İSTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

OBJECT SEGMENTATION AND RECOGNITION USING GRADIENT BASED DESCRIPTORS AND SHAPE DRIVEN FAST MARCHING METHODS

Ph.D. Thesis by
Abdulkerim ÇAPAR

Department : Computer Engineering Department

Programme: Computer Engineering Programme

JUNE 2010

İSTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

OBJECT SEGMENTATION AND RECOGNITION USING GRADIENT BASED DESCRIPTORS AND SHAPE DRIVEN FAST MARCHING METHODS

Ph.D. Thesis by
Abdulkerim ÇAPAR
(504012092)

Date of submission :   22 January 2010
Date of defence examination: 04 June 2010

Supervisor (Chairman) :   **Prof. Dr. Muhittin GÖKMEN (ITU)**
Members of the Examining Committee :   **Prof. Dr. Ethem ALPAYDIN (BU)**
Prof. Dr. Bilge GÜNSEL (ITU)
Prof. Dr. Coşkun SÖNMEZ (YTU)
Assoc. Prof. Dr. Zehra ÇATALTEPE (ITU)

JUNE 2010

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

GRADYAN TEMELLİ BETİMLEYİCİLER VE ŞEKİL GÜDÜMLÜ HIZLI YÜRÜME TEKNİĞİYLE NESNE BÖLÜTLEME VE SINIFLANDIRMA

DOKTORA TEZİ
Abdulkerim ÇAPAR
(504012092)

Tezin Enstitüye Verildiği Tarih : 22 Ocak 2010
Tezin Savunulduğu Tarih : 04 Haziran 2010

Tez Danışmanı : Prof. Dr. Muhittin GÖKMEN (İTÜ)
Diğer Jüri Üyeleri : Prof. Dr. Ethem ALPAYDIN (BÜ)
Prof. Dr. Bilge GÜNSEL (İTÜ)
Prof. Dr. Coşkun SÖNMEZ (YTÜ)
Doç. Dr. Zehra ÇATALTEPE (İTÜ)

HAZİRAN 2010

FOREWORD

I would like to express my deep appreciation and thanks to my advisor Prof. Dr. Muhittin GÖKMEN for giving me valuable advice and support, always when needed. Another important person who worked with me on my thesis is Dr. Binnur KURT to whom I want to extend special thanks. This work has been partly supported by ITU Institute of Informatics and ITU Multimedia Center.

TABLE OF CONTENTS

ABBREVIATIONS

FM   **:** Fast Marching
ANN   **:** Artificial Neural Network
KNN   **:** K-Nearest Neighbours
HMM   **:** Hidden Markov Model
GBSD   **:** Gradient Based Shape Descriptor
BP   **:** Back Propagation
CSS   **:** Curvature Scale Space
FD   **:** Fourier Descriptors
WD   **:** Wavelet Descriptors
FT   **:** Fourier Transform
WT   **:** Wavelet Transform
CWT   **:** Continuous Wavelet Transform
DWT   **:** Discrete Wavelet Transform
DFT   **:** Discrete Fourier Transform
CBIR   **:** Content Based Image Retrieval
H-J   **:** Hamilton-Jacobi

x

LIST OF TABLES

LIST OF FIGURES

OBJECT SEGMENTATION AND RECOGNITION USING GRADIENT BASED DESCRIPTORS AND SHAPE DRIVEN FAST MARCHING METHODS

SUMMARY

In this thesis, a gradient based shape description and recognition methodology to use with active contour-based object segmentation systems has been proposed.

The Fast Marching (FM) active contour evolving model is utilized for boundary segmentation. A new speed functional has been defined to use first and second order image intensity derivatives. A local front stopping algorithm has also been proposed to improve the boundary handling performance of the FM model.

The most critical improvement of the thesis is defining a new shape descriptor called the Gradient Based Shape Descriptor (GBSD) [1]. GBSD is a new boundary-based shape descriptor that can operate on both binary and gray-scaled images. The recognition performance of GBSD is measured on a license plate character database, MPEG-7 Core Experiments shape data set and Kimia data Set. The success rates are compared with other well-known boundary-based shape descriptors and it is shown that GBSD achieves better recognition percentages.

A new recognition approach that utilizes the progressive active contours while iterating towards the real object boundaries has been proposed. This approach provides the recognizer many trials for shape description; it removes the limitation of traditional recognition systems that have only one chance for shape classification. Test results shown in this study prove that the voted decision result among these iterated contours outperforms the ordinary individual shape recognizers.

GRADYAN TEMELLİ BETİMLEYİCİLER VE ŞEKİL GÜDÜMLÜ HIZLI YÜRÜME TEKNİĞİYLE NESNE BÖLÜTLEME VE SINIFLANDIRMA

ÖZET

Bu çalışmada, aktif çevrit nesne bölütleyici yöntemlerle birlikte kullanılabilecek yeni bir şekil betimleme ve tanıma sistemi önerilmiştir. Önerilen sistem daha önce yapılan çalışmalar gibi aktif çevriti önceden tanımlı şekillerden birine zorlamak yerine, çevrit nesne sınırlarına yapışırken aynı zamanda şekil betimleme yapmayı amaçlamıştır.

Aktif çevrit bölütleyici olarak Hızlı Yürüme (Fast Marching) algoritması kullanılmış, Hızlı Yürüme metodu için yeni bir hız işlevi tanımlanmıştır. Ayrıca çevriti nesne sınırlarından geçtiği sırada durdurmayı amaçlayan özgün yaklaşımlar önerilmiştir.

Çalışmanın en önemli katkılarından birisi yeni ortaya atılan Gradyan Temelli Şekil Betimleyicisi (GTŞB) dir [1]. GTŞB, aktif çevrit bölütleyicilerin yapısına uygun, sınır tabanlı, hem ikili hem de gri-seviyeli görüntülerle rahatça kullanılabilecek başarılı bir şekil betimleyicidir. GTŞB nin araç plaka karakter veritabanı, MPEG-7 şekil veritabanı, Kimia şekil veritabanı gibi farklı şekil veritabanlarında elde ettiği başarılar diğer çok bilinen sınır tabanlı betimleyicilerle de karşılaştırılarak verilmiştir. Elde edilen sonuçlar GTŞB nin tüm veritabanlarında diğer yöntemlere göre daha başarılı olduğunu işaret etmektedir.

