Medialness-based Shape Invariant Feature Transformation

Prashant Aparajeya

B.Eng, MBM Engineering College, Jodhpur, India, 2009
M.Tech, Indian Institute of Information Technology, Allahabad, India, 2011

Thesis submitted for the degree of

Doctor of Philosophy

Department of Computing

Goldsmiths, University of London

Goldsmiths, University of London 2016

Declaration

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

Prashant Aparajeya
2nd of April, 2016

Dedicated to my Family and PhD Supervisor

Acknowledgements

First and foremost I want to express my gratitude and thanks to my thesis advisors, Professor Frederic Fol Leymarie, Professor Stefan Rueger and Professor Jonathan Freeman, who taught me a lot about research and life in general. They were always there to provide everything not only to support my research but also to help me adjusting to life in the United Kingdom. Frederic has been an ideal adviser: a great mentor, a collaborator, a guide, and a friend. This thesis would not have been completed without his commitment and diligent efforts which not only influenced the content of the thesis but also the language in which it has been written.

My PhD studies were partly supported by the European Union (FP7 – ICT; Grant Agreement *#258749*; CEEDs project). This research work also led to three major collaborations: first with Professor Stefan Rueger at the Knowledge Media Institute at The Open University on shape based information retrieval aspects; second with Professor Ilona Kovacs at the department of Psychology at Budapest University of Technology and Economics on pshychophysical aspects; and third with Visual Media Artist Dr. Vesna Petresin Robert on movement design in arts. I am very grateful to all of them for the exciting collaboration.

I would like to thank my colleagues and staff in the department of Computing and Psychology at Goldsmiths, whose support I always had on my side. I am very thankful to my upgrade examiners Dr. Rebecca Fiebrink and Prof. Robert Zimmer for their thoughtful and critical feedback. I would like to name and thank all my dear friends, for which I

would write another thesis.

Friends and colleagues from EAVI and IGGI made Goldsmiths a fun and interesting place to work. Many thanks got to my close friends Daniel Berio, Pierre-Francois Gerard, Rui Filipe Antunes, Mohammad Majid al-Rifaie, Ezwan Mokhtar and Carol MacGillivray for putting up with me during these past four years, and for interesting discussions. I also want to thank Jane Lessiter, Andrea Miotto, Eva Ferrari, Alessia Eletta Coppi, Christian Guckelsberger , Tom Cole, Hooman Oroojeni, Francisco Bernardo, Alessandro Altavilla, Alexander Brigden, Janet Gibbs, Zoe O'Shea, Tara Colingwoode-Williams and Rob Homewood for the lively working atmosphere and interesting lunchtime discussions. I would also like to thank Wendy McDonald, Rose Hepworth, Karen Hodgson and Tuula Juvonen for bearing with me for all the administrative goof ups.

Finally, I give my deepest gratitude to my family for their endless support and encouragement. I am in debt to my thesis committee – Prof. Anne Verroust-Blondet and Dr. Marco Gillies for their patience when I struggled to finish the manuscript, and for their interest in my work.

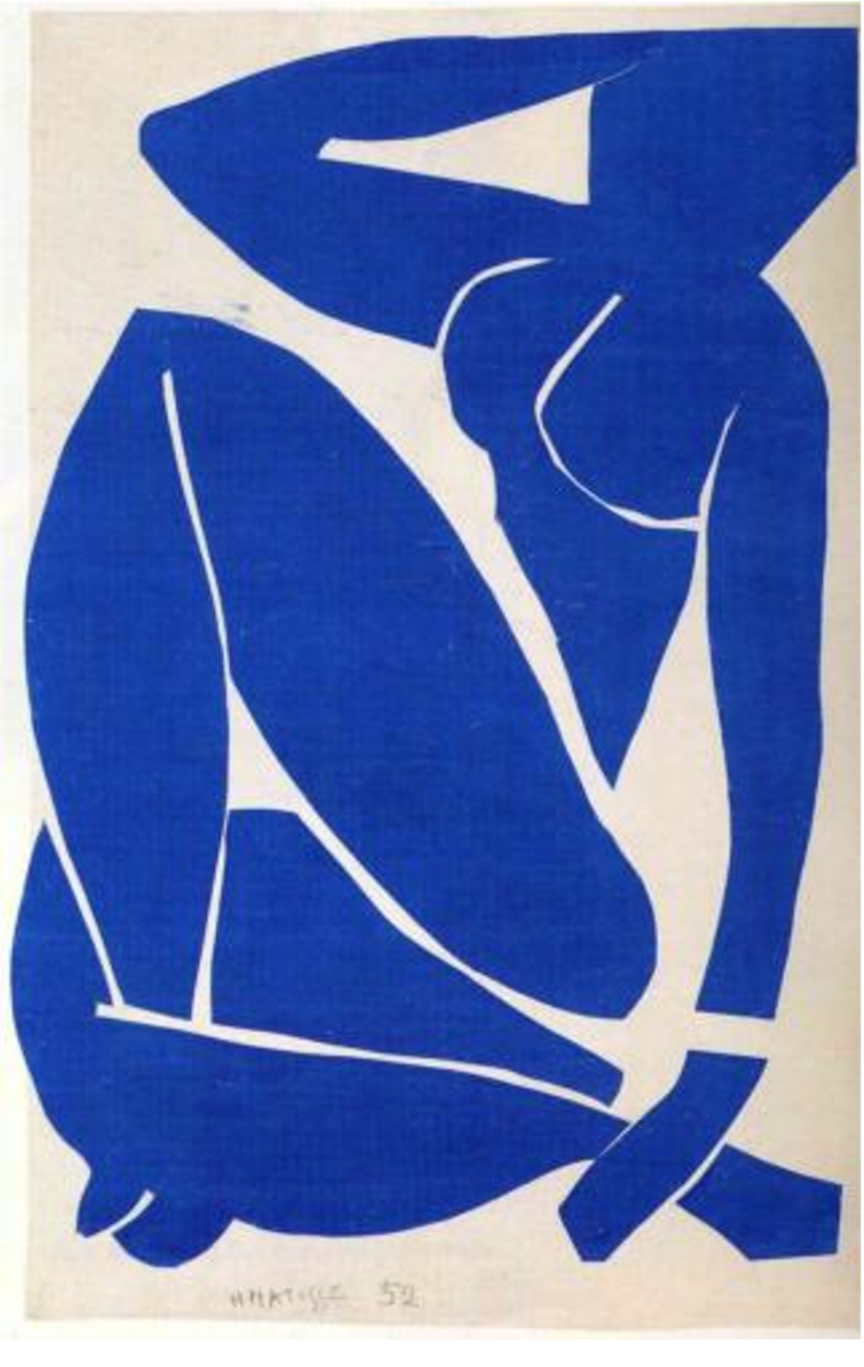
Abstract

This research is about the perception-based medial point description of a natural form (2D static or in movement) as a generic framework for a part-based shape representation, which can then be efficiently used in biological species identification, as well as more general pattern matching and shape movement tasks. We consider recent studies and results in cognitive science that point in similar directions in emphasizing the likely importance of *medialness* as a core feature used by humans in perceiving shapes in static or dynamic situations. This leads us to define an algorithmic chain composed of the following main steps. The first step is one of fuzzy medialness measurements of 2D segmented objects from intensity images that emphasizes main shape information characteristic of an object's parts, *e.g.* concavities and folds along a contour. We distinguish interior from exterior shape description. Interior medialness is used to characterise deformations from straightness, corners and necks, while exterior medialness identifies the main concavities and inlands which are useful to verify parts extent and reason about articulation and movement. The second main step consists on defining a feature descriptor, we call ShIFT: Shape Invariant Feature Transform constructed from our proposed medialness-based discrete set, which permits efficient matching tasks when treating very large databases of images containing various types of 2D objects. Our defined shape descriptor ShIFT basically captures elementary shape cues and hence it is able to characterise any 2D shape. In summary, our shape descriptor is strongly footed in results from cognitive psychology while the algorithmic part is influenced by techniques from more traditional computer vision.

Contents

List of Figures

List of Tables



Henry Matisse, *Blue Nude I*, 1952. @Exhibition opening Tate Modern, London.

Chapter 1

Introduction

> *"In the beginning you must subject yourself to the influence of nature. You must be able to walk firmly on the ground before you start walking on a tightrope."*
>
> *– Henri Emile Benoit Matisse, Artist (1869–1954)*

Technology is progressing day-by-day and scientists seek to enhance computers ability to understand the in-surrounding world through the development of computer vision. Visual information plays a crucial role in our society, it is increasingly pervasive. One of the basic and most fundamental problem in computer vision is that of object representation, which can be further exploited for recognition and categorization tasks. These tasks are complex enough that it is often not sufficient to simply regard the raw images as training examples and apply latest machine leaning algorithms. Rather, there is structure in natural images, which should be exploited to extract machine understandable features. This thesis concentrates on mapping our current understanding of the human perception of shape and creating algorithms, which transform an image into an intermediate representation through medialness measurement and provide far superior input to later modules in a processing chain for high-level tasks such as shape matching and movement analysis.

In this chapter, first I describe my motivation for this thesis followed by the characterisation of the problem statement. Further, I discuss most relevant and commonly used definitions and methods for 2D shape representation found in the computer vision literature.

1.1 Motivation

Over the past few decades, computer vision scientists and psychologists have put enormous effort in the understanding of images of objects and their mathematical representation. An important theory on object detection is that humans have very powerful visual sensors and processors that successfully identify an object by its parts and their aggregation, despite changes in size and orientation of an image. This theory, known as recognition-by-components (RBC, Biederman (1987)), explains how moderately occluded or degraded objects are successfully recognized by the human visual system. A similar strategy is formed in artistic rendering and animation where an artist represents any character as the combination of primitive structures of different sizes (Figure 1.1, here approximate disks of various radii), a technique referred to as "geometric drawing" (Simmons and Winer (1977)). Different poses or body movements are characterised by a particular orientation and combination of these primitives. Each body part can be fleshed out and refined in successive sketches (Loomis (1951); Simmons and Winer (1977)). Such sketches of characters can then be directed using the Line of Action technique from animation: a single curve running through the "middle" of the character, which represents an overall force and direction of movement for the character (Bregler et al. (2002); Guay et al. (2013)).

On the other hand, from the point of view of psychophysical investigations on the perception of shape and their movements by humans, Kovács et al. (1998) have indicated that such articulated movements of a biological character can be captured via a minimal

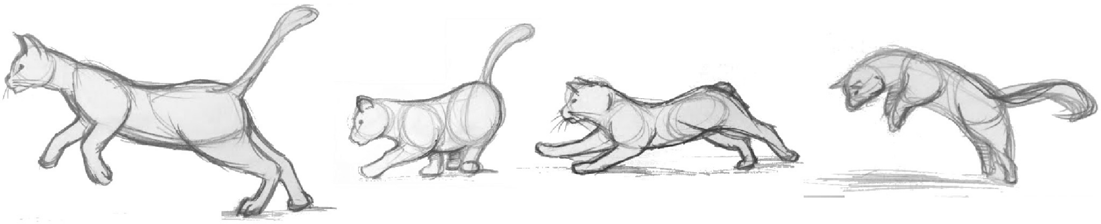


Figure 1.1: An artistic way to draw animal shapes. An artist perceives an animal character as the combination of primitives of varying size (*e.g.* in its simplest form as a series of disks of varying radius positioned at important junctions and capturing the main body parts). The particular orientations and combinations of these primitives indicate different body pauses and leads to an animation of natural-looking movements. Artist: Mr. Kelvin Chow.

set of dominant features, potentially being represented as isolated points or “hot spots” positioned near a medial axis locus; such ideas have been reinforced recently with further studies and results by Lescroart and Biederman (2013); Firestone and Scholl (2014).

Inspired with the theory of RBC and these two approaches to the perception of the shape of articulated objects by humans, we have investigated a possible scheme based on the notion of robust medialness initially formulated by Kovács et al. (1998) that can efficiently capture the important structural part-based information commonly used in artistic drawings and animations. The main advantage over traditional medial-based representations of 2D shape is one of combined compactness, robustness and capacity of dealing with articulated movements.

1.2 Problem Statement

Almost all the classic approaches on machine vision (Belongie et al. (2002); Loffler (2008); Poppe (2010); Premachandran and Kakarala (2013); Wei and Li (2014)) try to integrate some knowledge from current theories on visual perception and cognition, in order to try to solve problems linked with human perception. However, despite continuous progress in recent decades, machine vision systems still have a long way to go before being able to replicate human visual performance. A general result from visual perception is that a human can effortlessly detect the shape of any object, despite of the variations in their size, orientation and moderate occlusion (Biederman (1987)). One finds numerous ways of defining shape in the computer vision literature; for example, one-dimensional functions for shape representation (Wang et al. (1999); Loncaric (1998); Chang et al. (2001); Zhang and Lu (2004)), polygonal approximation (Latecki and Lakämper (1999); ShuiHua and ShuangYuan (2005)), spatial interrelation feature (Davies (2004); Bauckhage and Tsotsos (2005); Liu et al. (2007)), moments (Mukundan et al. (2001); Celebi and Aslandogan (2005)), scale-space methods (Abbasi et al. (2000)), shape transform domains (Chuang and Kuo (1996); Chen and Bui (1999); Khalil and Bayoumi (2001)). However , to achieve the goals of the RBC theory, shape representation requires a set of properties or features that are invariant under affine changes and prederably relate to what is known of human vision. After an in-depth investigation on the state-of-the-art of shape representation and object matching tasks, we found that: (a) the classical shape descriptors and matching tasks mainly rely on well segmented object, as contours are proven to be a semantically strong representation of the object, and (b) published results work only on specific datasets (Zhang and Lu (2004); Yang et al. (2008); Poppe (2010); Weinland et al. (2011)). Moreover, most of these shape descriptors produce poor or false matching results when changed by a simple affine transformation. Although, some shape descriptors are capable in handling affine transformation (Nasreddine et al. (2010); Wang et al. (2012)), they fail to describe the shape through its parts to meet the basic part-based shape matching task. While on the other hand partial shape matching approaches (Chen et al. (2008); Riemenschneider et al. (2010)) perform relatively well in case of occlusions, they are not fully invariant to affine transformation and work only on well segmented images. The inner-distance approach (Ling and Jacobs (2007)) works well in handling articulations but is overly sensitive to holes, object deformation, occlusions, cuts and perturbations.

Feature based descriptors (Lowe (1999, 2004); Mikolajczyk and Schmid (2004, 2005); Dalal and Triggs (2005)) work well in detecting objects in natural scenes, but they do not consider an explicit shape representation of an object. As a consequence, accuracy of these methods reduces drastically for detecting closely related objects with similar shape, but with different textures: *i.e.* the local intensity variations take over the overall shape and parts organisation of an object. Furthermore, to our knowledge, none of the shape descriptors are able to define a shape generically, i.e., shape definition applicable for every 2D shape of object.

Therefore, the major challenge of this thesis is to create a perception based descriptor that can

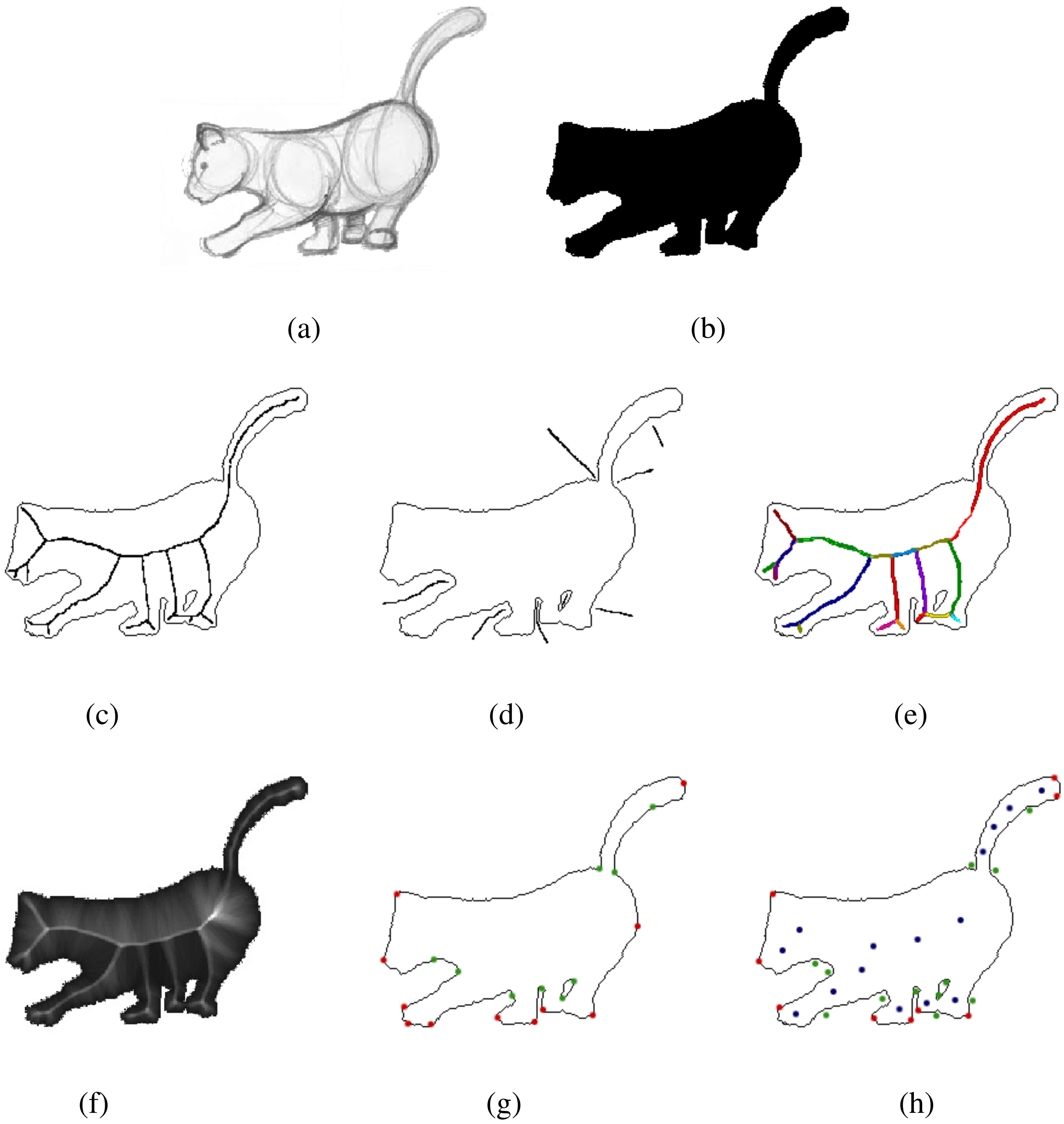
1. provide a generic definition of a 2D (possibly articulated) shape;
2. show invariance towards affine transformation;
3. be easily indexable over a very large dataset.

Moreover, a robust and reliable matching algorithm is desired to handle part based matching as well as be robust to occlusions, cuts and articulated movements.

Our current approach is to develop a shape matching algorithm that is invariant to translation, scale and rotation and is inspired by the now classic Scale Invariant Feature Transform (SIFT, Lowe (1999, 2004)). Our shape representation is derived from the region-based medial point description of shape proposed by Kovács et al. (1998) in cognitive science and perception studies. In the current work, an algorithmic chain is further developed to highlight the feature points. The purpose of evaluating such medialness measurement is to provide a representation of shape that is local, compact, can easily be applied at different spatial scales, and mimics human sensitivity to contour stimuli. This process maps the whole shape information into a few number of points we call “dominant” and hence make it compact. Contrary to classic medial-based representations, ours is *not overly sensitive* to small boundary deformations and furthermore gives high response in those regions where the object has high curvature with *large boundary support* and in the vicinity of *joints* between well-delineated parts, such as the limbs of an animal. We further augment interior hot-spots as proposed by Kovács et al. (1998); Kovács (2010) with significant concave and convex points by locating these at the end of medialness ridges. This permits to tie medialness with the "codon" theory of Richards and Hoffman (1985), proposed in the 1980's as potent object part separators and often used to characterise shape complexity (Kayaert et al. (2011)) as well as being used in scale-space analyses of contours (Yang et al. (2008)). Evidence for the importance of significant contour curvature extrema as shape-rich features has also been increasing in recent years in the psychology and vision literature (De Winter and Wagemans (2008); Rodríguez-Sánchez and Tsotsos (2012)).

1.3 Shape Representation

Shape representation towards matching has been addressed in many ways by computer scientists in recent years, including by directly characterising and grouping contour points (Chui and Rangarajan (2003); Liu et al. (2011b); Van Wamelen et al. (1999, 2004)), by contour analysis (Bai et al. (2009a); Berretti et al. (2000); Chen et al. (2008); Latecki et al. (2000); Srestasathiern and Yilmaz (2011)), using Blum's medial axis transform (MAT) (Kimia (2003); Sebastian et al. (2004)) and the closely related skeleton (Bai et al. (2008, 2009a)), inner distance (Ling and Jacobs (2007)), medial point transform (MPT) (van Tonder and Ejima (2003)), fusion of contour and local interest points descriptions (Mouine et al. (2013a)), multiscale triangulation representation (Mouine et al. (2013b)), and contour enclosure-based symmetries (Kelly and Levine (1993, 1995a,b)). Some of the main medial representations are illustrated in Figure 1.2, including our proposed method. Other classic approaches emphasise similarly either boundary information (*e.g.* Fourier



(a) (b)

(c) (d) (e)

(f) (g) (h)

Figure 1.2: (a) Sketch of a cat built from a small number of approximate disks (visible sketched lines); (b) corresponding segmented and binarized image; (c) classic internal medial-axis approximation; (d) external medial-axis; (e) 2D shock graph; (f) proposed interior medialness map; (g) recovered concave (green dots) and convex (red dots) points, and (h) final dominant (medial) point set (in blue for internal ones, in green for external/concave ones, and in red for convex points) obtained via our method.

(Zhang and Lu (2002); Chen and Bui (1999)), wavelet (Chuang and Kuo (1996)) and scale-space analyses of closed contours (Mokhtarian and Mackworth (1992); Mokhtarian et al. (1997); Zhang and Lu (2003))) or interior information (*e.g.* primitive retro-fitting or approximation). The general approach to matching is then to find good ways to put in correspondence the whole shape representation from a query with an equivalent complete shape representation of a target object (*e.g.* extracting a skeleton from a segmented image and defining a process to match it with another skeleton description in the database).

1.4 Contributions

I propose instead to find an efficient medial representation which remains discrete (point-based), is (at least approximately) invariant to scaling, rotations and translations, and can be the basis of a feature vector map for efficient query-target matching tasks as produced in the discipline of Information Retrieval. Note that we do not require to have a complete object segmented and thus will also address partial shape matching. Note also that most of other classical shape-based approaches do not consider deformations and articulated movements, while we do.

Mathematically, following Kovács et al. (1998), *medialness* of a point in the image space is defined by computing the D_ε function based on an equidistance metric to boundary segments. The D_ε value at any point in image space is represented as the containment of all segment of boundary (information, edges) falling into the annulus of thickness parameterised by the tolerance value ε, and with interior radius taken as the minimum radial distance of a point from boundary (Figure 1.3). I modify the original definition of medialness by introducing weight and orientation to bounding contour points to better localise the medialness measurements. On completion of medialness measurements each pixel in the transformed image space holds a local shape information of cumulated medialness. Assuming figure-ground separation, thickness variations, bulges and necks of an object

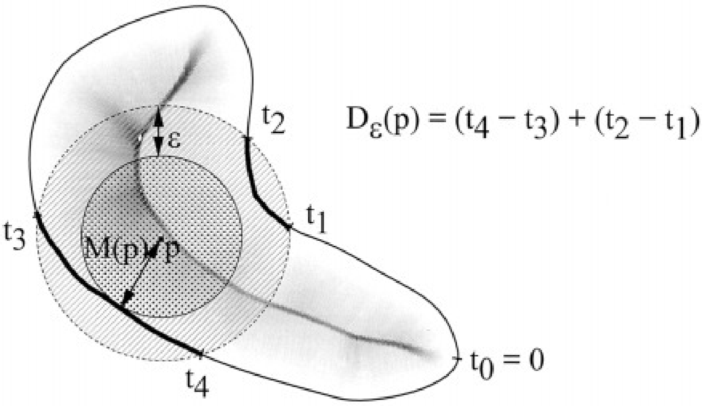


Figure 1.3: Adapted from Kovács et al. (1998) with permission from the lead author: the D_ε function for a simple shape is defined as an accumulation of curve segments falling inside the annulus neighborhood of thickness ε (thick boundary segments within the gray ring) centered around the circle with center p. $M(p)$ is taken as the minimum radial distance from point p to the nearest contour point.

are captured via *interior* medialness measurement, while the concavities and joints are defined via *exterior* medialness measurement. The interior and exterior medialness measurement gives different types of information about the shape of an object, and they are processed separately in our proposed method.

In the work of Kovács et al. (1998) it is shown that humans are most sensitive to a small number of localised areas of medialness which coarsely correspond to joints for animated bodies. Our equivalent (extended) notion is defined as *dominant points* and can be applied to any objects, animated or not. Dominant points are constrained to be a relatively small number of points of high medialness obtained by filtering out the less informative, redundant and noisy data from the initial medialness image space.

To identify *internal* dominant points a top-hat transform (Serra (1983); Leymarie and Levine (1988); Vincent (1993)) is applied to isolate peaks in the medialness signal.[1] Peaks are filtered using an empirically derived threshold. The selected peaks are then each char-

[1] In mathematical morphology, images are considered as set functions on which operations such as dilations and erosions can be applied iteratively to gradually modify the initial sets and emphasise certain characteristics. The hat-transforms are used to isolate peaks or valleys in function sets (such as an intensity image).

acterised by a single representative point. To avoid considering large numbers of nearby isolated peaks which are characteristic of object regions with many small deformations, only peaks at a given minimum distance away from each other are retained. The extraction process of *external* dominant point is achieved by combining a concavity measure together with length of support on the contour. Again, a spatially localised filtering is applied to isolate representative dominant points. Furthermore, to improve the robustness of our representation, we extracted the set of *convex points* to capture the blob like structure from the shape. We have shown that the articulation and limbs movement can be captured via the *exterior* dominant points. All the selected internal and external dominant points along with convex points are then considered as the *representative feature points* of the shape. Our matching algorithm is designed in such a way that it first compares internal dominant points of a query object with internal representative dominant points of target shapes in a database. External dominant points are then processed similarly. The matching algorithm first analyses the amount of scale, rotation and translation of the query with respect to the target image. These values are then applied over the query image to find the best possible matching location in the target image.

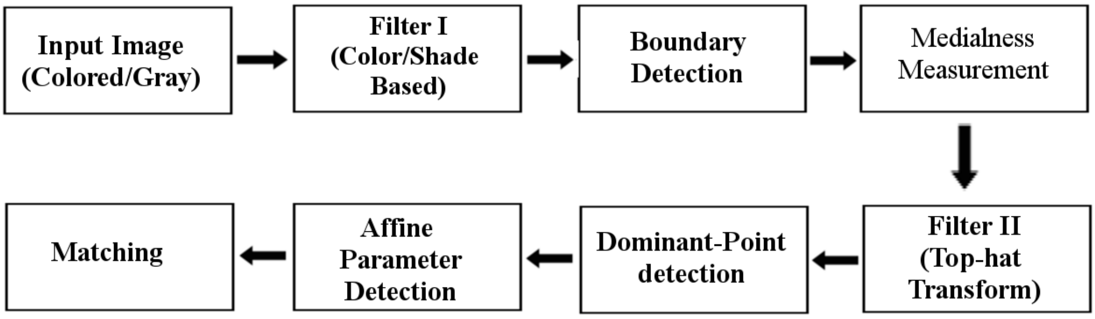

Input Image (Colored/Gray)
Filter I (Color/Shade Based)
Boundary Detection
Medialness Measurement
Filter II (Top-hat Transform)
Dominant-Point detection
Affine Parameter Detection
Matching

Figure 1.4: Block diagram of medialness based matching task.

The overall working process of the system is shown via a block-diagram in Figure 1.4. This thesis is mainly focused around providing a generic definition of the 2D shape; driven by previous psychophysical investigation on human perception. I have evaluated this definition on diverse datasets of 2D objects and proved them to work well on a number of tasks where 2D shapes can play significant role. The thesis includes such tasks as object classification, matching and recognition.

1.5 Thesis Outline

Chapter 2: *Background*, reviews the state-of-the-art of medialness measurement and object-based image matching. The chapter is categorised into three sections: review on medialness measurement, contour-based object matching and feature-based image matching methods. The main goal of this chapter is to summarise the evolution from the medial axis to medialness measurements, also comparing different contour based and feature based matching methods.

Chapter 3: *Medialness Measure via Oriented Boundary*, details the mathematical formulation of our modified definition of medialness measure. This chapter has two major sections: (a) calculating orientation and weight (actual length) of a contour point; (b) formulating and comparing two possible functions for the adaptive tolerance ε as a function of the minimum radial distance. It also provides definitions to both interior and exterior medialness calculations, which are further used to extract different feature or dominant points from the given shape.

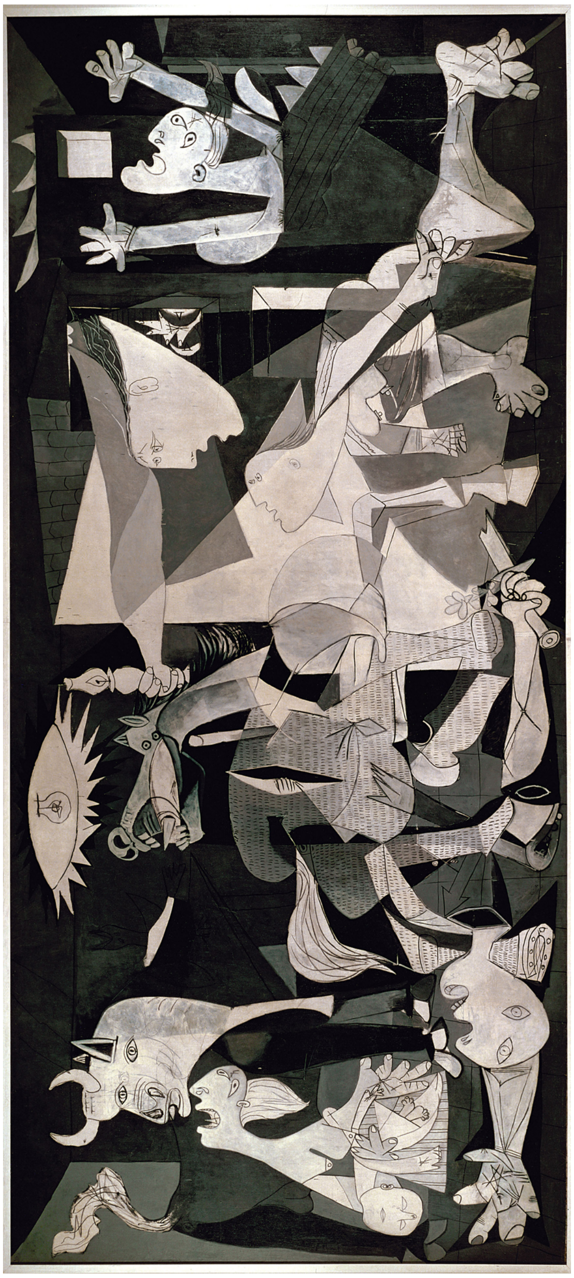
Chapter 4: *Shape Invariant Feature Extraction and Description*, presents the mathematical model of our designed feature descriptor: the Shape Invariant Feature Transform (ShIFT); the chapter also describes the extraction process of different dominant points. The sections are: (a) detail mathematical description of the ShIFT vector and finding key-point orientations; (b) interior, concave or exterior and convex dominant points extraction process; and (c) indexing on ShIFT feature vectors. The ShIFT descriptor's affine invariance is outlined, as well as it's behavior under viewpoint changes. The chapter also considers those cases where an object is inverted (upside-down or mirror via a reflector).

We also provide two complementary methods to extract extremities (concave and convex points) from an outline.

Chapter 5: *Shape Retrieval and Matching*, provides details on shape retrieval and matching techniques. The sections are: (a) illustrating a strategical process to index and retrieve shape through ShIFT vectors; (b) finding a homography amongst two shapes and exploiting it further for the matching task; (c) different ranking metrics to evaluate the outcomes for a given query shape; (d) a bag-of-words model for our matching system; and (e) how we evaluate the overall performance of our system for a given set of shapes or dataset.

Chapter 6: *Results and Discussion*, evaluates our framework for different set of shape databases. We introduce two datasets: (a) a dataset of 2D heterogeneous objects, and (b) dataset of cartoon characters, focused on the Simpson's family characters; we further evaluate the performance of our system via different metrics. The chapter also compares the results with the recent state-of-the-art 2D-shape descriptors. The robustness of our matching algorithm is further tested by perturbing the data-set at different levels.

Chapter 7: *Conclusion and Future Work*, concludes the thesis and suggests potential future extensions for our work.



Pablo Picasso, *Guernica*, 1937. @Museo Reina Sofia, Madrid, Spain.

Chapter 2

Background

> *"All perceiving is also thinking, all reasoning is also intuition, all observation is also invention."*
>
> *– Rudolf Arnheim, Perceptual Psychologist (1904–2007)*

Visual information plays an important role in our society, where it is increasingly pervasive in our human lives. Images and videos are used in many application areas like architectural and engineering design, fashion, journalism, advertising, entertainment, security, military, planning, science. Object recognition and detection in 2D images has received a lot of attention in the computer vision and pattern recognition communities. Computer vision seeks to enhance the ability of machines to understand the visual world through the development of algorithms for object detection, understanding, recognition and reconstruction. In the early days of computer vision Blum (1967) proposed a then novel geometrical interpretation of 2D objects by introducing the concept of medial axis, which captures both global and local shape properties of an object. The medial axis transform (MAT) of an object (also referred to as $skeletal$ representation) is defined as the locus of the centers of the maximal disks that fit an object's outline. Each symmetry axis locus can be linked to curvatures at the boundary of the object, hence unifying local differential structure with global topological properties (Siddiqi and Pizer (2008)).

Many researchers have studied shape representation, matching, comparison and recognition. Such work can be loosely organised into two groups: (i) contour vs region and (ii) feature-based. Contour-based and region methods work directly from a representation of object boundaries, while feature-based methods attempt to first extract salient features such as corners and use a small amount of associated local information, *e.g.* such as the correlation within a small image patch (Förstner (1986)).

In the remaining of this chapter, we first summarise in Section 2.1 the main approaches that relate to our work. Then in Section 2.2, we provide a more general background on the main recent approaches to 2D shape analysis and retrieval.

2.1 Background on Medialness Measurement

In mathematical morphology, the medial axis (*MA*) became a central concept (Serra (1983)). In computational geometry the *MA* is closely related to the Voronoi diagram (Aurenhammer (1991)) and bisector sets. A mathematical definition of Medial loci can be given as (adopted from Siddiqi and Pizer (2008)): Let S be a connected closed set in $\mathbb{R}^n$. A closed ball $B \subset \mathbb{R}^n$ is called a *maximal inscribed ball* in S if $B \subset S$ and there does not exist another ball $B^{'} \neq B$ such that $B \subset B^{'} \subset S$. The medial locus is thus a subset of the space $\mathbb{R}^n \times [0, +\infty]$.

Blum, a true visionary in the history of pattern recognition, was interested in developing a mathematical description for biological shapes, in order to study growth and movement (Blum (1962a,b); Kotelly (1963)). Blum provided a suitable class of descriptors in the form of a skeletal representation of shape as the result of a grassfire transformation also known as the medial-axis transformation or *MAT* (see Figure 2.1). The *MAT* results in a graph with segments capturing local symmetries and junctions relating to parts of the original object's contour. A medial-axis segment is a symmetry axis in the sense that each pair of associated contour boundaries forms a local mirror-like symmetry.

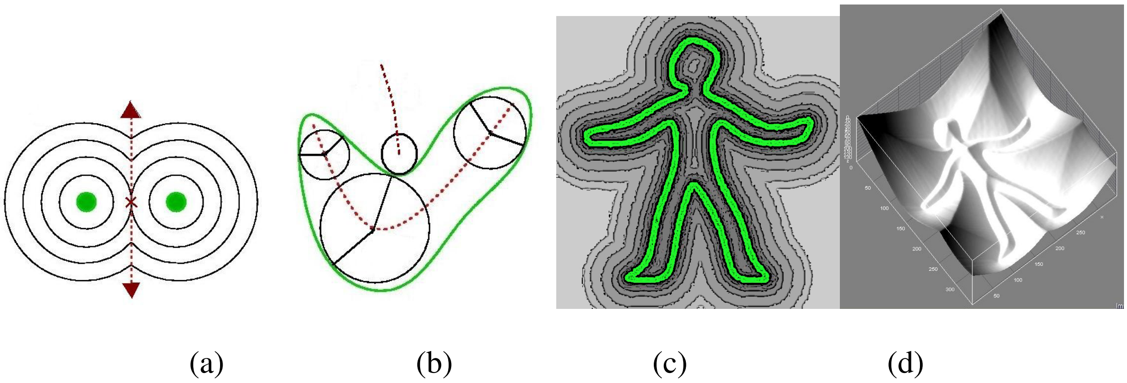


Figure 2.1: Adapted from Leymarie (2011) after Van Tonder et al. (2003), Figure 3: The Medial Axis of Blum: (a) for two sample points as the locus of meeting Euclidean wavefronts, where the (central) cross indicates the initial shock when the fronts first meet, and the arrows indicate the direction of growth or flow of the medial axis segments; (b) as the loci of centres of maximal contact disks (a dual view to propagating wavefronts such as in (a)); (c) derived from a distance map computed for a humanoid outline, as the ridge lines of the corresponding height field, (d), where the (signed) distance from the outline maps to a height value.

For a given silhouette, each segment of the medial axis can correspond to categorically different types of configurations of the corresponding pair. This problem is resolved by introducing shock graphs (Kimia et al. (1995)), which is a form of directed planar graph, produced as a result of the grouping of medial axis points with common directions of distance flow or increasing radius of the associated bi-tangent circles.

2.1.1 Medialness, Shape and Psychology

In the 1950's, Rudolf Arnheim (1974), a Berlin gestaltist who had emigrated to the United States and became professor of the Psychology of Art at Harward University, intuitively arrives at a notion similar to medialness and talks of the "structural skeleton" of a canvas and its object traces. Around the same time, Fred Attneave (1954) studies shows the importance of curvature features while observing 2D objects, where the outline of a shape is encoded in the form of curvature extrema, i.e, concavities or convexities. A proof was still lacking whether the human visual system captures these type of information or not.

To develop a geometry of visual form, an appropriate intermediate-level representation became a requirement (Kimia (2003)), which must be able to mediate between ambiguous, noisy regional fragments and entire object models. After the initial work of Harry Blum and his collaborators, in the 1960's and 1970's (Kotelly (1963); Blum (1967, 1973)), during the 1980's, Irving Biederman (1987) and his collaborators defined a shape based on a notion of parts and referred to these as elementary geometric primitives or *geons*. Over many years of psychophysics studies, it was shown how humans recognise more complex objects in terms of hierarchies of parts and their relationships. In the same period, Richards and Hoffman (1985) defined a shape as sequences of triplets of significant concavities and convexities, i.e., two concavities with an intermediate convexity, also known as (shape) codons.

In the 1990's Kovacs and Julesz (1993, 1994) presented a finding that the human visual system is more sensitive towards a closed contour than to open contours. Their experimental setup was basically based on a series of Gabor patches with varying orientations. These psychological experiments were aimed at finding correlates of perceptual organisation at the level where spatial interactions occurs among early cortical filters. While Polat and Sagi (1993, 1994) described the architecture of pairwise spatial interactions, Kovacs and Julesz (1993, 1994) described the activity pattern of a large number of interacting filters. By the late 1990's Kovács et al. (1998) derived an analogous form of medial-axis, called medial-point description. A major drawback of Blum's *MA* lies in the fact that it is sensitive to noise. The D_ε–function (definition is given in Section 1.4) makes prominent certain medial points along the *MA*, potentially leading to a very compact representation with greater robustness to noise than Blum's *MA*. Kovács (2010) refers to such points of high medialness as "hot-spots", which could play a key role in analysing animal forms, particularly in the case of articulated movements. A similar, but more direct and non-algorithmic technique is used for producing moving virtual characters in films and games, where a human wears a special suit with localised "hot-spot". The system is based on Computerised Movement Capture system (also commonly known as mo-cap, Amor et al. (2009)).

Some other studies in psychology are done on human-driven attention when presented with simple shape, such as rectangles and ovals. In such studies, non-expert human subjects are asked to place a dot within the outline of presented shapes (Psotka (1978)). The resulting cumulative patterns closely resemble "fat" Blum's *MA* loci. Recently, this study has been repeated by Firestone and Scholl (2014), using a computerised tablet (to support the interface) and asking random people on the street of New York City to tap on the screen. Here we note that the results not only resemble the *MA* type of responses but they reflect traces alike the results indicated by Kovács et al. (1998), i.e., higher cumulative values appear to be concentrated as "hot-spots".

Also in recent studies, Lescroart and Biederman (2013) indicated that parts representation and their relative orientation can be encoded in V3 and higher visual stages, while they are in support of *MA* based shape coding. Explicit coding of *MA* shape in higher-level object cortex (also known as inferotemporal cortex or IT) is further supported and demonstrated by Hung et al. (2012). They report that such IT neurons encode both the MA and contour segments simultaneously.

2.1.2 Scale-space Medialness Measurement

In the early 1990s, Kelly and Levine (1993) introduced the notion of contours as *symmetric enclosure* and used annular symmetric operators to detect contour configurations.

Two physical analogies for enclosure were proposed that provide the basis for selecting appropriate parameters. Later they showed that the subsequent grouping of symmetric points can result in a set of parts that makes it possible to identify the location of a target object within an image (Kelly and Levine (1995b)). Around the same time, Burbeck and Pizer (1995) showed how an object medialness can be sensed at multiple scales by "boundariness" detectors that give integrated responses. While these ideas were being developed in computer vision, Kovács et al. (1998) in perception and cognition studies had showed that a "biological shape" in movement can be summarised by a few informative points thereby defining a medial-point description of an object. A related model was then proposed more recently by van Tonder and Ejima (2003) who reformulated the maximal disk methods for calculating medial-points and presented the Hybrid Symmetry Transformation (HST), which is a mixture of shunting inhibition networks and wave dynamics. The maximal disk paradigm is made parametric (and can be implemented in hardware or a neural network) by introducing an adjustable shunting coefficient.

Multiscale medialness measurement on gray level images

In all the above measures, normally, the input images are in the binary form and hence the extracted boundary consists of either white or black pixels. In the field of medical imaging, Xu and Pycock (1998, 1999) introduced the Concordance-based Medial Axis Transform (CMAT) that computes medial evidences from gray-scale boundary measures. CMAT computation takes the advantage of a look up table (LUT) that contains a series of groups of boundariness points, where the groups are identified in second LUT. The medial responses are calculated at multiscale where progressively large blurring filters are used to produce a set of scale-space images, which discards the need of prior segmentation and avoids removing important boundary information at an early stage of processing. For each multiscale medial axis (MMA) point, the spatial coordinate, x, indicates the middle position of the object; the scale parameter, r, specifies the approximate width of the object at that position. The MMA curves are obtained by first computing medialness[1] over scale-space and then detecting scale-space ridges in the medialness function. The medialness function, $M(x, \sigma)$, is defined as the degree to which a position x resembles an object central locus when examined at a particular scale r (Xu and Pycock (1999)). Furthermore, Xu (2004) presented an improvement of the MMA, where a sliding window algorithm is used to detect locally optimal scale ridges in scale space, in order to distinguish different objects in the input image.

The method maintains a pair of LUTs, which can take a large memory portion if the images are of high resolution. The CMAT algorithm is not only computationally expensive, but also the representation is not concise and hence it can not be applied in real time system. The CMAT representation also suffers from a "halo" effect[2] and cannot be said as a true medialness definition because it aggregates the nearby boundary segments oriented towards the point in consideration, but they represent different object parts. The method also seeks an algorithmic formulation to make it compact, i.e., extract feature points from the map.

2.2 Background on Shape Analysis and Image Retrieval

Shape-based image retrieval consists of the measuring of similarity between objects represented by their features. In this section, we mainly focus on the representation and descriptive aspects of shape analysis. Shape representation methods spatially transform the object's outline shape (e.g. into a graph) so that the "important" or "most informative" characteristics of an object are preserved. Here, the word "important" is context sensitive

[1] The medialness at a particular position x in the image space is the integral of the boundariness contribution to the true medialness structure in a certain circular area, centered at that position x.

[2] The halo effect is a cognitive bias in which an observer's overall impression of an entity influences the observer's feelings and thoughts about that entity's character or properties.

and typically has different meanings for different applications. Shape description refers to methods that result in a numerical descriptor and is a step subsequent to shape representation. Typically, a *shape descriptor vector* (also called a feature vector) is generated. The goal is to uniquely characterise the object using its feature vector. The usually required properties of a description scheme are invariance to translation, scale and rotation. This implies that such transformations do not change the perceived shape of the object.

Looking back on recent developments, a shape can be described according to: Digital bending energy (Young et al. (1974)), Axis of least inertia (Tsai and Chen (1995)), Eccentricity (Peura and Iivarinen (1997)), Convexity (Peura and Iivarinen (1997); De Berg et al. (2000); Grunbaum (2003)), Hole area ratio (Soffer and Samet (1997)), Earth Mover's distance (Rubner et al. (2000); Ling and Okada (2007)), Solidity (Chang et al. (2001)), Circularity ratio (Zhang and Lu (2004)), Inner Distance (Ling and Jacobs (2005, 2007)), Center of gravity (Yang et al. (2008)), Elliptic variance (Yang et al. (2008)), Height function (Wang et al. (2012)), Multiscale triangular representation (Mouine et al. (2013b)). Such parameters can then be used to define various types of shape descriptors.

A number of recent review papers (Loncaric (1998); Jain et al. (2000); Biederman (2001); Zhang and Lu (2004); Yang et al. (2008); Poppe (2010); Weinland et al. (2011); Cope et al. (2012)), as well as books (Serra (1983); Amit (2002); Forsyth and Ponce (2002); Osher and Paragios (2003); Davies (2004); Berg and Malik (2006); Siddiqi and Pizer (2008)) have been written on the subject of shape analysis.

Shape analysis methods can be classified according to many criteria. We organise these methods in two main groups: (a) contour vs region based methods, and (b) feature based methods. We further compare each of the methods based on these criteria: (i) provide the definition of concavity and convexity in the shape, (ii) able to describe and preserve the topology, (iii) remains invariant with affine parameters changes, (iv) provide part description, (v) able to handle articulation, and (vi) scalability.

2.2.1 Contour vs Region based Methods

In the early 1990s, Vernon (1991) presented an overview on the mathematical description of shape using boundary features and categorised into four distinct types of shape descriptors: (a) external scalar transform techniques utilizing features of the shape boundary; (b) internal scalar transform techniques utilizing features of the shape region; (c) external space domain techniques utilizing the spatial organization of the shape boundary; and (d) internal space domain techniques utilizing the spatial organization of the shape region. During the mid-1990s, object deformation was another aspect to cover in the shape representation and detection. Jain et al. (1996) introduced a statistical approach of object matching by using deformable templates representing the object's class. They were trying to solve the problem of locating and retrieving an object from a complex image using its 2D boundary information. To do so, a set of parametric transformations were used to deform the template and a probabilistic distribution (Bayesian inference scheme) was defined on the set of deformation mappings to bias the choice of possible templates.

In Contour based approaches shape features are first extracted, helping to find relevant images from a shape database (Zhang and Lu (2004)). The key idea behind contour based approaches is to define shape similarity measures for example by analyzing convex and concave curve segments of an object's outline (Latecki et al. (2000)). In order to train the retrieval method on the shape database while considering affine changes and deformations, one possibility is to use deformable templates consisting of the representative contour (Jain et al. (1996)). Later, a shape retrieval process was proposed by shape similarity using local descriptors and effective indexing. To create the feature descriptor, shapes were partitioned into tokens in correspondence with their protrusions, and each token was modelled according to a set of perceptually salient attributes (Berretti et al. (2000)).

Other main approaches to define similarity include: (i) providing a parametric equation of the contour and convolving it with Gaussian kernel (Veltkamp (2001)); (ii) contour point correspondence and computing an alignment transform between two shapes (Belongie et al. (2002); Chui and Rangarajan (2003)); (iii) B-spline modeling of the 2D planar curve (Wang and Teoh (2004)); (v) computing (modified) skeletons (Aslan et al. (2008)).

Recent detailed reviews on shape representation, description and matching techniques are reported by Stegmann and Gomez (2002); Zhang and Lu (2004); Yang et al. (2008). We now detail the most relevant contour vs region based methods that relate to this thesis. These methods can be further sub-categorised as:

1. Contour-point distance based methods
2. Curvature-based contour partitioning methods
3. Medial-axis transform and skeleton-based methods
4. Shape deformation
5. Shape decomposition
6. Other popular contour-based methods

2.2.1.1 Contour-point distance based methods

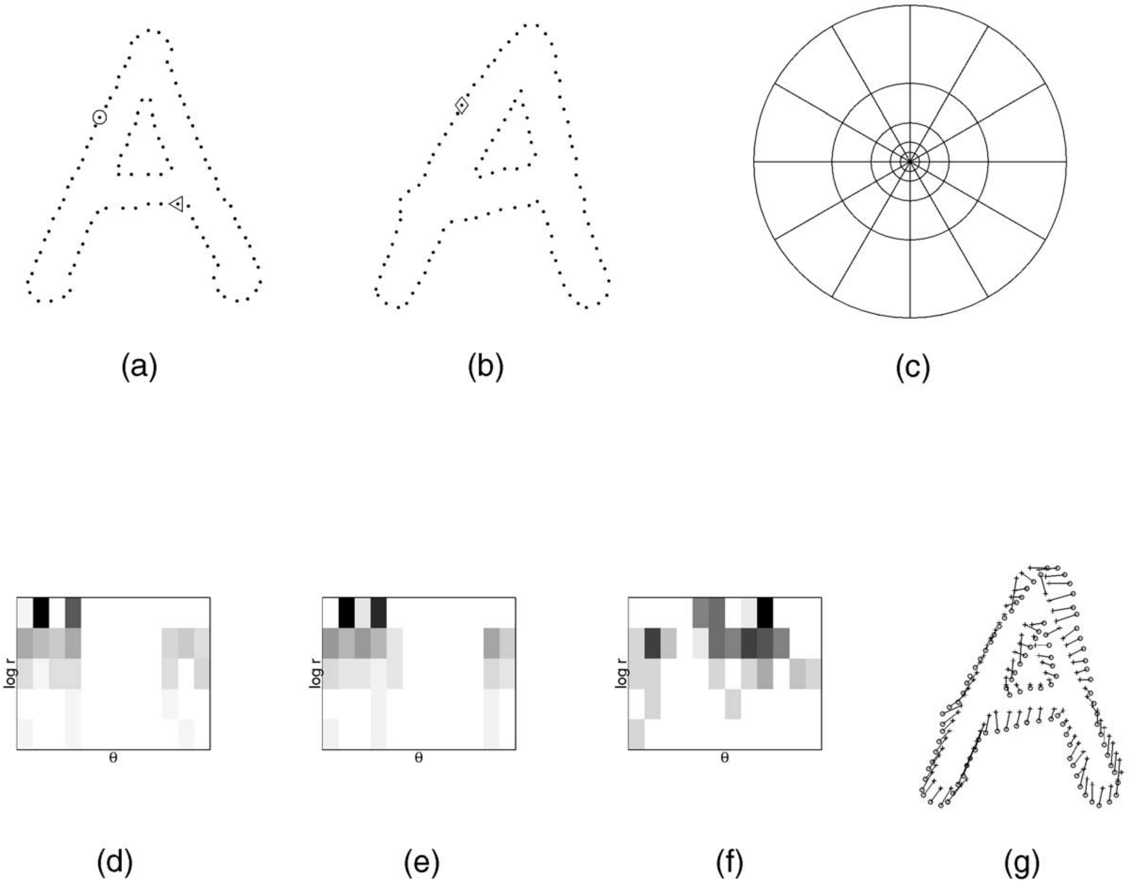


Figure 2.2: Adapted from Belongie et al. (2002), Figure 3: Shape context computation and matching. (a) and (b) Sampled edge points of two shapes. (c) Diagram of log-polar histogram bins used in computing the shape contexts. Here, they use five ranges for log r and 12 ranges for θ giving 60 bins. (d), (e), and (f) Example shape contexts for reference samples marked by ○,♢, ◁ in (a) and (b). Each shape context is a log-polar histogram of the coordinates of the rest of the point set measured using the reference point as the origin (dark=large value). Note the visual similarity of the shape contexts for and ○ and ♢, which were computed for relatively similar points on the two shapes. By contrast, the shape context for ◁ is quite different. (g) Correspondences found using bipartite matching, with costs defined by the χ^2 distance between histograms.

Most approaches equate the task of shape-matching to the matching of the respective object boundaries. The shape boundaries are often discretized into a set of n landmark points for easier representation and matching. Belongie et al. (2002) argued that these points could be located at any place on the object boundary and that they need not be restricted to extrema points on the curve. They also defined the notion of using *shape contexts* (SC) at each of these sampled points (Figure 2.2). Each SC is given by the relative distribution of the rest of the $n-1$ points which is represented as a 2D histogram of distances and angles, i.e., a histogram of the relative polar coordinates of all other points. SC can be made invariant to translation, rotation, and scale. However, while SC matching performs relatively well on rigid objects, it does not capture parts and handle articulations. This is because the SC histogram is composed of spatial Euclidean distances and set angles. To overcome this problem, Ling and Jacobs (2005, 2007) proposed a variant of SC, the Inner Distance Shape Context (IDSC), which uses the length of a shortest path connecting two boundary points, such that the path lies completely within the shape. The IDSC can be applied to articulated objects, but it is not designed to characterise the interior shape variations, e.g. in terms of widths, necks, indentations. In the recent work on boundary based methods, Nasreddine et al. (2010) defined geodesics based a multi-scale distance between shapes in order to address local and global variabilities. This similarity measure is invariant to affine parameters. Mostly, these boundary-based and contour point distances based methods are robust towards matching the static 2D shapes, but their accuracy reduces drastically when there is deformation or occlusion or holes in the shape. In the more recent work by Mouine et al. (2012, 2013a,b), the *SC* is fused with a multiscale triangular description where the latter is associated to a sequence of N sample points uniformly distributed over the contour and numbered in a clockwise order. Each triangle is further given its: (i) Triangle Area Representation (*TAR*, defined as $TAR(T) = A(T)$, where $A(T)$ is the signed area of triangle T[3]), (ii) Triangle side lengths representation (TSL, defined as ratios of two smaller triangle sides with the larger ones), (iii) Triangle represented by two side lengths and an angle ($TSLA$, defined as (TSL, θ), where θ is the absolute value of the vertex angle at point in operation, i.e., p_i), and (iv) Triangle represented by two oriented angles (TOA, defined as two successive angles, i.e. ($\angle p_{i-k}p_ip_{i+k}$ and $\angle p_ip_{i+k}p_{i-k}$)). This approach is affine-invariant, robust to noise, works well on occluded objects and intuitively captures some coarse information on local extremities. However, this method fails to quantify the true extremities and hence cannot in our evaluation produce an appropriate and promising part description. It also cannot handle well significant object's deformations and articulation. We note also that, while the multiscale triangular based method works well on rigid objects, it is not directly able, in general, to handle flipped (or mirror-inverted) cases.

Concave/Convex These methods do not provide any explicit definition to the concavity and convexity in the shape. The shape boundaries are often discretized into a set of n landmark points and need not be restricted to extrema points on the curve, therefore sampled points may or may not contain the concavity (minima) and/or convexity (maxima) along the bounding contour. The TAR-based approach partitions the shape into triangles and further calculates parameters such as signed area of triangle, TSL, TSLA, & TOA to find the coarse information on local extremities. Hence this heuristic approach fails to quantify the true extremities in the shape.

Topology The descriptors created by these methods do not hold topology information. Nature of these descriptors change with a small deformation in the shape.

[3] If (x_{i-k}, y_{i-k}), (x_i, y_i) and (x_{i+k}, y_{i+k}) are three respective coordinates of the points p_{i-k}, p_i and p_{i+k}, then the signed area of the triangle T formed by this triplet is given as: $\frac{1}{2}\begin{bmatrix} x_{i-k} & y_{i-k} & 1 \\ x_i & y_i & 1 \\ x_{i+k} & y_{i+k} & 1 \end{bmatrix}$.

Affine Invariance These methods remain invariant with affine changes.

Part-description None of these contour-point distance based methods provide explicit definition of parts. However, IDSC finds the best matching of parts in terms of shortest path connecting two boundary points.

Articulation Up to a certain extent SC and IDSC descriptors remain invariant toward shape articulations, but they do not inform explicitly on the behavior of articulations, i.e., which part of the shape is moved and where it is moved.

Scalability On the one hand the computational complexity of these methods is high while on the other hand the descriptors cannot be indexed for matching tasks. Therefore, these methods cannot be scaled on a big shape database.

2.2.1.2 Curvature-based contour partitioning methods

Another perspective on shape definition is to seek to split up the boundary using curvature. The approach can be loosely categorised into two groups of techniques: (a) approximation based, and (b) curvature (analytic) based. Important examples of approximation techniques include that of Nelson and Selinger (1998) who detect *key curves*: long segments of an *edgel-chain*[4] comprised between two high discrete curvature points. A key curve's size and orientation define a square image patch, which is then described using all edgels falling within it. Such edge patches prove to be robust to occlusions and some level of clutterness. However, these patches are also likely to include clutter edgels lying near the object boundary, which may corrupt their descriptive power and make it difficult to proceed with matching tasks. Ferrari et al. (2008) improved this method by introducing a scale-invariant local shape features called PAS (pairs of adjacent segments), by using

[4] Chained group of pixels in an image that is recognised as the edge of an object; this goes back to the classic Freeman chain-code description of a discretised contour.

chains of k *adjacent segments* (kAS). First, the Berkeley natural boundary detector (Martin et al. (2004)) is used to detect the edgels which are further chained and partitioned into roughly straight contour segments. Then, a complex branching structure is formed by organizing these segments in a network by connecting them along the edgel-chains and across their links. Depth-first search is used to form the kAS where the complexity of features is directly proportional to k. On one hand, by increasing the value of k the feature sets become more and more informative while on the other hand the repeatability[5] reduces gradually. kAS are capable to encode fragments of an object boundary in a robust way without including nearby clutter. A support vector machine (SVM) is used for the training stage, where the shape descriptors are based on a sliding window object detection scheme and separate positive examples from negative ones. They rely on evaluating a quality function, e.g. a classifier score, over many rectangular subregions of the image and taking its maximum as the object's location. Because the number of rectangles in an image of size $n \times n$ is of the order n^4, this maximization usually cannot be done exhaustively. Typically, these consist of reducing the number of necessary function evaluations by searching only over a coarse grid of possible rectangle locations and by allowing only rectangles of certain fixed sizes as candidates. In more recent work (Ferrari et al. (2010)), a principal component analysis (PCA) is used to learn the deformation model, while a class model is learned by localizing the novel instances in presence of intra-class variation, clutter and scale changes. A Hough style voting technique with a non-rigid point matching algorithm (TPS-RPM) is applied to localize the object in cluttered images. Furthermore, the objective function of the TPS-RPM is extended by adding two terms: (i) minimize the orientation difference between corresponding points, and (ii) maximize the edge strength of matching image points. The approximation methods not only computationally expensive, but also they are non-scalable, because they do not produce any feature vector and

[5] Repeatability is defined as the variability of the measurements obtained by one person while testing and retesting of an event repeatedly. This is also known as the inherent precision of the measurement equipment.

hence objects cannot be indexed.

In contrast to approximation methods, Berretti et al. (2000) described a shape as the analytic curvature function and further partitioned the shape at the minima of this function. The curvature function is computed by parameterising the Gaussian smoothed planar curve according to its arc-length. Each partition (token) is further described via the combination of $editing\ distance$ and $orientation$ while final shape indexing is performed by the application of M-tree. The main drawback of this method is its sensitivity to noise. Later, Felzenszwalb and Schwartz (2007) used the curvature function to represent a shape in a hierarchical form, called $shape-tree$ deformation model, which captures both global and local geometric properties. An elastic matching between two shapes is further performed by finding the correspondence between two curves. One advantage of such a technique is that it can handle some level of articulations. However, this group of techniques based on computing curvature also suffer from the same drawback alike approximation methods; i.e., mainly, no obvious indexing scheme is available.

Concave/Convex Both the approximation and curvature (analytic) based methods hold the information of minima and maxima points on the closed contour, and hence the concavity and convexity in the shape.

Topology The topology information is not always preserved as the small deformation changes the nature of curvature function. However, some methods use PCA to learn the deformation in the shape and tries to preserve the topology.

Affine Invariance Change in rotation and scale modify the behavior of curvature functions, and as a result these curvature-based methods do not hold affine invariance property.

Part-description Curvature function partitions shape at the minima of function, but it doesn't confirm that these partitions describes the correct parts of the shape. Approximation based methods totally fails in providing the part descriptions to the shape.

Articulation Approximation-based methods do not work well on articulated objects, while the analytic curvature function can handle some level of articulations, but fails when the object is moderately articulated. In such cases, if a shape is articulated moderately, the nature of the curvature function for the modified shape differ from the original one.

Scalability These methods do not produce any feature vector and hence no obvious indexing scheme is available. Therefore these methods are not scalable.

2.2.1.3 Medial-axis transform and skeleton-based methods

The most popular and the most studied space domain method is the medial axis transform (MAT) originally proposed by Blum (1967, 1973). The idea of this approach is to represent ab object by a graph where important shape features are made explicit. A skeletal pair consisting of the skeleton and the "quench" function is used by collaborators of Blum, Calabi and Hartnett (1968). This approach is motivated by the study of neural physiology and visual psychology. In particular, Blum hypothesized that the process of image formation on the retina is a chain reaction in the following sense. When an object image is formed on the retina a certain number of neurons are excited, lowering the excitation levels of neighboring neurons and causing them to fire a short interval of time later. This process is repeated until the whole area of the object is "tiled" with ring neurons. The inhibited neurons cannot fire again for a short time due to the underlying neurophysiological processes (Shepherd (1974)). Therefore the wave front of the ring cells cannot move back towards the retinal areas containing inhibitory neurons. This mechanism is similar to the spreading of a prairie fire. In fact, the first approach Blum used was a temporal function showing the arc length of wave front versus time. This approach did not prove very useful for shape description purposes. Blum's second concept, the concept of medial axis, has proven to be much more useful.

Later, Blum and Nagel (1978) proposed a *generalized medial axis* transform, based on the touching disk defined as a circle which is tangent to the shape boundary without intersecting it. The *r-symmetric axis* of a shape is then defined as the union of all points that have a touching circle of radius greater than *r* and at least two points that touch the boundary. The requirement that the radius is greater than *r* prevents little noisy spikes from generating skeleton segments. The two touching point requirement limits the selection to only skeleton points. The generalized skeleton has proven to have better noise robustness. This geometric view is the fundamental idea behind the concept of medial axis (Blum (1967)), which has found a variety of applications in describing the shape of virtually all kinds of objects from the infinitely large to the infinitely small including biological entities (Leymarie and Kimia (2008)).

In the early 1990s, Leymarie and Levine (1992) developed a new method for the extraction of symmetry axes which does not suffer from the discretization problems that many other algorithms do have. This method was based on the use of *snakes* for active contour representation, high curvature points on the boundary, and symmetry axis transform. The result was a dynamic multi-scale skeleton representation. Another variant of the medial axis, called *shock graph*, came into existence in the mid 1990s, when Siddiqi and Kimia (1996) defined this concept as an abstraction of the medial axis of a shape onto a direct acyclic graph (DAG). A shock graph decomposes a shape into a set of hierarchically organized primitive parts. Shock segments are curve segments of the medial axis with monotonic flow, giving a more refined partition of medial axis segments. The skeleton points are first labeled according to the local variation of the radius function at each point. A disadvantage of such an approach is that the calculation for evaluating the distance between two shock graphs is of polynomial time, a complexity making the method not scalable.

In more recent developments, the skeleton has been combined with its associated contour, where the skeletal features are robust against articulation and non-rigid transformation, while contour features are more stable and informative with respect to global and affine transformations (Bai et al. (2009a)). This type of integration captures both local as well as global shape information which is later used for shape classification. Here a generative model is used to compute the shape likelihood, where first the contour segments are extracted using a discrete curve evaluation and later a polygon approximation. Then a skeletal path (or graph) is evaluated by detecting end points, junction points, and connection points. Both the contour and skeletal features are normalized and combined to perform learning over the dataset, later used for classification purpose. However, for more accurate and robust recognition, the skeleton branches are needed to be pruned (Shaked and Bruckstein (1998); Choi et al. (2003); Bai et al. (2007)). Moreover, another major restriction of recognition methods based on skeleton is a complex structure of resulting trees or graph representations of the skeletons. Graph edit operations are applied to the graph structures, such as merge and cut operations (Zhu and Yuille (1996); Liu and Geiger (1999); Pelillo et al. (1999); Di Ruberto (2004)), in the course of the matching process. The most important challenge for skeleton similarity is the fact that the topological structure of skeleton graphs of similar objects may be completely different (Bai and Latecki (2008)). Skeleton-based approach are further used for non-rigid object detection (Bai et al. (2009b)), where each branch on the skeleton is associated with a few part-based templates and the object boundary information.

Concave/Convex The convexity or maxima in the shape is described by the medial-axes endings.

Topology Topology is preserved because the medial axis can be deformed back to the original contour, as it holds distance to contour information

Affine Invariance These methods remains invariant with affine changes and produce the same results under scaling, rotation and translation.

Part-description In medial-axes based approaches, each branch or loop is treated as a potential part. However, small boundary-based perturbation results into towards spurious branches. In the recent development (Bai et al. (2007); Bai and Latecki (2008); Bai et al. (2008, 2009b)) these spurious branches are pruned and more robust medial axes are produced.

Articulation The medial axis tree structure (in 2D) can deal with articulations to the extent that occlusions are not complete.

Scalability The polynomial time one-to-one matching makes these method hard to scale on big database. In the best case scenario of medial-axis transform, the matching complexity is $O(n \log n)$ or pseudo-linear, which is still hard to scale it with the growing database (1000 times increment in the database increases the retrieval time by the factor of 1000).

2.2.1.4 Deformation Models

Shapes have also been described via deformation models. In this particular approach, the objective is to solve the problem of locating and retrieving an object from a complex image using its 2D shape/boundary information. In this particular approach, one of the important early work is done by the team of Jain et al. (1996). On one hand, their probabilistic transformation based deformation model is an extension of Grenander and his colleagues (Chow et al. (1989); Amit et al. (1991); Miller et al. (1993)), while the potential functions is taken from the “snake-model” introduced by Kass et al. (1988). They obtained the deformed templates by applying Bayesian objective function to the prototype, while variability in the template shape is acquired by imposing a probability distribution function. Their method is based on the 2D planar rubber sheet principle which has closed boundary, but it can be deformed by twisting, squeezing, and stretching. The technique is too slow for practical use, because the possible non-rigid transformations of a template is very large. In order to solve this problem, Felzenszwalb (2005); Felzenszwalb and Schwartz (2007) described a global optimal solution to the non-rigid shape matching problem, where a shape deformation is determined through the triangulated polygonal method: here, the shape (using the contour information) is divided into various triangles to generate a polygonal structure. Furthermore an energy function is estimated, one for each triangle, which is the sum of two terms: a data term, which attracts the deformed model towards salient image features, and a penalty term, which penalises large deformations of the model. The final shape matching problem is based on dynamic programming, where the objective is to determine the optimal fitting map that minimizes the energy, i.e., corresponds to the best location for the deformable template in the image. This work is further improved by Kim and Shontz (2010), where they modified the definition of the energy function to consists of a cost term that considers the center of mass of an image and is invariant to translation, rotation and uniform scaling. Their shape matching process is also based on a dynamic programming method and consists of three steps: (i) determining the boundary vertices, (ii) generating a triangular mesh using a constrained Delaunay method, and (iii) finding the optimal mapping from the source image to the target image that minimises the energy function. In more recent work, Pan et al. (2012) used angle gradients to extract critical points of the contour. A pairwise similarity measure is then evaluated that uses a spectral technique to solve a quadratic assignment problem.

