook, A. M. and Polgar, J. M. (2007). *Cook and Hussey's Assistive Technologies: Principles and Practice.* Mosby.

[Cook and Polgar, 2015] Cook, A. M. and Polgar, J. M. (2015). *Assistive Technologies Principles and Practice.* Mosby.

[Cook, 1992] Cook, J. (1992). Formal methods: What? why? and when? *The Computer Journal*, 35(5).

[Corrigan, 2015] Corrigan, J. (2015). Haptic Designing Assistive Technology for People who are Blind. Technical report, Vermont Law School, United States.

[Crandall et al., 2001] Crandall, W., Bentzen, B. L., Myers, L., and Brabyn, J. (2001). New orientation and accessibility option for persons with visual impairment: Transportation applications for remote infrared audible signage. *Clinical and Experimental Optometry*, 84(3):816–4622.

[Crown, 2019] Crown (2019). Protection against disability discrimination,disability discrimination law. https://www.nidirect.gov.uk. Last checked on Mar 10, 2019.

[Dabek and Caban, 2017] Dabek, F. and Caban, J. J. (2017). A grammar-based approach for modeling user interactions and Generating Suggestions During the Data Exploration Process. *IEEE Transaction on Visualizationand Computer Graphics*, 23(1).

[Davydov and Lozynska, 2016] Davydov, M. and Lozynska, O. (2016). Linguistic models of assistive computer technologies for cognition and communication. *CSIT*, pages 6–10.

[de Mauro et al., 2001] de Mauro, C., Gori, M., Maggini, M., and Martinelli, E. (2001). Easy access to graphical interfaces by voice mouse. *Technical report, Universit'a di Siena.*

[D'Ulizia et al., 2007] D'Ulizia, A., Ferri, F., , and Grifoni, P. (2007). A Hybrid Grammar-Based Approach to Multimodal Languages Specification . In *OTM Confederated International Conferences On the Move to Meaningful Internet Systems*, volume 4805, pages 367–376.

[Editor of Texas School for the Blind and Visually Impaired, 2019] Editor of Texas School for the Blind and Visually Impaired (2019). An orientation and mobility curriculum for students with visual impairments. https://www.tsbvi.edu/publications/.

[Editor, The National Federation of the Blind, 2002] Editor, The National Federation of the Blind (2002). Sight unseen. *The Braille Monitor*, 45(9). Last checked on Sep 04, 2019.

[Erdem, 2017] Erdem, D. R. (2017). Students with special educational needs and assistive technologies: A literature review. *TOJET: The Turkish Online Journal of Educational Technology*, 16(128-146).

[Eriksson, 2010] Eriksson, Y. (2007-2010). How to make tactile pictures understandable to the blind reader. Technical report.

[Evans, 2009] Evans, S. B. (2009). *E-Learning and blindness: Evaluating the quality of the learning experience to Inform policy and practice.* PhD thesis, University Of Birmingham.

[Eyehelp, 2019] Eyehelp (2019). What Does it Mean to Be Legally Blind? http://www.eyehelp.co.uk/what-does-mean-be-legally-blind.html. Last checked on May 31, 2019.

[Fakrudeen et al., 2013] Fakrudeen, M., Ali, M., Yousef, S., and Hussein, A. H. (2013). Analysing the mental model of blind users in mobile touch screen devices for usability. *Proceedings of the World Congress on Engineering*, II(3-5).

[Fernando and Ohene-Djan, 2020] Fernando, S. and Ohene-Djan, J. (2020). An empirical evaluation of a graphics creation technique for blind and visually impaired individuals. *British Journal of Visual Impairment.*

[Ferreira et al., 2012] Ferreira, S. B. L., Nunes, R. R., and Silveira, D. (2012). Aligning usability requirements with the accessibility: Guidelines focusing on the visually impaired. *Procedia Computer science*, 14:263–273.