Çalışmada geliştirilen bir diğer önemli yaklaşım da Hızlı Yürüme çevritinin nesne sınırına yaklaşırken örneklenerek şeklin birden fazla defa betimlenmesine olanak veren yeni sınıflandırıcı yapıdır. Bu yaklaşım nesne tanımayı bir denemede sonuçlandıran geleneksel yöntemlerin bu sınırlamasını aşarak aynı nesneyi birçok kez tanıma olanağı sunmaktadır. Bu tanıma sonuçlarının tümleştirilmesiyle tek tanımaya göre daha yüksek başarılar elde edildiği çalışmanın ilgili bölümlerinde başarıları karşılaştıran tablolar yardımıyla gösterilmektedir.

1. INTRODUCTION

An object recognition system consists of three main stages: object *segmentation*, object *description* and object *classification*. Object segmentation aims for extraction of an object from its background. Segmentation includes object detection and localization steps. Invariant features against rotation, scaling, translation, etc. are obtained at the object description section. Another necessary property of the description section is compactness of output feature vectors in order to decrease the computational complexity of classification. The object classification part is designed to assign a class label to the object.

Given an arbitrary still image, the goal of object segmentation is to determine whether there are any pre-defined objects (faces, eyes, persons, cars, number plate characters, etc.) in the image and, if present, return the image location and extent of each object. Object segmentation techniques can be investigated in two parts: *Region based methods* and *contour based methods*. Color and texture are essential features for region-based image segmentation since these features are commonly observed in most images. Researchers have utilized uniform color spaces [3], filter banks [4, 5, 6], or machine learning [7, 8] to segment the objects with the help of color and texture information. Several attempts to combine color and texture have been made to enhance the basic performance of color or texture segmentation. These attempts, namely color–texture segmentation, include region growing approaches [9, 10, 11], watershed techniques [12, 13], edge-flow techniques [14], and stochastic model-based approaches [15, 16]. Region based image segmentation algorithms need a higher level post processing to handle the optimal object boundaries. Contour based image segmentation methods are investigated in Section 3.

In this study the Fast Marching (FM) method [17] which is an active contour segmentation technique, is utilized for object detection and segmentation. Active contours are techniques in vision used to detect objects in a given image using methods of curve evolution. FM is a special case of the Level Set method [18] that has one-way evolving fronts (see Section 3.2). In FM the passing time of an active

contour on any image location is calculated with the help of pre-calculated speed values over the scene. A new speed function using first and second order intensity derivatives has been proposed. In order to obtain the shapes properly, the evolving front is asked to be stopped near real object boundaries. Nevertheless, it is impossible for ordinary FM systems because of the non-zero speed functions. One of the contributions of the thesis is to provide a new FM contour stopping algorithm (see Section 5.3.2). The proposed algorithm uses first and second order derivatives of local image intensities to determine whether an evolving node should stop or not. A smoothing term is also added in to the front stopping criterion set.

The next stage of an object recognition system is shape description. Shape representation and description play an important role in many areas of computer vision and pattern recognition. Neuromorphometry, character recognition, contour-matching for medical imaging, 3-D reconstruction, industrial inspection and many other visual tasks can be achieved by shape recognition [19]. Zhang and Lu [31] classified the problem into two classes: contour-based methods and region-based methods. The classification is based on whether shape features are extracted from the contour only or are extracted from the whole shape region. Under each class, the different methods are further divided into structural approaches and global approaches. This sub-class is based on whether the shape is represented as a whole or represented by segments/sections (primitives). The whole hierarchy is shown in Figure 1.1.
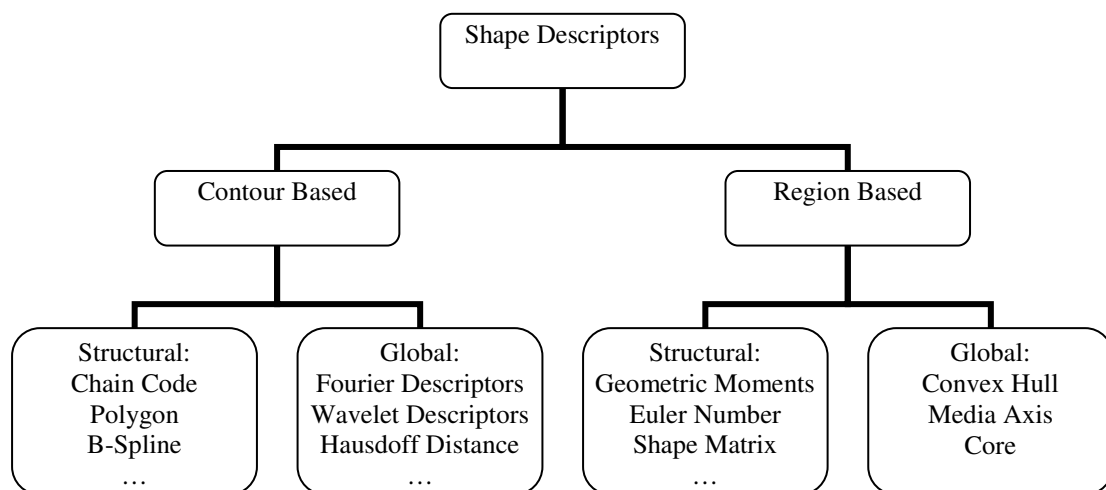
Figure 1.1 : Classification of shape description techniques [31].

As mentioned before, the proposed system is capable of both segmentation and identification of shapes simultaneously. Since an active contour-based segmentation approach for detecting objects has been utilized, a contour-based shape descriptor is needed. In this work, a contour-based shape description scheme, named Gradient Based Shape Descriptor (GBSD), using some rotated gradient filter responses along the object boundary has been proposed (see Section 4). Although the descriptors have been extracted by tracing an object boundary, local image gradient information has been utilized. The rotated gradient filter kernels, a type of steerable filters, are employed to obtain the local image gradient data. These filter responses along the shape boundary are treated as a one-dimensional shape signature. Fourier Descriptors (see Section 2.2.1) of this feature signature are computed to provide starting point invariance and to have compact feature set. There are several contour-based shape description techniques using Fourier Descriptors. Recently, a general evaluation and comparison on these FD methods has been published by Zhang and Lu [20]. Zhang and Lu studied different shape signatures and Fourier transform methods for the purpose of content based image retrieval (CBIR). They have studied different ways of acquiring FDs, retrieval effectiveness of different FDs and the compactness of FD. They came to the following important conclusions: on retrieval performance, centroid distance and area function signatures are the most suitable methods, and 10 FDs are sufficient for a generic shape retrieval system. The description performance of the proposed GBSD was compared with other well-known contour-based shape descriptors such as centroid distance, curvature and complex coordinates (see Section 4.4).

