



SHAPE DESCRIPTORS BASED ON INTERSECTION CONSISTENCY AND GLOBAL  
BINARY PATTERNS

A THESIS SUBMITTED TO  
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES  
OF  
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ERDAL SIVRI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR  
THE DEGREE OF MASTER OF SCIENCE  
IN  
COMPUTER ENGINEERING

SEPTEMBER 2012

Approval of the thesis:

**SHAPE DESCRIPTORS BASED ON INTERSECTION CONSISTENCY AND GLOBAL  
BINARY PATTERNS**

submitted by **ERDAL SIVRI** in partial fulfillment of the requirements for the degree of  
**Master of Science in Computer Engineering Department, Middle East Technical Uni-  
versity** by,

Prof. Dr. Canan Özgen  
Dean, Graduate School of **Natural and Applied Sciences**

---

Prof. Dr. Adnan Yazıcı  
Head of Department, **Computer Engineering**

---

Assist. Prof. Dr. Sinan Kalkan  
Supervisor, **Computer Engineering Department, METU**

---

**Examining Committee Members:**

Prof. Dr. Fatoş Tünay Yarman-Vural  
Computer Engineering Department, METU

---

Assist. Prof. Dr. Sinan Kalkan  
Computer Engineering Department, METU

---

Prof. Dr. Sibel Tarı  
Computer Engineering Department, METU

---

Assist. Prof. Dr. Ahmet Oğuz Akyüz  
Computer Engineering Department, METU

---

Assist. Prof. Dr. Alptekin Temizel  
Informatics Institute, METU

---

**Date:**

---

**I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.**

Name, Last Name: ERDAL SIVRI

Signature :

# ABSTRACT

## SHAPE DESCRIPTORS BASED ON INTERSECTION CONSISTENCY AND GLOBAL BINARY PATTERNS

Sivri, Erdal

M.S., Department of Computer Engineering

Supervisor : Assist. Prof. Dr. Sinan Kalkan

September 2012, 71 pages

Shape description is an important problem in computer vision because most vision tasks that require comparing or matching visual entities rely on shape descriptors. In this thesis, two novel shape descriptors are proposed, namely Intersection Consistency Histogram (ICH) and Global Binary Patterns (GBP). The former is based on a local regularity measure called Intersection Consistency (IC), which determines whether edge pixels in an image patch point towards the center or not. The second method, called Global Binary Patterns, represents the shape in binary along horizontal, vertical, diagonal or principal directions. These two methods are extensively analyzed on several databases, and retrieval and running time performances are presented. Moreover, these methods are compared with methods such as Shape Context, Histograms of Oriented Gradients, Local Binary Patterns and Fourier Descriptors. We report that our descriptors perform comparable to these methods.

Keywords: shape descriptors, shape matching, intersection consistency, local binary patterns

# ÖZ

## KESİŞİMLERİN TUTARLIĞI VE GLOBAL İKİLİ ÖRÜNTÜ TABANLI ŞEKİL TANIMLAYICILAR

Sivri, Erdal

Yüksek Lisans, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi : Yrd. Doç. Dr. Sinan Kalkan

Eylül 2012, 71 sayfa

Görsel öğeleri karşılaştırma ve eşlemeye gereksinim duyan çoğu görme işi şekil tanımlayıcılara ihtiyaç duyduğu için, şekil simgeleme bilgisayarlı görme alanında önemli bir problemdir. Bu tezde, Kesişimlerin Tutarlılığı Histogramı ve Global İkili Örüntüler olarak adlandırılan iki yeni şekil tanımlayıcı önerilmektedir. İlki, Kesişimlerin Tutarlılığı isimli, kenar piksellerinin görüntü parçasının merkezine doğru yönelip yönelmediğini belirleyen bir yerel düzenlilik ölçütüne dayanmaktadır. Global İkili Örüntüler adı verilen yöntem ise, şekilleri düşey, yatay, çapraz ve ana eksenler üzerinde ikili formda simgelemektedir. Bu iki yöntemin, birkaç veri tabanında kapsamlı bir şekilde analizi gerçekleştirilmiş, ve şekil getirme ve işleyiş süresi performansları sunulmuştur. Bu yöntemler ayrıca, Şekil Bağlamı, Yönelmiş Gradyent Histogramları, Yerel İkili Örüntüler ve Fourier Tanımlayıcıları isimli yöntemlerle karşılaştırılmıştır. Önerdiğimiz şekil tanımlayıcılar bu yöntemlerle karşılaştırılabilir sonuçlar üretmiştir.

Anahtar Kelimeler: şekil tanımlayıcılar, şekil eşleme, kesişimlerin tutarlılığı, yerel ikili örüntüler

*To nothingness*

## ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my supervisor, Assist. Prof. Dr. Sinan Kalkan for his support and assistance through my thesis study. Without his guidance, this study would not be possible. I also would like to thank the thesis committee members, Prof. Dr. Fatoş Tünay Yarman-Vural, Prof. Dr. Sibel Tarı, Assist. Prof. Dr. Ahmet Oğuz Akyüz and Assist. Prof. Dr. Alptekin Temizel. Especially, I would like to extend my gratitudes to Prof. Dr. Sibel Tarı for the fruitful discussions and comments on the theory of shape representation.

I am forever grateful to my parents, Erol Sivri and Selvinaz Sivri, and my sister Cansu Meltem Sivri. I was able to get over hard times through their ever lasting support.

My special thanks go to my friends who were always there for me during this study. This is a long list and I do not want to left out anyone so I will not list their names here.

The HPC resources used in this thesis were provided by the Department of Computer Engineering, Middle East Technical University.

I gratefully acknowledge financial support of TUBITAK via TUBITAK-BIDEB scholarship.

## LIST OF TABLES

### TABLES

Table 4.1 TopRank scores of different methods on the Brown University Kimia-99 Shape Database (taken from [6]). . . . .	34
Table 4.2 TopRank Scores of different methods on the Brown University Kimia-216 Shape Database (taken from [5]). . . . .	35
Table 4.3 Bull’s Eye test scores of different methods on the MPEG-7 CE Shape-1 Part-B Database (adapted from [6]). . . . .	36
Table 4.4 Accuracies of different methods on the Coil-100 Database. . . . .	37
Table 4.5 Tabular Description of Classification Metrics . . . . .	39
Table 4.6 SC Parameter Settings. . . . .	40
Table 4.7 HOG Parameter Settings. <b>Grid Size</b> specifies the number of rows and columns in a grid. . . . .	40
Table 4.8 LBP Parameter Settings. <b>Grid Size</b> specifies the number of rows and columns in a grid. . . . .	41
Table 4.9 FD Parameter Settings. . . . .	41
Table 4.10 ICH Parameter Settings. <b>Grid Size</b> specifies the number of rows and columns in a grid. . . . .	42
Table 4.11 GBP Parameter Settings. <b>Grid Size</b> specifies the number of rows and columns in a grid. . . . .	42
Table 4.12 Scores of GBP parameter settings (Table 1 of 2). . . . .	45
Table 4.13 Scores of GBP parameter settings (Table 2 of 2). . . . .	46
Table 4.14 Scores for all methods with best parameter settings. . . . .	48
Table A.1 Sample queries on the Kimia-99 database. For each method, the best per- forming parameter setting is used. . . . .	62

Table A.2 Sample queries on the Kimia-216 database. For each method, the best performing parameter setting is used. . . . .	63
Table A.3 Sample queries on the MPEG-7 database. For each method, the best performing parameter setting is used. . . . .	64
Table A.4 Sample queries on the Natural Silhouettes database. For each method, the best performing parameter setting is used. . . . .	65
Table A.5 Sample queries on the Coil-100 database. For each method, the best performing parameter setting is used. . . . .	66

# LIST OF FIGURES

## FIGURES

Figure 1.1 Objects can be recognized easily based on their shape (shapes taken from the MPEG-7 Shape Database [38]). . . . .	2
Figure 1.2 Intersection Consistency Histogram computation inside a $5 \times 5$ image patch. (a) Computation of IC values in a window. (b) Computed IC values for each pixel. (c) Resulting ICH descriptor. . . . .	3
Figure 1.3 Global Binary Patterns computation along horizontal direction (denoted $GBP_h$ ). (a) The original image. (b) After thresholding with brightness level 10. (c) After rows are multiplied by powers of two ( $2^0 2^1 2^2 2^3 2^4$ ). (d) Resulting $GBP_h$ descriptor. . . . .	4
Figure 2.1 Examples of contour-based and region-based similarity (taken from [11]). . . . .	6
Figure 2.2 Classification of shape representation and description techniques. . . . .	6
Figure 2.3 Illustration of centroid-distance signature (adapted from [47]). (a) The shape contour. (b) Radius-angle, $R - \theta$ , graph. . . . .	8
Figure 2.4 Illustration of Curvature Scale Space convolution (reprinted from [78]). (a) The original image. (b-e) The evolution of shape boundary as scale ( $\sigma$ ) increases (inflection points marked). . . . .	9
Figure 2.5 Illustration of elastic deformations (reprinted from [18]). (a) The original image. (b) Several steps of the deformation process. . . . .	10
Figure 2.6 Illustration of Grid descriptor (adapted from [44]). (a) Grid overlaid on the original shape. (b) Cells with at least 15% of pixels covered by the shape. (c) Bit string representation of the shape. Resulting Grid descriptor constructed by concatenating rows: 1100000011100000011110000111111101111110 . . . . .	12

Figure 2.7 Illustration of Shape Matrix polar quantization. (a) Polar raster of concentric circles are superimposed at the center of mass. (b) Bit string representation of the shape. . . . .	13
Figure 2.8 Illustration of concavity tree construction (reprinted from [4]). (a) The original shape, (b,c) its concavities, (d) meta-concavities, and (e) corresponding concavity tree. . . . .	14
Figure 2.9 Illustration of Blum’s medial axis. (a) Medial axis model using the grass-fire analogy. (b) Medial axis skeletons for various shapes. . . . .	14
Figure 2.10 Shape Context computation (taken from [7]). (a,b) Original shapes. (c,d) Sampled edge points. (e-g) Example shape contexts for reference points marked by $\circ$ , $\diamond$ and $\triangleleft$ . Note the visual similarity of the shape contexts for homologous points, $\circ$ and $\diamond$ . On the contrary, the shape context for $\triangleleft$ is quite different. (h) Correspondences found using bipartite matching, with costs defined by the $\chi^2$ statistic. . . . .	17
Figure 2.11 Histograms of Oriented Gradients computation (taken from [36]). . . . .	18
Figure 2.12 General Local Binary Patterns computation (reprinted from [45]). The LBP value for the center pixel is equal to $1 + 2 + 4 + 8 + 128 = 143$ . . . . .	20
Figure 2.13 Local Binary Patterns $LBP_{P,R}$ computation (reprinted from [45]). (a) $(P, R) = (8, 1.0)$ . (b) $(P, R) = (12, 2.5)$ . (c) $(P, R) = (16, 4.0)$ . . . . .	21
Figure 3.1 Intersection Consistency computation inside a $5 \times 5$ image patch (adapted from [91]). . . . .	24
Figure 3.2 Sample images (resized to $32 \times 32$ ) from the MPEG-7 [38] database and their corresponding ICH descriptors (brighter intensities indicate higher values). Descriptors are extracted using Algorithm 5 with parameters $T$ and $V$ set to 4. The length of each descriptor vector is, thus, equal to $T \cdot V \cdot L = 4 \times 4 \times 10 = 160$ . Figures (a) and (b) contain shapes from mcategories <i>Apple</i> and <i>Bone</i> respectively. On the other hand, Figures (c) and (d) contain shapes from different categories. ICH descriptors in Figures (a) and (b) form a more uniform distribution compared to descriptors in Figures (c) and (d). . . . .	27

Figure 3.3	Global Binary Patterns computation along horizontal and vertical directions, denoted $GBP_h$ and $GBP_v$ respectively. (a) The original image. (b) After thresholding with brightness level 10. (c) After rows are multiplied by powers of two: $(2^0 2^1 2^2 2^3 2^4)$ . (d) After each row is summed horizontally. (e) After columns are multiplied by powers of two: $(2^0 2^1 2^2 2^3 2^4)$ . (f) After columns are summed vertically. (g) Resulting GBP descriptor. . . . .	28
Figure 3.4	Illustration of the projection of a pixel onto a line that passes through the pixel $p_o$ with orientation $\theta$ . . . . .	29
Figure 4.1	Images from the Brown University Kimia-99 Shape Database. . . . .	33
Figure 4.2	Arbitrary images from the Brown University Kimia-216 Shape Database. . . . .	34
Figure 4.3	Arbitrary images from the MPEG-7 CE Shape-1 Part-B Database. . . . .	35
Figure 4.4	Arbitrary images from the Natural Silhouettes database. . . . .	36
Figure 4.5	Arbitrary images from the Coil-100 Database. . . . .	37
Figure 4.6	ROC curves for Intersection Consistency Histogram. For the sake of visibility, only $tpr > 0.4$ is shown without loss of any information. . . . .	44
Figure 4.7	ROC curves for all methods with best parameter settings. For the sake of visibility, only $tpr > 0.4$ is shown without loss of any information. . . . .	47
Figure 4.8	Running time measurements of all methods on the Kimia-216 database. Running times are reported in log-seconds. . . . .	50
Figure 4.9	Running time measurements of all methods on the Kimia-216 database. Running times are reported in log-seconds. For each method, the best performing parameter setting is used. . . . .	51
Figure 5.1	Figure (a) contains a set of points that form a circle perception. Figure (b) illustrates a peanut shape as its neck gets thin or thick. The peanut transforms from a single blob to two discrete blobs as its neck thins. Images are taken from [74]. . . . .	53
Figure B.1	ROC curves for Shape Context. For the sake of visibility, only $tpr > 0.4$ is shown without loss of any information. . . . .	68

Figure B.2 ROC curves for Histograms of Oriented Gradients. For the sake of visibility, only $tpr > 0.4$ is shown without loss of any information. . . . .	69
Figure B.3 ROC curves for Local Binary Patterns. For the sake of visibility, only $tpr > 0.4$ is shown without loss of any information. . . . .	70
Figure B.4 ROC curves for Fourier Descriptors. For the sake of visibility, only $tpr > 0.4$ is shown without loss of any information. . . . .	71

# CHAPTER 1

## INTRODUCTION

The amount of digital media is astonishing and ever increasing, requiring algorithms to automate certain tasks. For instance, a video surveillance system can use object recognition algorithms to detect suspicious people [30]; old documents can be converted into a digital form using optical character recognition algorithms in just seconds, eliminating the need for hours of manual work [53]; content-based image retrieval systems can be used to search images based on shape, color or texture [79]. Such and many other applications require recognition and identification of objects.

Object recognition is perhaps one of the most challenging problems of computer vision, mostly because it is an *ill-posed* problem<sup>1</sup> in which 3-D objects are to be recognized using their 2-D projections under varying viewpoints. A 2-D projection of an object is just **one** of the infinitely many projections that are actually possible. Besides this inherent ambiguity, objects in real world are also subject to visual transformations such as occlusion, deformation and illumination change, and to make matters even worse, background clutter and intra-class variation introduce further complications.

The ultimate goal of object recognition is to recognize objects in real world settings in which objects co-exist with other objects on cluttered backgrounds. Therefore, prior to object recognition, preprocessing algorithms are required to separate regions as *figure* and *background*. After visual entities of interest are extracted from background, object descriptors are used to represent them in a compact form. The last step is to match each object to a set of objects in a database to determine the category of the object.

---

<sup>1</sup> A problem is *ill-posed* if either a solution does not exist or the solution is not unique or it does not depend solely on the data [26].

## 1.1 Problem Definition

There are a number of visual features that can be used to describe objects, such as *shape*, *brightness*, *color*, *texture* and *motion* (if sequence of images is available). Among these features, *shape* is generally the most informative one. Human beings, for instance, can recognize objects rapidly based solely on their appearance or shape. The category of each shape given in Figure 1.1 is obvious to a human observer at the first look.



Figure 1.1: Objects can be recognized easily based on their shape (shapes taken from the MPEG-7 Shape Database [38]).

Motivated from the representative power of visual shape, this thesis approaches the object recognition problem using shape descriptors. We propose two novel shape descriptors, namely Intersection Consistency Histogram and Global Binary Patterns. Intersection Consistency Histogram (ICH) is based on a local regularity measure called Intersection Consistency (IC), which was previously shown to improve junction localization [32] and edge detection [91]. Intersection Consistency checks whether hypothetical lines going through edge pixels in a window are oriented towards the center pixel or not. This simple measure is utilized to construct a rich shape descriptor to represent shapes (see Figure 1.2).

The second method, called Global Binary Patterns (GBP), is a computationally simple algorithm that describes shapes using projections of binary pixel values along horizontal, vertical, diagonal and principal directions. See Figure 1.3 for an illustration of GBP along horizontal direction. Since direct summation of pixel values along an axis causes the descriptor to lose the spatial distribution of pixels, it cannot successfully capture the shape information. Therefore, sequences of binary values, i.e. bits, are interpreted as a single large binary number and converted to a decimal value. This idea is similar to the one adopted by the popular texture descriptor called Local Binary Patterns (LBP) [57].

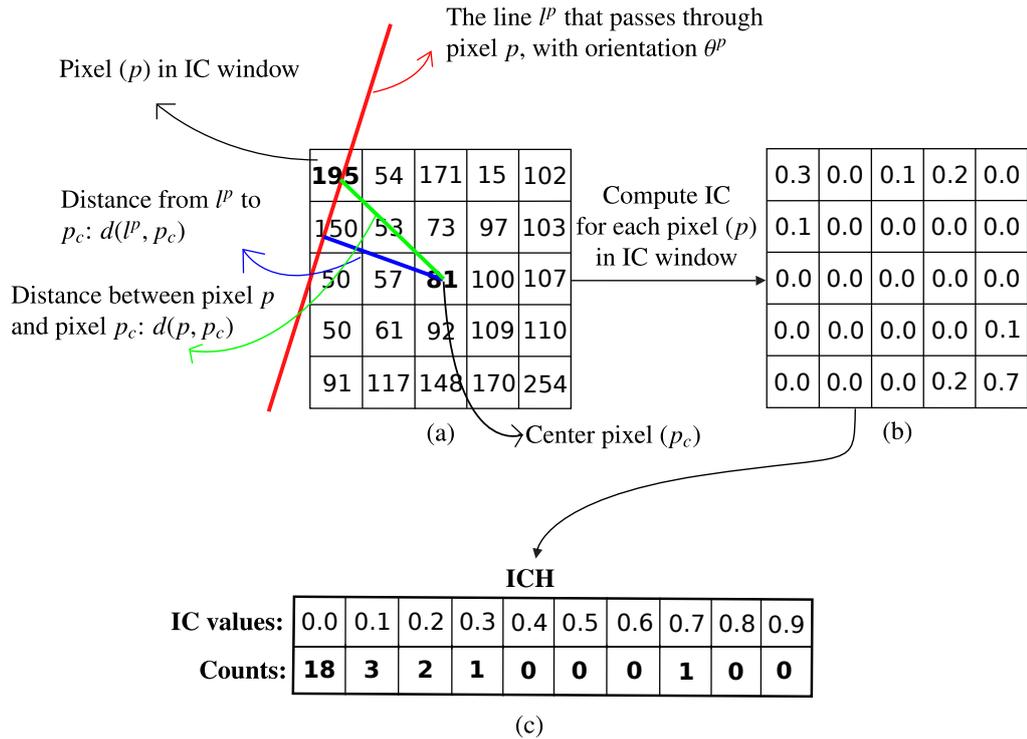


Figure 1.2: Intersection Consistency Histogram computation inside a  $5 \times 5$  image patch. (a) Computation of IC values in a window. (b) Computed IC values for each pixel. (c) Resulting ICH descriptor.

## 1.2 Contributions and Outline

This thesis has two major contributions:

- A novel shape descriptor, named “Intersection Consistency Histogram”, is proposed.
- Another novel shape descriptor, named “Global Binary Patterns”, that represents the shape as binary numbers, is proposed. This work is submitted to “1st ACCV Workshop on Computer Vision with Local Binary Pattern Variants” in conjunction with “ACCV 2012, The 11th Asian Conference on Computer Vision”, and work is in progress for a submission to “EURASIP Journal on Image and Video Processing, Special Issue on Local Binary Patterns (LBP)-based Image and Video Analysis”.

Performances of these descriptors are analyzed in detail and compared to widely-used methods in the literature: Shape Context [8], Histograms of Oriented Gradients [15], Local Binary Patterns [57] and Fourier Descriptors [25, 61, 65, 83, 85]. A total of six techniques are eval-

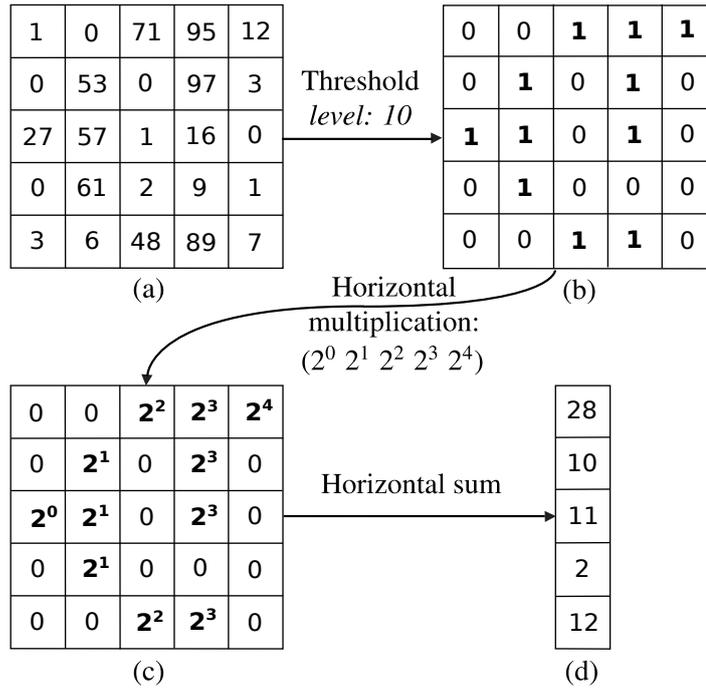


Figure 1.3: Global Binary Patterns computation along horizontal direction (denoted  $GBP_h$ ). (a) The original image. (b) After thresholding with brightness level 10. (c) After rows are multiplied by powers of two ( $2^0 2^1 2^2 2^3 2^4$ ). (d) Resulting  $GBP_h$  descriptor.

uated on five different databases, and retrieval and running time performances are reported using various metrics and plots.

The rest of the thesis is structured as follows. In Chapter 2, background information on 2-D shape representation is provided along with detailed descriptions of selected descriptors. Chapter 3 describes the proposed methods, namely Intersection Consistency Histogram and Global Binary Patterns in detail. Performance measurements and experimental results are presented in Chapter 4. Finally, Chapter 5 concludes the thesis with a brief discussion and future work.

## CHAPTER 2

### BACKGROUND

There is a vast amount of literature on shape representation and description. This is due to shape being an important visual cue for object recognition, which is one of the most important goals of computer vision. That aside, applications of shape descriptors range from image compression [33], shape-based indexing [89] and medical image analysis [31, 42] to content-based image retrieval (CBIR) [79], especially shape retrieval using human drawn object sketches [18].

In this chapter, shape representation and description methods are summarized and four of the most popular descriptors that are compared against the descriptors proposed in this thesis, are described in detail.

#### 2.1 Shape Descriptors

In real world, object shapes are three dimensional, which can be described using 3-D shape descriptors. Although there is research on 3-D shape representation [73, 82], most of the research focuses on 2-D shape representation because 3-D shape descriptors require either stereo vision (which are not always available) or extraction of monocular depth cues to construct the 3-D pose of the object (which is a complicated and time consuming task). 2-D shape representation methods are generally divided into two broad categories: *contour-based* and *region-based* methods [11, 86]. The former exploits the shape contour, while the latter utilizes the area enclosed within the shape contour.