[Fitzpatrick and McMullen, 2008] Fitzpatrick, D. and McMullen, D. (2008). Distance Learning of Graphically Intensive Material for Visually Impaired Students. In Miesenberger, K., Klaus, J., Zagler, W., and Karshmer, A., editors, *Computers Helping People with Special Needs. ICCHP*, volume 5105, pages 219–225, Berlin, Heidelberg. Springer.

[Foundation, 2019] Foundation, P. S. (2019). Python. https://www.python.org/. Last checked on Jun 01, 2019.

[Freedom Scientific, 2018] Freedom Scientific, I. (2018). Blindness solutions, assistive technology products. 7.https://www.freedomscientific.com/Products/Blindness/JAWS.

[Fujiyoshi et al., 2014] Fujiyoshi, M., Akio, Fujiyoshi, Osawa, A., KurodaYuta, Y., and Sasaki (2014). Development of Synchronized CUI and GUI for Universal Design Tactile Graphics Production System BPLOT3. *Computers Helping People with Special Needs*, 8548(18-25).

[Fujiyoshi et al., 2018] Fujiyoshi, M., Fujiyoshi, A., Tanaka, H., and Ishida, T. (2018). Universal Design Tactile Graphics Production System BPLOT4 for Blind Teachers and Blind Staffs to Produce Tactile Graphics and Ink Print Graphics of High Qualit. *ICCHP*, pages 167–176.

[Fujiyoshi et al., 2008] Fujiyoshi, M. F. A., Ohtake, N., Yamaguchi, K., and Teshima, Y. (2008). The Development of a Universal Design Tactile Graphics Production System BPLOT2. In *ICCHP*, pages 938–945, Verlag Berlin Heidlberg.

[Gaunet and Thinus-Blanc, 1996] Gaunet, F. and Thinus-Blanc, C. (1996). Early blind subjects' spatial abilities in the locomotor space: Exploratory strategies and reaction to change performance. *Perception.*

[Glass et al., 2007] Glass, J., Hazen, T. J., Cyphers, S., Malioutov, I., Huynh, D., and Barzilay, R. (2007). MIT Spoken Lecture Processing Project. *INTERSPEECH*, pages 27–31.

[Google, 2019] Google (2019). Cloud Speech-to-Text API Client Library for Java. https://developers.google.com/api-client-library/java/apis/speech/v1. Last checked on May 31, 2019.

[Google, 2020a] Google (2020a). Google Chrome. https://www.google.com/chrome/.

[Google, 2020b] Google (2020b). Google home. https://store.google.com.

[Green, 1989] Green, T. R. G. (1989). Cognitive dimensions of notations. in a. sutcliffe and l. macaulay. *People and Computers V*, pages 443–460.

[Heller, 1989] Heller, M. (1989). Picture and pattern perception in the sighted and the blind: The advantage of late blind. *Perception*, 18(3):379–389.

[Hersh et al., 2008] Hersh, M. A., Johnson, M. A., and Michael, A. (2008). *Assistive Technology for Visually Impaired and Blind Peopl*. Number 1. Springer-Verlag London, Berlin.

[Hill et al., 1993] Hill, E. W., Rieser, J. J., Hill, M.-M., and et al, M. H. (1993). How persons with visual impairments explore novel spaces: Strategies of good and poor performers. *Journal of Visual Impairment and Blindness*, (87):295–301.

[Hollins, 1989] Hollins, M. (1989). Understanding blindness. *An Integrative Approach, chapter Blindness and Perception*.

[Holst, 1999] Holst, A. (1999). How blind users mental models affect their perceived usability of an unfamiliar screen reader. In *Technology and Person with Disability*.

[Hornbæk, 2011] Hornbæk, K. (2011). Some Whys and Hows of Experiments in Human–Computer Interaction. *Human–Computer Interaction*, 5(4):299–373.

[Hsiao and Chen, 1997] Hsiao, S.-W. and Chen, C.-H. (1997). A semantic and shape grammar based approach for product design. *Design Studies*, pages 275–296.