Although shape deformation models are affine invariant, they tend to be very slow. Moreover, a deformation model does not bring any explicit definition of a shapes through its parts and hence the recognition rate reduces drastically when an object is occluded.

These template based matching methods also do not support feature vectors, and as a result the indexing over a large shape dataset is not practical.

Concave/Convex Deformation models do not conform on boundary minima and maxima points and hence do not provide explicit concavity and convexity points in the shape.

Topology Deformation models are able to hold the topology of a shape and remains invariant under shape deformation.

Affine Invariance These methods remains invariant under affine changes.

Part-description Deformation models uses triangulation method to partition the shape. However, they do not provide any spatial correlation and behavior of such partitioning.

Articulation An articulated movement produces different types of deformation in the shape. The model finds the corresponding matching original object via optimal mapping. Therefore, deformation models work very well under shape articulations, but they not inform on semantics of articulation, i.e. what type of articulation is performed, which part is moved and where it is moved.

Scalability These methods do not support feature vectors, and hence can not be indexed over a large shape dataset.

2.2.1.5 Shape decomposition

In shape decomposition, an object is represented as a combination of primitive parts. The shape decomposition task has been done in several ways focused on: (i) boundary-points, (ii) stroke detection, and (iii) inner distance. Early on Pavlidis (1980) stated the problem of global shape analysis (decomposition) as: among the boundary points find sets which are closely related. Such sets may be used to assign labels to corresponding parts of the object. In this approach, shape decomposition is based on the properties of boundary points. Several authors have used this approach for shape decomposition. Decomposition criteria can be defined in terms of the medial-axis transform, can require convex components or visibility of boundary points. In the medial axis transform approach, two boundary points are labeled related if they are both on the circle contained in the object shape. Decomposition criteria can be formulated to require that the line segment between two points on the boundary is contained in the shape that is described. This kind of criterion leads to a decomposition into convex components.

In stroke detection approaches points are related if they are close to each other across the boundary. Semantic considerations about shapes being described were taken into account in the method for shape decomposition by collinearity (Kim et al. (1987)). di Baja and Thiel (1994) developed a method for shape decomposition by its weighed skeleton; this method is well designed for elongated, ribbon-like objects. Berretti et al. (2000) partitioned the shape into segments in correspondence with their protrusions, and each such segment is modelled according to a set of perceptually salient attributes. Latecki and Lakamper (2000) converted an object's outline into a polygon by applying a polygonal approximation method. The polygonal boundary is then further decomposed into visual parts based upon a notion of convex boundary arcs. These visual parts are grouped together in the order of their connection for the measurement of shape similarity.

Ling and Jacobs (2005, 2007) proposed shape descriptors by using the inner-distance model, which can capture the part structure of the shape. The inner distance is defined as the length of a shortest path between landmark points while being constrained to stay within the shape silhouette. To improve the shape classification result, along with shortest path, texture information is also added. Chen et al. (2008) proposed a partial shape matching with the help of local and global shape descriptors. A global shape descriptor at a feature point takes into account the position of all other points relative to this point, while a local shape descriptor depends only on the neighboring points. The mathematical formulation of the shape features are dependent on turning angle and distance across the space (DAS). For shape matching a dynamic programming approach is used that depends on the designed similarity function and a gap function, which determines the cost of deleting or inserting a feature point x.

The above shape decomposition methods work well mainly in determining the articulations and shape by its parts. However, they cannot be converted into a reliable feature vector and hence they do not explicitly support indexing and are non-scalable.

Concave/Convex Since these methods are based on convex components analysis, hence they provide the information on shape extrema.

Topology The deformation in the shape does not change the nature of such methods and the descriptor remain invariant under such transformation. Therefore shape decomposition models preserved the topology of the shape.

Affine Invariance The shape decomposition method remains invariant under affine changes.

Part-description These methods provides explicit definition of the shape parts via convex components.

Articulation The shape articulations are predicted via its parts. Therefore these methods are capable in handling articulations.

Scalability These methods do not support feature vectors, and hence can not be indexed over a large shape dataset.

2.2.1.6 Other popular contour-based methods

Other popular methods used for shape description, recognition and matching tasks are: generative model, expectation-maximization (EM), histogram of the contour points distribution (CPDH), height function. Tu and Yuille (2004) proposed a generative model that allows for a class of transformations (affine and non-rigid), and induces a similarity measure between shapes. For the matching task, the EM algorithm is used. Tsai et al. (2005) also used the EM approach to organise a shape database into different classes and simultaneously estimate the shape contours to give descriptions of each of the different shape classes. In the histogram-based technique, Shu and Wu (2011) define a shape descriptor by evaluating the histogram of the contour points distribution (CPDH descriptor). To construct this CPDH descriptor, first the object centroid is calculated and the maximum value of the distance between the centre and a point on the contour is selected as radius to build a minimum circumscribed circle. The region of this minimum circumscribed circle is divided into several bins, by using a concentric circle and equal interval angle and a histogram is calculated by using such bins. One of the major drawback of this system is that it is not fully invariant to rotation. However, to resolve the rotation invariant problem to some extent, the authors adopted a circular shift and mirror matching scheme (random images are selected from the database and rotated in clockwise and anticlockwise directions to 12 selected angles). The CPDH extraction mainly involves contour extraction, points sampling, counting and calculating the distances between the sampled points and the object centroid. The shape similarity is measured by the Earth Mover's Distance (EMD, Rubner et al. (2000); Ling and Okada (2007)) metric by matching different CPDHs: Bin-By-Bin Dissimilarity Measures, Cross-Bin Dissimilarity Measures, and Parameter-Based Dissimilarity Measures. Through a voting scheme, the minimum among the three is chosen as the final shape distance. Also, the recall rate reduces rapidly as a function of the movement and perturbation of shapes.

In another recent contour-based approach, Wang et al. (2012) used a height function to generate the shape descriptor. In height function, the contour of each object is represented by a fixed number of sample points. For each sample point, height function is defined based on the distances of the other sample points to its tangent line. At the end, a compact and robust shape descriptor is obtained by smoothing the height functions that is invariant to affine changes and insensitive to nonlinear deformations. For matching, a dynamic programming algorithm is used to find the optimal correspondence between sample points of two shapes. The method lacks scalability because the shape descriptors are not convertible into a reliable feature vector that can support indexing over a large shape database.

2.2.2 Feature-based Methods

An important work in feature-based methods was initiated by David Lowe (1999, 2004) in the late 1990s who presented a method for extracting distinctive invariants from an image for matching purpose using a Difference of Gaussian (DoG) filter through which a substantial amount of information about the spatial intensity pattern would be captured, with a goal of making the matching algorithm scale invariant (SIFT: Scale Invariant Feature Transform, detailed in Section 2.2.2.1). In a related approach, Brown et al. (2005) localized features using the Harris Corner detectors while the matching task was performed by a fast nearest-neighbor algorithm. Very similar to the SIFT descriptor is the gradient location and orientation histogram (GLOH) descriptor of Mikolajczyk and Schmid (2005) that replaces the Cartesian location grid used in SIFT with a log-polar one, and applies PCA to reduce the size of the descriptor. Later refinements included: Speeded-Up Robust Features (SURF, Bay et al. (2008)), binary robust independent elementary features (BRIEF, Calonder et al. (2010, 2012)) and Oriented Fast and Rotated BRIEF (ORB, Rublee et al. (2011)) descriptors, which seek to reduce the execution time by optimising various parts of the feature computations and matching algorithms.

Such feature-based methods exploit the most relevant features for object detection or classification that provide invariance to changes in illumination, differences in viewpoint and shifts in object contours. Such features can be based on points (Harris and Stephens (1988); Mikolajczyk and Schmid (2002)), blobs (Laplacian of Gaussian (LoG, Lindeberg (1998)) or Difference of Gaussian (DoG, Lowe (1999, 2004))), gradients (Mikolajczyk and Schmid (2004)), Histogram of Oriented Gradient (HOG) (Dalal and Triggs (2005)), texture, color, or combinations of several. These approaches extract local image features at a sparse set of salient image points – usually called points of interest or key points. The final detectors are then based on feature vectors computed from these key point descriptors. The hypothesis is that key point detectors select stable and more reliable image regions, which are especially informative about local image content. The overall detector performance thus depends on the reliability, accuracy and repeatability with which these key points can be found for the given object class and the informativeness of the points chosen. For the detection and classification task, the final descriptors need to characterise the image sufficiently well.

2.2.2.1 Laplacian or Gaussian based methods

The use of salient local points or regions for object detection has a long history (Schiele and Crowley (1996); Lindeberg (1998); Lowe (1999, 2004); Mikolajczyk et al. (2005); Bay et al. (2006, 2008); Calonder et al. (2010); Rublee et al. (2011); Calonder et al. (2012)). In his work, David Lowe (1999, 2004) presented a method for extracting distinctive invariants from an image for matching purpose using a Different of Gaussian (DoG) filter through which a substantial amount of information about the spatial intensity pattern would be captured, with a goal of making the matching algorithm scale invariant (SIFT algorithm). The input image is successively smoothed with a Gaussian kernel and sampled. The DoG representation is obtained by subtracting two successive smoothed images. Thus, all the DoG levels are constructed by combined smoothing and sub-sampling. SIFT then computes histograms over rectangular grids. The local 3D extrema in this pyramidal representation determine the localization and the scale of the interest points. The DoG operator is a close approximation of the LoG function (Lindeberg (1998)) but the DoG can significantly accelerate the computation process Burt (1980); Lowe (1999, 2004)). The common drawback of the DoG and the LoG representation is that local maxima can also be detected in the neighborhood of contours or straight edges, where the signal change is only in one direction. These maxima are less stable because their localization is more sensitive to noise or small changes in neighboring texture. In a related approach, Brown et al. (2005) localized features using the Harris Corner detectors while the matching task was performed by a fast nearest-neighbor algorithm.

Very similar to the SIFT descriptor is the GLOH descriptor proposed by Mikolajczyk and Schmid (2005), which replaces the Cartesian location grid used in SIFT with a log-polar grid, and applies PCA to reduce the size of the descriptor. Another SIFT like approach is SURF (Bay et al. (2006, 2008)), which refines the definition of detector and descriptors to perform a fast and robust computation and comparison. The detector is based on the Hessian matrix and uses Gaussian second order partial derivatives in y-direction and x,y-plane as box filters. The filter responses are further normalised with respect to the mask size while descriptors are extracted over a square region as Haar wavelet responses. A fully affine invariance of the SURF algorithm is presented by Pang et al. (2012), where they normalize all six affine parameters along with selecting the number of latitudes and longitudes and use epipolar geometry to filter out false matches.

2.2.2.2 Histogram Based

Dalal and Triggs (2005) showed the feature extraction process for human detection in which the detector window is tiled with a grid of overlapping blocks to extract the HOG feature vectors. For this, the image window is first divided into small cells, where for each cell a local 1-D histogram of gradient directions or edge orientations over the pixels of the cell are piled up. For better invariance to illumination and shadowing, local contrast-normalization is done. Chandrasekhar et al. (2012) used histogram-of-gradients descriptor and directly captured the gradient distribution by using a histogram binning scheme. They established a distance function between descriptors in the compressed domain to discard the need for decoding. Ren and Ramanan (2013) proposed Histograms of Sparse Codes (HSC) - a sparse representation for object detection, where the sparse codes are computed with dictionaries and form local histograms by aggregating per-pixel sparse codes. Further, a dimensionality reduction is applied by computing supervised singular value decomposition (SVD) on the learned models.

2.2.2.3 Other Recent Popular Methods

The Earth Mover's Distance (EMD) was first proposed by Rubner et al. (2000), which is the measure of dissimilarity between signatures that are compact representations of distributions. EMD is a *cross-bin* distance function that addresses the histogram alignment problem raised due to shape deformation, non-linear lighting change, and heavy noise. EMD defines the distance between two histograms as the solution of the transportation problem that is a special case of linear programming. Ling and Okada (2007) improved the EMD method by reducing the number of unknown variable and constraints.

The early work of Lepetit et al. (2004) is based on the point matching approach for object pose estimation based on classification. The system is first trained by synthesizing a large number of views of individual keypoints and then later a compact description of this view set is produced by using statistical classification tools. For validation, *k*-mean in addition to Nearest Neighbor classifier is used. Later, Lepetit et al. (2005) used randomized trees as the classification technique which not only matches keypoints but also selects those keypoints during a training phase that are the most recognizable ones. Calonder et al. (2010) proposed to use binary strings as feature point descriptors (BRIEF), which are defined over a function of bitstring. The descriptors' similarity measure is evaluated using the Hamming distance. Later, Calonder et al. (2012) showed the fast computation of binary descriptor and matching, on the basis of simple intensity difference tests.

Other recent works of interest include Hough transform based class detection (Fast PRISM : the Principled Implicit Shape Model, Lehmann et al. (2011)), FAST-based (Features from Accelerated Segment Test, Rosten and Drummond (2006)), BRISK (Binary Robust Invariant Scalable Keypoints, Leutenegger et al. (2011)) descriptor which is composed as a binary string by concatenating the results of simple brightness comparison tests, improvement of best bin first (BBF) search algorithm (Liu et al. (2011a)) by introducing more features in k-d tree to perform fast and robust search, comparative evaluation of SIFT, SURF, ORB, BRIEF, BRISK etc. (Heinly et al. (2012)), and the radial gradient transform (RGT) and a fast approximation (Takacs et al. (2013)).

2.3 Conclusion

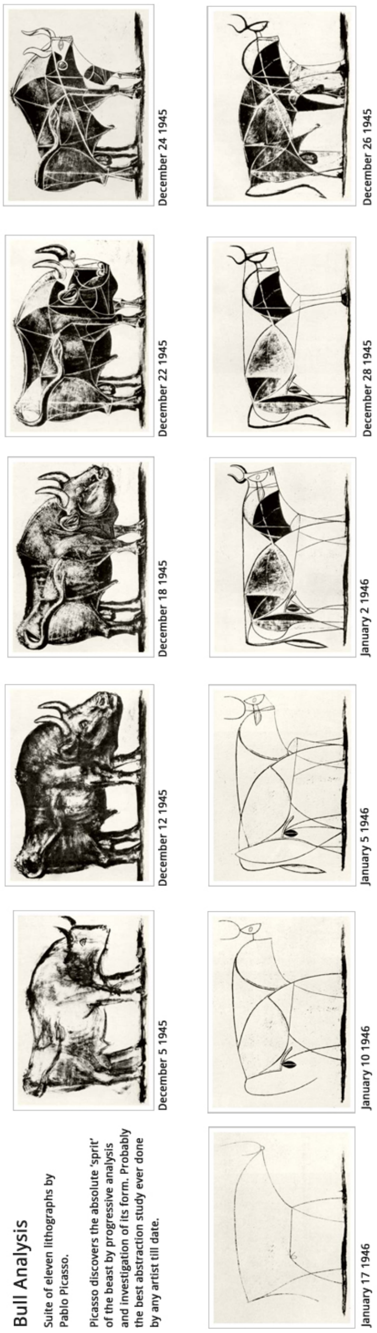
A common critique of medial-axis based methods is their high sensitivity to small perturbations and noise and their relative lack of scaling with large problems (large database of targets). On one hand, curvature extrema can provide the most important points on the object's boundary, however, they encounter all the drawbacks that edge-based representations have to face. As a result, the spatial organisation of the object components is not made explicit. On the other hand, medial-axis based representations are compact, but not compact enough, i.e., it is not as local as the point-based curvature extrema methods. Kovács et al. (1998) medial-point representation has the benefits of both representations, i.e., it defined local points, and it is region-based, but it lacks making a distinction between neighboring boundary segments part of separate object regions. Also, there is no description on the extraction process of these points and what do they correspond to.

A common drawback of SIFT-like approaches is that they do not consider an explicit shape representation of an object, and in consequence their accuracy reduces drastically for detecting closely related objects (with similar shape) but with different textures*: i.e.* the local intensity variations take over the overall shape and parts organisation of an object. Our proposed method combines these approaches in order to use their strong points and avoid the limitations of each category of methods when considered separately.

Table 2.1: Comparative Table of Approaches

	Concave/Convex	Topology	Affine Invariance	Part-Description	Articulation	Scalable
Contour	√					
Medial-Axis	√	√	√			
Shape-Context			√		√	
Inner-Distance			√	√	√	
Features	√		√			√
Primitives		√	√	√		

A comparative analysis of different approaches has been shown in Table 2.1.



Bull Analysis

Suite of eleven lithographs by Pablo Picasso.

Picasso discovers the absolute 'sprit' of the beast by progressive analysis and investigation of its form. Probably the best abstraction study ever done by any artist till date.

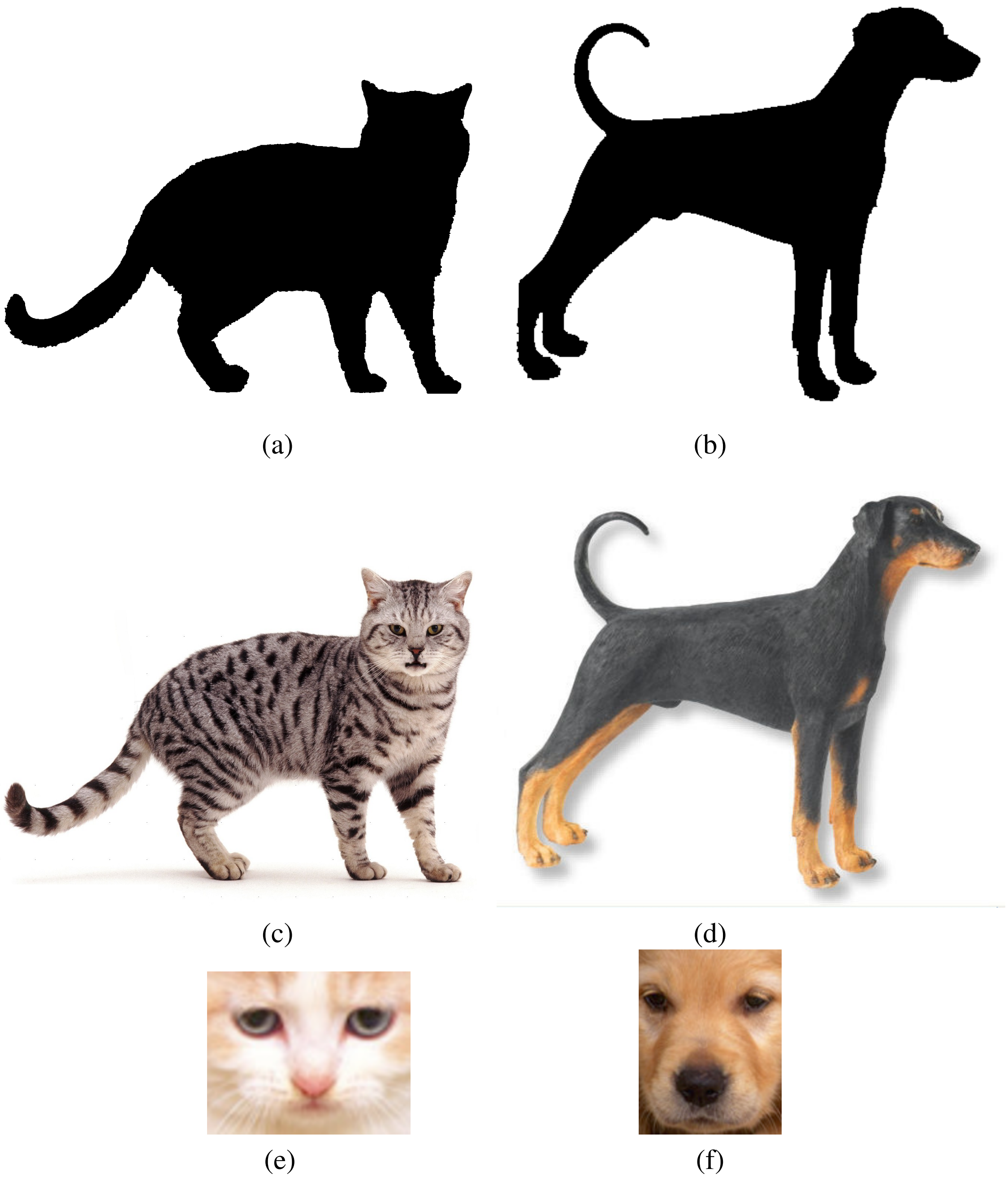
Pablo Picasso, *Bull Analysis,* 1945.

Chapter 3

Medialness Measure via Oriented Boundary

> *"We can only see a short distance ahead, but we can see plenty there that needs to be done."*
>
> *– Alan Mathison Turing, Computer scientist (1912–1954)*

When it comes to visual perception, there are many cues one can use to describe an object, for example with reference to modalities such as color, texture, shape. Which one carries the most information, i.e., which one alone can be sufficient to identify and characterize an object, is a practically important question to ask. Consider the case of a cat-silhouette in Figure 3.1(a). Human's vision system is intelligent enough in perceiving the given silhouette as a cat-build; by just looking at its shape or contour. Let us modify the scenario by introducing a dog-silhouette alongside with this cat-silhouette (see Figure 3.1(b)). Again, we can easily distinguish the two shapes, i.e., dog and cat, by looking at their shapes. Now if we further modify the problem domain of object detection by introducing the natural colored and textured picture of the same cat and dog (Figures 3.1 (c) & (d)), it is still very easy for us to say Figures 3.1 (a) and 3.1 (c) represent the



(a) (b)

(c) (d)

(e) (f)

Figure 3.1: Object comparison through shapes, texture and color.

same cats while Figures 3.1 (b) and 3.1 (d) represent the same dogs. Here, shape plays the noticeable role in identifying similar objects or distinguishing two objects while texture and color have no or little role. The opposite situation can be seen in Figures 3.1 (e) & (f). Shape alone can provide powerful cues to detect and recognise objects, and therefore we decided to start our investigation by exploring the elementary shape features that would also be found in psychological aspects of human vision.

In this chapter we discuss the representation of 2D shapes. Among the different shape representations that exist in the literature, I explore the use of medialness as defined by Kovács et al. (1998), a psychophysically derived mathematical model. The reasons behind such a choice are:

- Medialness had to be studied in an algorithmic form; and hence this gave us an opportunity to investigate the model.

- This psychophysically derived model provides new insights on shape representation.

- The model relates to the most used shape representation in computer vision and computational geometry – medial axis and its closely related Voronoi diagram.

- Medialness informs on both local and global features, allowing us to represent an object via parts.

- Medialness relates topology with geometry, i.e., the representation provides enough information in order to characterise the connectivity between different extracted feature points. In addition, the model permits to acquire the concavities and convexities of an object.

According to Kovács *et al.*, the definition of *medialness* of a point in the image space is given by the accumulation of sets of boundary segments falling into an annulus of thickness parameterised by the tolerance value (ε) and with interior radius taken as the minimum radial distance of a point from boundary (Kovács et al. (1998)), Figure 1.3. The so defined medialness maps an image of the interior of an object to a grey level 2D map, where greyness is a direct measure of the accumulated medialness measure for each point of the original image under consideration. One can think of such a map as a landscape with peaks and pits, ridges and valleys, *i.e.* a 2D set function. A *dominant point* for this medialness map then intuitively corresponds to a well localised peak.

In order to implement this notion of medialness we first need to characterise the D_ε medialness measure. This is a sampling function constrained to annular sectors. Given a minimum radius function $R(p)$ with center locus p, which defines the interior shell of the annulus, and $R_{max}(p) = R(p) + \varepsilon$ the exterior shell, a given sector S_i is specified by an angular opening θ_i with bounding segments defined by the intercepts t_i and t_{i+1}. The area of the sector is then: $A_i = \theta_i \varepsilon (R + 0.5\varepsilon)$. In practice, the intercepts bounding a sector $[t_i, t_{i+1}]$ are given by the extremal points of contour or edge segments entering and exiting the annulus, *i.e.,* crossing either of its interior or exterior shells (Figure 1.3). The medialness measure is then taken as a sampling over the sum of one or more annular sectors containing boundary information (*i.e.* with contour or edge segments); what is actually measured in each sector is application dependent; in our case one can count the amount of boundary information present in each sector, *e.g.,* the number of edge pixels, the contour segment length, a binary counter for fixed-length elementary sub-sectors, bins for fixed sub-tended elementary angular steps $\delta\theta$, or even simply as the area A_i of the sector. We denote the measure taken over an annular sector S_i: $\widehat{S_i}$. Then, we can express the medialness measure first proposed by Kovács *et al.* in a general form as:

$$D_\varepsilon = \sum_{k=0}^{n} \widehat{S_{k+1}}\,, n \geq 1\,. \tag{3.1}$$

This formulation implies that at least one separate annular sectors are traversed by the boundary. Notice that when the tolerance value ε reduces to zero, D_ε reduces to a maximal inscribed disk and leads to the traditional medial axis graph measure.

One noticeable drawback of this definition when seeking to retrieve dominant points is that it does not make a distinction with neighboring boundary segments part of separate object parts and which ought not to be considered in the support annular zones; *e.g.,* this may be the case with the fingers of one's hand when these are kept near each other (Figure 3.2). That is, by introducing a boundary information catchment area via the tolerance annular width ε we loose the implicit property of a maximal disk being inscribed. In Figure 3.2 (top) the medialness mapping has been performed using Kovács et al. (1998) original method where the medialness of each finger also is influenced by neighbouring fingers and result in extra information being added creating a typical "halo"[1] effect in the vicinity of ridges shown in dark grey.

This imprecision in D_ε can be remedied by introducing a measure of orientation at boundary points when assuming figure-ground segmentation *(i.e.,* when knowing the interior versus the exterior of an object); we propose to use such information, *e.g.,* obtained from a classic gradient boundary filter to modify the medialness function, resulting in D^*_ε (derived in the following sub-section). This not only reduces the impact of neighbouring parts on medialness measurements, but also emphasize the sharpness of medialness at the tips of ridges, see Figure 3.2, which proves helpful in practice to locate convex and concave features points as will become clearer later.

On completion of medialness measurements, each pixel in the transformed image space holds a local shape information of accumulated medialness. When assuming figure-ground separation, thickness variations, bulges and necks of an object are captured via *interior* medialness measurement. Kovács et al. (1998) have shown that humans are most sensitive to a small number of localised areas of medialness which coarsely (by visual in-

[1] a wider more diffused impact

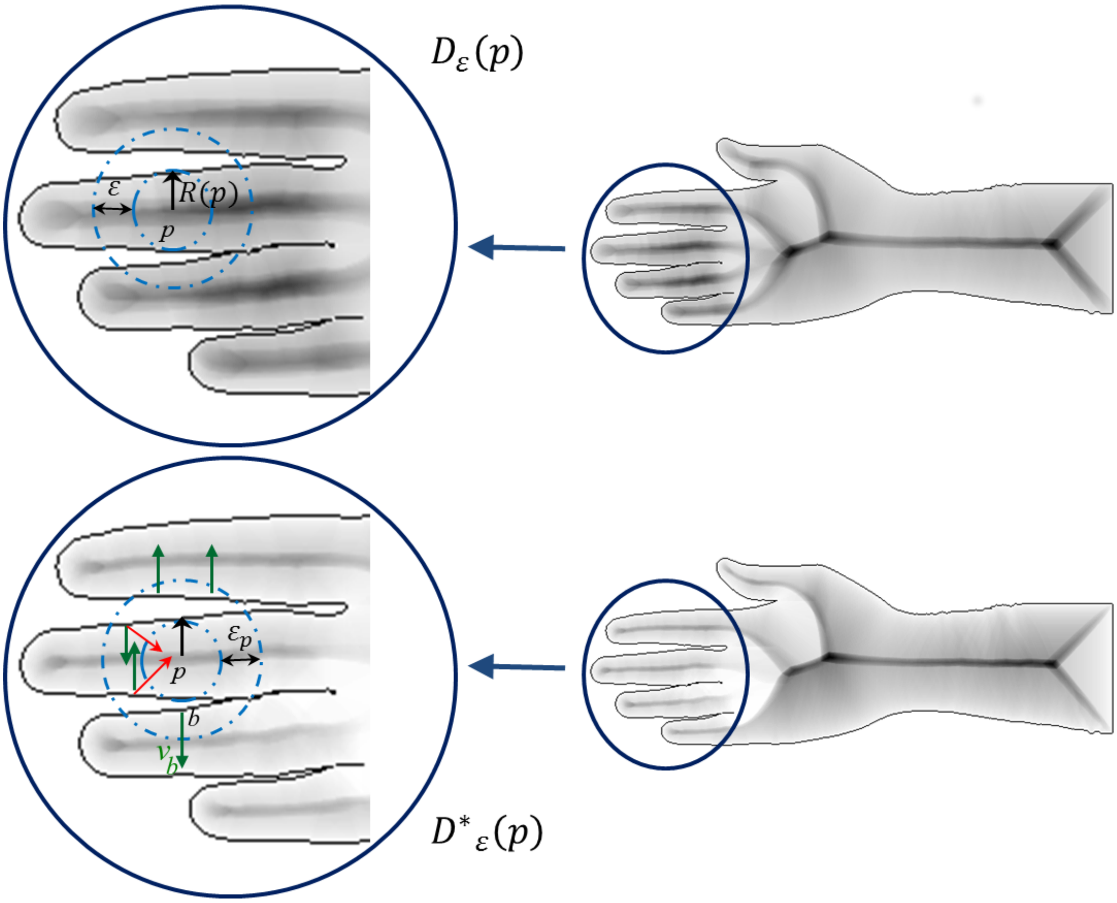


Figure 3.2: Comparison of the results of computing medialness functions on the image of a human hand. *Top*: the result of applying the original D_ε function as proposed by Kovács *et al*. *Bottom*: the result of our proposed method which takes into account orientation at the boundary (D^*_ε). In computing D_ε, the medialness at a point p may take support from neighbouring fingers, while when using the proposed D^*_ε such extraneous information can be discarded by only taking those boundary segments into account that have a positive scalar product of their location vector $(p-b)$ with its boundary vector v_b.

spection) correspond to joints for animated bodies (Kovács et al. (1998); Kovács (2010)). Our extended notion is defined as *dominant points* and can be applied to any object, animated or not. Dominant points are constrained to be a relatively small number of points of high medialness obtained by filtering out the less informative, redundant and noisy data from the initial medialness image space. Note that, I compute medialness *both for the* figure (*interior*) and the ground (*exterior*) as I seek to also characterise concavities of an object.

To identify dominant points a *morphological top-hat transform* (Serra (1983); Leymarie and Levine (1988); Vincent (1993)) is applied to isolate ridges and their peaks in the medialness 2D function. Ridges are filtered using an empirically derived threshold. The selected ridges are then locally characterised by isolated representative points associated to peaks in the ridge data. To avoid considering large numbers of nearby isolated peaks that are characteristic of elongated object parts or parts with many small deformations, only peaks at a given minimum distance away from each other are retained. The extraction process of *external* dominant points is achieved by combining a concavity measure at the tip of medialness ridges together with a length of support on the associated contour segments. Again, a spatially localised filtering is applied to isolate representative dominant points. The set of *convex points* is similarly extracted. The details of the recovery of the three types of selected points is given in Chapter 4.

Together, the three types of selected dominant points are then considered as the *representative feature points* of the shape. Our matching algorithm, detailed in Chapter 5, is designed in such a way that it first compares internal dominant points of a query object with internal representative dominant points of target shapes in a database. External concave points are then similarly processed, and convex points are used in a final refinement step. The matching algorithm first analyses the amount of scale, rotation and translation of the query with respect to the target image. These values are then applied over the query image to find the best possible matching location in the target image. We next provide the details of the various stages of computation.

3.1 Medialness for Oriented Boundary Segments

We define the computation of 2D medialness, modifying Kovács et al. (1998) original definition, by introducing orientation to each considered object boundary points. In a continuous curve, orientation of a boundary point b is given by a vector v_b, which is the direction of a normal at this point b with respect to a positively oriented horizontal X-axis (see Figure 3.3).

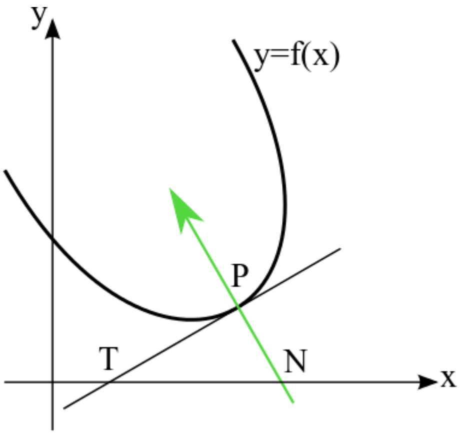


Figure 3.3: For the curve $y = f(x)$, tangent at point P is the line joining points P & T. The orientation of point with respect to this curve is the direction of the normal: a perpendicular line on tangent PT passing through point P, i.e., line $\overrightarrow{NP}$.

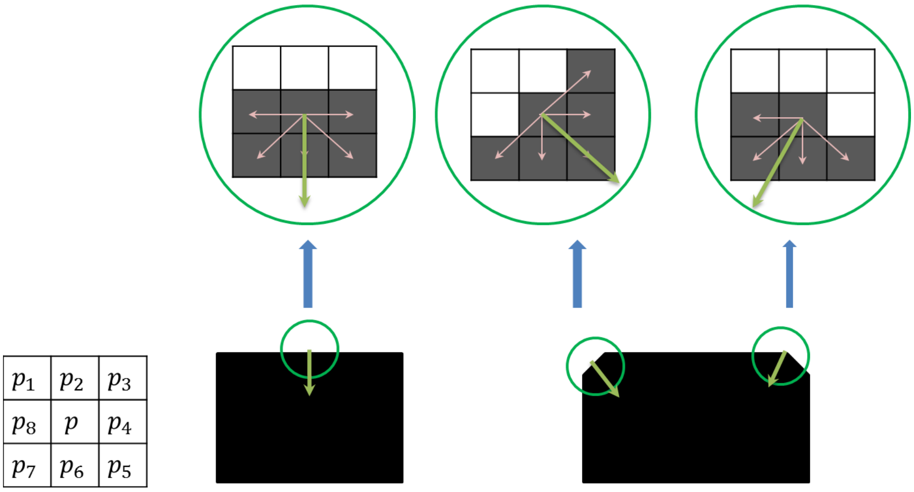
In the case of a digitized curve, orientation of a boundary pixel b can be evaluated by either calculating its local gradient or calculating a vector sum for the foreground pixels (here it is black) in the specified window size $n \times n$ centered at this point b. For example, a window of size 3×3 centered at boundary-pixel b creates a set of 8-neighbor pixels $\{p_1, p_2, \ldots, p_8\}$ and a set of vectors $\{v_1, v_2, \ldots, v_8\}$ joining b and its neighbors, i.e., $\{(p_1 - b),$

$(p_2 - b)$, ... , $(p_8 - b)\}$ (Figure 3.4), then v_b is defined as (the vector sum):

$$v_b = \sum_{i=1}^{8} v_i \cdot \delta_i = \sum_{i=1}^{8} (p_i - b) \cdot \delta_i \tag{3.2}$$

where

$$\delta_i = \begin{cases} 1 & \text{foreground pixel} \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$



p_1	p_2	p_3
p_8	p	p_4
p_7	p_6	p_5

(a) (b)

Figure 3.4: Examples of the orientation assessment of candidate boundary points. (a) 8-connected neighbor-points of p, and (b) three different examples of boundary points: horizontally straight, right diagonal and at a corner of the object. The orientation of a boundary-point b is illustrated by a green arrow; in practice this can be approximated by the summation from other vectors in its 8-direct neighborhood linking to object's foreground pixels (shown as pink arrows).

In general, in a digital curve, the orientation of a boundary point b with the window-

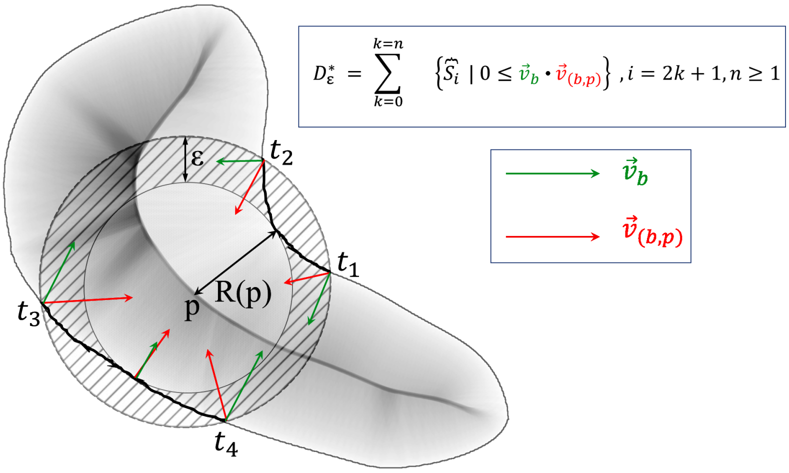

$D_{\varepsilon}^{*} = \sum_{k=0}^{k=n} \{\overbrace{S}_i \mid 0 \leq \vec{v}_b \cdot \vec{v}_{(b,p)}\}, i = 2k+1, n \geq 1$
$\vec{v}_b$
$\vec{v}_{(b,p)}$
ε
t_2
t_1
t_3
t_4
p
R(p)

Figure 3.5: Illustration of the D_{ε}^{*} function for a simple shape, defined as an accumulation of boundary segments falling inside an annulus neighborhood of thickness ε (shown as darker boundary segments within the annulus' ring) centered around a position p, and such that the associated orientation vector $\overrightarrow{v}$ has a dot product with the unit radius vector (taken in the direction of the line from $b(t)$ to p) which is positive. $R(p)$ is taken as the minimum radial distance from p to the nearest contour point.

size $n \times n$ is calculated as:

$$v_b = \sum_{i=1}^{(n^2-1)} v_i \cdot \delta_i = \sum_{i=1}^{(n^2-1)} (p_i - b) \cdot \delta_i \tag{3.4}$$

The medialness gauge at a point p is defined similarly to D_{ε} by adding an orientation constraint such that only those boundary loci which are pointing inward (with respect to the figure) are considered for evaluation (Figure 3.5):

$$D_{\varepsilon}^{+} = \sum_{k} \widehat{S_{k+1}} \cdot \delta_{v_b \cdot (p-b)} \tag{3.5}$$

$$D_{\varepsilon}^{-} = \sum_{k} \widehat{S_{k+1}} \cdot \delta_{-v_b \cdot (p-b)} \tag{3.6}$$

where

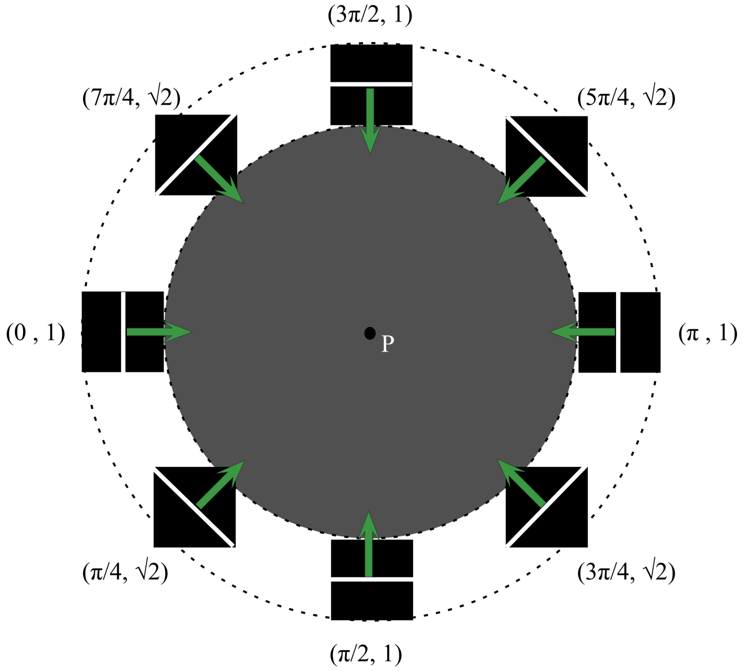


Figure 3.6: Weight assignment to the boundary pixels based on their orientations. The value is periodic with an interval of $\pi/2$.

$$\delta_x = \begin{cases} 1 & \text{if x} \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.7}$$

for a point $p = (x_p, y_p)$, vector $b(t) = (x(t), y(t))$ describing (pixel) loci along a piece of 2D bounding contour (B) of the object, and such that v_b is the orientation of the boundary point $b(t)$, $\overrightarrow{v_{(b,p)}}$ is the orientation of the line joining $b(t)$ to p. The positiveness (case of internal medialness, i.e., D_ε^+) or negativeness (case of external medialness, i.e., D_ε^-) of the scalar product $v_b{\cdot}(p-b)$ is used to rule out boundary pixels which are oriented away from the given annulus center. We do not consider the geometry (differential continuity) of a contour other than provided by that gradient orientation. NB: this criterion is efficient if we have reliable figure-ground information. This is a limit of the modified gauge D_ε^*; however we can always fall back on the original gauge D_ε if object segmentation is not reliable, i.e., no interior versus exterior can be pre-identified.

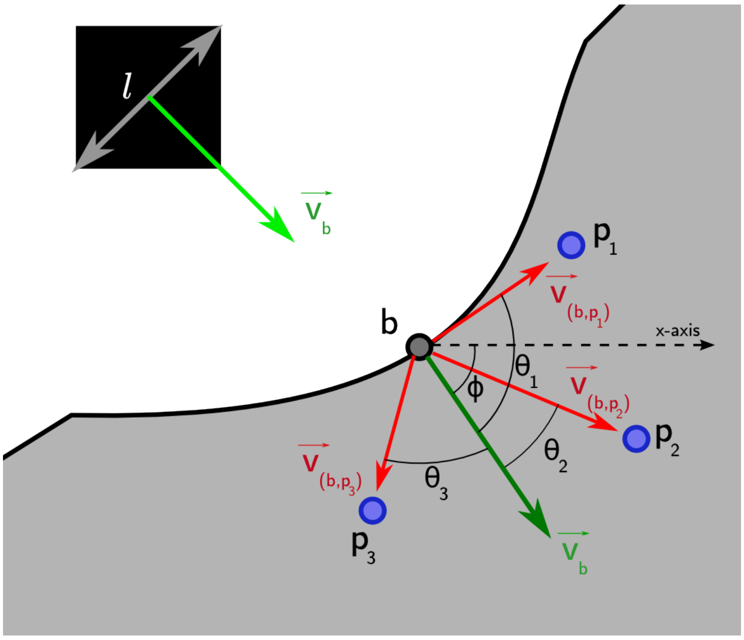


Figure 3.7: Contribution of an oriented boundary point b to different points p_1, p_2, and p_3 in the image space.

One problem in the digitized image is evaluating the length of a curve segment. Counting pixel is wrong as a digital diagonal in a square is made of as many pixels as each side but 41% longer. Curve fitting is another approach but again, this produces erroneous result at the end.[2] Here we define a method that assigns weight to each boundary pixel in the range of $[1, \sqrt{2}]$. From a given frame of reference, a boundary pixel weight is 1 when it is either horizontally or vertically aligned and is $\sqrt{2}$, when it is aligned diagonally (i.e. slanted at 45 degree), otherwise it will take a weight in between these two values (see Figure 3.6). If a boundary pixel has an orientation ϕ (see Figure 3.7), its weight is the value of projection of the square-pixel in the direction of ϕ. Mathematically, this is calculated

[2] Curve fitting is the process of constructing a curve, or mathematical function, that has the best fit to a series of data points, possibly subject to constraints. Curve fitting can involve either interpolation, where an exact fit to the data is required, or smoothing, in which a "smooth" function is constructed that approximately fits the data. For the discrete set of data-points, interpolation is used, while for the continuous set, approximate smoothing function is applied. If approximation smoothing leads to under-fitting/over-fitting or leaves out those data which are not fitting on the curve then there is loss of information.

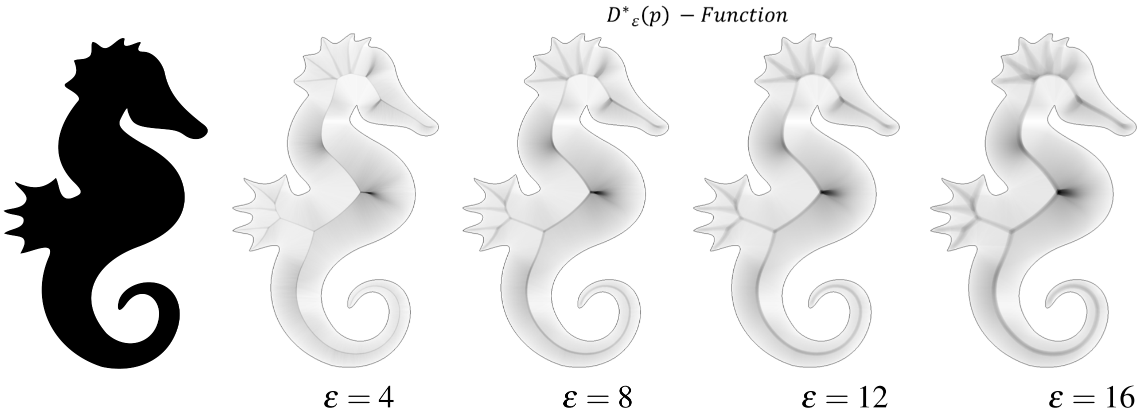


Figure 3.8: Medialness and tolerance — *Left*: silhouette of sea-horse (image size: 630 × 1010); other images to the right show variations in D^*_ε for increasing tolerance values (ε, in pixel units).

as:

$$l = \frac{1}{\max(|\sin(\phi)|, |\cos(\phi)|)} \tag{3.8}$$

The final contribution ∂b of a boundary point b to $\widehat{S_i}$, with orientation ϕ and angular distance θ (Figure 3.7), is calculated as:

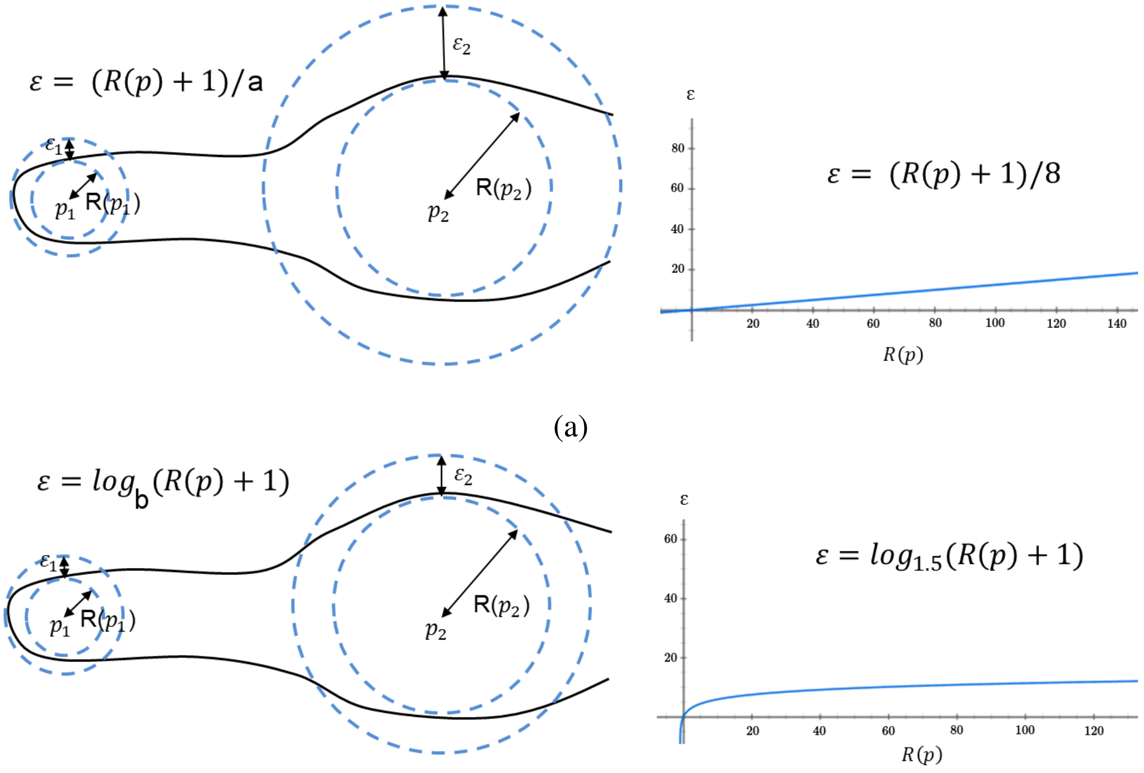
$$\partial b = \begin{cases} l\cos\theta, & \text{if } -\pi/2 \leq \theta \leq \pi/2 \\ 0, & \text{otherwise} \end{cases} \tag{3.9}$$

The metric $R(p)$, the minimum radial distance to the interior annular shell, is taken as the smallest available distance between p and a bounding contour element:

$$R(p) = \min_t \left\{ |p - b(t)| \;\middle|\; 0 \leq \overrightarrow{v_b} \cdot \overrightarrow{v_{(b,p)}} \right\} \tag{3.10}$$

In Figure 3.8 the medialness measurement is performed on a sea-horse profile illustrating the evolution in interior medialness measures when varying the value of tolerance (ε), which reflects a smoothing effect: as the tolerance increases, medial symmetries with smaller boundary support are averaged out in favor of larger scale ones — their associated ridges become more dominant. The tolerance value (ε) is currently set as the elementary pixel size (and so is related to the resolution used).

3.2 Adaptive Tolerance $\boldsymbol{\varepsilon}$ as a Function of the Minimum Radial Distance $\boldsymbol{R(p)}$



(a)

(b)

Figure 3.9: Tolerance ε defined as two different functions of the minimum radial distance $R(p)$. (a) Linear: variation of ε with respect to $R(p)$ at a typical value of slope $1/8$; and (b) Logarithmic function: variation of ε with respect to $R(p)$ value of base of 1.5.

The medialness measure of a point p varies with the two parameters: $R(p)$ and ε, where $R(p)$ is the minimum radial distance between p and the bounding contour, and ε is the width of the annulus region (capturing object trace or boundary information). Any boundary point b falling inside this annulus that satisfies the definition of equation 3.5 (for interior medialness) or equation 3.6 (for exterior medialness) is added in support for medialness at p. We can think of the width of the annulus as dictated by ε as an equivalent to a scaling parameter: the larger the width, the more averaging of nearby contour information is considered. How to set the tolerance ε in order to have desirable scaling properties thus needs to be addressed.

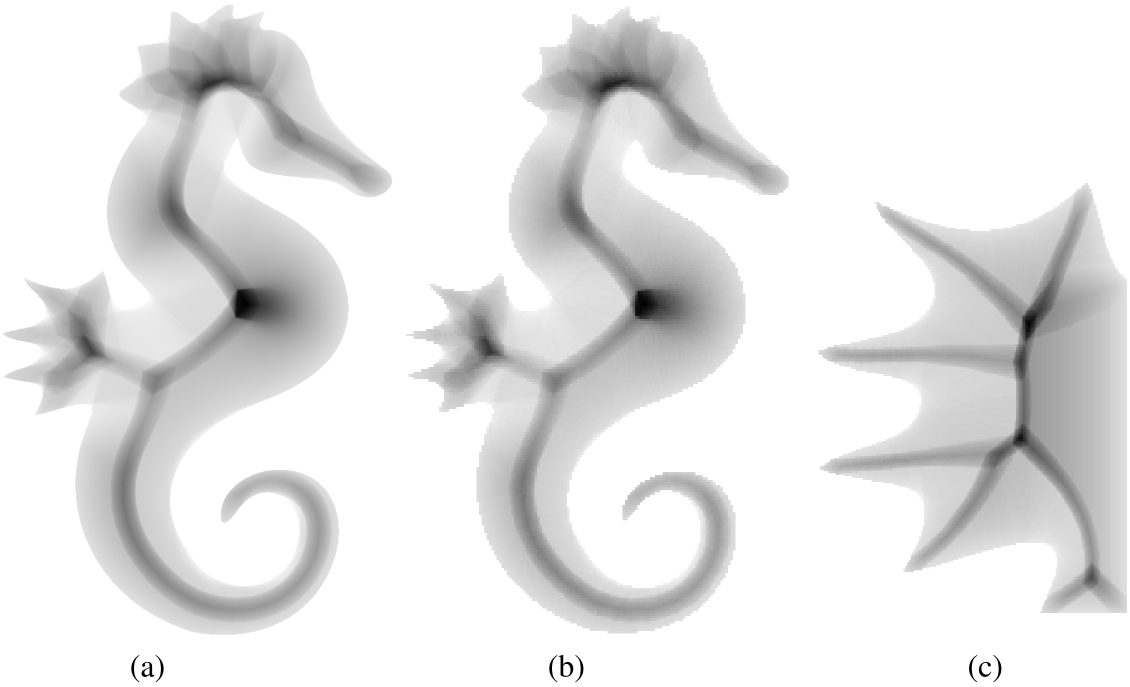


Figure 3.10: Medialness measures, D_ε(Kovács et al. (1998) method), of (a) full seahorse at scale 1, (b) full seahorse at scale 0.25, i.e., uniformly scaled-down to 25%, and (c) mid-left part of the seahorse at scale 1. Medialness measure is affected by uniform scaling effect, as the amount of boundary points is scaled proportionally, and this explains why medialness measures of seahorses in (a) and (b) are same. On the other hand, when there is a cut or occlusion or deformation in the shape, D_ε produces very different medialness results. Such case can be seen by comparing (a) and (b) with (c), where (c) is a part of (a). In these images, ε is taken as 1% of the total boundary length, as suggested by Kovács et al. (1998).

In the literature [Kovács et al. (1998); van Tonder and Ejima (2003)], ε is fixed for a given image, which is a function of the total length of boundary (e.g. x% of the boundary length). The method works well and produces similar medialness field in those cases where the same image is scaled up or down proportionally. But the medialness field changes for the same object if there is a cut in the shape or another object is introduced in the same image (Figure 3.10). Consequently, this changes the total boundary length and hence the value of ε.

I first noticed that as the minimum radius value $R(p)$ augments, it is useful to augment the corresponding value of ε so as to filter away noisy variations in what is an increasing part of a lower curvature (flatter) set of boundary segments. On the other hand, for smaller values of $R(p)$ we should use a sharper, smaller width ε in order to more accurately capture the augmenting local boundary curvature variations. Furthermore, when considering the tip of sharp fine elongated limbs, *e.g.* the end of fingers, a decreasing value of ε is also beneficial in practice to avoid blurring the corresponding centre of curvature locus. I have therefore studied various ways to set automatically an augmenting value of the tolerance ε as the minimum radius value $R(p)$ is increased. The obvious first choice is to use a simple linear relation, such as:

$$\varepsilon_p = \kappa \left(\frac{a \cdot R(p)}{\kappa} + 1 \right), 0 < a \leq 1 \tag{3.11}$$

where κ is the smallest measurable length in the given image.[3] But it has been observed in practice that this leads to rapidly increasing averaging effects which generate overly thick ridge traces or generate ridge doubling (alike the halo effect when using D_ε). To overcome this effect, I select ε as a logarithmic function of $R(p)$ (Figure 3.11):

$$\varepsilon_p = \kappa \log_b \left(\frac{R(p)}{\kappa} + 1 \right), b > 1 \tag{3.12}$$

Having selected ε to be adaptive as a logarithmic function of $R(p)$, the next step is

[3] κ is the smallest measurable length in an image, that is directly proportional to the sampling rate. The sampling rate (or pixel clock) of the digitiser determines the spatial resolution of the digitized image.

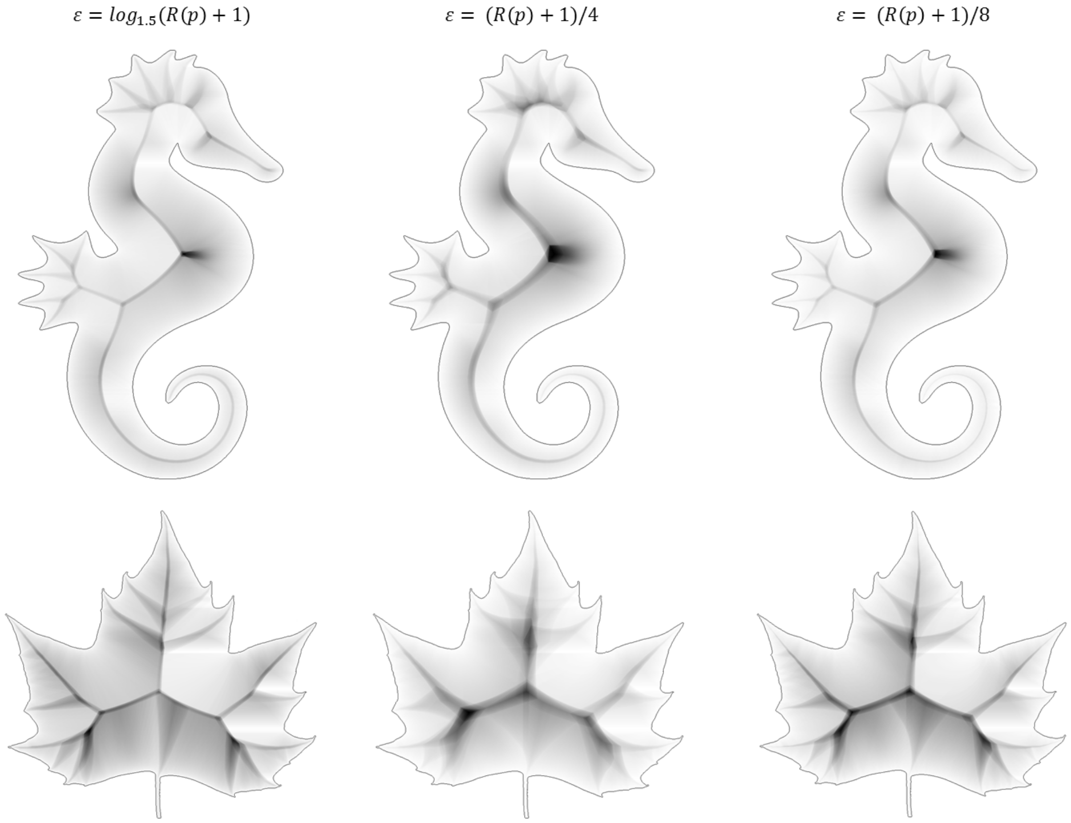

$\varepsilon = log_{1.5}(R(p) + 1)$
$\varepsilon = (R(p) + 1)/4$
$\varepsilon = (R(p) + 1)/8$

Figure 3.11: Comparing (interior) medialness gauge maps for different adapting functions to set the tolerance ε. On the left are the results of setting ε as a logarithmic function of $R(p)$, while to the right ε varies linearly with $R(p)$; notice less sharpness in the medialness ridges and tips near convex corners when using linear functions.

to define a useful logarithmic *base*. When such a base has a large value it produces very sharp medialness for large radii values response but can emphasise noisy responses. This can be intuitively understood since a relatively large base makes the medialness measure become more and more like a traditional medial axis transform as the relative width of the annulus region becomes smaller and smaller in comparison to $R(p)$ for increasing radii. Setting the logarithmic base to a relatively low value (*i.e.* nearer unity) gives smoother responses, which can augment considerably the halo effect as with a linear function (Figure 3.16, first two rows). By empirical investigation we have observed that a good compromise can be obtained for bases with values in the range of $[e/2, 3e/4]$. For our reported

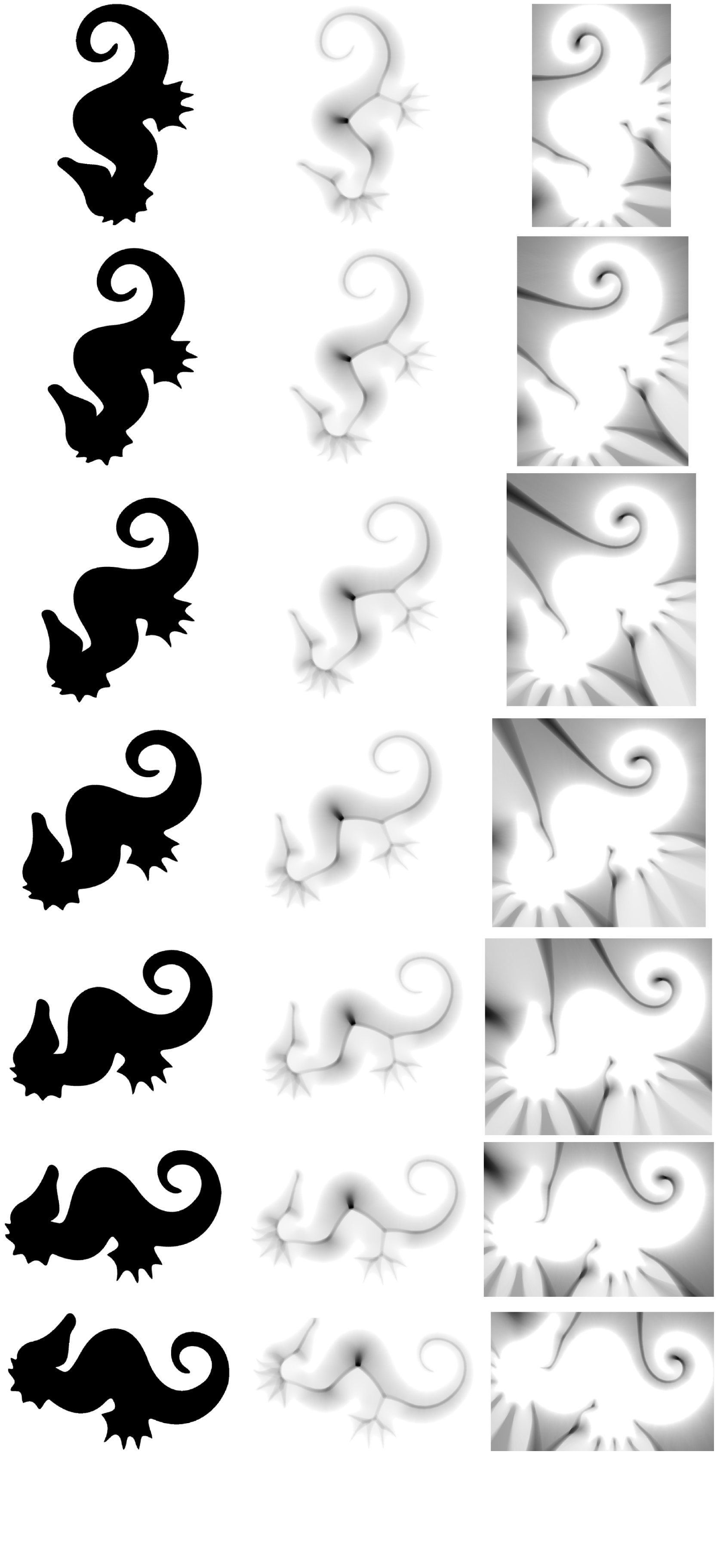


Figure 3.12: Rotation invariance property of medialness. Top-row: Silhouette of seahorses rotated sequentially by $\pi/12$. Middle-row: respective interior medialnesses. Bottom-row: respective exterior medialnesses.

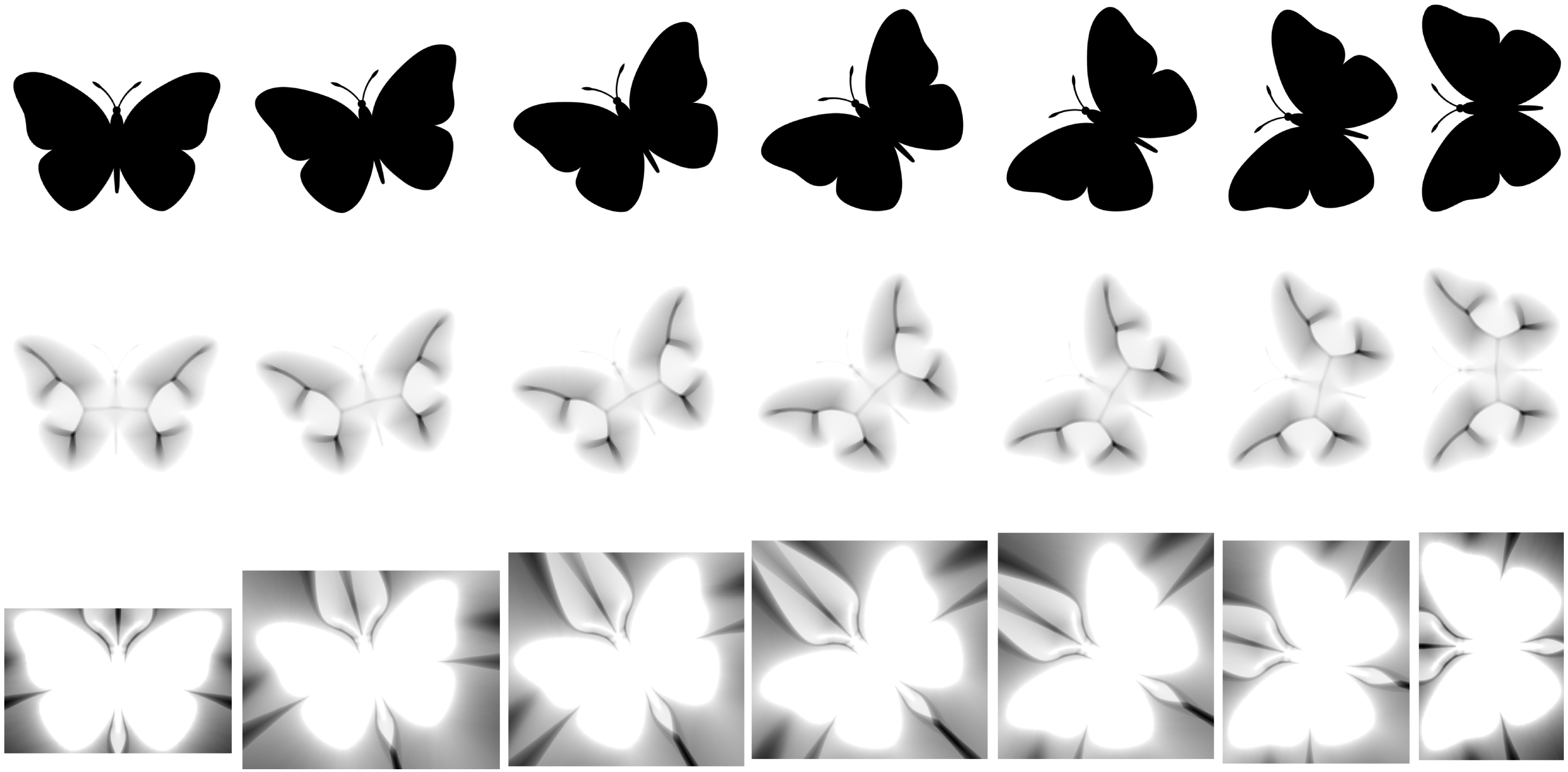


Figure 3.13: Rotation invariance property of medialness. Top-row: Silhouette of butterflies rotated sequentially by $\pi/12$. Middle-row: respective interior medialnesses. Bottom-row: respective exterior medialnesses.