When the proposed shape descriptor GBSD is combined with Fast Marching (FM) approach, a descriptor vector for each FM evolving iteration is obtained. That means there is more than one feature vector for a single shape. In addition, each vector will be able to be fed into a classifier to obtain different decisions. Each decision result can be threaded as a different source of information and a decision fusion process can be applied to get final decision. This is another contribution of the thesis.

Decision fusion techniques can be divided into three categories: majority voting, weighted linear combination and classifier of classifiers. Among these techniques, majority voting is the simplest and most effective way to collaborate the classifier outputs.

The majority voting algorithm is employed as the decision fusion method. This algorithm creates the classification label histogram and chooses the label, which has maximum number bin, as final decision.

There are many studies on fusion of separate decision sources to get better object recognition results. However, obtaining separate decision result from the same decision source and applying decision fusion is a new approach which is presented in this thesis.

One of the challenges in the field of image segmentation is the incorporation of prior knowledge on the shape of the segmenting contour. Several methods of incorporating prior shape information into object location determination have been developed. In [21] a statistical model of shape variation is established from a set of corresponding points across the training images, and then, a Bayesian formulation based on this prior knowledge and edge information of the image is employed to find the object boundary. In [22] an elliptic Fourier decomposition of the boundary is utilized to incorporate the global shape information into the segmentation.

Integration of statistical shape variation into the level set methods was first proposed by Leventon et.al. [23]. They compute a statistical shape model over a training set of curves implicitly. The segmentation process embeds an initial curve as a level set of a higher dimensional surface and evolves the surface locally based on image gradients and curvature, and globally toward a maximum a posteriori (MAP) estimate of shape and pose. The MAP estimate is computed at each step of the surface evolution based on the prior shape and the image information. Since training shapes are embedded by the signed distance function dimension of the input space increases drastically and it is unclear in what way the surface representation affects the shape learning, since only the zero level set of the surface corresponds to a perceivable shape.

Chen et al. [24] used the same signed distance level sets for shape representation but they selected a variational method which minimizes an energy functional depending on the information of the image gradient and the shape of interest, instead of a probabilistic method as in [23].

Gastaud et al. [25] proposed a variational approach, based on a criterion featuring a shape prior allowing free-form deformation. The shape prior is defined as a functional of the distance between the active contour and a contour of reference.

Cremers et al. [26,27] present a variational integration of nonlinear shape statistics into a Mumford–Shah based segmentation process [28]. The nonlinear statistics are derived from a set of training silhouettes by a novel method of density estimation which can be considered as an extension of kernel PCA to a stochastic framework. They applied the proposed algorithm to find boundaries of specific objects, such as human hands and license plate characters. They presented good results for segmentation of plate characters with nonlinear shape prior statistics. But again only object boundary segmentation is the aim of the study and recognition issues were not a concerned of the authors.

Rousson and Paragios [29] developed an approach consisting of two stages. The first stage is for shape modelling, built directly on the level set space using a collection of samples. Then, this model is used as a basis to introduce the shape prior in an energetic form. This prior aims at minimizing the non-stationary distance between the evolving interface and the shape model in terms of their level set representations. The limitations are similar with [23], that is embedding the shapes into a signed distance level set map increasing the complexity and there is indefiniteness about the effects of this shape representation to the shape learning because the contour is defined only on the zero level sets.

Cremers et al. [59] recently published the first survey about integrating statistical information (color, shape, texture, motion, etc.) into the Level Set segmentation process. They have presented some specific class of region-based level set segmentation methods and clarified how they can all be derived from a common statistical framework.

Ayed et al. [60] represented a study investigates variational image segmentation with an original data term, referred to as "statistical overlap prior", which measures the conformity of overlap between the nonparametric distributions of image data within the segmentation regions to a learned statistical description. They claimed that it leads to image segmentation and distribution tracking algorithms that relax the assumption of minimal overlap and, as such, are more widely applicable than existing algorithms.

In this study, a novel object segmentation and description system has been proposed. It has the following advantages compared with other concurrent object segmentation-recognition approaches:

- In previous studies, the evolving front is always forced to have the prior shape. However, we stop the front near object boundaries

- It is stated that the proposed method in [29] does not work when the number of prior object classes is more than one. However, our system is capable of segmenting and recognizing different classes of characters.

- Previous researchers obtained the shape statistics from the whole map of level set values; however we employ only the front itself for shape description.

- Previously proposed systems need high calculation power because they have two optimization stages, one for minimization of image energies, and the other for minimizing shape similarity energies. On the other hand, our system has one optimization step for minimizing both energies.

- Recognition errors mostly occur because of segmentation problems. An object cannot be easily recognized if it cannot be properly extracted from the background. In this study, many segmentation results are employed as input of classifiers to reduce effects of the segmentation errors on recognition.

- In traditional recognition systems only one recognition chance exists for a single object but here many decision results can be obtained while the active contour is capturing the shape. In Section 5.4 it is shown that voting among these results raises the recognition performance as compared to single decision cases.

- In this study, there is a feedback mechanism between segmentation and description. This feedback provides better segmentation and recognition results.

The shape description models which depend on object boundaries are discussed in Section 2. Section 3 is for active contour based object segmentation approaches. Our new shape descriptor, called Gradient Based Shape Descriptor is introduced in Section 4. Integration of object segmentation and recognition methods are discussed in Section 5.