[Humanware, 2018] Humanware (2018). BrailleNote Touch 32 Braille Notetaker / Tablet. https://store.humanware.com/heu/braillenote-touch-32.html. Last checked on Mar 10, 2019.

[Iachini and Giusberti, 2004] Iachini, T. and Giusberti, F. (2004). Metric properties of spatial images generated from locomotion: The effect of absolute size on mental scanning. *European Journal of Cognitive Psychology*, 16(573-596).

[ICF, 2019] ICF (2019). ICF e-learning tool 2018. https://www.icf-elearning.com. Last checked on Mar 10, 2019.

[inTACT, 2014] inTACT (2014). intact sketchpad. http://www.easytactilegraphics.com/product/intact-sketchpad/. Last checked on Aug 16, 2019.

[Ishihara et al., 2006] Ishihara, T., Takagi, H., and et al (2006). Analyzing visual layouts for non-visual presentations-document interface. In *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*.

[ISO, 2006] ISO (2006). Ergonomics of human-system interaction – part 110: Dialogue principles. https://www.iso.org/standard/38009.html. Last checked on Jun 19, 2019.

[ISO, 2011] ISO (2011). Assistive products for person with disability. In *Classification and Terminology*, Geneva, Switzerland.

[ISO, 2016] ISO (2016). Systems and software engineering – systems and software quality requirements and evaluation (square) – measurement of system and software product quality. https://www.iso.org/standard/35747.html.

[ISO, 2019] ISO (2019). Ergonomics of human-system interaction – part 210: Human-centred design for interactive systems. https://www.iso.org/standard/52075.html. Last checked on Mar 16, 2019.

[Italy, 2019] Italy, A. D. (2019). Web sites accessibility. https://www.agid.gov.it/it/design-servizi/accessibilita-siti-web. Last checked on Mar 16, 2019.

[Jacobson, 2013] Jacobson, W. H. (2013). *The Art and Science of Teaching Orientation and Mobility to Persons with Visual Impairment.* AFB Press, New York.

[John, 1996] John, B. (1996). *SUS: A quick and dirty usability scale, Usability Evaluation In Industry.* Taylor and Francis, London.

[Kamel and Landay, 2000] Kamel, H. M. and Landay, J. A. (2000). A study of blind drawing practice: Creating graphical information without the visual channel. In *Fifth International ACM Conference on Assistive Technologies*, pages 13–15, Arlington, Virginia. Assets '00.

[Kamel and Landay, 2002] Kamel, H. M. and Landay, J. A. (2002). Sketching images eyes-free: A grid-based dynamic drawing tool for the blind. In *Fifth International ACM Conference on Assistive Technologies*, number 33-40, Edinburgh, Scotland. ASSETS 02, ACM Press.

[Kamel et al., 2001] Kamel, H. M., Roth, P., and Sinha, R. R. (2001). Graphics and user's exploration via simple sonics (GUESS): Providing interrelation representation of objects in a non-visual environment. In *2001 International Conference on Auditory Display*, Espoo, Finland.

[Kassim et al., 2016] Kassim, A. M., Yasuno, T., Suzuki, H., Jaanfar, H., and Aras, M. (2016). Indoor navigation system based on passive RFID transponder with digital comass for visually impaired people. *Inernation Journal of Advanced Computer Science and Applications*, 7(2):604–610.

[Kennedy, 1997] Kennedy, J. (1997). How the blind draw. *Scientific American*, 233(276):76–81.

[Kennedy, 1988] Kennedy, J. M. (1988). What can we learn about pictures from the blind?. *American Scientist*, 71:19–26.

[Kerr, 1983] Kerr, N. H. (1983). The role of vision in "visual imagery" experiments: Evidence from the congenitally blind. *Journal of Experimental Psychology: General*, 112(265-277).

[Kirakowski and Corbett, 2006] Kirakowski, U. and Corbett, M. (2006). SUMI: the software usability measurement inventory. *British Journal of Educational Technology*, 24(3):210–212.