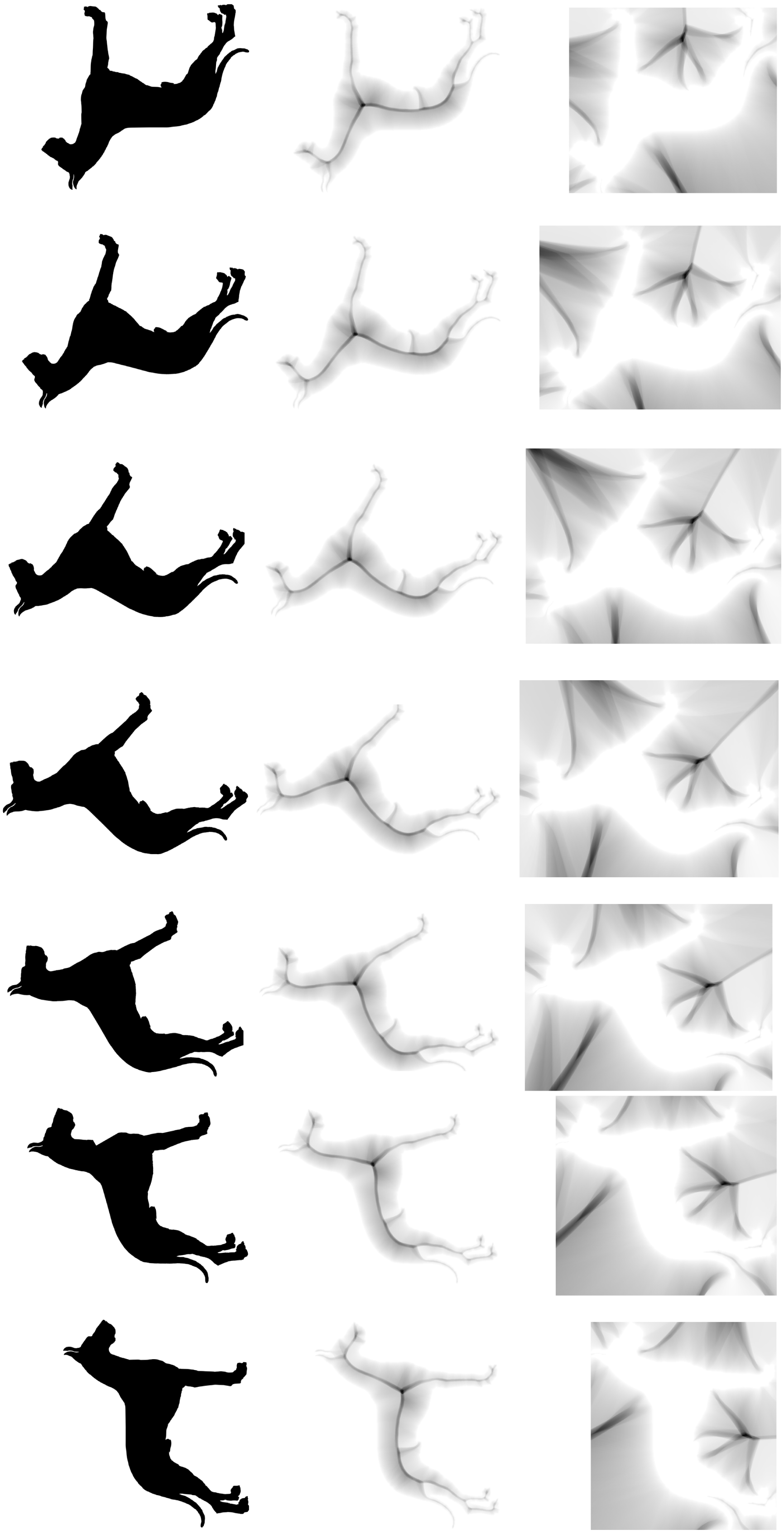


Figure 3.14: Rotation invariance property of medialness. Top-row: Silhouette of dogs rotated sequentially by $\pi/12$. Middle-row: respective interior medialnesses. Bottom-row: respective exterior medialnesses.

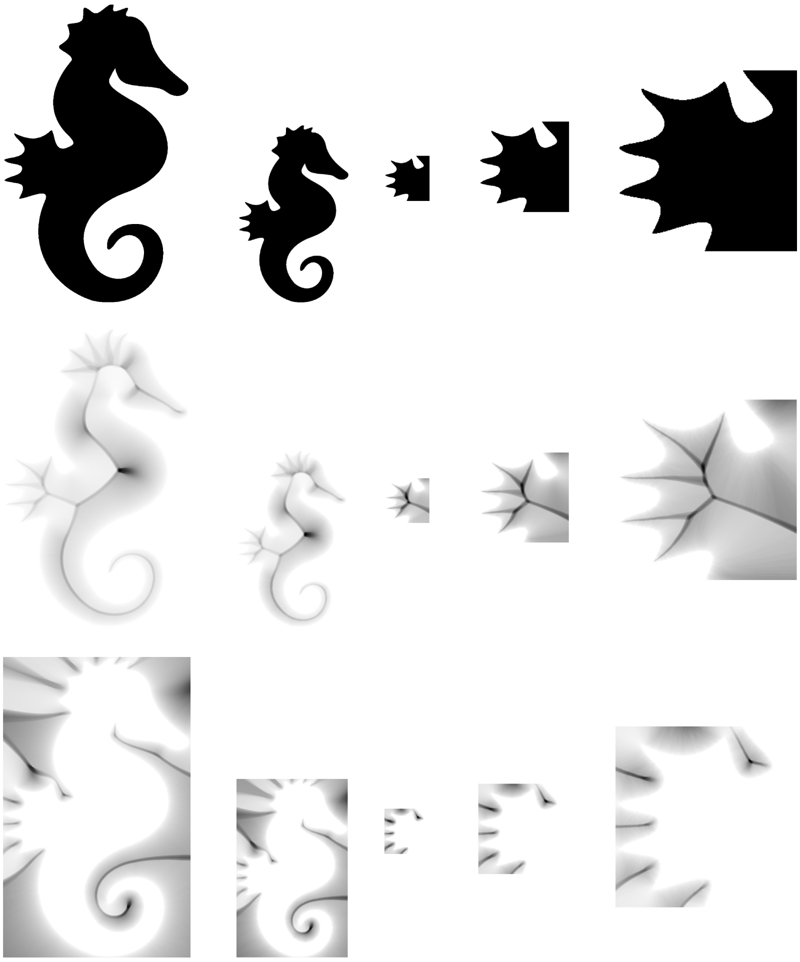


Figure 3.15: Medialness under scale-invariance. Medialness also remains invariant when there is a cut or occlusion or deformation in the shape, D_{ε}^{*} produces same medialness results.

results we use a logarithmic base of $e/2$, where e is the Euler's number (approximately 2.718281828). By visual inspection, we observe that the main medialness features remain present under varying rotations (Figure 3.12, 3.13, & 3.14), scaling and cut (Figure 3.15). These features include the main ridges necessary to retrieve hot spots and end of ridges mapping to significant convex and concave loci.

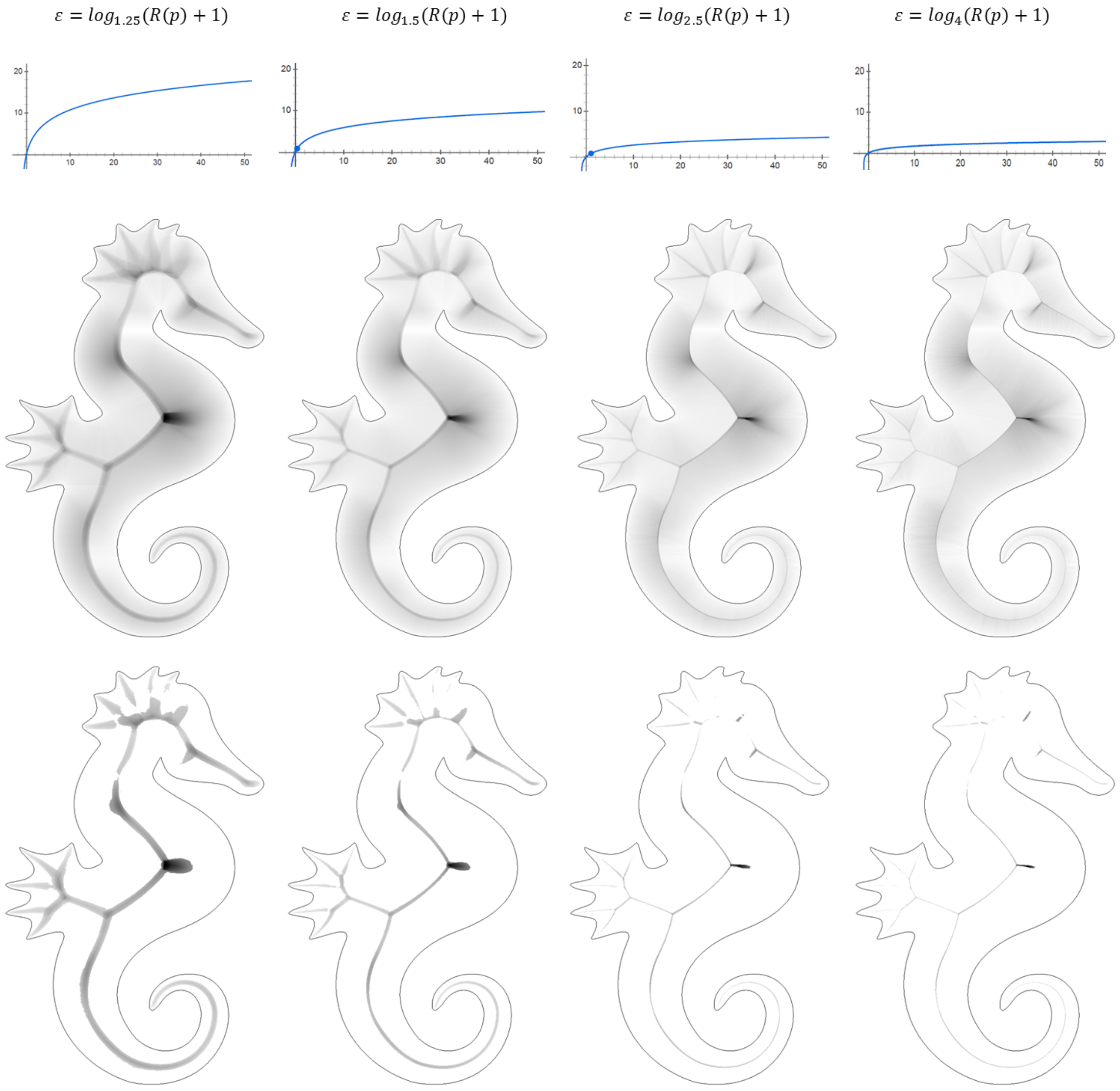


Figure 3.16: Comparing internal medialness measure different adaptive tolerance levels ε set as logarithmic functions of $R(p)$. We illustrate from left to right how augmenting the logarithmic base impacts the resulting medialness map (2nd row) and associated ridge traces retrieved by applying a morphological hat transform (3rd bottom row). Note that for a base nearer unity, the medialness response suffers more from halo effects making dominant points more difficult to localise, while higher values of the base augments sharpness of the results but also emphasise noisier, less significant (with small support) boundary features.

3.3 Summary

In this chapter we presented a psychophysically inspired mathematical model — *medialness*, which provides a description for 2D shape. Different from state-of-the-art approaches, we moved in the direction of a psychological aspect of human vision. We augmented the Kovács et al. (1998) medialness definition by introducing orientation to the boundary points and then used this information to modify the medialness function accordingly. In contrast to the work of Kovács et al. (1998), we applied this model on both internal and external regions, and as a result we obtained the medialness field for the given shape. The resulting field contains one or multiple prominent thick ridges, subsequently isolated by morphological top-hat transformation. In the case of digital images, we also considered the pixel under-counting and/or over-counting to evaluate the length of boundary segments.

In other methods (Kelly and Levine (1995a); Kovács et al. (1998); van Tonder and Ejima (2003)), adaptive tolerance ε is either kept constant or some percentage of the length of the boundary. We argued through examples that this value must be adaptive to the changes in the size of a shape. Our empirical observations showed that ε should change with the minimum radial distance $R(p)$ logarithmically, not linearly. We also showed how the value of this logarithmic base should fall in a certain range, in order to obtain a good medialness result.

At the end of this chapter, we obtain the medialness field and the isolated ridge. The following questions arise at this point:

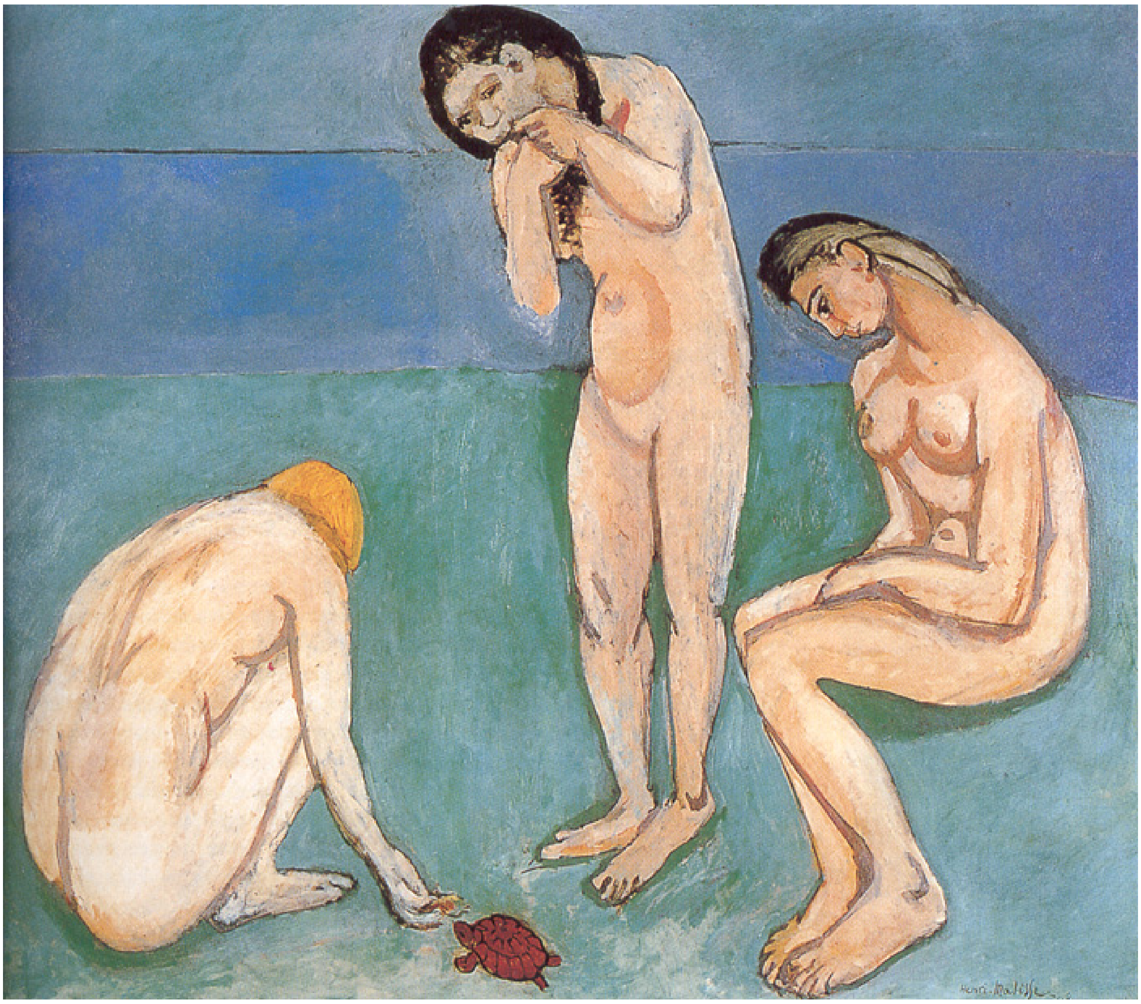
1. The medialness field or ridge contains overwhelming and redundant information. Processing of all of these points is an extremely expensive task in terms of both time and space. The proposal by Kovács et al. (1998) indicates that there are few number of points, they call it “hot-spot”, which contain maximal information about the shape. Within our extracted ridge, how such hot-spots can be extracted?

2. Once those hot-spots have been identified (also called *dominant points*) how can they be intelligible to a machine? In other words, can we represent these dominant points in form of feature vectors? If yes, in what form and how?

3. A frequent problem in shape representation and identification is to make the method Affine invariant. Can we address this issue as well?

4. If we are able to represent a 2D shape with a number of feature points, how can we identify the homography amongst the two shapes?

5. Can we address the scalability issue as well? If yes, how?



Henry Matisse, *Bathers With A Turtle*, 1908. @City Art Museum, St. Louis.

Chapter 4

Shape Invariant Feature Extraction and Description

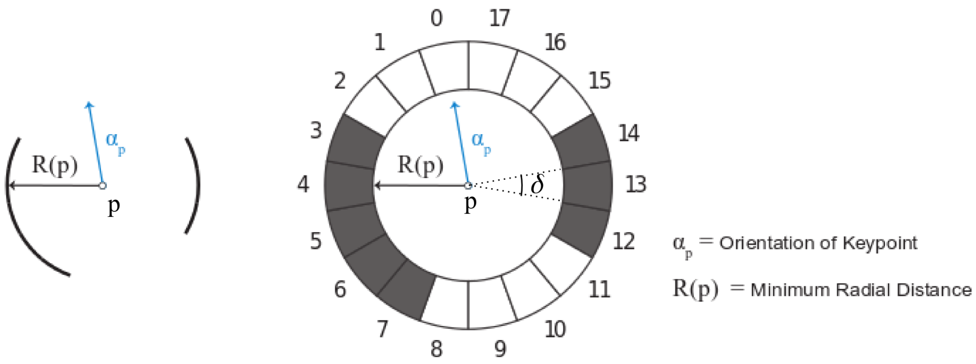
> *"There's no sense in being precise when you don't even know what you're talking about."*
>
> *– John von Neumann, Mathematician (1903–1957)*

This chapter addresses the feature extraction technique and how to represent in descriptive and understandable format, the extracted feature points corresponding to visual cues. I began by an overview of the ShIFT descriptor, followed by the details of the dominant points recovery. The process of converting the set of dominant points into vectors is then explained. This conversion process starts by subdividing the annulus sector into 2π bins followed by erroneous hole filling, which is generally caused by inaccurate contour extraction. Next step is to find the correct orientations of feature points. Depending on the type and configuration of boundary segments contained by the dominant points, a dominant point can have one or more than one orientations. Starting from each evaluated orientation, the set of 2π bins is then converted into n buckets, depending on the value of selected step-angle. Here, my aim is to describe each bucket as a notion of boundary segment (1-valued) or non-boundary segment (0-valued), therefore I quantize the ShIFT descriptor into a binary representation. I then show that the ShIFT descriptor holds the affine invariance property intrinsically. I also comment on its behavior under viewpoint changes and when the object gets inverted. Internal dominant points are extracted by isolating medialness ridges via morphological operations and applying minimum distance parameter. Concave and convex points can be extracted by two different methods: (a) contour following, and (b) using the ShIFT vector. For a given database, the ShIFT vectors obtained from each image are indexed using a B+ tree.

The ShIFT vector holds the same definition for all the three dominant points, i.e., internal hot-spots, external concave points, and internal convex points. Chapter 5 details how to use these for indexing and retrieval purposes. Internal dominant points (hot-spots) are derived from the model of medialness in perception proposed by Kovács et al. (Kovács et al. (1998); Kovács (2010)). Such hot-spots represent the interior structure of the shape and remains invariant with affine changes and articulated movements; however, they don't provide any explicit part description and external structure of the shape. Codons (Richards and Hoffman (1985)), on the other hand, provide an explicit definition of parts by identifying concave (minima) and convex (maxima) points along a closed contour. We adopted this concept to our medialness field to identify significant concave and convex points from a contour. Together all three types of dominant points provide a powerful representation and remain invariant to affine changes and articulations.

In my work, medialness measurement is done separately for internal and external regions and takes advantage of the perceptual figure-ground dichotomy known to be a powerful perceptual cue in humans (Arnheim (1974); Layton et al. (2014)). This also enables our method to consider more easily articulated objects as potential targets in pattern recognition tasks. On the other hand it requires that some good image segmentation is available prior to initiating medialness computations.[1] Under these constraints I can now

[1] In the recent cognitive science literature, arguments are presented to support the idea that medialness


R(p)
α_p
p
0
17
1
16
2
15
3
14
4
13
5
12
6
11
7
10
8
9
δ
α_p = Orientation of Keypoint
R(p) = Minimum Radial Distance

Bin No.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
	□	□	□	■	■	■	■	■	□	□	□	□	■	■	■	□	□	□
Binary Code	0	0	0	1	1	1	1	1	0	0	0	0	1	1	1	0	0	0

Feature Vector <0,0,0,1,1,1,1,1,0,0,0,0,1,1,1,0,0,0>

Figure 4.1: *Top-Left:* Example where two segments (here circular arcs) of various lengths are in medial symmetry (with minimum radial distance $R(p)$ and orientation α_p). *Top-Middle-Right:* Division of the annulus in $2\pi/\delta$ (here 18) bins and filling of the bins where pixels correspond to the presence of boundary segments. *Bottom:* Quantization process leading to the ShIFT feature vector for this configuration.

detail how feature points can be retrieved from the medialness map D^*_ε with logarithmic base setting of tolerance ε. Before going in the details of the dominant point extraction process, first I introduce the Shape Invariant Feature Transform (**ShIFT**) feature vector or descriptor[2] that will be used to obtain a shape-type associated with dominant points (or keypoints).

4.1 ShIFT Feature Vector

Let us consider an extracted candidate feature p from an arbitrary 2D segmented object, with parameters: $\{(x,y), R(p), \varepsilon_p, \alpha_p, D^*_\varepsilon(p)\}$ — *i.e.* for an annulus centered at $p = (x,y)$

can itself be the basis of figure-ground segregation (Layton et al. (2014)).

[2] In this thesis, ShIFT feature vector and ShIFT descriptor have the same meaning and they are used interchangeably.

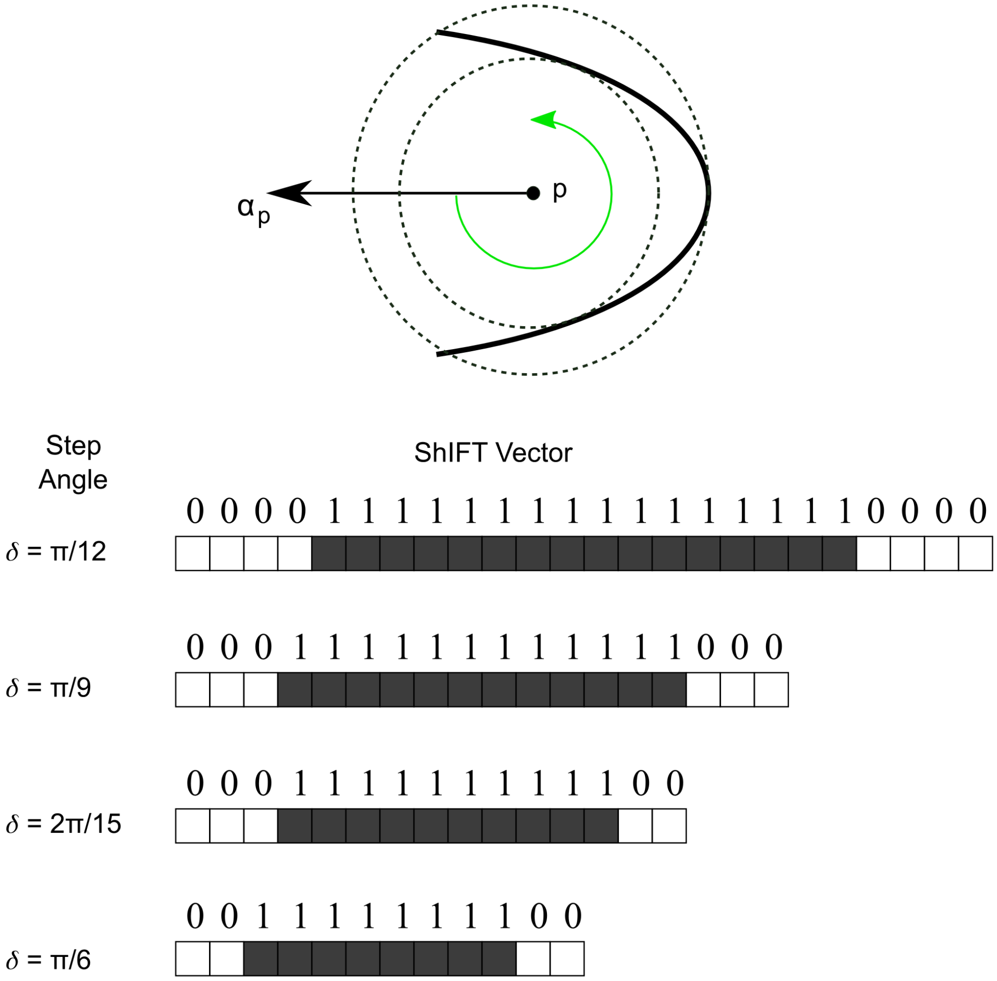


Figure 4.2: The length of ShIFT descriptor is controlled by the step-angle δ. *Top*: A boundary segment is pitching inside the annulus centered at p. *Bottom*: Depending on the value of δ, the quantization process leads to the ShIFT descriptor of relative length. A lower value of δ produces large number of bins and vice-verse higher δ results into a lesser number of bins.

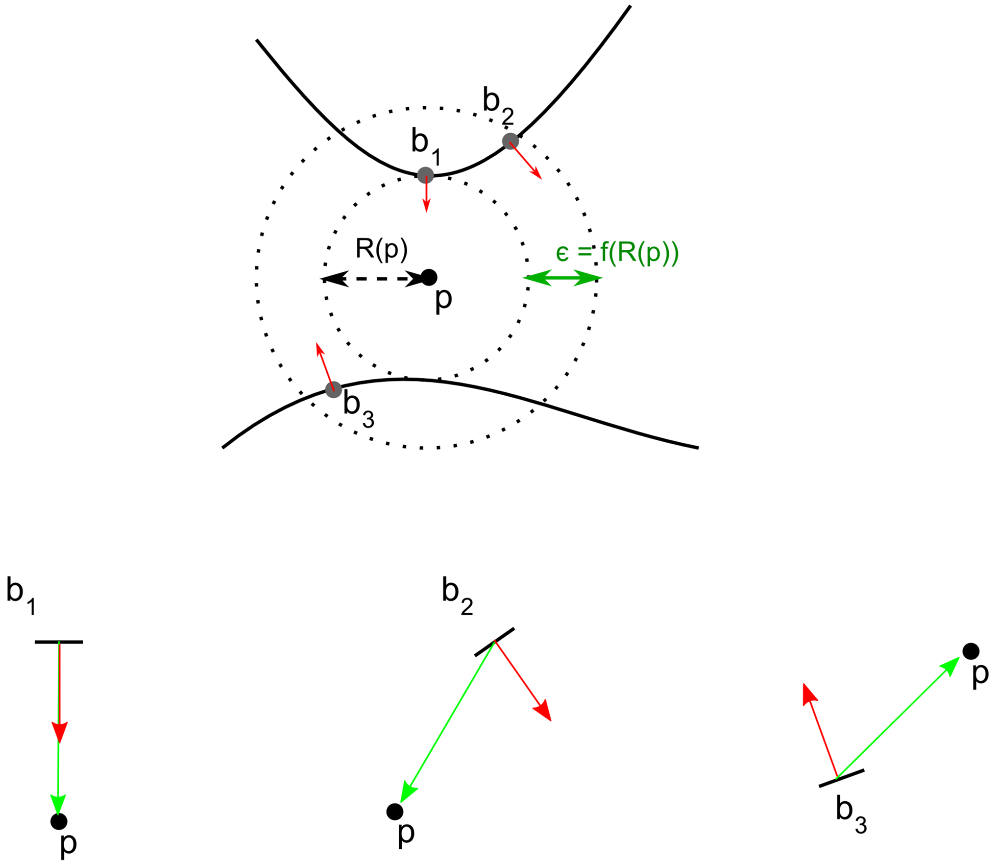


Figure 4.3: Three boundary points b_1, b_2, and b_3 are sampled from the given contour segment illustrating three different possible situations, which lie inside the annulus and provide different support values for medialness calculation at p. Arrows in red color indicate their orientations with respect to the +x-axis, while green arrows in bottom row show their orientation with respect to p. Considering equation 3.5, b_1 contributes the most while b_2, and b_3 have less impact.

with inner radius $R(p)$, thickness ε_p, orientation with respect to +x-axis α_p $(0 \leq \alpha_p \leq 2\pi)$, and total medialness value $D^*_\varepsilon(p)$. The annulus band is then *subdivided* into $2\pi/\delta$ empty bins, where δ is the step-angle parameter defined as $n\delta = 2\pi$ with $n \in \mathbb{N}^+$(Figure 4.1). A bin of the descriptor is then populated by integrating the l values (equation 3.8) of boundary segments falling inside the corresponding sector of the annulus. Following equation 3.5, an elementary part of a boundary segment tends to provide maximum support value to the respective bin when it is oriented towards the feature point p. The support value is lower when the boundary's elementary part is less oriented towards p (Figure 4.3).

4.1.1 Filling of Erroneous Holes

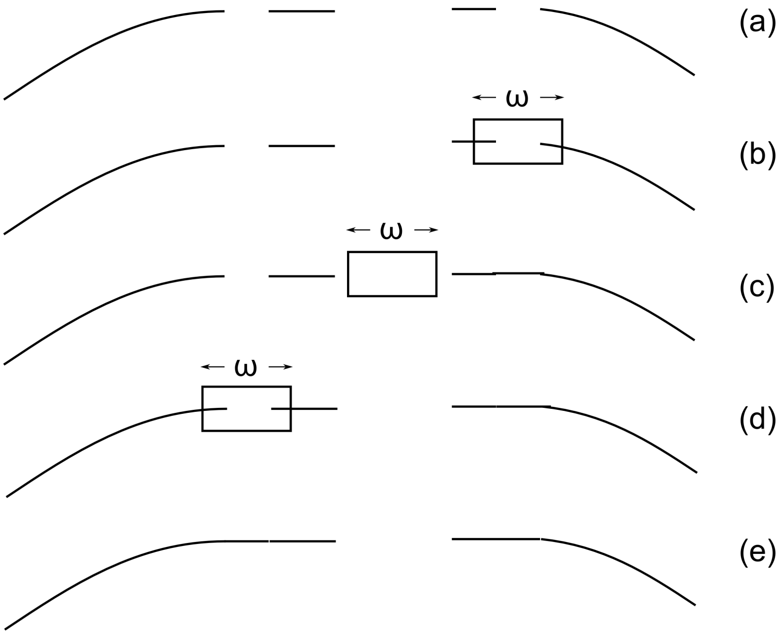


Figure 4.4: Sliding window protocol: (a) An input boundary image. (b) and (d) A sliding window of size ω covering holes between two boundary segments, thus the holes are replaced by boundary segments in next steps. (c) The sliding window is unable to fit properly for hole filling, hence the area is not replaced. (e) Final resultant image.

Sometimes inaccurate contour extraction followed by binarization leads to erroneous pores or holes in the resulting boundary segments, i.e., some of the segments of the resulting boundary are represented as background color rather than foreground or boundary color. This can lead to a false descriptor and hence can affect further calculations. This situation can be handled via a **sliding-window** protocol[3], which replaces the erroneous boundary part with boundary infilling segments (see Figure 4.4). The process is recursive and takes a sliding window of width ω, and can be originated from any direction.[4] Since the hole filling process can alter a keypoint orientations, it must be applied prior to orientation assessment.

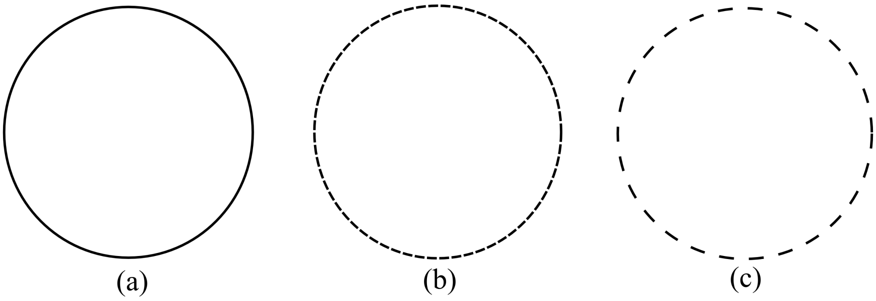


Figure 4.5: (a) A perfect circle. Dotted circles with (b) small pores, and (c) large pores in the boundary.

Hole filling is not always desirable and can be omitted based on the application or query requirement. For example, take the case of disks as shown in Figure 4.5 where (a) depicts a perfectly drawn circle, (b) a circle with small pores, and (c) a circle with relatively larger pores. *Case I*: if the query demands to identify circles only (global perspective only), despite their boundary types, hole filling becomes necessary. On selecting an appropriate value for ω, circles in Figure 4.5(b) & (c) eventually result into the circle as shown in Figure 4.5(a). *Case II:* on the contrary, if the task is to distinguish amongst perfect and dotted circles (local perspective) then we don't want to fill-in the pores. There-

[3] A sliding window protocol is usually found in the literature of network and communication, i.e., Data Link Layer (OSI model) and Transmission Control Protocol (TCP); defined as a feature of packet-based data transmission protocols (Peterson (2000)). Our sliding-window protocol is motivated from this classic model, where we replace the notion of packets with the contour fragments.

[4] The process of hole filling handles the gaps in the boundary segments that could be caused by an artefact or inaccurate boundary extraction process.

fore, the hole filling step should be omitted, or we set ω to 0. However, in the majority of the real-world cases, hole-filling remains in demand. In the current work $\omega = \delta = 2\pi/n$ has been used.

4.1.2 Orientation α_p Assignment

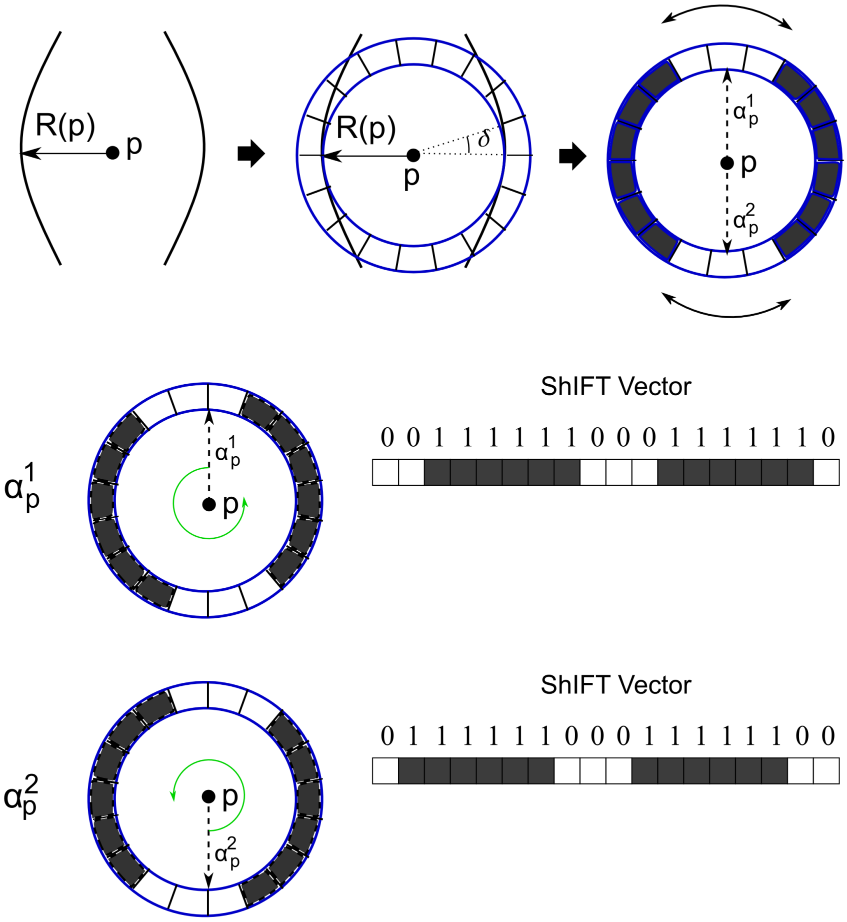


Figure 4.6: Process of orientation assessment for a dominant point p. In this particular configuration two orientations α_p^1 & α_p^2 are possible and hence after the quantization process two ShIFT descriptors are obtained. Here, step angle $\delta = \pi/9$ is selected, i.e., length (n) of the ShIFT descriptor is set to 18 bins.

Alike the hole filling step, rotation invariance depends on the application (Belongie et al. (2002)). For example, an optical character recognition (OCR) does not need rotation invariance when identifying decimal number system; otherwise **6** and **9** will be labelled as the same number. However, often real-world problems require the rotation invariance property for a feature vector.

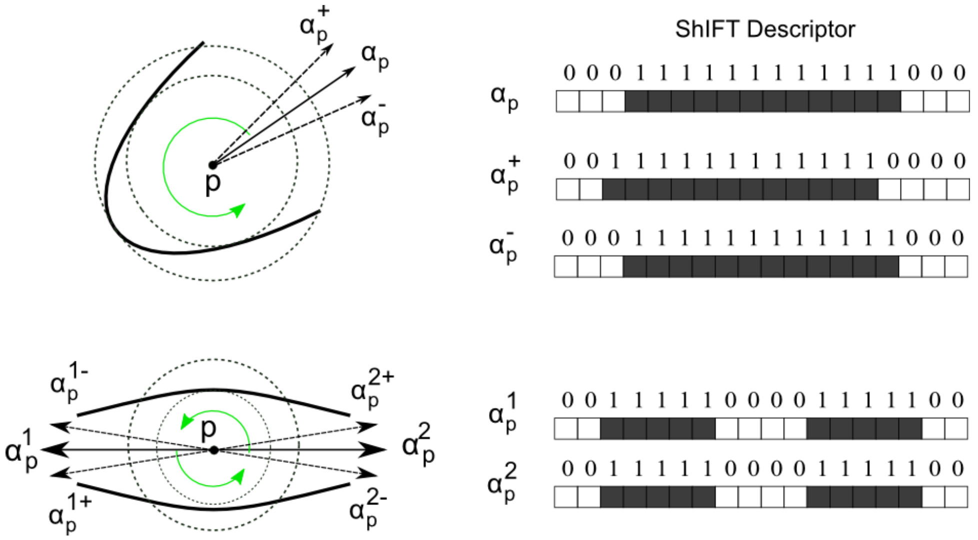


Figure 4.7: Tolerances in keypoint orientations. *Top*: A feature point p holds an orientation α_p. Two bounding orientations $\alpha_p^+ = \alpha_p + \delta/2$ and $\alpha_p^- = \alpha_p - \delta/2$ are created in anti-clockwise and clockwise directions respectively. Together, the resulting orientation range allows to handle cases of noisy boundary where the orientation might slightly deviate and result into different ShIFT descriptors. Resulting ShIFT descriptors are shown on the right hand side. *Bottom*: Bounding orientations in the case of a bumpy surface.

In order to be invariant to rotations, first we identify all possible orientations for the selected dominant points (Figure 4.6). For a particular dominant point p we sub-divide the annulus (of width ε_p) in 360 circular-bins, where each of these is populated with a value of 1 if and only if at least one boundary segment/pixel is located in that particular angular bin. Each consecutively connected 0's in this circular-bin provides an orientation to the dominant point where we take the mid point of the empty segment as the

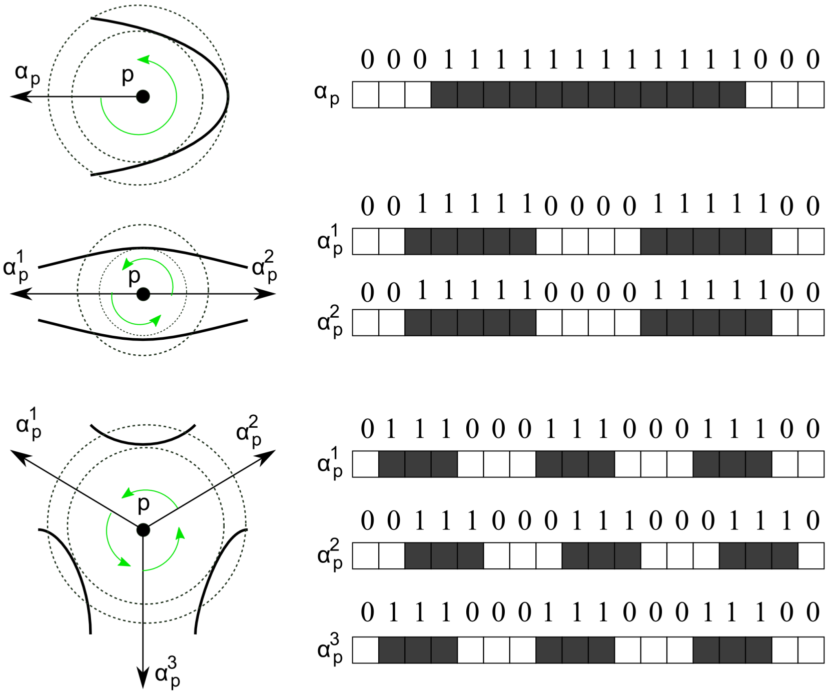


Figure 4.8: ShIFT descriptors with different boundary conditions of the keypoint p: *Top*: Shape is either concave (if p is outside) or convex (if p is inside), limit cases for medialness description. Here p holds a single orientation α_p, i.e., identified by only one descriptor. *Middle*: Shape is bumpy or has a thick structure. Here p confines two orientations α_p^1 & α_p^2, and represented by two ShIFT descriptors. *Bottom*: Shape is a joint structure. Here p has three orientations, α_p^1, α_p^2, and α_p^3, and hence is represented by three descriptors. For each configuration of p, their ShIFT descriptor is shown on the right-side.

direction of orientation. Secondly, we introduce some robustness in our feature vector by including some tolerance for the accepted orientation values. For this, we take angular tolerances with value $\delta/2$ in both clockwise and anticlockwise directions with respect to the keypoint orientation, i.e. $\alpha_p + \delta/2 = \alpha_p^+$ and $\alpha_p - \delta/2 = \alpha_p^-$. Thus for each keypoint orientation, two bounding orientations α_p^+, and α_p^- are added to the set of descriptors (see Figure 4.7). This strategy handles cases where the boundary is slightly deformed or noisy, which otherwise would lead to different ShIFT descriptors.

4.1.3 Bin Aggregation and Quantization

Once the keypoint orientations are identified, the next step is to aggregate the sub-divided 360 bins into n buckets (Figure 4.9). For each keypoint orientation, the process is initialised from the bin pointed at the keypoint orientation and moving in anti-clockwise direction. After mapping to n bins, the next step is to quantize the vector, so that it can easily be used for indexing and shape-retrieval purposes. For binary shapes, any point can be seen either as a boundary or non-boundary point. Similarly, each bin of the ShIFT descriptor reflects whether it is a boundary segment or not. To achieve such representation we map this descriptor to a binary representation, i.e., 0 - non-boundary, and 1 - boundary. However, in case of colored or gray-scaled images, boundary segments can take a range of values, which depends on the range of image intensity. For instance, if the image intensity range is 0-255 and needs to be mapped into the range of $0-1$ with 10 equal subdivision, then the granularity of the feature vector becomes 0.1. Here the intensity value I in the mapped vector becomes $10I/255$.

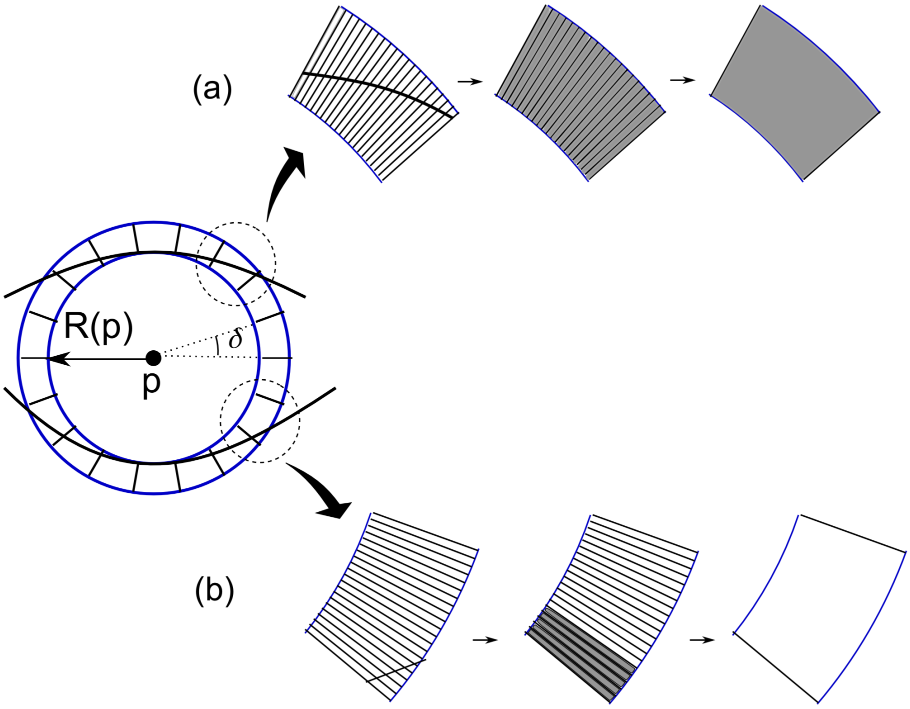


Figure 4.9: Bin aggregation process: In this scenario, $\delta = \pi/9$ has been taken, i.e. the length of the ShIFT descriptor is 18. Initially, the annulus is divided into 360 sectors, which results in having each of the 18 buckets be further divided into 20 sub-sectors. We consider a bin threshold b_{th}= 0.4. (a) In this case, each of the 20 bins receives a contribution through boundary segments, which is 20. Hence, the total value of bin counts reaches 20, which is larger than 40% of the maximum value. As a result, this bucket is considered as part of boundary segment and the bucket value is mapped to 1. (b) In this case, only 5 out of 20 bins receive contributions from boundary segments and their cumulative value falls below the threshold value (0.4×20=8), hence this ShIFT bucket is considered empty (mapped to 0).

Continuing with the binary representation, if W is the maximum value amongst n buckets and b_{th} $(0 < b_{th} < 1)$ is a selected threshold value, then all buckets having value greater than $W \times b_{th}$ are lifted to 1, and otherwise are grounded to 0. Empirically, we have identified that a minimum of one third of W is required in a bucket to consider it as being part of a contour. At the end of this step, a quantized ShIFT descriptor of length n is obtained.

Algorithm 4.1 Outline of ShIFT descriptor Algorithm

For each *dominant point* d_i, a spatial position in a 2D image, where d_i can be internal, concave and convex dominant points:

```
1. Populate first part of vector as $\langle (x,y), R(d_i), \varepsilon_{d_i}, D^*_\varepsilon(d_i) \rangle$.
2. Create an annulus, centered at $d_i$ with radii $R(d_i)$ and $R(d_i)+\varepsilon_{d_i}$.
3. Subdivide the annulus band in 360 empty bins.
4. Populate each bin with $l$ values of boundary segments falling inside the corresponding sector of the annulus.
5. Find possible orientations $\alpha_j^{d_i}$ of $d_i$ as described in Section 4.1.2.
6. Apply hole-filling algorithm as described in Section 4.1.1.
7. Recalculate orientations $\alpha_j^{d_i}$ of $d_i$ as described in Section 4.1.2.
8. Aggregate 360 bins into $2\pi/\delta$ buckets, where $\delta$ is the step-angle. Followed by bin aggregation, quantize the vector. Both methods are described in Section 4.1.3.
9. For each orientation $\alpha_j^{d_i}$, populate the binary ShIFT vector by moving in anti-clockwise direction.
```

4.1.4 Scale, Rotation, and Translation Invariance

By design the resulting feature vector can be shown to remain **invariant to uniform scalings, rotations, and translations,** and up to certain extent to **view-point** changes (i.e., change under projections). Figure 4.10 shows the scale invariance property. Here, the shape on the left side is scaled-up (shown in middle) and scaled-down (on right) uniformly. The resulting annulus for medialness calculation at dominant point p also get resized in a way such that it still captures a set of similar boundary segments in the relative sectors. Hence, at the end, a silimar ShIFT descriptor is obtained demonstrating its invariance to uniform scaling.

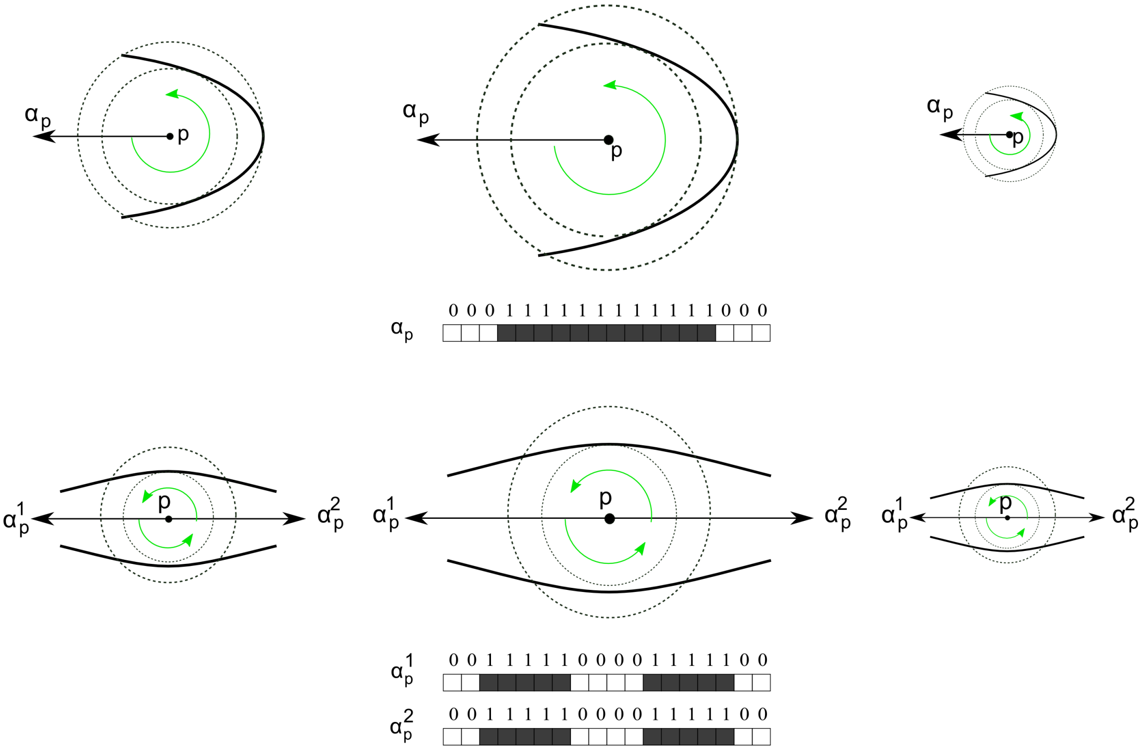


Figure 4.10: Scale Invariance: Shape segments (left) are uniformly scaled-up (middle) or down (right). The medialness annulus operator is correspondingly left invariant.

Figure 4.11 depicts the situation of rotation invariance, where three different shape attributes are represented by three shape segments. To check the rotation invariance property, random rotations are applied over them. As we always start binning ShIFT descriptors from the keypoint orientation, therefore the shape rotation only changes the values of keypoint orientations, however their ShIFT descriptors remain unchanged and makes the rotation invariant property intrinsic. Alike rotation, ShIFT descriptors are not affected by translation while this can only change their spatial location, not descriptor value. Therefore, again, ShIFT holds the translation invariance property intrinsically.

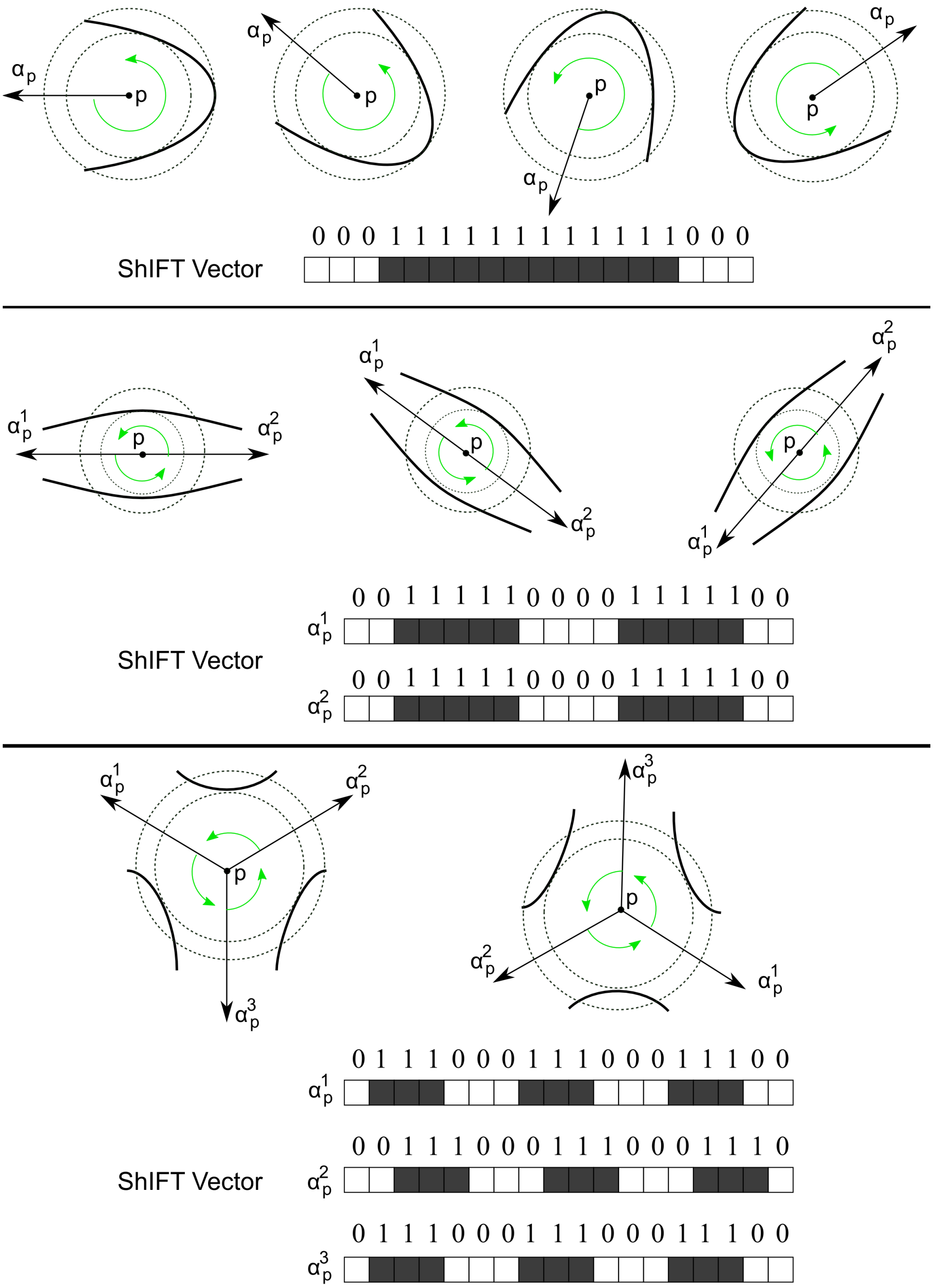


Figure 4.11: Rotation Invariance: Random rotations are applied on three different shape segments. Shape rotation does not affect the resultant ShIFT descriptor for each keypoint orientation.

4.1.5 Behavior Under Viewpoint Changes

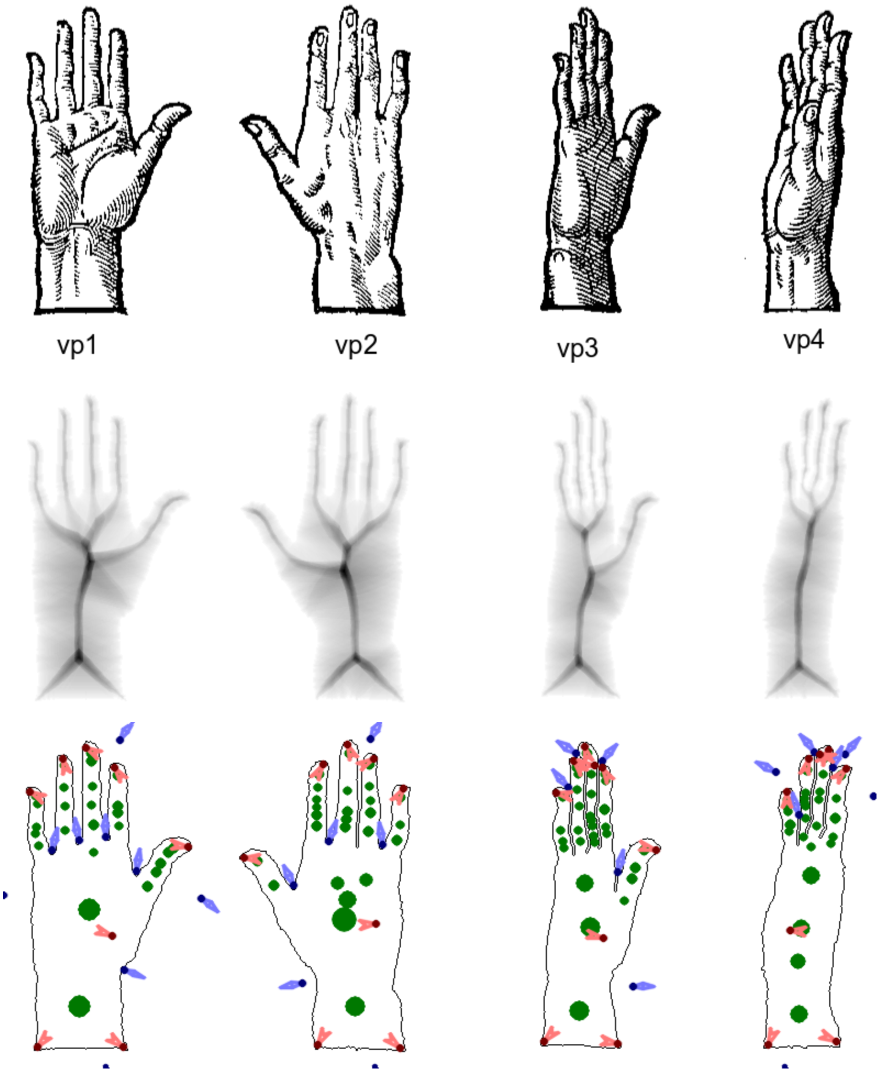


Figure 4.12: *Top*: A hand seen from different viewpoints. *Middle*: Our D_{ε}^{+} function. Medialness measure D_{ε}^{+} remains almost consistent in vp1, vp2, and vp3. While in vp4, the thumb is lost and results in slightly different medialness field. Note that, the function still captures other four fingers. *Bottom*: Dominant points along with their orientations: Interior in green, convex in blue and concave in red. Note here that these images are manually segmented and each finger is isolated, as the goal here is to illustrate the use of medialness, not produce a fully robust segmentation.

2D shape descriptors can be made invariant to changes in viewpoint up to a certain extent but the task is critical as the change in viewpoint alters the shape appearance. For the same object, two different viewpoints can create completely different 2D shapes. For instance, in Figure 4.12 (top), the 2D appearance of a hand is very different at viewpoint vp1 and vp3 or vp4. However, if the viewpoint variation still captures a similar shape information and their extracted 2D shape descriptors categorize them as the same object (i.e., the retrieved feature vectors from these images are mostly same), then this 2D shape descriptor can be said to be partly invariant to *viewpoint* change. Our designed ShIFT descriptor is capable of handling such situations. In the original input image, the fingers are slightly touching or overlapping and there are no easy automatic segmentation solution. Therefore, I do a manual segmentation in that special case, in order to make sure each finger is isolated, as the goal here is to illustrate the use of medialness, not produce a fully robust segmentation. Figure 4.12 (middle and bottom) provides an example behind this claim where medialness and dominant points are consistent in vp1, vp2 and vp3 and hence the ShIFT descriptor still holds the same representation even for a noticeable change in viewpoint.

4.1.6 Object Inversion and ShIFT descriptor

Visually, the shape information should remain the same when an object is inverted.[5] Our designed ShIFT descriptor can easily handle such situations if we apply an inversion rule, i.e. by reversing the order of the ShIFT descriptor. In Figure 4.13, an airplane image is inverted horizontally and vertically. For the original object, the ShIFT descriptor is created by binning the sub-divided annulus in anti-clockwise direction. If we reverse the order from anti-clockwise to clockwise, then the resulting descriptors will correspond to the inverted cases. If a database contains only inverted forms of a query shape, in practice it

[5] The shape is transformed upside down or in left-right (mirror-like reflection).

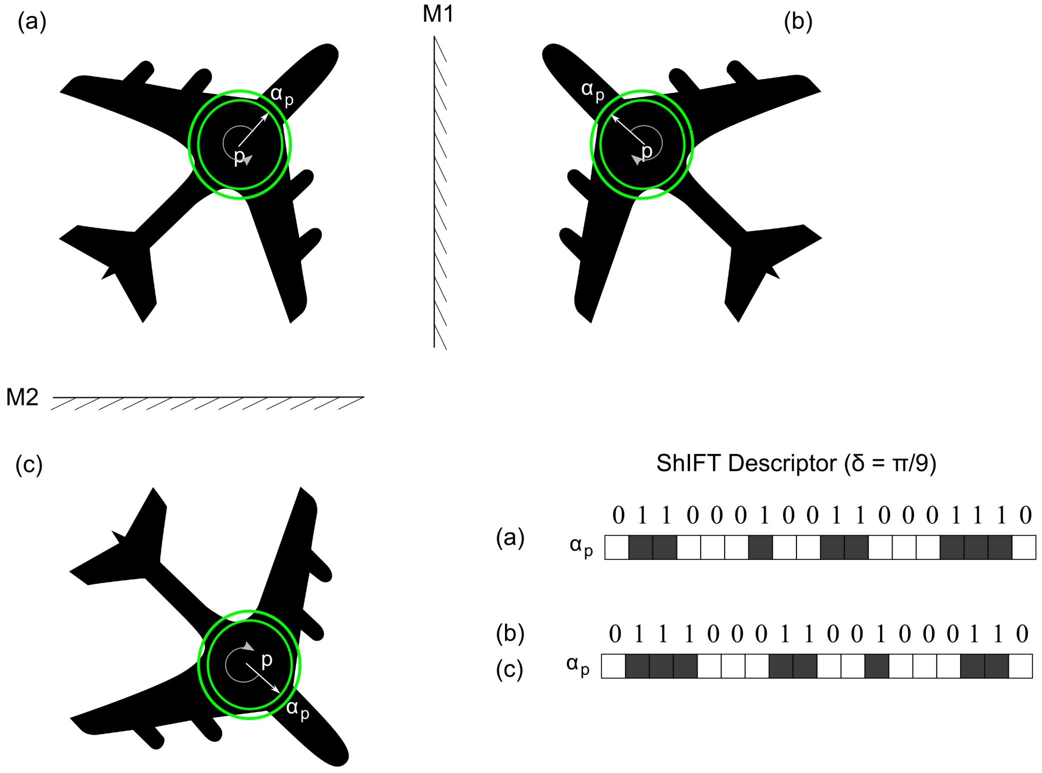


Figure 4.13: (A) A silhouette of an airplane. p is an internal dominant point with α_p as its one of the orientations. (b) Horizontally inverted shape through mirror M1. (c) Vertically inverted object through mirror M”. Inversion not only changes the keypoint orientation but also inverts the ShIFT descriptor representation. Bottom-right side shows the ShIFT descriptors with step-angle $\delta = \pi/9$. Since one time inversion process produces same inverted objects, hence (b) and (c) are same airplanes with different rotations. Therefore, keypoint p has same representation in these two cases, which is the inverted representation of case (a).

is usually important to not miss a potential shape match because of a simple mirror inversion. To handle such situations, we can use two queries per shape, i.e. one representing an original query and another for its corresponding inverted shape. This additional query only cost for the retrieval process from the indexed dataset, which is almost negligible compared to overall time cost. Moreover, this inverted query requires equal space alike the original. Hence the process does not affect overall time and space complexities. Note here that, for homography purpose, values in the affine matrix (discussed in section 5.2) for the inverted query must be changed accordingly.

Object inversion includes those images to the query, which are relevant but got inverted, and hence improves values of both precision[6] and recall[7].

4.2 Dominant Points

The computation of a medialness map result in a field of medial symmetries (alike a terrain where height corresponds to medialness). The thick ridges in the map hold the most informative loci including *dominant points*, which are preferably isolated. Assuming a figure-ground situation, we focus first on the interior medialness representation, and on isolating the interior dominant points. Then we can precede to recover significant interior and exterior ends of ridges as candidate convex and concave loci. Once we have extracted the three types of feature points, we shall make these intelligible to a machine by representing them in the form of feature vectors usable for archiving and retrieval tasks.

[6] In information retrieval, precision (also called positive predictive value) is the fraction of retrieved instances that are relevant.

[7] In information retrieval, recall (also known as sensitivity) is the fraction of relevant instances that are retrieved.

4.2.1 Interior Dominant Points

The interior dominant points extraction process consists of three steps: (a) isolating the medialness ridge via morphological top-hat transform; (b) mapping the medialness ridge to the response value to get the correct position of dominant point in flat regions; and (c) avoid the cluttering of dominant points via minimum distance parameter.

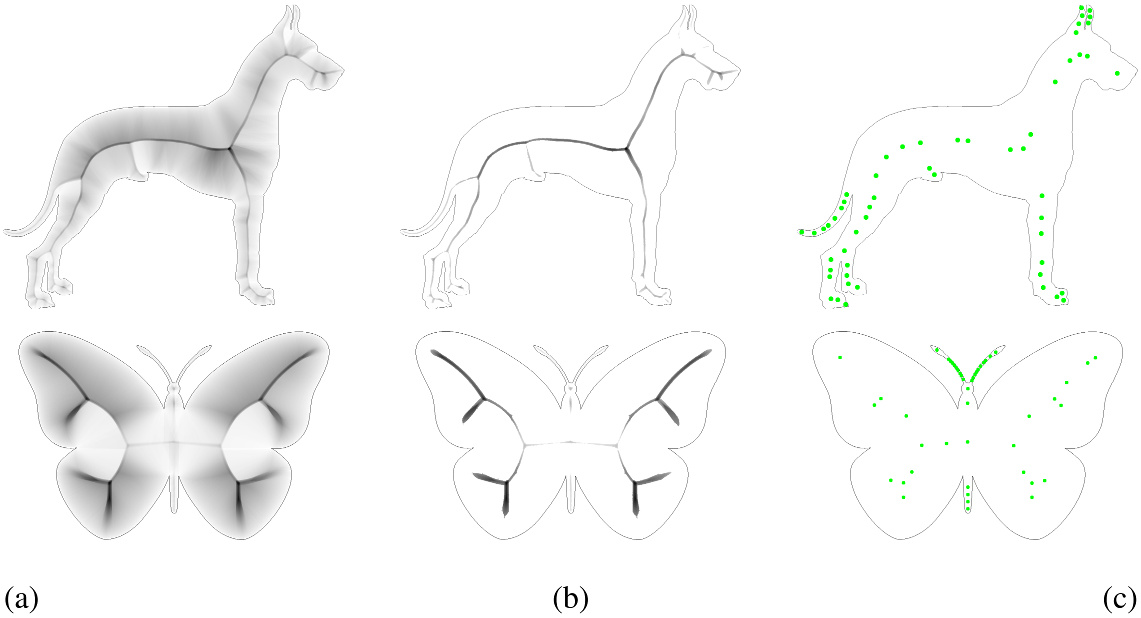


Figure 4.14: Illustration of the three successive steps in isolating *internal* dominant points: (a) medialness representation of (the interior of) a standing dog (top) and butterfly (bottom); (b) corresponding top-hat transform; and (c) internal dominant points illustrated as **green** dots together with the original object's pre-segmented boundary.

Step A:

Medialness increases with blackness in our transformed images (this is for visualization purpose). To select points of internal dominance, a black top-hat transform is applied, resulting in a series of dark black areas which typically correspond to peaks, ridges and passes of the medialness map when considered as a height field, where black means high or proximity to a peak or a ridge. The black top-hat transform is defined as the difference of the *morphological closing* of an input function by a flat structural element, i.e., a disk with radius as single parameter (Serra (1983)). Closing is a set operator on functions which removes small holes from the foreground of an image, placing them in the background (augmenting the local function set values, Serra (1983); Leymarie and Levine (1988); Vincent (1993); Dougherty and Lotufo (2003)). Here, the size of the structuring element varies with the location and is a linear function of ε_p, i.e. $w\varepsilon_p$.[8] This filtering is followed by a thresholding[9] to discard remaining areas of relatively low medialness significance. Figures 3.16 (bottom row) and 4.14 (b) show the result obtained after applying the black top-hat transform on a medialness map. Detailed technical details on Top-hat Transform has been appended as Appendix B.