2. BOUNDARY-BASED SHAPE DESCRIPTION MODELS

Shape is one of the most important image features for classifying and recognizing objects. Human beings tend to perceive scenes as being composed of individual objects, which can be best identified by their shapes. Besides, as far as query is concerned, shape is simple for users to describe, either by giving example or by sketching. Shape representation and description play an important role in many areas of computer vision and pattern recognition. Content based image retrieval (CBIR), character recognition, medical imaging, 3-D reconstruction, industrial inspection and many other visual tasks can be achieved by shape features as well as other vision properties such as color, texture and motion [19].

There are two recent tutorials on shape description and matching techniques [30, 31]. Veltkamp and Hagedoorn [30] investigated the shape matching methods in four parts: global image transformations, global object methods, voting schemes and computational geometry. They also worked on shape dissimilarity measures. Another review on shape representation methods was accomplished by Zhang and Lu [31]. They classified the problem into two classes: contour-based methods and region-based methods, also referred to as external and internal techniques, respectively. Classification is based on whether the shape features are extracted only from the contour or are extracted from the whole shape region. Under each class, the different methods are further divided into structural approaches and global approaches. This sub-classification is based on whether the shape is represented as a whole or by segments/sections called primitives.

Region-based methods can be applied to more general applications than contour-based methods. However, they usually involve more computation and storage. Compared with region-based shape representation, contour-based shape methods are more popular in the literature. The reasons are in three aspects. First, it is generally recognized in the literature that shape can be described solely by its boundary features and humans are able to discriminate shapes by their contours or outlines. Second, most real world objects have clear contours, which are readily available. In

fact, contour-based shape methods can easily find applications and have produced satisfactory results in many situations. In this sense, applications of contour-based shape techniques are also quite general. Third, contour-based shape descriptors are usually more easy to derive. Contour-based methods represent shape as a 1D signal which is easier to analyze than a 2D signal. Contour-based shape methods include global shape descriptors, shape signatures, autoregressive models, structural methods, geometric invariants, spectral descriptors and curvature scale space (CSS) [32] methods.

This thesis is concerned with the global contour-based shape description methods. Two approaches most related to this study are shape signatures and shape matching.

2.1 Shape Signatures

A shape signature represents a shape by a function extracted from object boundary points. In general, a shape signature $u(t)$ is any 1-D function representing 2-D areas or boundaries. Signatures obtained along the object boundary are the focus of our interest. The boundary of any object Ω can be represented with an ordered sequence of points $\lambda_i = (x_i, y_i)$, $i = 0, 1, ..., N-1$ where N is the number of the points. These points are assumed to be extracted by a preprocessing module with an 8-connected contour tracing procedure. Many boundary based shape signatures are introduced in the literature.

2.1.1 Complex Coordinate Signature

Complex coordinate or position function is simply the complex number generated from the object boundary point coordinates

$$z_1(t) = \left[x(t) - x_c \right] + i \left[y(t) - y_c \right] \tag{2.1}$$

(x_c, y_c) is the centroid of the object which is defined as $x_c = \dfrac{1}{N} \sum_{t=0}^{N-1} x(t)$ and

$y_c = \dfrac{1}{N} \sum_{t=0}^{N-1} y(t)$, where N is the arc length of the boundary. $z_1(t)$ is a direct representation of the object boundary and is a translation invariant vector. The most important advantage of this signature is its computational simplicity.

2.1.2 Centroid Distance Signature

The centroid distance function is represented as the distance of the boundary points to the centroid of the shape

$$z_2(t) = sqrt\left(\left[x(t) - x_c\right]^2 + \left[y(t) - y_c\right]^2\right)$$

(2.2)
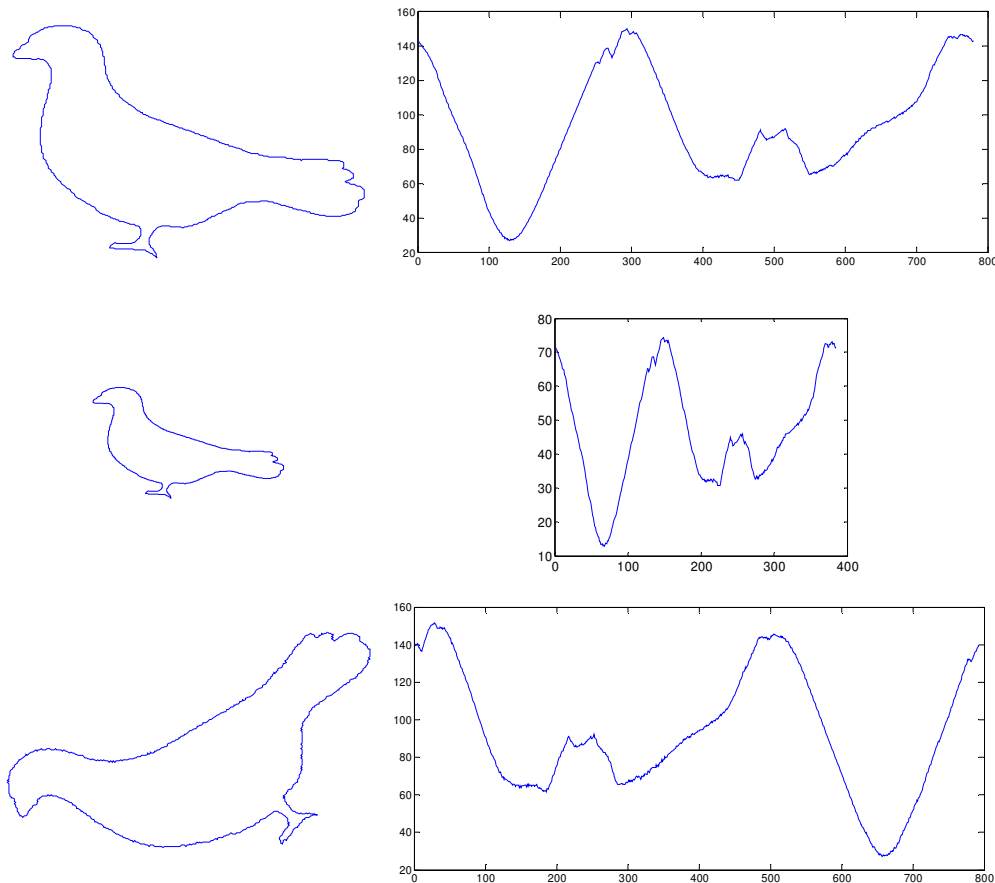
Figure 2.1 : The behavior of the centroid distance shape signature against scaling and rotation. First row, original shape and its signature; second row, shape scaled by 0.5 and its signature; third row, shape rotated ccw by 60^0, and its signature.