[Knauff and May, 2006] Knauff, M. and May, E. (2006). Mental imagery, reasoning, and blindness. *The Quarterly Journal of Experimental Psychology*, 59(1):161–177.

[Kosslyn and et al, 1990] Kosslyn, S. and et al, D. O. (1990). Mental imagery. *Visual Cognition and Action*, pages 73–97.

[Kurniawan et al., 2003] Kurniawan, S. H., Sutcliffe, A. G., and Blenkhorn, P. L. (2003). How blind users mental models affect their perceived usability of an unfamiliar screen reader. *Human-Computer Interaction–INTERACT'03*, pages 631–638.

[Kurze, 1996] Kurze, M. (1996). A computer-based tactile drawing tool for blind people. In *ACM Conference on Assistive Technologies*, pages 131–138.

[Lahav and Mioduser, 2002] Lahav, O. and Mioduser, D. (2002). Multisensory virtual environment for supporting blind persons' acquisition of spatial cognitive mapping, orientation, and mobility skills. In *Disability, Virtual Reality and Associated Technologies*, Eszprem, Hungray. University of Reading.

[Lahav and Mioduser, 2008] Lahav, O. and Mioduser, D. (2008). Construction of cognitive maps of unknown spaces using a multi-sensory virtual environment for people who are blind. *Computers in Human Behavior*, (24):1139–1155.

[Lambert et al., 2004] Lambert, S., Sampai, E., Mauss, Y., and Scheiber, C. (2004). Blindness and brain plasticity: Contribution of mental imagery? *Cognitive Brain Research, 20(1), 1–11.*, 20(1):1–11.

[Leo et al., 2016] Leo, F., Cocchi, E., and Brayda, L. (October 2016). The effect of programmable tactile displays on spatial learning skills in children and adolescents of different visual disability. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, PP.

[Lewis, 1993] Lewis, J. R. (1993). IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use. *International Journal of Human-Computer Interaction.*

[Luck, 2003] Luck, R. (2003). Design studies. *Elsevier*, 24:523–535.

[Lukach, 2013] Lukach, M. O. (2013). Types of linguistic models and their application to solving of linguistic tasks. *UDC 81' 32*.

[MACS, 2019] MACS (2019). Microphthalmia, anophthalmia and coloboma support. https://macs.org.uk/. Last checked on Jun 04, 2019.

[Magnusson et al., 2009] Magnusson, C., Rassmus-Gröhn, K., Tollmar, K., and Stigmar, H. (2009). Exploring user requirements for non-visual mobile navigation systems. *INTERACT, IFIP International Federation for Information Processing*, pages 754–757.

[Mattheiss et al., 2017] Mattheiss, E., Regal, G., Sellitsch, D., and Tscheligi, M. (2017). User-centred design with visually impaired pupils: A case study of a game editor for orientation and mobility training. *International Journal of Child-Computer Interaction*, (11):12–18.

[MaxiAids, 2019] MaxiAids (2019). Sewell e-z write n draw raise line drawing kit with clip. https://www.maxiaids.com/sewell-e-z-write-n-draw-raise-line-drawing-kit-with-clip. Last checked on Aug 16, 2019.

[Mcnair and Waibel, 1994] Mcnair, A. and Waibel, A. (1994). Improving recognizer acceptance through robust, natural speech repair. In *International Conference on Spoken Language Processing*, pages 1299–1302. ICSLP.

[Medgizmo, 2018] Medgizmo (2018). Wearable technology for visually impaired. http://medgizmo.info/news/medgizmo-update-wearable-technology-for-visually-impaired-august-2016. Last checked on Mar 10, 2019.

[Meijer, 2018] Meijer, P. B. (2018). Augmented reality for the totally blind. http://www.seeingwithsound.com. Last checked on Jun 04, 2019.

[Meliones and Sampson, 2018] Meliones, A. and Sampson, D. (2018). Blind museumtourer: A system for self-guided tours in museums and blind indoor navigation. *Technologies*, 6(4).