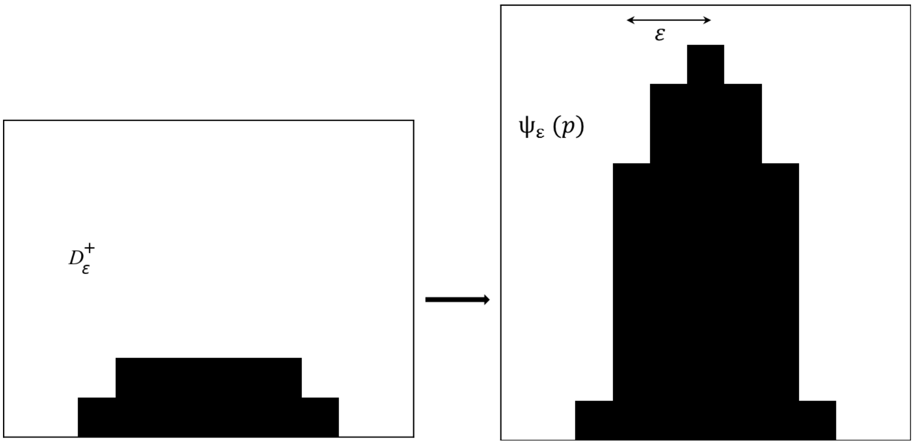


Figure 4.15: *Left*: Illustration of a plateau in medialness response following a top-hat transform where no peak locus is isolated. *Right*: Response-value ψ_ε after filtering the plateau.

[8] Empirically, I found that $w = 1$ works better and produces affine invariant medialness-ridge.

[9] 4-5% of either maximum medialness within the structuring element if the medialness map is not interpolated or maximum of the range of interpolation within the structuring element. Our empirical analysis suggests that the value higher than this threshold range includes spurious and noisy points in the filtered space, which are not along the medialness ridge. A value lower than this designed threshold range can destroy the ridge connectivity and loss of important information also occurs sometimes.

Step B:

We still require to process further the output of the top-hat transform to isolate the most dominant points amongst the remaining selected medialness loci. We also consider the cases where the resulting ridges are more like plateaus and thus rather flat at their top. In order to identify isolated representative dominant points for such plateaus we pull-up such flatten regions and map the central locus of a plateau to the highest local peak value (Figure 4.15). A simple way to achieve this result is to locally modify the value of medialness at each filtered point of the top-hat image to ensure a non-flat "response-value" ($\psi_\varepsilon(p)$):

$$\psi_\varepsilon(p) = \sum_{q \in s} D^+_\varepsilon(q) \tag{4.1}$$

where:

$$s = \left\{ q \mid (|p-q| \leq \varepsilon_{p)} \wedge D^+_\varepsilon(q) \leq D^+_\varepsilon(p) \right\} \tag{4.2}$$

An equivalent solution is to propagate a distance map over a plateau to identify the most central, interior point (Toivanen (1996); Osher and Paragios (2003)).

Step C:

To provide some control of the possible clustering of dominant points, a flat circular structuring element of radius ε_p (but of at least 2 pixels in width) is also applied over the output of a top-hat transform to pick-up maxima. We also impose that no remaining points of locally maximised medialness are too close to each other; this is currently implemented by imposing a minimum distance of length $2\varepsilon_p$ taken between any pair of selected points. An example of applying these steps to identify interior dominant points in medialness is given in Figure 4.14 (c).

This way of identifying interior dominant points is robust and conforms to the preservation of most informative points. However, the minimum distance parameter in Step C is somewhat sensitive to the value of the multiplier of ε_p and there might be chances of inclusion of too many dominant points if the distance parameter has a low value or of discarding some informative points if the distance parameter is too high. Since our designed ε_p is adaptive (equation 3.12), hence it take cares of both these problems and maintain robustness, i.e., the chosen value is designed carefully to be able to both discard having packed clusters of dominant points while keeping important dominant points. Since this algorithmic chain is using heuristics, hence it is not based on first principles. In the future, a more formal approach would consider paths generation over a medialness map to *walk* from peaks to passes to peaks via ridges. What we have designed here instead is approximated computation that is quickly obtained and produces robust results.

4.2.2 (Exterior) Concave Points

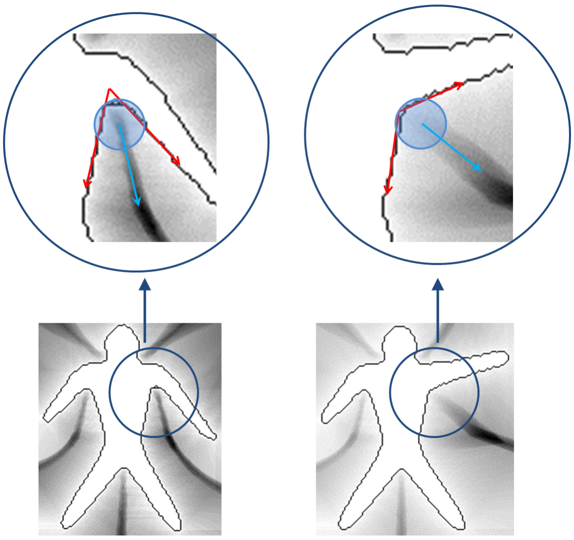


Figure 4.16: *Bottom row*: External medialness processing on a humanoid object. The articulated movement of the left arm changes the location and orientation of the associated external dominant point (near the concave curvature peak). If the external dominant point is reasonably far from the contour, then it proves difficult to retrieve a (shape-based) match with the modified form. *Top row*: Red arrows show the local support for concavity while blue arrows indicate the direction of flow of medialness (away from the concavity).

In practice, if an object can be deformed or is articulated, salient concavities can be identified in association to those deforming or moving areas (such as for joints of a human body). Considering this empirical observation, the location of an external dominant point can be made invariant to this deformation/articulation only up to a certain extent. For example, if the location of an external dominant point in medialness is initially relatively far away from the corresponding contour segment, a slight change in the boundary shape near the movable part (such as an arm movement) can considerably change the position of that associated dominant point (Figure 4.16). On the other hand, if a potential concavity point is located very close to the contour, it can easily be due to noise or small perturbations in the boundary reflecting a local maximum of curvature. Therefore, to be able to retrieve reliable concave points, it is first required to provide an adapted definition of concavity as a significant shape feature.

Method I (Contour following): We define a point of local concavity if it falls under a threshold angular region, under the constraint of length of support which itself depends on the tolerance value (ε). The value of the threshold (θ_{ext}) is tunable but is always less than π, which permits to control the angular limit of the concave region. A point whose local concavity is larger than θ_{ext} is considered a flat point. In our experiments we tune the value of θ_{ext} int the range $[5\pi/6, 8\pi/9]$. In association, we define an external circular region (of radius function of ε) centered at each locus containing candidate external dominant points. Each such region may provide only one representative dominant point, where the dominance of a particular point is decided by the maximum containment of boundary points inside the associated annular gauge (of exterior medialness) and corresponds to (our definition of) the *maximum length of support*. Finally, we position the representative dominant point to be near the contour at a fixed distance outside the form.[10]

[10] This heuristic, of positioning the representative concavity near the object contour trace is useful both for visualisation and for greater robustness in matching under articulated movements.

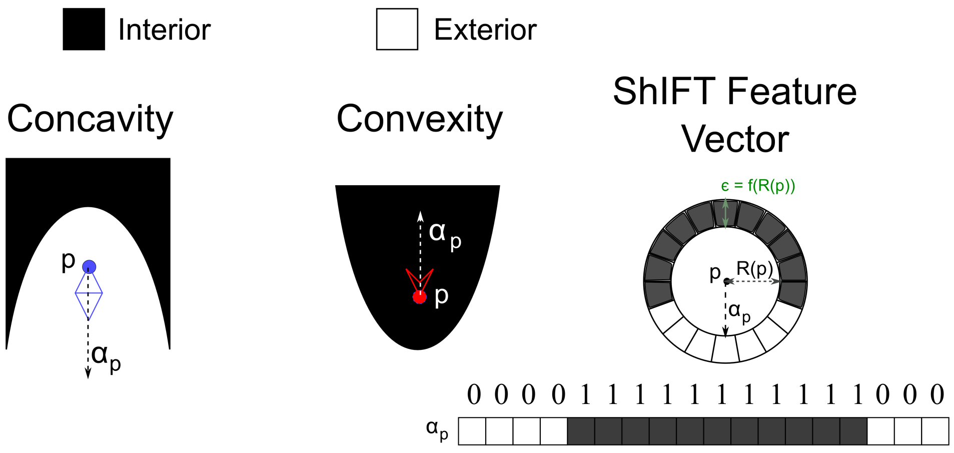


Figure 4.17: A feature point p represents *left*: concavity, and *middle*: convexity. Their respective orientations α_p are shown as blue needle and red arrow. *Right*: Their resultant ShIFT descriptor.

Method II (Using the ShIFT Vector): We note that points lying outside the shape can have more filled bins when the shape is concave and have fewer filled bins when the shape is locally convex, whereas the opposite is observed for interior candidate points near the end of a ridge of medialness. To improve the labeling of whether a contour segment is significantly concave or convex, we identify a minimum number of *concurrently filled bins*. The point can then be confirmed as concave (convex) if it is associated with at least k successive filled bins (typically $k = 4n/9$, proves empirically useful) and has highest medialness value in the surrounding of a circle with radius $R(p)$. With this verification stage, typical feature vector configurations are alike the following:

(a) $0\cdots01\cdots10\cdots0$: convex or concave

(b) $1\cdots10\cdots0$: convex or concave

(c) $0\cdots01\cdots1$: convex or concave

(d) $\ast\cdots\ast10\cdots01\ast\cdots\ast0$: neither convex or concave, where “$\ast$” is a “don’t care” (can be either “0” or “1”).

4.2.3 Interior Convex Points

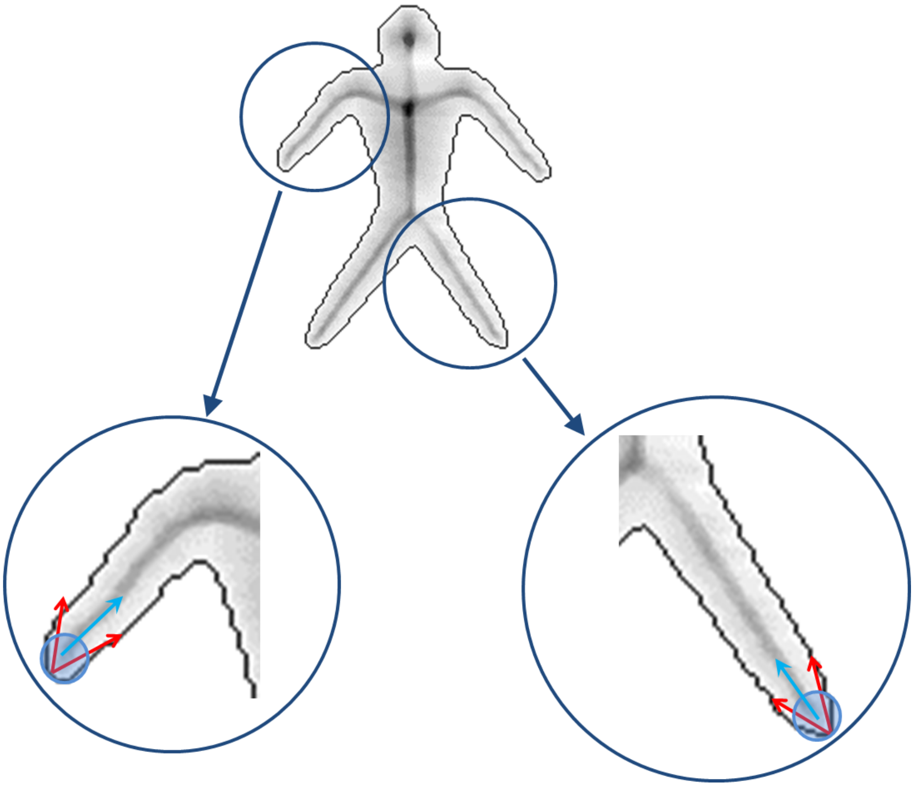


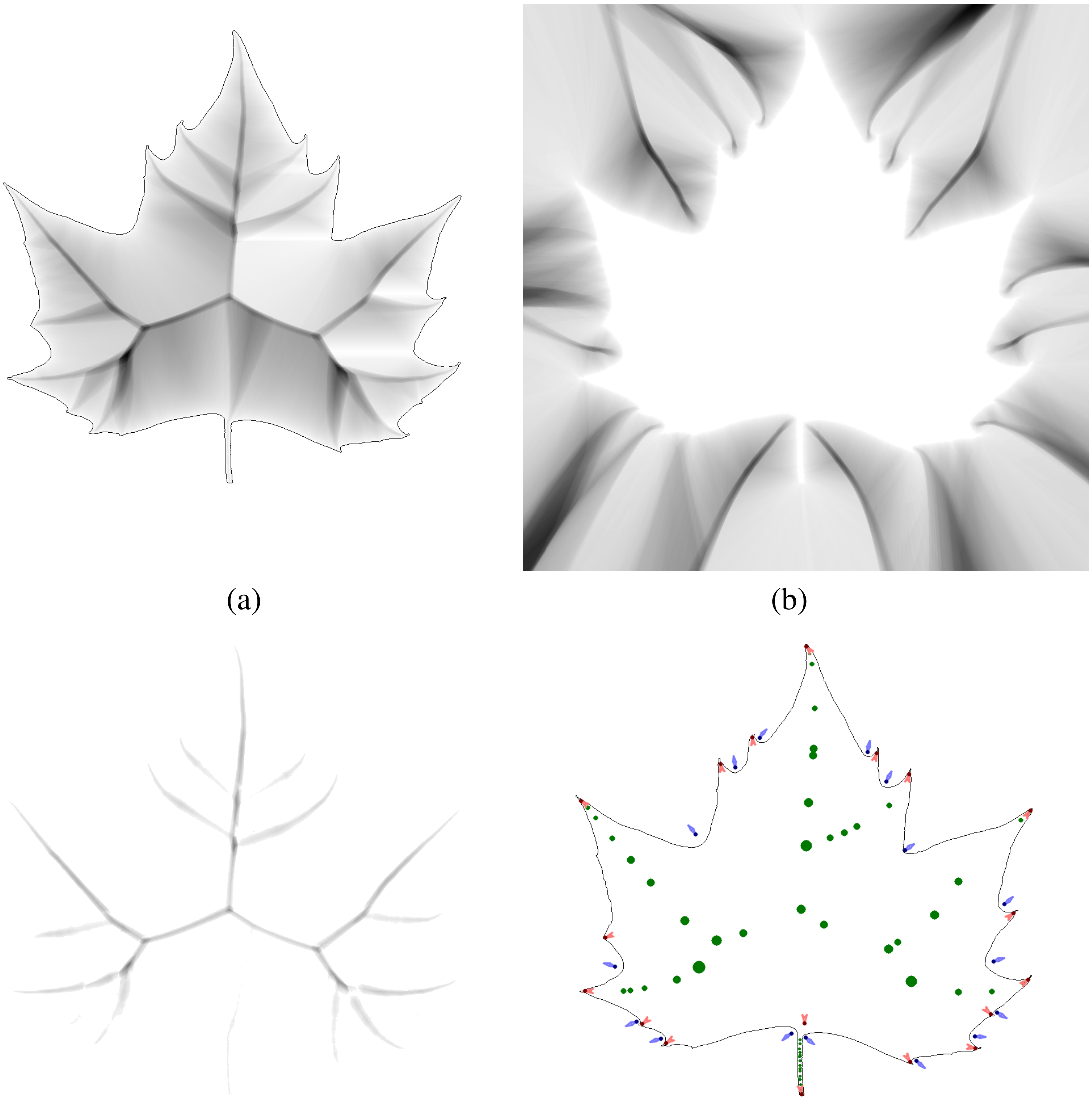
Figure 4.18: Illustration of interior medialness processing on a humanoid object to identify significant convexities in the vicinity of the ends of medialness (ridge) trace.

Our final shape feature is a set of *convex points*, where an object's figure has sharp local internal bending and gives a signature of a blob-like part or significant internal curvature structure (*i.e.* a peak in curvature with large boundary support). The goal is to represent an entire protruding sub-structure using one or a few boundary points. In has been recognised for a long time that such protrusions have a significant contribution in characterising a form's figure (Richards and Hoffman (1985); Leymarie and Levine (1992); Berretti et al. (2000); Srestasathiern and Yilmaz (2011)). The process of extraction of convex points is similar to the extraction of concave points, the main difference being the value of threshold angle (θ_{int}), where $\pi < \theta \leq 2\pi$. In our experiments we have found useful values to be in the range: $[1.25\pi,\ 1.33\pi]$ (Figure 4.18).

Convex and concave feature points are complementary to each others and have been used in the *codon* theory of shape description: a codon is delimited by a pair of negative curvature extrema denoting concavities and a middle representative positive maximum of curvature denoting a convexity (Richards and Hoffman (1985)). In our case we relate these two sets with the extremities of the medial axis of Blum (1973): end points of interior branches correspond to center of positive extrema of curvature and end points of exterior branches are mapped to negative extrema of curvature of the boundary. Furthermore, we note that the repositioning of these extrema near the boundary is alike the end points of the Process Inferring Symmetry Axis (PISA) representation of Leyton (1992).

4.2.4 Feature Point Set

Together, the three point sets derived from medialness — Interior, Concave and Convex — form a rich description of shape which can, in particular, be used to address the difficult problem of building a shape-based matching system. Each feature point p has associated information made from two distinctive parts: part 1 describes the spatial and geometrical information of the dominant point (associated with medialness):$\langle (x,y),\, R(p),\, D^*_\varepsilon(p),\, \alpha_p \rangle$, while part 2 is the ShIFT vector made of up to n entries corresponding to the way we partition a medial annulus band into sectors (or bins). Our designed ShIFT is a "rich" descriptor of shape and it can be used to address different important problems, such as indexing over a large shape datasets (information retrieval aspect), shape retrieval and matching, movement computing. Moreover, from this set a coarse medial axis (*MA*) graph is recoverable, hence it has *MA* applications. As our ShIFT feature is quantized in binary form, so in total it can have a maximum of 2^n different forms. A larger value of δ will produce a smaller number of bins and hence reduce accuracy. On the other hand, a lower value of δ will produce a larger number of bins, perhaps making the mapping



(a) (b)

(c) (d)

Figure 4.19: Maple leaf shape with (a) Interior medialness, (b) exterior medialness, (c) interior ridge obtained after top-hat transform, and (d) feature points recovered: medial hot spots (in green), and significant convexities (in red) and concavities (in blue). Here size of interior feature points is proportional to their weight, i.e., the value of D_{ε}^{+}.

more sensitive to noise. Our empirical study shows that a value of δ in the range of $[\pi/12, 2\pi/15]$ leads to a good compromise (with the considered datasets). In this thesis we report results with $\delta = \pi/9$ resulting in feature vectors of length $n = 18$.

4.3 Shape-based Image Indexing using ShIFT Features

Processing efficiently the exponentially growing numbers of digital images and video sequences is more and more needed in order to archive, retrieve, modify and manipulate such sources of big-data (Wiederhold (1995, 2016)) . In daily life, one of the major search query on the Internet is to find the most relevant images in such large databases, which contain some samples of the object of interest. To make such searching queries practical, effective shape-based image encoding and indexing based on object semantics is becoming increasingly important. In current real-world image databases and query processing, two different techniques are frequently in use: (a) the prevalent retrieval techniques that involve human-supplied text annotations to describe object semantics in images or video sequences, and (b) content or feature-based methods. There are many problems in using the former approach, for example, different people may supply different textual annotations for the same image and hence, this makes it difficult to reliably answer user queries. Furthermore, entering textual annotations manually is expensive for very large and growing image databases. While on the other hand, feature-based methods are proven to be fast, although the compromise is with robustness and accuracy.

With large image databases, indexing is the key technique for fast searching. Such indexing can be performed in two ways: (a) concept-based image indexing, and (b) content-based image indexing. Concept-based image indexing, usually referred as "description-based" or "text-based" image indexing/retrieval, refers to retrieval from text-based indexing of images that may employ keywords, subject headings, captions, or natural language text. Content-based image retrieval (CBIR), also known as query by image content

(QBIC), is the application of computer vision techniques to the image retrieval problem. “Content-based” means that the search process analyzes the main contents of the image rather than metadata. In our case, content refers to shape features.

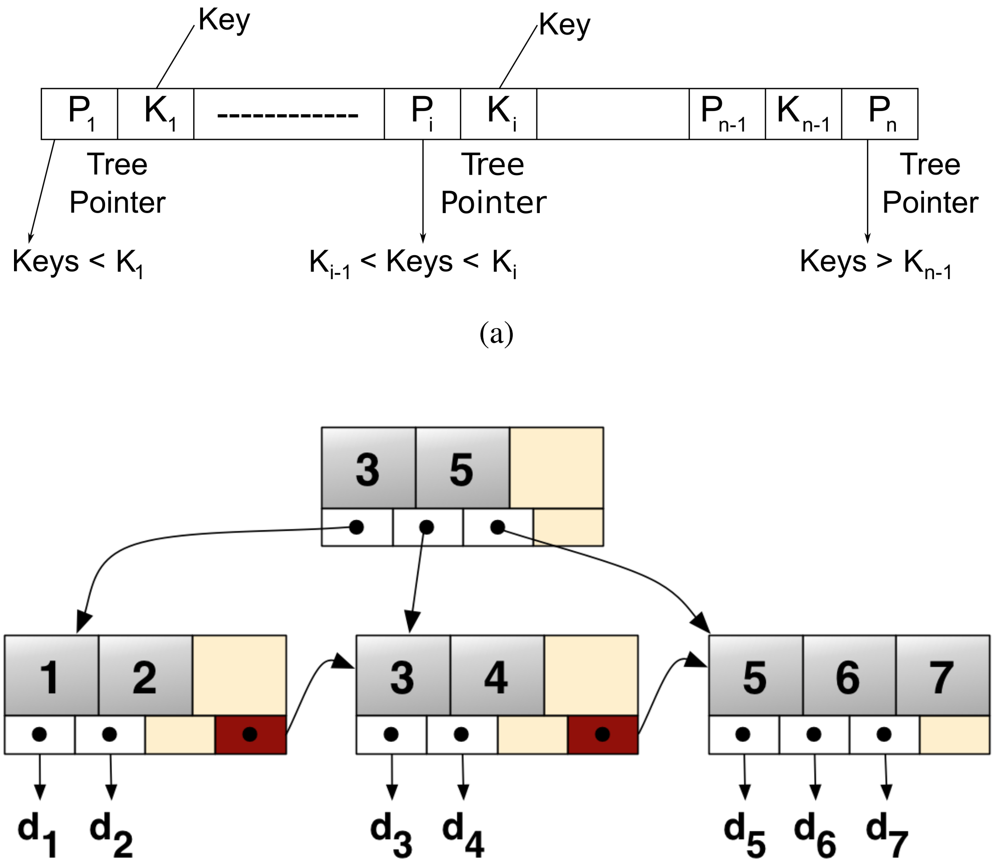
A feature vector is indexable if it shows the properties of granularity and quantization. The published popular shape descriptors, such as Shape Context (SC) and Inner Distance Shape Context (IDSC), are not in a form which makes them applicable to the practical case of information retrieval amongst millions (or more) of shape samples. On the other hand, feature descriptors like SIFT (Scale-Invariant Feature Transform) Lowe (1999, 2004) and SURF (Speeded-Up Robust Features) Bay et al. (2006, 2008) were designed to work well for such indexing purpose, but they lack in providing explicit shape representation or information. Our ShIFT feature vector is already binarized and hence quantized, and can thus be used for indexing. Hence, this descriptor representation provides explicit shape information as well as make it indexable.

One can use the sequential indexing method (Indexed Sequential Access Method, ISAM) to organise the feature vectors. The main disadvantage of this indexing technique is that performance degrades as the file grows, both for index lookups and for sequential scans through the data. Although this degradation can be remedied by reorganization of the file, frequent reorganizations are undesirable. On the other hand, B+ tree index structure maintains their efficiency despite insertion and deletion of data (Silberschatz et al. (1997)).

In the current work, a B+ tree is used to index ShIFT feature vectors.[11] The B+ tree is a variation of the B-tree data structure, where it combines the features of both ISAM and B-tree (Ramakrishnan and Gehrke (2000)). Here, data pointers are stored only at the leaf nodes of the tree and hence the structure of leaf nodes differ from the structure of internal nodes (Figure 4.20). The leaf nodes have an entry for every value of the search

[11] A B+ tree is an n-ary tree with a variable number of children per node; it is an extension of the classic B-tree (Elmasri (2011)).

field, along with a data pointer to the record (or to the block that contains this record). The leaf nodes are linked together to provide ordered access on the search field to the records, while internal nodes are used to guide the search. Some search field values from the leaf nodes are repeated in the internal nodes. More extended details on the B+ tree can be found in Appendix C.



(a)

(b)

Figure 4.20: (a) The structure of the internal nodes in a B+ tree. (b) A simple B+ tree example linking the keys $1-7$ to data values $d_1 - d_7$. The linked list (red) allows rapid in-order traversal. This particular tree's branching factor is b=4.

Some characteristics of the B+ tree with h levels of index and order[12] b in the tree are:

[12] The order, or branching factor of a B+ tree measures the capacity of nodes (i.e., the number of children nodes) for internal nodes in the tree.

- The maximum number of records stored is $n_{max} = b^h - b^{h-1}$
- The minimum number of records stored is $n_{min} = 2\left\lceil \frac{b}{2} \right\rceil^h$
- The minimum number of keys is $n_{kmin} = 2\left\lceil \frac{b}{2} \right\rceil^h - 1$
- The maximum number of keys is $n_{kmax} = b^h$
- The space required to store the tree is $O(n)$
- Inserting a record requires $O(\log_b n)$ operations
- Finding a record requires $O(\log_b n)$ operations
- Removing a (previously located) record requires $O(\log_b n)$ operations
- Performing a range query with k elements occurring within the range requires $O(\log_b n + k)$ operations

4.4 Summary

In this chapter we addressed most of the issues raised in Chapter 3. Followed by the detailed description of our designed feature descriptor ShIFT, we presented the algorithmic chain in order to extract the dominant points also known as feature points or hot-spots, as indicated in the literature of Kovács et al. (1998); Kovács (2010), from a medialness map. We provided a representative description of these dominant points through our designed descriptor ShIFT, which provides a machine intelligible representation.

We illustrated the affine invariance property of the ShIFT descriptor along with its behavior under viewpoint changes. Motivated by the literature on data transfer protocol in networking, i.e., using a sliding-window protocol, we addressed the case of inaccurate contour extraction followed by a binarization process that leads to erroneous holes in the resulting boundary segments. We presented finer details on the orientation assessments and quantization process of the descriptor. Moreover, we addressed the behavior of the descriptor under object inversion, which is not handled by most of the existing shape descriptors.

The notion of articulated movement is captured via the analysis of concavities and convexities in the shape. We presented two complementary methods to detect these extremities in the shape and also commented on the notion of flatness in the shape. At the end, the ShIFT descriptor is defined by two parts, one is responsible for affine parameter calculation and homography, while the second part (binarized and quantized) is helpful for indexing. To index our ShIFT feature vectors, we used a B+ tree index structure, which proves efficient in retrieval queries.

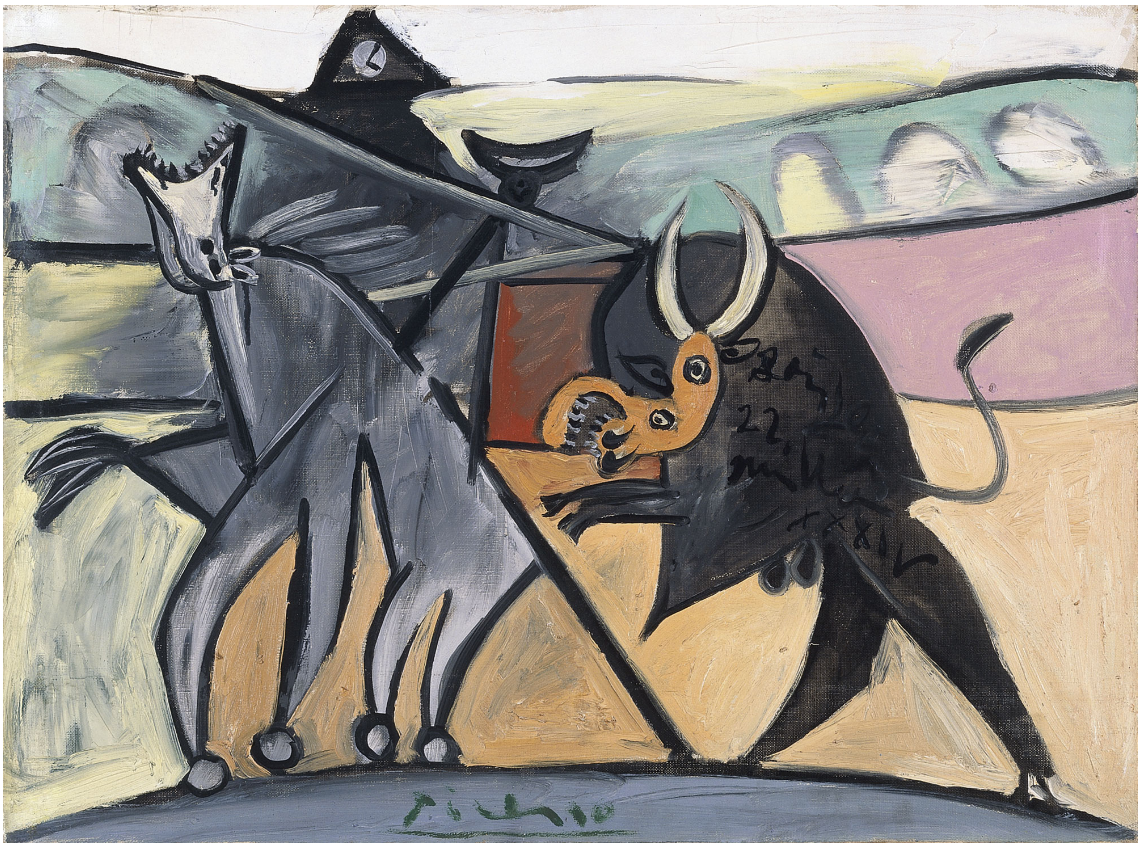
At this particular point, the following questions are still unanswered:

1. We have obtained the ShIFT descriptor containing two parts: (a) a set of parameters and (b) an n-length binarized and quantized vector. How can we use these parameter to identify affine parameters?

2. After getting these parameters, how to perform a homography and shape matching task?

3. Once the shape matching task is accomplished between two objects, can we distinguish the case of matching of a query image with two different image? How can we quantify such situations? Or in other words, how can we rank the shape matching task on a large dataset?

4. It is an inappropriate task to perform one-to-one shape matching when the dataset is too large. Is there any way to filter out irrelevant objects quickly and rank the others on some criteria, in order to check the homography on only the N top-ranked shapes?



Pablo Picasso, *Bullfight*, 1934. @Museo Thyssen-Bornemisza, Madrid.

Chapter 5

Shape Retrieval and Matching

"We should not give up and we should not allow the problem to defeat us."

– Avul Pakir Jainulabdeen Abdul Kalam, Missile Man of India (1931–2015)

With the invention of digital photography and video, the world has become to be small enough to be stored in anyone's collection. Large digital image and video collections typically contain large amount of information related to human activities such as image based web searching, architectural and engineering design, movement computing, fashion and design technology, medical diagnosis, cultural heritage, geographical information and remote sensing systems, home entertainment, human computer interaction, education and training, etc. After image and video acquisition, people usually tend to reorganize their collections in a way such that it is good for searching later, i.e., spending minimal time for finding best result. The comprehensive manual analysis, search and retrieval on collections are acceptable when these remain small. Such manual retrieval time increases rapidly when the collections grow large enough and contain diversified categories making it almost impossible to produce effective results in a reasonable time. Therefore, an important issue associated with such problems is how to offer users more effective technologies to supply results acceptable in terms of both time and accuracy.

The objective of this chapter is to design robust indexing and retrieval algorithms by exploiting ShIFT descriptors. We discuss how we use our ShIFT descriptor for indexing large image databases, retrieving top best matches from this indexed databases. Furthermore the retrieved results are used to find the homography between query and target images by means of matching feature points in an efficient way. In order to perform the homography, we first identify affine parameters namely: scale, rotation and xy- translation parameters. When the test and target images represent similar shapes, the homography process finds the scaling and rotation factor between these. If the images contain dissimilar shapes (for example matching butterfly with a horse) then it finds the best possible part-based match. The retrieved results are needed to be ranked based on how much relevance they have with the test case. In this thesis we use three ranking metrics: Percentage Match or Precision, standard F-measure, and Relative Score, which provides a weight to each matching pair. We also provide an overall performance measure to our system based on standard Mean Average Precision (MAP) value.

5.1 ShIFT Descriptor-based Shape Retrieval

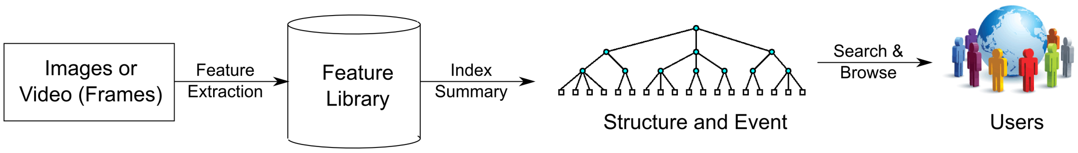


Figure 5.1: Flow chart: Content-based image/video retrieval and indexing model.

In the current context "content-based" is in terms of shape information, here it is defined through ShIFT descriptors interpreted as primitive visual features. These descriptors provide both low and medium level shape information. For example, when observing a hand, local discriminative parts include finger tips, while global features like the palm are also captured and described through this descriptor. Since these descriptors are quantized, they are directly available for indexing. A conventional content-based image/video retrieval and indexing system is illustrated in Figure 5.1.

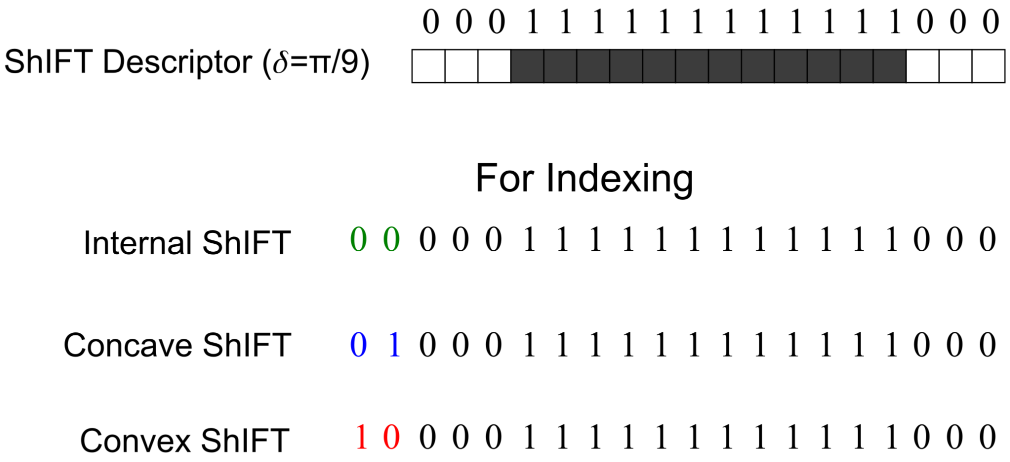


Figure 5.2: Type conversion or identity detection of the ShIFT descriptors for indexing purpose. *Top*: A ShIFT descriptor obtained after feature extraction process, which can be internal, concave, or convex. *Bottom*: For dominant type representation, 2-bits are prefixed before the descriptor: 00 for internal, 01 for concave, and 10 for convex dominant points.

In this work, first, feature points are located from query and target images and kept with separate identities as members of three types: internal, concave and convex. To utilize their identity in the descriptors, each ShIFT feature vector is first prefixed with two bits such that: 00 = internal, 01 = concave or external, and 10 = convex (Figure 5.2). This means the length of the descriptor is always incremented by 2. As introduced in section 4.3, a B+ tree is used to index a ShIFT descriptor, for which the complexity to index the database is $O(n\log_b n)$, while retrieval is $O(\log_b n)$.

For a given query shape, each descriptor is exploited to find the respective target in the indexed database which is stored at a leaf node of the B+ tree. The retrieved results are then needed to get scored on the basis of a ranking metric, in order to find the amount of correspondence with the given query. Once these outcomes obtain scores, a voting scheme is applied to rank them. Out of these ranked outcomes, if required by the search engine, the top N outcomes can be used for illustration purpose. Query to target (spatial) fitting can be done by finding the corresponding homography[1], which is explained next.

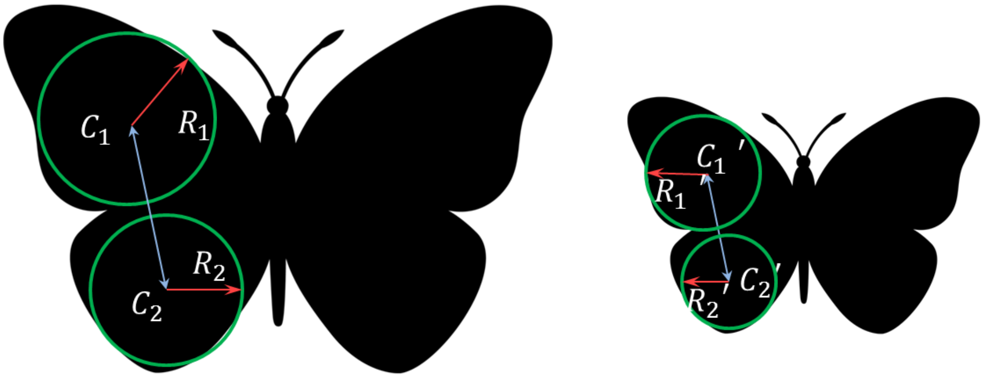
5.2 Homography and Shape Matching Algorithm

For query to target fitting, it is required to find their best possible homography. A homography is a transformation from one projective plane to another, in our case, homography is a transform to map objects of one image to the objects of another image. Consider a point $p = (x_i, y_i)$ in one image and $q = (x_j, y_j)$ in another image. The homography is a 3×3 (for a 2D transform) or a 4×4 (for a 3D transform) matrix H, and relates the pixel co-ordinates in the two images if $q = Hp$.

Once the feature points are located from query and target images and kept as separate identities, the next task is to evaluate their relative affine parameters, i.e. scale, rotation and translation. The order of evaluating these parameters are invariant and final result will be same. In our representation, internal dominant points holds discriminating feature to identify all these three affine parameters correctly. However, concave and convex dominant points can be used to identify rotation and translation, but they cannot be used to get the correct scaling factor between query and targetobject. Hence, internal dominant points are used as the initial keypoints for evaluating scale, rotation and translation of the query image with respect to a particular target image. The next step is to improve the correctness of the matching algorithm. For this, concave and convex points are used as additional information. Each feature point p has associated information made from two distinctive parts: part (1) describes the spatial and geometrical information of the dominant point (associated with medialness):$\langle (x,y), R(p), D_{\varepsilon}^{*}(p), \alpha_p \rangle$, while part 2 is the ShIFT vector made of up to $n = 2\pi/\delta$ entries corresponding to the way we partition a medial annulus band into sectors (or buckets). Here (x,y) is the 2D location of a feature point p, $R(p)$

[1] In projective geometry, a homography is an isomorphism of projective spaces, induced by an isomorphism of the vector spaces from which they are derived. It is a bijection that maps lines to lines, and thus a collineation (Berger (2009)).

is the minimum radial distance of the point from the contour, $D^{*}_{\varepsilon}(p)$ is the medialness measurement, and α_p is the orientation with respect to the positive x-axis. The test case (query Q) is represented as: Q= $\{Q_I$,Q_E,$Q_c\}$, for internal (Q_I), concave (or external) (Q_E) and convex (Q_C) feature points; while the target image (T) is similarly represented as: T = $\{T_I$,T_E,$T_C\}$. For affine parameter evaluation, we follow a brute-force method where each element of Q_I is compared with each element of T_I following four stages.



(a) (b)

Figure 5.3: Illustration of how scaling of an object is evaluated towards matching. Here (a) a butterfly form is used, to be matched with (b) a scaled-down version.

If an object is uniformly scaled by a factor β, we expect the distance between any two similar parts to be scaled by the same factor. In Figure 5.3, an example of scaling is shown for a butterfly object: consider the top-left (centered at C_1 with radius R_1) and bottom-left (centered at $C_2$with radius R_2) parts of a butterfly (a) being scaled-down to corresponding disks: centre $C_1^{'}$ with radius $R_1^{'}$ and centre $C_2^{'}$with radius $R_2^{'}$ part of the butterfly (b), then:

$$\frac{R_1}{R_1^{'}} = \frac{R_2}{R_2^{'}} \Rightarrow R_2^{'} = R_2 \times \frac{R_1^{'}}{R_1} \tag{5.1}$$

Here, the scaling factor for butterfly (b) having a disc with radius $R_2^{'}$ w.r.to butterfly (a) having a disc with radius R_2 is $\beta \equiv {}^{R_1^{'}}\!/_{R_2^{'}}$. Similarly, their respective distance vectors C_1C_2

and $C_1^{'}C_2^{'}$ get scaled by the same factor, *i.e.*:

$$\frac{R_1}{R_1^{'}} = \frac{C_1C_2}{C_1^{'}C_2^{'}} \Rightarrow C_1^{'}C_2^{'} = C_1C_2 \times \frac{R_1^{'}}{R_1} \tag{5.2}$$

Stage I is thus used to identify a scale (β) as well as identify the required translation of the query image by matching a dominant point. Stage II is a check on the scale β to make sure at least one more internal dominant point can be used in the matching process (if not, move to a different dominant point not yet considered). In stage III, the rotation of the query image (w.r.to the target) is evaluated. Finally, stage IV modifies the Cartesian positions of each feature point of the query image by applying the evaluated scale, rotation and translation and we proceed to measure a matching performance value.

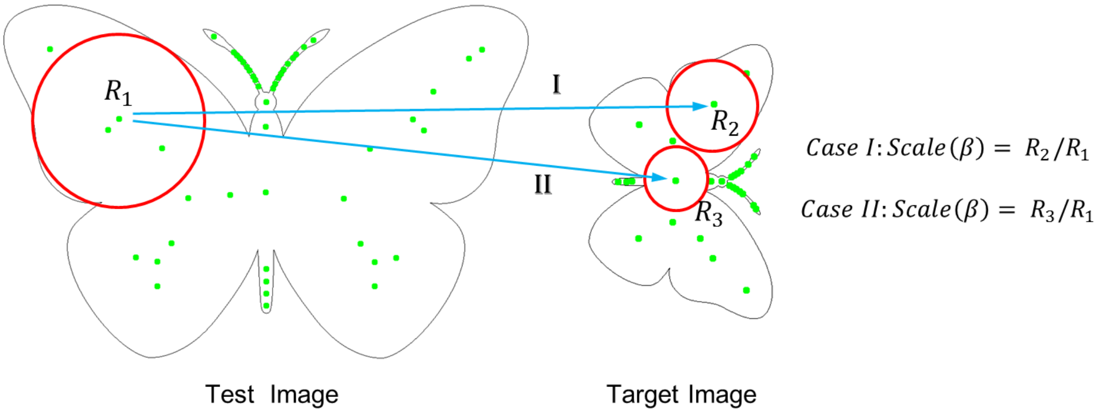

R_1
I
II
R_2
R_3
$Case\ I\!: Scale(\beta) = R_2/R_1$
$Case\ II\!: Scale(\beta) = R_3/R_1$
Test Image
Target Image

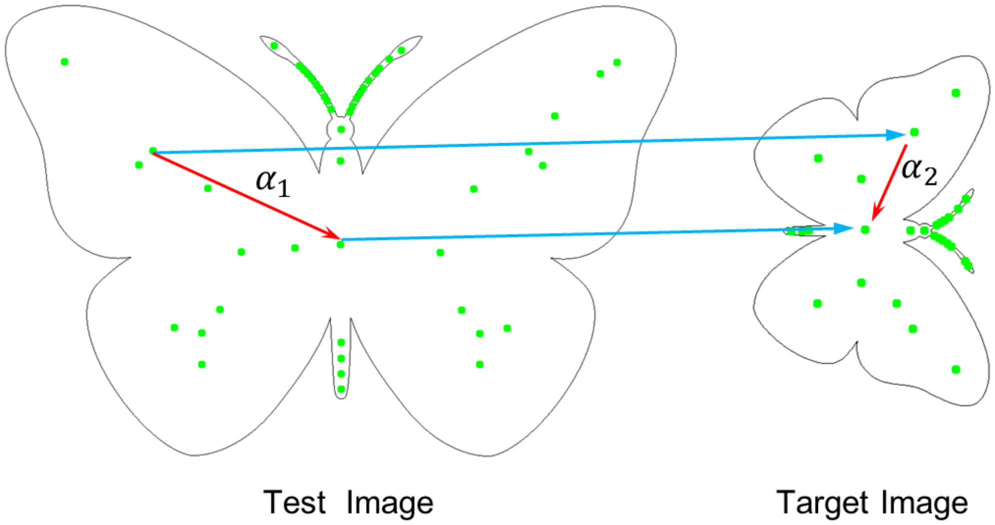
Figure 5.4: Illustration of the evaluation of scale (β). In the query image on the left, for a particular dominant point falling inside a red circle, two possible matching locations in the target image, are shown: cases I and II. In each case the circle's radius is dictated by the minimum distance to the contour (from medialness) and the scale (β) is given as the ratio of the minimum radial distances of target vs query.

Stage I: Take an element (q_i) from set Q_I and match it with each element (t_j) of set T_I. For each pair of (q_i,t_j), the scale (β) is evaluated as (Figure 5.4):

$$scale(\beta) = \frac{R_{t_j}}{R_{q_i}} \tag{5.3}$$

The scale (of query image w.r. to target) for the matching pair (q_i,t_j) is defined via two translations, one for each axial direction: $\overrightarrow{q_x t_x} = t_x - q_x$ and $\overrightarrow{q_y t_y} = t_y - q_y$, where q_i = (q_x,q_y) and t_j = (t_x,t_y).

Stage II: Now take the next element (q_{i+k}) from set Q_I and match it again with each element (t_j) of set T_I. For each pair of (q_{i+k},t_j) find the scale $\beta^{'}$. If the ratio of $\beta^{'}$ over β is under the tolerance level β_{th}, then goto stage III. Otherwise, repeat at another point and check the same tolerance criterion β_{th} and repeat if necessary until all the elements of Q_I are counted. The value $\beta^{'}$ (if it is under the tolerance level) ensures compatibility under scaling of the query image (with respect to a target) and helps in finding the matching location of the next internal dominant point to consider (blue arrows in Figure 5.4).



$$Rotation\ (\theta) = \alpha_2 - \alpha_1$$

$$Image\ Orientation\ (\alpha) = Angle\ between\ line\ joining\ matching\ dominant\ points\ and\ positive\ X - axis$$

Figure 5.5: For both query (test) and target images, the orientation (α) is the angle between a line joining the two matching internal dominant points (shown with blue arrows) and a positive x-axis. The required rotation (θ) of the query image w.r.to the target is given by the difference in orientations.

Stage III: After stage II, if (q_i,q_{i+k}) are the matching dominant points in Q_I and (t_j,t_{j+l}) are matched dominant points in T_I, then the orientation satisfies the condition: $|(\alpha_{q_i} - \alpha_{t_j}) - (\alpha_{q_{i+k}} - \alpha_{t_{j+l}})| \leq \theta_{th}$, where $\theta_{th} \geq 0$ is a tolerance level under image rotation at a keypoint p. The rotation (θ) of the image Q is then defined by the difference of orientations of matching keypoints, *i.e.:*

$$rotation(\theta) = (\alpha_{q_i} - \alpha_{t_j}) \tag{5.4}$$

Usually $\theta_{th} \geq \delta/2$, while $\theta_{th} = 0$ is the hard tolerance which assumes that the shape is not deformed.

Stage IV: Upon obtaining the values of translation, rotation and scale of the image Q (w.r.to T), our next task is to transform the positions of all feature points (Q_I, Q_E and Q_C) of the image Q into the space of image T and finally check for a match. This is done as follows:

1. Construct the 4×4 homogeneous matrix H, of the form $\begin{bmatrix} a & b & 0 & 0 \\ c & d & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \delta x & \delta y & 0 & 1 \end{bmatrix}$, to perform the required linear (rotation and scaling, represented by the parameters (a,b,c,d)) and affine (translation $(\delta x, \delta y)$) transforms for all feature points found in image Q.[2]

2. Calculate the modified coordinate positions by matrix multiplication of H with the feature point positions.

3. For each modified q_i ($q_i \varepsilon Q_I$) if there is a t_j ($t_j \varepsilon T_I$) within a tolerance radius of $r \times \varepsilon$, their β-value is then compared. If the β ratio is within β_{th}, then count it as a match.

[2] We use the traditional 3D graphics notation when performing affine transformation using 4×4 matrices; as we are only dealing with a 2D problem, one of the spatial dimensions is redundant, but this is not a problem in practice.

4. Repeat step 3 for *external* dominant and convex points, *i.e.* each element of Q_E and Q_C with T_E and T_C respectively.

Figure 5.6 shows the case where a query (horse silhouette) and a target image (butterfly silhouette) hold completely different shape configurations from each other. Our matching algorithm finds a best possible fitting of the horse (part-based) within the butterfly image by comparing ShIFT feature vectors and looking at their corresponding association. For these two objects, the best fit is found to be the horse's tail with the butterfly's abdomen.

In other words, we can say that when the query and target images represent the same objects then the matching algorithm finds their corresponding affine parameters (rotation, scale and translation). While on the other hand, when the query and target images has dissimilar shapes, then the matching algorithm finds a best possible fitting in between the two.

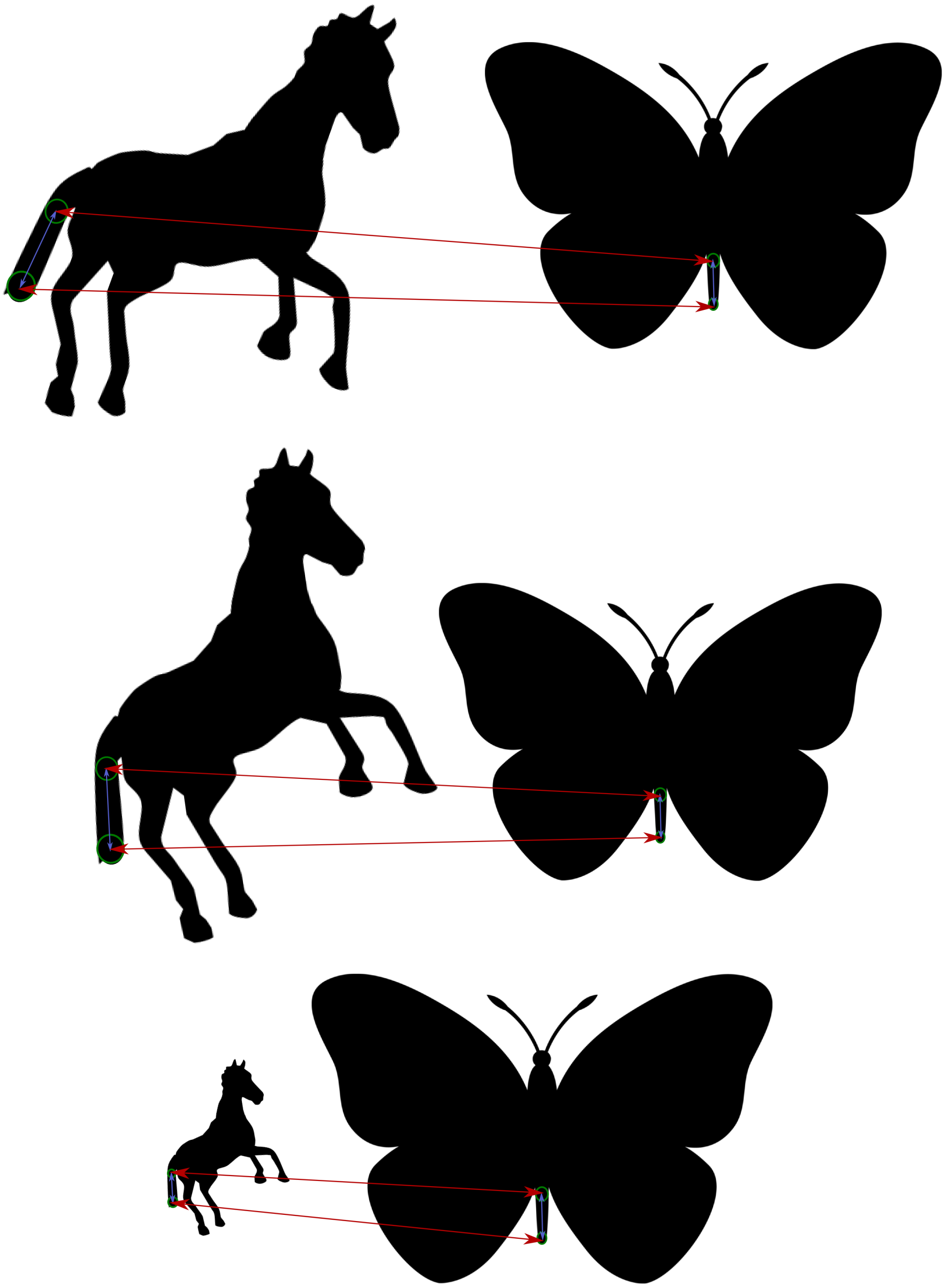


Figure 5.6: An example of homography between two dissimilar objects: the silhouettes of a horse (test-image) and a butterfly (target-image). Here, our matching process finds a best fit of the horse within the butterfly, by considering the part-configuration. These two images are very different from each other and only the horse's tail has a similar shape alike butterfly's abdomen. *Top-row:* finding the best matching parts based on the ShIFT descriptor matching; *middle-row*: rotation of the horse matching the butterfly's abdomen; and *bottom-row*: scaling of the horse with respect to the matched butterfly's abdomen.

5.3 Ranking Metrics

Consider M_I, M_E and M_C as the sets of internally, externally (concave) and convex matching feature points. Intuitively, more shape discrimination is present in internal and external dominant points while convex points add details (end points of protruding parts delimited by external (concave) points). Hence we make use of the following heuristic: our matching metric is biased towards internal and external (concave) dominant points. If $|A|$ is the cardinality of set A, then we express the problem of finding a best matching location of Q in T as the maximization of (although there can be many other ways to combine the information from the three sets of feature points):

- Percentage Match or Precision
- F-measure
- Relative Score

5.3.1 Percentage Match or Precision

Percentage match or precision is calculated as:

$$\%match = \frac{\sum w_{type}|M_{type}|}{\sum w_{type}|Q_{type}|}, \tag{5.5}$$

where $type = \{I, E, C\}$. w_I, w_E and w_C are the weights for internal, external(concave) and convex matching score, where $w_I > (w_E = w_C)$. In this metric, regardless of the number of keypoints in the target image, if all keypoints in the query image find a corresponding match then the value become 100%, i.e. the precision will become 1.[3] This is an example of surjective[4] mapping which can be injective or non-injective. However, this measure

[3] Precision, also known as positive predictive value, is the fraction of retrieved instances that are relevant.

[4] The function is surjective (onto) if every element of the co-domain is mapped to by at least one element of the domain. That is, the image and the co-domain of the function are equal.

does not explain the nature of the target image, i.e., whether the query is a sub-part of the target image or a replica. Such a quantitative measure can be evaluated by recall.[5]

5.3.2 F-measure

Our mathematical formulation of F-measure is:

$$F = 2 \times \frac{\sum w_{type} |M_{type}|}{\sum w_{type} (|Q_{type}| + |T_{type}|)}, \tag{5.6}$$

where $type = \{I, E, C\}$. w_I, w_E and w_C has the same meaning alike in section 5.3.1. This is another measure of a match accuracy, consisting of both precision and recall with a value lying in the interval [0,1]. As equation 5.6 in itself describes, the F-measure is the harmonic mean of precision and recall. This metric produces a value of 1 if and only if Q and T have surjective mapping. This means each keypoint in the query image finds a mapping with one keypoint in the target image and all the keypoints in the target image are mapped. If f is the mapping function, then the F-measure punishes those cases where (a) $f : Q \to T$ is an injective non-surjective mapping, i.e. some elements of T are not mapped (precision = 1 while recall<1), (b) in $f : Q \to T$, all the elements in T have a mapping from Q, but some elements of Q are not mapped, i.e., recall = 1 while precision<1, and (c) in $f : Q \to T$, only some elements of Q are mapped with some elements of T, i.e. precision<1 and recall<1.

5.3.3 Relative Score

The relative score is the combination of two functions, f and g, where f evaluates the score of the mapping $f : Q \to T$, while g evaluates its importance (see equation 5.9 below).

[5] Recall, also known as sensitivity, is the fraction of relevant instances that are retrieved.

Relative to the query image Q, the matching score is expressed as:

$$RS_Q = \sum w_{type} \frac{|M_{type}|}{|Q_{type}|}, \tag{5.7}$$

and, relative to the target image T, the matching score is expressed as:

$$RS_T = \sum w_{type} \frac{|M_{type}|}{|T_{type}|}, \tag{5.8}$$

where $0 < w_{type} < 1$. $RS_Q > RS_T$ means the query image has more overlapping portion than the target image, while the reverse is true for $RS_Q < RS_T$. The overall relative score RS is measured as:

$$RS = f(RS_Q, RS_T) + g(RS_Q, RS_T) \tag{5.9}$$

$f(RS_Q, RS_T)$ is the score for mapping that can be expressed in terms of an arithmetic mean (AM), a geometric mean (GM) or a harmonic mean (HM) of RS_Q and RS_T. The importance function, $g(RS_Q, RS_T)$, is calculated as:

$$g(RS_Q, RS_T) = \min(RS_Q, RS_T)\left(\frac{\max(RS_Q, RS_T) - \min(RS_Q, RS_T)}{2}\right) \tag{5.10}$$

5.4 Classification: Bag-of-Words Model

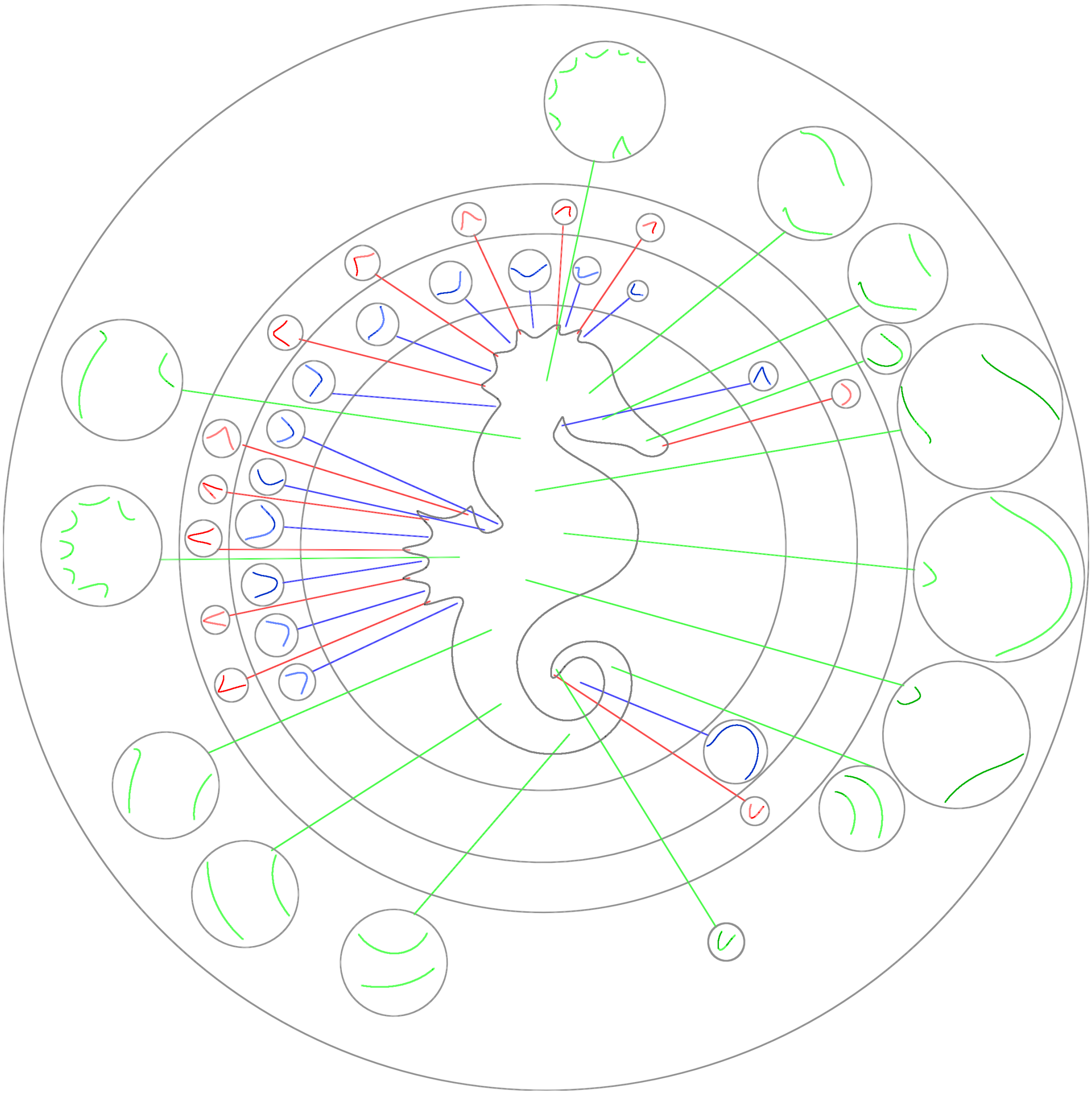


Figure 5.7: Bag-of-words model for a seahorse shown at the central circular disk. Moving in outward direction, concavities (exterior dominant regions in blue), convexities (in red), and interior dominant regions (in green) are shown respectively.

In computer vision, the bag-of-words model (BoW model, Salton and McGill (1983); Han et al. (2011)) also known as bag-of-features model is used for the purpose of image classification such as finding relevant target images out of a trained or indexed database.

To represent an image by the BoW model, the image is treated as a document and the image features are treated as “words”. Usually, there are three steps involved in the process: feature detection, feature description, and codebook generation. The BoW model is nowadays often used in content based image indexing and retrieval (CBIR, Gudivada and Raghavan (1995)).

Feature Representation:

In our mathematical model, features are detected algorithmically by following the steps of medialness measure, top-hat transform and keypoint detection. The shape is then abstracted by several dominant points that can be internal, concave or convex. Our feature representation method ShIFT, deals with the representation of these keypoints as numerical vectors, we refer to as feature descriptors. The designed ShIFT descriptor is able to handle rotation, scale and affine variations. The length of the descriptor (i.e. its dimension) depends on the step-angle δ. After this step, each image is a collection of vectors of same dimension, $2\pi/\delta$. Figure 5.7 illustrates how our bag of features approach maps to elementary visual cues as captured by dominant points. Here, an example of sea-horse is processed and concave, convex and internal elementary parts are shown, surrounding the object for illustrative purpose.

Codebook Generation:

The final step of the BoW model is to convert the descriptors to “codewords” (in analogy to words in text documents), which eventually produces a “codebook” (in analogy to a word dictionary). A codeword can be considered as a representation of several similar descriptors. As mentioned in section 4.3, the B+ tree is used to maintain such a dictionary over all the vectors obtained after training the dataset.

k-Nearest Neighbors Classifier:

K-nearest neighbors (k-NN) is a non-parametric method used for classification in pattern recognition. In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is typically a small positive integer). For example, k=1, then the object is simply assigned to the class of the nearest neighbor. The accuracy of the k-nearest neighbor (k-NN) classification depends significantly on the metric used to compute distances between different examples. In this thesis, we used three different distance metrics: Percentage Match or Precision, standard F-measure, and Relative Score for k-NN classification from an indexed database. In our approach, metrics are trained with the goal that the k-nearest neighbors always belong to the same class while examples from different classes are separated by a large margin. This is among the simplest of all machine leaning algorithms and provides effective results. It can also be useful to assign a weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones.

5.5 Performance Evaluation Methods

The most common performance evaluation method is based on precision and recall:

$$Precision = \frac{t_p}{t_p + f_p} = \frac{|(relevant\, results) \cap (retrieved\, results)|}{|retrieved\, results|}, \tag{5.11}$$

$$Recall = \frac{t_p}{t_p + f_n} = \frac{|(relevant\, results) \cap (retrieved\, results)|}{|relevant\, results|}, \tag{5.12}$$

where, t_p= true positive, f_p= false positive, and f_n= false negative.

For each query image, the system returns a ranked sequence of images. To compare results and methods, an overall measure can be obtained by computing the Average Precision (AP), which is defined in the Information Retrieval literature as the average of precision measurements evaluated at the position of each of the relevant documents in the ranked sequences Manning et al. (2008):

$$AP = \frac{\sum P(r)rel(r)}{I_r}, \tag{5.13}$$

where r is the rank of the current image in the returned image set, N is the number of retrieved images, $rel()$ is a binary function on the relevance of a given rank.[6] $P(r)$ and I_r are respectively the precision and the number of relevant images at the given cut-off rank. Furthermore, to test the performance of the full system, the standard metric known as Mean Average Precision (MAP) can be evaluated for a given set of queries:

$$MAP = \frac{\sum AP(q)}{Q}, \tag{5.14}$$

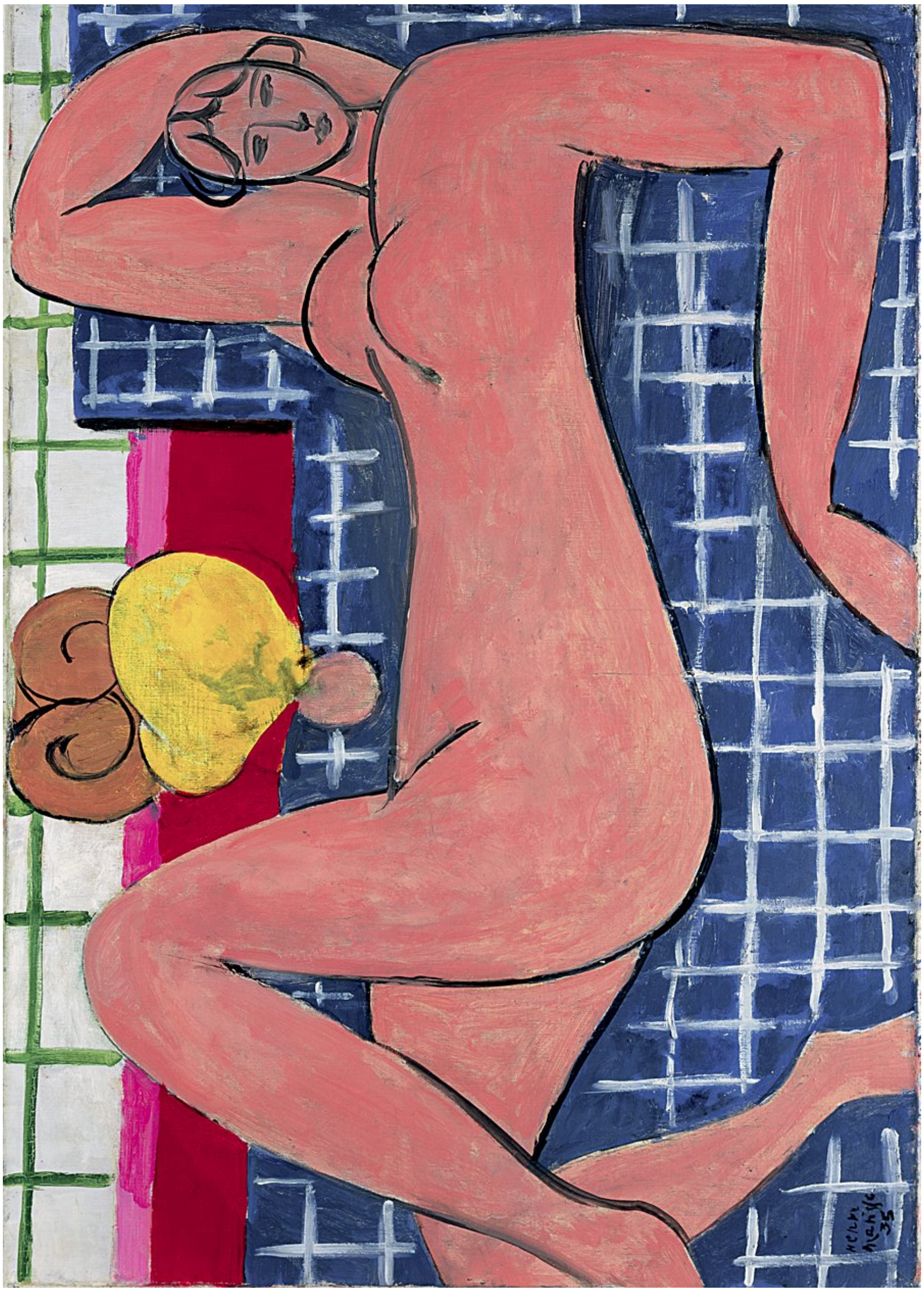
where Q is the number of queries in the benchmark. The system that holds the highest MAP on a benchmark dataset is considered as the state-of-the-art in the area defined by the benchmark.