$z_2(t)$ is a translation invariant vector as a complex coordinate signature vector. It has shifting property against rotation, and scaling of the object changes the signature linearly. The behavior of the centroid distance signature against rotation and scaling is sketched in Figure 2.1.

2.1.3 Chord Length Signature

The chord length function $z_3(t)$ is derived on the boundary of the shape without using any reference point (i.e. object centroid). For each point p, $z_3(t)$ is the distance between p and another boundary point p' such that pp' is perpendicular to the tangent vector at p. This causes problems when pp' crosses more than two boundary points as in Figure 2. To solve this problem pp' is limited within the shape. In Figure 2.2, pp' also crosses p_1, but p_1 is eliminated since $p'p_1$ is not within the shape (dashed line). $z_3(t)$ overcomes the biased reference point (which means the centroid is often biased by boundary noise or defections) problems, however, the non-reference-point representation can cause problems when a shape is traced in different directions. In addition, it is very sensitive to noise; there may be a drastic burst in the signature of even a smoothed shape boundary. To reduce noise sensitivity, a post processing using an average filter may be used. $z_3(t)$ is invariant to translation. The computation to derive $z_3(t)$ is expensive.
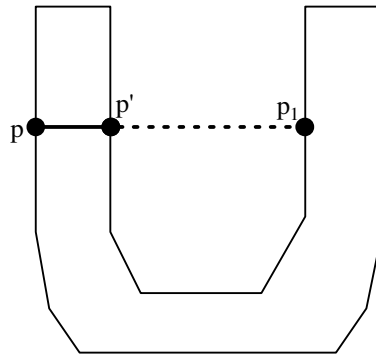
Figure 2.2 : Chord length signature on an U-type binary object.

2.1.4 Cumulative Angular Function Signature

Tangent angles of the shape boundary indicate the change of angular directions of the shape boundary. The change of angular directions is important to human perception. Therefore, shape can be represented by its boundary tangent angles as

$$\theta(t) = \arctan \frac{y(t) - y(t-w)}{x(t) - x(t-w)} \tag{2.3}$$

where w is any integer to indicate the jump step. The angle function $\theta(t)$ is defined in a range of length 2π, usually in the interval of $[-\pi, \pi]$ or $[0, 2\pi]$. Therefore $\theta(t)$ has discontinuities of size 2π. The cumulative angular function is introduced to overcome the discontinuities problem of angle function,

$$\varphi(t) = \big[\theta(t) - \theta(0)\big] \bmod(2\pi) \tag{2.4}$$

$\varphi(t)$ is the net amount of angular bend between the starting and the current position on the shape boundary. A normalized version of $\varphi(t)$ can be expressed as

$$\psi(t) = \varphi\left(\frac{Lt}{2\pi}\right) - t \tag{2.5}$$

where L is the shape perimeter. The subtraction of t from the cumulative angles makes $\psi(t) = 0$ for circle and $\psi(t) \neq 0$ for other shapes.

$\psi(t)$ is invariant under translation, rotation and scaling. The cumulative angular signature uniquely describes a shape. However, boundary noise can cause a much bigger change in the representation than the change in centroid distance; therefore, the structure of $\psi(t)$ is usually much more rugged than $z_2(t)$. Since the cumulative angular signature is derived from boundary tangents which are actually the first derivatives of the boundary coordinates, it usually contains discontinuities in the representation. As can be expected, its Fourier series converges rather slowly.

2.1.5 Curvature Signature

Curvature of a contour at a point is represented by the first and second derivatives of coordinate functions as

$$\kappa(t) = \frac{x'(t) y''(t) - y'(t) x''(t)}{\left(x'(t) + y'(t)\right)^{3/2}} \tag{2.6}$$

where $x'(t), y'(t)$ and $x''(t), y''(t)$ are the first and second derivatives of coordinate functions. In order to use the curvature signature for shape representation a

smoothing process should be applied. A Gaussian smoothing kernel has been utilized on coordinate functions as

$$x^s(t) = x(t) * G(\mu, \sigma), \quad y^s(t) = y(t) * G(\mu, \sigma) \tag{2.7}$$

where $x^s(t)$ and $y^s(t)$ are smoothed coordinate functions and $G(\mu, \sigma)$ is the Gaussian kernel.

2.1.6 Area Function Signature

When the boundary points change along the shape boundary, the area of the triangle formed by the two boundary points and the center of gravity also changes (Figure 2.3(a)). This forms an area function that can be exploited as shape representation.

For the triangle formed by O, P_1 and P_2 in Figure 2.3(b), its area (gray colored region) is given by

$$A(t) = \left| x_1(t) y_2(t) - x_2(t) y_1(t) \right| \tag{2.8}$$

The area function signature is similar but more rugged than the centroid distance signature. It is linear under affine transformation.

(a)   (b)
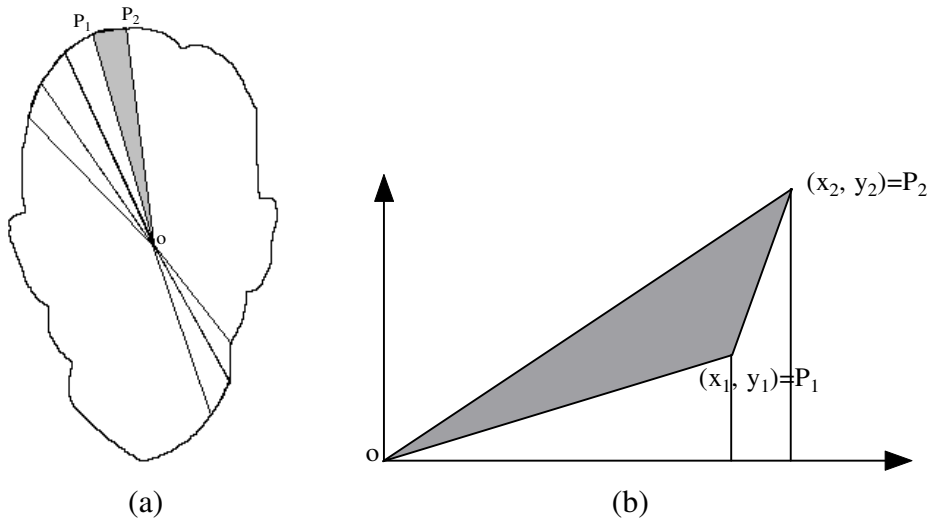
Figure 2.3 : Area function signature on a human face silhouette.