[Mental-Health-Foundation, 2020] Mental-Health-Foundation (2020). Social model of disability.

[Microsystems, 2005] Microsystems, S. (2005). A speech synthesizer written entirely in the javaTM programming language. https://freetts.sourceforge.io/. Last checked on May 31, 2019.

[Millar, 1994] Millar, S. (1994). *Understanding and representing space: Theory and evidence from studies with blind and sighted children.* Oxford: Oxford University Press, Clarendon Press.

[Miller and Johnson-Laird, 1976] Miller, G. A. and Johnson-Laird, P. N. (1976). *Language and Perception.* The Belknap Press of Harvard University Press, Cambridge, MA.

[Mitchell et al., 2015] Mitchell, V., Ross, T., Sims, R., and Parker, C. J. (2015). Empirical investigation of the impact of using co-design methods when generating proposals for sustainable travel solutions. *CoDesign*, 12(4):205–220.

[Morris, 2014] Morris, C. (2014). *Seeing Sense: The Effectiveness of Inclusive Education for Visually Impaired Students if Further Education.* PhD thesis, Cardiff University.

[Mukherjee et al., 2014] Mukherjee, A., Garain, U., and Biswas, A. (2014). Expermenting with text-to-diagram conversion: A novel teaching aid for the blind people. *Education Technology*, 17(3):40–53.

[Munhall et al., 1996] Munhall, K. G., Gribble, P., Sacco, L., and Ward, M. (1996). Temporal constraints on the mcGurk effect. *Perception and Psychophysics*, 58(3):35–362.

[Mynatt, 1997] Mynatt, E. D. (1997). Transforming graphical interfaces into auditory interfaces for blind users. *Human Computor Interaction*, 12(1):7–45.

[NHS, 2019] NHS (2019). Blindness and vision loss. https://www.nhs.uk/conditions/vision-loss/. Last checked on Mar 10, 2019.

[Norvell, 2002] Norvell, T. (2002). A short introduction to regular expressions and context-free grammars. http://www.engr.mun.ca/ theo/Courses/fm/pub/context-free.pdf. Last checked on Mar 10, 2019.

[NV-Access, 2020] NV-Access (2020). NV Access. https://www.nvaccess.org/.

[OGL, 2019] OGL (2019). Equality Act 2010: Guidance. https://www.gov.uk/guidance/equality-act-2010-guidance.

[Overleaf, 2019] Overleaf (2019). Latex graphics using tikz: A tutorial for beginners (part 1)—basic drawing. https://www.overleaf.com. Last checked on Aug 18, 2019.

[Perkins School, 2019] Perkins School (2019). Art in science: Tools for drawing for students with visual impairment. http://www.perkinselearning.org/accessible-science/blog/art-science-tools-drawing-students-visual-impairment. Last checked on Mar 22, 2019.

[Petrie et al., 2002] Petrie, H., Schlieder, C., Blenkhorn, P., Evans, G., A. King, A. M. O., Ioannidis, G. T., Gallagher, B., Crombie, D., Mager, R., and Alafaci, M. (2002). TeDUB: A system for presenting and exploring technical drawings for blind people. In *8th International Conference ICCHP*, volume 2398, Linz, Austria. ICCHP 2002, ACM Press.

[Pinker, 1984] Pinker, S. (1984). Visual Cognition. *MIT Press*.

[Postma et al., 2007] Postma, A., Zuidhoek, S., Noordzij, M. L., and Kapper, A. M. L. (2007). Differences between early-blind, late-blind, and blindfoldedsighted people in haptic spatial-configuration learning and resulting memory traces. *Perception*, 36:1253–1265.

[Postma et al., 2008] Postma, A., Zuidhoek, S., Noordzij, M. L., and Kappers, A. M. L. (2008). Haptic orientation perception benefits from visual experience: Evidence from early-blind, late-blind, and sighted people. *Perception and Psychophysics*, 70(7):1197–1206.