5.6 Summary

In this chapter we described how we exploited our ShIFT descriptor to perform a spatial matching in the form of an homography. Using information associated with the descriptor, we evaluated different affine parameters and used them for the shape matching task between a query and the target image. To identify the most relevant targets for a given query shape, we referred to the bag-of-words model. Furthermore, we presented three

[6] The binary value of $rel(r)$ is set automatically by parsing a text file which contains the ground truth for each image: *i.e.*, which type it corresponds to, such as a dog, a horse, etc. Thus, the AP measure is only available given ground truth (identification) is provided.

discrete ranking metrics, used for scoring a particular matching task. From this model we can then pick the top N targets on which a homography is performed. We need a parameter that can evaluate the performance of the whole system which can then be used for comparing the performances. For this purpose we used a standard evaluation metric Mean Average Precision (MAP). In the following chapter, we will see the behavior of our system under these metrics.



Henry Matisse, *Large Reclining Nude (The Pink Nude)*, 1935. @The Baltimore Museum of Art.

Chapter 6

Results and Discussion

> *"My success will not depend on what A or B thinks of me. My success will be what I make of my work."*
>
> *– Homi Jehangir Bhabha, Nuclear Physicist (1909–1966)*

An attractive broad view of computer vision is that it is an inference problem: we have a model and some measurements, and we wish to determine what caused the measurement. In the past couple of chapters of this thesis have been dedicated for the quantitative analysis of the shape via psychophysically driven medialness-based description. Furthermore, distinct feature point sets are derived from such a description and then mapped to a feature vector. To deal with large or overgrowing datasets, those feature vectors are converted into affine invariant descriptors, we refer to as the Shape Invariant Feature Transform (ShIFT), which can be used for indexing purpose. We further discuss retrieval method and homography in detail, and each query image produced ranked results based on the ranking metric used. Our current matching algorithm is hierarchical in its use of feature points: internal (medial hot spots) –> external (concave tips) –> internal (convex tips), and has proven effective even with articulated body movements (Leymarie et al. (2014b); Aparajeya and Leymarie (2015)).

This chapter poses object recognition as a correspondence problem – which shape feature corresponds to which feature on which object in the trained shape dataset? This simple view of recognition focuses on the relationship among object/shape features and image features.

6.1 Shape Databases and Performance Evaluation

During my research program I have performed extensive experiments on different heterogeneous databases containing biological forms, including large animals, plants, and insects, to verify the performance in efficiency and accuracy of the devised method.[1] Now I detail the databases along with their experimental setups for the result analysis.

6.1.1 Result analysis with feature-based methods

6.1.1.1 Our (GOLD) Shape Database

Our GOLD shape database mainly consists of :

(i) animals taking a static posture and in movement (see Figure 6.1),

(ii) humans taking a static posture or in action, i.e., articulated movements (see Figure 6.2),

(iii) insects, e.g., bugs, butterflies (see Figure 6.3),

(iv) plant leaves (see Figure 6.4).

In order to verify the robustness of the designed algorithm under some random rescaling, rotation and translation have been performed on this collected dataset. On top of this, a number of occlusions have been added performed through random cuts, to test the method's response beyond affine transforms. Furthermore, to evaluate the robustness of

[1] The first results on such data were presented in a paper at the 1st International Workshop on "Environmental Multimedia Retrieval" (EMR) held in conjunction with the ACM International Conference on Multimedia Retrieval (ICMR) in Glasgow (UK), April 1, 2014 Aparajeya and Leymarie (2014).

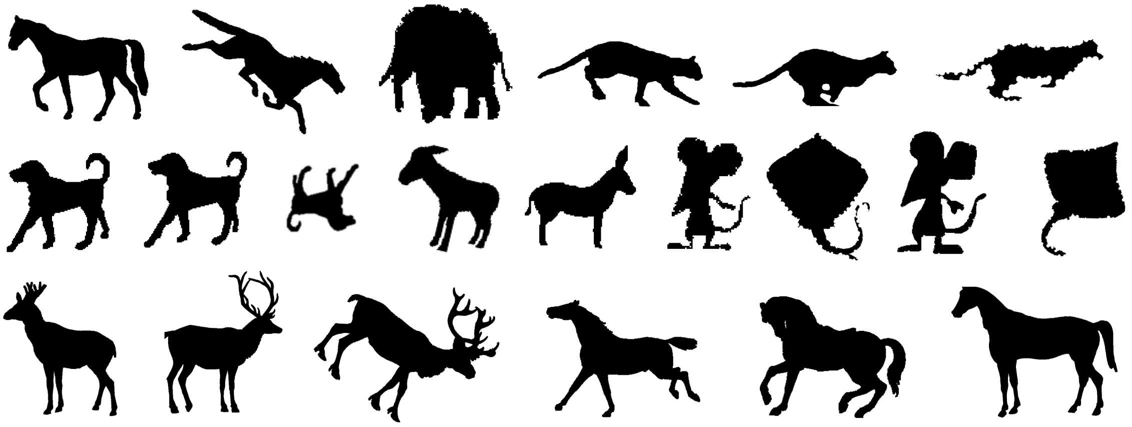


Figure 6.1: Samples from our GOLD database: Here animals are taking a static posture or they are in movement.

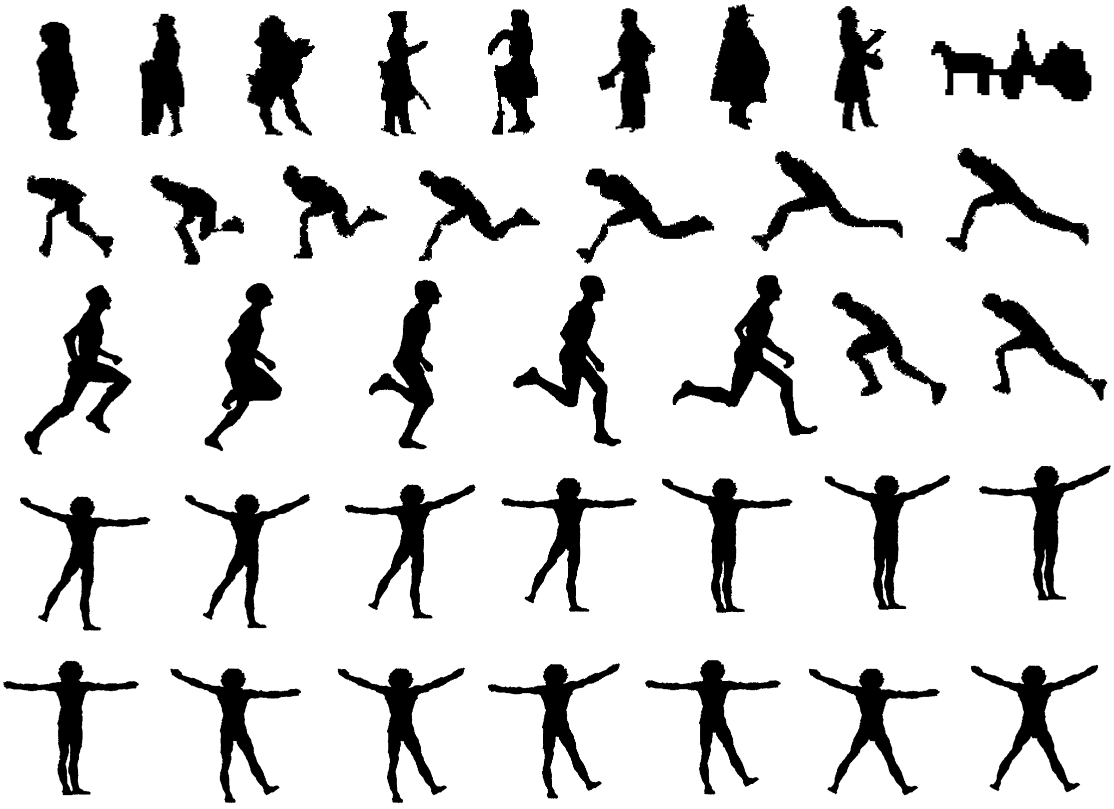


Figure 6.2: Samples from our GOLD database: Here humans are taking a static posture or they are in movement.

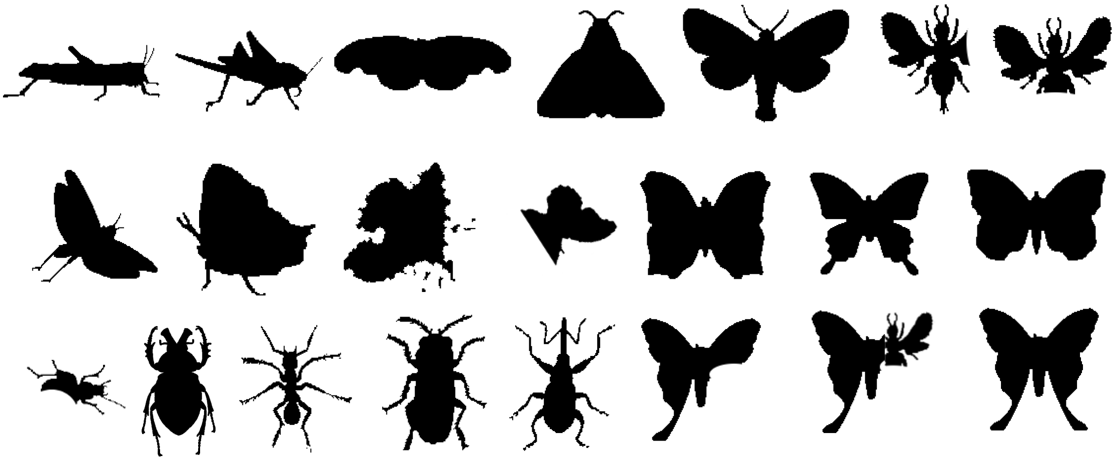


Figure 6.3: Samples from our GOLD database: Insects.

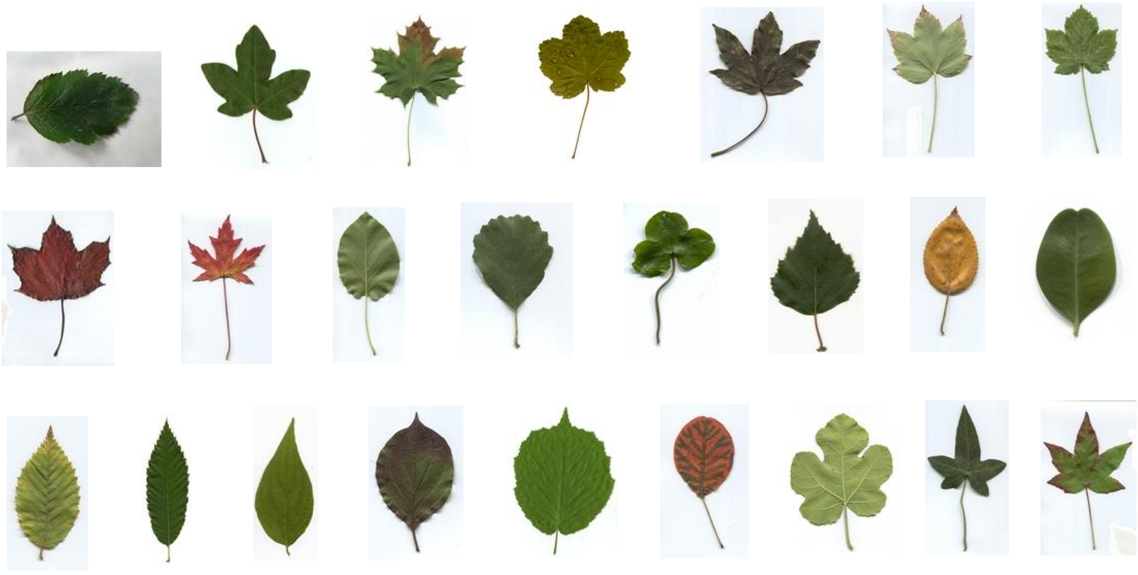


Figure 6.4: Samples from our GOLD database: Plant Leaves.

the algorithm, some structural noise is added, by introducing scalable random geometric deformations, designed by performing randomised morphological set operations on the binarised shapes/object. In this experiment, three levels of such structural perturbations: small or less perturbed, medium and large or highly perturbed are defined, Figure 6.5. We note that other methods relying on smooth continuous contours, such as methods based on the use of codons or curvature scale-space, as well as many of the traditional medial-axis methods, will have great difficulty in dealing with such deformations — which are to be expected in noisy image captures and under varying environmental conditions such as due to decay and erosion.[2]

To construct such shape database we used the standard MPEG-7 (Latecki et al. (2000); Latecki and Lakamper (2000)), ImageCLEF-2013 (Caputo et al. (2013)) and Kimia (Brown University's) (Sebastian et al. (2003, 2004)) datasets. Furthermore, we also initiated our own database where we selected different sequences of animals in motion from videos. From these datasets, we have collected a total of 2215 samples belonging to Animals other than insects (650 samples, including Human, Horse, Rat, Cat, Panther, Turtle, Elephant, Bat, Deer, Dog, and Ray forms), Insects (410 samples, including Butterfly, Bug, Mosquito, Ant, and other miscellaneous insect forms), and Plant leaves (1155 samples, including Acercampestre, Aceropalus, Acerplatanoides, Acerpseudoplatanus, Acersaccharinum, Anemonehepatica, Ficuscarica, Hederahelix, Liquidambarstyraciflua, Liriodendrontulipifer, Populusalba specimens). Furthermore, to check the robustness of our algorithm, we deformed each such sample at the previously indicated three levels of perturbation, thus bringing our total dataset count to $2215 \times 4 = 8860$.

We note that we are limiting the sizes of our test databases as we require segmented binary forms (distinguishing figure from ground) to initiate our medialness transform (*e.g.* ImageCLEF includes circa 5000 plant images, among which only 1155 contain clearly

[2] We do not claim that this way of perturbing the data is physically accurate in modeling natural decay or erosion. Rather it provides a simple (computational) way to approximate such effects and produce deformations which visually appear credible in modeling these.

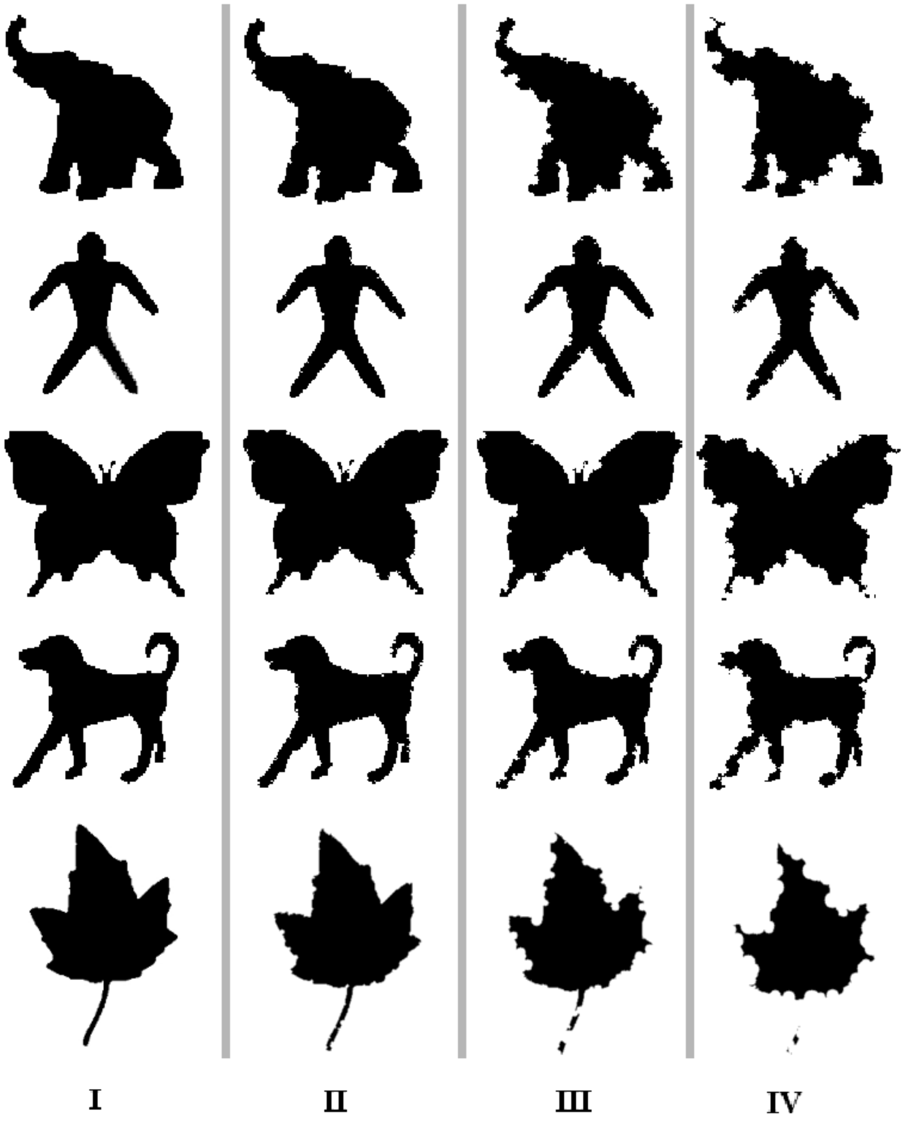


Figure 6.5: Three levels of (added) perturbation: I - none (originals), II - small , III - medium and IV - large.

distinguishable leaf forms). Also, rather than focus on one type of biological forms, say butterflies, we decided to test and show the potential power of our approach for a number of very different biological forms, from plant leaves, to various species of insects to larger animals, including humans.

To check the performance of our method on this dataset, we exploited our current ranking metric F-measure (see equation 5.6) to find the returned top-10 matches. Examples of such top-10 results can be seen in Figures 6.6, 6.7, 6.8, and 6.9.

First, in Figure 6.6 we notice the ranking follows a gradual degradation in spatial overlays of query and target, but where the first 10 forms are all of the right type. We also notice a graceful degradation in ranking when dealing with articulated planar movements of a human and horse figures.

Second, in Figure 6.7 we study the use of our current matching method on plant leaf queries and also obtain a graceful degradation in rankings with a similar behavior in spatial form overlaps. We note that our feature points can be used to support a higher level semantic description of leafs for better taxonomic characterisation (Cope et al. (2012)), in particular by permitting to identify: the apex, main lobes, and petiole using convex hot spots. The insertion point (where the petiole reaches the leaf) can be characterised by a codon: pair of bounding (significant) concavities together with the intermediate main convexity (denoting the petiole). The apex and lobes extent and relative size can be obtained by finding their associated codon information (apex and lobes tips as convexities associated with nearby concavities). If teeth at the border of a leaf are detectable, with sufficient resolution in the input image, then we can select a related scale of analysis, *i.e.* setting the annular gauge with corresponding parameters (R, ε), once the other previous main features have been identified. It would then be of interest to study the relation between the medial hot spots and main vein layout; the current approach in the literature is to directly process the textured input images to either segment out veins and then char-

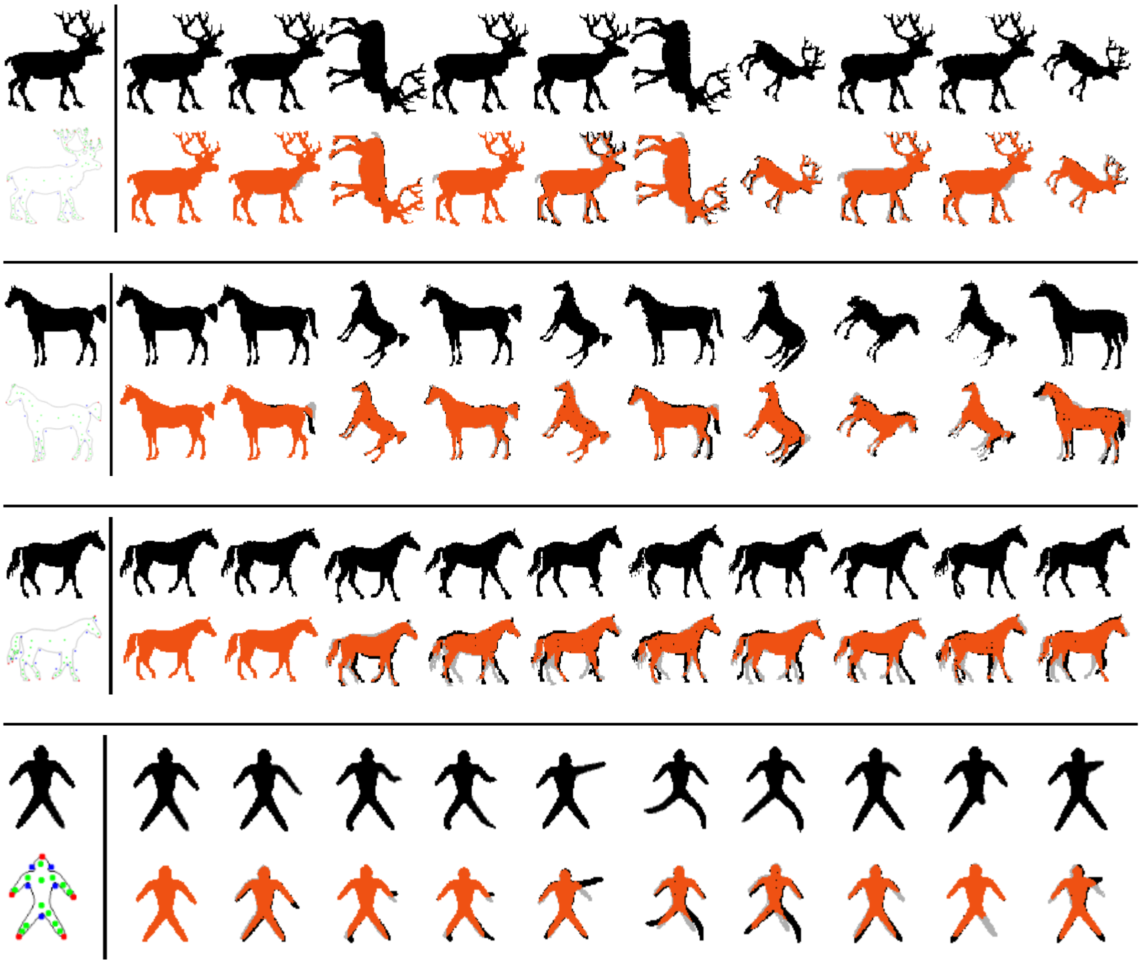


Figure 6.6: Top-10 results on some of the collected samples from our extended database (containing 8860 forms): for deer, horse and humanoid forms when using our $F - measure$ as the basis for ranking. The leftmost image for each ranking results is the query, while the images on the top row are the target images matched at successive rank location from best to worst. The bottom row then shows the overlaying (in orange) of the query on the respective target image (after transformation) and their spatial differences (in grey). The top two series show tests for the invariance under scaling, rotation and translation. The third and fourth series show the behavior of matching using the $F - measure$ in the presence of articulated movements.

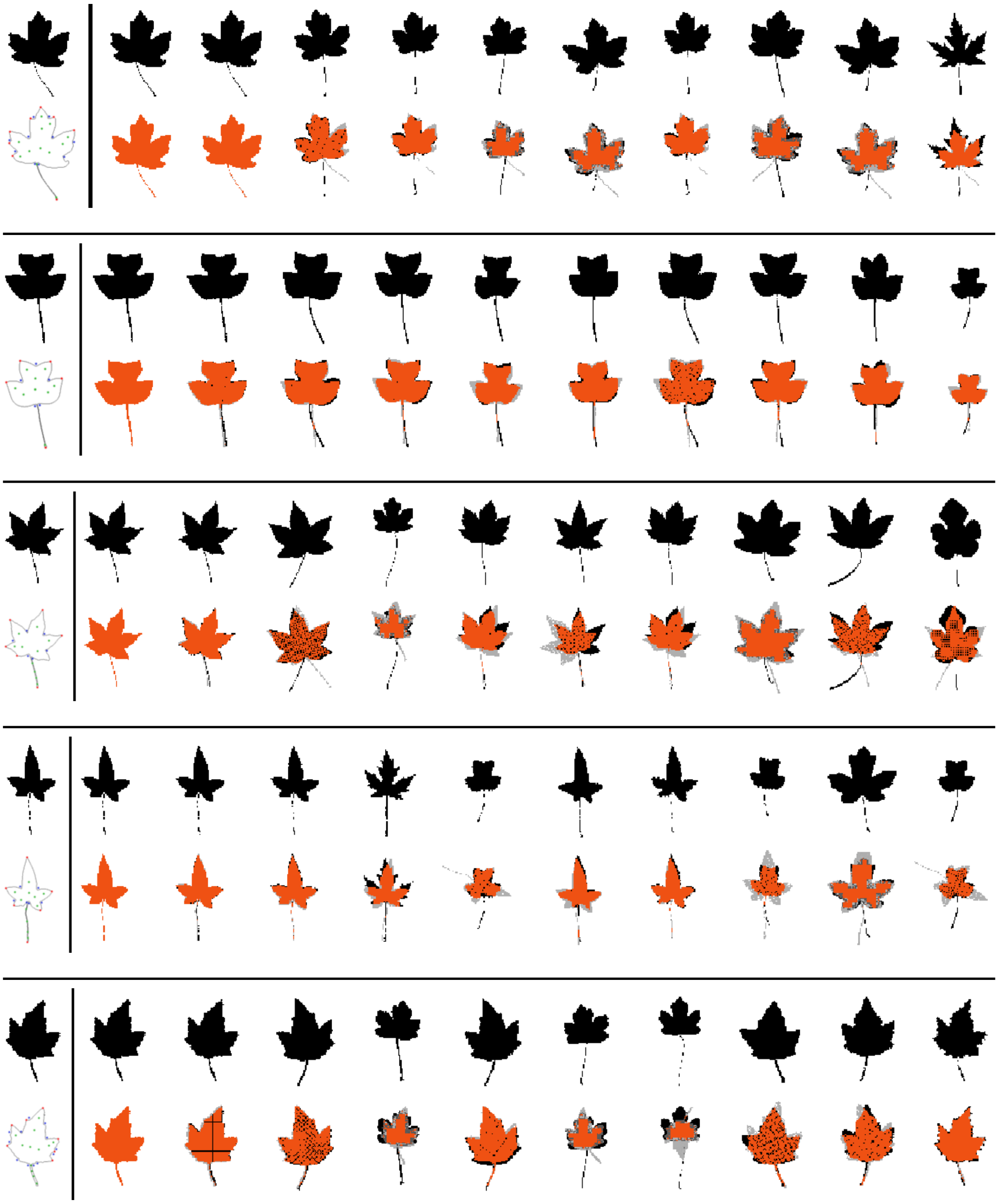


Figure 6.7: Top-10 results on some of the samples from the *Plant* database (ImageCLEF-2013, Caputo et al. (2013)) *with shape perturbations* using our $F-measure$ as the basis for ranking. NB: The first match is always the desired target.

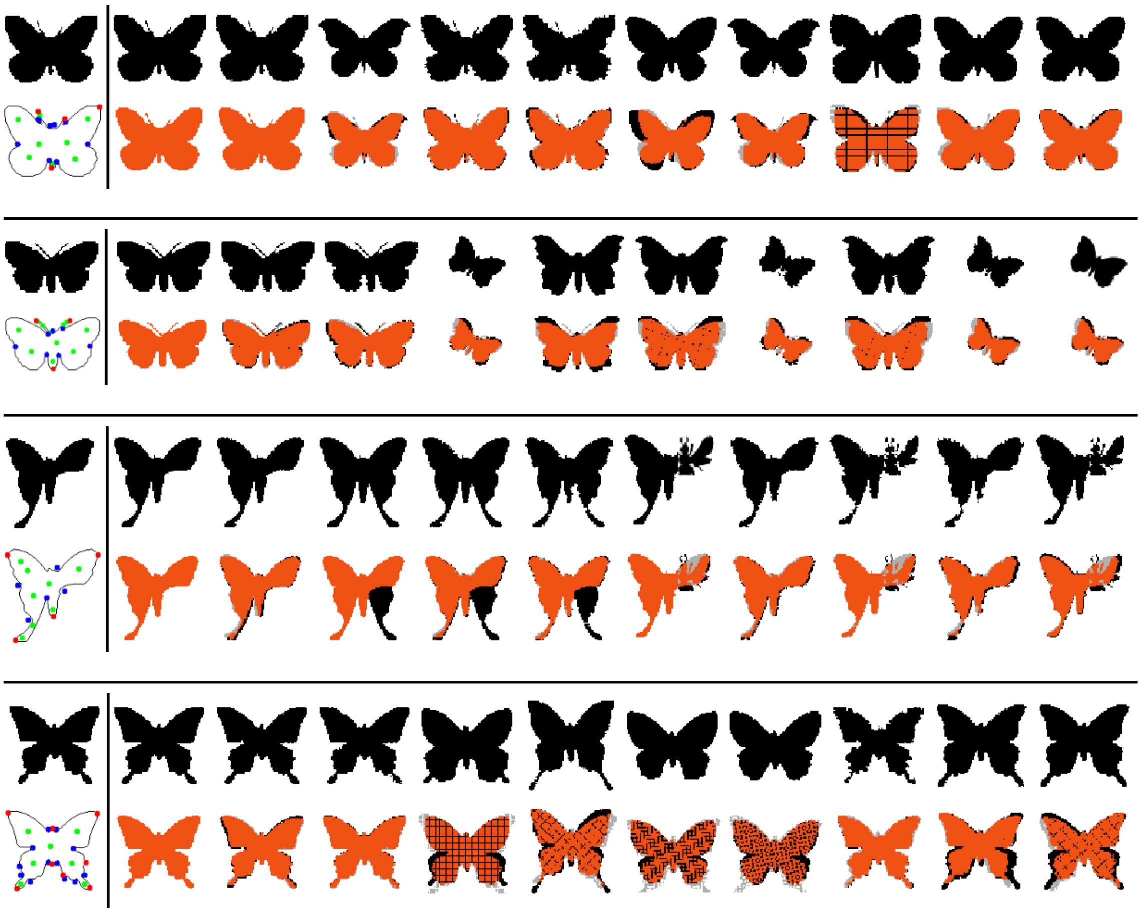


Figure 6.8: Top-10 results on some of the samples for *butterfly* forms from the *Insect* database including structural perturbations when using our $F - measure$ as the basis for ranking. NB: A partial shape query (3rd series from the top) returns valid and interesting results.

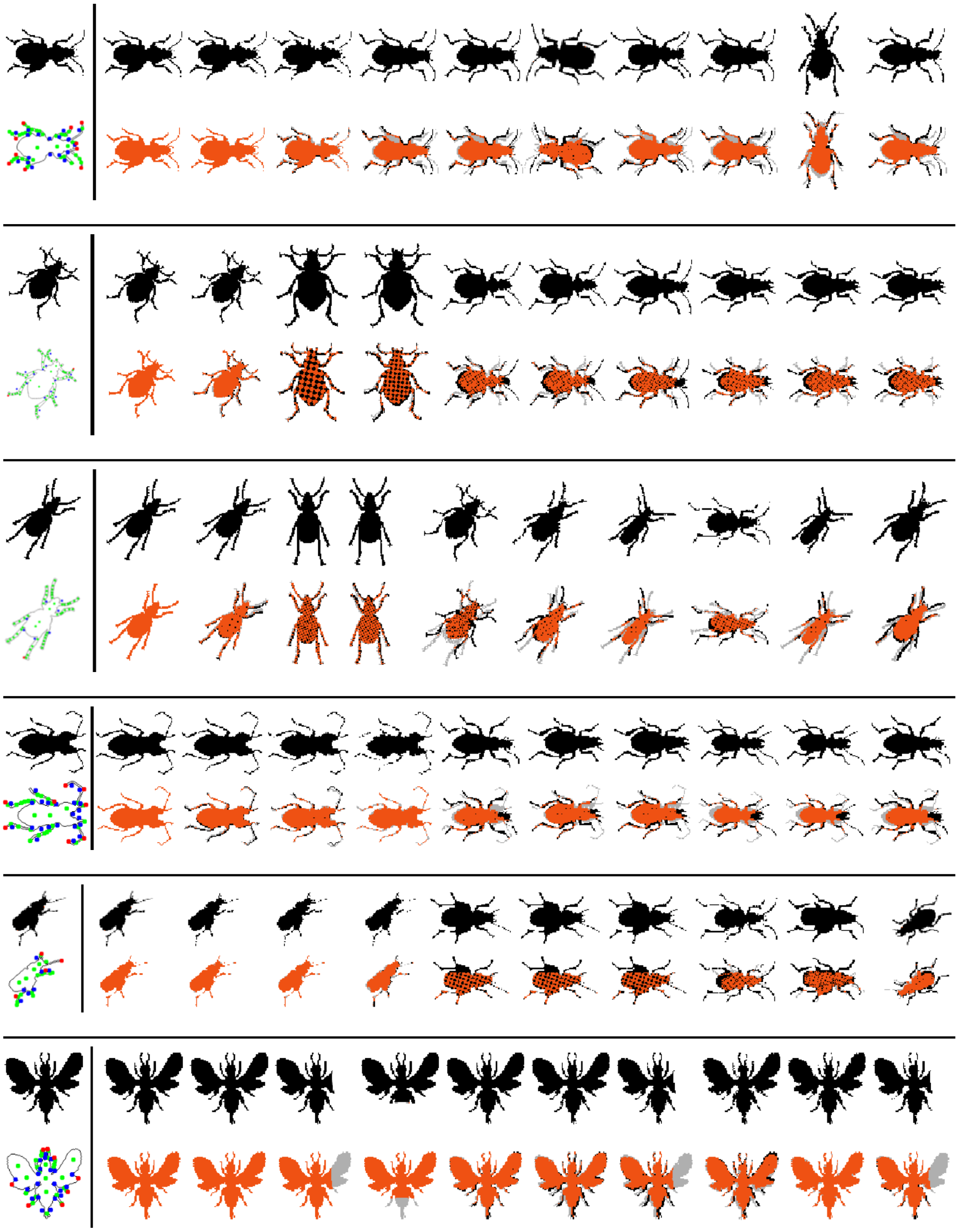


Figure 6.9: Top-10 results on some of the samples from the *Insect* database including structural perturbations when using our $F - measure$ as the basis for ranking.

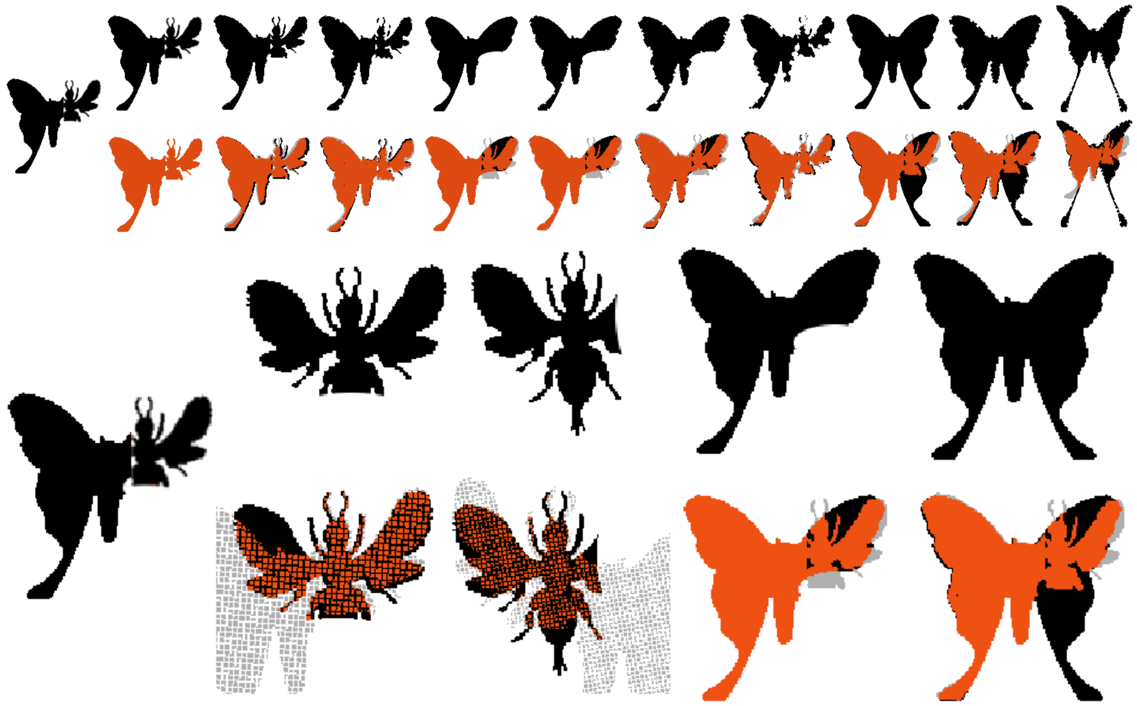


Figure 6.10: A special query image obtained as a juxtaposition of two different insect cuts finds a best fitting location in the target image, and results into an interesting series of part-based matches (eventually retrieving the correct pair of individual insects (before juxtaposition) from the DB). The top series shows as usual the top 10 matches. The bottom series shows the interesting matches where the individual original insects are found (ranked 12th and 16th (for the smaller insect) and 4th and 8th for the larger butterfly.

acterise their networks (Park et al. (2008)) or by directly extracting approximate feature points, *e.g.* using the Harris (corner) detector (Mouine et al. (2012)). By simple visual inspection, some leafs with strong well delimited lobes and/or teeth with clear associated convexity tips may see their dominant veins well approximated by medial hot spots; but this is clearly not the case for many other leaf types with rather smooth outlines (Larese et al. (2014)).

Third, in Figures 6.8, 6.9 and 6.10 we show some results with queries from insects forms. In Figure 6.8 we study the use of our current matching method on butterfly queries and again obtain a graceful degradation in rankings with a similar behavior in spatial form overlaps as for leaves and mammals. Note that the 3rd series indicated in that figure is for a butterfly with a large part cut away: we either get top retrievals for butterflies with similar cuts or complete butterflies which represent likely completed forms (including the correct original butterfly, before being cut, ranked no. 3). Good matches are also found for an artificially constructed case of fusing the cut form with another smaller insect (Figure 6.10). Figure 6.9 gives more evidence for the quality of retrieved matches on other insect forms. Figure 6.10 illustrates a special case where we fabricated an artificial query by merging together two different insects with large cuts to their original form. The returned top matches are interesting as they include the original two individual insects with varying degrees of cuts.

In these experiments, the matching algorithm always finds the best fitting shape area for the query in the target image (Figure 6.10). For empirical analysis, we performed two individual comparisons: (a) precision obtained on different sets of data types (Figure 6.11), and (b) precision obtained at different perturbation levels (Figure 6.12). When the query image belongs to the original set, the retrieval rate at different ranks is very high, while the performance significantly decreases for high levels of perturbations. Note however that even under a large structural perturbation, a given form which may have lost

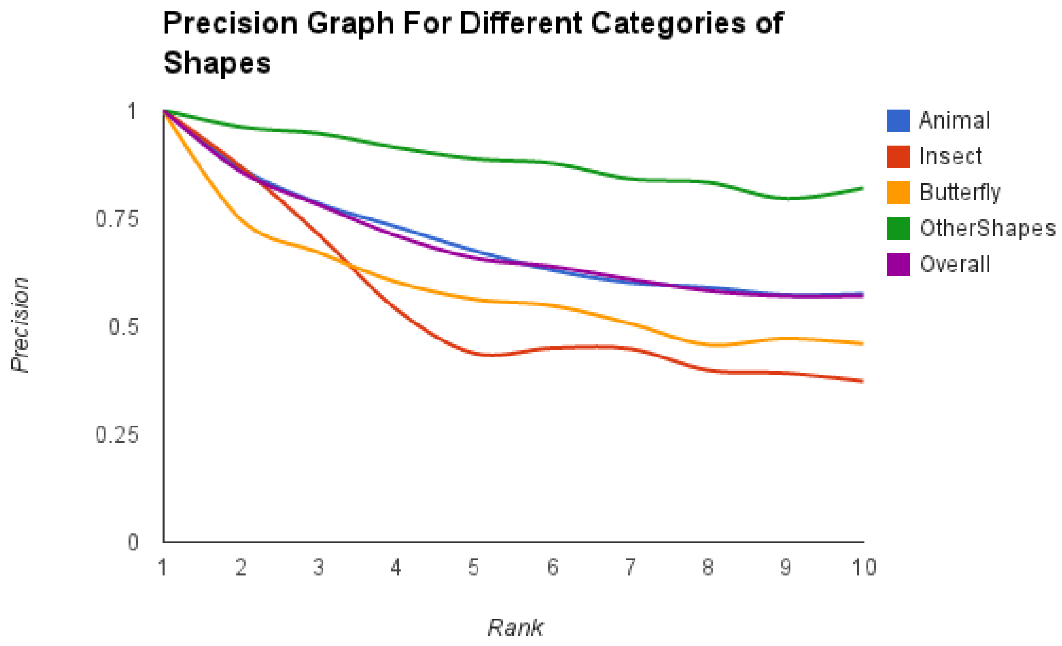


Figure 6.11: Result analysis (precision) on the different datasets containing originals of large animal, plant and insect images, using our proposed $F - measure$ as the ranking metric.

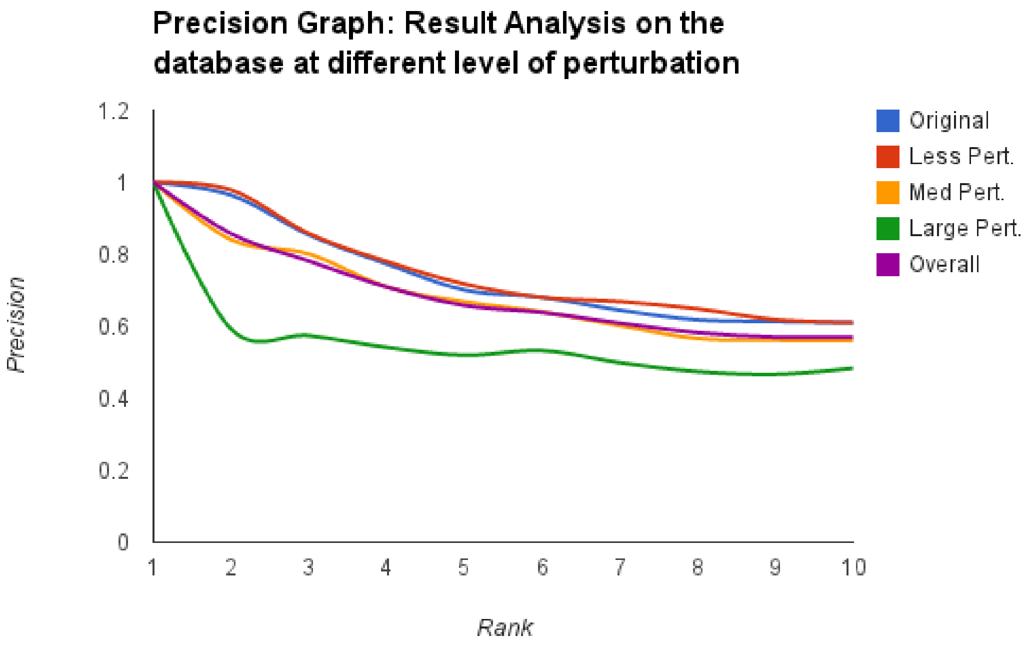


Figure 6.12: Result analysis of the entire dataset for different levels of perturbations, using our $F-measure$ as the ranking metric. NB: Many of the images are also randomly rotated, scaled and translated.

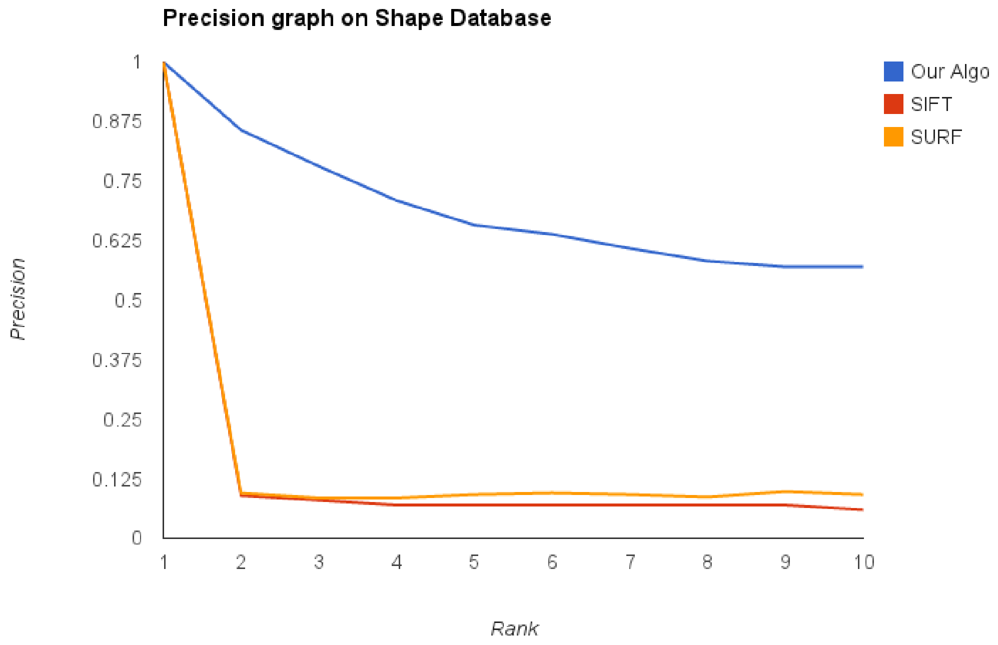


Figure 6.13: Comparative result analysis (precision) of the entire dataset at different ranking locations, using our shape descriptor with $F - measure$ as the ranking metric w.r.to SIFT (Lowe (2004)) and SURF (Bay et al. (2008)). NB: Many of the images are also randomly rotated, scaled and translated.

some of its significant shape features (*e.g.* a limb) can still be matched with perceptually similar targets — as judged by a human observer and validated here as we know the ground truth.

Figure 6.13 shows a comparative result analysis by computing precision at different ranking locations with respect to two of the currently most popular image matching methods: SIFT (Lowe (2004)) and SURF (Bay et al. (2008)), which are proven to be good at retrieval from large datasets as their features are indexable and hence they are scalable. The image features used by these methods (such as edge or corner responses) prove not particularly good on their own at capturing the shape structure. As a consequence, their accuracy reduces drastically when there is a slight deformation, occlusion and/or perturbation in the same shape. On our test database, already from the 2nd ranked match, SIFT and SURF both show a rapid reduction in accuracy (to circa 10%) in detecting a relevant target shape. While, on the other hand, our shape descriptor shows a slow fall in overall accuracy — from 85.7% at 2nd rank to 57% at 10th rank (Figure 6.13). We compare our method with SIFT and SURF as they represent the dominant vector-based indexing schemes used in computer vision. These methods were however developed mainly to deal with highly textured images rather that focusing on scenes where the shape of objects varies mainly in their contour features. As most (if not all) shape-based methods do not provide efficient indexing schemes, we wanted to compare ours when processing video streams with other efficient schemes such as SIFT and SURF. In the future, we would aim to further develop our implementations and compare to more recent object recognition schemes, such as based on advances made with Neural Networks (NN) methods, in particular when based on Convolutional NN (or CNN) schemes applicable to image and video data.

For each query image, the system returns a ranked sequence of images. By computing the Average Precision (AP) we can obtain an overall measure to compare results and methods. The obtained comparative *MAP* graph at different ranking locations on our dataset is shown in Figure 6.14. MAP values for our method stay well above the 50% mark even for 10-th place in rankings, and decrease smoothly from 100% for 1st place rankings. The SIFT and SURF methods on the other hand show poor MAP results with the values 11% (SIFT) and 13% (SURF) for 10th place in ranking.

6.1.1.2 Simpsons’ Video Frames Database

We have used different Simpson’s video frames to populate our second shape database, where the Simpson family – Bart, Homer, Lisa, Maggie and Marge Simpson has a particular yellow-color and shape (Figure 6.15). Color based segmentation is used to extract their facial shape, but in conjunction we also obtain other objects as well which have sim-

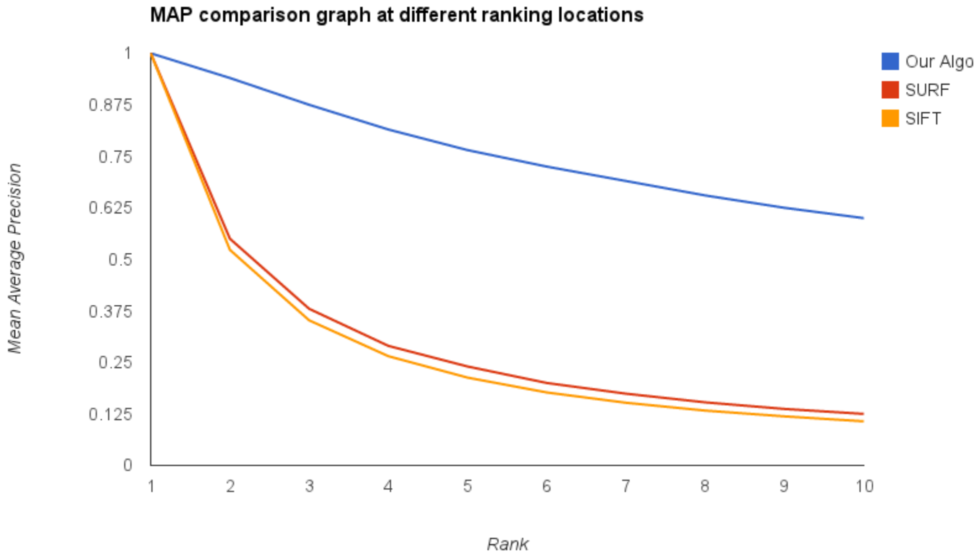


Figure 6.14: Result analysis of the entire dataset by evaluating Mean Average Precision (MAP) at different ranking locations, using our $F - measure$ as the ranking metric with SIFTLowe (2004) and SURFBay et al. (2008). NB: Many of the images are also randomly rotated, scaled, translated and perturbed.

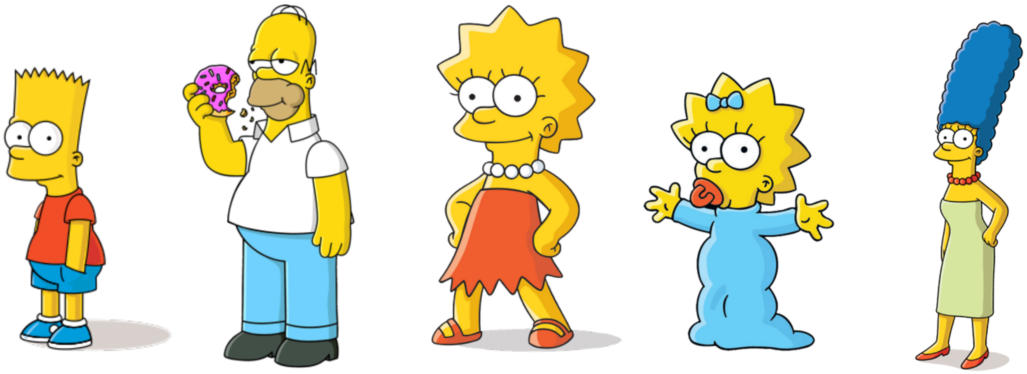
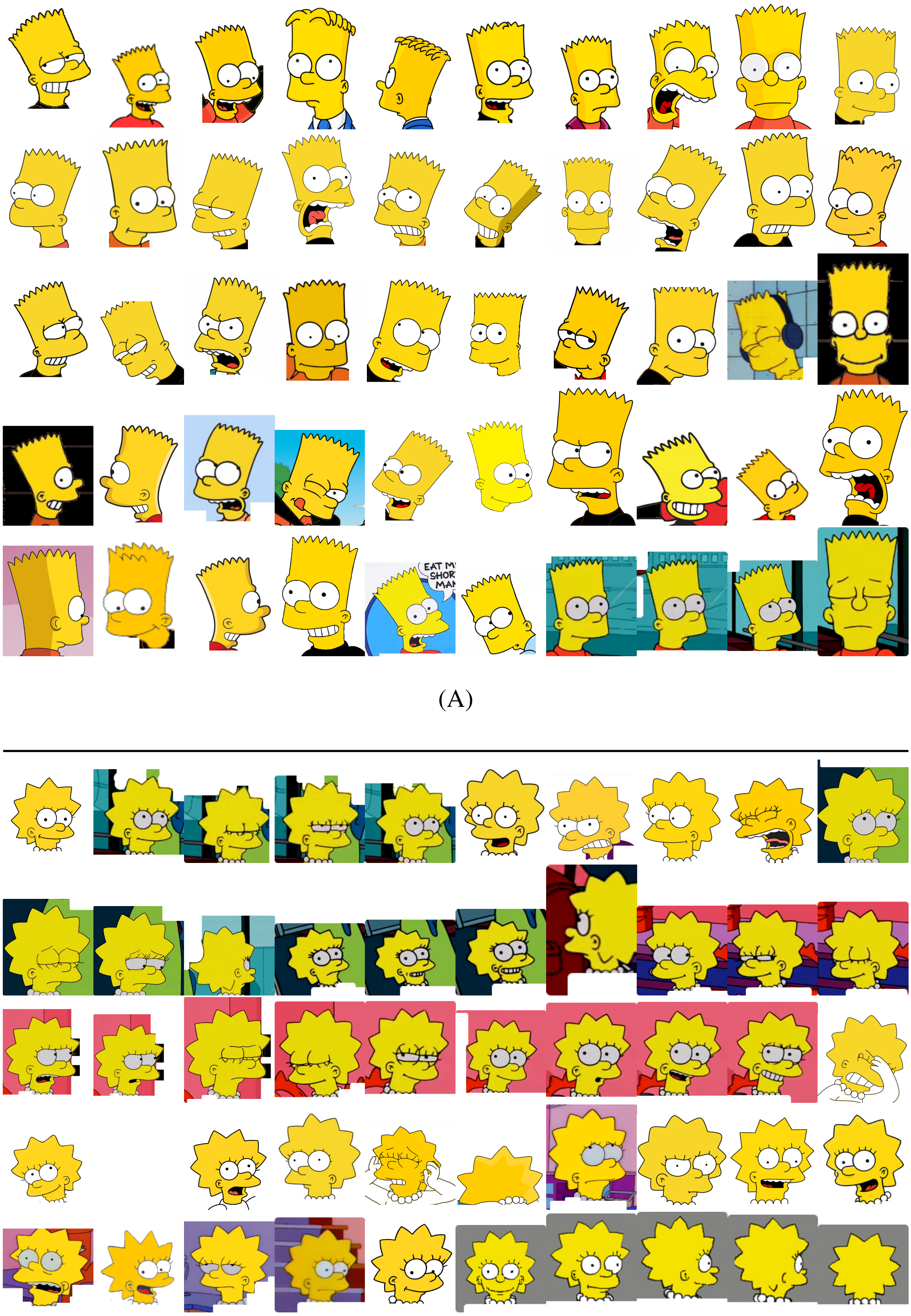


Figure 6.15: Simpson's Family Characters, from left to right: Bart Simpson, Homer Simpson, Lisa Simpson, Maggie Simpson, and Marge Simpson. Source and Copyright: Twentieth Century Fox Film Corporation.

ilar yellow-color. As a result, sometimes we get occluded and cluttered frames. From the videos, we have created a ground truth of 15K video frames (each frame of size 854×480 containing Bart-22.13%, Homer-51.21%, Lisa-19.20%, Maggie-13.82%, and Marge-28.48%). These cartoon videos have many redundant frames since the scene does not change within few milli-seconds (or sometime couple of seconds) of frames. If a video was recorded at 24fps and the scene is not changing for 2s then 48 frames are redundant. To create fast ground truth, we have created a semi-automatic tool to label all five characters at their respective position in different frames and provide textual as well as visual labeling of them.



(A)

(B)

Figure 6.16: (A) Bart, and (B) Lisa Simpson's Templates

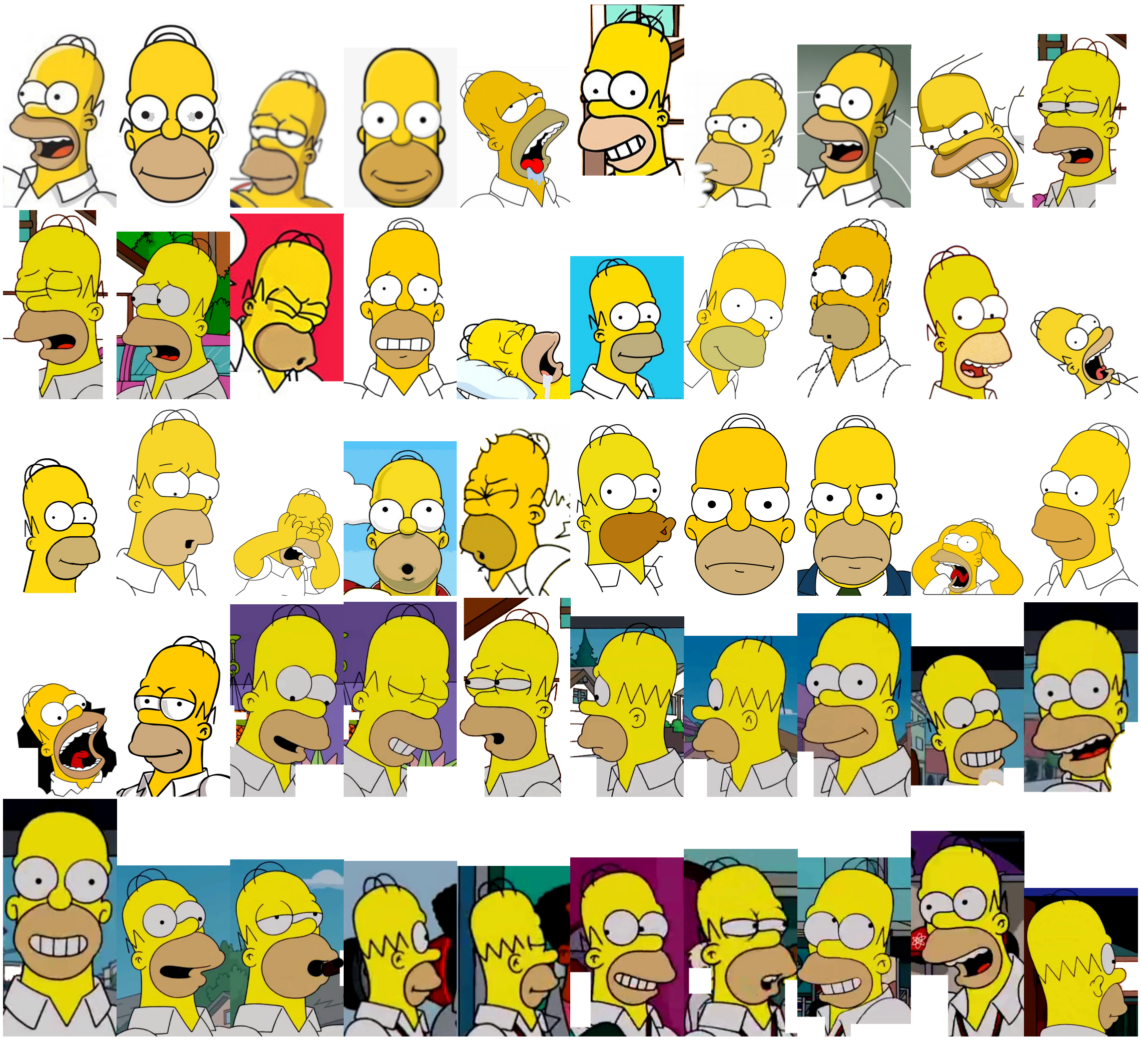
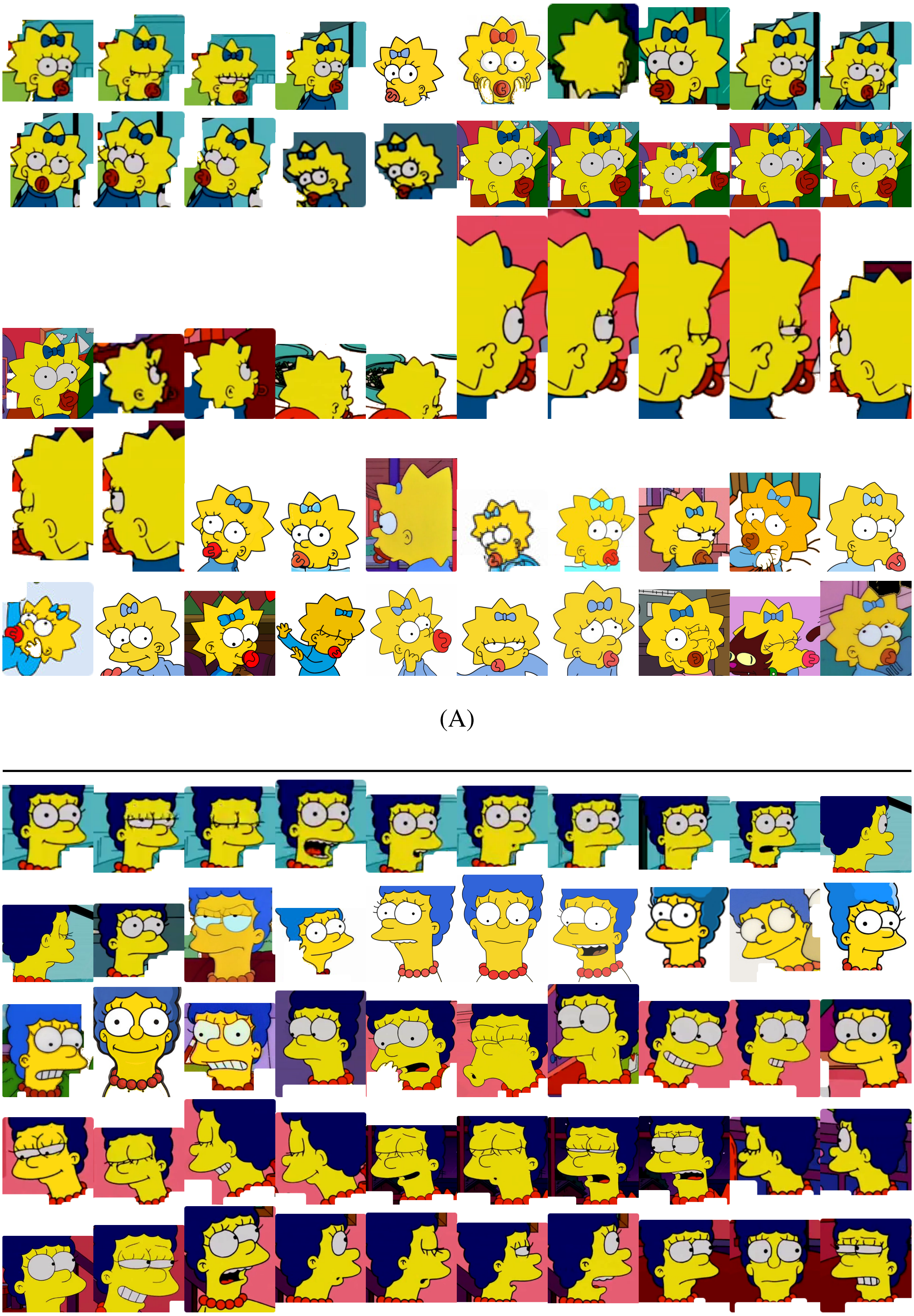


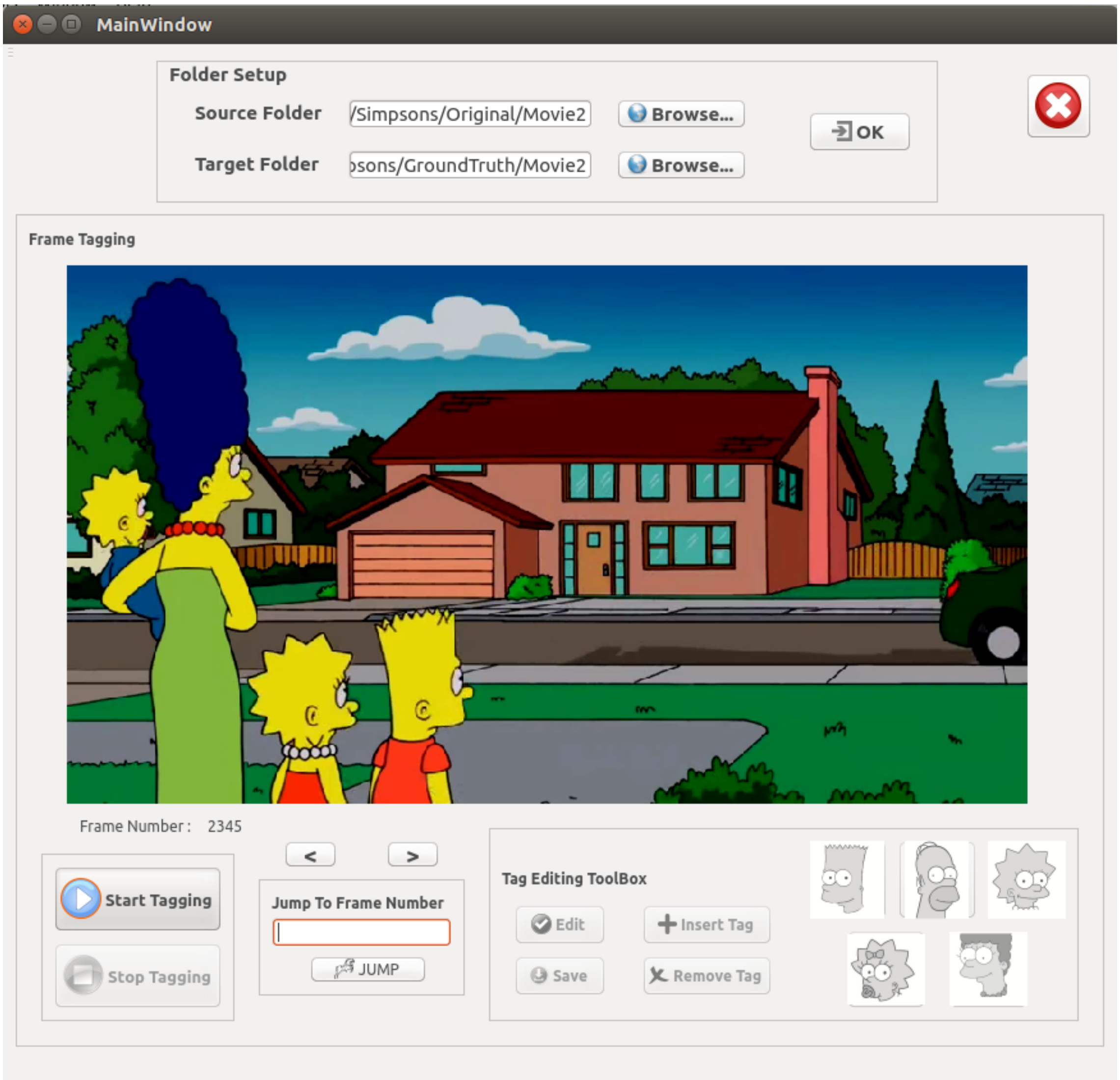
Figure 6.17: Homer Simpson's Templates



(A)

(B)

Figure 6.18: (A) Maggie, and (B) Marge Simpson’s Templates

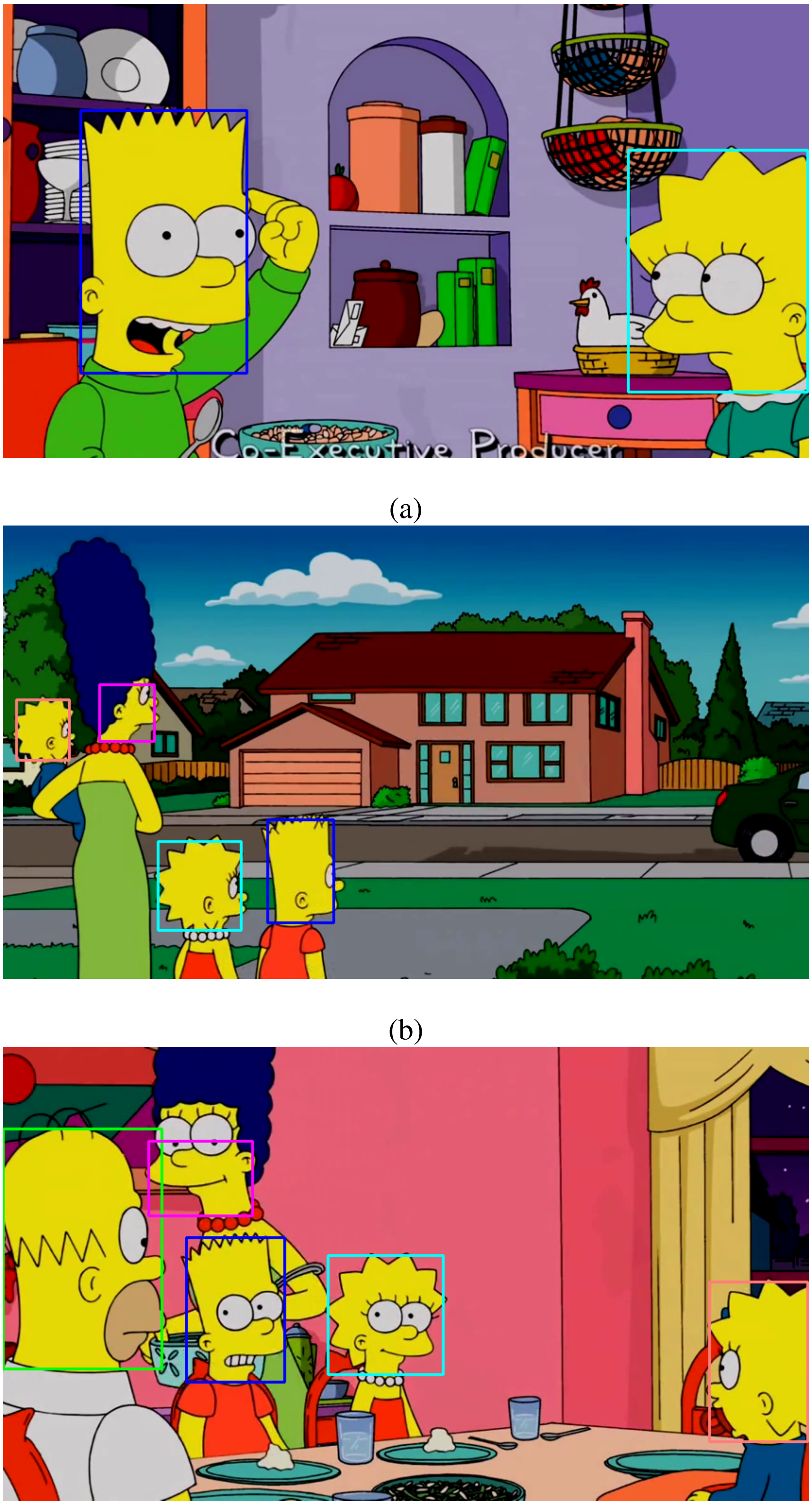


Source Folder /Simpsons/Original/Movie2

Target Folder ›sons/GroundTruth/Movie2

Frame Number : 2345

Figure 6.19: A semi-automatic tool for tagging Simpson's characters in the video.