Some shapes are represented better with a region-based descriptor while some are better suited to a contour-based description. This phenomenon is illustrated in Figure 2.1. Shapes in

the first row are similar if a region-based descriptor is used. On the other hand, objects in each column is similar according to a contour-based descriptor rather than a region-based descriptor. If the shape at the intersection of boxes is used as a query, contour-based and region-based descriptors return two disjoint sets of shapes.

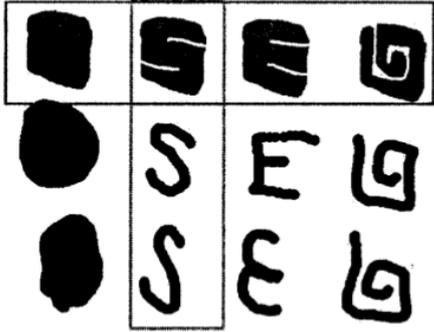


Figure 2.1: Examples of contour-based and region-based similarity (taken from [11]).

Shape representation techniques can further be classified according to the scale at which visual signature is represented: *global* and *local* approaches. Global methods consider the shape as a whole whereas local techniques represent the shape as sub-parts or primitives. Figure 2.2 illustrates shape representation and description methods in hierarchical form. Important algorithms under each category are listed. They will be described in detail later in this chapter.

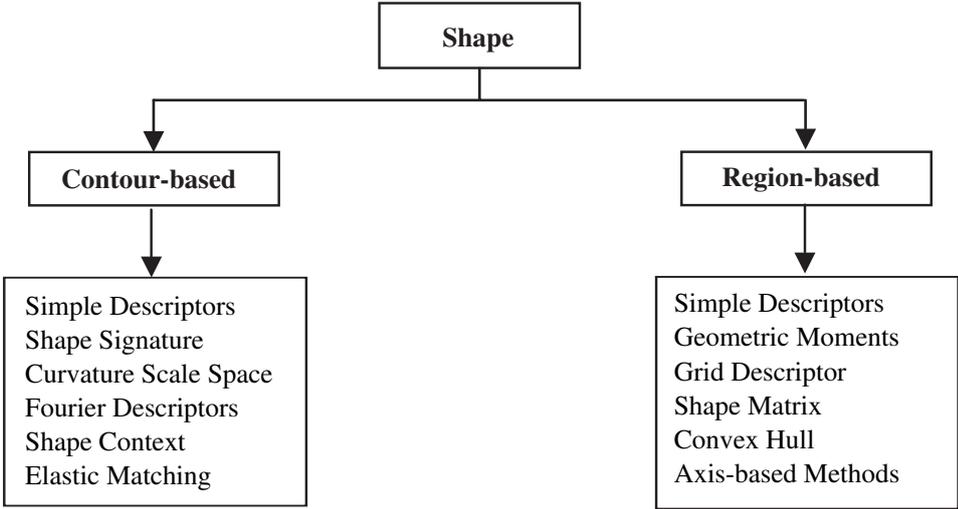


Figure 2.2: Classification of shape representation and description techniques.

Shape descriptors are expected to satisfy certain requirements to be useful in practice. Gen-

erally, they should be invariant to *rotation*, *scale* and *translation* changes. The first one, namely *rotation invariance*, is obtained if the representation of the shape does not significantly change as a result of arbitrary rotations. Although rotational invariance is a desired property for a shape descriptor, complete rotational invariance is not always desired to be able to distinguish, for instance, the shape of digit 9 from that of digit 6. The second property, *scale invariance*, is achieved if the representation of the shape is similar to the representation of a scaled version of the original shape. Finally, *translation invariance* is achieved if the representation of the shape does not deviate in the presence of translations in any direction.

### **2.1.1 Contour-Based Descriptors**

In contour-based descriptors, pixels on the boundary of the shape are used to describe shapes. Most of the contour-based methods extract a contour using a boundary detection or an edge detection algorithm. Moreover, some of them sample the contour to make it more compact. In following sections, some of the important contour-based shape representation methods are reviewed.

#### **2.1.1.1 Simple Contour-Based Descriptors**

Descriptors based on extracting simple geometric properties from shape contours can be investigated under the field of computational geometry. These simple descriptors, such as perimeter, eccentricity, elongation, rectangularity and principal orientation [12, 86], are able to classify shapes that are significantly dissimilar in appearance. To start with, the perimeter of a shape is equal to the number of pixels residing on the contour. Eccentricity is the ratio of the lengths of major and minor axis. Elongation measures how much elongated a shape is by computing the ratio of the height and width of the bounding box of the shape. Rectangularity measure for a shape is proportional to how much space it fills inside the bounding box.

#### **2.1.1.2 Shape Signatures**

Shape signatures describe the shape as a one-dimensional vector constructed from shape boundary. They capture global properties that are translation, scale and rotation invariant.

Central distance, tangent angle, cumulative angle, curvature, complex coordinates, polar coordinates and centroid distance (see Figure 2.3) are among the few of the well-known shape signatures [17, 84]. Most of the signatures are sensitive to noise and small deviations so they require further processing such as normalization or quantization [86] to be effective. In addition, to provide rotation tolerance, signature matching is normalized to shift matching in 1-D and 2-D space. This phase adds additional computational complexity, making shape signatures inapplicable to rapid, online image retrieval.

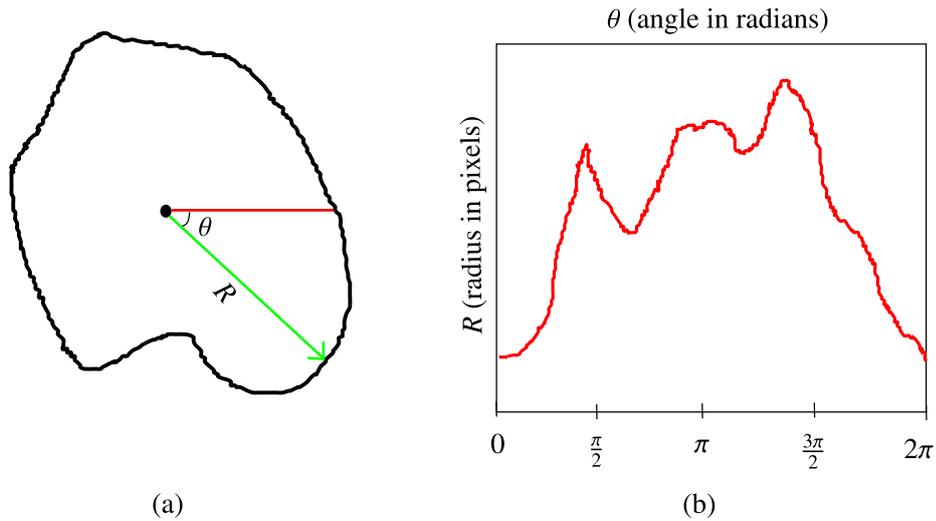


Figure 2.3: Illustration of centroid-distance signature (adapted from [47]). (a) The shape contour. (b) Radius-angle,  $R - \theta$ , graph.

### 2.1.1.3 Curvature Scale Space

Curvature Scale Space utilizes a scale space representation of shape contour to describe the shape. This method, proposed by Mokhtarian and Mackworth and adopted as a part of the MPEG-7 contour-based shape descriptors [11], satisfies criteria expected from a general purpose shape descriptor: *invariance*, *uniqueness*, *stability*, *efficiency* and *ease of implementation* [51, 50, 52]. Formally, given a contour,  $C$ , parameterized by arc-length  $s$ :  $C(s) = (x(s), y(s))$ , the method starts with convolving the coordinates with a Gaussian kernel  $\phi_\sigma$  of width  $\sigma$ :

$$x_\sigma(s) = \int x(s)\phi_\sigma(t - s)dt, \quad (2.1)$$

$$y_\sigma(s) = \int y(s)\phi_\sigma(t - s)dt, \quad (2.2)$$

where  $\phi_\sigma$  is the Gaussian kernel of width  $\sigma$ :

$$\phi_\sigma(t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-t^2/2\sigma^2}. \quad (2.3)$$

As the value of  $\sigma$  gets larger, the contour gets smoother until curvature becomes positive (see Figure 2.4).

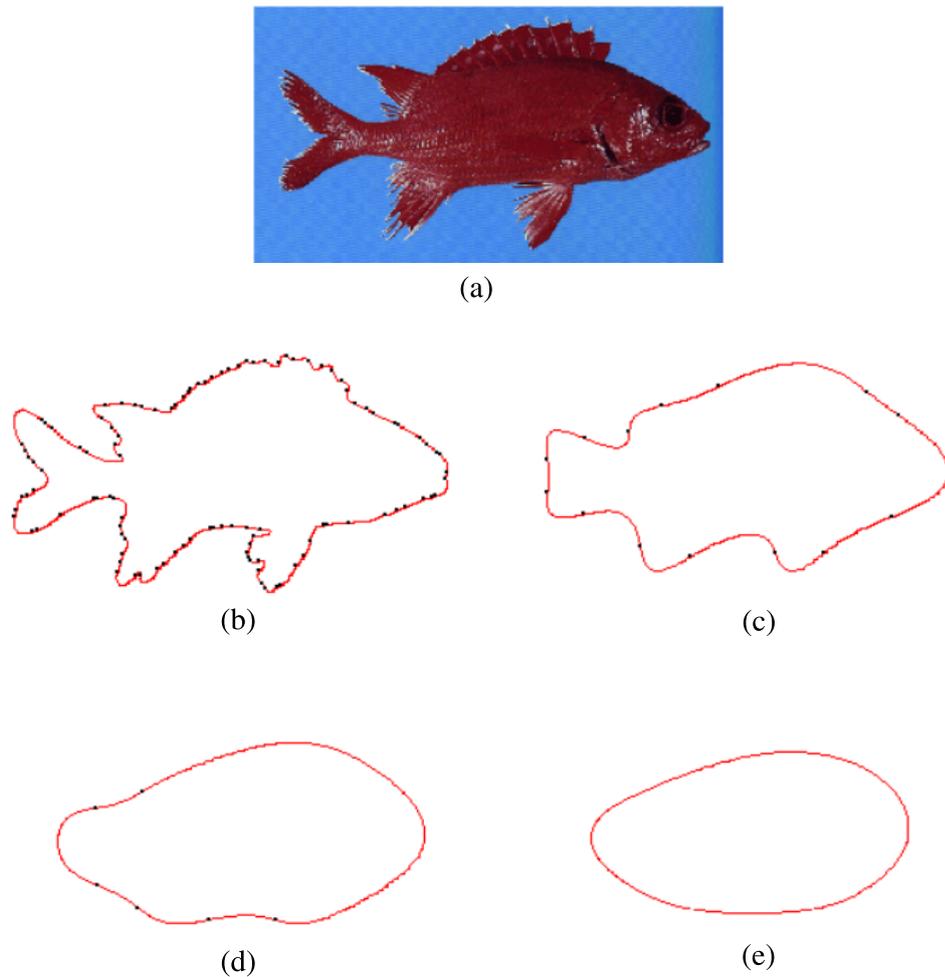


Figure 2.4: Illustration of Curvature Scale Space convolution (reprinted from [78]). (a) The original image. (b-e) The evolution of shape boundary as scale ( $\sigma$ ) increases (inflection points marked).

#### 2.1.1.4 Elastic Matching

Elastic approaches are applied to the problem of visual image retrieval and object matching by Bimbo and Pala [18] and Anil et al. [29] respectively. The model proposed by Bimbo and Pala

is based on matching a sketch to an object in the presence of deformations (see Figure 2.5). This is accomplished by letting the sketch warp under two conflicting requirements. First, the edges of the image should be followed as closely as possible, and second, deformation of the template should be taken into account. A deformed template is generated as the sum of the original template  $\rho(s)$ , which is a second order spline denoted as  $\rho = (\rho_x, \rho_y)$ , and a warping deformation  $\theta(s)$ :

$$\varphi(s) = \rho(s) + \theta(s). \quad (2.4)$$

The similarity between the sketch and the object is computed by minimizing the following function:  $F = S + B + M$  where  $S$  is called strain energy:

$$S = \alpha \int_0^1 \left[ \left( \frac{d\theta_x}{ds} \right)^2 + \left( \frac{d\theta_y}{ds} \right)^2 \right] ds, \quad (2.5)$$

$B$  is called the bend energy:

$$S = \beta \int_0^1 \left[ \left( \frac{d^2\theta_x}{ds^2} \right)^2 + \left( \frac{d^2\theta_y}{ds^2} \right)^2 \right] ds, \quad (2.6)$$

and  $M$  is the amount of overlapping between the deformed sketch and the object:

$$M = \int_0^1 I(\varphi(s)) ds. \quad (2.7)$$

The drawback of this method is its computational complexity. Both feature extraction and matching phases require heavy computations [86].

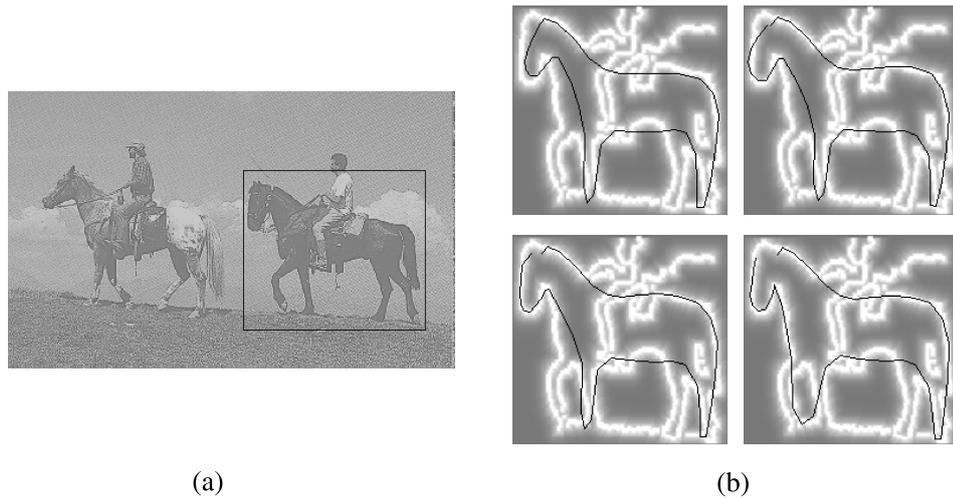


Figure 2.5: Illustration of elastic deformations (reprinted from [18]). (a) The original image. (b) Several steps of the deformation process.

## 2.1.2 Region-Based Descriptors

Region-based descriptors use all pixels inside the area enclosed by the shape boundary to build descriptors. This is in contrast to contour-based representations, which use pixels only from the shape boundary.

### 2.1.2.1 Simple Region-Based Descriptors

Similar to shape contour properties, region properties are simple descriptors extracted from the region of a shape. These include, but not limited to, area, compactness and circularity. To start with, area is equal to number of pixels that exist inside the shape, and compactness or circularity is computed with the following formula:  $C = P^2/A$  where  $P$  is the perimeter and  $A$  is the area of the shape. These measures are easy-to-compute but are not suitable as a generic shape descriptor since they cannot capture the appearance of complex shapes. They may be used to eliminate false positives or combined with other shape descriptors to be effective.

### 2.1.2.2 Geometric Moments

Image moment invariants are first introduced by Hu [28] for pattern recognition applications. They are based on the idea that an image can be represented as a set of moments. A moment of an image is defined to be a particular weighted average of pixel intensities. Given a  $N \times M$  binary image, the  $(p, q)$  moment of a shape  $O \in \mathfrak{X}$ , denoted by  $m_{p,q}$ , is defined as follows.

$$m_{p,q} = \sum_{x=1}^N \sum_{y=1}^M x^p y^q f(x, y). \quad p, q = 1, 2, 3, \dots \quad (2.8)$$

If  $p$  and  $q$  go to infinity, the representation of the shape becomes exact. In practice, algorithms use a nonlinear combinations of a number of low order moments, ignoring the less critical high order moments. The problem with this approach is that the shape cannot be accurately described using only a few invariants constructed from lower order moments. In fact, geometric moment invariants are evaluated on MPEG-7 shape database [87] and are shown to perform well only on affinely transformed contour-based shapes. They tend to perform poorly on arbitrarily deformed ones.

Teague introduced the concept of orthogonal moments to describe an image using moments

based on the theory of orthogonal polynomials [76]. This method is called Zernike moments and they allow independent moment invariants to be recovered to an arbitrarily high order. Among other orthogonal moments are Legendre moments [19], rotational moments [13] and complex moments [77].

Angular Radial Transformation (ART) [34] is another moment-based shape description method adopted by MPEG-7 [11]. ART is an efficient method that describes multiple disjoint regions in a compact way. In addition, it is robust against salt and pepper noise and able to retrieve objects that are split into disconnected sub-regions.

**2.1.2.3 Grid Descriptor**

The grid-based shape descriptor is a simple and intuitive method developed by Lu and Sajjan [44]. The idea is to scan the image from left to right and top to bottom, and create a *bitmap* which contains 1 where a cell is covered by the shape or 0 if the cell is outside of the shape (see Figure 2.6). Shapes are represented as bitmaps and comparison is performed using the *Hamming Distance* [27]. Grid-based representations are easy to compute and compact but they suffer from noise and are not generally tolerant to scale and rotation changes. To make the descriptor scale and rotation invariant, preprocessing operations called “scale normalization” and “rotation normalization” are applied. Scale normalization is achieved by proportionally scaling all shapes so that their major axes have the same length, and rotation normalization is achieved by rotating the shape so that its major axis is parallel with the x-axis.

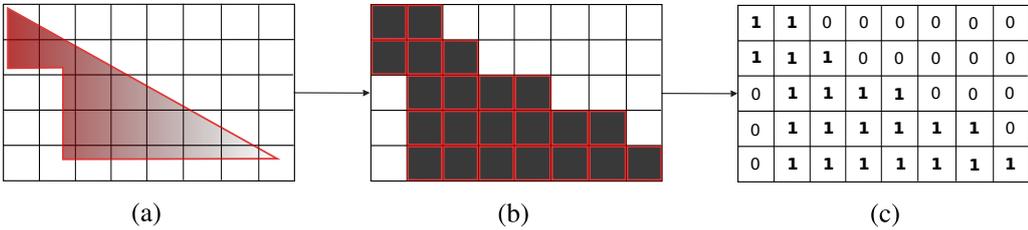


Figure 2.6: Illustration of Grid descriptor (adapted from [44]). (a) Grid overlaid on the original shape. (b) Cells with at least 15% of pixels covered by the shape. (c) Bit string representation of the shape. Resulting Grid descriptor constructed by concatenating rows: 110000001110000001111000011111101111110

### 2.1.2.4 Shape Matrix

Shape Matrix is a translation, rotation and scale invariant shape descriptor proposed by Gosh-tasby [24]. The method is based on transforming the shape into a matrix by polar quantization. This is similar to raster sampling (grid-based descriptors) but instead of the regular square grid on a shape image, a polar raster of concentric circles located at the center of the mass is used (see Figure 2.7). Drawbacks of this method are its sensitivity to noise and computational complexity [86]. To overcome these disadvantages, a shape descriptor based on the relative areas of the shape inside concentric circles is proposed [43, 59].

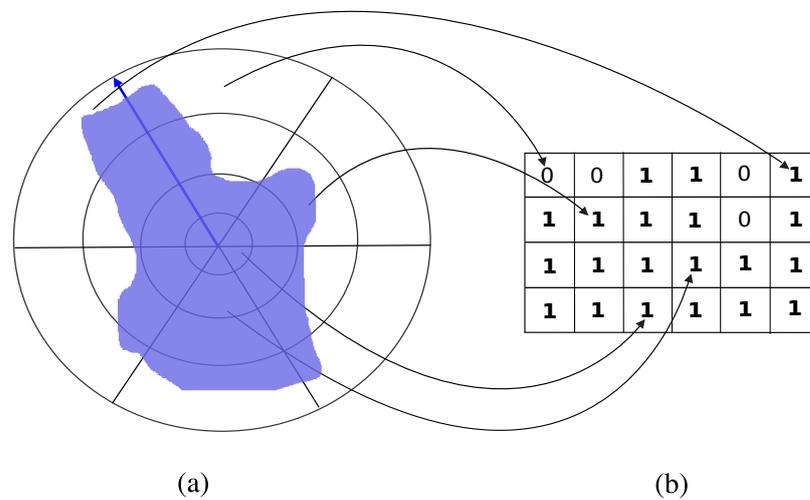


Figure 2.7: Illustration of Shape Matrix polar quantization. (a) Polar raster of concentric circles are superimposed at the center of mass. (b) Bit string representation of the shape.

### 2.1.2.5 Convex Hull

The *convex hull* of a region is defined to be the smallest convex region which covers the region. In other words,  $H$  is a convex hull of region  $R$  if  $R \subset H$ . The difference between  $H$  and  $R$ ,  $H - R$ , is called the *convex deficiency*. The convex hull can be extracted using a boundary tracing method [71] or morphological methods [17, 22]. After concavities are extracted the shape is represented as a string of concavities or concavity tree (see Figure 2.8). Because of the irregularities of shape boundaries, convex deficiency has small meaningless components distributed throughout the boundary. The most common solution to this problem is to apply a smoothing operation to boundary before partitioning.

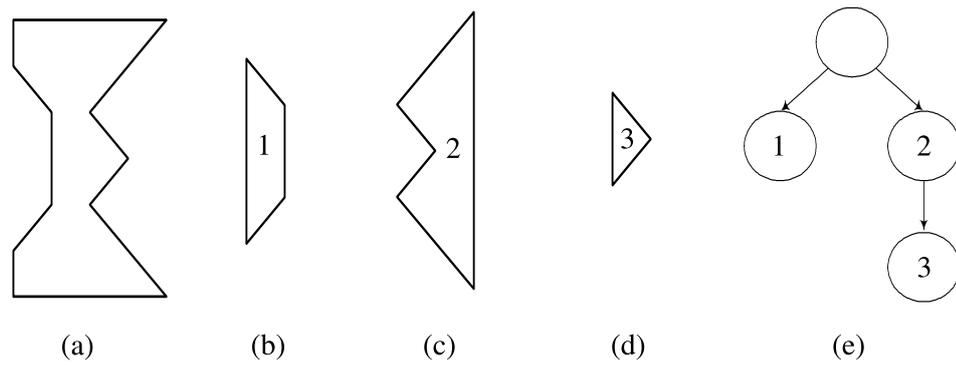


Figure 2.8: Illustration of concavity tree construction (reprinted from [4]). (a) The original shape, (b,c) its concavities, (d) meta-concavities, and (e) corresponding concavity tree.

### 2.1.2.6 Axis-Based Methods

Axis-based models, first introduced by Blum [10], capture the part information of shapes and are generally insensitive to articulations and occlusions. The representation proposed by Blum, called the *medial axis*, shifts the attention from the boundary to interior region of a shape. His model is illustrated using the grass-fire analogy in which the interior of the shape is assumed to be filled with grass and fire is started on every point on the boundary, forming fire waves (see Figure 2.9). Fire wave fronts meet along lines inside the shape. These lines constitute the *medial axis* of the shape.