[Preece et al., 2015] Preece, J., Sharp, H., and Rogers, Y. (2015). *Interaction Design: Beyond Human-Computer Interaction*. John Wiley and Sons Ltd.

[Python, 2019] Python (2019). Turtle graphics for tk. https://docs.python.org/2/library/turtle.html. Last checked on May 31, 2019.

[QIAT, 2012] QIAT (2012). Quality indicators for assistive technology. https://qiat.org/. Last checked on Mar 16, 2019.

[Rassmus-Grohn et al., 2007] Rassmus-Grohn, K., Magnusson, C., and Efiring, H. (2007). AHEAD - Audio-Haptic Drawing Editor And Explorer for Education. In *IEEE International Workshop on Haptic Audio Visual Environments and their Applications*, Ottawa, Canada. IEEE.

[Rassmus-Gröhn et al., 2013] Rassmus-Gröhn, K., Szymczak, D., and Magnusson, C. (2013). Non-visual drawing with the hipp application. *Journal on Technology and Persons with Disabilities.*

[Ricciardi and Bonisno, 2009] Ricciardi, E. and Bonisno, D. (2009). Do We Really Need Vision,? How Blind People See the Actions of Others. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 29(31):9719.

[RNIB, 2018] RNIB (2018). What assistive technology is available? https://help.rnib.org.uk/help/daily-living/technology/technology-assistive.

[RNIB, 2019] RNIB (2019). Geometry mat. https://shop.rnib.org.uk. Last checked on Jun 29, 2019.

[RNIB, 2020] RNIB (2020). Support workers. https://www.rnib.org.uk/information-everyday-living-work-and-employment-staying-work/support-workers.

[Robson, 2019] Robson, A. P. (2019). The backnaur package, (BNF) definitions. http://anorien.csc.warwick.ac.uk/mirrors/CTAN/macros/latex/contrib/backnaur/backnaur.pdf.

[Saaid et al., 2009] Saaid, M. F., Ismail, I., and Noor, M. Z. H. (2009). Radio frequency identification walking stick (RFIWS): A device for the blind. *2009 5th International Colloquium on Signal Processing Its Applications*, pages 250–253.

[Saei et al., 2010] Saei, S. N., Sulaiman, S., and Hasbullah, H. (2010). Mental model of blind users to assist designers in system development. information technology (itsim). In IEEE, editor, *International Symposium on Information Technology.*

[Sahib et al., 2013] Sahib, N. G., Stockman, T., Tombros, A., and Metatla, O. (2013). Participatory design with blind users: A scenario-based approach. *IFIP International Federation for Information Processing*, part I(8117):685–701.

[Schiewe et al., 2009] Schiewe, M., Köhlmann, W., Nadig, O., and Weber, G. (2009). What You Feel Is What You Get: Mapping GUIs on Planar Tactile Displays. *Universal Access in Human-Computer Interaction. Intelligent and Ubiquitous Interaction Environments*, 5615:564–573.

[Sears and Hanson, 2011] Sears, A. and Hanson, V. (2011). Representing users in accessibility research. In ACM, editor, *CHI*, volume 978.

[Sears et al., 2001] Sears, A., Karat, M., Oseitutu, K., Karimullah, A., and Feng, J. (2001). Productivity, satisfaction, and interaction strategies of individuals with spinal cord injuries and traditional users interacting with speech recognition software. *Universal Access in the Information Society*, 1(1):4–15.

[Sears et al., 2002] Sears, A., Lin, M., and Karimullah, A. (2002). Speech-based cursor control: understanding the effects of target size, cursor speed, and command selection. *Universal Access in the Information Society*, 2:30–43.

[Sensational Books, 2012] Sensational Books (2012). Sensational blackboard, an elegant design that is simply sensational! http://www.sensationalbooks.com/products.html. Last checked on Aug 16, 2019.