2.2 Shape Matching

The aim of shape matching is to find a similarity or dissimilarity measure between shapes. This measure is based on computing the distance between two shape signatures.

Direct matching of shape signatures in spatial domain for shape distance computing is not efficient for the following reasons:

- Lengths of the signature vectors are not constant even for the shapes in the same object class.

- Boundary-based shape signatures are very sensitive to noise and local shape deformations especially around object boundaries.

- Raw signature vectors are very long and complex for distance measurements. It is not suitable for CBIR systems with big datasets.

- Boundary based shape signatures are not invariant to the starting point and object rotation.

Spectral transformations such as the Fourier Transform and the Wavelet Transform are applied to the signatures to decrease the sensitivity to noise and local shape deformations, to reduce the feature vector dimension and to set up the invariance against rotation and starting point. At the end the Fourier Descriptors (FD) and the Wavelet Descriptors (WD) are extracted. Matching is done in transform space in this case.

2.2.1 Fourier Descriptors

Fourier Descriptor (FD) is one of the most widely used shape descriptors [32, 33, 34, 35] due to its advantages: (i) it is simple to compute, (ii) each descriptor has a specific physical meaning, (iii) it is simple to do normalization, making shape matching a simple task, (iv) it captures both global and local features, and (v) it has coarse to fine description capability.

Fourier Descriptors are simply obtained by applying Fourier Transform on a 1-D shape signature vector $u(t)$. The vector $u(t)$ is a periodic function since it is

obtained around a closed object boundary. $u(t) = u(t+nT)$ where T is the period. For any signature vector $u(t)$ its discrete Fourier transform is given by

$$a_n = \frac{1}{N} \sum_{t=0}^{N-1} u(t) \exp\left(\frac{-j2\pi nt}{N}\right) \quad n = 0,1,...,N-1 \tag{2.9}$$

A set of Fourier coefficients a_n is utilized for representation of the shape. A shape representation should be invariant to operations like translation, scaling and rotation. The selection of different starting points on the shape boundary should not affect the representation also.

From Fourier theory, the general form for the Fourier coefficients of a contour generated by translation, rotation, scaling and change of starting point from an original contour is given by [35]:

$$a_n = \exp(jn\tau) \times \exp(j\varphi) \times s \times a_n^{(o)} \quad n \neq 0 \tag{2.10}$$

where $a_n^{(o)}$ and a_n are the Fourier coefficients of the original and transformed shape. $\exp(jn\tau)$, $\exp(j\varphi)$ and s are the terms due to change of starting point, rotation and scaling respectively. Except for the DC component (a_0), all other coefficients are not affected by translation. Consider the following expression:

$$b_n = \frac{a_n}{a_0} = \frac{\exp(jn\tau) \times \exp(j\varphi) \times s \times a_n^{(o)}}{\exp(jn\tau) \times \exp(j\varphi) \times s \times a_0^{(o)}}$$

$$= \frac{a_n^{(o)}}{a_0^{(o)}} \exp\left[j(n-1)\tau\right] = b_n^{(o)} \exp\left[j(n-1)\tau\right] \tag{2.11}$$

where b_n and $b_n^{(o)}$ are the normalized Fourier coefficients of the derived and original shape, respectively. As seen in Eq. 11 b_n and $b_n^{(o)}$ have only difference of $\exp\left[j(n-1)\tau\right]$. When the phase information is ignored and only the magnitude of the coefficient is used, then $|b_n|$ and $|b_n^{(o)}|$ are the same. Therefore the normalized Fourier coefficient set $|b_n|$ is invariant to translation, rotation, scaling and the change of starting point. The reason for choosing a_0 as the normalization factor is that it is

the average energy of the signal. It is normally the largest coefficient, therefore, the normalized FD features are in $[0,1]$.

(a)

(b)

2 FDs 5 FDs 8 FDs 10 FDs 15 FDs

20 FDs 25 FDs 30 FDs 40 FDs 50 FDs
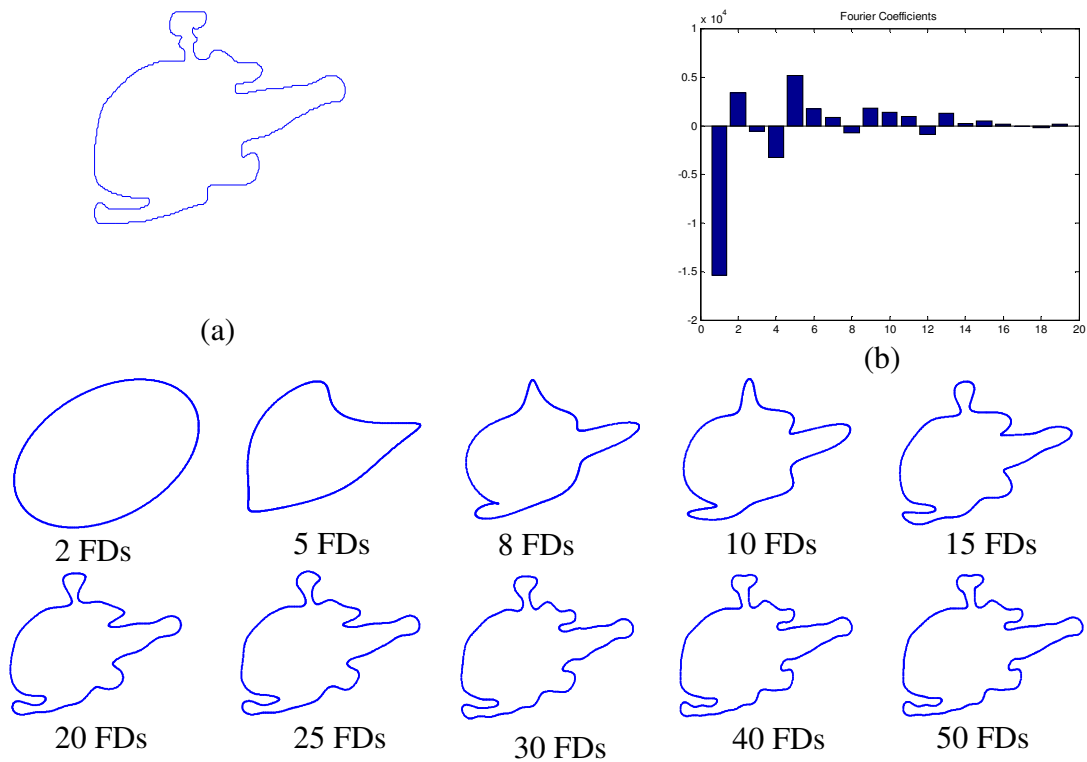
Figure 2.4 : Fourier description and shape reconstruction of a chopper image.