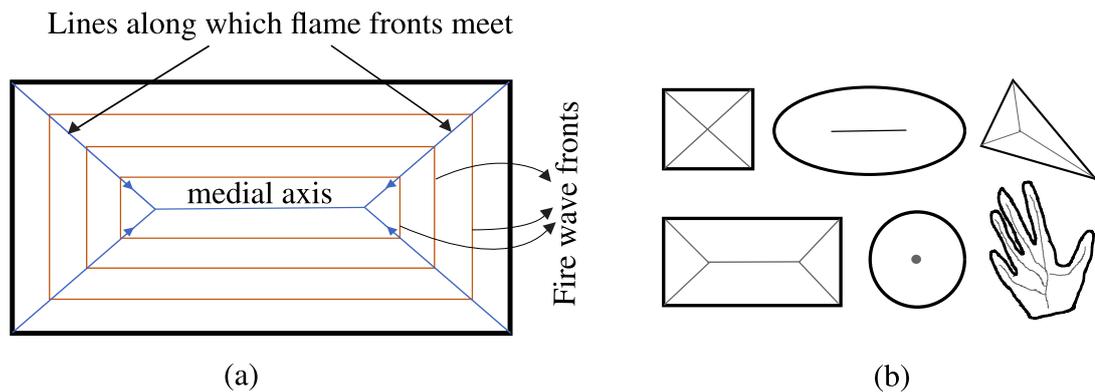


Figure 2.9: Illustration of Blum's medial axis. (a) Medial axis model using the grass-fire analogy. (b) Medial axis skeletons for various shapes.

There has been a great deal of research on axis-based representations after Blum. Some of the prominent method in this area are: Smoothed Local Symmetries [2], PISA [39], Symmetry

Set [14], Voronoi Skeletons [55], Shape Axis Tree [21], Shock Graphs [35, 66, 69, 70], TSP [75] and Disconnected Skeleton [3].

## 2.2 Shape Descriptors Analyzed in this Thesis

This section presents four of the shape descriptors that are compared with the descriptors proposed in this study. Shape Context and Fourier Descriptors are widely-used as shape representation techniques whereas Histograms of Oriented Gradients is mostly used for human detection [15] and Local Binary Patterns is generally known as a texture descriptor [57]. These methods are chosen because they are good representatives of shape descriptors that are available in the literature. Although the performances of these methods on widely-used databases are not among the best reported in the literature, they are the predecessors of better performing, more recent methods. The rest of this section is devoted to the description of these methods in detail.

### 2.2.1 Shape Context (SC)

Shape Context is a contour-based shape representation technique proposed by Belongie et al. [7, 8, 9]. The descriptor samples  $N$  points on the shape contour and, for each point, describes relative spatial distributions of other points in a compact representation (see Figure 2.10). Formally, given  $N$  points  $p_1, p_2, \dots, p_N$  from a contour  $C$ , the shape context at point  $p_i$  is defined as follows:

$$h_i(\theta, r) = \sum_{j=1}^N \delta(\theta - \theta_j, r - r_j), \quad (2.9)$$

where  $\theta_j$  is the slope of the line that passes through points  $p_i$  and  $p_j$ :

$$\theta_j = \tan(\vec{p}_i - \vec{p}_j), \quad (2.10)$$

$r_i$  is distance between points  $p_i$  and  $p_j$ :

$$r_i = |\vec{p}_i - \vec{p}_j|, \quad (2.11)$$

and  $\delta(\cdot, \cdot)$  is the Kronecker delta:

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = 0 \text{ and } y = 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.12)$$

In this formulation,  $h_i(\cdot, \cdot)$ , the shape context for  $p_i$ , is basically a histogram that counts the number of times a particular angle-distance pair, i.e.,  $(\theta, r)$ , occurs between  $p_i$  and  $p_j$  for  $j = 1, 2, \dots, N$ . Since the histogram has a fixed size  $K$ , (12 angle bins and 5 log-distance bins),  $h$  can be indexed with a single index, simplifying the notation,  $h_i(\theta, r) \equiv h_i(k)$ .

The shape context for the whole contour,  $C$ , is defined as the set of shape contexts for all points  $p_i$  for  $i = 1, 2, \dots, N$  on the contour. Algorithm 1 lists the steps required to compute a shape context,  $H$  from a gray scale image  $I$ .

---

**Algorithm 1** Shape Context Computation

---

**Require:** **I:** A gray-scale image.

**N:** number of sampling points.

- Apply an edge detection algorithm on  $I$  to get  $E$ .
- Apply a (preferably uniform) sampling algorithm to extract  $N$  points from  $E$  to get a contour,  $C$ .
- Initialize a  $N \times 5 \times 12$  matrix:  $H$ .

**for all**  $p_i \in C$  **do**

**for all**  $p_j \in C$  **do**

$$\vec{V} \leftarrow \vec{p}_i - \vec{p}_j$$

$$H(i, \theta, r) \leftarrow \#\{p_i \neq p_j \mid |\vec{V}| \in \text{bin}(r) \ \& \ \log(\tan(\vec{V})) \in \text{bin}(\theta)\}$$

**end for**

**end for**

- Reshape  $H$  to get a two dimensional  $N \times 60$  matrix.
- 

Given two shape contexts,  $h_i$  and  $h_j$  corresponding to two sampled points from two different images, comparison is carried out using the  $\chi^2$  test statistic:

$$C_{ij} = \frac{1}{2} \sum_{n=1}^N \frac{[h_i(n) - h_j(n)]^2}{h_i(n) + h_j(n)}, \quad (2.13)$$

where  $C_{ij}$  denotes the cost of matching shape context  $h_i$  with shape context  $h_j$ , i.e.  $C_{ij}$  is the degree of dissimilarity between  $h_i$  and  $h_j$ . Given  $C_{ij}$  for all pairs of sampled points  $(p_i, p_j)$ , matching is performed by solving a correspondence problem. This is known as the “linear assignment problem” and is efficiently solved using the “Hungarian algorithm” [37]. Figure 2.10 illustrates the steps of the whole process using three points sampled from two similar sample shapes.

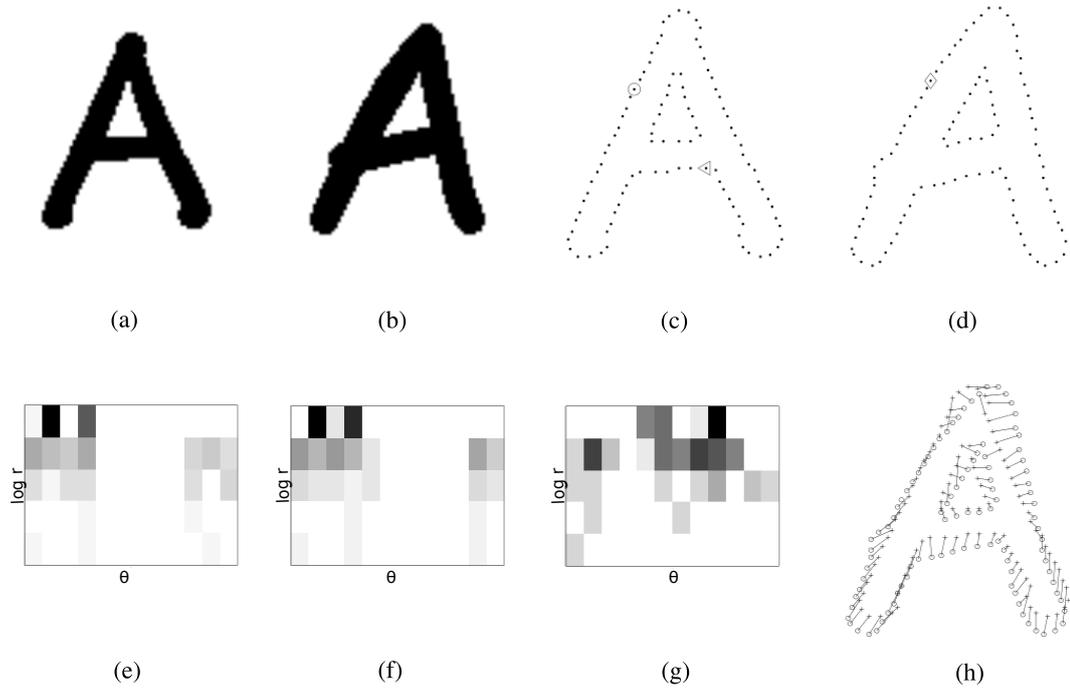


Figure 2.10: Shape Context computation (taken from [7]). (a,b) Original shapes. (c,d) Sampled edge points. (e-g) Example shape contexts for reference points marked by  $\circ$ ,  $\diamond$  and  $\triangleleft$ . Note the visual similarity of the shape contexts for homologous points,  $\circ$  and  $\diamond$ . On the contrary, the shape context for  $\triangleleft$  is quite different. (h) Correspondences found using bipartite matching, with costs defined by the  $\chi^2$  statistic.

In general, a shape descriptor should be invariant to *translation*, *rotation* and *scale* changes. Shape context is able to satisfy all three requirements. Translation invariance is naturally supported since the algorithm does not use absolute coordinates. Rotation invariance can be achieved by measuring angles at a point relative to the direction of the tangent. Finally, scale invariance is accomplished by normalizing all distances by the mean of all point pair distances on the contour.

A very well-known variation of Shape Context is the Inner Distance Shape Context (IDSC) algorithm, proposed by [40]. The difference between Shape Context and Inner Distance Shape Context is that the former uses Euclidean distance to build the log-polar histogram, while the latter uses a measure called *inner-distance*, which is defined as the length of the shortest path between two points inside the region enclosed by the shape boundary. This distance measure is shown to represent articulated shapes better [41].

## 2.2.2 Histograms of Oriented Gradients (HOG)

Histograms of Oriented Gradients (HOG) is an object detection algorithm proposed by Dalal and Triggs [15]. The method constructs a histogram of gradient occurrences in localized grid cells (see Figure 2.11). In the original paper, authors demonstrated the success of the technique on human detection problem, and in this study, the performance of HOG on shape silhouette databases is analyzed and compared with other popular descriptors. In mathematical terms, HOG in an image patch  $P$  is defined as follows:

$$HOG(k) = \sum_{p \in P} \delta\left(\left\lfloor \frac{\theta^p}{L} \right\rfloor\right), \quad (2.14)$$

where  $\delta(\cdot)$  is the Kronecker delta given in Equation 2.15,  $L$  is a normalizing constant and  $\theta^p$  is the orientation at point  $p$ , which is equal to the image gradient at that point.  $HOG(k)$  corresponds to the value of the  $k$ th bin in a  $K$ -bin histogram. The value of  $K$  used in the experiments is set to 9, and the value of the normalizing constant,  $L$ , is equal to  $180/K = 20$  [15].

$$\delta(x) = \begin{cases} 1 & \text{if } x = 0, \\ 0 & \text{otherwise.} \end{cases} \quad (2.15)$$

Using Equation 2.14, the steps required to compute HOG descriptor for a given image,  $I$ , is given in Algorithm 2. See Figure 2.11 for graphical description of the method.

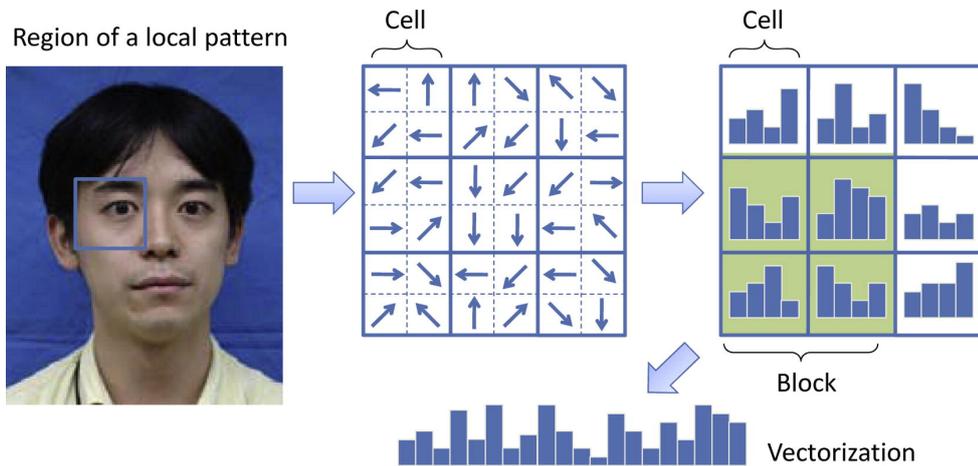


Figure 2.11: Histograms of Oriented Gradients computation (taken from [36]).

In Chapter 4, HOG descriptors are evaluated on several shape databases using square grids ( $T = V$ ) of various sizes. Results are compared with other well-known shape descriptors.

---

**Algorithm 2** Histograms of Oriented Gradients Computation

---

**Require:**  $I$ : A gray-scale image.

$T \times V$ : dimensions of the grid.

Let  $I$  be decomposed to sub-matrices as,

$$\begin{pmatrix} P_1^1 & P_1^2 & \dots & P_1^V \\ P_2^1 & P_2^2 & \dots & P_2^V \\ \dots & \dots & \dots & \dots \\ P_T^1 & P_T^2 & \dots & P_T^V \end{pmatrix}$$

- Let  $H$  be a  $T \times V \times K$  matrix.

**for all**  $t \in 1..T$  **do**

**for all**  $v \in 1..V$  **do**

$H(t, v) \leftarrow HOG(P_t^v)$       – Compute HOG for patch  $P_t^v$

**end for**

**end for**

- Reshape  $H$  to get a vector of size  $T \cdot V \cdot K$ .

---

### 2.2.3 Local Binary Patterns (LBP)

Local Binary Patterns (LBP), albeit computationally simple, is a powerful texture descriptor introduced by Ojala et al. [57]. The basic idea is to compute a bit string in an image patch by thresholding neighbor pixels with respect to the center pixel (see Figure 2.12). The bit string is interpreted as a binary number and converted to a decimal number, which is used to build a histogram. Formally, LBP for a single pixel  $p_c = (x_c, y_c)$  in a gray-scale image  $I$  is defined as follows:

$$LBP_K(p_c) = \sum_{p=0}^{K-1} s(n_p(p_c) - I(p_c))2^p, \quad (2.16)$$

where  $K$  is the number of neighbors in a  $3 \times 3$  window,  $n_p$  is the  $p$ th neighbor of  $p_c$  in clockwise order, and  $s$  is defined as follows:

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.17)$$

Given a gray-scale image  $I$ , assuming  $P$  denotes the set of pixels in  $I$ , LBP is defined as follows:

$$LBP(k) = \sum_{p \in P} \delta(LBP_K(p) - k), \quad (2.18)$$

where  $LBP_K$  is as defined in Equation 2.16 and  $\delta(\cdot)$  is the Kronecker delta given in Equation 2.15. Figure 2.12 depicts the computation of binary pattern for a center pixel in a  $3 \times 3$  window. Algorithm 3 lists a pseudo-code to compute LBP for a given gray scale image  $I$ .

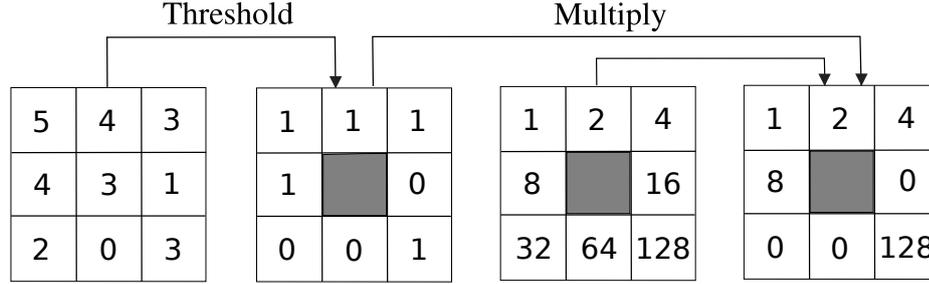


Figure 2.12: General Local Binary Patterns computation (reprinted from [45]). The LBP value for the center pixel is equal to  $1 + 2 + 4 + 8 + 128 = 143$ .

---

**Algorithm 3** Local Binary Patterns Computation

---

**Require:**  $I$ : A gray-scale image.

- Initialize a  $L \times 256$  matrix  $H$ , where  $L = N \cdot M$ .

**for all**  $p \in I$  **do**

- Let  $B$  a vector of size 8.

$N \leftarrow 8\text{neigh}(I, p)$

**for all**  $n_i \in N$  **do**

**if**  $I(p) > I(n_i)$  **then**

$B_i \leftarrow B_i + 1$

**end if**

**end for**

- Convert the binary vector  $B$  to a decimal number to get  $D$ .

$H(k) \leftarrow \#\{D \in \text{bin}(k)\}$      - Increment the  $k$ th bin of histogram

**end for**

---

Local Binary Patterns is modified by Ojala and Pietikäinen [56] to use neighborhoods of various sizes. They proposed a circular neighborhood defined with two parameters, number of sampling points  $P$ , and radius  $R$ . In this context, the notation  $LBP_{P,R}$  means LBP is computed

using  $P$  points sampled on a circle of radius  $R$ . Using this notation, Equation 2.16 is simplified as Equation 2.19.

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p, \quad (2.19)$$

where  $g_c$  is the gray-scale intensity value of the center pixel  $p_c = (x_c, y_c)$  and  $g_p$  is the gray-scale intensity value of the  $p$ th sampled point, the coordinates of which are given by Equation 2.20 and Equation 2.21 (see Figure 2.13).

$$C_{xp} = x_c + R\cos(2\pi p/P) \quad (2.20)$$

$$C_{yp} = y_c - R\sin(2\pi p/P) \quad (2.21)$$

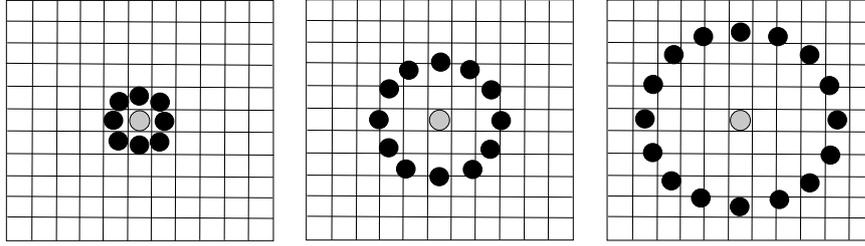


Figure 2.13: Local Binary Patterns  $LBP_{P,R}$  computation (reprinted from [45]). (a)  $(P, R) = (8, 1.0)$ . (b)  $(P, R) = (12, 2.5)$ . (c)  $(P, R) = (16, 4.0)$ .

LBP and its variants are studied extensively in the literature [62]. Another extension to the original descriptor is the introduction of *uniform patterns*, which are shown to reduce the length of the feature vector without losing important information [58]. There are also spatio-temporal representations of LBP. The original LBP descriptor and its variants work in the spatial domain; in other words, they do not consider the time axis. The Volume Local Binary Patterns (VLBP) extracts texture descriptor in three dimensional space  $(X, Y, T)$ , where  $X$  and  $Y$  denote the spatial coordinates and  $T$  denotes the time coordinate [90]. In this way, VLBP combines spatial and temporal information together to describe dynamic textures.

In this thesis, only the original LBP descriptor, as defined in Equation 2.18, is analyzed, and in Chapter 4, results of the original LBP implementation are presented.

## 2.2.4 Fourier Descriptors (FD)

Fourier transformation theory [22], a tool that is widely used in signal processing, has contributed to image processing for many years. Its uses vary from image compression and

image encoding to image filtering and shape description. This study focuses on the application of Fourier transform theory on shape description. This method is known as Fourier descriptors and it is investigated extensively in shape recognition and image retrieval literature [25, 61, 65, 83, 85].

Fourier descriptors are constructed by applying a Fourier transform on a shape signature, which is a one dimensional function derived from shape boundary [84]. Any shape signature can be used to obtain Fourier descriptors, such as complex coordinates, centroid distance and curvature function. It has been shown that Fourier descriptors of centroid distance outperforms other shape signatures [83]. This study, however, investigates Fourier descriptors derived from complex coordinates.

The first step to compute Fourier descriptors from the image is to extract boundary points:  $(x_0, y_0), (x_1, y_1), \dots, (x_{N-1}, y_{N-1})$ . Then, pixel coordinates  $(x_k, y_k)$  are converted to complex coordinates as follows:

$$z_i = x_k + jy_k. \quad (2.22)$$

where  $j^2 = -1$ . Complex numbers,  $z_0, z_1, \dots, z_{N-1}$ , in spatial domain are converted to frequency domain using the Discrete Fourier Transform [63] as follows:

$$Z_k = \sum_{n=0}^{N-1} z_n e^{-j\frac{2\pi}{N}kn}. \quad (2.23)$$

After complex coordinates are transformed to frequency domain, they are normalized by the first term,  $Z_0$  as follows:

$$Z_k = \frac{|Z_k|}{|Z_0|}. \quad k = 1, \dots, N - 1 \quad (2.24)$$

As a result, Fourier descriptors for the image, are formed using the first  $K$  terms of the series  $Z_k$  where  $k = 1, \dots, N - 1$ :

$$FD = [Z_1, Z_2, \dots, Z_K]. \quad (2.25)$$

## CHAPTER 3

### METHODS

The previous chapter introduced the literature on shape representation and the state-of-the-art shape descriptors. In this chapter, the two novel shape descriptors, namely Intersection Consistency Histogram and Global Binary Patterns, which are proposed in this thesis, are presented in detail. First, the concept of Intersection Consistency (IC) and its application to shape description is discussed. Then, a computationally simple, novel shape descriptor called Global Binary Patterns is described.

#### 3.1 Intersection Consistency Histogram (ICH)

Intersection Consistency Histogram is a shape representation technique based on a local consistency measure called Intersection Consistency (IC). The descriptor creates a histogram of IC values for each pixel in a window to build a rich global descriptor.

##### 3.1.1 Intersection Consistency (IC)

Intersection consistency is a local consistency measure, proposed by Kalkan et al. [32], to improve junction localization. It measures whether the pixels in a window point towards the center or not. Formally, intersection consistency at point  $p_c$  in an image patch  $P$  is defined as follows.

$$IC(p_c) = \frac{1}{|P|} \sum_{p \in P} [m(p)]^k \left[ 1 - \frac{d(l^p, p_c)}{d(p, p_c)} \right], \quad (3.1)$$

where  $l^p$  is the line passing through pixel  $p$ ,  $m(p)$  is the magnitude of the pixel  $p$ ,  $d(l^p, p_c)$  is the point-to-line distance between  $p_c$  and  $l^p$ ,  $d(p, p_c)$  is the point-to-point distance between  $p$  and  $p_c$ , and  $k$  is a parameter, empirically determined to be 2 [32].

The IC measure for the center pixel  $p_c$  is computed by taking the average value of the ratio of the point-to-line distance to the point-to-point distance with respect to the center pixel for all other pixels in a window (see Figure 3.1). Observe that, if the hypothetical line going through  $p$  passes the center point  $p_c$ , this ratio becomes zero, as the point-to-line distance evaluates to zero, the IC measure gets its highest possible value.

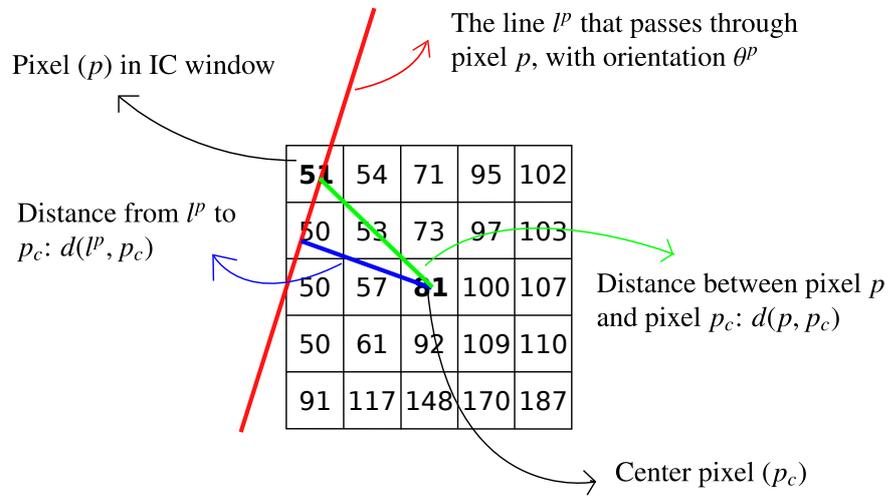


Figure 3.1: Intersection Consistency computation inside a  $5 \times 5$  image patch (adapted from [91]).