(a)

(b)

(c)

Figure 6.20: Tagging of Simpson's character (Bart within Blue, Homer in Green, Lisa in Cyan, Maggie in Orange, and Marge in Pink colored rectangular box) with our semi-automatic tagging tool.

The structure of created meta-data file has information in this template form: $< id, is, MinX, MinY, MaxX, MaxY >$, where id is the character identity (0 – Bart, 1 – Homer, 2 – Lisa, 3 – Maggie, and 4 – Marge), is holds a binary value representing whether the character is present in the scene or not, $\{MinX, MinY\}$ and $\{MaxX, MaxY\}$ are the top-left and bottom-right co-ordinates of the rectangle respectively. If a particular character is not in the scene, i.e. $is = 0$, then the rectangle holds the co-ordinates $\{-1, -1, -1, -1\}$. For example, for the frames shown in Figure 6.20 (a), (b), & (c), the respective meta-data are:

(a)

```
0 1  82 112 181 207
1 0  -1  -1  -1  -1
2 1 662 154 748 264
3 0  -1  -1  -1  -1
4 0  -1  -1  -1  -1
```

(b)

```
0 1 280 311 315 370
1 0  -1  -1  -1  -1
2 1 164 334 217 386
3 0  14 184  45 216
4 0 102 168 142 199
```

(c)

```
0 1 194 201 264 245
1 0  0  86  78 165
2 1 344 220 411 245
3 0 748 248 828 329
4 0 154  99 241 129
```

To evaluate our method on the created dataset a set of query images are selected. In this set, each of the five characters has 50 unique shapes related to their different facial orientation and/or expressions, Figures 6.16(A) & (B), 6.17, and 6.18(A) & (B). Out of this dataset, 42% belong to the dataset itself (Bart- 20%, Homer- 36%, Lisa-50%, Maggie-56%, Marge-48%), while the rest are either part of different video frames, i.e., not inside the training dataset of facial drawings. The task then boils down to either find those frames where a character has a very similar shape or at-least the same character is present in the frame.

Our retrieval process takes the advantage of the bag-of-word model (Salton and McGill (1983); Han et al. (2011)). For a query image Q, first all three sets of features are retrieved, and their frequency are counted on the basis of the ShIFT descriptor. For each retrieved feature vector in query, all documents are listed from the indexed database that contains the same feature. As a result, three sets of lists are created at the end. To rank those documents in order, the next task is to assign them a score. Here, we use a query-driven scoring system that aims to provide high scoring in those cases where a maximum matching value is obtained. Initiating the score of documents with 0, each time when a feature vector p in query finds a document D_i with frequency k in database, the score $S_{p,D_i,type}$ is incremented by:

$$S_{p,D_i,type} = \frac{min(f_p^{type}, k)}{\#Keypoints\,In\,Query\,Image} \tag{6.1}$$

Table 6.1: Metric types, their nature and use.

Type	Equation	Nature	Usage
Metric 1 (Mean Value)	$\frac{S_I^{D_i}+S_E^{D_i}+S_C^{D_i}}{3}$	When all the three types of dominant points have equal importance in defining a shape	Shape boundary is smooth or contain minimal boundary noise.
Metric 2 (H. M.)	$\frac{3}{\Sigma\frac{1}{S^{D_i}}}$	– Same –	Shape Boundary is less noisy.
Metric 3 (Weighted Mean)	$\frac{w_iS_I^{D_i}+w_eS_E^{D_i}+w_cS_C^{D_i}}{w_i+w_e+w_c}$	When the internal structure is more important than external and concave. Usually the order of the weights are: $w_i > w_e > w_c$.	Highly noisy shape boundary: many false concavity and convexity, makes internal structure more reliable.

In this way, for each retrieved document, the set of scores $\{S_I^{D_i}, S_E^{D_i}, S_C^{D_i}\}$ are obtained, consecutively representing internal, concave and convex matching scores. The final score S^{D_i} can be obtained in three different ways, based on the choice of a metric and their usability on the dataset (Table. 6.1). Typical matching results for each character type are shown in Figure 6.21, 6.22, 6.23, 6.24, 6.25, 6.26, 6.27, and 6.28.

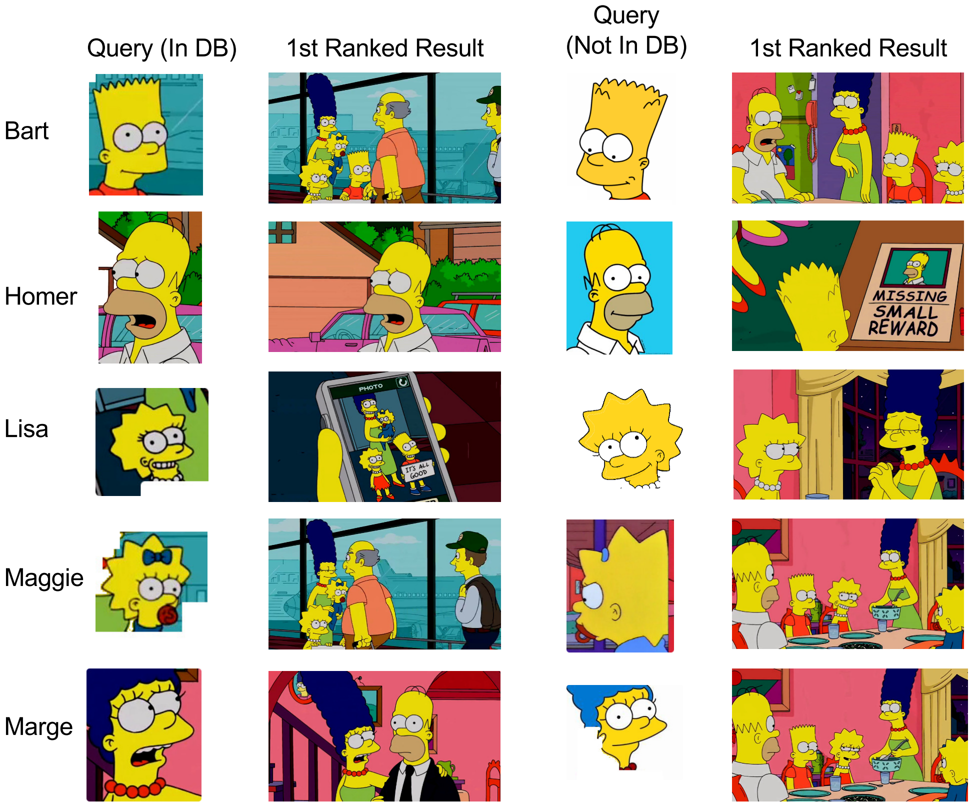


Figure 6.21: Results on our Simpson's dataset. *Left*: Retrieving the frames when the query is part of the database. *Right*: Retrieving the frames with closed possible shape matching when the query is *not* a part of database. Due to the dataset redundancy, only the 1st ranked result is shown.

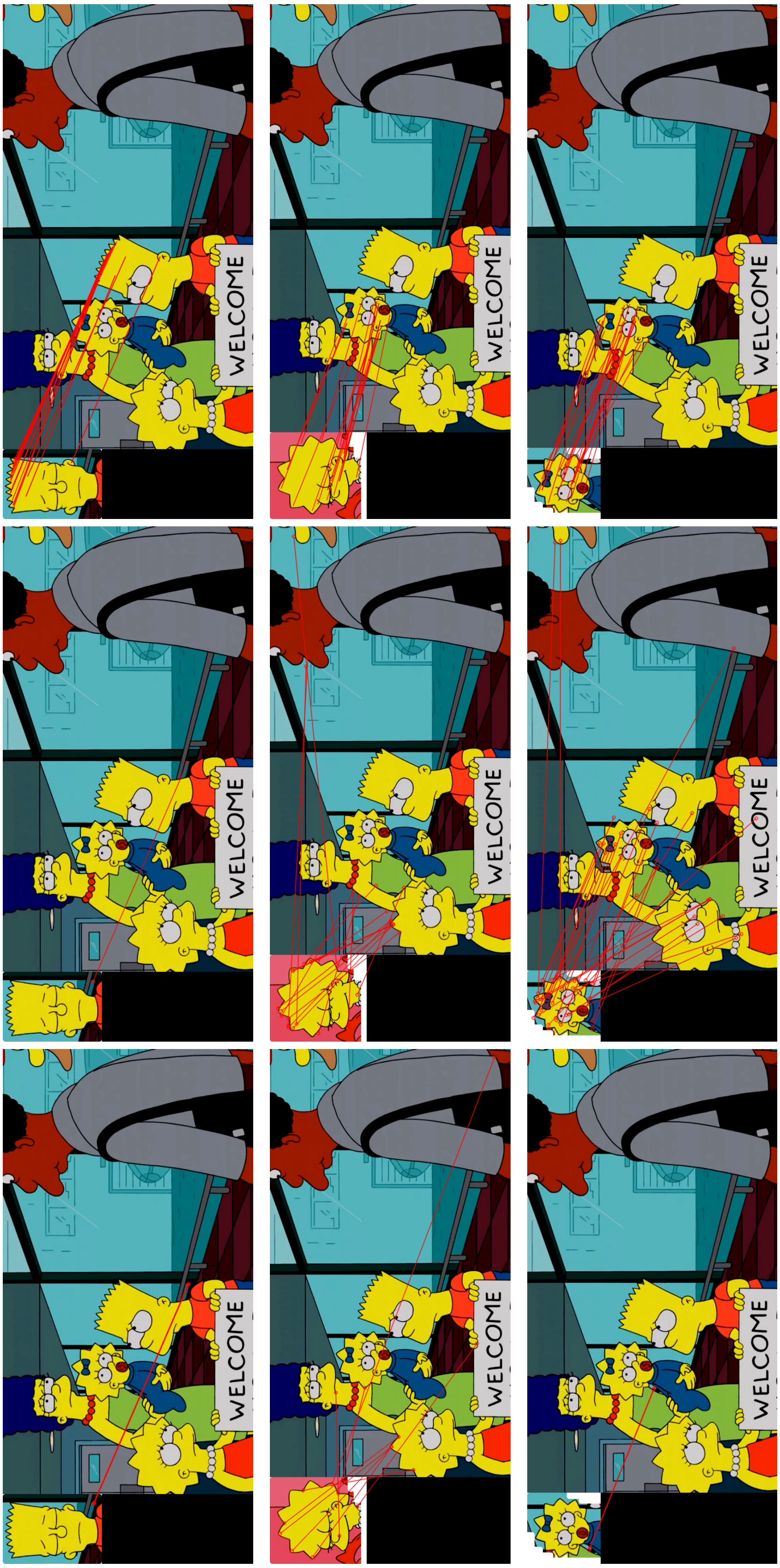


Figure 6.22: More results on Simpson's Characters: SIFT (left column), SURF (middle column) and our algorithm (aka ShIFT feature vector, right column).

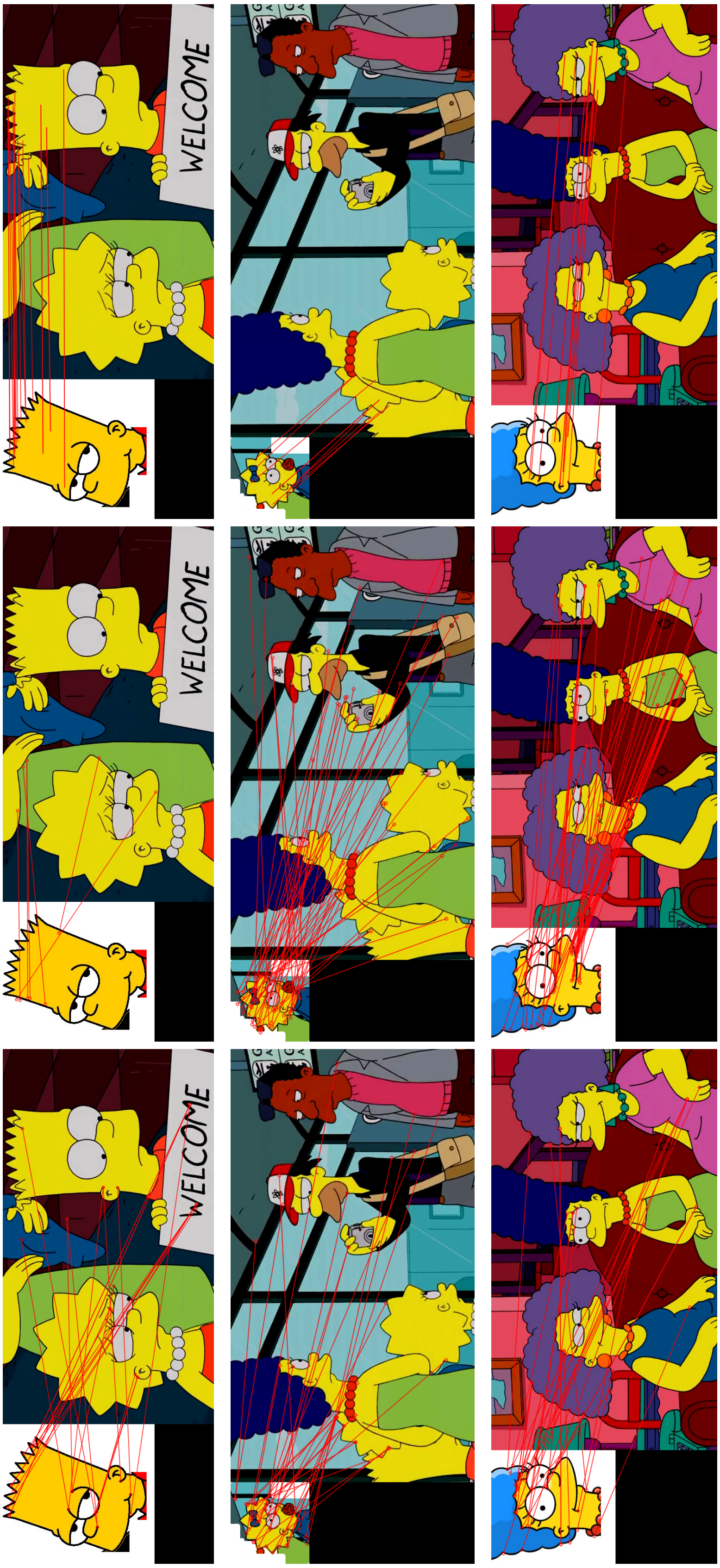


Figure 6.23: More results on Simpson's Characters: SIFT (left column), SURF (middle column) and our algorithm (aka ShIFT feature vector, right column).

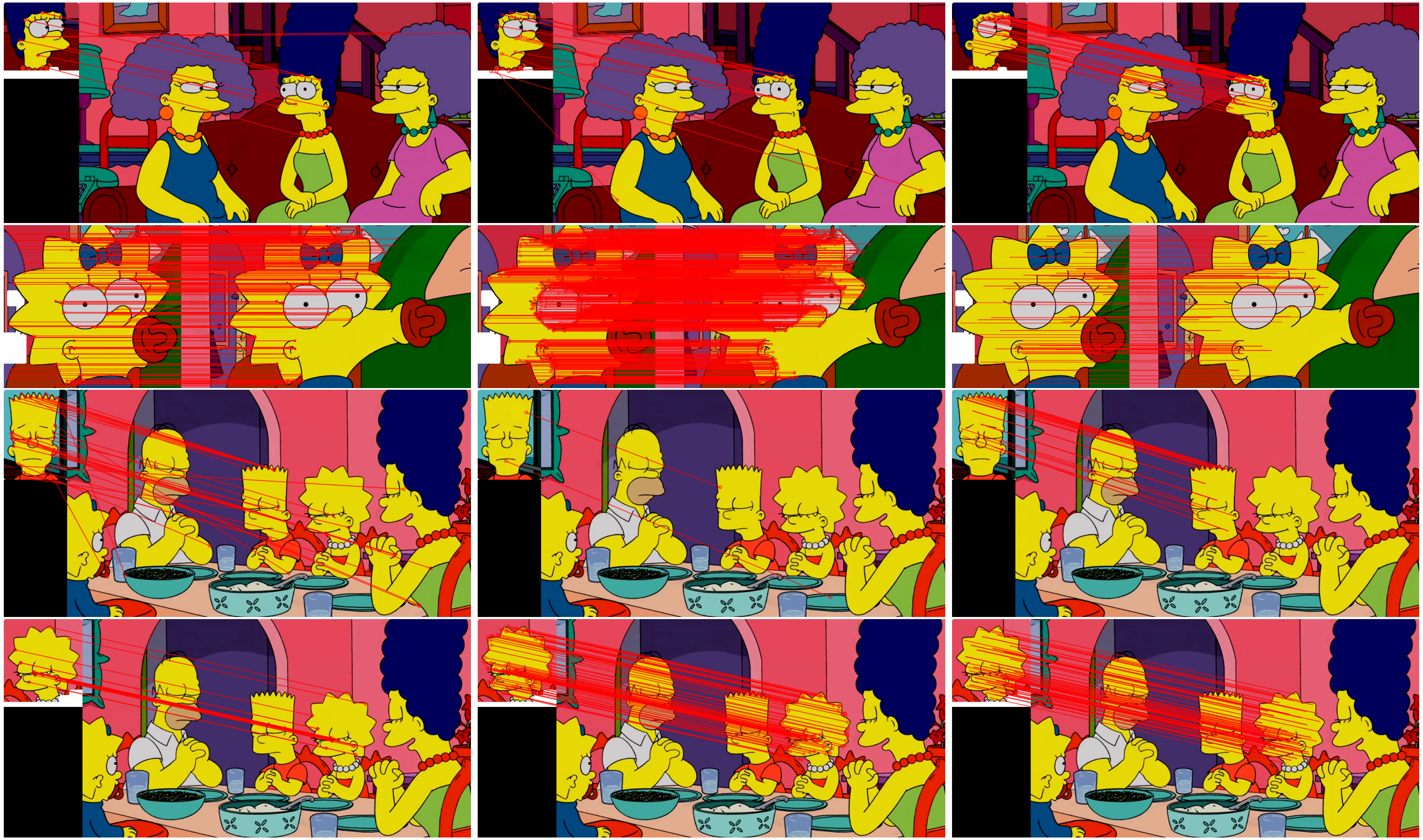


Figure 6.24: More results on Simpson's Characters: SIFT (left column), SURF (middle column) and our algorithm (aka ShIFT feature vector, right column).

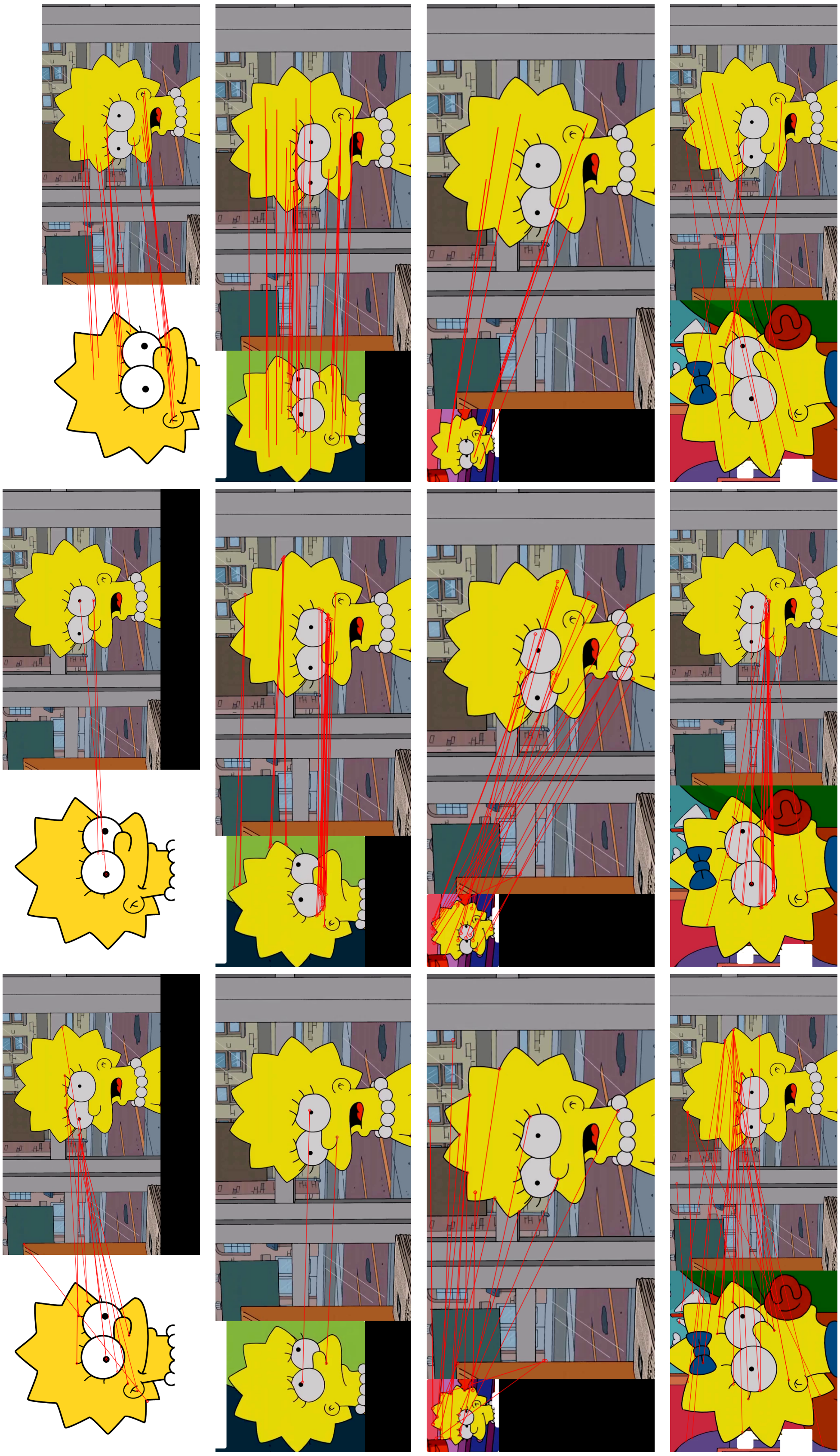


Figure 6.25: More results on Simpson's Characters: SIFT (left column), SURF (middle column) and our algorithm (aka ShIFT feature vector, right column).

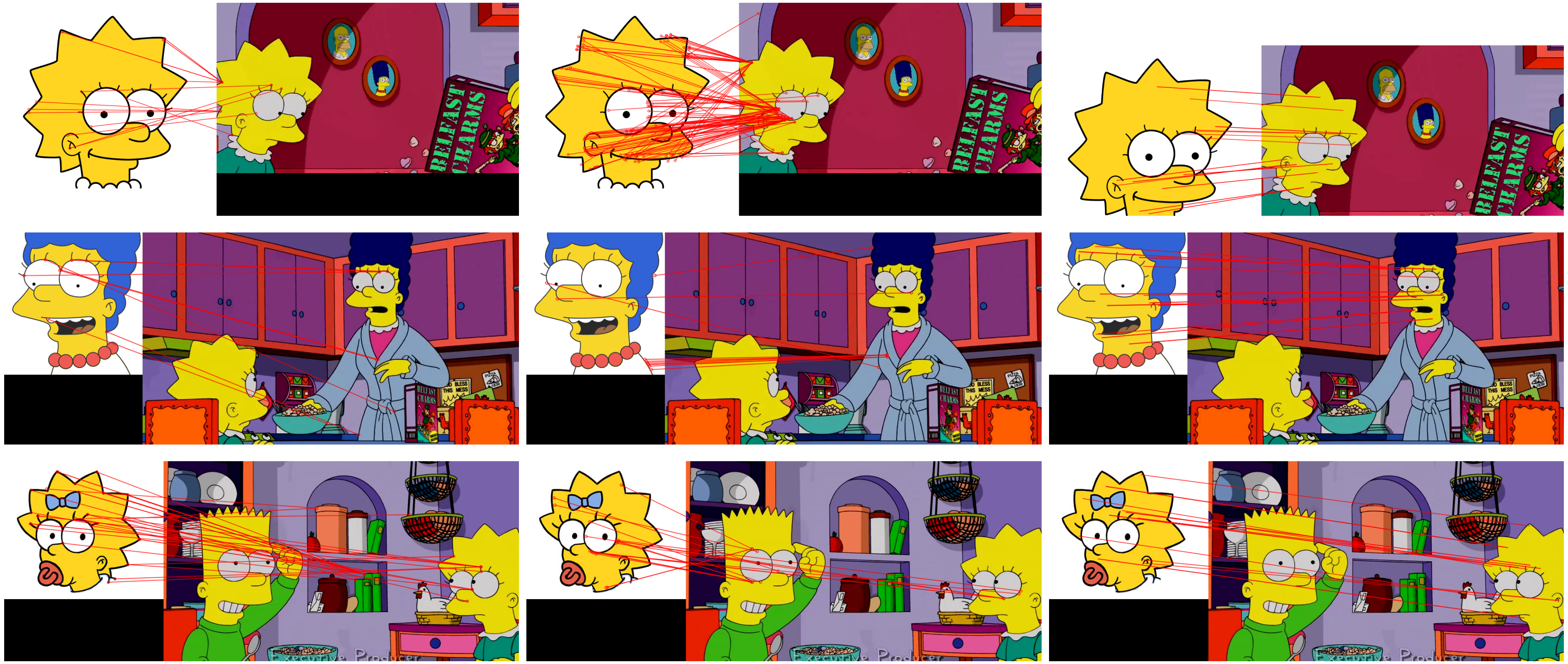


Figure 6.26: The feature-based algorithms SIFT (left column) and SURF (middle column) produce poor matching results when there is change in background texture and/or slight deformations in the target objects' shape. Our algorithm (aka ShIFT feature vector, right column) is robust enough in finding the most relevant shape (or area) in the target image, even when faced with relatively large object deformations and texture change.

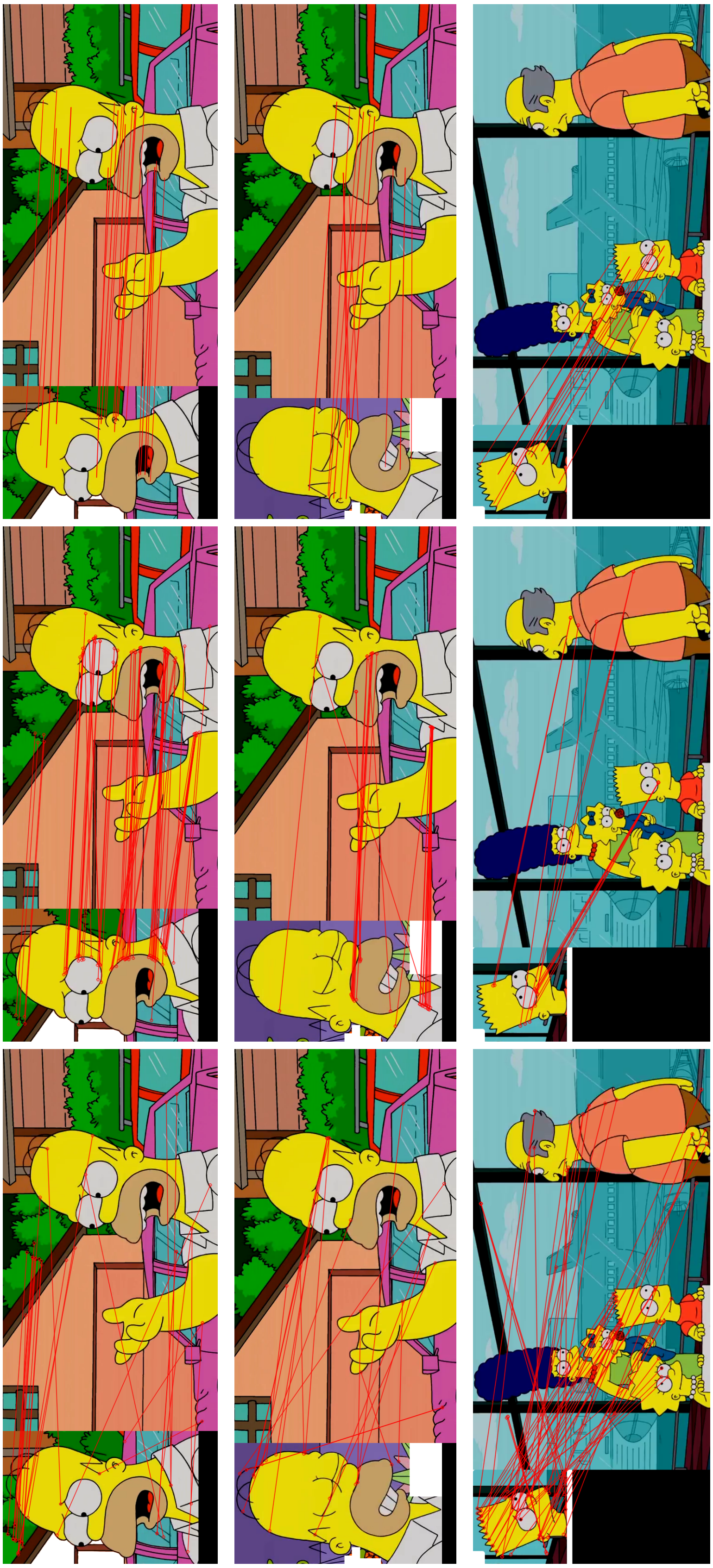


Figure 6.27: More results on Simpson's Characters when faced with relatively large object deformations and texture change: SIFT (left column), SURF (middle column) and our algorithm (aka ShIFT feature vector, right column).

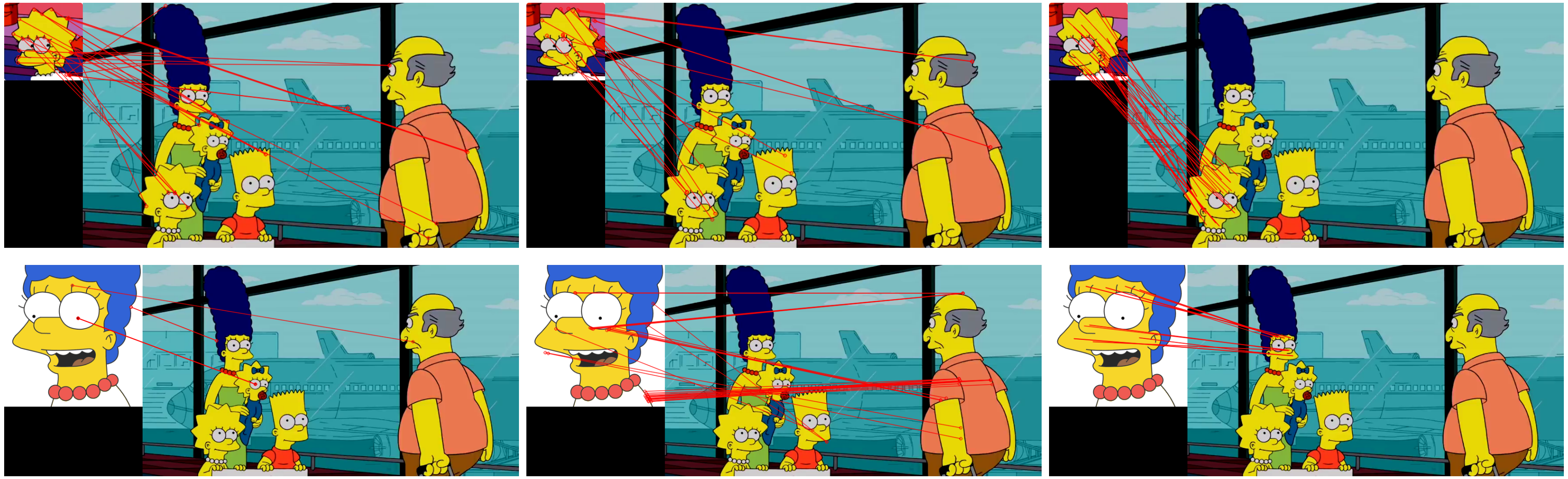
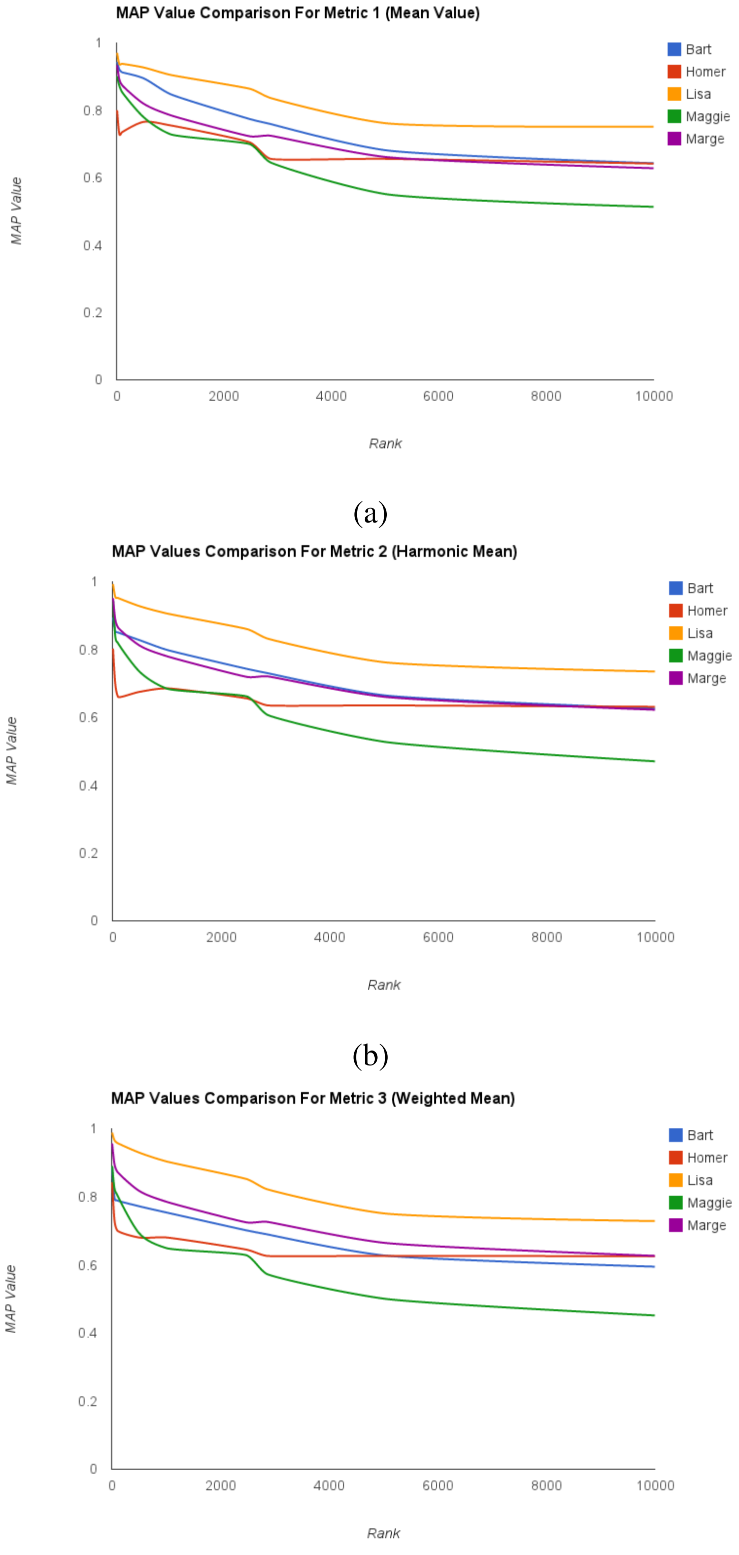


Figure 6.28: More results on Simpson's Characters when faced with relatively large object deformations and texture change: SIFT (left column), SURF (middle column) and our algorithm (aka ShIFT feature vector, right column).



(c)

Figure 6.29: MAP-values at different ranking locations on Simpson's characters for three metrics: (a) Mean Value as Metric 1, (b) Harmonic Mean as Metric 2, and (c) Weighted Mean as Metric 3.

We tested the performance of our shape-descriptor (ShIFT vector) on Simpson's video frame sequences with three ranking metrics – (a) Mean-value, (b) Harmonic Mean, and (c) Weighted Mean (with weights: w_i=0.45, w_e=0.3, w_c=0.25). For each of these metric, we calculated Mean Average Precision (MAP) value at each ranking location (see Figure 6.29. Overall MAP values for each metric are: (a) Metric 1: *0.64*, (b) Metric 2: *0.62*, (c) Metric 3: *0.61*. This shows that when the shape has minimal boundary noise, Metric 1 outperforms over other two metrics.

For any shape, the recognition rate increases when the query images are more likely parts of the dataset or when the dataset contains more samples. For example, the character Lisa Simpson (orange curves in Figure 6.29) has the highest retrieval rate. On the other hand, the retrieval rate reduces drastically when a character appearance is less frequent in the dataset, or there is much changes in the 3D appearances in the frame sequence. This situation is well explained with the character Maggie Simpson with the result of green curves in Figure 6.29. Moreover, the facial shape of Maggie Simpson is very similar to Lisa Simpson and because of this, many times it produces true-negative results.

Many frame sequences in our dataset has scaled, rotated, translated, mirror flipped and some times changes of view points. Our shape descriptor is robust against these transformations and produces more likely results at higher ranks. Also, it tends to find the similar shape when the query is not a part of the dataset. This situation is well illustrated with character Bart Simpson (blue curve in Figure 6.29).

This is not entirely conclusive as our database remains limited in size; furthermore, in the next following experiments, we compare our method with other ones which specifically rely on explicit shape measures other that simple edges or corner loci, such as "shape contexts" and its relatives (Belongie et al. (2002); Xie et al. (2008); Nanni et al. (2014)), although most of these techniques typically are not adapted to deformations, articulations, structural perturbations, occlusions and cuts and also most are not scalable to large databases.

6.1.2 Result analysis with shape-oriented approaches

6.1.2.1 MPEG-7 Shape Dataset

For our third set of experiments we tested the MPEG7 shape benchmark dataset (Latecki et al. (2000); Latecki and Lakamper (2000)). This dataset consists of 1,400 silhouette images from 70 2D classes including apples, bats, beetles, bells, birds, bones, bottles, bricks, butterflies, camels, cars etc., where each class contains 20 forms. Although, this dataset offers a possibility to compare various shape descriptors of non-rigid shapes with a single closed contour, some classes are fairly similar and there is often complicated appearance changes within the same class. Thus it is often hard to perfectly accomplish the task of shape retrieval on the dataset due to its inter- and intra- class variability.

The performance is measured by the standard Bullseye test: for each shape, we match it with all other shapes in the database and the top 40 matched shapes are considered. The retrieval accuracy is the proportion of the 20 correct shapes in the top 40 matches. Table 6.2 shows the reported results from different existing/recent state-of-the-art algorithms on this dataset. Based on the a priori designed 70 object classes contained in the MPEG7 dataset, the inner distance-based method as modified by Gopalan et al. (2010) reported the best performance, followed by IDSC + LCDP method (Yang et al. (2009)), SVS + DP Nguyen and Porikli (2013) and our method, which puts us near the top of the state-of-the art reported, when using this particular database and this particular base test.

Most of the other shape matching algorithms (Table 6.2) are designed with respect to robustness and accuracy. When it comes to the retrieval of target images/objects amongst a large database of forms, say with millions of images, then almost all of these methods show limitations due to their numerical complexity (most being $O(n^2)$). For example, the shape context (SC), inner distance shape context (IDSC) and their variants (which

Table 6.2: Comparison on the MPEG7 Shape dataset, where n is the number of samples in a dataset, articulation invariance (a), detecting concavity/convexity (c), hole handling (h), capability for indexing (i), shape-parts detection (p), rotation invariance (r), scale invariance (s), utilizing texture information (t), and view point invariance (v).

Algorithm	**A (%)**	$O(\cdot)$	**Properties**
Gopalan (Gopalan et al. (2010))	93.67	n^2	ac··prs·v
IDSC+LCDP (Yang et al. (2008))	93.32	n^2	··h··rs·v
SVS+DP (Nguyen and Porikli (2013))	91.07	n^2	a·h··rs·v
Shape Vocab (Bai et al. (2014))	90.41	$dn\log D$	·c·i·rs·v
ShIFT (our method)	**89.62**	$n\log n$	ac·iprs·v
Mix Gauss+tSL (Liu et al. (2010))	89.1	$n\log n$	··hi·rs·v
Shape-tree (Felzenszwalb and Schwartz (2007))	87.7	n^2	··h··rs·v
IDSC+DP+EMD (Ling and Okada (2007))	86.56	n^2	a·h··rstv
Biswas (Biswas et al. (2010))	86.48	n^3	a··i·rs·v
Hierarch. Prosc. (McNeill and Vijayakumar (2006))	86.35	n^2	··h·prs·v
IDSC+DP (Ling and Jacobs (2007))	85.4	n^2	a···prstv
Gen. Models (Tu and Yuille (2004))	80.03	n^2	··h·prstv
Curve Edit (Sebastian et al. (2003))	78.14	n^2	a····rs·v
SC+TPS (Belongie et al. (2002))	76.51	n^2	····prstv
Visual Parts (Latecki and Lakamper (2000))	76.45	n^2	·c··prs·v
CSS (Mokhtarian et al. (1997))	75.44	n^2	·····rs··

includes the reported top 2 methods in terms of accuracy, Table 6.2) cannot be used for practical indexing as a change in the shape boundary significantly impacts their representation. Also, most reported methods currently rely either on clustering-based class means or on one-to-one comparisons with other feature vectors to find an error or a score (based on minimum difference) providing the final target. As a result, the numerical complexity of such methods increases quadratically with the size of a database. Noticeably, Biswas et al. (2010) have introduced a robust feature vector, which is a mixture of several features (inner distance, relative angles contour distance, and center of mass) and more importantly can be quantised for meaningful indexing. The drawback of this method is that while retrieval has a linear time complexity, indexing scales with n^3. By contrast, we designed our feature vector ShIFT in a robust and efficient way such that it can be indexed with a complexity of $n\log n$ using B+ trees and retrieved with complexity$\log n$.

In Table 6.2, we have ordered the various shape descriptors including ours according to their reported *accuracy* (2nd column, adapted from Nguyen and Porikli (2013)). Our method shows an accuracy of 89.92% on the MPEG7 dataset, which is competitive being currently ranked in 4th place. However, when considering the maximum indexing or retrieval complexities our method is best (3rd column of Table 6.2).[3] The method of Liu *et al.* (Mix Gauss+ total Square Loss, Liu et al. (2010)) also reports a similar time complexity, however when considering space complexity, theirs goes to $n\log n$, while ours is n. In terms of retrieval, almost all these methods shows a complexity of n, except Liu et al. (2010) and ours (ShIFT) which have the better complexity $\log n$, *i.e.,* almost constant with increasing dataset size. The linear complexity in retrieval for most published methods makes it hard for these to retrieve matches from the large (and increasing in size) datasets as they do comparisons with each form to evaluate their score. In our case, the quantization property of our feature vector provides for an easy indexing and the retrieval is logarithmic (sub-linear) which outperforms most currently published shape-based methods, to the best of our knowledge.

In the fourth column in Table 6.2 we summarise via nine properties how the various shape-based methods can be further compared, in terms of: articulation invariance (a), detecting concavity/convexity (c), hole handling (h), capability for indexing (i), shape-parts detection (p), rotation invariance (r), scale invariance (s), utilizing texture information (t), and view point invariance (v) related to a shape. Our current version of the ShIFT algorithm possesses all properties except that of explicitly handling holes and utilizing texture features.

We have noted that our algorithm produces slightly poorer results in those cases where a shape has holes (for example, such as some cattle, dog, horse-shoe specimens, Figure 6.30). For these cases, we miss internal dominant points due to the presence of holes.

[3] We are evaluating the worst complexity amongst indexing or retrieval. The methods which uses clustering get a quadratic complexity to find a centroid. The methods which use kd-trees as their feature representation get a complexity of $n\log n$.

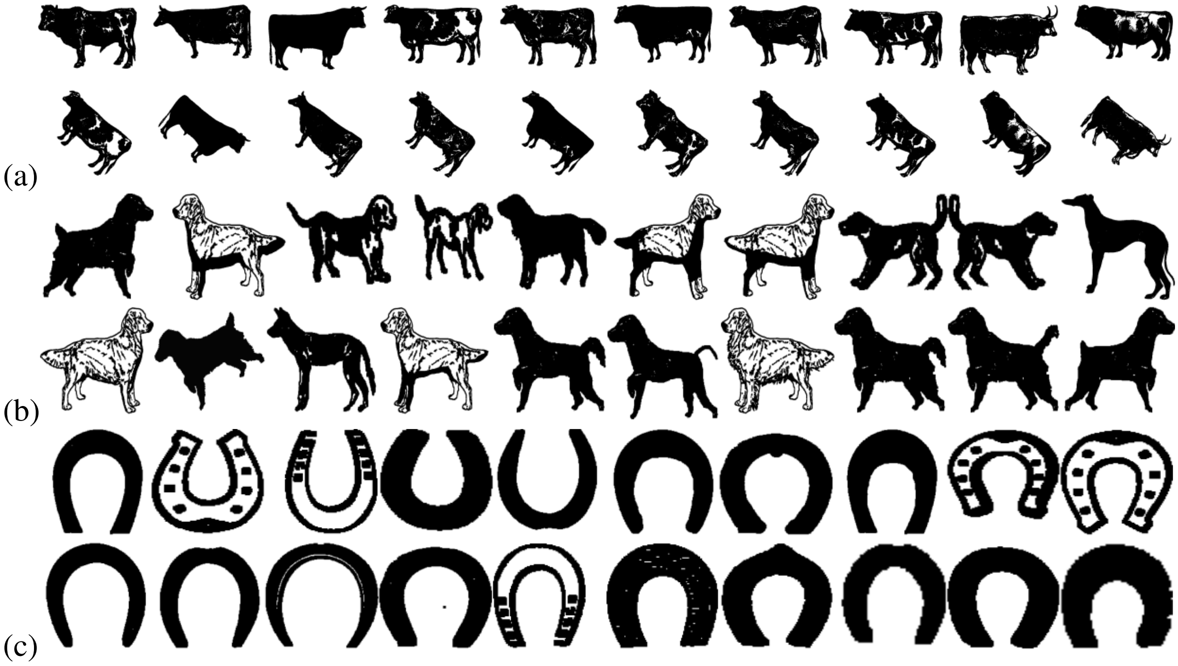


Figure 6.30: Some MPEG7 classes: (a) cattle, (b) dogs, and (c) horse-shoes that contain holes in different samples.

But still, we are able to retrieve other internal dominant points in those regions where the shape is unchanged, *i.e.* without holes, which provide us with correct affine parameters. However, we also tend to get more concave and convex dominant points (then for a similar form without holes), which tends to lower the final matching score. As a consequence of our built-in bias in our scoring function (Equation 5.6), the relevant target shape with holes might get pushed down in the ranking, while another less relevant target sample but with more internal dominant point matches may get a too high relative ranking.

Alike Nasreddine et al. (2010), we report here that the MPEG7 dataset has visually dissimilar samples in the same class (Figure 6.31) and also different classes have similar samples in their shape (Figure 6.33). Examples of such situations are shown in Figure 6.32, where we report our top 10 matches (*e.g.* a disc is matching with apples and squares with top hats). In these cases, the medialness field is very similar and as a result produce similar dominant points and create (perhaps) class confusion. Consequently, we have noticed that some classes could be broken into separate smaller classes as, arguably,

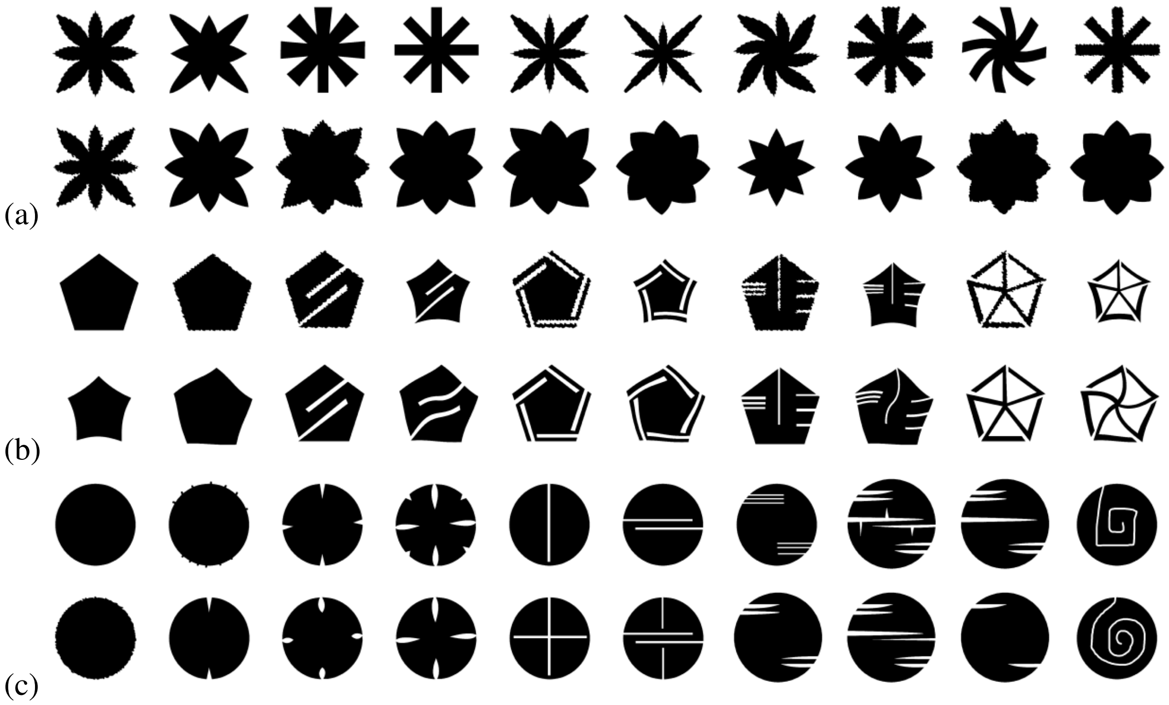


Figure 6.31: Some MPEG7 classes: (a) Devices 2, (b) Devices 6, and (c) Devices 9, which contain perceptually (and arguably) different samples (intra-class difference) but were put together in one class.

from a perceptual point of view, they indicate different types of objects (Figure 6.31). For example, samples which belong to the MPEG7 classes device0-device9 should be further segmented in sub-classes as they highlight sufficiently different shape characteristics. Our ShIFT method does highlight such (perhaps more subtle) differences which we argue are perceptually significant; on such a refined set of classes and sub-classes we expect our method would then outperform as well the other current top (3) methods in terms of accuracy, as these do not highlight shape features such as tapering, necks or concavities.

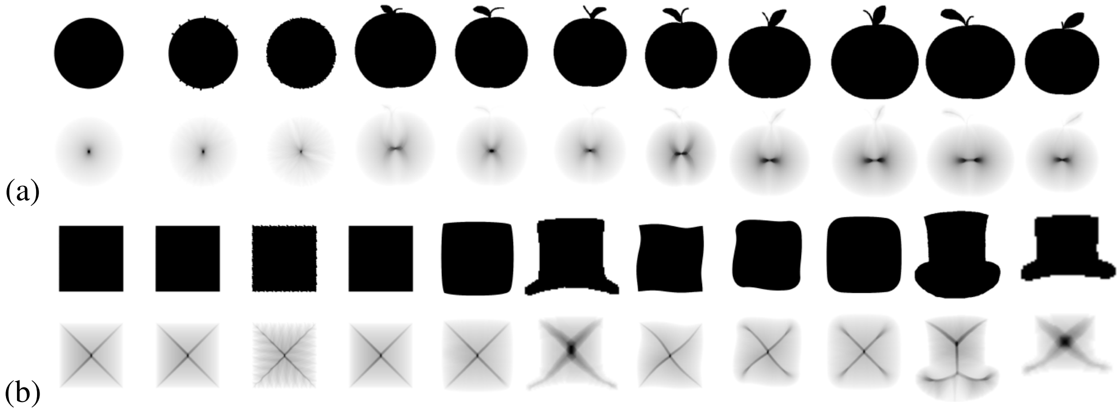


Figure 6.32: Top-10 results along with their medialness. All the shapes are visually similar but they are put in different classes.

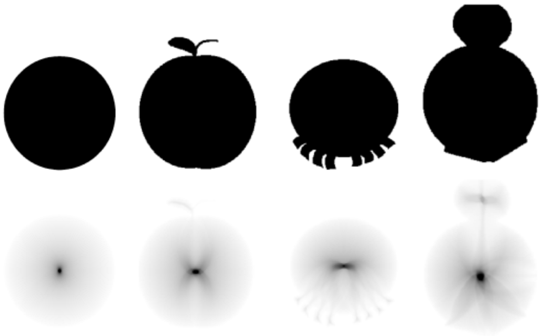


Figure 6.33: Examples of shapes issued from different classes but arguably highly similar. Bottom row shows their medialness which again reflects high similarity.

6.1.2.2 Swedish Leaf Dataset

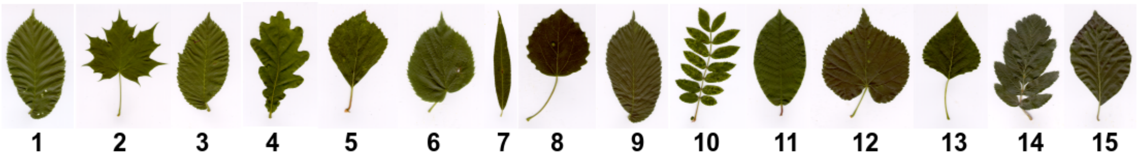


Figure 6.34: Swedish Leaf Dataset: one sample per class. Note that some classes are quite similar, e.g., the first, third and ninth classes.

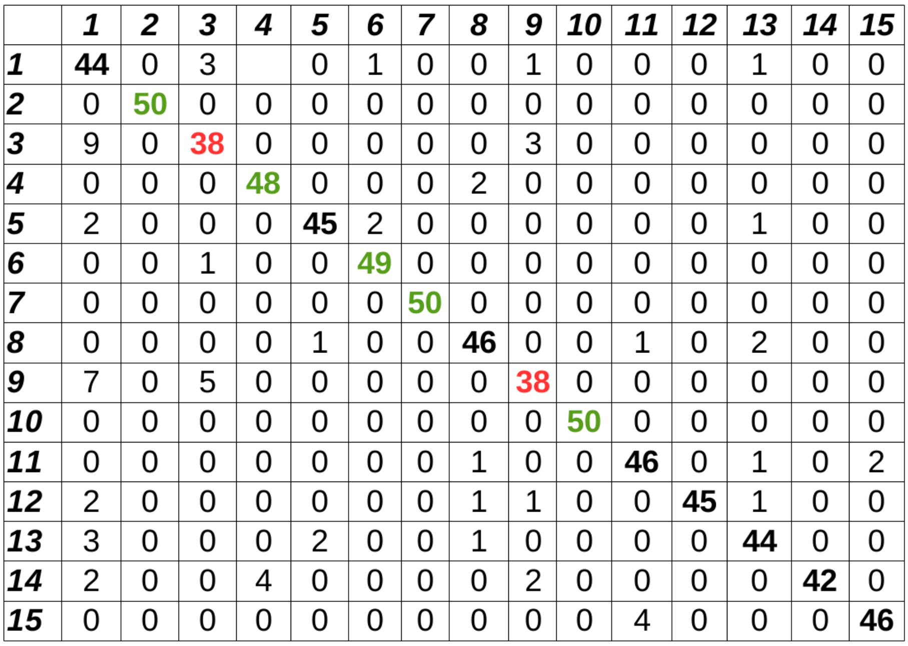
Table 6.3: Comparison on Swedish-Leaf dataset.

Algorithm	**Accuracy (%)**
BCF (Wang et al. (2014))	96.56±0.67
Shape-Tree (Felzenszwalb and Schwartz (2007))	96.28
IDSC+DP (Ling and Jacobs (2005))	94.13
Our method	**90.80**
Spatial PACT (Wu and Rehg (2011))	90.77
Fourier descriptors (Ling and Jacobs (2005))	89.60
SC+DP (Ling and Jacobs (2005))	88.12
Soderkvist (Söderkvist (2001))	82.40

Our fourth and final set of experiments is tested on the standard Swedish leaf dataset that comes from a leaf classification project at Linkoping University and the Swedish Museum of Natural History (Söderkvist (2001)). This consists of the isolated images belonging to 15 classes of Swedish leaves, with 75 samples per class (Figure. 6.34). Alike the MPEG-7 dataset, this dataset is challenging because of its high inter-class similarity (Serra (1983); Vincent (1993)). For example classes 1, 3 and 9 show similar outline structure. The first 25 images from each class are used for training and the rest for testing. We used the (mathematical morphology) watershed algorithm to segment and find contours of the images. As the training, we first extract our feature points out of the selected training samples and then index them using B+ trees. The extracted features from the test cases are then matched with this indexed database and ranked according to the score in equation 5.6. We use a *k-nearest neighbor* (kNN) classifier (with k=3) for the test cases and obtained an overall accuracy of **90.80%** which is tabulated along with other recent methods in table 6.3 (we are reporting only those methods that are using contours/shapes).

Table 6.4 shows the class confusion matrix that we obtained through our method. Letters in green-bold face represents those where we get an accuracy greater than 95% while red ones represent those cases where we get an accuracy less than 80%. This matrix clearly shows that for the classes 1, 3 and 9 our method is more confused with lower accuracy in classification results. Utilization of other features such as venation patterns,

Table 6.4: Confusion Matrix on Swedish-Leaf dataset.

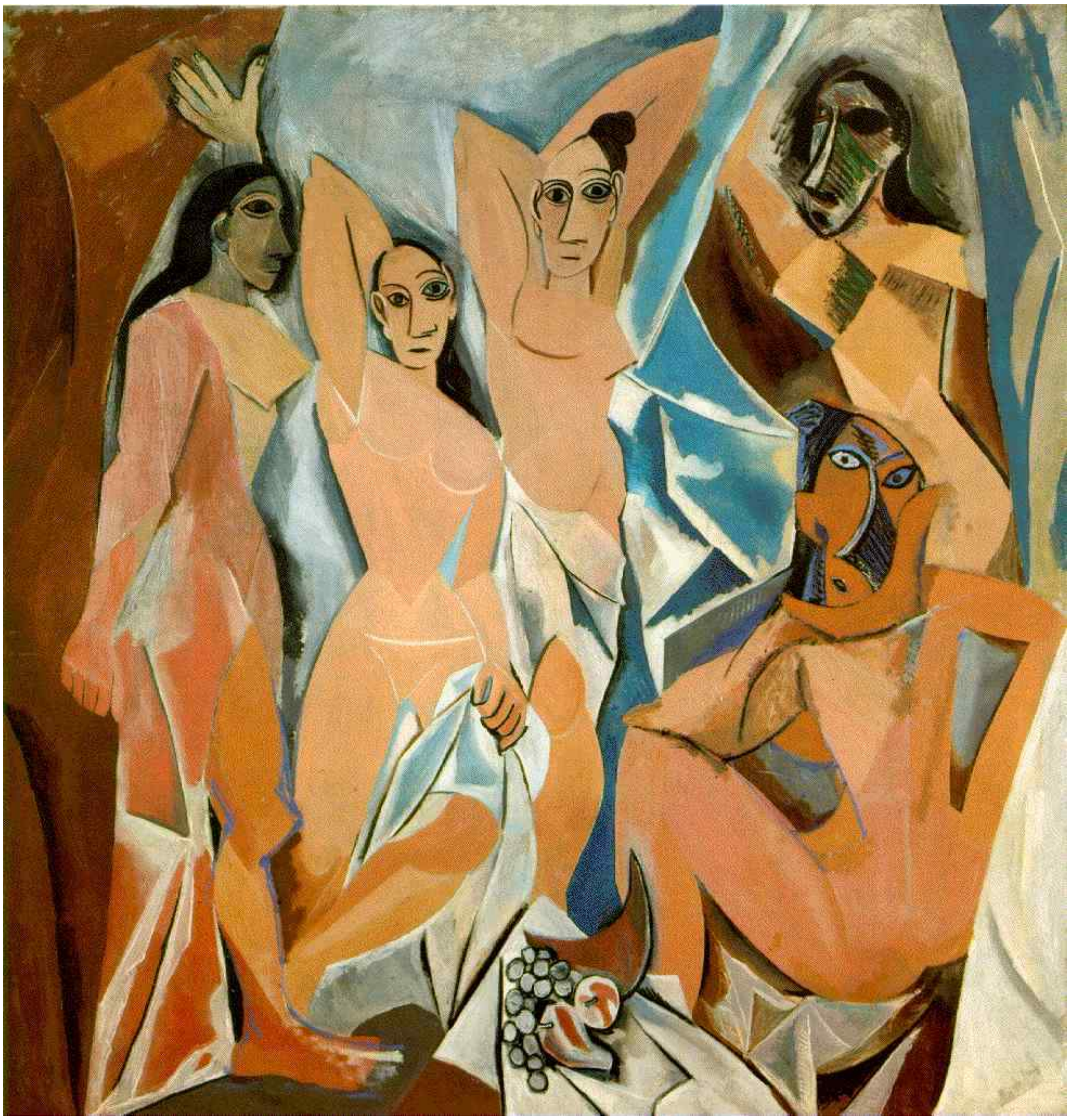


	1	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***	***10***	***11***	***12***	***13***	***14***	***15***
1	**44**	0	3		0	1	0	0	1	0	0	0	1	0	0
2	0	**50**	0	0	0	0	0	0	0	0	0	0	0	0	0
3	9	0	**38**	0	0	0	0	0	3	0	0	0	0	0	0
4	0	0	0	**48**	0	0	0	2	0	0	0	0	0	0	0
5	2	0	0	0	**45**	2	0	0	0	0	0	0	1	0	0
6	0	0	1	0	0	**49**	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	**50**	0	0	0	0	0	0	0	0
8	0	0	0	0	1	0	0	**46**	0	0	1	0	2	0	0
9	7	0	5	0	0	0	0	0	**38**	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	**50**	0	0	0	0	0
11	0	0	0	0	0	0	0	1	0	0	**46**	0	1	0	2
12	2	0	0	0	0	0	0	1	1	0	0	**45**	1	0	0
13	3	0	0	0	2	0	0	1	0	0	0	0	**44**	0	0
14	2	0	0	4	0	0	0	0	2	0	0	0	0	**42**	0
15	0	0	0	0	0	0	0	0	0	0	4	0	0	0	**46**

texture and color can further increase the accuracy of our system. Also variations in the length of our feature vector should help make the method more accurate. Finally, notice that, to the best of our knowledge, other published methods do not scale well with the size of a database, while ours do ($O(\log n)$).

6.2 Summary

In this chapter we have shown the evaluation of our framework with our designed shape database, the Simpson's family video frame database and other standard shape and leaf databases. Our ShIFT feature is robust under partial occlusion, invariant towards affine changes and also capable in matching the object when the pose is changed. ShIFT outperforms SIFT and SURF when detecting objects (with defined shapes) in both binary as well colored images since SIFT and SURF fail to provide the explicit shape description. A slight change in the shape leads to false feature matching and hence retrieval quality reduces drastically. The detailed evaluation on the MPEG7 and Swedish Leaf benchmark dataset shows that we are very close to the top of the published state-of-the-art algorithms (as measured by accuracy). But, we also emphasise that when taking into account the capacity to scale to larger databases, we then outperform other methods as our retrieval time is $O(\log n)$. In the future we shall augment our approach by considering in addition a representation of venation patterns.



Pablo Picasso, *Les Demoiselles d'Avignon/The Brothel on Avignon Street*, 1907. @Museum of Modern Art, New York City.

Chapter 7

Conclusion and Future Work

> *"Every child is an artist. The problem is how to remain an artist once he grows up."*
>
> *– Pablo Ruiz y Picasso, Painter (1881–1973)*

This chapter summarises the main contributions of the thesis and draws important conclusions. Mainly, the goals of this work have been to:

1. provide the novel generic definition for shape that is derived from human perception (vision psychology, where early neural layers in the visual cortex (such as V1 and V2) are likely responsible for shape detection, Hatori and Sakai (2014));

2. formulate a computational model for this given definition;

3. propose an algorithmic chain to extract the shape features, which is invariant to affine transforms;

4. facilitate search in a large image dataset by exploiting the indexing property of the designed ShIFT feature vectors;

5. find the best possible match (homography) amongst the query and the target shape.