Figure 2.4 illustrates Fourier decomposition of a black-white chopper image (Figure 2.4 (a)). After the boundary of the object is traced its Fourier Descriptors are calculated over the complex coordinate shape signature (Figure 2.4 (b)). The second and third rows represent the shape reconstruction with different number of Fourier Descriptors. As seen in the figure, better reconstructions are obtained while descriptor number is increasing. Another point is that the reconstruction process saturates if the number of descriptors exceeds a certain number (30 for this example).

2.2.2 Wavelet Descriptors

Similar to the Fourier Transform (FT), the Wavelet Transform (WT) uses elementary functions, called wavelets, to describe a given signal. In contrast to the FT, which uses harmonic functions with different dilation, compression and shifting, the WT uses only one basis wavelet (mother wavelet) to derive the reconstruction signals [36]. When applying the wavelets to the pattern representations, a limited number of levels is chosen, representative of the task and their coefficients normalized to

15

provide the appropriate invariance to the translation, scale and rotation. These are called wavelet descriptors [37].

The Continuous Wavelet Transform (CWT) transforms a continuous, square-integrable function $f(t)$ into a function $W_\psi(s, \tau)$ of two continuous real variables, scale $s > 0$ and translation τ

$$W_\psi(s, \tau) = \int_{-\infty}^{\infty} f(t) W_{s, \tau}(t) \tag{2.12}$$

where the function $W_{s, \tau}(t)$ is known as mother (or basis) wavelet and is given by

$$W_{s, \tau}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t - \tau}{s}\right) \tag{2.13}$$

The mother wavelet used to generate all the basis functions is designed based on some desired characteristics associated with that function. The translation parameter τ relates to the location of the wavelet function as it is shifted through the signal. Thus, it corresponds to the time information in the Wavelet Transform. The scale parameter s is defined as $|1/ \text{frequency}|$ and corresponds to frequency information. Scaling either dilates (expands) or compresses a signal.

In CWT, the signals are analyzed using a set of basis functions which relate to each other by simple scaling and translation. In the case of Discrete Wavelet Transform (DWT), a time-scale representation of the digital signal is obtained using digital filtering techniques. The signal to be analyzed is passed through filters with different cutoff frequencies at different scales. The wavelet descriptors are formed on the basis of discrete wavelet representation of the original shape signature of the boundary of the shape.

Although Wavelet Descriptors has the advantage over Fourier Descriptors in that it is of multi-resolution in both spatial space and spectral space, the increase of spatial resolution will certainly sacrifice frequency resolution. Therefore, only wavelet coefficients of the few low frequencies are used to represent shape. Most importantly, the complicated matching scheme of wavelet representation makes it impractical for online shape retrieval.

3. OBJECT DETECTION WITH ACTIVE CONTOUR MODELS

Parametric and non-parametric active contour models have been widely used in shape modeling, object detection and object tracking. Since "snake" was first introduced by Kass et al. in 1987 [38], various forms of active contour models have been popular for segmentation of noisy and low contrast images. Snakes are planar parametric deformable contours that are useful in a variety of image analysis tasks. They are often used to approximate the location and shapes of object boundaries, based on the assumption that boundaries are piecewise continuous or smooth [39]. In snake models, a contour is deformed to reach and stop on the boundary of the target object. This deformation is proceed by minimizing the following energy functional E,

$$E(S) = \int_S \left[E_{\text{int}}\left(S(u)\right) + E_{img}\left(S(u)\right) + E_{ext}\left(S(u)\right) \right] du \qquad \textbf{(3.1)}$$

where $S(u)$ is a parametric representation of the contour embedded in the image plane. E_{int} represents the energy of the contour resulting from the internal forces that maintain a certain degree of smoothness and even control point spacing along the curvature of the contour. E_{img} is the energy content resulting from image forces. Image forces are responsible for driving the contour toward certain image features, such as edges, and are computed based on the image data. Finally, E_{ext} represents the energy resulting from the external forces, which may or may not be applied, from a high-level source such as a human operator or other high-level mechanisms to maintain certain characteristics of the contour. Allowing the snake to change its shape and position minimizes the total energy of the contour.

In the classical snakes and active contour models, an edge-detector is used, depending on the gradient of the image, to stop the evolving curve on the boundary of the desired object [40]. This method is non-intrinsic and need parametric representation using marker particles. This does not allow for accurate modeling in

the presence of corners, cusps and multiple objects [41]. The snake model cannot deal with multiple objects and snake contours have no ability to merge or split - which is another weakness caused by the nature of the model - and they are noise sensitive and very slow.

To represent a better solution for the above problems, "Level Set Methods" for capturing moving fronts was introduced in 1987 by Osher and Sethian [42].

3.1 Level Set Methods

Applications of "Level Set Methods" range from capturing multiphase fluid dynamic flows to graphics, e.g. special effects in Hollywood, to visualization, image processing, control, epitaxial growth, computer vision and include many other [43]. Level Sets, a class of geometric deformable models, is an effective shape modeler due to its capability of topology preservation and fast shape recovery. Unlike the Lagrangian (solid) formulation associated with the snake models, level set methods are characterized by Eulerian(fluid) formulations.