In the original paper [32], IC is used to improve junction localization. In a follow-up work, IC is proposed as a method to improve edge detection [91]. In this thesis, the IC measure is utilized to build a rich shape descriptor called Intersection Consistency Histogram.

### 3.1.2 ICH Algorithm

In this thesis, the original IC equation is slightly modified to construct a rich shape descriptor, which is referred to as Intersection Consistency Histogram. ICH for a gray-scale image  $I$  is

<sup>1</sup> The line equation for  $l^p$ , is computed using the image gradient at point  $p$ . In other words,  $l^p$  is computed using orientation at the pixel  $p$ , denoted  $\theta^p$ .

defined as follows.

$$ICH(l) = \sum_{p \in P} \delta(\lfloor L \cdot IC(p) \rfloor - l), \quad (3.2)$$

where  $\delta(\cdot)$  is the Kronecker delta defined in Equation 2.15,  $P$  denotes the set of pixels in  $I$ ,  $IC$  denotes the intersection consistency formula defined in Equation 3.1, and  $l$  is the value of the histogram bin between 1 and  $L$  (the value of  $L$  is set to 10 for the implementation)<sup>2</sup>. This equation builds a histogram with  $L$  bins. Algorithm 4 presents the steps required to compute Intersection Consistency Histogram for a given image  $I$ . See Figure 1.2 (from Chapter 1) for a graphical illustration of the algorithm.

---

**Algorithm 4** Intersection Consistency Histogram Computation.

---

**Require:**  $I$ : A gray-scale image.

- Let  $ICH$  be a vector of size  $L$ .

**for all**  $i \in 1..N$  **do**

**for all**  $j \in 1..M$  **do**

$p \leftarrow I(i, j)$

$ic \leftarrow \lfloor L \cdot IC(p) \rfloor$

$ICH(ic) \leftarrow ICH(ic) + 1$     –See Equation 3.2

**end for**

**end for**

---

ICH produces a single  $L$ -bin histogram, which alone cannot successfully capture the whole shape structure. Therefore, it is extended in Algorithm 5 as a grid-based method. This version is based on dividing the image into grids and running Algorithm 4 separately for each grid. In the grid-based method, the ICH for the whole image is the concatenation of the ICH for each grid. In the rest of the thesis, the grid-based method is used.

Figure 3.2 displays ICH descriptors for sample images from the MPEG-7 database [38]. ICH descriptors are extracted using Algorithm 5 with parameters  $T$  and  $V$  set to 4. The length of each descriptor vector is, thus, equal to  $T \cdot V \cdot L = 4 \times 4 \times 10 = 160$ <sup>3</sup>. As Figure 3.2 suggests descriptors of similar images form a uniform distribution as illustrated in Figures 3.2(a) and 3.2(b). On the other hand, descriptors in Figures 3.2(c) and 3.2(d) constitute a more scattered distribution.

---

<sup>2</sup> IC values are in the range 0 – 1 since pixel magnitudes,  $m(p)$ , are normalized.

<sup>3</sup> The number of bins,  $L$ , is set to 10 for the experiments conducted in Chapter 4.

---

**Algorithm 5** Grid-based Intersection Consistency Histogram Computation.

---

**Require:**  $I$ : A gray-scale image.

$T \times V$ : dimensions of the grid.

Let  $I$  be decomposed to sub-matrices (grids) as,

$$\begin{pmatrix} P_1^1 & P_1^2 & \dots & P_1^V \\ P_2^1 & P_2^2 & \dots & P_2^V \\ \dots & \dots & \dots & \dots \\ P_T^1 & P_T^2 & \dots & P_T^V \end{pmatrix}$$

- Let  $ICH$  be a  $T \times V \times L$  matrix.

**for all**  $t \in 1..T$  **do**

**for all**  $v \in 1..V$  **do**

    - Assume  $P_t^v$  is an  $M \times N$  matrix

$p \leftarrow P_t^v(\frac{M}{2}, \frac{N}{2})$     – Center point of the image patch  $P_t^v$

$ic \leftarrow \lfloor L \cdot IC(p) \rfloor$     – See Equation 3.1

$ICH(t, v, ic) \leftarrow ICH(t, v, ic) + 1$     – See Equation 3.2

**end for**

**end for**

- Reshape  $H$  in row-major order to get a vector of size  $T \cdot V \cdot L$ .

---

## 3.2 Global Binary Patterns (GBP)

Global Binary Patterns is a simple, yet efficient, shape descriptor similar to Local Binary Patterns [57]. The method creates a set of bit strings for any direction of a thresholded binary image and interprets these bit strings as binary numbers to build a global descriptor (see Figure 3.3).

### 3.2.1 GBP Algorithm

In its simplest form, Global Binary Patterns of a row,  $r$ , of a binary image  $I$  is defined as follows:

$$GBP_h(r) = \sum_{j=1}^R I(r, j) \cdot 2^{j-1}, \quad (3.3)$$

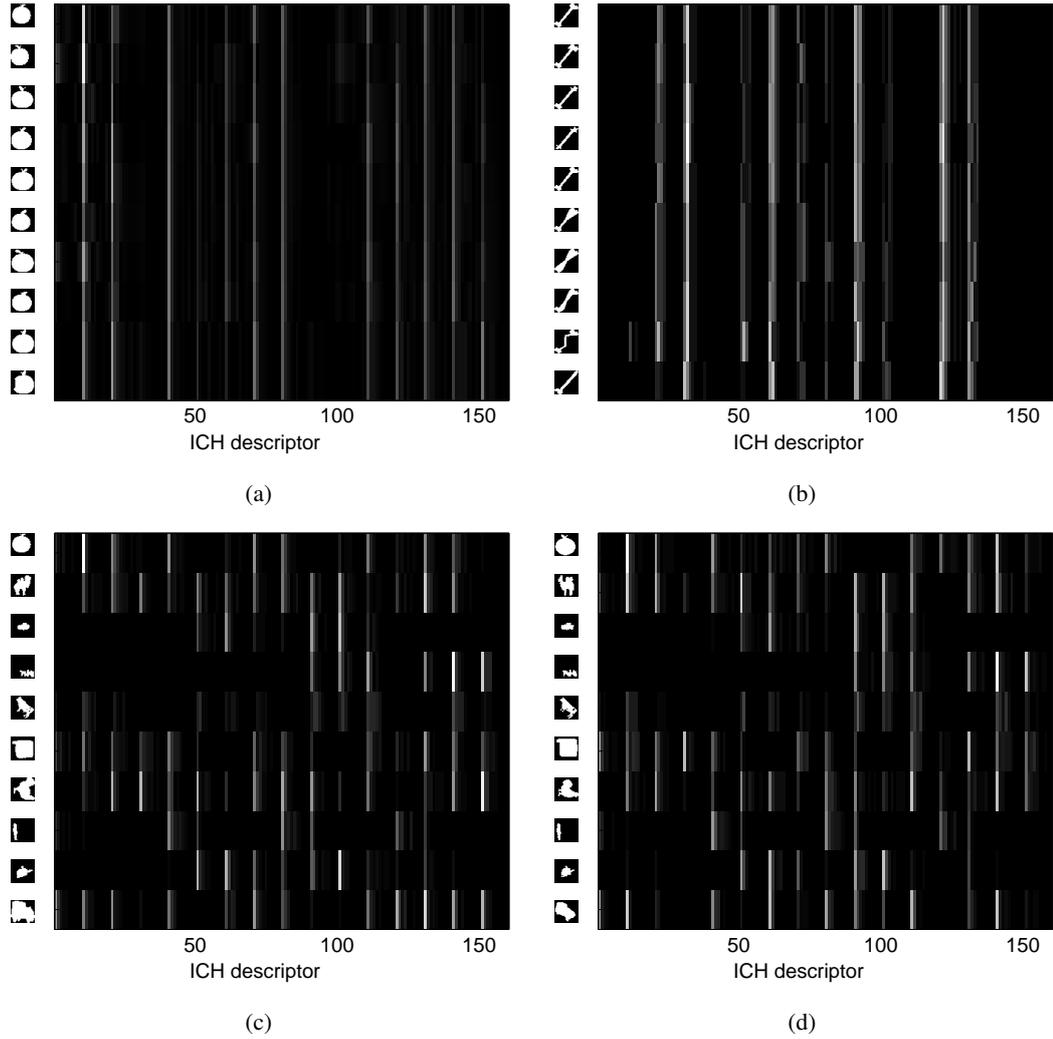


Figure 3.2: Sample images (resized to  $32 \times 32$ ) from the MPEG-7 [38] database and their corresponding ICH descriptors (brighter intensities indicate higher values). Descriptors are extracted using Algorithm 5 with parameters  $T$  and  $V$  set to 4. The length of each descriptor vector is, thus, equal to  $T \cdot V \cdot L = 4 \times 4 \times 10 = 160$ . Figures (a) and (b) contain shapes from categories *Apple* and *Bone* respectively. On the other hand, Figures (c) and (d) contain shapes from different categories. ICH descriptors in Figures (a) and (b) form a more uniform distribution compared to descriptors in Figures (c) and (d).

where  $R$  is the number of columns in image  $I$ .  $GBP_h$  computes Global Binary Patterns along horizontal direction. Algorithm 6 lists the steps required to compute the  $GBP_h$  descriptor. Similarly, GBP along vertical direction, denoted  $GBP_v$ , is defined as follows:

$$GBP_v(c) = \sum_{i=1}^C I(i, c) \cdot 2^{i-1}, \quad (3.4)$$

where  $C$  is the number of rows in image  $I$ . See Figure 3.3 for an illustration of GBP computation along horizontal and vertical directions, denoted  $GBP_{hv}$ .

---

**Algorithm 6** Global Binary Patterns ( $GBP_h$ ) Computation
 

---

**Require:**  $I$ : A  $R \times C$  thresholded image.

- Let  $GBP_h$  be a vector of size  $N$ .

**for all**  $i \in 1..R$  **do**

**for all**  $j \in 1..C$  **do**

$$GBP_h(i) \leftarrow GBP_h(i) + I(i, j) \cdot 2^{j-1}$$

**end for**

**end for**

---

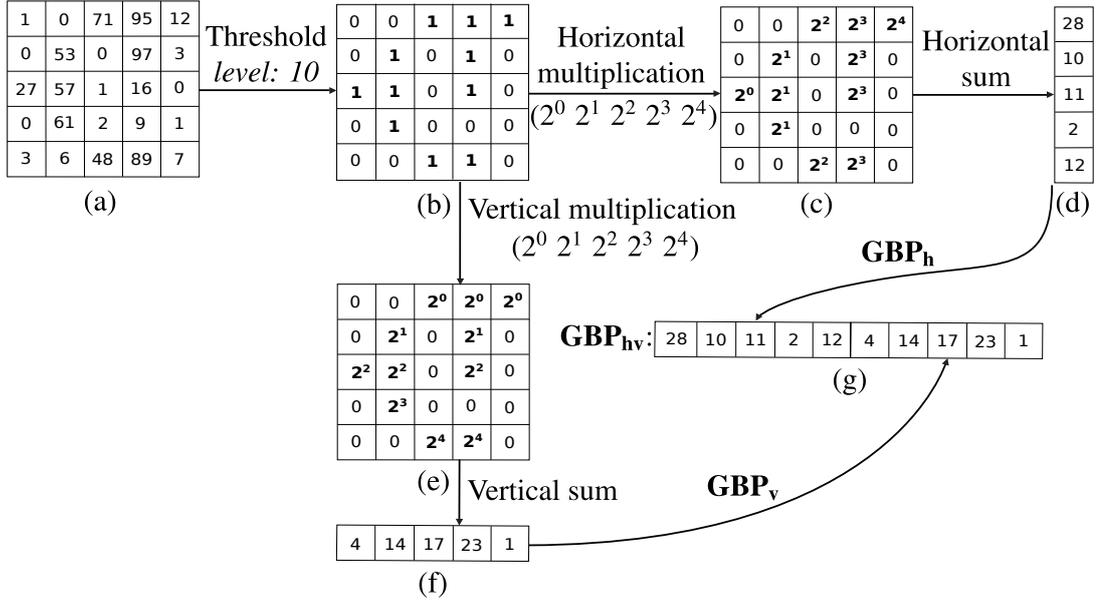


Figure 3.3: Global Binary Patterns computation along horizontal and vertical directions, denoted  $GBP_h$  and  $GBP_v$  respectively. (a) The original image. (b) After thresholding with brightness level 10. (c) After rows are multiplied by powers of two: ( $2^0 2^1 2^2 2^3 2^4$ ). (d) After each row is summed horizontally. (e) After columns are multiplied by powers of two: ( $2^0 2^1 2^2 2^3 2^4$ ). (f) After columns are summed vertically. (g) Resulting GBP descriptor.

$GBP_h$  and  $GBP_v$  are defined along horizontal and vertical directions. In fact, GBP can be constructed along any arbitrary direction, which may effect (as investigated in Chapter 4) the performance of the descriptor. To this end, formulations presented in Equations 3.3 and 3.4 are extended to incorporate projection along an arbitrary direction with orientation  $\theta$  (see Figure 3.4). Let  $l_\theta^{p_o}$  be the line, with orientation  $\theta$ , that passes through  $p_o$ , which is the bottom-right pixel in the image  $I$ ,  $GBP_\theta$  for an image  $I$  is defined as follows:

$$GBP_\theta(k) = \sum_{p \in I} \delta \left( L \frac{x^p - \min(x^{p_{ij}})}{\max(x^{p_{ij}}) - \min(x^{p_{ij}})} - k \right) \cdot I(p) \cdot 2^{d^p}, \quad (3.5)$$

where  $\delta(\cdot)$  is the Kronecker delta defined in Equation 2.15,  $d^p$  is the point-to-line distance between the pixel  $p$  and the line  $l_\theta^{p_o}$ ,  $x^p$  is the projection of the pixel  $p$  onto the line  $l_\theta^{p_o}$ , and  $L$  is the desired length of the  $GBP_\theta$  descriptor ( $L$  is taken to be 32 and 64 for the experiments conducted in this study).

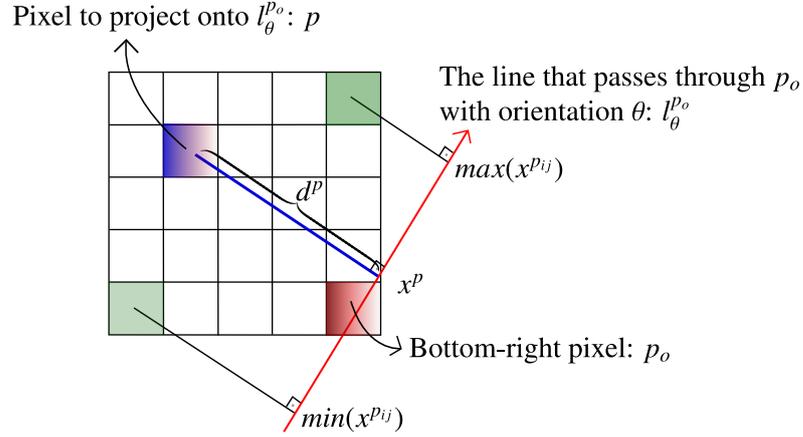


Figure 3.4: Illustration of the projection of a pixel onto a line that passes through the pixel  $p_o$  with orientation  $\theta$ .

Using Equation 3.5, it is possible to use any number of projections to form the GBP descriptor. Analysis in this study is performed using combinations of horizontal, vertical, diagonal and principal directions defined as follows.

$$GBP_h(I) = GBP_{90}(I), \quad (3.6)$$

$$GBP_{h'}(I) = GBP_{90}^{R_h}(I), \quad (3.7)$$

$$GBP_v(I) = GBP_0(I), \quad (3.8)$$

$$GBP_{v'}(I) = GBP_0^{R_v}(I), \quad (3.9)$$

$$GBP_d(I) = GBP_{45}(I), \quad (3.10)$$

$$GBP_p(I) = GBP_\phi(I), \quad (3.11)$$

where  $R_h$  and  $R_v$  denote the reverse of the image in horizontal and vertical directions, and  $\phi$  is the orientation of the principal axis of the shape, which is computed using principal component analysis (PCA) [60]. Based on the nature of the problem, different GBPs as defined in Equations 3.6, 3.8, 3.10 and 3.11 can be concatenated as a single feature descriptor.

Figure 3.5 displays GBP descriptors for sample images from the MPEG-7 database [38].

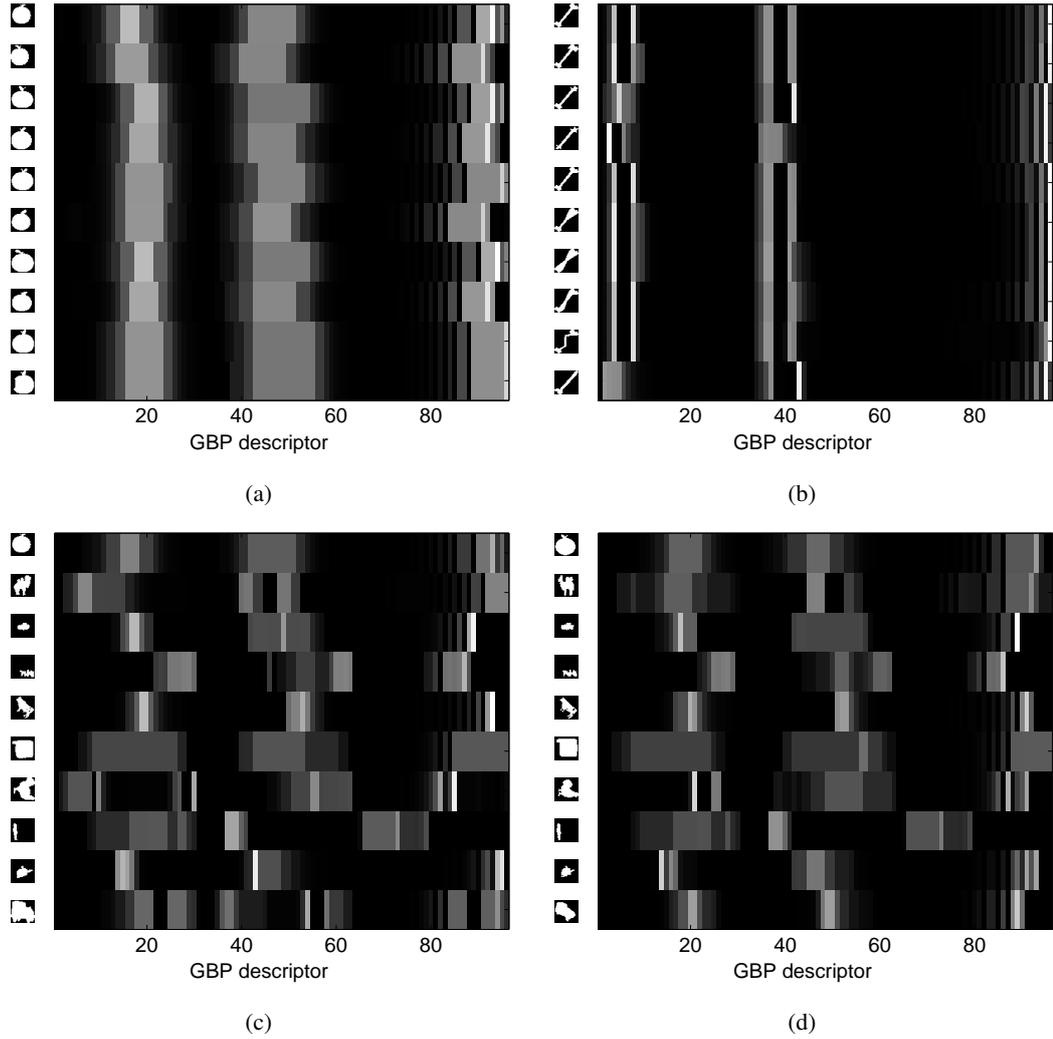


Figure 3.5: Sample images (resized to  $32 \times 32$ ) from the MPEG-7 [38] database and their corresponding GBP descriptors (brighter intensities indicate higher values). Descriptors are constructed by concatenating  $GBP_h$ ,  $GBP_v$  and  $GBP_d$  descriptors (Equations 3.6, 3.8 and 3.10). Figures (a) and (b) contain shapes from categories *Apple* and *Bone* respectively. On the other hand, Figures (c) and (d) contain shapes from different categories. GBP descriptors in Figures (a) and (b) form a more uniform distribution compared to descriptors in Figures (c) and (d).

GBP descriptors are constructed using horizontal, vertical and diagonal projections:  $GBP_h \oplus GBP_v \oplus GBP_d$  ( $\oplus$  concatenates two vectors). In this figure, the length of each GBP descriptor is equal to  $3 \cdot C$  for images with dimensions  $C \times C$ . Therefore, the length of the descriptors is equal to  $3 \cdot 32 = 96$  since images are resized to  $32 \times 32$ . As seen from the figure, descriptors extracted from similar shapes form a more uniform distribution compared to those extracted from dissimilar shapes.

### 3.2.2 Properties of GBP

GBP descriptors,  $GBP_h$ ,  $GBP_v$  and  $GBP_d$ , are not invariant to translation, scale and rotation changes. Translation invariance can be achieved by normalizing the descriptor vector or by removing empty rows and columns before applying Equation 3.3. Rotation invariance is provided, as presented in Equation 3.5, by incorporating projection along the principal direction, which is computed using principal component analysis (PCA) [60].

GBP is a powerful, efficient and simple descriptor. Computational requirements of GBP are very low. The memory requirement is  $O(N)$  where  $N$  is the number of pixels that exist along an axis. The running time complexity is  $O(N \cdot M)$  for a  $N \times M$  image. In addition to high computational efficiency, it is also a very straightforward algorithm.

## CHAPTER 4

### EXPERIMENTS AND RESULTS

In the previous chapter, two novel shape descriptors, namely Intersection Consistency Histogram (ICH) and Global Binary Patterns (GBP), were proposed and explained in detail. This chapter analyzes these methods in detail and compares them with state-of-the-art methods on several widely used databases. First, database parameters used in the experiments are introduced, and then image retrieval and running time performances of each method are analyzed in comparison with those of Shape Context, Histograms of Oriented Gradients, Local Binary Patterns and Fourier Descriptors.

#### 4.1 The Databases

The experiments in this study are carried out using five publicly available databases, which are selected according to their popularity in shape description and representation research.

##### 4.1.1 Brown University Kimia-99 Shape Database

The Brown University Kimia-99 shape database [67] contains a total of 99 images from 9 categories. Figure 4.1 displays thumbnails of all images from the database.

Evaluation on Kimia-99 shape database is performed using a score called *TopRank*, in which every shape is compared to all other shapes and number of correctly classified  $N$  nearest neighbors for each query image is reported in a tabular form. Since the database contains 11 images from each of 9 categories, the maximum score for each neighbor is 99. Table 4.1 lists the best reported results on this database [6].