7.1 Conclusion

This thesis is about providing a generic computational model to define any 2D shape, which is motivated and adapted from the cognitive science literature when studying the visual attention of human subjects presented with articulated biological 2D forms, possibly in movement. Medialness is shown to have deep roots in perception and cognition. Our main goal has been in developing a computational framework on the basis of the models proposed by Kovács et al. (1998) (medial-point representation) and Richards and Hoffman (1985) (contour codons), themselves inspired by early works such as that of Attneave (1954). We have combined these two representations, i.e., region and contours. The representation describes shapes by a small number of *dominant points* that are important not only in terms of representing significant portions of the boundary segments, but also in terms of the spatial configuration of those segments. Recent studies in perception and cognition shows that contour extrema are better thought of as combining significant curvature peaks with regional support De Winter and Wagemans (2008), rather than referring to the traditional mathematical definitions biased towards a purely local (point-based) concept and analysis (such as derived from calculus). This property makes it attractive from the point of view of real perceptual tasks such as biological locomotion, articulation, and feature based shape retrieval from big datasets. An important aspect of the medialness representation is that it reduces the redundant information present in an image, and a small set of resulting dominant points convey information about the larger, extended objects.

The strengths of our approach include:

1. emphasizing a representation of 2D shapes based on results from studies on human perception via robust medialness analysis;

2. introducing an algorithmic chain providing an implementation of this shape analysis

tested on current reference 2D binary animal and plant databases;

3. mapping of the whole form into a small number of dominant feature points with associated shape significance (in terms of medialness, concavity, convexity);

4. converting these dominant feature points into affine invariant descriptors, ShIFT, used for indexing and fast retrieval purpose;

5. achieving accuracy for top ranked matches (exactness) and with nice (observable) degradation properties for following highest ranking results;

6. testing and demonstrating robustness under some levels of articulation and other structural (noise, cuts) perturbations. Our method also shows promising results for part-based matching tasks, in the context of occlusions, cuts and mixed object parts.

The shape representation we have proposed to explore for the purpose of Information Retrieval (on objects present in images) is well rooted in established results from human visual perception (Richards and Hoffman (1985); Kovács et al. (1998); Kovács (2010); Layton et al. (2014)) and it also relates to the art of drawing and animation (Leymarie et al. (2014b)). We are the first to propose and explore a possible algorithmic chain which implements the intuitive notion of "hot spots" first indicated by Kovács et al. (Kovács et al. (1998); Kovács (2010)).We have refined and extended this notion in a number of ways: (i) we have introduced orientation to the boundary in the computation of medialness (metric D^*_ε, equations 3.5 & 3.6) which reduces the effect of nearby extraneous contours (halo effect); (ii) we have proposed an adaptive scheme to automatically set the tolerance, or annulus width, in a logarithmic relation to an increasing disc minimum radius $R(p)$ (equation 3.10); (iii) we have extended the computation of medialness as proposed by Kovács et al. (1998) to regions exterior to an object, in particular to characterise significant concavities; (iv) we have added to the notion of "hot spots" of Kovács et al. (1998) significant concave and convex points at the tip ends of ridges of medialness; and

(v) on the basis of this representation we have introduced here a Shape Invariant Feature Transform (or ShIFT) which is made of two parts: the medialness feature vector and the quantized feature vector useful for indexing in large databases and for rapid (sub-linear) retrieval. The matching algorithm we have proposed represents a first step in exploiting this shape representation for its application in Information Retrieval problems for objects in images.

We emphasise that our motivation is inter-disciplinary: we aim at being able to address semantic gaps and mimic human pattern recognition performance, and this explains our exploitation of results from cognitive science in order to design a notion of feature points which relate explicitly emerging models of the human perception of shape. The original perceptual model of medialness proposed by Kovács et al. (1998) was studied in the context of the articulated movement of biological forms (such as a running animal). Our results presented in this communication confirm the potential of this method when applied to the Information Retrieval problem in dealing with static scenarios when accessing snapshots of either animals or plants or cartoon characters, and is proving also powerful for image sequences of articulated movement of animals.

7.2 Potential Future Threads

There are several ways further research could go, but there are also a variety of obvious extensions to the existing framework we have presented in this thesis. Subsequently I discuss them organised by topics: Movement Computing with Locomotion and Articulation, Gesture recognition for human-robot interaction, Understanding artistic images, Shape compression and reconstruction, Psychophysical investigation for illusory images, and Medialness-based 3D object description and recognition.

7.2.1 Movement Computing

By movement computing we understand a developing discipline which aims at providing computational schemes to automatically annotate movement and be capable of producing meaningful qualitative dynamics descriptions, including the inherent quality of a movement, its phrasing, embodiment and motivation, which are integral and should be included as parameters (Chi et al. (2000); Lehoux (2013)). Figure 7.1 shows a snapshot of video sequence of an artist performing on stage, where each shot reflects a different dance pose.

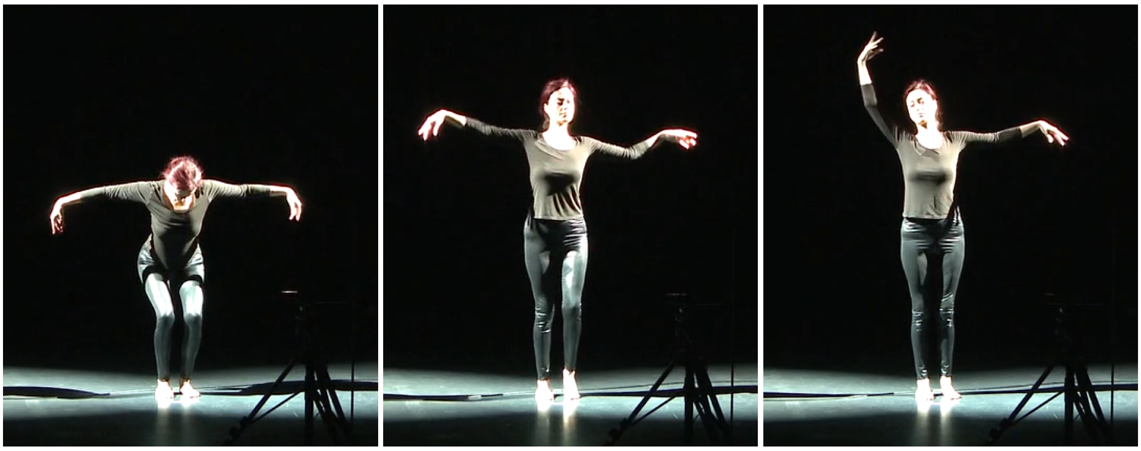


Figure 7.1: Video sequences of a dance pose of an artist performing on stage. Artist: Dr. Vesna Petresin.

The earliest known attempts to annotate movement were in dance and date from the 15th century (Guest (1998)), but the first true annotator of movement to include a strong qualitative element was Rudolf Laban who created a Labanotation in the first half of the 20th Century, based on the idea of harmonic movement (Foster (1986), p.77). The animator has no problem communicating and an animated version could include full body movement and motivation. It has been pointed out by Bishko (1993) that there is some similarity between Labanotation and the Principles of Animation in expressing the functional aspects of movement. While animation principles permit to impose a specific style of movement (such as in cartoons) they lack the depth and richness of Labanotation in exposing not only functional aspects of movement but their detailed expression (Bishko (2007); MacGillivray (2014)).

One important tool used in animation for controlling qualia which offers the potential for implementation is the idea of a central line of deformation being key to our assimilation of full body movement. The Line of Action (LoA) is a single line running through the character, which represents the overall force and direction of each drawing. In traditional celluloid animation such as used by early Disney full-length features like Snow White, before drawing the full character, an animator would frequently draw in a LoA to help determine the position of the character. By simply changing the LoA – making it more curved, sloped or arched in a different direction – the entire essence of the drawing can be changed. Bregler et al. (2002) have investigated capturing such a contour (the LoA) as the source of the motion and re-targeting it to other characters in other media. Although there is not enough information in this contour to solve for more complex motion, such as how the legs move relative to each other, the investigators discovered that a surprising amount of information comes from this single curve: the essence of the motion is still present in the re-targeted output. A more recent application of the LoA to 3D character interactive animation is being explored at the INRIA in France by Guay et al. (2013).

Our medialness map and extracted *dominant points* can be seen as a potential psychophysically motivated support for the LoA, where the later is a simplification of a full medial trace. Our pseudo-distance function, D^*_ε, which captures medial symmetries within a pair of disks defining an annulus region, where "ε" denotes the parametric thickness of the annulus. Under this metric, the special nodes of the skeletal field are those, which locally maximise the amount of outline trace they capture: they are "the most informative points along the medial-axis" of an object in motion.[1] Such a compact medial

[1] Our earlier works in this direction were presented in a paper at the ACM *International Workshop on Movement and Computing (MOCO*, Leymarie et al. (2014b))

representation shares strong resemblance to the chronophotographs of Marey (circa 1880) and to G. Johansson's work (circa 1970) on the perception of biological motion for point-light walker displays (Kovács (2010)). Our medialness function D^*_ε can be computed for each frame of a video sequence, when applied directly to available outlines. "The maxima of the function are good candidates as primitives for biological motion computations" (Kovács et al. (1998); Kovács (2010)). Such a representation of bodies in movement by a graph connecting nodes of high (visual) interest is a potentially richer model than the single LoA used in (cartoon) animation. It also offers an explicit markerless computational model and relates to human perception.

7.2.1.1 Locomotion and Articulation

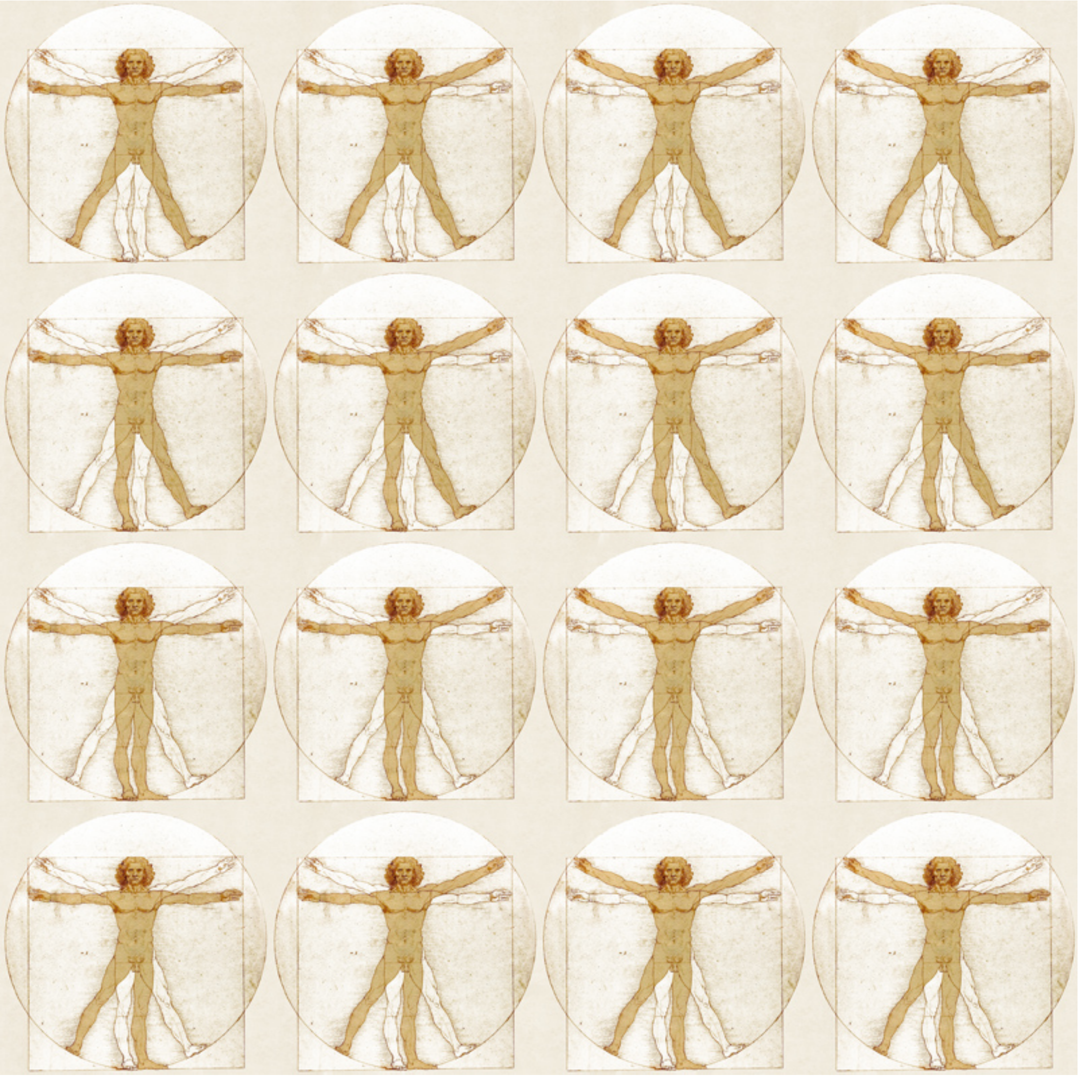


Figure 7.2: Original Image: An articulated set of 16 frames of Vitruvian men (after L. da Vinci, 1490).

Eadweard Muybridge (1830-1904), known as “the father of motion pictures”, pioneered the application of photography to the study of human and animal locomotion (Muybridge (1887, 1901, 1957, 1984)). During 1884-1887, he took photo sequences and then projected them onto a screen with a device he called a *zoopraxiscope* resulting in the world’s first illusion of moving pictures (Muybridge (1887)). While visiting the University of

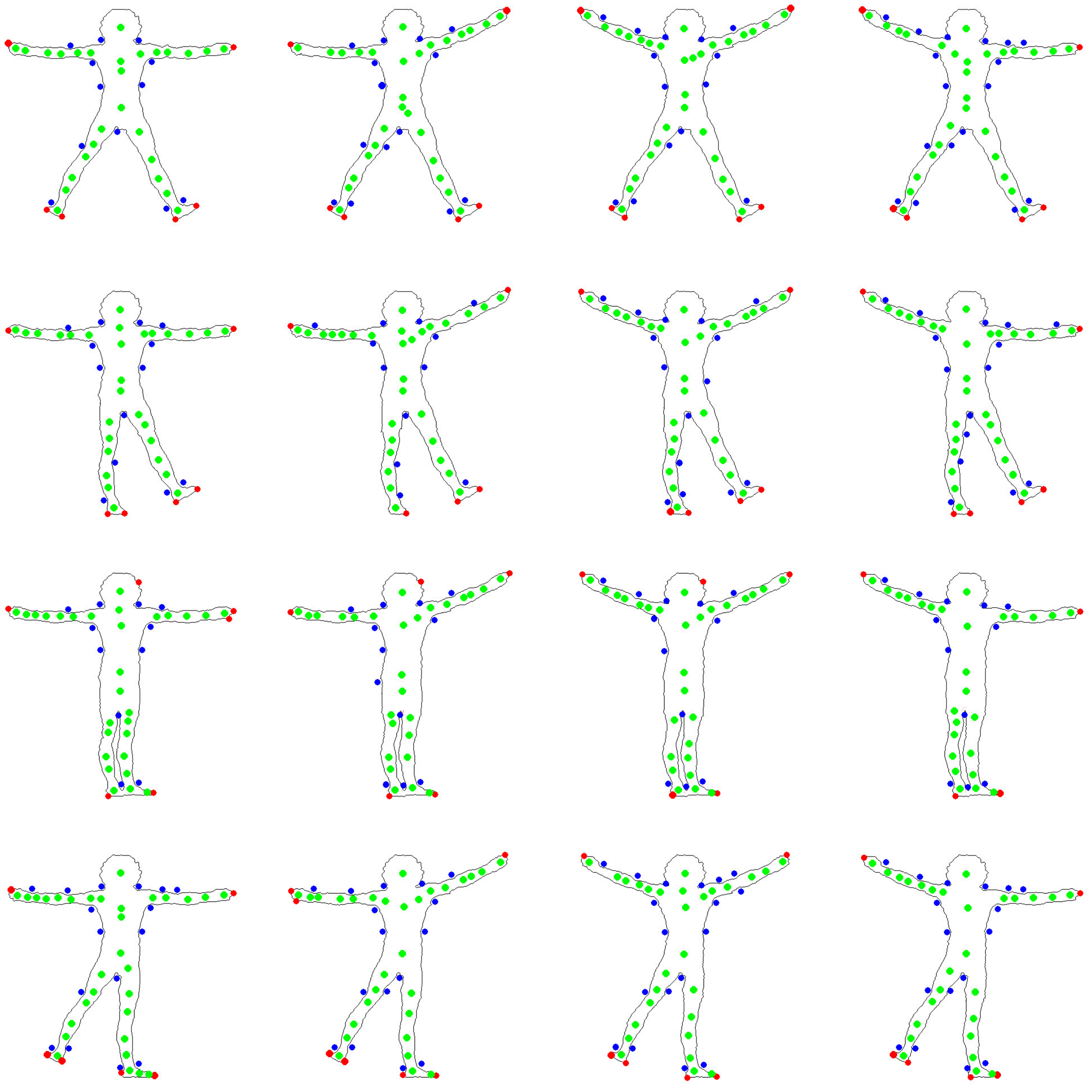


Figure 7.3: Dominant Points Image: 16 samples from an articulated set of Vitruvian men (after L. da Vinci, 1490) illustrating some features of our perception-based selection of dominant points.

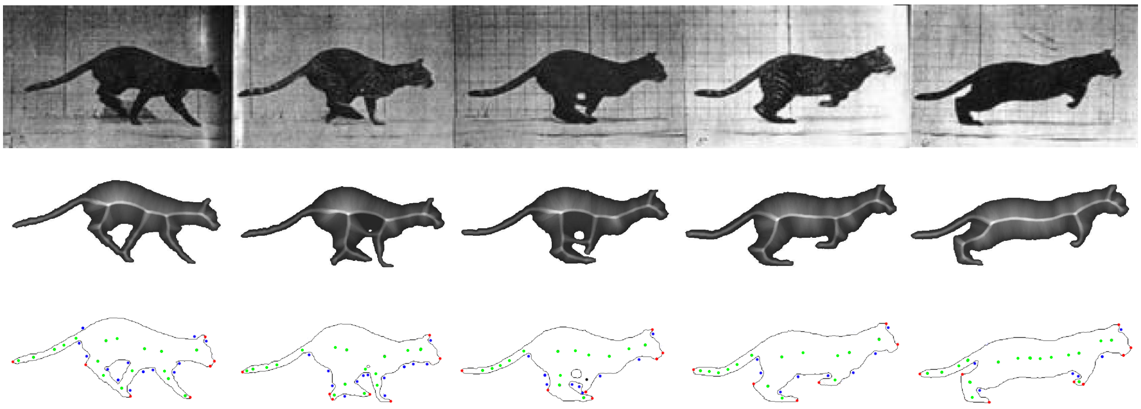


Figure 7.4: *1st Row*: Muybridge's original sequential set of frames of the movement of a running cat (Muybridge (1887, 1957)). *2nd Row*: Interior medialness map, D_ε^+-function. The maxima (white spots) of the function are good candidates as primitives for biological motion representation. *3rd Row*: Illustration of the changes in feature points loci for the same running cat.

Pennsylvania, Muybridge photographed many men and women along with four-legged animals. He found that whenever they ran, they lifted all their legs off the ground at once, even two-legged humans did. That complete lack of contact with the ground, in fact, came to define the act of running. Muybridge's photographs also revealed other rules. When four-legged animals walk rather than run, their feet usually hit the ground in the same pattern: hind left, front left, hind right, front right.

Anatomically, an animal's articulated movement is dependent on the point of connection between two bones or elements of a skeleton. Our results show that the concave points (representative exterior dominant points) have good potential to indicate and trace such articulations, unless the shape is highly deformed. For usual movements (e.g. walking, jogging, gesticulating), these feature points remain present most of the time and are identifiable in association to an underlying bone junction and hence can provide a practical signature for it; examples of this property are given in Figures 7.2 & 7.3.

Our work also directly relates to the early findings of Marey (1883), a French medical doctor and engineer: "In the method of photographic analysis the two elements of

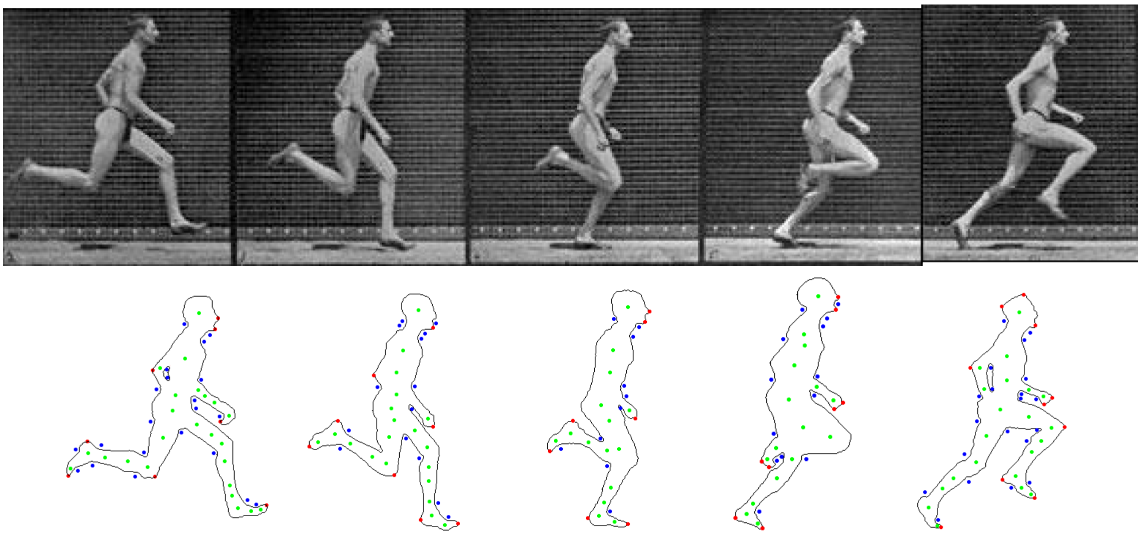


Figure 7.5: *1st Row*: Original sequential set of frames of the movement of a running athlete, captured by E. Muybridge in 1887 (Muybridge (1887, 1901, 1984)). *2nd Row*: Illustration of the changes in feature points loci for the same running human.

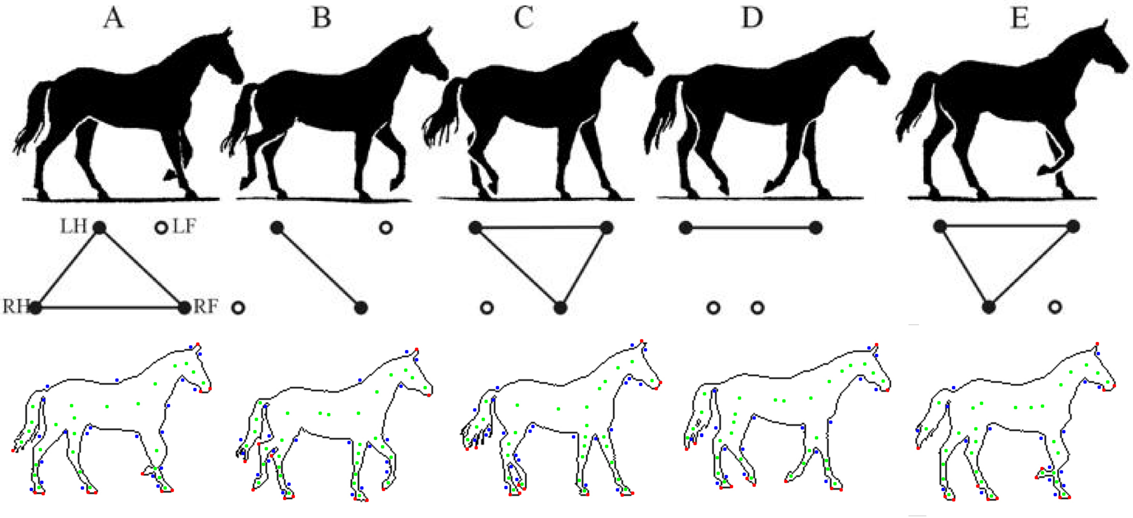


Figure 7.6: *1st Row*: Sequential set of frames of the trotting horse, captured by E. Muybridge in 1887 (Muybridge (1887, 1957)). When four-legged animals walk rather than run, their feet usually hit the ground in the same pattern: hind left (LH), front left (LF), hind right (RH), front right (RF). *2nd Row*: Illustration of the changes in feature points loci for the same running horse.

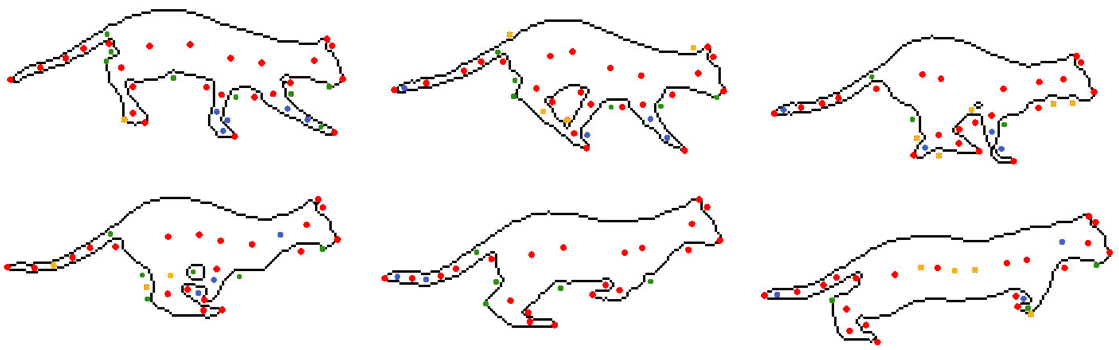


Figure 7.7: A running cat captured in 6 different frames (from Muybridge (1887, 1957)). Different dominant points (internal dominant, external dominant and convex points) are shown using colours to indicate their persistence in time: Red: present in all frames; Green: highly frequent (4-5 times); Blue: less frequent (2-3 times), and Yellow: not consistent (single occurrence). NB: Due to the animal's movement, sometimes dominant points overlap each other.

movement, time and space, cannot both be estimated in a perfect manner. Knowledge of the positions the body occupies in space presumes that complete and distinct images are processed; yet to have such images, a relatively long temporal interval must be had between two successive photographs. But if it is the notion of time one desires to bring to perfection, the only way of doing is to greatly augment the frequency of images, and this forces each of them to be reduced to lines."[2] Ours can be seen as providing a marker-less approximation to solve the inherent problem of capturing space and time information in one notation from an analysis based on photographic snapshot sequences or video (Figures 7.4, 7.6 and 7.5). In future work, the full potential of this representation should be explored; e.g., in Figure 7.7 we track our feature point set over a frame sequence to highlight their dynamics (here persistence in time).

[2] Marey (1883) was able to read the successive postures of the body on his plates, and follow the important trajectory of motion of those selected points, which he considered the most informative points.

7.2.2 Gesture Recognition for Human-Robot Interaction

We consider the 2D shape representation and its associated gesture characterisation problems in computer vision and robotics. To perform a gesture identification task efficiently (ultimately classification, recognition, learning), our point-based medialness shape descriptor can robustly capture the important structural information related to a deforming and moving form, in particular, in the context of humans and robots performing gestures-driven tasks. This work is part of a larger project on human perception and computer vision applied to artistic creativity and how it can impact social robotics, in developing new modes of communication and collaboration between robots and humans (Tresset and Leymarie (2012); Tresset and Fol Leymarie (2013)).

As we indicated previously, articulated movement is dependent on the point of connection between two bone elements of a skeleton. For usual movements, gesticulating, these feature points remain present and identifiable in association to an underlying bone junction and hence can provide a practical signature for it; examples of this property are given in Figure 7.9. Next we consider a study we conducted in collaboration with graffiti artist Daniel Berio.

The practice of writing graffiti tags is usually assimilated through years of observation and practice. With experience gestures are learned and embodied to the point of becoming “second nature” to the writer, leading to a drawing process that is performed almost automatically. With experience the hand will draw more efficient lines, which will ultimately result in more harmonious curves and evoke a strong sense of dynamism in the tag (Berio and Leymarie (2015)). Berio and Leymarie (2015) have observed that the stroke gestures used when writing tags assume an asymmetric ‘bell’ shaped velocity curve.

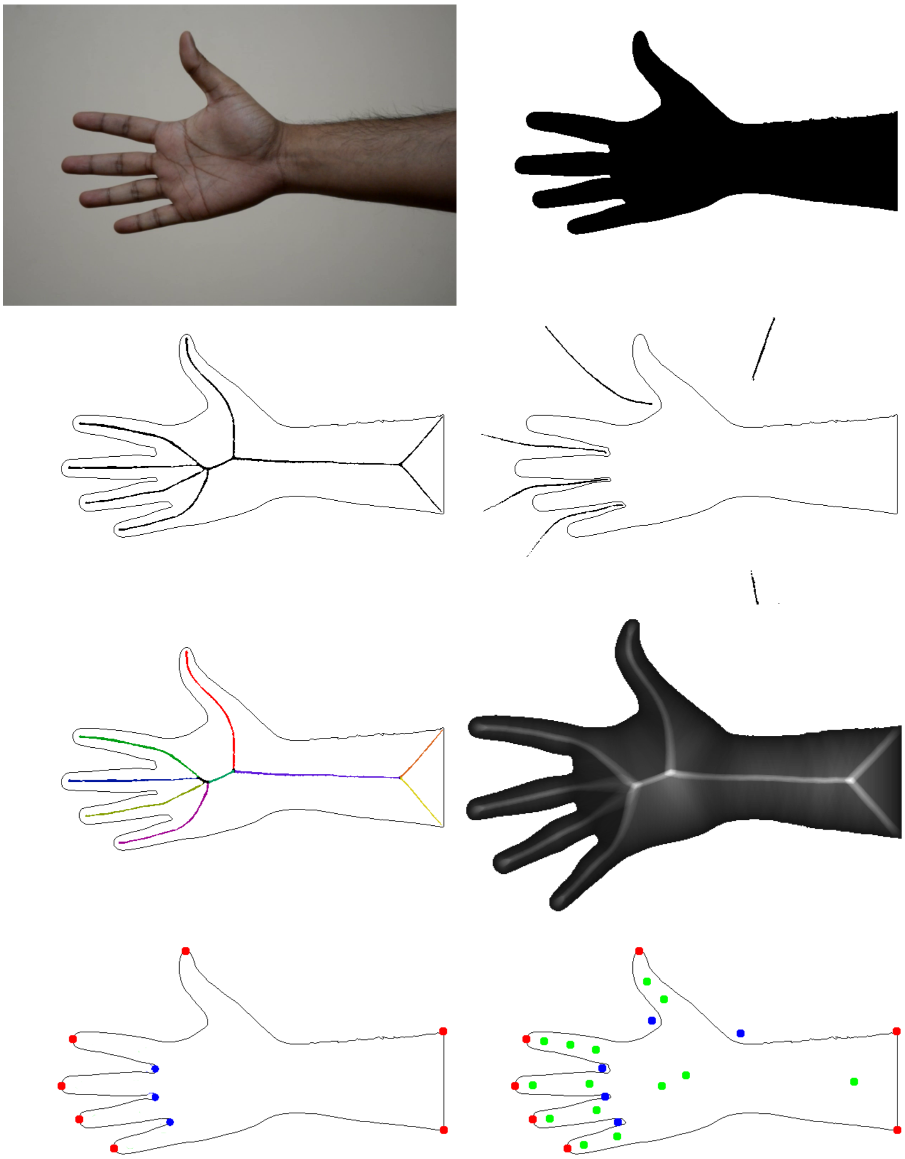


Figure 7.8: *1st row: Left:* Original shot of an open hand gesture; *Right:* corresponding segmented and binarised (figure-ground) image. *2nd row: Left:* Classical internal medial-axis approximation; *Right:* external medialaxis; *3rd row: Left:* 2D shock graph; *Right:* interior medialness map; *4th row: Left:* recovered concave (blue dots) and convex (red dots) points; *Right:* final dominant point set (where interior medial points are in green) obtained via our method.

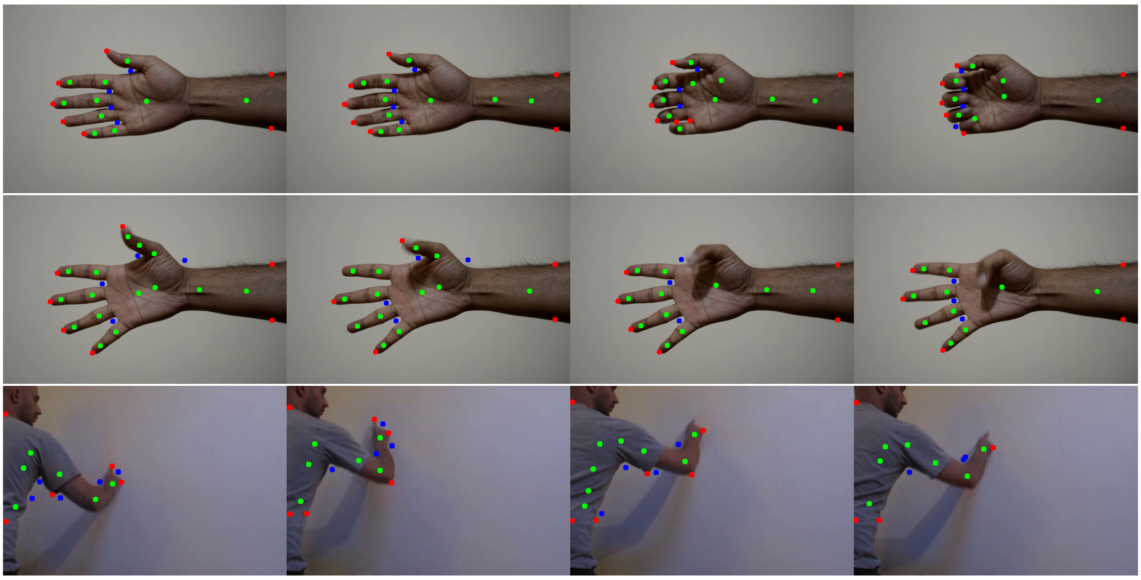


Figure 7.9: Some examples of tracking three gestures of a hand and (bottow row) of the artist performing a graffiti tracing.

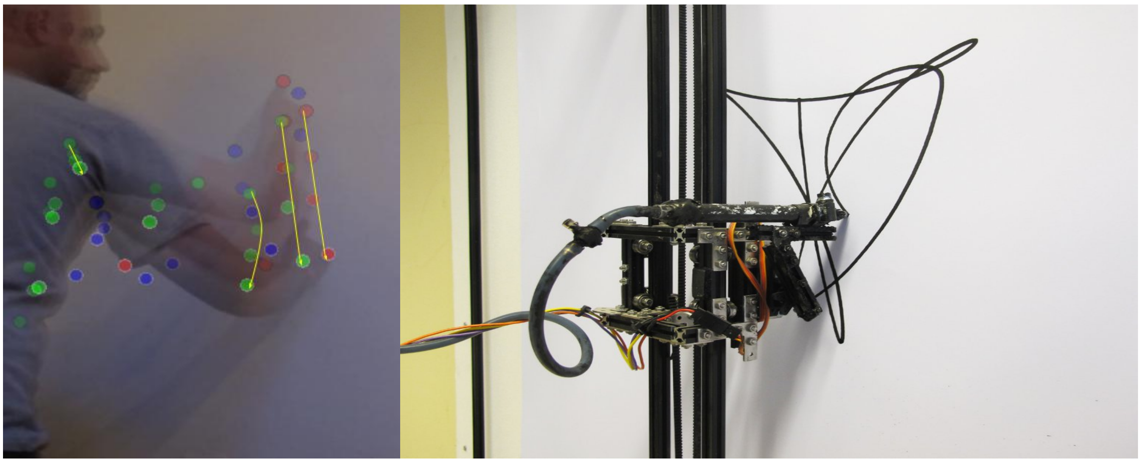


Figure 7.10: Illustration of the artist performing a graffiti tag and its transfer to a plotter. Graffiti artist: Daniel Berio.

In order to recognise strokes and their associated gestures, we need to develop a concise yet robust representation, which is adequate for computer vision methods. This can inform both an analysis of the human performance and drive a robotic platform to reproduce or learn by example how to create a similar artistic piece (Figures 7.10 & 7.11).

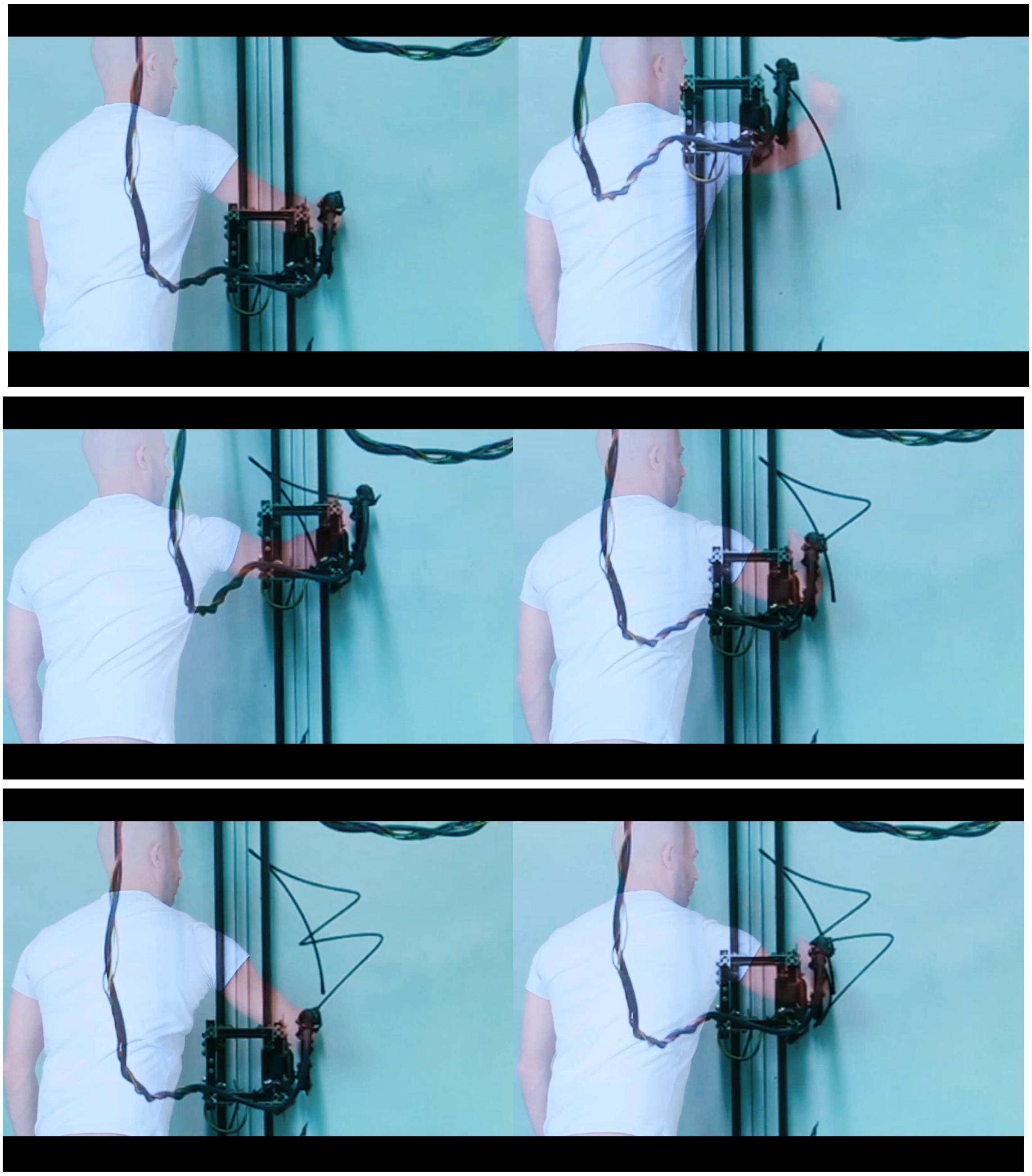


Figure 7.11: An early-stage experiment on gesture transfer from a graffiti artist to the pen-plotter using the medialness approach. Here, artist's frame is overlaid on plotter's frame, where plotter follows the dominant point of palm. Graffiti artist: Daniel Berio.

We have presented early results in applying our proposed point-based medialness representation to support marker-less gesture recognition from image sequences (Leymarie et al. (2014a)). In Figure 7.10, we illustrate our current work on tracking over time persistent dominant points in order to identify useful waypoints which can then be fed into our robotic platforms under the Sigma Lognormal Model (SLM) of Plamondon et al. (2013) to generate gesture traces alike those performed by the human artist. The early graffiti writing robots were simple Cartesian platforms. As a future work, experiments with articulated arms as well as with visual feedback can be planned.

7.2.3 Understanding Artistic Images

In this section, we explore the application of medialness as a representation substrate for a class of works of visual art. Here, we show an early study, which is only meant as an entry into the subject and thus clearly not exhaustive. We apply our framework the painting on two important 20th century artists, Picasso and Matisse, who exploited the power of line drawing and simplet object representations in conveying finished art works. While we apply our model on those images, we propose that such detailed analysis can be made more explicit in the form of our medialness representation and extracted dominant points.[3]

Figure 7.12 shows the application of our framework on the famous “Les Demoiselles d’Avignon” painting by Pablo Picasso, where we show: (a) the five female figures in the piece, (b) which we refer to by numerals 1, 2, 3, 4, 5 from left to right (from the observer’s vantage point), (c) the computed medialness field in between the bounding canvas rectangular limits and the regions exterior to the female figures;[4] (d) the exterior medialness field for the bottom-right figure; (e) the medialness field for the interior of all

[3] Our earlier works in this direction is recently accepted in the journal of *Art and Perception,* titled: “Medialness and the Perception of Visual Art”

[4] The roles of exterior and interior regions can be interchanged, e.g. if the focus is on the “negative spaces” (such as in architecture Leymarie and Kimia (2008)).

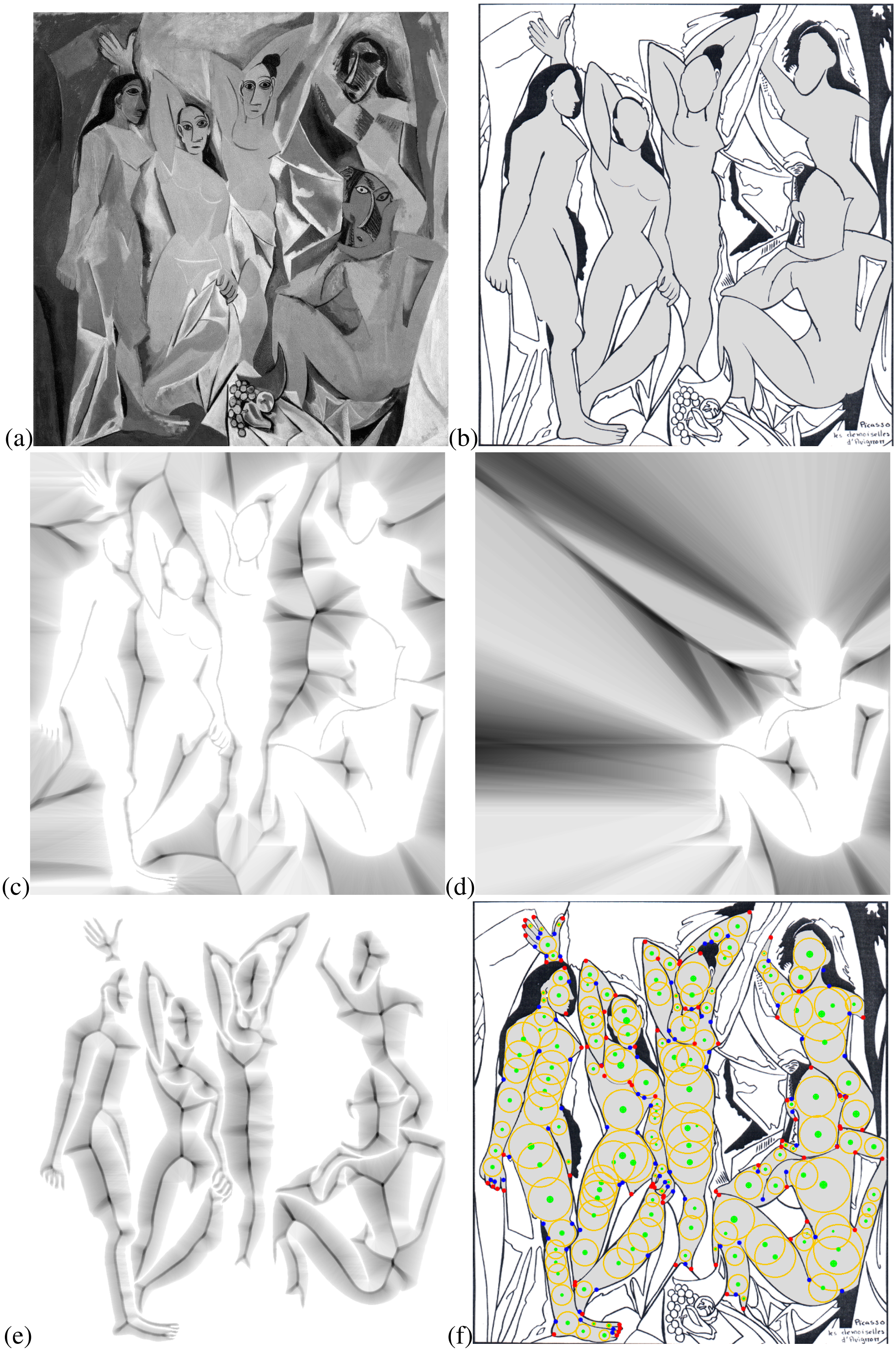


Figure 7.12: (a) Les Demoiselles d'Avignon, 1907, Pablo Picasso (as a grey level photo). (b) Approximate segmentation (using skin colors from the original). (c) Exterior medialness field. (d) Exterior medialness for female 4. (e) Interior medialness fields. (f) Feature points retrieved for the five demoiselles (vis.#1).

five female figures; and (f) the result of our recovery of the three types of feature points based on medialness (a visualisation we refer to as “vis.#1”).

In Leymarie and Aparajeya (2016), we propose that such feature points and their underlying medialness map can be further used to perform more careful studies of the artworks comprised of different objects, for example Les Demoiselles. We also approximate ridge following on the medialness field (thick varying green paths) to link the various feature points. In this visualisation (we refer to as “vis.#2”), we show significant convexities with red arrows (single headed) and significant concavities with blue arrows (double headed); the orientation of these arrows corresponds to the direction of the associated end of ridges of medialness. We note that our choices of parameters and thresholds remain purely experimental and can only be used with care as a more in depth study over series of works will be required to provide perhaps some systematic methods of parameter selection.

Figure 7.13 illustrates a series of three Picasso drawings from the 1940’s, mainly centered around the female form (originals can be found in the 1950 book by Bouret and Picasso (1950)) which is further cleaned-up by Koenderink et al. (2012) for their extensive study on cartoon-style line drawings. Their result produced distinct analyses of these drawings, where they were particularly interested in capturing the 3D percepts that human observes report. One of their analyses can be seen in this figure, which is based on boundary fitting of drawn stroke. Further analyses of curvature and pairing with boundary fitting indicates strong medial-axes type cues and associated circular primitives, such as for the buttocks of the female bodies. Our medialness framework not only produces very similar results but also provides much finer detailed analyses, as we compute medialness for a larger set of line traces (we do not require carefully drawn strokes). Our medialness can be computed from partial data, points or line segments. In these drawings, the long linear structures of the arms is made explicit by the ridges of medialness; important body

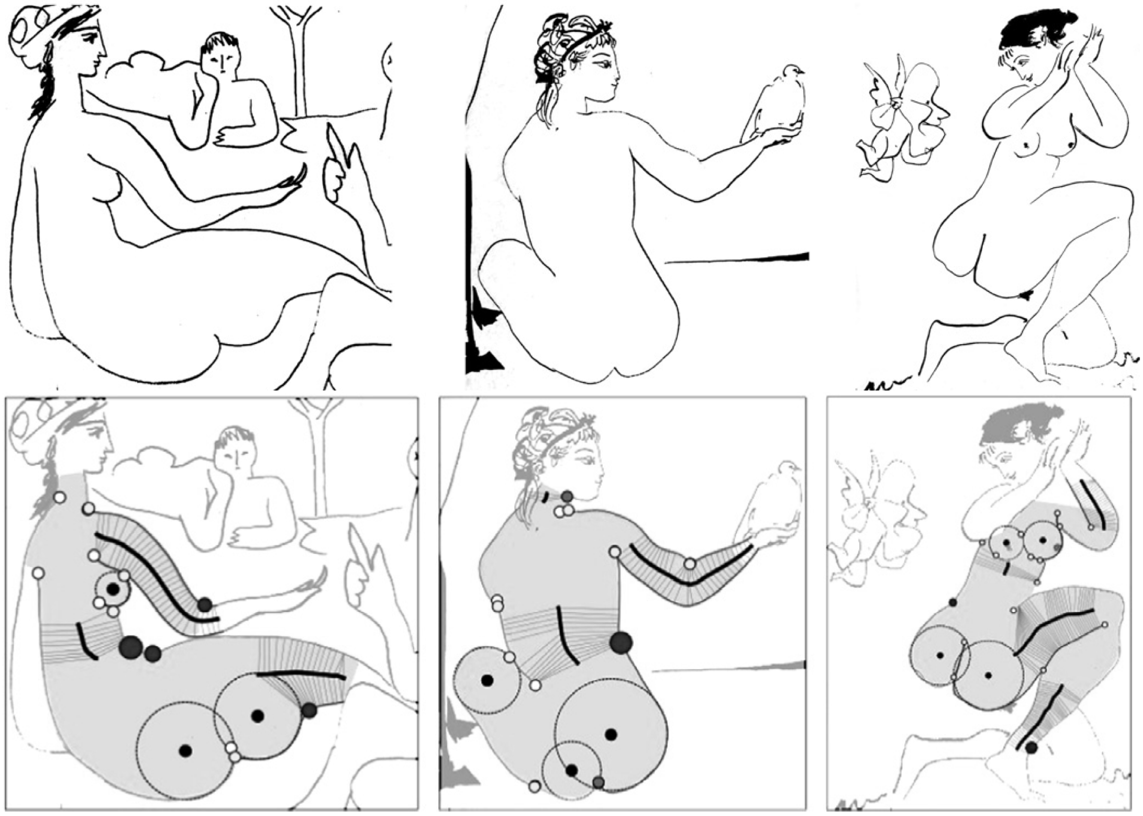


Figure 7.13: Top row: Three 1940's Picasso drawings of the female form (adapted from Koenderink et al. (2012)). Bottom row: Contour-based analysis highlighting certain pairings (resulting in medial axis segments) and important circles of curvature (convexities) and external concavities (shown as darken small disks, where the size reflects the significance of the concavity, while the shade/color reflects the type); adapted from Koenderink et al. (2012), Figure 20 — originals in color.

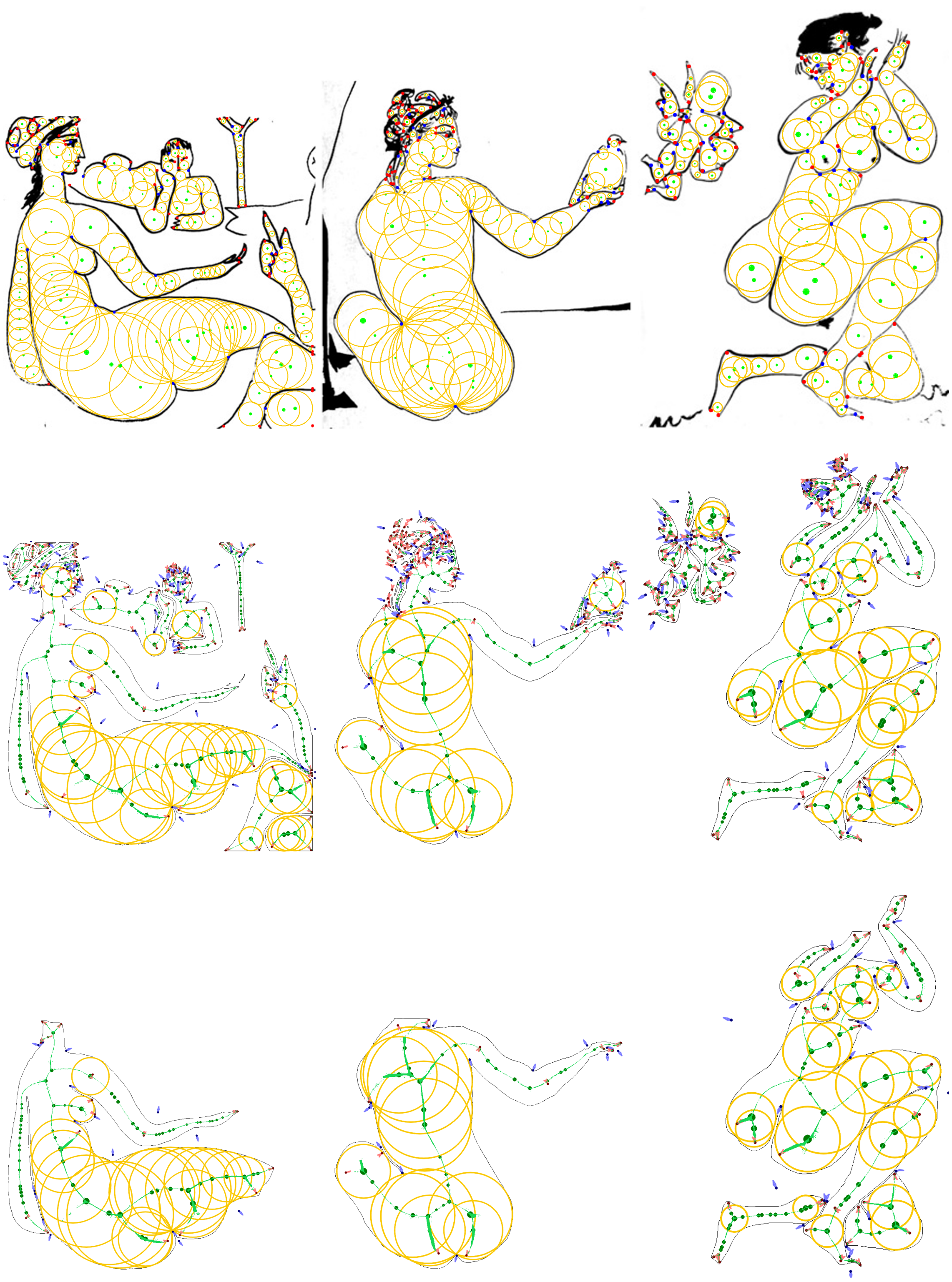


Figure 7.14: Women Drawn Series processed; top row: vis.#1, middle and bottom rows: vis.#2 with and without heads and details.

parts, e.g. buttocks, bulging knees, breasts, are well captured as hot spots with associated medial disks, Figure 7.14.

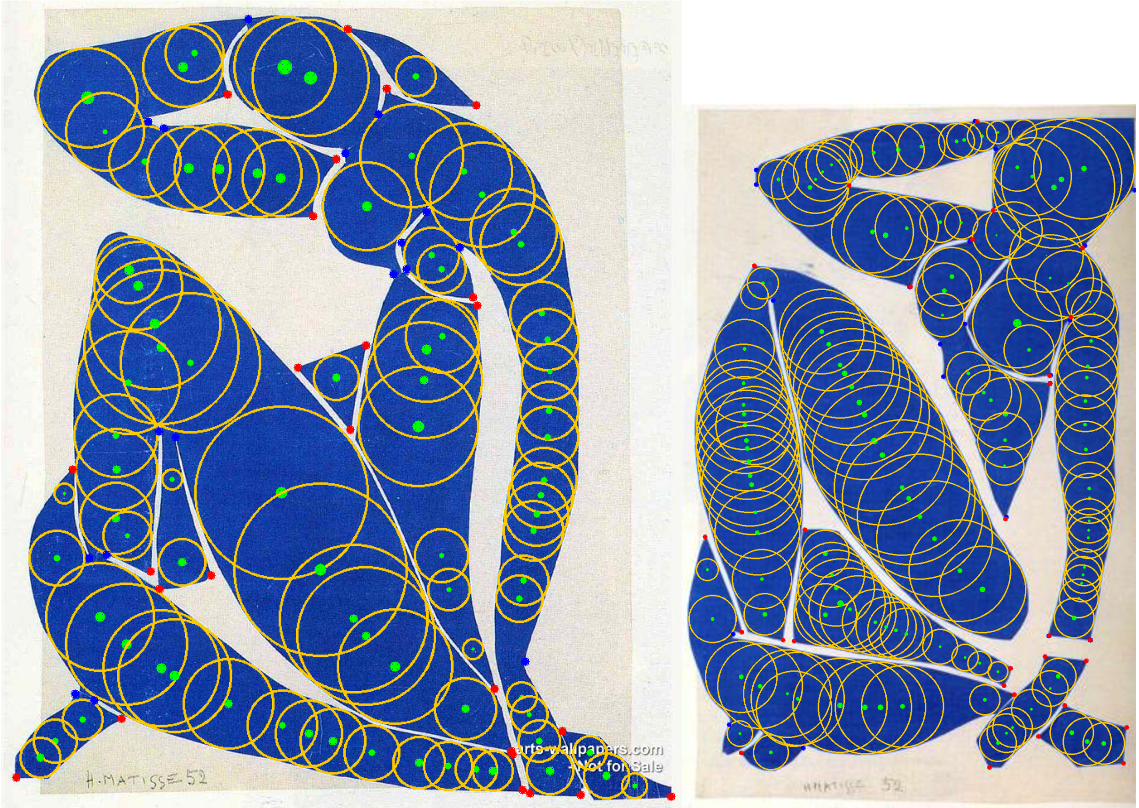


Figure 7.15: Matisse: Women Cut-Outs Series. 1952. Our medialness analysis is shown, superimposed (vis.#1).

Another example, as shown in Figure 7.15, presents some works by Henri Matisse (another important artist of the 20th century) along with our dominant points represented in different colors. In this figure, we have two examples of the famous series of blue cuts in which Matisse explored the female form.

In essence, another extension of the thesis could be in the field of visual art and perception. As a future step, more careful studies are needed to specify useful parametric ranges, useful visualisation modes, and have artists, art historians, and other specialists use and comment on the framework when seen as a toolbox for exploring the works of other artists or of one's own art.

7.2.4 Other Potential Domains/Applications

The generic nature of our framework opens many opportunities where the system can be applied. Apart from the above mentioned extensions and future works, the framework can also be extended to the following applications: Shape compression and reconstruction, Psychophysics, and 3D.

7.2.4.1 Shape Compression and Reconstruction

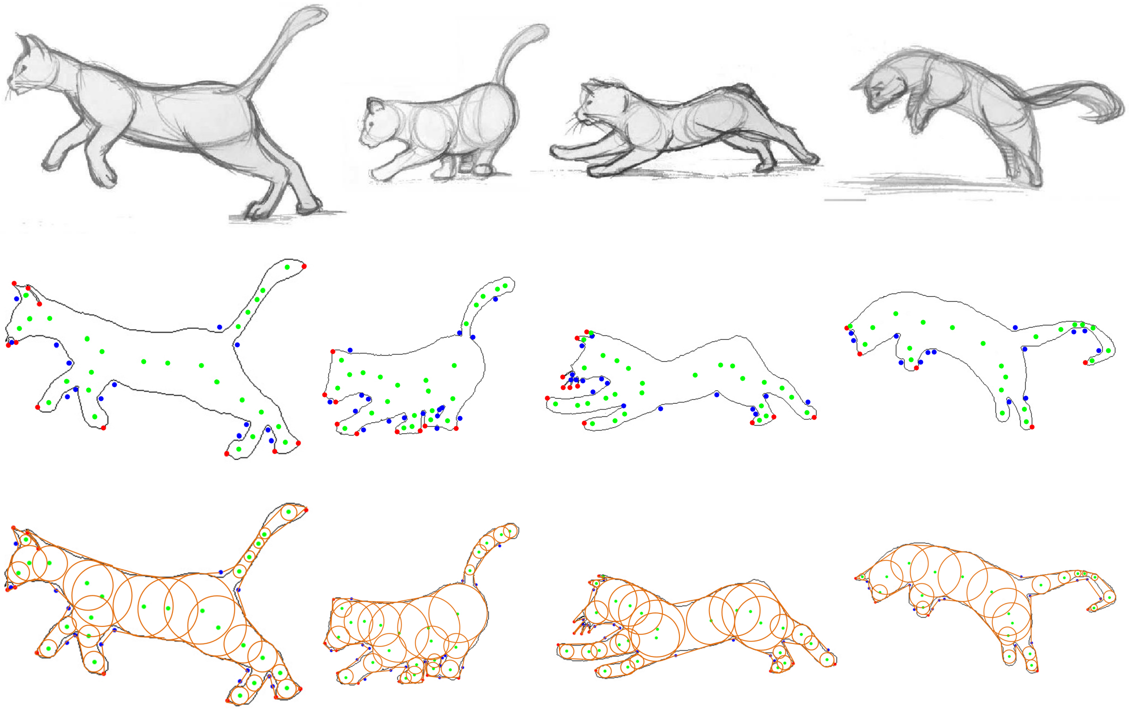


Figure 7.16: Top row: An artistic way to draw animal forms, here of a playful cat (Artist: Mr. Kelvin Chow). Middle row: Our proposed shape representation in terms of dominant medial (in green) and contour (convex (red) and concave (blue)) points. Bottom row: A possible set of contour reconstructions of the moving cat using our proposed point-based medialness representation.

Compression is usually considered part of signal (image) processing, where it is defined as minimizing the size in bytes of a (graphics) file without degrading the quality of the original data to an unacceptable level. The reduction in file size allows more data to be stored in a given amount of disk or memory space. It also reduces the time required for it to be sent over the Internet or downloaded from Web pages.

In the current context, our focus is mainly on the 2D shapes in the given image, which we represent as the point-based medialness field followed by dominant points extraction. Each of these dominant points hold the minimum radial distance along with the 2D Cartesian location and medailness information. A given shape can be compressed to these few number of dominating feature points with associated information. Our early prototype shows a naive reverse-engineering applied on different pose of cat-builds, where the approximate (currently lossy) reconstruction has been shown in Figure 7.16. Here, we join two circles (two radii of neighbouring internal dominant points) by drawing tangents. We implemented a heuristic that joins internal-internal, concave-internal, concave-concave, convex-internal, convex-concave, and convex-concave points that are in neighbours. As a future work, more careful studies are needed to achieve lossless and accurate reconstruction of the shape.

7.2.4.2 Psychophysical Investigation for Illusory Images

One of the classic problem in vision psychology is to model the perception of illusory contours (ICs) that reproduces the human response. ICs constitute ’simple’ stimuli for the investigation of vision (Figure 7.17). They have attracted considerable attention in the past for a number of reasons. First, ICs provide access to mental operations that link sensation and perception by generating experiences in the absence of physically present information. Second, ICs can be used to understand binding mechanisms and their perceptual consequences, particularly because IC stimuli can be readily used in experiments

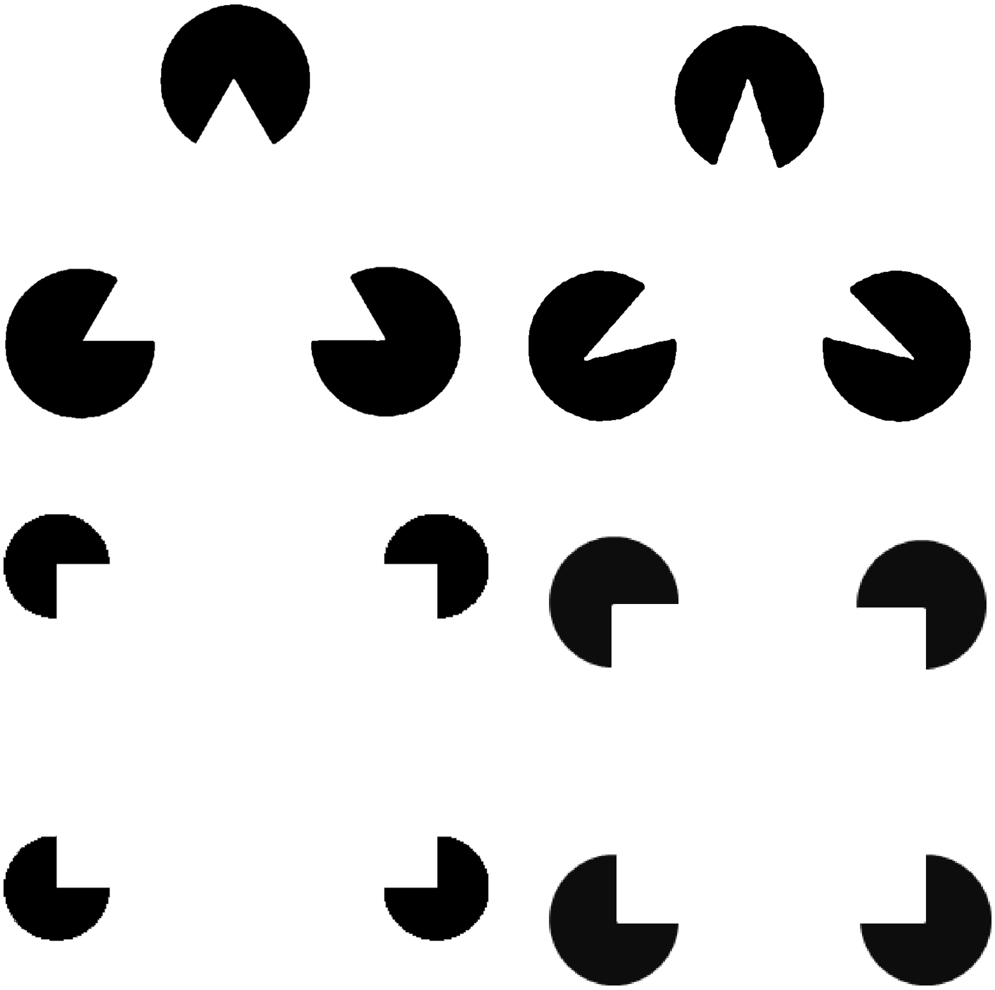


Figure 7.17: Examples of illusory contours. *Top-Left*: Kanizsa Triangle; *Top-Right*: Kanizsa Triangle Squeezed; *Bottom-Left*: Kanizsa Rectangle; and *Bottom-Right*: Kanizsa Square.

across artificial vision (Nieder (2002); Murray and Herrmann (2013)).

We perform our medialness measure on various ICs images. For example, Figure 7.18 is our medialness measure over the ICs shapes in Figure 7.17. Figure 7.19 shows an example of a normal triangle and a Kanizsa triangle. While looking at the middle of figure, one can easily observe that the Kanizsa's medialness representation reflects the normal triangle's medialness representation along with additional medialness line extending to infinity (corresponding to gaps int he triangle contour). Moreover, in the bottom part

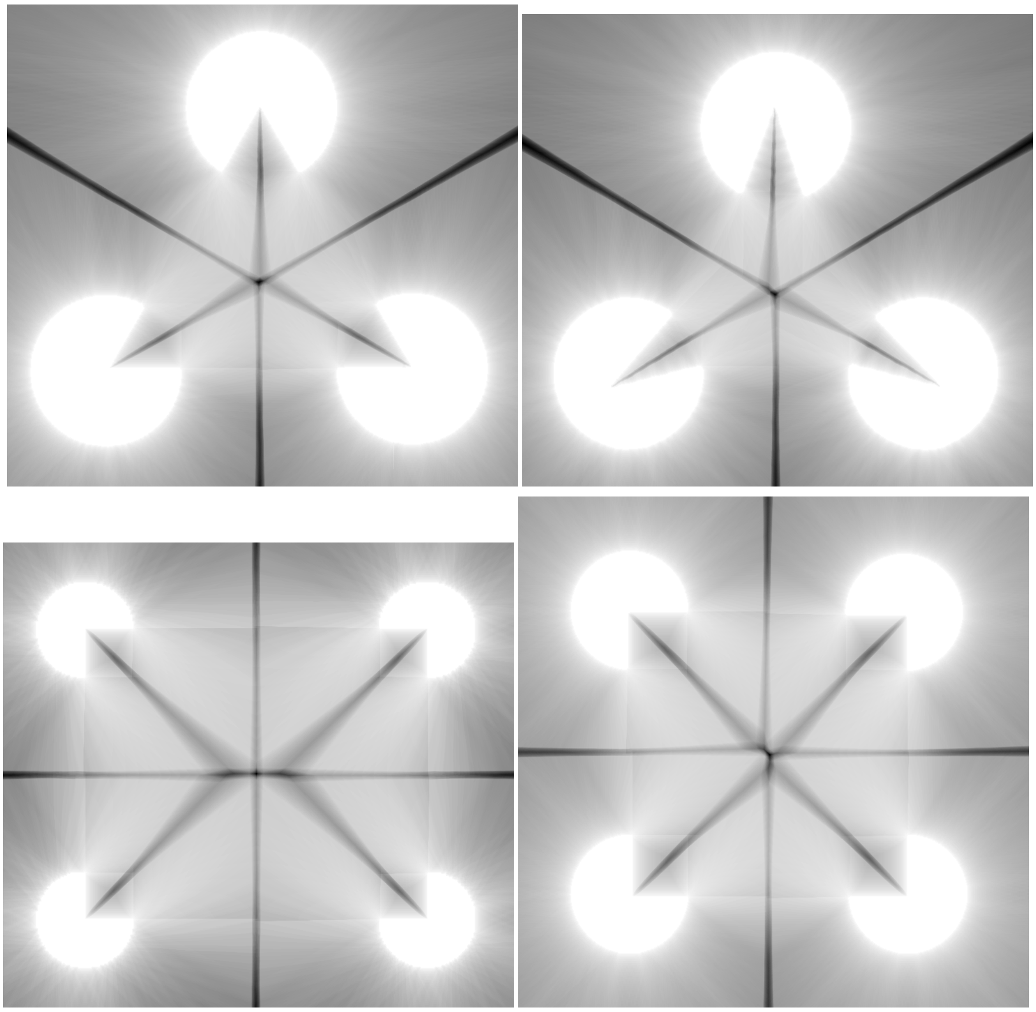


Figure 7.18: Our medialness field on illusory contour images on (a) Kanisza triangle, (b) squeezed Kanisza triangle, (c) Kanisza rectangle, and (d) Kanisza Square.

of the figure, all three convexities as well as the circumcentre is retained. As a future work, this part needs further thoughtful study to establish the connection of medialness and ShIFT features with the ICs perceived shapes.