[SETT, 2002] SETT (2002). SETT Framework. https://assistedtechnology.weebly.com/. Last checked on Mar 16, 2019.

[SightandSound-Technology, 2017] SightandSound-Technology (2017). TACTIPAD: a versatile drawing board. http://www.sightandsound.co.uk/tactipad.html. Last checked on Mar 16, 2019.

[Silva and Wimalaratne, 2017] Silva, C. and Wimalaratne, P. (2017). Towards a grid based sensor fusion for visually impaired navigation using sonor and vision measurememnts. *IEEE Humanitarian Technology Conference*.

[Srihari, 1995] Srihari, R. K. (1995). Computational models for integrating linguistic and visual information: A survey. *Artificial Intelligence Review*, 8:349–369.

[SSMR, 2009] SSMR (2009). Understanding the needs of blind and partially sighted people. Technical report, University of Surrey.

[Stiny, 1980] Stiny, G. (1980). Introduction to Shape and Shape Grammar. *Environment and planning B*, 7:343–351.

[Struiksma et al., 2009] Struiksma, M. E., Noordzij, M. L., and Postma, A. (2009). What is the link between language and spatial images? behavioral and neural findings in blnid and sighted individuals. *Acta Psychologia*, 132(2):145–156.

[Su et al., 1999] Su, J. C., Uslan, M. M., and Schnell, B. K. (1999). A Review of Supernova Screen Magnification Program for Windows. *Journal of Visual Impairment and Blindness*, 93(2):108–110.

[Suhm et al., 2001] Suhm, B., Myers, B., and Waibel, A. (2001). Multimodal error correction for speech user interfaces. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 8(1):60–98.

[Sun, 1998] Sun (1998). RuleGrammar. https://docs.oracle.com. Last checked on May 31, 2019.

[Sweller, 1994] Sweller, J. (1994). Cognitive load theory, learning difficulty, and instructional design. *Learning and Instruction*, 4(4):295–312.

[Takagi, 2009] Takagi, N. (2009). Mathematical figure recognition for automationg production of tactile graphics. In *IEEE International Conference on Systems, Man and Cybernetics(SMC 2009)*, pages 4651–4656. IEEE.

[Thatcher, 1994] Thatcher, J. (1994). Screen reader/2: Access to os/2 and the graphical user interface. In ACM, editor, *Proceedings of the First Annual ACM Conference on Assistive Technologies (Assets 94)*, pages 39–46, New York, NY, USA.

[Thinus-Blanc and Gaunet, 1997] Thinus-Blanc, C. and Gaunet, F. (1997). Representation of space in blind persons: Vision as a spatial sense? *Vision as a spatial sense?*, 121(20-42).

[Tnpia, 1999] Tnpia, M. (1999). A visual implementation of a shape grammar system. *Planning and Design*, 26:59.

[TouchPad Pro, 2019] TouchPad Pro (2019). Touchpad pro, the future of electronics for the blind and visually impaired. https://www.touchpadpro.org/. Last checked on May 31, 2019.

[Ungar, 2000] Ungar, S. (2000). *Cognitive mapping without visual experience*. Routledge is an imprint of the Taylor and Francis group, London.

[Vanlierde et al., 2003] Vanlierde, A., Volder, A. G. D., Wanet-Defalque, M.-C., and Veraart, C. (2003). Occipitoparietal cortex activation during visuo-spatial imagery in early blind humans. *NeuroImage*, 19(3):698–709.

[Vaza et al., 2018] Vaza, R., Fernandesb, P. O., and Veigac, A. C. R. (2018). Designing an Interactive Exhibitor for Assisting Blind and Visually Impaired Visitors in Tactile Exploration of Original Museum Pieces. *Procedia Computer Science*, (138):561–570.

[W3C, 2018] W3C (2018). Web Accessibility Initiative (WAI) at W3C. https://www.w3.org/standards/webdesign/accessibility. Last checked on Mar 16, 2019.