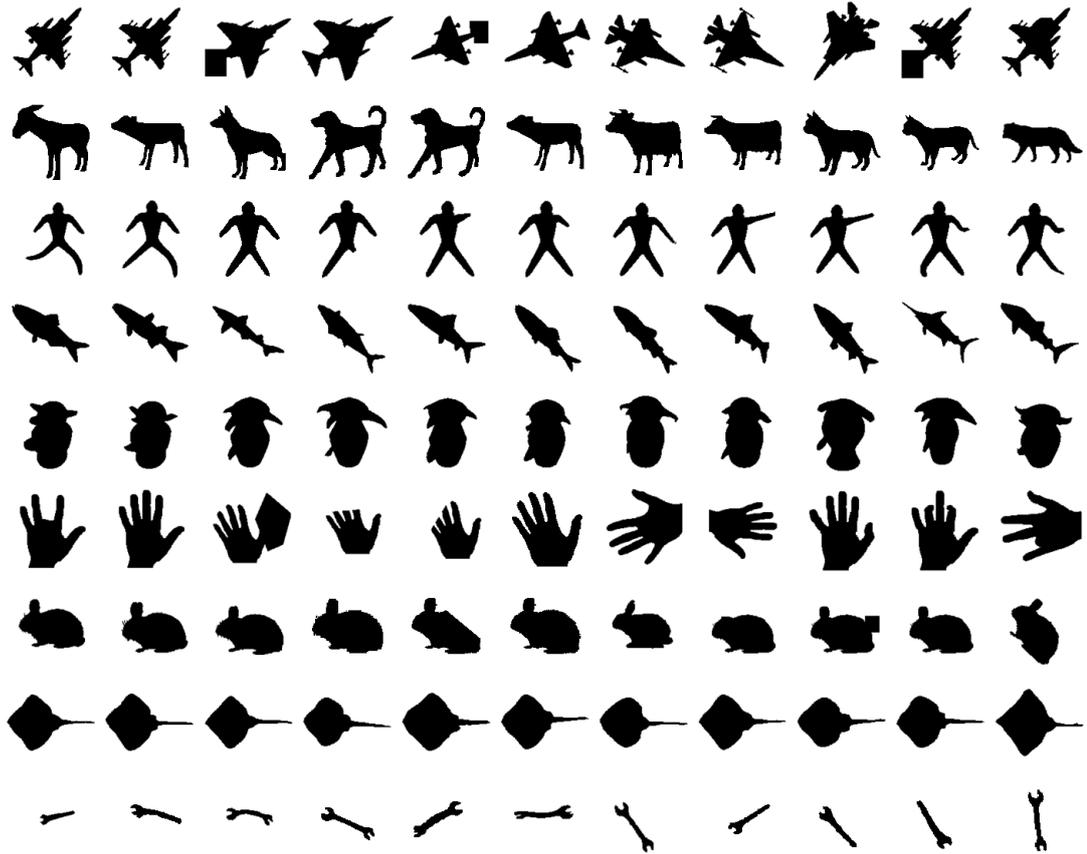


Figure 4.1: Images from the Brown University Kimia-99 Shape Database.

#### 4.1.2 Brown University Kimia-216 Shape Database

The Brown University Kimia-216 shape database [67] contains a total of 216 images from 18 categories. Figure 4.2 displays 10 sample images from each category.

As is the case for Kimia-99 shape database, evaluation on Kimia-216 database is performed using the *TopRank* score. Since this database contains 12 images from each of 18 categories, the maximum possible score for each neighbor is 216. Table 4.2 lists some of the best performing algorithms on this database [5].

#### 4.1.3 MPEG-7 CE Shape-1 Part-B Database

The MPEG-7 CE Shape-1 Part-B database [38] contains a total of 1400 images from 70 categories. Figure 4.3 depicts a subset of images each from a different category.

Table 4.1: TopRank scores of different methods on the Brown University Kimia-99 Shape Database (taken from [6]).

Algorithm	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
Graph Transduction [6]	99	99	99	99	99	99	99	99	97	99
Symbolic Rep. [16]	99	99	99	98	99	98	98	95	96	94
Shape Tree [20]	99	99	99	99	99	99	99	97	93	86
Triangle Area [1]	99	99	99	98	98	97	98	95	93	80
IDSC [41]	99	99	99	98	98	97	97	98	94	79

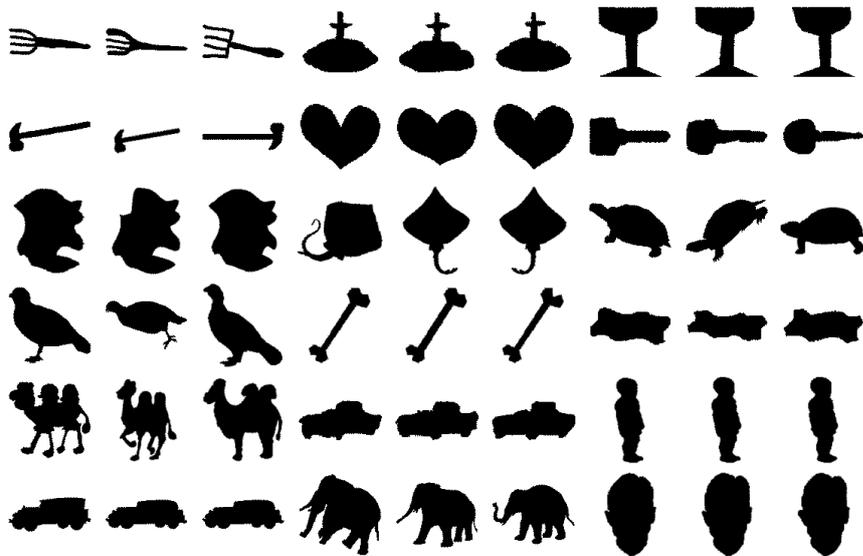


Figure 4.2: Arbitrary images from the Brown University Kimia-216 Shape Database.

Evaluation on MPEG-7 database is performed using the so-called *Bull's Eye Score*, in which every shape is compared to all other shapes and top 40 similar results are collected. The final score is the ratio of correct matches to the highest possible number of matches, which is  $20 \times 70$  since the database contains 20 images from each category. Table 4.3 lists the best reported results on MPEG-7 database [6].

#### 4.1.4 Natural Silhouettes Database

The Natural Silhouettes database [23] contains a total of 490 images from 12 categories. Figure 4.4 shows 8 sample images from each category.

Table 4.2: TopRank Scores of different methods on the Brown University Kimia-216 Shape Database (taken from [5]).

Algorithm	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
Path Similarity [5]	216	216	215	216	213	210	210	207	205	191
Shock Edit [66]	216	216	216	215	210	210	207	204	200	187
SC [8]	214	209	205	197	191	178	161	144	131	101

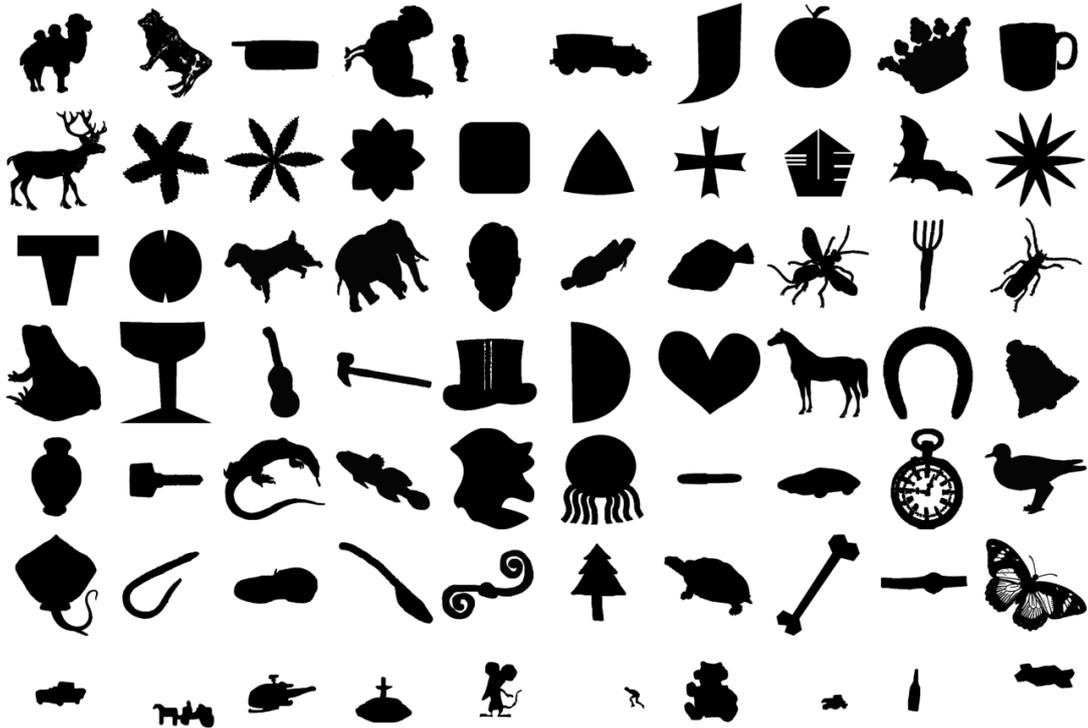


Figure 4.3: Arbitrary images from the MPEG-7 CE Shape-1 Part-B Database.

#### 4.1.5 Columbia Object Image Library (Coil-100) Database

The Coil-100 Database [54] contains a total of 7200 images from 100 categories, each of which includes a single object viewed from different angles. In this study, 25 images from 20 objects are selected to create a subset of this database. Each object is viewed with rotation increments of  $10^\circ$ , from  $0^\circ$  to  $240^\circ$ , for a total of 25 images per object. Figure 4.5 shows sample images from the database.

Evaluation on the Coil-100 database is usually performed by using four views for each object. The first two results listed in Table 4.4 use eight to ten images, and the last method uses

Table 4.3: Bull’s Eye test scores of different methods on the MPEG-7 CE Shape-1 Part-B Database (adapted from [6]).

Algorithm	Score
Diffusion Process [81]	93.32%
Graph Transduction [6]	91.61%
Shape Tree [20]	87.70%
Triangle Area [1]	87.23%
Hierarchical Procrustes [46]	86.35%
Symbolic Representation[16]	85.92%
Inner Distance [41]	85.40%

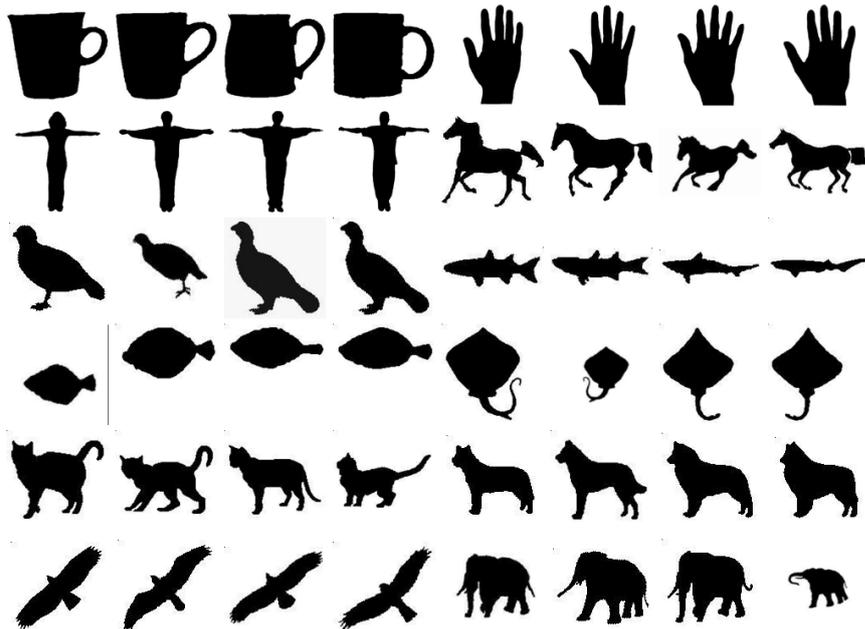


Figure 4.4: Arbitrary images from the Natural Silhouettes database.

only four images. Experiments conducted in this thesis, on the other hand, use 25 images per object; thus, results reported in this study are not consistent with the results provided in Table 4.4.

## 4.2 Performance Measurements

Experimental evaluation is based on image retrieval performance. Performance of algorithms are presented using several methods: *Bull’s Eye Score*, *TopRank Score* and *Receiver Operating Characteristics* curve. In addition, sample query results for each database are provided to



Figure 4.5: Arbitrary images from the CoIL-100 Database.

Table 4.4: Accuracies of different methods on the CoIL-100 Database.

Algorithm	Accuracy
Nearest Prime Simplicial Complex [88]	97.2%
Multiple-View Object Recognition [80]	95.0%
Deep Learning from Temporal Coherence [49]	92.5%

illustrate the top  $K$  query results of each algorithm for randomly selected images.

#### 4.2.1 Bull’s Eye Score

*Bull’s Eye Score* is computed by comparing each image in a database to all other images and the percentage of correct labels among top  $N$  results are reported for each category.  $N$  is twice the number of images available in a category. Therefore, the highest possible score is  $N/2 \times M$  where  $M$  is the number of distinct categories.

#### 4.2.2 TopRank Score

*TopRank Score* is computed by counting the number of  $N$  nearest neighbors that are correctly classified for all query images. This is accomplished by comparing each image to all other images and taking the top  $N$  most similar images for each query image. The number of correct

classifications for each of the  $N$  neighbors is reported in a tabular form.

### 4.2.3 Receiver Operating Characteristic

Receiver operating characteristics (ROC) curves are graphical plots that illustrate the performance of a classifier. ROC curves are especially useful in cases where evaluation metrics such as *accuracy*, *precision* or *recall* are not informative enough to judge the quality of a classifier, mostly because they require a decision *threshold* to be set arbitrarily [48, 64]. A ROC curve is plotted by computing the ratio of *sensitivity*, also known as *true positive rate* or *recall rate*, to *specificity*, also known as *false positive rate*, for various *thresholds*. ROC curves are well-suited to classifiers that output scores or probabilities rather than labels.

Definitions of *accuracy* (*accuracy*), *true positive rate* (*tpr*) and *false positive rate* (*fpr*) are presented in Equations 4.1, 4.2 and 4.3 respectively (see Table 4.5).

$$accuracy = \frac{tp + tn}{tp + fp + tn + fn}, \quad (4.1)$$

$$tpr = \frac{tp}{tp + fn}, \quad (4.2)$$

$$fpr = \frac{fn}{tn + fp}, \quad (4.3)$$

where  $tp$ ,  $fp$ ,  $tn$  and  $fn$  represent *True Positive*, *False Positive*, *True Negative* and *False Negative* respectively. Semantics of these metrics are illustrated in Table 4.5. ROC curves are plotted setting various thresholds for classifier outputs. This means, ROC curve plotters expect the output of classifiers to be a similarity metric, preferably normalized to reflect probabilities. Algorithms studied in this thesis produce distance or cost between images; therefore, these values need to be converted to similarity values. This is accomplished by first normalizing costs and then by converting distance to similarity as proposed by Shepard [68].

$$s(A, B) = e^{-d(A,B)}. \quad (4.4)$$

## 4.3 Results

In this section, comparative analysis of six shape descriptors, two of which are proposed in this work, is presented in detail. Experiments are performed on five widely-used databases

Table 4.5: Tabular Description of Classification Metrics

		Prediction outcome		total
		p	n	
Actual value	p'	True Positive (tp)	False Negative (fn)	P'
	n'	False Positive (fp)	True Negative (tn)	N'
total		P	N	

using several parameters for each method, and retrieval and running time performances are reported using various metrics and plots.

### 4.3.1 Parameter Investigation

This section is devoted to the investigation of parameters for the proposed methods. Parameter studies are carried out on five databases, and performance curves are reported separately for each database. The effect of these parameters on the retrieval performance are analyzed and the best-performing parameter settings are reported for both Global Binary Patterns (GBP) and Intersection Consistency Histogram (ICH). Parameter investigation for the methods selected for comparison with GBP and ICH are presented in Appendix B. In the following section, methods are compared using the best performing parameter settings determined in this section.

#### 4.3.1.1 Shape Context

Shape Context, as outlined in Algorithm 1, has a single parameter, namely the *Number of Sampling Points*. As shown in Table 4.6, there are three parameter settings for Shape Context, named *SC-1*, *SC-2* and *SC-3*. See Section B.1 in Appendix B for the effects of these parameters on the retrieval performance of Shape Context.

Table 4.6: SC Parameter Settings.

Method ID	Number of Sampling Points	Image Dimensions
SC-1	50	$32 \times 32$
SC-2	50	$64 \times 64$
SC-3	100	$64 \times 64$

Table 4.7: HOG Parameter Settings. **Grid Size** specifies the number of rows and columns in a grid.

Method ID	Grid Size	Image Dimensions
HOG-1	$2 \times 2$	$32 \times 32$
HOG-2	$4 \times 4$	$32 \times 32$
HOG-3	$8 \times 8$	$32 \times 32$
HOG-4	$2 \times 2$	$64 \times 64$
HOG-5	$4 \times 4$	$64 \times 64$
HOG-6	$8 \times 8$	$64 \times 64$

#### 4.3.1.2 Histograms of Oriented Gradients

Histograms of Oriented Gradients computes histograms of image gradients in image patches, which are obtained by dividing the image into grid cells. HOG, as described in Algorithm 2, requires a gray-scale image and dimensions of the grid. The *Grid Size* column in Table 4.7 lists various grid dimensions used in the experiments<sup>1</sup>, and the *Image Dimensions* column contains image dimensions. See Section B.2 in Appendix B for the effects of these parameters on the retrieval performance of Histograms of Oriented Gradients.

#### 4.3.1.3 Local Binary Patterns

Local Binary Patterns, as outlined in Algorithm 3, does not require any parameters. However, experimental evaluations are performed by dividing the image into grids and running LBP

<sup>1</sup> A grid with size  $T \times V$  has  $T \cdot V$  cells

Table 4.8: LBP Parameter Settings. **Grid Size** specifies the number of rows and columns in a grid.

<b>Method ID</b>	<b>Grid Size</b>	<b>Image Dimensions</b>
LBP-1	$1 \times 1$	$32 \times 32$
LBP-2	$2 \times 2$	$32 \times 32$
LBP-3	$4 \times 4$	$32 \times 32$
LBP-4	$1 \times 1$	$64 \times 64$
LBP-5	$2 \times 2$	$64 \times 64$
LBP-6	$4 \times 4$	$64 \times 64$

separately for each sub-grid. This is specified with the *Grid Size* parameter. For instance, for a  $32 \times 32$  image, with a grid size of  $2 \times 2$ , the algorithm is run 4 times for each sub-grid. See Section B.3 in Appendix B for the effects of these parameters on the retrieval performance of Local Binary Patterns.

Table 4.9: FD Parameter Settings.

<b>Method ID</b>	<b>Number of Fourier Terms</b>	<b>Image Dimensions</b>
FD-1	5	$32 \times 32$
FD-2	20	$32 \times 32$
FD-3	40	$32 \times 32$
FD-4	50	$32 \times 32$
FD-5	5	$64 \times 64$
FD-6	20	$64 \times 64$
FD-7	40	$64 \times 64$
FD-8	50	$64 \times 64$

#### 4.3.1.4 Fourier Descriptors

Fourier Descriptors algorithm has a single parameter: the *Number of Fourier Terms*. The effect of this parameter on performance is tested using several image dimensions (see Table 4.9).

Refer to Section B.4 in Appendix B for the effects of these parameters on the retrieval performance of Fourier Descriptors.

Table 4.10: ICH Parameter Settings. **Grid Size** specifies the number of rows and columns in a grid.

<b>Method ID</b>	<b>Grid Size</b>	<b>Image Dimensions</b>
ICH-1	$2 \times 2$	$32 \times 32$
ICH-2	$4 \times 4$	$32 \times 32$
ICH-3	$8 \times 8$	$32 \times 32$
ICH-4	$2 \times 2$	$64 \times 64$
ICH-5	$4 \times 4$	$64 \times 64$
ICH-6	$8 \times 8$	$64 \times 64$

Table 4.11: GBP Parameter Settings. **Grid Size** specifies the number of rows and columns in a grid.

<b>Method ID</b>	<b>Grid Size</b>	<b>Image Dimensions</b>	<b>GBPs Used</b>	<b>Method ID</b>	<b>Grid Size</b>	<b>Image Dimensions</b>	<b>GBPs Used</b>
GBP-1	$1 \times 1$	$32 \times 32$	$GBP_{hv}$	GBP-13	$1 \times 1$	$64 \times 64$	$GBP_d$
GBP-2	$2 \times 2$	$32 \times 32$	$GBP_{hv}$	GBP-14	$2 \times 2$	$64 \times 64$	$GBP_d$
GBP-3	$4 \times 4$	$32 \times 32$	$GBP_{hv}$	GBP-15	$4 \times 4$	$64 \times 64$	$GBP_d$
GBP-4	$1 \times 1$	$32 \times 32$	$GBP_d$	GBP-16	$1 \times 1$	$64 \times 64$	$GBP_p$
GBP-5	$2 \times 2$	$32 \times 32$	$GBP_d$	GBP-17	$2 \times 2$	$64 \times 64$	$GBP_p$
GBP-6	$4 \times 4$	$32 \times 32$	$GBP_d$	GBP-18	$4 \times 4$	$64 \times 64$	$GBP_p$
GBP-7	$1 \times 1$	$32 \times 32$	$GBP_p$	GBP-19	$1 \times 1$	$32 \times 32$	$GBP_{hv'}$
GBP-8	$2 \times 2$	$32 \times 32$	$GBP_p$	GBP-20	$2 \times 2$	$32 \times 32$	$GBP_{hv'}$
GBP-9	$4 \times 4$	$32 \times 32$	$GBP_p$	GBP-21	$4 \times 4$	$32 \times 32$	$GBP_{hv'}$
GBP-10	$1 \times 1$	$64 \times 64$	$GBP_{hv}$	GBP-22	$1 \times 1$	$64 \times 64$	$GBP_{hv'}$
GBP-11	$2 \times 2$	$64 \times 64$	$GBP_{hv}$	GBP-23	$2 \times 2$	$64 \times 64$	$GBP_{hv'}$
GBP-12	$4 \times 4$	$64 \times 64$	$GBP_{hv}$	GBP-24	$4 \times 4$	$64 \times 64$	$GBP_{hv'}$

### 4.3.1.5 Intersection Consistency Histogram

Similar to Histograms of Oriented Gradients, Intersection Consistency Histogram computes histograms for image patches, which are obtained by dividing the image into grid cells. ICH, as described in Algorithm 5, requires a gray-scale image and dimensions of the grid. The *Grid Size* column in Table 4.7 lists various grid sizes used in the experiments, and the *Image Dimensions* column contains image dimensions.

Figure 4.6 includes performance curves for Intersection Consistency Histogram. Different parameter settings for ICH perform close to each other. However, *IC-1* and *IC-4*, which are the two methods that use the smallest grid size ( $2 \times 2$ ), perform poorly compared to others. For this reason, it is safe to argue that using large cells (larger than  $16 \times 16$ ) worsens the performance of ICH.

### 4.3.1.6 Global Binary Patterns

Global Binary Patterns can be applied using combinations of several projections listed in Equations 3.6, 3.7, 3.8, 3.9, 3.10 and 3.11. Evaluations in this study are performed using the following GBP descriptors:  $GBP_d$  given in Equation 3.10,  $GBP_p$  given in Equation 3.11, and  $GBP_{hv}$  and  $GBP_{hv'}$  defined as follows:

$$GBP_{hv}(I) = GBP_h(I) \oplus GBP_v(I) \quad (4.5)$$

$$GBP_{hv'}(I) = GBP_h(I) \oplus GBP_v(I) \oplus GBP_{h'} \oplus GBP_{v'} \quad (4.6)$$

where  $\oplus$  concatenates two vectors. GBP runs are performed using various image dimensions and grid sizes as specified in Table 4.11. The column *Grid Size* specifies the size of the grid, and the *Image Dimensions* column lists image dimensions.

Tables 4.12 and 4.13 include the retrieval performance score of each GBP parameter setting listed in Table 4.11. As seen from the tables, *GBP-19*, *GBP-20* and *GBP-21*, outperform other parameter settings on almost all databases. These descriptors use several projections along vertical and horizontal directions ( $GBP_{hv'}$ ), which are shown to capture the shape information better.