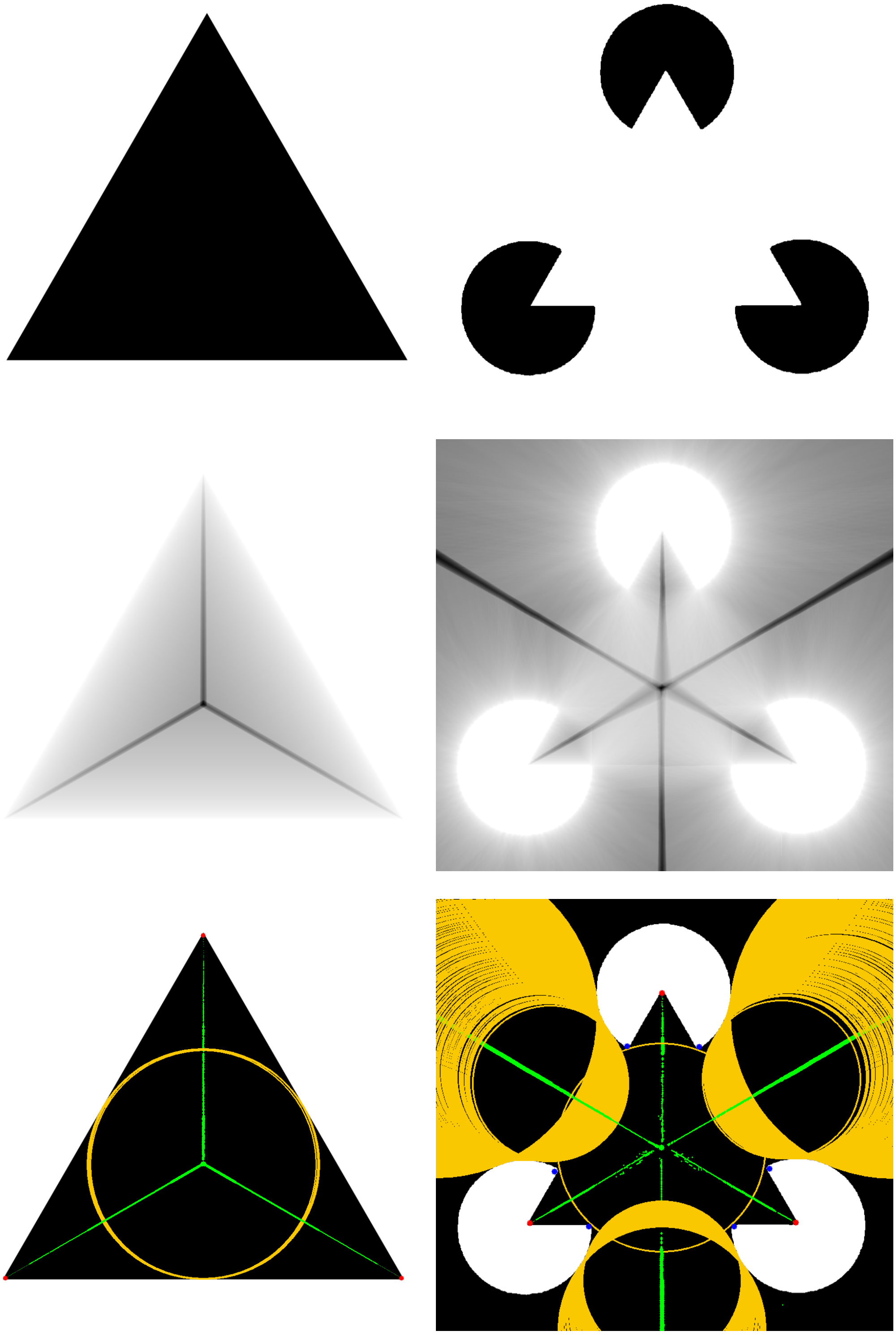
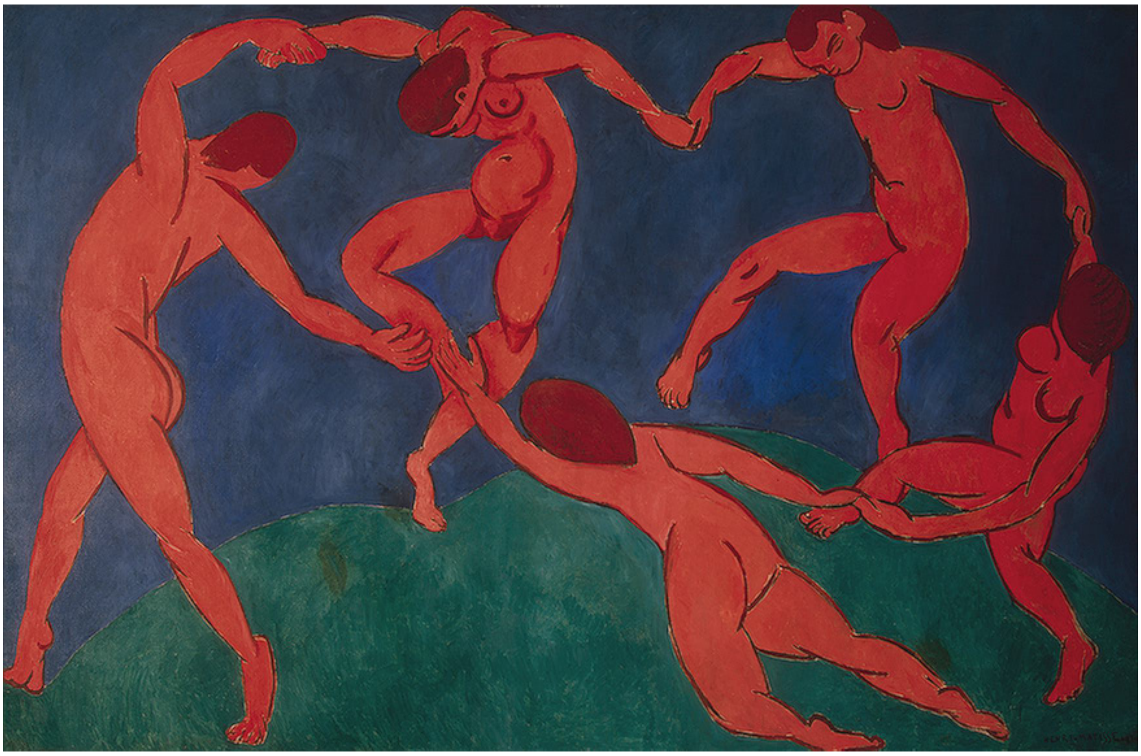


Figure 7.19: Our medialness field on illusory contour images. *Left*: Moving from top-to-bottom, the column shows a simple triangle, its medialness field and dominant points (green – internal dominant points with yellow circles indicating their minimal radial distances, and red – convex points. The similar representation had been shown on *Right,* but here the input is a Kanisza triangle.

7.2.4.3 Medialness-based 3D object description and recognition

It would be worth the effort to investigate how such perception driven 2D shape definition and recognition could be extended to work with 3D object models. There might be many ways to address this problem and here we can think of two obvious ones: (i) converting our 2D annulus into some 3D structuring element and finding medialness in the 3D space; or (ii) converting the 3D object into several 2D views and combining them via some heuristics (Yasseen et al. (2015)).



Henry Matisse, *Dance,* 1910. @The Hermitage, St. Petersburg.

Appendix A

Accepted Papers and Posters

Publications (Journals and Conference Papers)

1. Leymarie, F. F. and Aparajeya, P. (2016). Medialness and the Perception of Visual Art. in the journal of *Art and Perception*, Brill. (Accepted)

2. Aparajeya, P. and Leymarie, F. F. (2015). Point-based Medialness for 2D Shape Description and Identification. in the journal of *Multimedia Tools and Application (MTAP)*, Springer, 75 (3):1667–99.

3. Aparajeya, P. and Petresin, V. and Leymarie, F. F. and Rueger, S. (2015). Movement description and gesture recognition for live media arts. in Proceedings of the *12th European Conference on Visual Media Production (CVMP).* pages 19–20. ACM.

4. Berio, D. and Aparajeya, P. and Leymarie, F. F. (2015). Computational Models for the Analysis and Synthesis of Graffiti Art. in *Visual Science of Art Conference (VSAC).* Liverpool.

5. Aparajeya, P. and Leymarie, F. F. (2014). Point-based medialness for animal and plant identification. In Vrochidis, S. et al., editors, *Proceedings of the 1st Inter-*

national Workshop on Environnmental Multimedia Retrieval co-located with ACM International Conference on Multimedia Retrieval, volume 1222 of *EMR '14*, pages 14–21, Glasgow, UK. CEUR-WS.org.

6. Leymarie, F. F. and Aparajeya, P. and MacGillivray, C. (2014). Point-based Medialness for Movement Computing. in ACM *International Workshop on Movement and Computing (MOCO)*. pages 31–36. ACM.

7. Leymarie, F. F. and Aparajeya, P. and Berio, D. (2014). Towards Human-Robot Gesture Recognition using Point-based Medialness. in *Real Time Gesture Recognition for Human Robot Interaction (GRHRI)*, pages 366–371.

8. Chiara Piccoli, Prashant Aparajeya, Georgios Th Papadopoulos, John Bintliff, Frederic Fol Leymarie, Philip Bes, Mark van der Enden, Jeroen Poblome, and Petros Daras. "Towards the automatic classification of pottery sherds: two complementary approaches." In Across Space and Time. in the 41st Computer Applications and Quantative Methods in Archaeology (CAA) Conference. 2013.

Posters

1. Aparajeya, P. and Petresin, V. and Leymarie, F. F. and Rueger, S. (2015). Movement description and gesture recognition for live media arts. in the *12th European Conference on Visual Media Production (CVMP),* London.

2. Aparajeya, P. and Leymarie, F. F. and Rueger, S. (2015). Multimedia Information Retrieval Based on Shape. in *20 Years of Knowledge Media*, Open University, MK.

3. Leymarie, F. F. and Aparajeya, P. and Rueger, S. (2015). Multimedia Information Retrieval Based on Shape. in the *2nd workshop on Visual Image Interpretation in Humans and Machines (ViiHM)*, Bath.

4. Leymarie, F. F. and Aparajeya, P. and Berio, D. (2014). Point-based Medialness for Shape Understanding. in the *1st workshop on Visual Image Interpretation in Humans and Machines (ViiHM)*, Stratford, UK.

Appendix B

Top Hat Transform

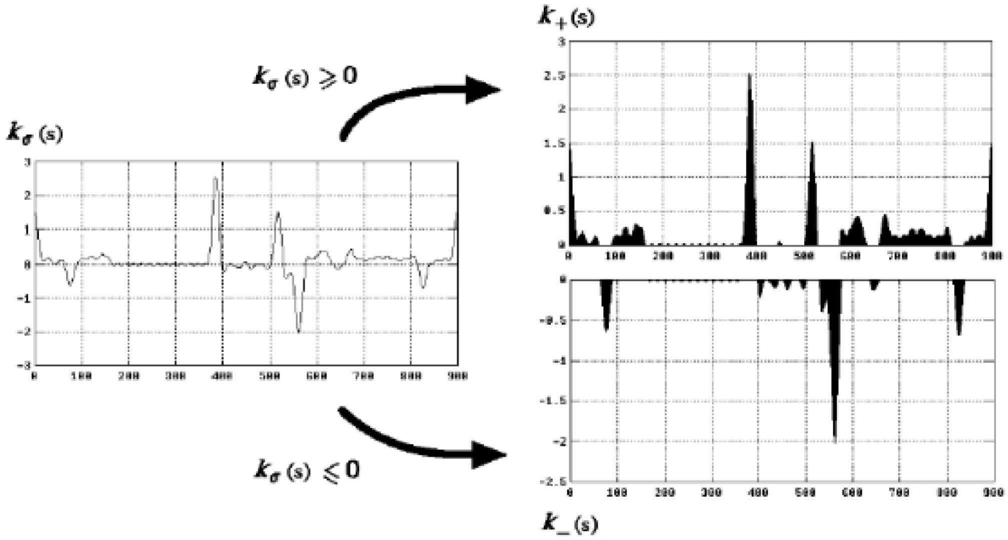
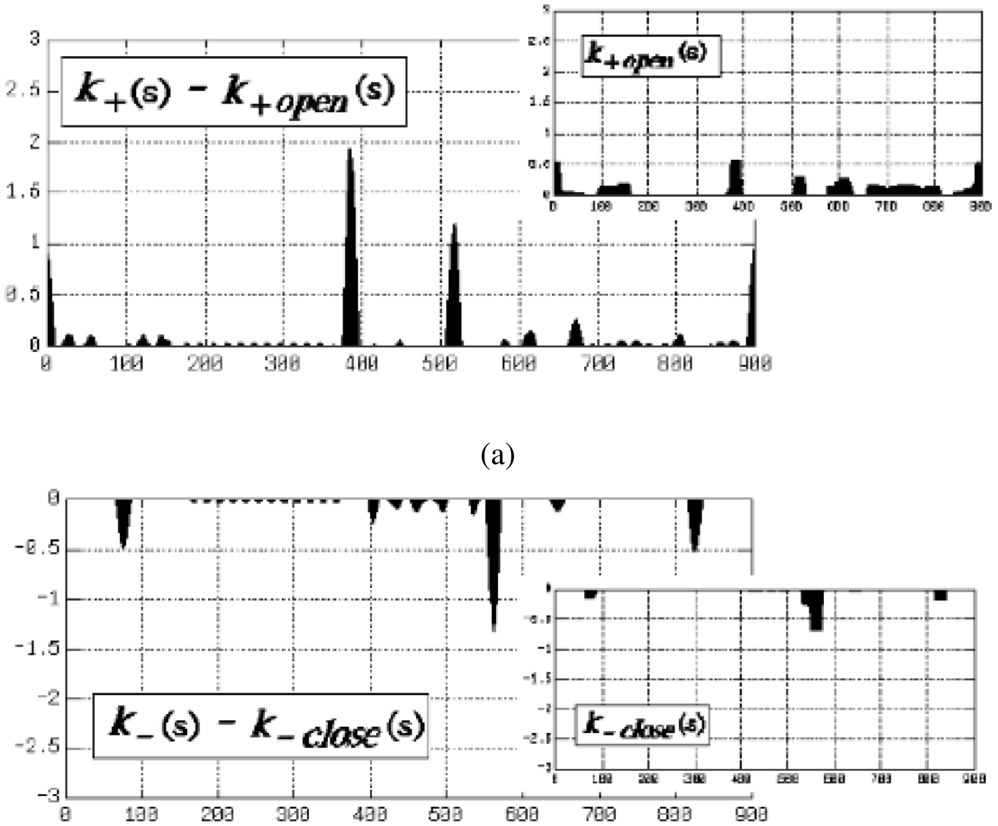


Figure B.1: Curvature function $k_\sigma(s)$ as the sum of two functions: $k_+(s)$, and $k_-(s)$, adopted from Leymarie and Levine (1988).

Mathematical morphology applied to functions such as $f = k_\sigma(s)$ provides us with useful tools for the extraction of primitives or dominant shapes found in such functions. Two dual key operations are *erosion* and *dilation*. For a signal (or image), erosion ($\ominus$) is a shrinking operation while dilation ($\oplus$) is an expanding operation. These operations are performed locally by observing the structure of the neighborhood at point of the function.



(a)

(b)

Figure B.2: (a) Top-hat transform of the input signal (curvature function $k_\sigma(s)$), and (b) Bottom-hat transform of the input signal (curvature function $k_\sigma(s)$), adopted from Leymarie and Levine (1988).

Combinations of dilation and erosion forms two set of complementary morphological operations: *closing* and *opening*. Opening is defined as the dilation of an eroded function, i.e., *dilation followed by erosion*, while Closing is the erosion of the dilated function, i.e., *erosion followed by dilation*. For an input binary image f and structuring element s, closing ($f \bullet s$) and opening ($f \circ s$) operation are mathematically denoted as:

$$f \bullet s = (f \oplus s) \ominus s \tag{B.1}$$

$$f \circ s = (f \ominus s) \oplus s \tag{B.2}$$

Opening removes convexities or *bumps* of increasing size with the use of different sized structuring elements. Closing, on the other hand, is used to fill-in concavities or *holes* in the input signal.

Openings and closings can be used to derive useful image operations. Using the difference between original signal, significant informations can be extracted from the signal. Top-hat transform is another morphological operation that is used for extracting small or narrow, bright or dark features in an image. It is useful when variations in the background mean that this cannot be achieved by a simple threshold. Top-hat transform has two types: White top-hat (or simply top-hat) transform (Figure B.2(a)), and black top-hat (or bottom-hat) transform (Figure B.2(b)). White top-hat transform $T_w(f)$ is defined as the difference between the input signal and its opening by a structuring element, while black top-hat or bottom-hat transform $T_b(f)$ as he difference between the closing and the input image. Mathematically:

$$T_w(f) = f - f \circ s \tag{B.3}$$

$$T_b(f) = f \bullet s - f \tag{B.4}$$

Appendix C

B+ Tree: A Dynamic Index Structure

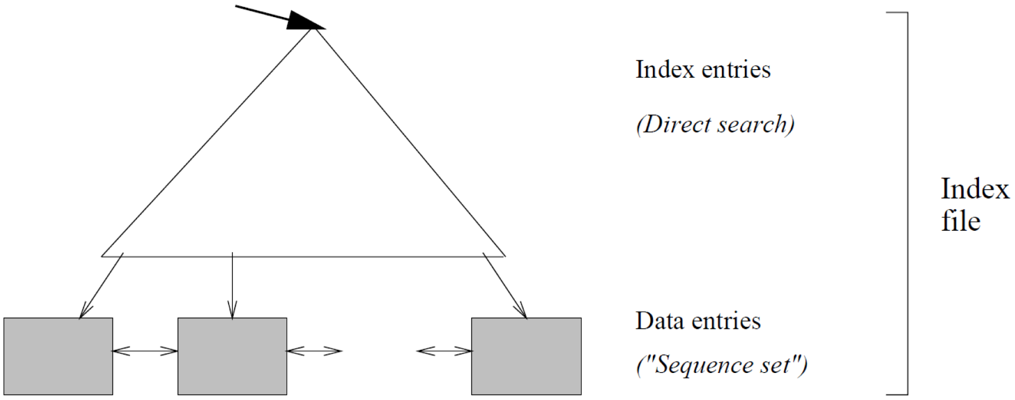


Figure C.1: Structure of a B+ Tree

The B+ tree is a dynamic structure that is used for database indexing purpose. It adjusts well to changes and supports both equality and range queries. The static structure like “indexed sequential access method” (ISAM) suffers from the problem that long overflow chains can develop as the database size grows, resulting in poor performance. The B+ tree search structure overcomes this problem by allowing more flexible, dynamic structures that adjust gracefully to inserts and deletes. The tree structure is balanced in which the internal nodes direct the search and the leaf nodes contain the data entries. Since the tree structure grows and shrinks dynamically, it is not feasible to allocate the leaf entries sequentially as in ISAM, where the set of primary leaf entries was static. In order to retrieve all leaf entries efficiently, it is required to link them using pointers. By organizing them into a doubly linked list, we can easily traverse the sequence of leaf entries (called also as the sequence set) in either direction. This structure is illustrated in Figure C.1.

Following are some of the main characteristics of a B+ tree:

1. Insert and delete operations on the tree keep it balanced.

2. A minimum occupancy of 50% is guaranteed for each node except the root if the deletion algorithm is implemented. However, deletion is often implemented by simply locating the data entry and removing it, without adjusting the tree as needed to guarantee the 50% occupancy, because database typically grow rather than shrink.

3. Searching for a particular value requires just a traversal from the root to the appropriate leaf. The length of a path from the root to a leaf (any leaf, because the tree is balanced) is referred as the height of the tree.

Let us assume that every node (except the root node) in the B+ tree contains m entries, where $d \leq m \leq 2d$. Here the parameter d is called the **order** of the tree, which is a measure of the capacity of a tree node. The root node is only the exception to this requirement on the number of entries; for the root it is simply $1 \leq m \leq 2d$. For the space overhead of storing the index entries, we obtain all the advantages of a sorted entry plus efficient insertion and deletion algorithms. B+ tree typically maintain 67% space occupancy.

Format of a Node

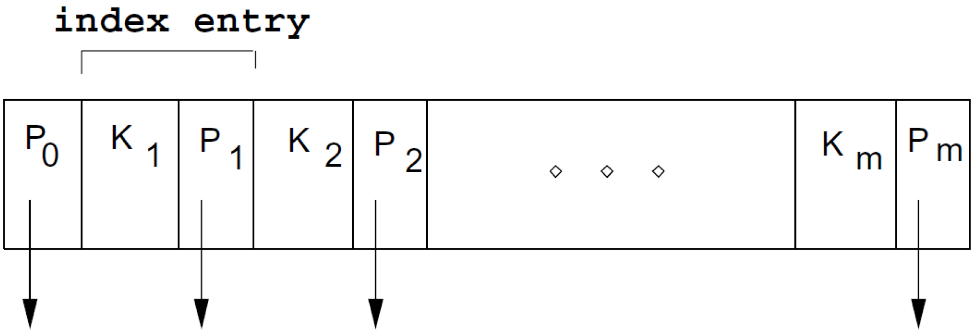


Figure C.2: A node structure of the B+ Tree.

Format of the B+ tree has been shown in Figure C.2. Non-leaf nodes with m *index entries* contain $m+1$ pointers to children. Pointer P_i points to a subtree in which all key values K are such that $K_i \leq K < K_{i+1}$. As special cases, P_0 points to a tree in which all key values are less than K_1, and P_m points to a tree in which all key values are greater than or equal to K_m. Regardless of the alternative chosen for leaf entries, the leaf pages are chained together in a doubly linked list. Thus, the leaves form a sequence, which can be used to answer range queries efficiently.

Search

The search algorithm finds the leaf node in which a given data entry belongs. Pseudocode of the algorithm is given in AlgorithmC.1.

Insert

The algorithm for insertion takes an entry, finds the lead node where it belongs, and inserts it there. Pseudocode for the B+ tree insertion algorithm is given in AlgorithmC.2.

Algorithm C.1 Algorithm for Search

```
procedure search(value $V$) returns node-pointer $P$
  //Given a search key value, finds its leaf node
  return tree_search(root-node, $V$);              //searches from root
end procedure

procedure tree_search(node-pointer $P$, value $V$) returns node-pointer $P$
  //Searches tree for entry
  if $P$ is a leaf
    if there is a value $K$ in $P$ such that $V = K$
      return $P$;
    else
      no record with value $V$ exists;
      return;
  else
    if $V < V_1$
      return tree_search($P_0, V$);
    else
      if $V \geq V_m$                              //m = #entries
        return tree_search($P_m, V$);
      else
        find $i$ such that $V_i \leq V < V_{i+1}$;
        return tree_search($P_i, V$);
end procedure
```

The insert algorithm is recursive and inserts the entry at the appropriate child node. In case, when the node is full, it gets split and an entry pointing to the node created by the split must be inserted into its parent; this entry is pointed by the *newchildentry* pointer. If the root is needed to split, then a new root node is created and the height of the tree is increased by one. If a split occurs at the leaf level, however, we have to retrieve a neighbor in order to adjust the previous and next-neighbor pointers with respect to the newly created leaf node.

Delete

Deletion task takes an entry, finds the leaf node where it belongs, and deletes it. Pseudocode for the B+ tree deletion is given in Algorithm C.3. The algorithm is recursive and deletes the entry from the appropriate child node. Usual way is to go down to the leaf node (from root node) where the entry belongs, remove the entry from there, and return all the way back to the root node. Sometimes, before deletion operation, a node can be at its minimum occupancy. Deletion, in this particular case, causes it to go below the minimum occupancy. Such cases are handled by redistributing entries from an adjacent sibling or merging the node with a sibling to maintain minimum occupancy. When the redistribution occurs, the parent node must get updated in order to reflect this change, i.e., the key value in the index entry pointing to the second node must be changed to the lowest search key in the second node. When the merging happens, again, the parent node get updated to reflect this change, which is done by deleting the index entry for the second node. If the last entry in the root node is deleted, the height of the B+ tree decreases by 1.

Algorithm C.2 Algorithm for Insertion into B+ Tree of Order *d*

```
procedure insert_entry(node-pointer, entry, newchildentry)
  // Inserts entry into subtree with root '*nodepointer'; degree is d;
  // 'newchildentry' is null initially, and null upon return unless child is split

  if *nodepointer is a non-leaf node, say $N$
    find $i$ such that $K_i \leq$ entry's key value $< K_{i+1}$;     // choose subtree
    insert($P_i$, entry, newchildentry);                        // insert entry recursively
    if newchildentry is null
      return;                        //don't split child
    else                          // split child, must insert *newchildentry in N
      if $N$ has space
      put *newchildentry on it;
      set newchildentry to null;
      return;
    else                    // note difference with respect to splitting of leaf page
      split $N$:           // 2d + 1 key values and 2d + 2 nodepointers
      first $d$ key values and $d+1$ nodepointers stay;
      last $d$ keys and $d+1$ pointers move to new node, $N2$;
      // *newchildentry set to guide searches between N and N2
      newchildentry = & (<smallest key value on $N2$, pointer to $N2$>);
      if $N$ is the root                  // root node was just split
        create new node with <pointer to $N$, *newchildentry>;
        make the tree's root-node pointer point to the new node;
      return;

if *nodepointer is a leaf node, say $L$
  if $L$ has space
    put entry on it;
    set newchildentry to null;
    return;
  else,
    split $L$: first $d$ entries stay, rest move to the new node $L2$;
    newchildentry = & (<smallest key value on $L2$, pointer to $L2$>);
    set sibling pointers in $L$ and $L2$;
    return;
end procedure
```

Algorithm C.3 Algorithm for Deletion from B+ Tree of Order *d*

```
procedure delete(parentpointer, nodepointer, entry, oldchildentry)
// Deletes entry from subtree with root '*nodepointer'; degree is d;
// 'oldchildentry' null initially, and null upon return unless child deleted
if *nodepointer is a non-leaf node, say N
   find i such that $K_i \leq$ entry's key value $< K_{i+1}$;     // choose subtree
   delete(nodepointer, $P_i$, entry, oldchildentry);        // recursive delete
   if oldchildentry is null
      return;                                   // child not deleted
   else                                         // child node is discarded
      remove *oldchildentry from N;             // check for minimum occupancy
      if N has entries to spare
         set oldchildentry to null;
         return;
      else // note difference with respect to merging of leaf pages
         get a sibling S of N:                  // parentpointer is used to find S
         if S has extra entries
            redistribute evenly between N and S through parent;
            set oldchildentry to null;
            return;
         else
            merge N and S;
            oldchildentry = & (current entry in parent for M);   // call node on rhs as M
            pull splitting key from parent down into node on left;
            move all entries from M to node on left;
            discard empty node M;
            return;
if *nodepointer is a leaf node, say L
   if L has entries to spare
      remove entry;
      set oldchildentry to null;
      return;
   else,
      get a sibling S of L;                     // parentpointer is used to find S
      if S has extra entries
         redistribute evenly between L and S;
         find entry in parent for node on right, i.e., M;
         replace key value in parent entry by new low-key value in M;
         set oldchildentry to null;
         return;
   else,
      merge L and S;
      oldchildentry = & (current entry in parent for M);
      move all entries from M to node on left;
      discard empty node M;
      adjust sibling pointers;
      return;
end procedure
```

Appendix D

Visualisation modes

First mode: vis.#1

- The minimum separation between selected hot spots (dominant medialness loci) is set to $2 \times \varepsilon$ (twice the annulus operator's width).

- We indicate an associated medial disk (radius+annular width) using a yellow circle. The centre is colored green and the corresponding dot size reflects the medialness value.

- Both concavities and convexities measures are performed by doing *contour* analysis. To do such an analysis, we used two operators

  - Length of support – This basically avoids small bumpy regions in the shape; and

  - Threshold angle – This limits the angle (of opening) of the concavity or convexity.

- Detected concavities (blue dots) and convexities (red dots) are projected on the contour.

Second mode: vis.#2

- The minimum separation between selected hot spots (dominant medialness loci) is set to ε (the annulus operator's width). This tends to generate more dominant points (hot spots) than for vis.#1. It brings us closer to a medial axis graph structure as ε becomes smaller. The centre is colored green and the corresponding dot size reflects the medialness value.

- Again, we indicate an associated medial disk (radius+annular width) using a yellow circle, but we do this only for those hot spots whose medialness value is equal or more than 80% of the maximum value (for a given image). Thus only the top 20% of hot spots having contour support are illustrated.

- Both concavities and convexities measures are done by analysing medialness (concavity - external medialness, convexity - internal medialness) values. Only ends of medialness ridges are considered as candidates convexities/concavities. The equivalent of "length of support" in terms of medialness is used to decide on which candidates to keep.

- We use blue and red arrows to indicate the orientations of concave dominant and convex dominant points respectively.

- The ridge trace left from the hat-transform filters is thinned downed and showed as a green trace of varying thickness (still reflective of the local medialness values). This visualises an approximate path of medialness linking the various features.

Appendix E

List of Symbols

Symbols	Meaning
2D	Two Dimensions
3D	Three Dimensions
$p(x,y)$	x- and y- coordinate of a point/pixel p in 2D Cartesian system
ε	Tolerance value (constant) in Kovács et al. (1998) representation
ε_p	Tolerance value (variable) at point $p(x,y)$ in our representation
δ_i	Kronecker delta
δ	Step angle
∂b	Contribution of a boundary point b to $\widehat{S_i}$
ϕ	Orientation of boundary point b
θ	Rotation
θ_{th}	Rotation tolerance or threshold value
κ	Smallest measurable length in the given image
ω	Window's width for hole filling
ψ_ε	Response value
α_p	Orientation of a dominant point p with respect to +x-axis

α_p^+	Tolerant orientation of dominant point p in clockwise direction
α_p^-	Tolerant orientation of dominant point p in anti-clockwise direction
β	Scale
β_{th}	Scale tolerance of threshold value
$\|.\|$	Cardinality of a set
b	boundary point
b_{th}	ShIFT bin threshold for bin binarization
D_ε	Medialness function described by Kovács et al. (1998)
D_ε^+	Interior medialness function (our modified representation)
D_ε^-	Exterior medialness function (our modified representation)
D_ε^*	Either D_ε^+ or D_ε^-
e	Euler's constant
F	F- measure
H	4×4 matrix for homography calculation
k	Number of successive filled ShIFT-bins
l	weight of a boundary pixel
M	Set of matching feature points
$\{M_I, M_E, M_C\}$	Matching $\{$Internal,External,Convex$\}$ feature points
n	Number of ShIFT bins
$\mathbb{N}^+$	Positive natural numbers
$p^{'}$	Transformation of dominant point p with respect to matrix H
$\overrightarrow{q_xt_x}$	x-translation
$\overrightarrow{q_yt_y}$	y-translation
q_i	A feature point from set Q
Q	Set of feature points from a query (test) sample
$\{Q_I, Q_E, Q_C\}$	Query $\{$Internal,External,Convex$\}$ feature points

$R(p)$	Minimum radial distance
$\widehat{S}_i$	i^{th} annular sector
t_i	A feature point from set T
T	Set of feature points from a trained (target) image
$\{T_I, T_E, T_C\}$	Target {Internal,External,Convex} feature points
v_b	Normal at boundary point b
w_{type}	weight of dominant type "type" (type = I, E, C)
W	Maximum value in ShIFT bins

References

Abbasi, S., Mokhtarian, F., and Kittler, J. (2000). Enhancing css-based shape retrieval for objects with shallow concavities. *Image and Vision Computing*, 18(3):199–211.

Amit, Y. (2002). *2D Object Detection and Recognition: Models, Algorithms, and Networks*. The MIT Press.

Amit, Y., Grenander, U., and Piccioni, M. (1991). Structural image restoration through deformable templates. *Journal of the American Statistical Association*, 86(414):376–387.

Amor, H. B., Berger, E., Vogt, D., and Jung, B. (2009). Kinesthetic bootstrapping: Teaching motor skills to humanoid robots through physical interaction. In *KI 2009: Advances in Artificial Intelligence*, number LNAI 5803, pages 492–9. Springer.

Aparajeya, P. and Leymarie, F. F. (2014). Point-based medialness for animal and plant identification. In Vrochidis, S. et al., editors, *Proceedings of the 1st International Workshop on Environnmental Multimedia Retrieval co-located with ACM International Conference on Multimedia Retrieval*, volume 1222 of *EMR '14*, pages 14–21, Glasgow, UK. CEUR-WS.org.

Aparajeya, P. and Leymarie, F. F. (2015). Point-based medialness for 2d shape description and identification. *Multimedia Tools and Applications*, 75(3):1667–1699.

Arnheim, R. (1974). *Art and Visual Perception: A Psychology of the Creative Eye*. University of California Press, new version expanded and revised edition of the 1954 original edition.

Aslan, C., Erdem, A., Erdem, E., and Tari, S. (2008). Disconnected skeleton: shape at its absolute scale. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(12):2188–2203.

Attneave, F. (1954). Some informational aspects of visual perception. *Psychological Review*, 61(3):183–193.

Aurenhammer, F. (1991). Voronoi diagrams–a survey of a fundamental geometric data structure. *ACM Computing Surveys (CSUR)*, 23(3):345–405.

Bai, X. and Latecki, L. J. (2008). Path similarity skeleton graph matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(7):1282–1292.

Bai, X., Latecki, L. J., and Liu, W.-Y. (2007). Skeleton pruning by contour partitioning with discrete curve evolution. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(3):449–462.

Bai, X., Liu, W., and Tu, Z. (2009a). Integrating contour and skeleton for shape classification. In *IEEE 12th International Conference on Computer Vision (ICCV) Workshops*, pages 360–367. IEEE.

Bai, X., Rao, C., and Wang, X. (2014). Shape vocabulary: A robust and efficient shape representation for shape matching. *Image Processing, IEEE Transactions on*, 23(9):3935–3949.

Bai, X., Wang, X., Latecki, L. J., Liu, W., and Tu, Z. (2009b). Active skeleton for non-rigid object detection. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 575–582. IEEE.

Bai, X., Yang, X., Yu, D., and Latecki, L. J. (2008). Skeleton-based shape classification using path similarity. *International Journal of Pattern Recognition and Artificial Intelligence*, 22(04):733–746.

Bauckhage, C. and Tsotsos, J. K. (2005). Bounding box splitting for robust shape classification. In *ICIP (2)*, pages 478–481.

Bay, H., Ess, A., Tuytelaars, T., and Gool, L. V. (2008). Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359.

Bay, H., Tuytelaars, T., and Van Gool, L. (2006). Surf: Speeded up robust features. In *Computer Vision–ECCV 2006*, pages 404–417. Springer.

Belongie, S., Malik, J., and Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(4):509–522.

Berg, A. C. and Malik, J. (2006). *Shape matching and object recognition*. Springer.

Berger, M. (2009). *Geometry I*. Springer Science & Business Media.

Berio, D. and Leymarie, F. F. (2015). Computational models for the analysis and synthesis of graffiti tag strokes. In *Symposium on Expressive Graphics, Istanbul*. ACM Eurographics.

Berretti, S., Bimbo, A. D., and Pala, P. (2000). Retrieval by shape similarity with perceptual distance and effective indexing. *IEEE Transactions on Multimedia*, 2(4):225–239.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94(2):115.

Biederman, I. (2001). Recognizing depth-rotated objects: A review of recent research and theory. *Spatial Vision*, 13(2–3):241–253.

Bishko, L. (1993). Relationships between Laban Movement Analysis and computer animation. In *Dance and Technology I: Moving Toward the Future*, pages 1–9. Fullhouse, University of Wisconsin, Madison, USA.

Bishko, L. (2007). The uses and abuses of cartoon style in animation. *Animation Studies Online Journal*, 2:24–35.

Biswas, S., Aggarwal, G., and Chellappa, R. (2010). An efficient and robust algorithm for shape indexing and retrieval. *Multimedia, IEEE Transactions on*, 12(5):372–385.

Blum, H. (1962a). An associative machine for dealing with the visual field and some of its biological implications. *Biological Prototypes and Synthetic Systems*, pages 244–260.

Blum, H. (1962b). A machine for performing visual recognition by use of antenna-propagation concepts. *Institute of Radio Engineers, Wescon Convention Record.*

Blum, H. (1967). A transformation for extracting new descriptors of shape. *Symposium on Models for the perception of speech and visual form*, 19(5):362–380.

Blum, H. (1973). Biological shape and visual science. *Journal of Theoretical Biology*, 38(2):205–287.

Blum, H. and Nagel, R. N. (1978). Shape description using weighted symmetric axis features. *Pattern recognition*, 10(3):167–180.

Bouret, J. and Picasso, P. (1950). *Picasso: dessins*. Ed. des deux mondes.

Bregler, C., Loeb, L., Chuang, E., and Deshpande, H. (2002). Turning to the masters: Motion capturing cartoons. *ACM Transactions on Graphics*, 21(3):399–407.

Brown, M., Szeliski, R., and Winder, S. (2005). Multi-image matching using multi-scale oriented patches. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005.*, volume 1, pages 510–517. IEEE.

Burbeck, C. A. and Pizer, S. M. (1995). Object representation by cores: Identifying and representing primitive spatial regions. *Vision research*, 35(13):1917–1930.

Burt, P. J. (1980). Fast, hierarchical correlations with gaussian-like kernels. Technical report, DTIC Document.

Calabi, L. and Hartnett, W. E. (1968). Shape recognition, prairie fires, convex deficiencies and skeletons. *The American Mathematical Monthly*, 75(4):335–342.

Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., and Fua, P. (2012). Brief: Computing a local binary descriptor very fast. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(7):1281–1298.

Calonder, M., Lepetit, V., Strecha, C., and Fua, P. (2010). Brief: Binary robust independent elementary features. In *Computer Vision–ECCV 2010*, pages 778–792. Springer.

Caputo, B., Muller, H., Thomee, B., Villegas, M., Paredes, R., Zellhofer, D., Goeau, H., Joly, A., Bonnet, P., Gomez, J. M., et al. (2013). ImageCLEF 2013: The vision, the data and the open challenges. In *Information Access Evaluation. Multilinguality, Multimodality, and Visualization*, pages 250–268. Springer.

Celebi, M. E. and Aslandogan, Y. A. (2005). A comparative study of three moment-based shape descriptors. In *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, volume 1, pages 788–793. IEEE.

Chandrasekhar, V., Takacs, G., Chen, D. M., Tsai, S. S., Reznik, Y., Grzeszczuk, R., and Girod, B. (2012). Compressed histogram of gradients: A low-bitrate descriptor. *International Journal of Computer Vision*, 96(3):384–399.

Chang, C., Liu, W., and Zhang, H. (2001). Image retrieval based on region shape similarity. In *Photonics West 2001-Electronic Imaging*, pages 31–38. International Society for Optics and Photonics.

Chen, G. and Bui, T. D. (1999). Invariant fourier-wavelet descriptor for pattern recognition. *Pattern recognition*, 32(7):1083–1088.

Chen, L., Feris, R., and Turk, M. (2008). Efficient partial shape matching using Smith-Waterman algorithm. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08.*, pages 1–6. IEEE.

Chi, D., Costa, M., Zhao, L., and Badler, N. (2000). The emote model for effort and shape. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 173–182, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.

Choi, W.-P., Lam, K.-M., and Siu, W.-C. (2003). Extraction of the euclidean skeleton based on a connectivity criterion. *Pattern Recognition*, 36(3):721–729.

Chow, Y., Grenander, U., and Keenan, D. (1989). Hands, a pattern theoretic study of biological shapes. *Research Mongraph. Brown University, Providence, RI.*

Chuang, G. C. and Kuo, C. J. (1996). Wavelet descriptor of planar curves: Theory and applications. *Image Processing, IEEE Transactions on*, 5(1):56–70.

Chui, H. and Rangarajan, A. (2003). A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2):114–141.

Cope, J., Corney, D., Clark, J., Remagnino, P., and Wilkin, P. (2012). Plant species identification using digital morphometrics: A review. *Expert Systems with Applications*, 39(8):7562–7573.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE.

Davies, E. R. (2004). *Machine vision: theory, algorithms, practicalities*. Elsevier.

De Berg, M., Van Kreveld, M., Overmars, M., and Schwarzkopf, O. C. (2000). *Computational geometry*. Springer.

De Winter, J. and Wagemans, J. (2008). Perceptual saliency of points along the contour of everyday objects: A large-scale study. *Perception & Psychophysics*, 70(1):50–64.

di Baja, G. S. and Thiel, E. (1994). (3, 4)-weighted skeleton decomposition for pattern representation and description. *Pattern Recognition*, 27(8):1039–1049.

Di Ruberto, C. (2004). Recognition of shapes by attributed skeletal graphs. *Pattern Recognition*, 37(1):21–31.

Dougherty, E. R. and Lotufo, R. A. (2003). *Hands-On Morphological Image Processing*. Tutorial Texts in Optical Engineering, Vol. TT59. SPIE Publications.

Elmasri, R. (2011). *Fundamentals of database systems*. Addison-Wesley, Boston.

Felzenszwalb, P. F. (2005). Representation and detection of deformable shapes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(2):208–220.

Felzenszwalb, P. F. and Schwartz, J. D. (2007). Hierarchical matching of deformable shapes. In *Computer Vision and Pattern Recognition, 2007. CVPR ’07. IEEE Conference on*, pages 1–8. IEEE.

Ferrari, V., Fevrier, L., Jurie, F., and Schmid, C. (2008). Groups of adjacent contour segments for object detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(1):36–51.

Ferrari, V., Jurie, F., and Schmid, C. (2010). From images to shape models for object detection. *International Journal of Computer Vision*, 87(3):284–303.

Firestone, C. and Scholl, B. J. (2014). "please tap the shape, anywhere you like": Shape skeletons in human vision revealed by an exceedingly simple measure. *Psychological Science*, 25(2):377–386.

Förstner, W. (1986). A feature based correspondence algorithm for image matching. *International Archives of Photogrammetry and Remote Sensing*, 26(3):150–166.

Forsyth, D. A. and Ponce, J. (2002). *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference.

Foster, S. L. (1986). *Reading Dancing: Bodies and Subjects in Contemporary American Dance*. University of California Press.

Gopalan, R., Turaga, P., and Chellappa, R. (2010). Articulation-invariant representation of non-planar shapes. In *Proceedings of the 11th European Conference on Computer Vision – ECCV'10*, Lecture Notes in Computer Science (LNCS), pages 286–299, Berlin, Heidelberg. Springer-Verlag.

Grunbaum, B. (2003). *Convex polytopes*. Springer, New York.

Guay, M., Cani, M.-P., and Ronfard, R. (2013). The Line of Action: An intuitive interface for expressive character posing. *ACM Transactions on Graphics (TOG)*, 32(6):Article no. 205.

Gudivada, V. N. and Raghavan, V. V. (1995). Content based image retrieval systems. *Computer*, 28(9):18–22.

Guest, A. H. (1998). *Choreo-graphics: a comparison of dance notation systems from the fifteenth century to the present*. Psychology Press.

Han, J., Kamber, M., and Pei, J. (2011). *Data mining: concepts and techniques: concepts and techniques*. Elsevier.

Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Citeseer.

Hatori, Y. and Sakai, K. (2014). Early representation of shape by onset synchronization of border-ownership-selective cells in the v1-v2 network. *JOSA A*, 31(4):716–729.

Heinly, J., Dunn, E., and Frahm, J.-M. (2012). Comparative evaluation of binary features. In *Computer Vision–ECCV 2012*, pages 759–773. Springer.

Hung, C.-C., Carlson, E. T., and Connor, C. E. (2012). Medial axis shape coding in macaque inferotemporal cortex. *Neuron*, 74(6):1099–1113.

Jain, A. K., Duin, R. P., and Mao, J. (2000). Statistical pattern recognition: A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(1):4–37.

Jain, A. K., Zhong, Y., and Lakshmanan, S. (1996). Object matching using deformable templates. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(3):267–278.

Kass, M., Witkin, A., and Terzopoulos, D. (1988). Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331.

Kayaert, G., Wagemans, J., and Vogels, R. (2011). Encoding of complexity, shape, and curvature by macaque infero-temporal neurons. *Frontiers in Systems Neuroscience*, 5(51).

Kelly, M. F. and Levine, M. D. (1993). *The symmetric enclosure of points by planar curves*. McGill Research Center for Intelligent Machines.

Kelly, M. F. and Levine, M. D. (1995a). Annular symmetry operators: A method for locating and describing objects. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 1016–1021, Cambridge, UK. IEEE.

Kelly, M. F. and Levine, M. D. (1995b). Region-based grouping operations for locating & describing objects. In *Vision Interface*, volume 95, pages 15–19.

Khalil, M. I. and Bayoumi, M. M. (2001). A dyadic wavelet affine invariant function for 2d shape recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(10):1152–1164.

Kim, H. S., Park, K. H., and Kim, M. (1987). Shape decomposition by collinearity. *Pattern Recognition Letters*, 6(5):335–340.

Kim, J. and Shontz, S. (2010). An improved shape matching algorithm for deformable objects using a global image feature. *Advances in Visual Computing*, pages 119–128.

Kimia, B. B. (2003). On the role of medial geometry in human vision. *Journal of Physiology – Paris*, 97(2):155–190.

Kimia, B. B., Tannenbaum, A. R., and Zucker, S. W. (1995). Shapes, shocks, and deformations i: the components of two-dimensional shape and the reaction-diffusion space. *International journal of computer vision*, 15(3):189–224.

Koenderink, J., van Doorn, A., and Wagemans, J. (2012). Picasso in the mind's eye of the beholder: Three-dimensional filling-in of ambiguous line drawings. *Cognition*, 125(3):394–412.

Kotelly, J. C. (1963). Mathematical model of blum's theory of pattern recognition. *AIR FORCE CAMBRIDGE RESEARCH LABS*, pages 63–164.

Kovács, I. (2010). "Hot spots" and dynamic coordination in Gestalt perception. In *Dynamic Coordination in the Brain: From Neurons to Mind*, Strüngmann Forum Reports, chapter 14, pages 215–228. MIT Press.

Kovács, I., Fehér, Á., and Julesz, B. (1998). Medial-point description of shape: a representation for action coding and its psychophysical correlates. *Vision research*, 38(15):2323–2333.

Kovacs, I. and Julesz, B. (1993). A closed curve is much more than an incomplete one: Effect of closure in figure-ground segmentation. *Proceedings of the National Academy of Sciences*, 90(16):7495–7497.

Kovacs, I. and Julesz, B. (1994). Perceptual sensitivity maps within globally defined visual shapes. *Nature*, 370(6491):644–646.

Larese, M. G., Namías, R., Craviotto, R. M., Arango, M. R., Gallo, C., and Granitto, P. M. (2014). Automatic classification of legumes using leaf vein image features. *Pattern Recognition*, 47(1):158–168.

Latecki, L. J. and Lakämper, R. (1999). Convexity rule for shape decomposition based on discrete contour evolution. *Computer Vision and Image Understanding*, 73(3):441–454.

Latecki, L. J. and Lakamper, R. (2000). Shape similarity measure based on correspondence of visual parts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1185–1190.

Latecki, L. J., Lakamper, R., and Eckhardt, T. (2000). Shape descriptors for non-rigid shapes with a single closed contour. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 424–429. IEEE.

Layton, O. W., Mingolla, E., and Yazdanbakhsh, A. (2014). Neural dynamics of feedforward and feedback processing in figure-ground segregation. *Frontiers in Psychology*, 5(Article 972). Perception Science Series.

Lehmann, A., Leibe, B., and Van Gool, L. (2011). Fast prism: Branch and bound hough transform for object class detection. *International journal of computer vision*, 94(2):175–197.

Lehoux, N. (2013). Dance literacy and digital media: Negotiating past, present and future representations of movement. *International Journal of Performance Arts and Digital Media*, 9(1):153–174.

Lepetit, V., Lagger, P., and Fua, P. (2005). Randomized trees for real-time keypoint recognition. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 775–781. IEEE.

Lepetit, V., Pilet, J., and Fua, P. (2004). Point matching as a classification problem for fast and robust object pose estimation. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages 244–244. IEEE.

Lescroart, M. D. and Biederman, I. (2013). Cortical representation of medial axis structure. *Cerebral cortex*, 23(3):629–637.

Leutenegger, S., Chli, M., and Siegwart, R. Y. (2011). Brisk: Binary robust invariant scalable keypoints. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2548–2555. IEEE.

Leymarie, F. and Levine, M. D. (1992). Simulating the grassfire transform using an active contour model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(1):56–75.

Leymarie, F. F. (2011). On the visual perception of shape analysis and genesis through information models.

Leymarie, F. F. and Aparajeya, P. (2016). Medialness and the perception of visual art. *Art and Perception*. Accepted.

Leymarie, F. F., Aparajeya, P., and Berio, D. (2014a). Towards human-robot gesture recognition using point-based medialness. pages 366–371.

Leymarie, F. F., Aparajeya, P., and MacGillivray, C. (2014b). Point-based medialness for movement computing. In *Proceedings of the 2014 International Workshop on Movement and Computing (MOCO)*, pages 31–36, IRCAM, Paris, France. ACM.

Leymarie, F. F. and Kimia, B. B. (2008). From the infinitely large to the infinitely small. In *Medial representations*, pages 327–351. Springer.

Leymarie, F. F. and Levine, M. (1988). *Curvature morphology*. Citeseer, Computer Vision and Robotics Laboratory, McGill University, Montreal, Quebec, Canada.

Leyton, M. (1992). *Symmetry, Causality, Mind*. Bradford Books. MIT Press.

Lindeberg, T. (1998). Feature detection with automatic scale selection. *International journal of computer vision*, 30(2):79–116.

Ling, H. and Jacobs, D. W. (2005). Using the inner-distance for classification of articulated shapes. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 719–726. IEEE.

Ling, H. and Jacobs, D. W. (2007). Shape classification using the inner-distance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(2):286–299.

Ling, H. and Okada, K. (2007). An efficient earth mover's distance algorithm for robust histogram comparison. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(5):840–853.

Liu, H., Deng, M., and Xiao, C. (2011a). An improved best bin first algorithm for fast image registration. In *International Conference onElectronic and Mechanical Engineering and Information Technology (EMEIT)*, volume 1, pages 355–358. IEEE.

Liu, M., Vemuri, B. C., Amari, S.-I., and Nielsen, F. (2010). Total bregman divergence and its applications to shape retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3463–3468. IEEE.

Liu, T.-L. and Geiger, D. (1999). Approximate tree matching and shape similarity. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 456–462. IEEE.

Liu, Y. K., Wei, W., Wang, P. J., and Žalik, B. (2007). Compressed vertex chain codes. *Pattern Recognition*, 40(11):2908–2913.

Liu, Z., An, J., and Meng, F. (2011b). A robust point matching algorithm for image registration. In *Fourth International Conference on Machine Vision (ICMV 11)*, volume SPIE 8350. International Society for Optics and Photonics.

Loffler, G. (2008). Perception of contours and shapes: Low and intermediate stage mechanisms. *Vision research*, 48(20):2106–2127.

Loncaric, S. (1998). A survey of shape analysis techniques. *Pattern recognition*, 31(8):983–1001.

Loomis, A. (1951). *Successful Drawing*. Viking Books.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157. IEEE.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110.

MacGillivray, C. (2014). *Choreographing Time: Developing a System of Screen-less Animation*. PhD thesis, Goldsmtihs, University of London.

Manning, C. D., Raghavan, P., Schütze, H., et al. (2008). *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.

Marey, E. J. (1883). Emploi des photographies partielles pour etudier la locomotion de i'homme et des animaux. *Comptes rendus*, 96.

Martin, D. R., Fowlkes, C. C., and Malik, J. (2004). Learning to detect natural image boundaries using local brightness, color, and texture cues. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(5):530–549.

McNeill, G. and Vijayakumar, S. (2006). Hierarchical procrustes matching for shape retrieval. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 885–894. IEEE.

Mikolajczyk, K., Leibe, B., and Schiele, B. (2005). Local features for object class recognition. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1792–1799. IEEE.

Mikolajczyk, K. and Schmid, C. (2002). An affine invariant interest point detector. In *European Conference on Computer Vision (ECCV)*, pages 128–142. Springer.

Mikolajczyk, K. and Schmid, C. (2004). Scale & affine invariant interest point detectors. *International journal of computer vision*, 60(1):63–86.

Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1615–1630.

Miller, M. I., Christensen, G. E., Amit, Y., and Grenander, U. (1993). Mathematical textbook of deformable neuroanatomies. *Proceedings of the National Academy of Sciences*, 90(24):11944–11948.

Mokhtarian, F., Abbasi, S., and Kittler, J. (1997). Efficient and robust retrieval by shape content through curvature scale space. volume 8, pages 51–58. World Scientific.

Mokhtarian, F. and Mackworth, A. K. (1992). A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (8):789–805.

Mouine, S., Yahiaoui, I., and Verroust-Blondet, A. (2012). Advanced shape context for plant species identification using leaf image retrieval. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval (ICMR)*, number Article 49, Hong Kong, China. ACM.

Mouine, S., Yahiaoui, I., and Verroust-Blondet, A. (2013a). Plant species recognition using spatial correlation between the leaf margin and the leaf salient points. In *ICIP 2013-IEEE International Conference on Image Processing*. IEEE.

Mouine, S., Yahiaoui, I., and Verroust-Blondet, A. (2013b). A shape-based approach for leaf classification using multiscaletriangular representation. In *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, pages 127–134. ACM.

Mukundan, R., Ong, S., and Lee, P. A. (2001). Image analysis by tchebichef moments. *Image Processing, IEEE Transactions on*, 10(9):1357–1364.

Murray, M. M. and Herrmann, C. S. (2013). Illusory contours: a window onto the neurophysiology of constructing perception. *Trends in cognitive sciences*, 17(9):471–481.

Muybridge, E. (1887). *Animal Locomotion: An Electro-photographic Investigation of Consecutive Phases of Animal Movements; Prospectus and Catalogue of Plates*. University of Pennsylvania.

Muybridge, E. (1901). *The human figure in motion: an electro-photographic investigation of consecutive phases of muscular actions*. Chapman and Hall, London.

Muybridge, E. (1957). *Animals in motion*. Dover Publications, New York.

Muybridge, E. (1984). *The male and female figure in motion: 60 classic photographic sequences*. Dover Publications, New York.

Nanni, L., Lumini, A., and Brahnam, S. (2014). Ensemble of shape descriptors for shape retrieval and classification. *International Journal of Advanced Intelligence Paradigms*, 6(2):136–156.

Nasreddine, K., Benzinou, A., and Fablet, R. (2010). Variational shape matching for shape classification and retrieval. *Pattern Recognition Letters*, 31(12):1650–1657.

Nelson, R. C. and Selinger, A. (1998). A cubist approach to object recognition. In *Computer Vision, 1998. Sixth International Conference on*, pages 614–621. IEEE.

Nguyen, H. V. and Porikli, F. (2013). Support vector shape: A classifier-based shape representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):970–982.

Nieder, A. (2002). Seeing more than meets the eye: processing of illusory contours in animals. *Journal of Comparative Physiology A*, 188(4):249–260.

Osher, S. and Paragios, N. (2003). *Geometric level set methods in imaging, vision, and graphics*. Springer Science & Business Media.

Pan, Z., Xiao, G., Chen, K., and Li, Z. (2012). A spectral matching for shape retrieval using pairwise critical points. In *Foundations of Intelligent Systems*, pages 475–482. Springer.

Pang, Y., Li, W., Yuan, Y., and Pan, J. (2012). Fully affine invariant surf for image matching. *Neurocomputing*, 85:6–10.

Park, J., Hwang, E., and Nam, Y. (2008). Utilizing venation features for efficient leaf image retrieval. *Journal of Systems and Software*, 81(1):71–82.

Pavlidis, T. (1980). Algorithms for shape analysis of contours and waveforms. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (4):301–312.

Pelillo, M., Siddiqi, K., and Zucker, S. W. (1999). Matching hierarchical structures using association graphs. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(11):1105–1120.

Peterson, L. (2000). *Computer networks : a systems approach*. Morgan Kaufmann Publishers, San Francisco, Calif.

Peura, M. and Iivarinen, J. (1997). Efficiency of simple shape descriptors. *Aspects of visual form*, pages 443–451.

Plamondon, R., O'Reilly, C., Rémi, C., and Duval, T. (2013). The lognormal handwriter: learning, performing, and declining. *Frontiers in psychology*, 945(4):1–14.

Polat, U. and Sagi, D. (1993). Lateral interactions between spatial channels: suppression and facilitation revealed by lateral masking experiments. *Vision research*, 33(7):993–999.

Polat, U. and Sagi, D. (1994). The architecture of perceptual spatial interactions. *Vision research*, 34(1):73–78.

Poppe, R. (2010). A survey on vision-based human action recognition. *Image and vision computing*, 28(6):976–990.

Premachandran, V. and Kakarala, R. (2013). Perceptually motivated shape context which uses shape interiors. *Pattern recognition*, 46(8):2092–2102.

Psotka, J. (1978). Perceptual processes that may create stick figures and balance. *Journal of Experimental Psychology: Human Perception and Performance*, 4(1):101–111.

Ramakrishnan, R. and Gehrke, J. (2000). *Database management systems (2nd Edition)*. McGraw Hill.

Ren, X. and Ramanan, D. (2013). Histograms of sparse codes for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3246–3253.

Richards, W. and Hoffman, D. D. (1985). Codon constraints on closed 2d shapes. *Computer Vision, Graphics, and Image Processing*, 31(3):265–281.

Riemenschneider, H., Donoser, M., and Bischof, H. (2010). Using partial edge contour matches for efficient object category localization. In *Computer Vision–ECCV 2010*, pages 29–42. Springer.

Rodríguez-Sánchez, A. J. and Tsotsos, J. K. (2012). The roles of endstopped and curvature tuned computations in a hierarchical representation of 2D shape. *PLOS ONE*, 7(8).

Rosten, E. and Drummond, T. (2006). Machine learning for high-speed corner detection. In *Computer Vision–ECCV 2006*, pages 430–443. Springer.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). Orb: an efficient alternative to sift or surf. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2564–2571. IEEE.

Rubner, Y., Tomasi, C., and Guibas, L. J. (2000). The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121.

Salton, G. and McGill, M. J. (1983). Introduction to modern information retrieval.

Schiele, B. and Crowley, J. L. (1996). Object recognition using multidimensional receptive field histograms. In *European Conference on Computer Vision (ECCV)*, pages 610–619. Springer.

Sebastian, T. B., Klein, P. N., and Kimia, B. B. (2003). On aligning curves. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(1):116–125.

Sebastian, T. B., Klein, P. N., and Kimia, B. B. (2004). Recognition of shapes by editing their shock graphs. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(5):550–571.

Serra, J. (1983). *Image Analysis and Mathematical Morphology*. Academic Press, Inc., Orlando, FL, USA.

Shaked, D. and Bruckstein, A. M. (1998). Pruning medial axes. *Computer vision and image understanding*, 69(2):156–169.

Shepherd, G. M. (1974). The synaptic organization of the brain.

Shu, X. and Wu, X.-J. (2011). A novel contour descriptor for 2d shape matching and its application to image retrieval. *Image and Vision Computing*, 29(4):286 – 294.

ShuiHua, H. and ShuangYuan, Y. (2005). An invariant feature representation for shape retrieval. In *Parallel and Distributed Computing, Applications and Technologies, 2005. PDCAT 2005. Sixth International Conference on*, pages 1052–1054. IEEE.

Siddiqi, K. and Kimia, B. B. (1996). A shock grammar for recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 507–513. IEEE.

Siddiqi, K. and Pizer, S. (2008). *Medial representations: mathematics, algorithms and applications*, volume 37. Springer Science & Business Media.

Silberschatz, A., Korth, H. F., Sudarshan, S., et al. (1997). *Database system concepts*, volume 4. McGraw-Hill Singapore.

Simmons, S. and Winer, M. S. (1977). *Drawing: The creative process*. Prentice Hall.

Söderkvist, O. J. O. (2001). Computer vision classification of leaves from swedish trees. Master's thesis, Linköping University, SE-581 83 Linköping, Sweden. LiTH-ISY-EX-3132.

Soffer, A. and Samet, H. (1997). Negative shape features for image databases consisting of geographic symbols. In *Proc. 3rd International Workshop on Visual Form*.

Srestasathiern, P. and Yilmaz, A. (2011). Planar shape representation and matching under projective transformation. *Computer Vision and Image Understanding*, 115(11):1525 – 1535.

Stegmann, M. B. and Gomez, D. D. (2002). A brief introduction to statistical shape analysis. *Informatics and mathematical modelling, Technical University of Denmark, DTU*, 15:11.

Takacs, G., Chandrasekhar, V., Tsai, S. S., Chen, D., Grzeszczuk, R., and Girod, B. (2013). Fast computation of rotation-invariant image features by an approximate radial gradient transform. *IEEE Transactions on Image Processing*, 22(8):2970–2982.

Toivanen, P. J. (1996). New geodosic distance transforms for gray-scale images. *Pattern Recognition Letters*, 17(5):437–450.

Tresset, P. and Fol Leymarie, F. (2013). Portrait drawing by paul the robot. *Computers & Graphics*, 37(5):348–363.

Tresset, P. and Leymarie, F. F. (2012). Human robot interaction and drawing. *Bulletin de l'AFIA (Association Francaise pour l'Intelligence Artificielle)*, (78):44–9.

Tsai, A., Wells, W. M., Warfield, S. K., and Willsky, A. S. (2005). An em algorithm for shape classification based on level sets. *Medical Image Analysis*, 9(5):491–502.

Tsai, D.-M. and Chen, M.-f. (1995). Object recognition by a linear weight classifier. *Pattern recognition letters*, 16(6):591–600.

Tu, Z. and Yuille, A. (2004). Shape matching and recognition - using generative models and informative features. In *Computer Vision - ECCV 2004*, volume 3023 of *Lecture Notes in Computer Science*, pages 195–209. Springer Berlin Heidelberg.

van Tonder, G. J. and Ejima, Y. (2003). Flexible computation of shape symmetries within the maximal disk paradigm. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 33(3):535–540.

Van Tonder, G. J., Lyons, M. J., and Ejima, Y. (2003). Visual structure in japanese gardens. *Journal of the IEICE*, 86(10):742–746.

Van Wamelen, P., Li, Z., and Iyengar, S. (1999). A fast algorithm for the point pattern matching problem. *IEEE Trans. PAMI. last revised*, 37:preprint.

Van Wamelen, P., Li, Z., and Iyengar, S. (2004). A fast expected time algorithm for the 2-d point pattern matching problem. *Pattern Recognition*, 37(8):1699–1711.

Veltkamp, R. C. (2001). Shape matching: similarity measures and algorithms. In *Shape Modeling and Applications, SMI 2001 International Conference on.*, pages 188–197. IEEE.

Vernon, D. (1991). *Machine vision - Automated visual inspection and robot vision*. Englewood Cliffs, NJ (US); Prentice Hall, United States.

Vincent, L. (1993). Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. *Image Processing, IEEE Transactions on*, 2(2):176–201.

Wang, J., Bai, X., You, X., Liu, W., and Latecki, L. J. (2012). Shape matching and classification using height functions. *Pattern Recognition Letters*, 33(2):134–143.

Wang, X., Feng, B., Bai, X., Liu, W., and Latecki, L. J. (2014). Bag of contour fragments for robust shape classification. *Pattern Recognition*, 47(6):2116–2125.

Wang, Y. and Teoh, E. K. (2004). A novel 2d shape matching algorithm based on b-spline modeling. In *Image Processing, 2004. ICIP'04. 2004 International Conference on*, volume 1, pages 409–412. IEEE.

Wang, Y.-P., Lee, S. L., and Toraichi, K. (1999). Multiscale curvature-based shape representation using b-spline wavelets. *Image Processing, IEEE Transactions on*, 8(11):1586–1592.

Wei, H. and Li, H. (2014). Shape description and recognition method inspired by the primary visual cortex. *Cognitive Computation*, 6(2):164–174.

Weinland, D., Ronfard, R., and Boyer, E. (2011). A survey of vision-based methods for action representation, segmentation and recognition. *Computer Vision and Image Understanding*, 115(2):224–241.

Wiederhold, G. (1995). Digital libraries, value, and productivity. *Communications of the ACM*, 38(4):85–96.

Wiederhold, G. (2016). Large-scale information systems. *Database Applications Semantics*, page 34.

Wu, J. and Rehg, J. M. (2011). Centrist: A visual descriptor for scene categorization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(8):1489–1501.

Xie, J., Heng, P.-A., and Shah, M. (2008). Shape matching and modeling using skeletal context. *Pattern Recognition*, 41(5):1756–1767.

Xu, M. (2004). The multiscale medial properties of interfering image structures. *Pattern recognition letters*, 25(1):21–34.

Xu, M. and Pycock, D. (1998). Estimating true symmetry in scale-space. In *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, volume 5, pages 4620–4625. IEEE.

Xu, M. and Pycock, D. (1999). A scale-space medialness transform based on boundary concordance voting. *Journal of Mathematical Imaging and Vision*, 11(3):277–299.

Yang, M., Kpalma, K., and Ronsin, J. (2008). A survey of shape feature extraction techniques. In *Pattern Recognition Techniques, Technology and Applications*, pages 43–90. InTech.

Yang, X., Koknar-Tezel, S., and Latecki, L. J. (2009). Locally constrained diffusion process on locally densified distance spaces with applications to shape retrieval. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 357–364. IEEE.

Yasseen, Z., Verroust-Blondet, A., and Nasri, A. (2015). Sketch-based 3d object retrieval using two views and visual part alignment. In *Proceedings of the 2015 Eurographics Workshop on 3D Object Retrieval*, pages 39–46. Eurographics Association.

Young, I. T., Walker, J. E., and Bowie, J. E. (1974). An analysis technique for biological shape. *Information and control*, 25(4):357–370.

Zhang, D. and Lu, G. (2002). Enhanced generic fourier descriptors for object-based image retrieval. In *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, volume 4, pages IV–3668. IEEE.

Zhang, D. and Lu, G. (2003). A comparative study of curvature scale space and fourier descriptors for shape-based image retrieval. *Journal of Visual Communication and Image Representation*, 14(1):39–57.

Zhang, D. and Lu, G. (2004). Review of shape representation and description techniques. *Pattern recognition*, 37(1):1–19.

Zhu, S. C. and Yuille, A. L. (1996). Forms: a flexible object recognition and modelling system. *International Journal of Computer Vision*, 20(3):187–212.