[W3C, 2019] W3C (2019). Speech recognition grammar specification version 1.0. https://www.w3.org/TR/speech-grammar/. Last checked on May 31, 2019.

[Wai and Stephen., 2003] Wai, Y. and Stephen., B. (2003). Evaluation of multimodal graphs for blind people. *Universal Access in the Information Society*, 2:105–124.

[Walker and Mauney, 2010] Walker, B. N. and Mauney, L. M. (March 2010). Universal design of auditory graphs: A comparison of sonification mappings for visually impaired and sighted listeners. *ACM Transactions on Accessible Computing (TACCESS)*, Volume 2 Issue 3.

[WANG et al., 2019] WANG, D., GUO, Y., LIU, S., ZHAN, Y., XU, W., and XIAO, J. (2019). Haptic display for virtual reality: progress and challenges. *Virtual Reality and Intelligent Hardware*, 1(2):136–162.

[WATI, 2018] WATI (2018). The process from assistive technology consideration to assistive technology assessment.

[WebAIM, 2019] WebAIM (2019). Designing for screen reader compatibility. https://webaim.org. Last checked on April 30, 2019.

[WebAIM, 2020] WebAIM (2020). Screen Reader User Survey 8 Results. https://webaim.org/projects/screenreadersurvey8/.

[WHO, 2015] WHO (2015). Visual disturbances and blindness. https://icd.who.int. Last checked on Mar 10, 2019.

[WHO, 2018] WHO (2018). Blindness and vision impairment. http://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment. Last checked on Mar 10, 2019.

[Wilkinson, 2011] Wilkinson, C. R. (2011). *Evaluating the Role of Prior Experience in Inclusive Design*. PhD thesis, Cambridge University Engineering Department.

[Williams et al., 2014] Williams, G. J., Zhang, T., Lo, A., Gonzales, A., and D. Baluch, B. D. (2014). 3d printing tactile graphics for the blind: Application to histology. In *Anual Rehabilitation Engineering Society of North America Conference 2014*, Indianapolis,IN.

[Wisconsin-Milwaukee, 2015] Wisconsin-Milwaukee, U. (2001-2015). Models and taxonomies relating to assistive technology. http://www.r2d2.uwm.edu/atoms/archive/technicalreports/fieldscans/fs7/tr-fs-taxonomiesmodels-resource.html.

[Withagen et al., 2013] Withagen, A., Kappers, A. M., Vervloed, M. P., Knoors, H., and Verhoeven, L. (2013). Short term memory and working memory in blind versus sighted children. *Research in Developmental Disabilities*, 34:2161–2172.

[Wood, 1993] Wood, C. C. (1993). A cognitive dimensional analysis of idea sketches. *Cognitive Science Research Paper, School of Cognitive and Computing Sciences, University of Sussex*, (275).

[Zagler et al., 1994] Zagler, W., Busby, G., and Wagner, R. (1994). Synthesizing non-speech sound to support blind and visually impaired computer users. *Computers for Handicapped Persons*, pages 385–393.

[Zhang et al., 2017] Zhang, T., Duerstock, B. S., and Wachs, J. P. (February 2017). Multimodal perception of histological images for persons who are blind or visually impaired. *ACM Transactions on Accessible Computing (TACCESS)*, Volume 9 Issue 3.

[Zhao et al., 2009] Zhao, Z., andTing Zhang, P.-L. P. R., and Salvendy, G. (2009). Visual search-based design and evaluation of screen magnifiers for older and visually impaired users. *International Journal of Human Computer Studies*, 67(8):663–675.

[Zhong et al., 2013] Zhong, Y., Garrigues, P. J., and Bigham, J. P. (2013). Real time object scanning using a mobile phone and cloud-based visual search engine. In ACM, editor, *In Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '13)*, volume 20, page 8, ACM, New York, NY, USA.

[Zhu and Feng, 2010] Zhu, S. and Feng, J. (2010). Investigating grid-based navigation: The impact of physical disability. *ACM Transaction on Accessible Computing*, 3(1).