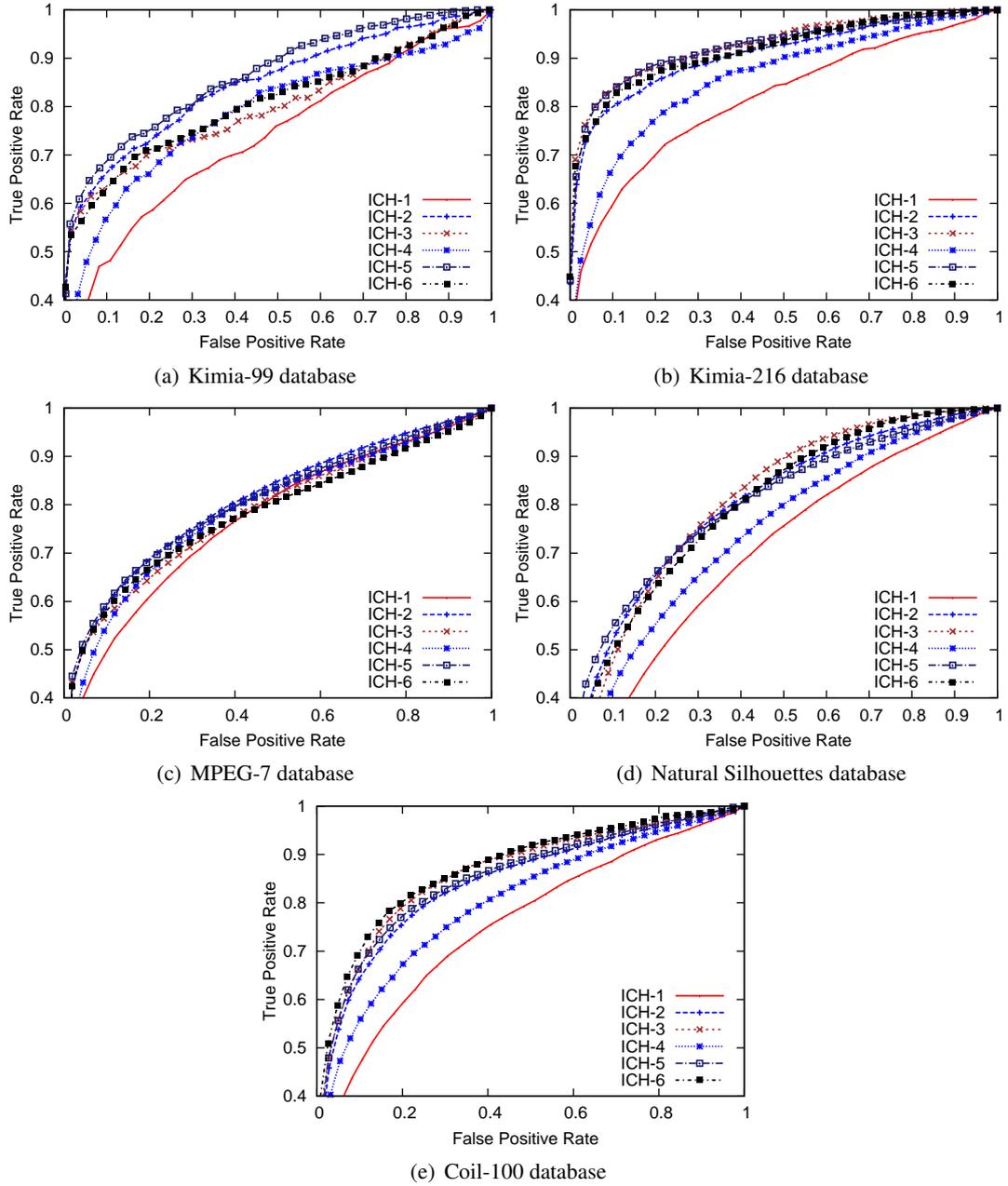


Figure 4.6: ROC curves for Intersection Consistency Histogram. For the sake of visibility, only  $tpr > 0.4$  is shown without loss of any information.

### 4.3.2 Comparison of Retrieval Performances

In this section, retrieval performances of the methods proposed in this study, namely ICH and GBP, are analyzed in comparison with the selected descriptors: Shape Context, Histograms of Oriented Gradients, Local Binary Patterns and Fourier Descriptors. Results are provided using database-specific score metrics and ROC curves as introduced in Section 4.2. In addition,

Table 4.12: Scores of GBP parameter settings (Table 1 of 2).

(a) Kimia-99 database TopRank scores											(b) Kimia-216 database TopRank scores										
Method	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	Method	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
GBP-21	95	85	84	79	76	73	68	66	57	56	GBP-19	205	192	183	179	179	163	163	163	152	140
GBP-3	95	85	82	79	77	73	68	66	59	53	GBP-22	203	197	182	180	165	165	169	153	146	147
GBP-24	96	84	84	77	75	73	69	63	66	49	GBP-21	203	190	185	176	178	164	152	150	149	134
GBP-12	96	86	85	78	74	67	74	61	61	53	GBP-3	204	190	183	175	175	162	158	150	145	138
GBP-20	95	86	82	79	75	78	70	69	55	46	GBP-24	204	190	179	174	173	162	157	155	148	137
GBP-15	96	87	86	83	73	74	63	65	55	49	GBP-20	205	192	182	179	168	161	159	152	140	140
GBP-6	95	88	83	79	75	75	68	62	52	52	GBP-12	200	193	181	177	168	161	153	150	152	141
GBP-18	96	84	84	77	74	66	71	61	60	50	GBP-6	200	190	186	168	169	165	156	150	143	139
GBP-9	96	85	88	75	73	70	67	63	55	49	GBP-15	200	192	179	176	166	161	158	157	146	129
GBP-19	94	86	89	76	75	71	62	59	55	50	GBP-23	203	192	179	173	164	157	157	150	140	145
GBP-23	94	83	85	78	75	72	66	63	59	42	GBP-2	199	189	177	171	165	161	155	153	140	137
GBP-2	95	82	80	78	75	73	64	67	59	39	GBP-11	195	189	178	174	158	154	152	147	145	146
GBP-8	93	89	84	73	68	66	67	60	55	46	GBP-1	196	189	177	166	163	158	155	153	142	138
GBP-11	95	82	83	74	70	69	69	62	54	39	GBP-10	194	185	178	162	158	155	154	151	146	135
GBP-22	95	86	85	73	69	65	59	60	49	44	GBP-5	188	173	168	154	158	142	144	139	133	127
GBP-17	95	82	83	80	62	65	66	57	47	34	GBP-14	189	178	163	156	143	146	139	134	131	129
GBP-1	95	81	82	80	71	66	57	51	44	38	GBP-4	178	168	152	149	143	137	131	128	127	99
GBP-10	94	83	83	70	66	56	54	50	55	36	GBP-13	180	169	158	147	141	130	125	125	124	106
GBP-5	91	80	76	72	67	58	53	54	53	37	GBP-8	184	169	154	142	142	134	124	126	110	98
GBP-14	93	81	70	71	66	60	51	46	49	36	GBP-18	184	166	157	138	142	136	130	124	111	93
GBP-7	86	79	71	71	57	56	57	48	43	37	GBP-17	182	163	145	144	135	131	128	122	117	96
GBP-16	88	80	71	69	61	51	55	48	37	24	GBP-9	182	163	157	145	141	130	122	117	105	90
GBP-4	82	67	63	52	44	38	43	38	32	25	GBP-7	180	164	141	154	130	129	124	121	107	94
GBP-13	82	65	65	49	41	37	43	27	34	21	GBP-16	176	162	153	141	134	131	124	112	102	89

retrieval results of each method for randomly selected query images are visualized in a table. Methods are compared using their best-performing parameter settings as determined in the previous chapter. In other words, from each method, the best parameter setting is selected for comparison with other methods (see Appendix B for the performance curves of each method).

Figure 4.7 includes performance curves for all databases. For the sake of visibility, only  $tpr > 0.4$  is shown. As the figure suggests, GBP, HOG and SC perform very close to each other followed by ICH, LBP and FD on all databases except the Coil-100 database. GBP outperforms SC and HOG in the Kimia-99 database and performs comparable to them in other databases. The performance of ICH follows the performances of GBP, HOG and SC. ICH outperforms LBP on most databases and FD on all databases. In fact, FD is the worst performing method in all databases. Performances of methods are almost the same on all

Table 4.13: Scores of GBP parameter settings (Table 2 of 2).

(a) MPEG-7 database		(b) Coil-100 database		(c) Natural Silhouettes database	
Method	Bull's Eye	Method	Accuracy	Method	Accuracy
GBP-19	59.38	GBP-20	90.60	GBP-20	93.27
GBP-22	57.81	GBP-21	90.40	GBP-23	92.86
GBP-20	56.55	GBP-23	90.20	GBP-3	92.65
GBP-1	56.38	GBP-2	88.80	GBP-21	92.45
GBP-23	55.95	GBP-3	88.80	GBP-19	92.24
GBP-24	55.75	GBP-11	88.60	GBP-24	92.24
GBP-12	55.54	GBP-24	88.40	GBP-22	91.63
GBP-21	55.48	GBP-19	88.00	GBP-12	91.22
GBP-2	55.33	GBP-5	86.60	GBP-2	90.61
GBP-3	55.26	GBP-12	86.20	GBP-11	90.00
GBP-11	54.90	GBP-22	86.20	GBP-1	89.59
GBP-10	54.65	GBP-6	84.80	GBP-15	89.39
GBP-15	53.73	GBP-15	84.40	GBP-6	88.16
GBP-6	52.92	GBP-14	84.20	GBP-10	87.55
GBP-14	49.91	GBP-1	82.40	GBP-14	83.88
GBP-5	49.12	GBP-18	82.20	GBP-5	82.04
GBP-18	48.42	GBP-9	80.60	GBP-17	78.98
GBP-9	48.15	GBP-10	80.40	GBP-18	78.98
GBP-7	45.97	GBP-17	76.80	GBP-8	78.78
GBP-8	45.75	GBP-8	75.20	GBP-13	78.57
GBP-16	45.21	GBP-13	74.00	GBP-16	77.76
GBP-17	45.17	GBP-16	73.20	GBP-4	77.55
GBP-4	41.02	GBP-4	71.00	GBP-7	76.73
GBP-13	40.08	GBP-7	69.40	GBP-9	76.73

databases except for the Coil-100 database. As described in Section 4.1, the Coil-100 database contains images of real world objects. This makes the Coil-100 database different from others, which contain silhouettes of artificial shapes. The low performance of GBP in this database indicates that it is not yet very successful at describing real world objects with texture. ICH, on the other hand, performs better in the Coil-100 database compared to other databases. It is the second best performing method on this database.

In Table 4.14, database-specific scores are listed. As introduced in Section 4.1, the MPEG-7 database and the Kimia databases use *Bull's Eye* and *TopRank* scores respectively. Tables 4.14(a) and 4.14(b) list *TopRank* scores respectively for the Kimia-99 and the Kimia-216 databases, and Table 4.14(c) includes *Bull's Eye* scores on the MPEG-7 database. For the

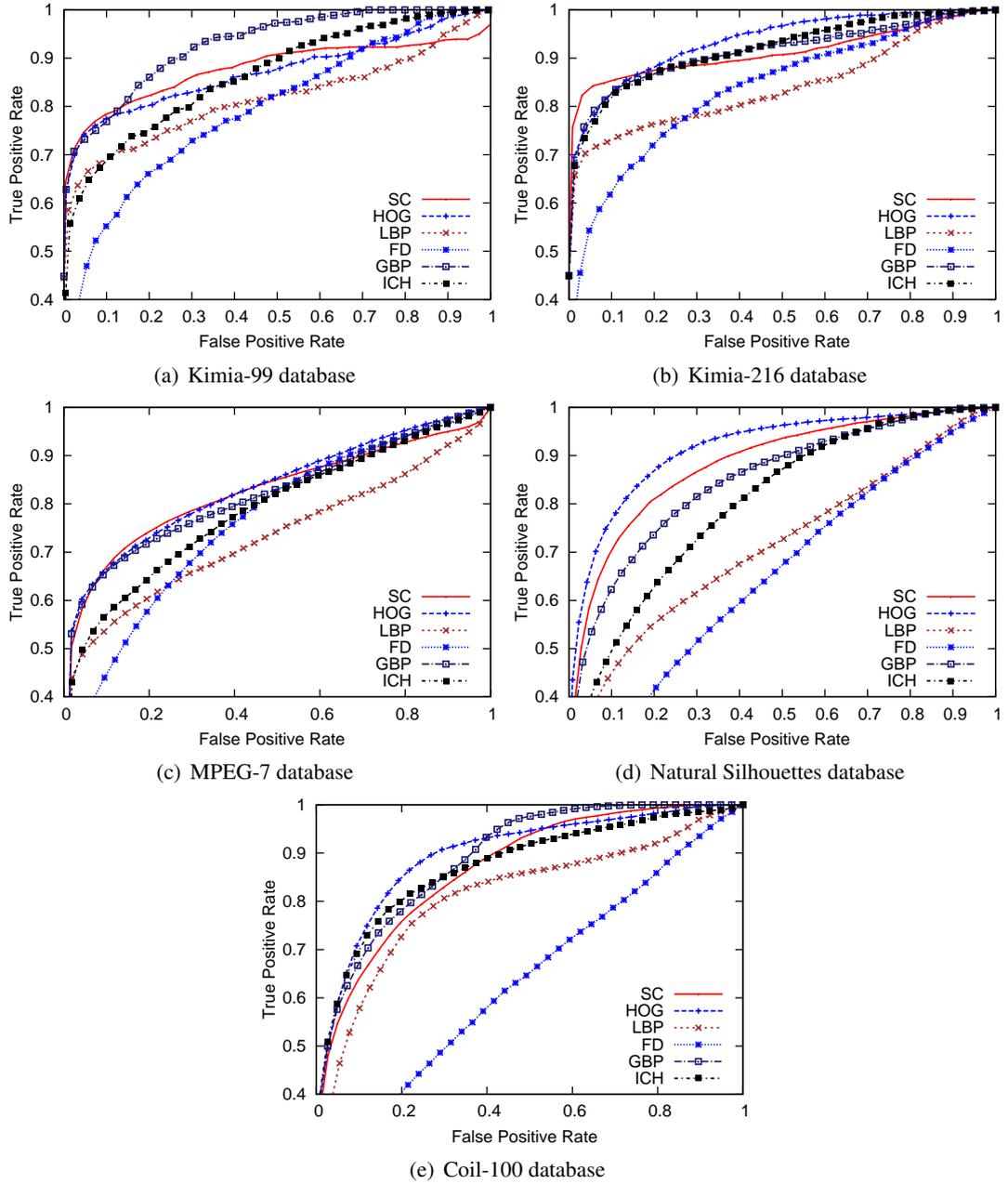


Figure 4.7: ROC curves for all methods with best parameter settings. For the sake of visibility, only  $tpr > 0.4$  is shown without loss of any information.

other two databases, namely the Natural Silhouettes database and the Coil-100 database, *accuracy* measures, as defined in Section 4.2, are listed. As seen from the tables, HOG performs better than other methods on all databases, except the MPEG-7 database. GBP is the second best-performing method after HOG, and it even outperforms HOG on the MPEG-7 database. GBP has the worst performance on the Coil-100 database. ICH follows GBP, HOG and SC closely, and it outperforms GBP on the Coil-100 database.

Table 4.14: Scores for all methods with best parameter settings.

(a) Kimia-99 database										
Method	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
HOG-6	96	88	89	87	80	70	69	64	52	45
GBP-21	95	85	84	79	76	73	68	66	57	56
SC-3	93	84	82	80	78	75	72	59	55	41
LBP-5	97	85	85	75	72	65	61	57	48	40
ICH-5	90	79	76	74	71	66	57	49	47	36
FD-5	72	61	59	54	41	37	35	35	34	27

(b) Kimia-216 database										
Method	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
HOG-2	209	205	201	195	194	188	180	172	155	143
SC-3	206	203	190	188	181	176	170	168	162	147
GBP-19	205	192	183	179	179	163	163	163	152	140
ICH-6	199	195	183	179	173	157	152	144	154	129
LBP-5	199	191	179	169	158	152	153	147	139	127
FD-5	156	139	128	125	107	103	87	86	82	69

(c) MPEG-7 database		(d) Natural Silhouettes database		(e) Coil-100 database	
Method	Bull's Eye	Method	Accuracy	Method	Accuracy
GBP-19	59.38	HOG-5	96.12	HOG-5	99.40
SC-3	58.97	GBP-20	93.27	SC-3	94.20
HOG-5	57.38	SC-3	93.06	LBP-5	94.00
ICH-3	50.90	ICH-6	91.43	ICH-6	92.20
LBP-5	50.38	LBP-3	86.94	GBP-20	90.60
FD-5	28.55	FD-1	57.96	FD-5	47.20

### 4.3.3 Comparison of Running times

Shape descriptors are expected to be fast enough to be useful in practice. The computational complexity of a shape descriptor becomes even more crucial for applications with real-time requirements such as *Content-based Image Retrieval* systems, which allow users to submit query images and fetch similar images as fast as possible. *CBIR* systems, therefore, need fast feature extraction and, especially, matching algorithms. For these reasons, running times of Global Binary Patterns and Intersection Consistency Histogram are analyzed in comparison with the selected descriptors.

Running time comparisons are investigated on an arbitrarily chosen database: Brown University Kimia-216 Shape Database. In order to judge the running time performance of each

technique fairly, each method is run on the same machine<sup>2</sup> using the same programming language (Matlab). Moreover, to make comparisons more accurate, experiment runs are divided into two phases, and real time measurements for both phases are reported separately. These phases, called *Extraction* and *Matching*, are described briefly as follows:

- **Extraction:** In this phase, descriptors are extracted from images that are pre-loaded into memory, i.e., reading and resizing images are performed before this step. After the successful completion of this step, the shape descriptors from all images would be extracted. Running times reported in this step are expected to reflect running time complexities of each method.
- **Matching:** After all descriptors are extracted, the matching phase follows. In this step, a simple matching algorithm is executed. This algorithm performs an *all-to-all* comparison between shape descriptors using a distance metric specific to the method. For instance, Shape Context uses an algorithm called “Hungarian algorithm” to compare descriptors [37]. Running times reported in this phase depend on the running time complexity of the distance algorithm and the length of the descriptor vector.

In Figure 4.8, running time measurements of both phases are visualized side-by-side for each method with parameter settings defined in Section 4.3.1. Among the six methods, the worst running time performance belongs to Shape Context. Durations of the *Matching* phases of *SC-1*, *SC-2* and *SC-3* are significantly higher than the durations of other methods. The reason for this low running time performance is attributed to the distance algorithm, i.e. “Hungarian algorithm”, which has a running time complexity of order  $O(n^3)$  [37]. The explanation as to why *SC-3* takes a little more time than *SC-1* and *SC-2* lies in the difference of the *Number of Sampling Points* parameter. As presented in Table 4.6, *SC-3* uses 100 sampling points whereas *SC-1* and *SC-2* use 50 sampling points. Note that the complexity of the “Hungarian algorithm” depends on the number of sampling points.

Fourier Descriptors and Histograms of Oriented Gradients are two methods whose running time measurements do not depend significantly on the choice of parameters. As suggested by Figure 4.8(b) and 4.8(e), variants of FD and HOG exhibit almost identical running time performances. On the other hand, Shape Context, Local Binary Patterns, Global Binary Patterns

---

<sup>2</sup> A machine with Intel Xeon E5430 Quad-Core CPU (2.66 GHz) and 16 GB RAM.

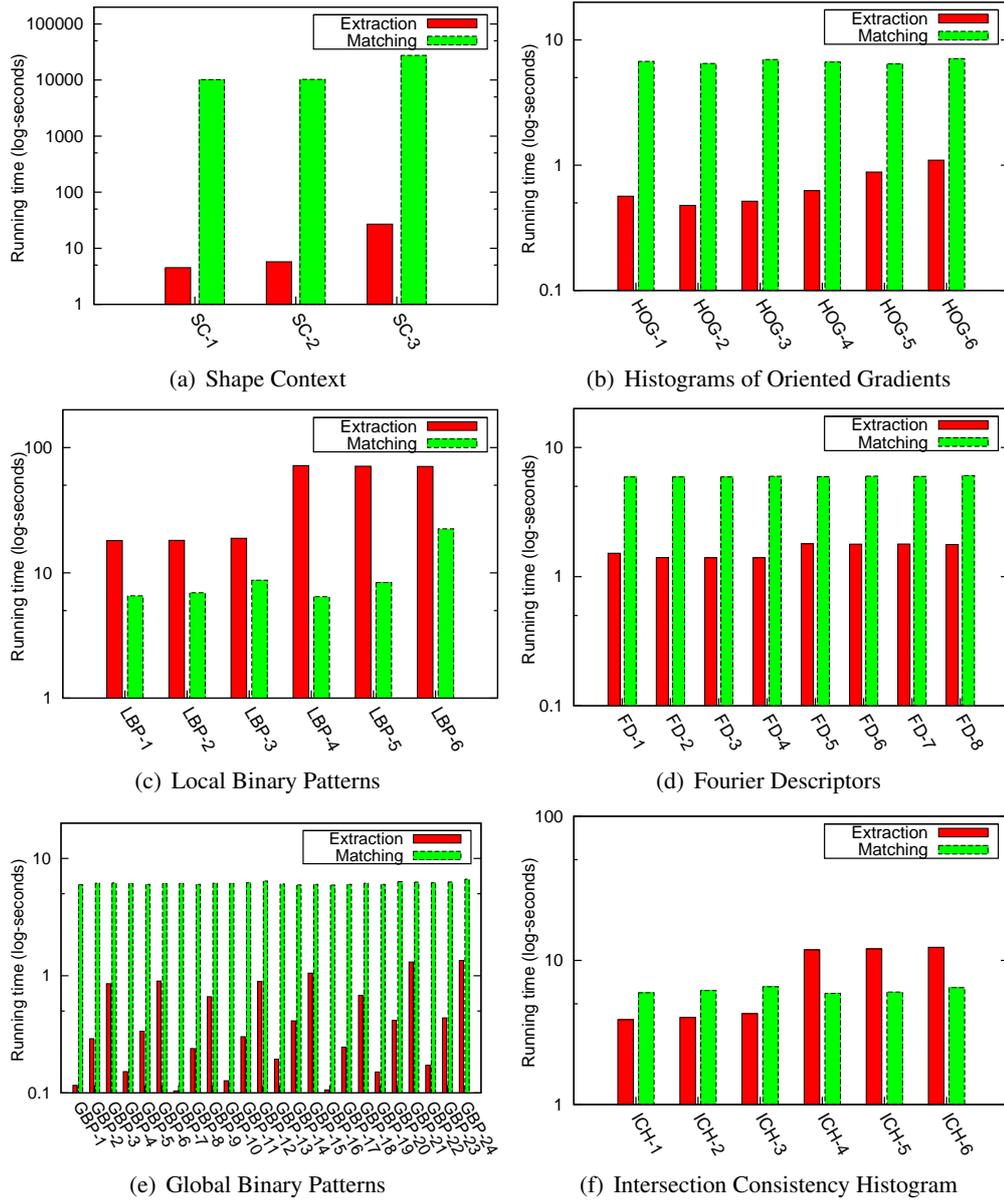


Figure 4.8: Running time measurements of all methods on the Kimia-216 database. Running times are reported in log-seconds.

and Intersection Consistency Histogram show varying extraction times depending on parameters. For instance, the *Extraction* phase durations of *LBP-4*, *LBP-5* and *LBP-6* are almost ten times higher than those of *LBP-1*, *LBP-2* and *LBP-3*. This difference is due to the dimensions of the images used in the runs; *LBP-1*, *LBP-2* and *LBP-3* use images with dimensions  $32 \times 32$  while *LBP-4*, *LBP-5* and *LBP-6* use images with dimensions  $64 \times 64$ . Similarly, the running time of Intersection Consistency Histogram depends on the dimension of the images. ICH runs faster on images with dimensions  $32 \times 32$  (*ICH-1*, *ICH-2* and *ICH-3*) compared to images with dimensions  $64 \times 64$  (*ICH-4*, *ICH-5* and *ICH-6*). The running time performance

of Global Binary Patterns depends strictly on the *Grid Size* parameter. For instance, the grid size of *GBP-3* is larger than *GBP-1*, causing *GBP-3* to run slower. Image dimension does not seem to affect the performance of GBP significantly.

Figure 4.9 depicts running time measurements of each method in a single graph for better comparison. Parameters for each method are selected according to their retrieval performances as presented in Section 4.3.1.

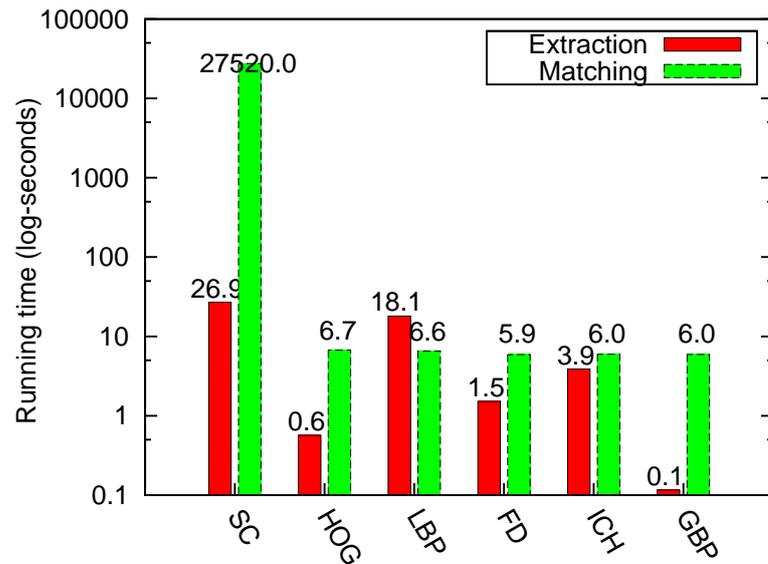


Figure 4.9: Running time measurements of all methods on the Kimia-216 database. Running times are reported in log-seconds. For each method, the best performing parameter setting is used.

Excluding Shape Context, one can safely state that the *Extraction* phase of LBP has the next worst performance. ICH follows LBP with approximately ten seconds of extraction duration. Among the six methods depicted in Figure 4.9, GBP has the best running time performance, followed by HOG. One final point worth mentioning is that durations of the *Matching* phase is roughly the same for each method except Shape Context. This is due to the computational complexity of the “Hungarian algorithm” as aforementioned.

## CHAPTER 5

### CONCLUSION

In this study, one of the most prominent problems in computer vision, namely shape representation and description, is studied using four popular shape descriptors as case studies, and two novel shape description methods are proposed that exhibit performance comparable to (and better than on some databases) widely-used algorithms in the literature. Proposed methods, named Intersection Consistency Histogram (ICH) and Global Binary Patterns (GBP), are evaluated on standard databases, and image retrieval and running time performances are compared to those of Shape Context, Histograms of Oriented Gradients, Fourier Descriptors and Local Binary Patterns.

Intersection Consistency Histogram is inspired from the well-known Histograms of Oriented Gradients method and Global Binary Patterns is a global version of the popular texture descriptor, Local Binary Patterns. The retrieval performance of ICH demonstrates its usage in shape representation problem as a *proof-of-concept*. Although the performance of ICH on shape databases does not reach the level of popular methods, it is open to improvements, which are discussed at the end of this chapter. In addition, the success of ICH on the Coil-100 database is an indication of its capability to describe real world objects with texture.

Global Binary Patterns shows a retrieval performance very close to the best performing method on most databases and it outperforms other methods on the MPEG-7 database. In addition, running time performance of GBP is the best one among the six methods analyzed in this study. Although, on the databases used in this thesis, there are methods other than Histograms of Oriented Gradients, Shape Context, Local Binary Patterns or Fourier Descriptors, with better performances, Global Binary Patterns manifests itself as a promising shape descriptor that is faster and simpler. For these reasons, GBP is well-suited to real-time applications or appli-

cations with quick development requirements.

## 5.1 On Shapes

As argued by Tari [74], shapes are continuous objects with respect to deformations, classifications and variations of input. Even if printed on digital media using discrete signals, shapes are continuous entities as already illustrated by Gestalt school of perception (see Figure 5.1).



Figure 5.1: Figure (a) contains a set of points that form a circle perception. Figure (b) illustrates a peanut shape as its neck gets thin or thick. The peanut transforms from a single blob to two discrete blobs as its neck thins. Images are taken from [74].

In addition, shapes coexist with other shapes in a continuous environment as opposed to being alone. Nearby objects affect the identity of the shape substantially. This requires a top-down approach to shape description, which is not investigated in this study. The descriptors proposed in this thesis assume isolated shape silhouettes as input.

A shape descriptor should not be sensitive to small changes in the shape. In other words, small variations in the shape should be reflected as small changes in the corresponding representation. ICH satisfies this property since a small orientation change in a pixel would yield a small deviation in the representation. GBP, on the other hand, is sensitive to small changes as a single pixel translation or noise may significantly change the decimal value of the binary sequence and thus the representation.

## 5.2 Future Work

The shape descriptors from the literature that are studied in this thesis are presented in their simplest forms. For instance, Fourier Descriptors and Local Binary Patterns have a number of modifications that are well-known in the literature. Shape Context has a very successful predecessor, named Inner-distance Shape Context, that can be analyzed. One possible future

work is to extend the scope of the analysis to include more recent methods. These studies could inspire new ideas for Global Binary Patterns and Intersection Consistency Histogram.

As mentioned above, methods proposed in this thesis perform comparable to, and in some cases better than, the widely-used methods in the field but there is still a lot of room for improvement. To start with, both algorithms use non-overlapping windows to extract features. However, HOG, an LBP variant [72] and many other algorithms use overlapping windows to better capture localized shape of an object. In a follow-up study, ICH and GBP can be extended to utilize overlapping grid cells.

Global Binary Patterns is not tolerant to changes in scale and translation. Thus, one possible improvement is to use more sophisticated distance metrics to make GBP robust to scale and translation changes. The version of GBP that uses the principal axis is expected to be invariant to rotation changes but this is not confirmed in this thesis and left out as a future work.

## REFERENCES

- [1] N. Alajlan, M. S. Kamel, and G. H. Freeman. Geometry-based image retrieval in binary image databases. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 30(6):1003–1013, 2008.
- [2] H. Asada and M. Brady. Smoothed local symmetries and their implementation. In *MIT AI Memo*, 1984.
- [3] Ç. Aslan and S. Tari. An axis-based representation for recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2005.
- [4] O. E. Badawy and M. S. Kamel. Hierarchical representation of 2-d shapes using convex polygons: a contour-based approach. *Pattern Recognition Letters*, 26:865, 2005.
- [5] X. Bai and L. J. Latecki. Path similarity skeleton graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7), 2008.
- [6] X. Bai, X. Yang, L. J. Latecki, W. Liu, and Z. Tu. Learning context-sensitive shape similarity by graph transduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.
- [7] S. Belongie, J. Malik, and J. Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *The Neural Information Processing Systems Foundation*, 2000.
- [8] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.
- [9] S. Belongie, G. Mori, and J. Malik. Matching with shape contexts. In *Proceedings of the IEEE Workshop on Content-based access of Image and Video-Libraries*, 2000.
- [10] H. Blum. Biological shape and visual science. *Journal of Theoretical Biology*, 38(2):205–287, 1973.
- [11] M. Bober. Mpeg-7 visual shape descriptors. *IEEE Transactions on Circuits and Systems*, 1(6):716–719, 2001.
- [12] J. Bowie and I. Young. An analysis technique for biological shape - ii. *Acta Cytologica*, 21(3):455–464, 1977.
- [13] J. F. Boyce and W. J. Hossack. Moment invariants for pattern recognition. *Pattern Recognition Letters*, 1(5-6):451–456, 1983.
- [14] J. W. Bruce, P. J. Giblin, and C. G. Gibson. Symmetry sets. *Proceedings of the Royal Society of Edinburgh*, 1985.

- [15] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [16] M. R. Daliri and V. Torre. Robust symbolic representation for shape recognition and retrieval. *Pattern Recognition*, 41(5):1799–1815, 2008.
- [17] E. R. Davies. *Machine Vision: Theory, Algorithms, Practicalities*. Morgan Kaufmann Publishers Inc., 2004.
- [18] A. Del Bimbo and P. Pala. Visual image retrieval by elastic matching of user sketches. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 19(2):121–132, Feb. 1997.
- [19] V. P. Dinesh Kumar and T. Tessamma. Performance study of an improved legendre moment descriptor as region-based shape descriptor. *Pattern Recognition and Image Analysis*, 18(1):23–29, 2008.
- [20] P. F. Felzenszwalb. Hierarchical matching of deformable shapes. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.
- [21] D. Geiger, T. Liu, and R. V. Kohn. Representation and self-similarity of shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):1129–1135, 2003.
- [22] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., 2nd edition, 2001.
- [23] L. Gorelick, M. Galun, E. Sharon, R. Basri, and A. Brandt. Shape representation and classification using the poisson equation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.
- [24] Ardeshir Goshtasby. Description and discrimination of planar shapes using shape matrices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(6):738–743, 1985.
- [25] G. H. Granlund. Fourier preprocessing for hand print character recognition. *IEEE Transactions on Computers*, 21(2):195–201, 1972.
- [26] J. Hadamard. *Lectures on Cauchy's Problem in Linear Partial Differential Equations*. Yale University Press, 1923.
- [27] R. W. Hamming. *Coding and information theory (2nd ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1986.
- [28] M. K. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, IT-8:179–187, 1962.
- [29] A. K. Jain, Y. Zhong, and S. Lakshmanan. Object matching using deformable templates. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 18(3):267–278, 1996.
- [30] O. Javed and M. Shah. Tracking and object classification for automated surveillance. In *Proceedings of the 7th European Conference on Computer Vision-Part IV, ECCV '02*, pages 343–357, 2002.

- [31] Y. Jeong and R. J. Radke. Reslicing axially sampled 3d shapes using elliptic fourier descriptors. *Medical Image Analysis*, 11(2):197–206, 2007.
- [32] S. Kalkan, Y. Shi, F. Pilz, and N. Krüger. Improving junction detection by semantic interpretation. *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2007.
- [33] H. M. Kim, J. H. Ryu, Y. G. Lee, and K. H. Lee. A new contour data compression method using fourier descriptor. *The International Journal of Advanced Manufacturing Technology*, 28:714–720, 2006.
- [34] W. Y. Kim, Y. S. Kim, and Y. S. Kim. A new region-based shape descriptor. Technical report, Hanyang University and Konan Technology, 1999.
- [35] B Kimia, A. R. Tannenbaum, and S. W. Zucker. Shapes, shocks, and deformations i: The components of two-dimensional shape and the reaction-diffusion space. *International Journal of Computer Vision*, 15:189–224, 1994.
- [36] T. Kozakaya, T. Shibata, M. Yuasa, and O. Yamaguchi. Facial feature localization using weighted vector concentration approach. *Image and Vision Computing*, 28(5):772–780, 2010.
- [37] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:83–97, 1955.
- [38] L. J. Latecki, R. Lakämper, and U. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2000.
- [39] M. Leyton. A process-grammar for shape. *Artificial Intelligence*, 34:213–247, 1988.
- [40] H. Ling and D. W. Jacobs. Shape classification using the inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(2):286–299, 2007.
- [41] H. Ling and D. W. Jacobs. Shape classification using the inner-distance. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29:286–299, 2007.
- [42] Y. Liu, M. Chen, H. Ishikawa, G. Wollstein, J. S. Schuman, and J. M. Rehg. Automated macular pathology diagnosis in retinal oct images using multi-scale spatial pyramid and local binary patterns in texture and shape encoding. *Medical Image Analysis*, 15(5):748–759, 2011.
- [43] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31:983–1001, 1998.
- [44] G. Lu and A. Sajjanhar. Region-based shape representation and similarity measure suitable for content-based image retrieval. *Multimedia Systems*, 7:165–174, 1999.
- [45] T. Mäenpää and M. Pietikäinen. Texture analysis with local binary patterns. In *Handbook Of Pattern Recognition And Computer Vision*. World Scientific, 2005.
- [46] G. Mcneill and S. Vijayakumar. Hierarchical procrustes matching for shape retrieval. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

- [47] H. K. Mebatsion, F. Boudon, C. Godin, C. Pradal, M. Génard, C. Goz-Bac, and N. Bertin. A novel profile based model for virtual representation of quasi-symmetric plant organs. *Computers and Electronics in Agriculture*, 75(1):113–124, 2011.
- [48] C. E. Metz. Basic principles of ROC analysis. *Seminars in nuclear medicine*, 8(4):283–298, 1978.
- [49] H. Mobahi, R. Collobert, and J. Weston. Deep learning from temporal coherence in video. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 737–744, 2009.
- [50] F. Mokhtarian, S. Abbasi, and J. Kittler. Efficient and robust retrieval by shape content through curvature scale space. In *Proceedings of the International Workshop on Image Databases and Multimedia Search*, pages 35–42, 1996.
- [51] F. Mokhtarian and A. K. Mackworth. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):789–805, 1992.
- [52] Farzin Mokhtarian, Sadegh Abbasi, and Josef Kittler. Robust and efficient shape indexing through curvature scale space. In *Proceedings of the British Machine Vision Conference*, pages 53–62, 1996.
- [53] S. Mori, H. Nishida, and H. Yamada. *Optical Character Recognition*. John Wiley & Sons, Inc., 1st edition, 1999.
- [54] S. Nene, S. Nayar, and H. Murase. Columbia object image library (coil-100). Technical report, Columbia University, 1996.
- [55] R.L. Ogniewicz and O. Kubler. Hierarchical voronoi skeletons. *Pattern Recognition*, 28(3):343–359, 1995.
- [56] T. Ojala and M. Pietikäinen. Unsupervised texture segmentation using feature distributions. In *Proceedings of the 9th International Conference on Image Analysis and Processing*, volume 1310, chapter 40, pages 311–318. Springer-Verlag, 1997.
- [57] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.
- [58] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.
- [59] S. K. Parui, S. E. Sarma, and D. D. Majumder. How to discriminate shapes using shape vector. *Pattern Recognition Letters*, 4(3):201–204, 1986.
- [60] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6):559–572, 1901.
- [61] E Persoon and K S Fu. Shape discrimination using fourier descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(3):388–397, 1986.

- [62] M. Pietikäinen, A. Hadid, G. Zhao, and T. Ahonen. *Computer Vision Using Local Binary Patterns*, volume 40. Springer-Verlag London Limited, 2011.
- [63] J. G. Proakis and D. K Manolakis. *Digital Signal Processing (4th Edition)*. Prentice Hall, 2006.
- [64] F. Provost, T. Fawcett, and R. Kohavi. The case against accuracy estimation for comparing induction algorithms. In *Proceedings of the Fifteenth International Conference on Machine Learning*, 1997.
- [65] Y. Rui, A. C. She, and T. S. Huang. Modified fourier descriptors for shape representation – a practical approach. In *Proceedings of First International Workshop on Image Databases and Multi-Media Search*, 1996.
- [66] T. B. Sebastian, P. N. Klein, and B. Kimia. Recognition of shapes by editing shock graphs. In *Proceedings of the IEEE International Conference on Computer Vision*, 2001.
- [67] T. B. Sebastian, P. N. Klein, and B. Kimia. Recognition of shapes by editing shock graphs. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2001.
- [68] R. N. Shepard. Toward a universal law of generalization for psychological science. *Science*, 237(4820):1317, 1987.
- [69] K. Siddiqi and B. Kimia. A shock grammar for recognition. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1996.
- [70] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock graphs and shape matching. In *Proceedings of the Sixth International Conference on Computer Vision (ICCV)*, 1998.
- [71] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Chapman & Hall, 2 edition, 1998.
- [72] V. Takala, T. Ahonen, and M. Pietikäinen. Block-based methods for image retrieval using local binary patterns. In *Proceedings of the 14th Scandinavian Conference on Image Analysis (SCIA)*, pages 882–891, 2005.
- [73] J. W. H. Tangelder and R. C. Veltkamp. A survey of content based 3d shape retrieval methods. In *Shape Modeling International*, 2004.
- [74] S. Tari. Fluctuating distance fields, parts, three-partite skeletons. In *Innovations for Shape Analysis: Models and Algorithms*. Springer, 2012.
- [75] Z. S. G. Tari, J. Shah, and H. Pien. Extraction of shape skeletons from grayscale images. *Computer Vision and Image Understanding*, 66(2):133–146, 1997.
- [76] M. R. Teague. Image analysis via the general theory of moments. *Journal of the Optical Society of America*, 70:920–930, 1980.
- [77] C. H. Teh and R. T. Chin. On image analysis by the methods of moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):496–513, 1988.
- [78] R. C. Veltkamp and M. Hagedoorn. State-of-the-art in shape matching. Technical report, Utrecht University, the Netherlands, 1999.

- [79] R. C. Veltkamp and M. Tanase. Content-based image retrieval systems: A survey. Technical report, Department of Computing Science, Utrecht University, 2002.
- [80] A. Yang, S. Maji, C. M. Christoudias, T. Darrell, J. Malik, and S. S. Sastry. Multiple-view object recognition in smart camera networks. In *Distributed Video Sensor Networks*, pages 55–68. Springer London, 2011.
- [81] X. Yang, S. Köknar-Tezel, and L. J. Latecki. Locally constrained diffusion process on locally densified distance spaces with applications to shape retrieval. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
- [82] T. Zaharia and F. Preteux. 3d shape-based retrieval within the mpeg-7 framework. In *Nonlinear Image Processing and Pattern Analysis*, volume 4304, pages 133–145, 2001.
- [83] D. Zhang and G. Lu. Shape retrieval using fourier descriptors. In *Proceedings of 2nd IEEE Pacific Rim Conference on Multimedia*, pages 1–9, 2001.
- [84] D. Zhang and G. Lu. A comparative study of fourier descriptors for shape representation and retrieval. In *Proceedings of the 5th Asian Conference on Computer Vision (ACCV)*, pages 646–651, 2002.
- [85] D. Zhang and G. Lu. Shape based image retrieval using generic fourier descriptors. In *Signal Processing: Image Communication 17*, pages 825–848, 2002.
- [86] D. Zhang and G. Lu. Review of shape representation and description techniques. *Pattern Recognition*, 37(1):1–19, 2004.
- [87] D. S. Zhang. *Image Retrieval Based on Shape*. PhD thesis, Monash University, Australia, 2002.
- [88] J. Zhang, Z. Xie, and S. Z. Li. Nearest prime simplicial complex for object recognition. *Clinical Orthopaedics and Related Research (CoRR)*, abs/1106.0987, 2011.
- [89] W. Zhang, S. Sclaroff, and S. Dunn. Shape-based indexing in a medical image database. In *Proceedings of IEEE Workshop on Biomedical Image Analysis*, pages 221–230, 1998.
- [90] G. Zhao and M. Pietikäinen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):915–928, 2007.
- [91] S. Çiftçi. A new edge detector using intersection consistency. Master’s thesis, Middle East Technical University, Ankara, Turkey, 2011.

## **APPENDIX A**

### **SAMPLE QUERIES**

Tables A.1, A.2, A.3, A.4 and A.5 show query results of all methods with best parameter settings for randomly selected images from the Kimia-99, Kimia-216, MPEG-7, Natural Silhouettes and Coil-100 databases respectively. For each query image, top 3 closest matches are listed with their corresponding costs.

Table A.1: Sample queries on the Kimia-99 database. For each method, the best performing parameter setting is used.

Query	Similar Images																	
	SC			HOG			LBP			FD		GBP		IGH				
																		
	14192	22432	22686	0.14	0.16	0.17	0.03	0.03	0.03	0.09	0.10	0.10	0.16	0.19	0.19	0.33	0.33	0.35
																		
	11508	12786	12938	0.12	0.13	0.15	0.02	0.02	0.02	0.06	0.20	0.22	0.10	0.11	0.15	0.19	0.23	0.26
																		
	10170	17434	19020	0.06	0.16	0.17	0.01	0.02	0.03	0.10	0.15	0.17	0.05	0.19	0.19	0.12	0.29	0.31
																		
	8562	20416	23008	0.05	0.16	0.16	0.01	0.02	0.02	0.02	0.10	0.13	0.05	0.14	0.18	0.18	0.24	0.33
																		
	16264	22328	23884	0.10	0.11	0.12	0.02	0.02	0.02	0.06	0.15	0.24	0.10	0.12	0.13	0.24	0.24	0.26
																		
	14862	18962	19300	0.01	0.12	0.12	0.00	0.02	0.02	0.08	0.14	0.16	0.02	0.11	0.11	0.04	0.22	0.24
																		
	14862	23346	25788	0.01	0.12	0.13	0.00	0.02	0.02	0.17	0.24	0.27	0.02	0.12	0.12	0.04	0.23	0.25
																		
	16842	17664	19112	0.10	0.11	0.11	0.01	0.02	0.02	0.13	0.16	0.22	0.08	0.10	0.10	0.22	0.23	0.24
																		
	12202	12296	17962	0.09	0.09	0.10	0.01	0.01	0.02	0.08	0.10	0.10	0.07	0.08	0.09	0.25	0.26	0.27
																		
	12284	18610	19570	0.09	0.13	0.14	0.02	0.02	0.02	0.14	0.15	0.15	0.11	0.17	0.18	0.27	0.31	0.33
																		
	10744	12492	12558	0.04	0.05	0.05	0.01	0.01	0.01	0.06	0.08	0.10	0.07	0.08	0.09	0.10	0.11	0.13
																		
	8284	9012	9350	0.04	0.04	0.04	0.01	0.01	0.01	0.06	0.10	0.11	0.07	0.07	0.08	0.11	0.12	0.13

Table A.2: Sample queries on the Kimia-216 database. For each method, the best performing parameter setting is used.

Query	Similar Images											
	SC	HOG	LBP	FD	GBP	ICM	ICM	ICM	ICM	ICM	ICM	ICM
6390	6726	7342	0.04	0.07	0.07	0.11	0.11	0.05	0.06	0.07	0.05	0.05
7164	8562	8642	0.06	0.07	0.13	0.03	0.03	0.08	0.11	0.06	0.07	0.09
22536	31598	40952	0.17	0.42	0.43	0.10	0.12	0.23	0.63	0.63	0.16	0.24
12184	13184	19250	0.08	0.09	0.16	0.03	0.08	0.12	0.14	0.25	0.31	0.13
14270	14890	15790	0.14	0.15	0.20	0.06	0.08	0.12	0.20	0.24	0.28	0.16
14890	16938	17320	0.14	0.16	0.20	0.06	0.07	0.08	0.28	0.28	0.29	0.17
20258	21894	28210	0.25	0.27	0.27	0.06	0.07	0.14	0.49	0.51	0.53	0.25
22128	22814	23134	0.27	0.27	0.28	0.07	0.17	0.20	0.26	0.27	0.28	0.10
4748	5784	6598	0.04	0.05	0.05	0.03	0.04	0.04	0.02	0.03	0.05	0.04
7640	10136	11200	0.02	0.16	0.17	0.00	0.11	0.14	0.02	0.22	0.28	0.06
11838	12172	12664	0.13	0.15	0.16	0.11	0.11	0.11	0.26	0.27	0.27	0.15
5546	7010	9112	0.18	0.19	0.21	0.05	0.05	0.08	0.24	0.26	0.31	0.17
5546	7010	9112	0.18	0.19	0.21	0.05	0.05	0.08	0.24	0.26	0.31	0.17
5546	7010	9112	0.18	0.19	0.21	0.05	0.05	0.08	0.24	0.26	0.31	0.17

Table A.3: Sample queries on the MPEG-7 database. For each method, the best performing parameter setting is used.

Query	Similar Images																	
	SC	HOG	LBP	FD	GBP	IGH	SC	HOG	LBP	FD	GBP	IGH						
	7484	12736	15248	0.17	0.19	0.21	0.03	0.03	0.03	0.09	0.10	0.10	0.30	0.34	0.39	0.13	0.13	0.14
	5778	12090	26212	0.42	0.43	0.44	0.03	0.04	0.04	0.05	0.07	0.10	0.55	0.60	0.15	0.18	0.18	0.18
	8058	11408	11942	0.16	0.17	0.19	0.04	0.05	0.05	0.11	0.13	0.13	0.30	0.31	0.38	0.08	0.08	0.11
	6900	7838	8264	0.17	0.18	0.18	0.01	0.02	0.02	0.10	0.13	0.13	0.20	0.25	0.27	0.12	0.12	0.12
	8128	13300	15634	0.17	0.18	0.19	0.06	0.06	0.06	0.17	0.17	0.18	0.62	0.63	0.63	0.16	0.16	0.16
	8096	8400	9286	0.03	0.03	0.03	0.00	0.00	0.01	0.00	0.02	0.05	0.08	0.09	0.13	0.03	0.03	0.04
	10972	12392	12420	0.01	0.02	0.02	0.00	0.00	0.00	0.04	0.07	0.09	0.07	0.07	0.08	0.02	0.02	0.02
	7436	15446	16492	0.23	0.23	0.24	0.05	0.05	0.05	0.08	0.08	0.09	0.51	0.51	0.51	0.16	0.18	0.18
	10870	19036	21134	0.25	0.29	0.30	0.03	0.03	0.03	0.13	0.15	0.16	0.32	0.33	0.37	0.08	0.10	0.12
	6782	7418	8992	0.13	0.15	0.15	0.03	0.03	0.03	0.07	0.09	0.11	0.09	0.09	0.09	0.06	0.07	0.07
	6312	6712	6802	0.13	0.13	0.16	0.03	0.03	0.03	0.08	0.09	0.09	0.08	0.10	0.11	0.07	0.07	0.07
	4998	8334	9178	0.08	0.10	0.11	0.01	0.01	0.01	0.04	0.08	0.09	0.09	0.10	0.14	0.07	0.07	0.07

Table A.4: Sample queries on the Natural Silhouettes database. For each method, the best performing parameter setting is used.

Query	Similar Images									
	SC	HOG	LBP	FD	GBP	IGH				
	7662	17514	21400	0.19	0.39	0.40				
	5608	6064	15016	0.02	0.21	0.27				
	20656	26034	26306	0.42	0.43	0.44				
	10894	11298	12996	0.25	0.26	0.27				
	17264	17916	18228	0.39	0.40	0.41				
	23364	26868	27506	0.41	0.41	0.41				
	21964	22194	22490	0.39	0.39	0.40				
	19160	24944	24990	0.35	0.39	0.48				
	9144	9774	10340	0.21	0.26	0.28				
	26022	28546	30456	0.28	0.37	0.40				
	10456	16504	21326	0.32	0.40	0.41				
	24192	26654	26752	0.42	0.42	0.46				

Table A.5: Sample queries on the Coil-100 database. For each method, the best performing parameter setting is used.

Query	Similar Images																		
	SC		HOG		LBP		FD		GBP		ICH								
	8812	9966	10316	0.25	0.27	0.32	0.13	0.13	0.14	0.06	0.08	0.08	0.09	0.10	0.15	0.07	0.07	0.08	
	8626	9276	12588	0.20	0.20	0.26	0.12	0.12	0.13	0.09	0.09	0.10	0.15	0.17	0.22	0.12	0.13	0.13	0.14
	14636	14890	16052	0.19	0.26	0.27	0.12	0.13	0.13	0.06	0.08	0.08	0.14	0.15	0.15	0.07	0.07	0.07	0.07
	9668	11266	15888	0.16	0.22	0.23	0.21	0.22	0.22	0.08	0.08	0.09	0.09	0.10	0.16	0.05	0.06	0.06	0.06
	10516	11150	13350	0.18	0.25	0.29	0.09	0.10	0.10	0.10	0.13	0.13	0.11	0.11	0.14	0.09	0.09	0.09	0.09
	8886	12874	17534	0.21	0.28	0.29	0.08	0.08	0.10	0.11	0.12	0.13	0.12	0.12	0.16	0.07	0.08	0.08	0.09
	23502	24458	28594	0.28	0.29	0.29	0.16	0.16	0.16	0.08	0.09	0.10	0.13	0.16	0.16	0.13	0.14	0.14	0.15
	5608	7246	8056	0.16	0.20	0.21	0.11	0.11	0.11	0.02	0.04	0.05	0.05	0.06	0.10	0.05	0.05	0.05	0.05
	17880	21904	28486	0.21	0.27	0.27	0.11	0.11	0.11	0.04	0.06	0.07	0.13	0.18	0.20	0.07	0.09	0.09	0.10
	10382	12274	12830	0.22	0.25	0.28	0.12	0.13	0.13	0.17	0.20	0.22	0.08	0.10	0.12	0.06	0.07	0.07	0.07
	6884	11172	13290	0.19	0.24	0.24	0.10	0.11	0.11	0.12	0.13	0.13	0.12	0.13	0.15	0.08	0.09	0.09	0.09
	21462	22160	22174	0.21	0.24	0.28	0.08	0.08	0.08	0.03	0.05	0.07	0.08	0.08	0.10	0.07	0.07	0.07	0.07

## APPENDIX B

### PARAMETER INVESTIGATION FOR THE SELECTED DESCRIPTORS

#### B.1 Parameter Investigation of Shape Context

Figure B.1 includes ROC curves for Shape Context on all databases. As the figure suggests, *SC-1*, *SC-2* and *SC-3* perform very close to each other. This result indicates that the dimensions of the image and the number of sampling points does not significantly affect the performance of Shape Context as long as they are large enough. In other words, these results do not demonstrate the performance of Shape Context using very few number of sampling points. A more comprehensive analysis is required to determine a lower bound for the number of sampling points.

#### B.2 Parameter Investigation of Histograms of Oriented Gradients

Results of Histograms of Oriented Gradients are presented in Figure B.2. On the Kimia-216, MPEG-7 and Natural Silhouettes databases, *HOG-3* and *HOG-6* perform worse compared to others. These two use smaller grids than others. However, the performance of *HOG-6* on the Coil-100 database suggests a reverse picture. The Coil-100 database is different than other databases since it contains images of real world objects while the others include silhouettes of artificial shapes. This means that the parameters of HOG needs to be adjusted according to the type of the database.

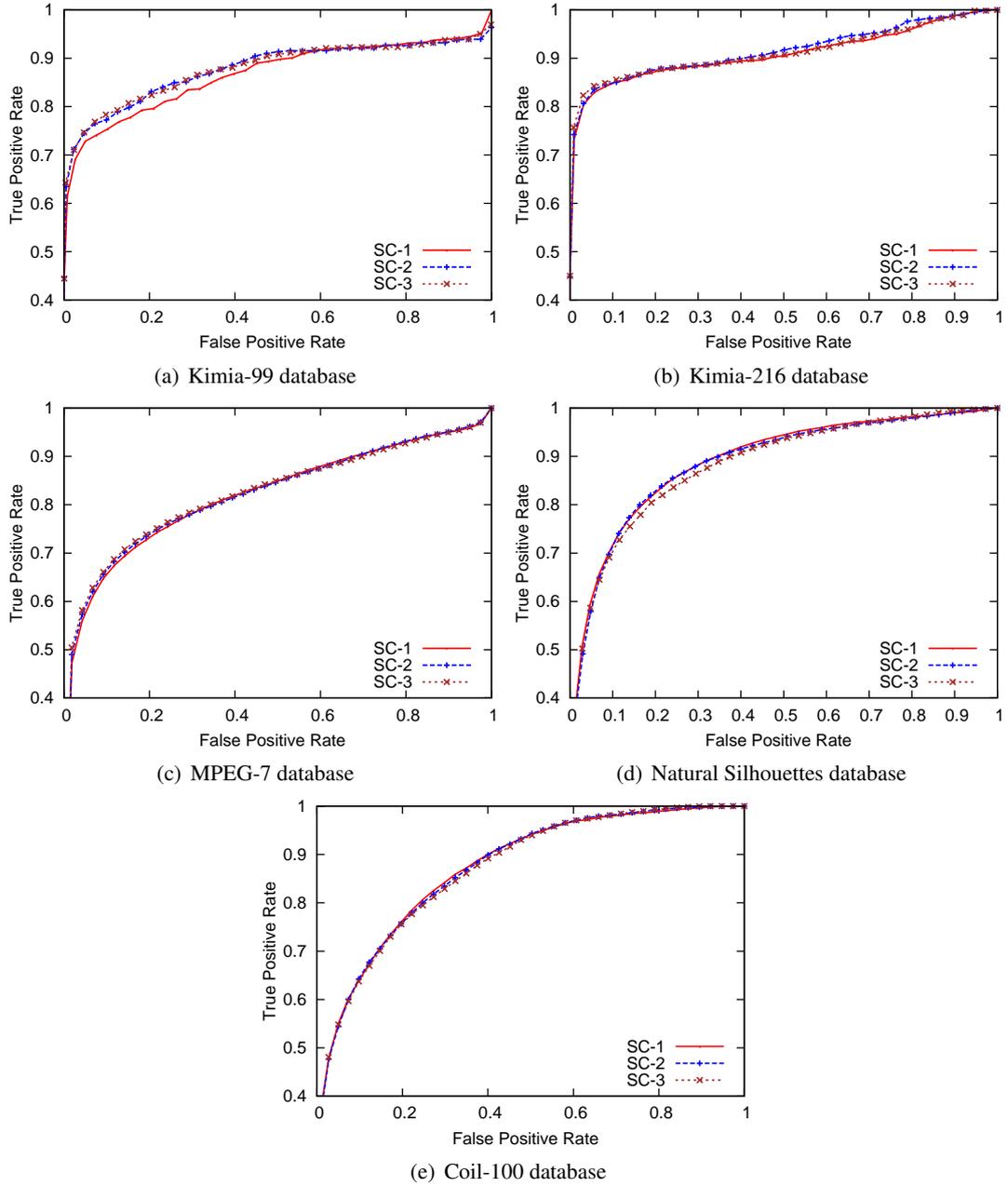


Figure B.1: ROC curves for Shape Context. For the sake of visibility, only  $tpr > 0.4$  is shown without loss of any information.

### B.3 Parameter Investigation of Local Binary Patterns

Local Binary Patterns are extracted by dividing the image into grids of sizes  $1 \times 1$ ,  $2 \times 2$  and  $4 \times 4$ . As seen from Figure B.3, variants of LBP with the same grid size perform very close to each other. For instance,  $LBP-1$  and  $LBP-4$ , both extracted from the whole image, demonstrate close performances. The effect of grid size to the performance is limited on all

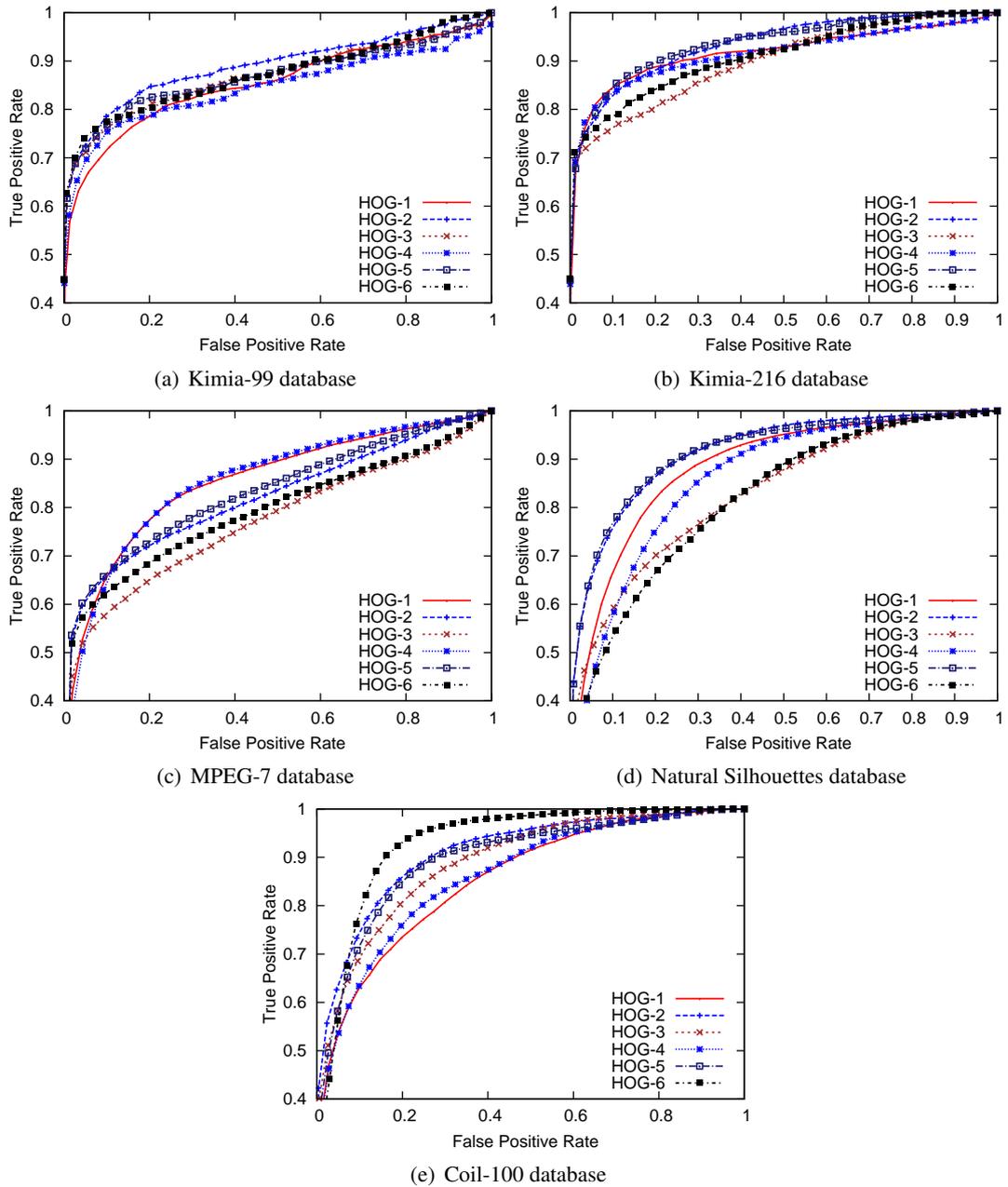


Figure B.2: ROC curves for Histograms of Oriented Gradients. For the sake of visibility, only  $tpr > 0.4$  is shown without loss of any information.

databases but judging the results on the Kimia databases and the MPEG-7 database, one can argue that applying LBP on the whole image is not a good idea. *LBP-1* and *LBP-4* use a  $1 \times 1$  grid and they perform worse compared to others.

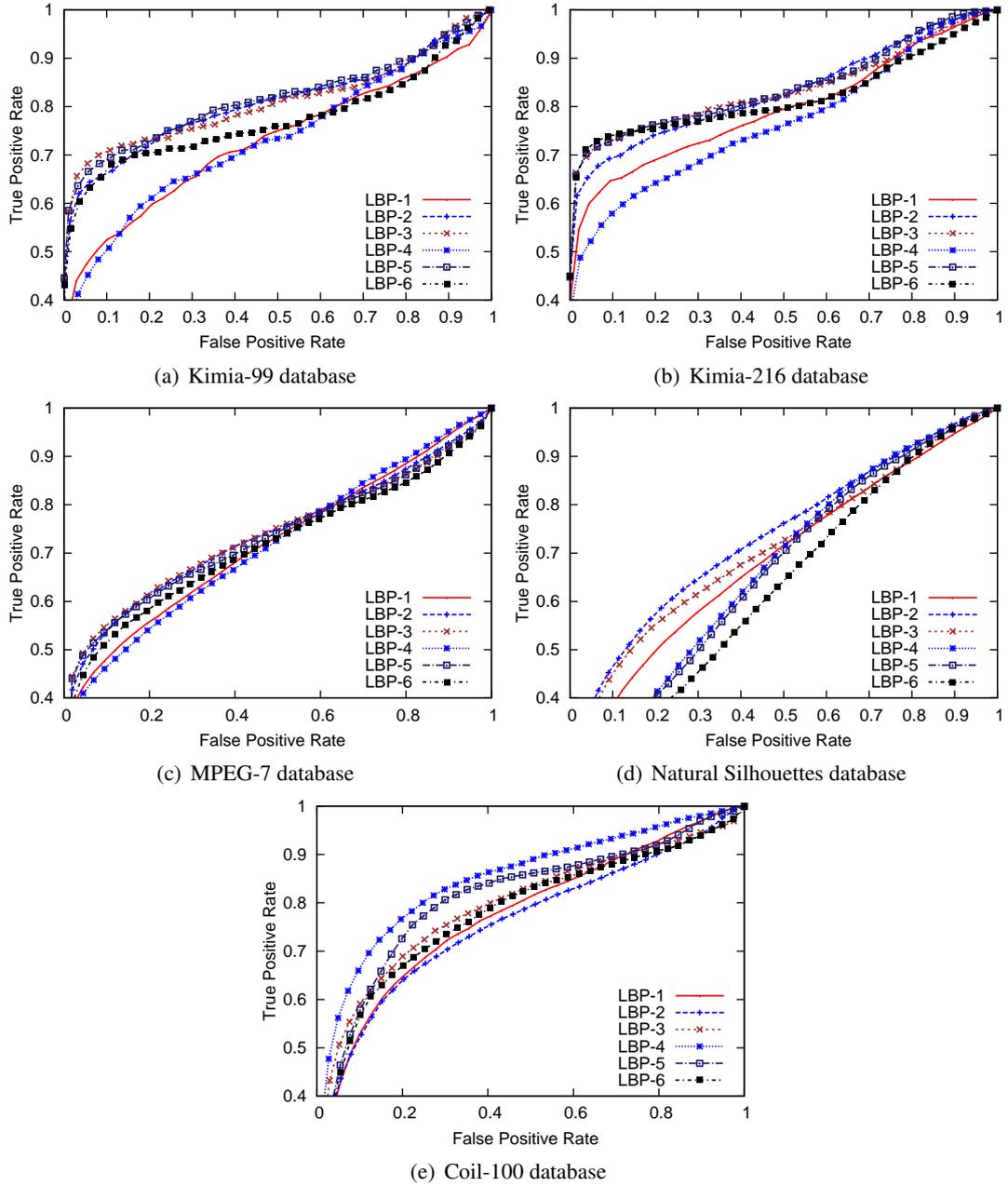


Figure B.3: ROC curves for Local Binary Patterns. For the sake of visibility, only  $tpr > 0.4$  is shown without loss of any information.

#### B.4 Parameter Investigation of Fourier Descriptors

Figure B.4 includes the performance curves of Fourier Descriptors. As seen from the figure, the number of Fourier terms does not need to be high to achieve a high performance.  $FD-1$  and  $FD-5$ , both use the first 5 Fourier terms, show good performances on the Kimia099 database. The effect of image dimension depends on the database since on some databases,

performances on  $32 \times 32$  resized images are better than those of  $64 \times 64$  images, and on some databases,  $64 \times 64$  resized images give better results.

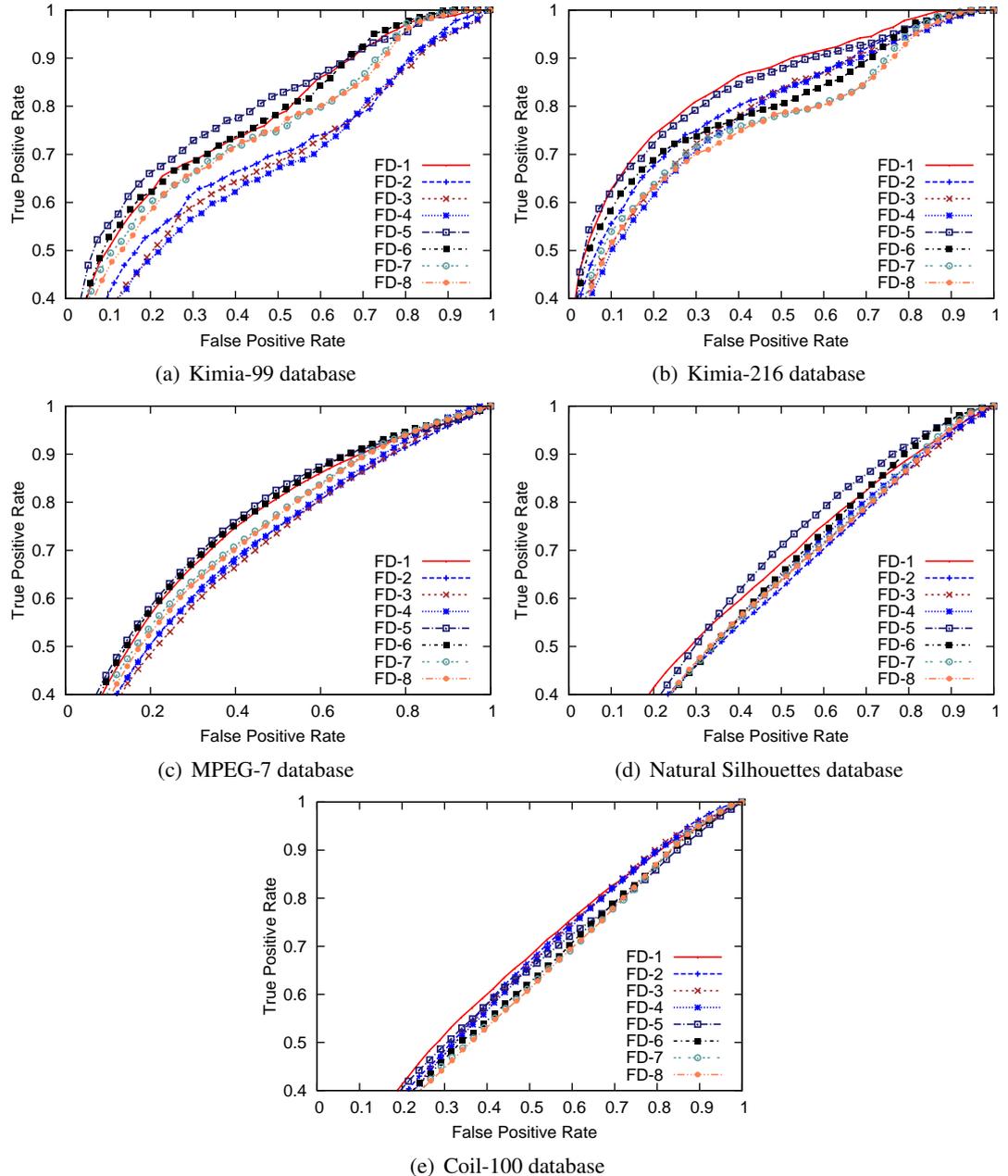


Figure B.4: ROC curves for Fourier Descriptors. For the sake of visibility, only  $tpr > 0.4$  is shown without loss of any information.