The original idea behind the level set method was a simple one. Given a front $C(t)$ in R^n, bounding an open region Ω, its subsequent motion under a velocity F is desired to analyze and compute. This velocity can depend on position, time, the geometry of the front, and the external physics. Osher and Sethian [42] used a smooth function $\phi(x,t)$ that represents the front as the level set, where $\phi(x,t) = 0$. It means the propagating interface (front) is represented as the zero level set of a higher dimensional distance function $\phi(x,t)$, which is defined as

$$\phi(x,t) = -d \quad x \in \Omega$$
$$\phi(x,t) = +d \quad x \notin \Omega$$
$$\phi(x,t) = 0 \quad x \in C(t)$$

(3. 2)

where d is the distance from x to $C(t)$ and the plus-minus sign is chosen if the point x is outside or inside of the evolving curve.

In level set methods the evolving front C is given by the zero level sets of the distance function ϕ [42]

$$C\left(t=t_i\right)=\left\{(x,y) \mid \phi\left(x,t=t_i\right)=0\right\}$$ **(3. 3)**

The evolving equation of the front can be obtained by

$$\phi\left(x(t),t\right)=0, \quad \phi_0=\phi\left(x(t),t=0\right)$$ **(3.4)**

$$\Rightarrow \phi_t+\nabla\left(\phi\left(x(t),t\right)\right)\dot{x}(t)=0$$ **(3.5)**

$$\Rightarrow \phi_t+\vec{F}\left|\nabla\phi\right|=0$$ **(3.6)**

where $\vec{F}$ is the desired speed on the front. Since, only the normal component of the speed is needed, Eq. 19 becomes

$$\frac{\partial\phi}{\partial t}+F_N\left|\nabla\phi\right|=0$$ **(3.7)**

Here $F_N=\vec{F}*\dfrac{\nabla\phi}{\left|\nabla\phi\right|}$ is the normal component of the speed function.

Figure 3.1 illustrates an expanding circle in the level set formulation. Let the initial front C at $t=0$ be a circle in xy-plane (Figure 3.1.a). It is imagined that the circle is the level set ($\phi=0$) of an initial surface $z=\phi\left(x,y,t=0\right)$ in R^3. Figures 3.1.c and 3.1.d show the expanded front and its level set representation at time t.

In the important special case, where F_N is a function of x,t and $\nabla\phi$, Eq. 20 becomes a Hamilton-Jacobi (H-J) equations whose solutions generally develop kinks (jumps in derivatives). The unique viscosity solutions [43] is sought. At the end, a viscosity scheme for the level set representation can be expressed as

$$\phi_{ij}^{n+1}=\phi_{ij}^n-\Delta t F_{ij}\left[\max\left(D_{ij}^{-x},0\right)^2+\min\left(D_{ij}^{+x},0\right)^2+\max\left(D_{ij}^{-y},0\right)^2+\min\left(D_{ij}^{+y},0\right)^2\right]^{1/2}$$ **(3.8)**

where $D_{i,j}^{-x}=\dfrac{\phi_{i,j}-\phi_{i-1,j}}{h}, D_i^{+x}=\dfrac{\phi_{i+1,j}-\phi_{i,j}}{h}, \ D_{i,j}^{-y}=\dfrac{\phi_{i,j}-\phi_{i,j-1}}{h}, D_i^{+y}=\dfrac{\phi_{i,j+1}-\phi_{i,j}}{h}$ and h is the grid spacing.
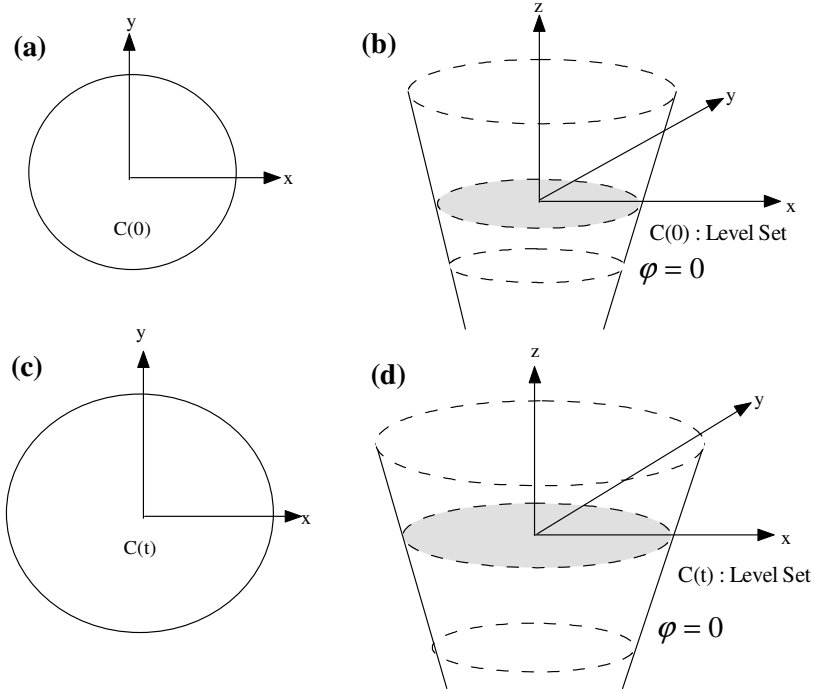
Figure 3.1 : Level Set representation, (a) and (b) shows the front C and surface φ at time t=0, (c) and (d) show the front C and surface φ at any time t.

3.2 Fast Marching Method

Fast Marching (FM) method is a very fast version of the level set methods with some limitations, that is curve propagation speed F must be of constant sign and the curve must evolve in one direction. If the speed value is always of constant sign, the FM method guarantees that one image element is passed only one time by the front. That means that the front never needs going back and revisiting a point again. Therefore the arrival time $T(x, y)$ of the front as it crosses the point (x, y) can be used to represent the position of the front. These arrival time values are calculated based on the well-known equation: *Distance = Time * Rate*. Than

$$1 = F \frac{dT}{dx} \tag{3.9}$$

If multidimensional time and speed terms are considered, the equation becomes

$$|\nabla T| F = 1 \tag{3.10}$$

Eq. 3.10 states that the gradient of the arrival time is inversely proportional to the speed of the front. That means the front around the object boundaries can be slowed down and even stopped by adjusting the speed value F. State that T is zero on the initial front.

- If the speed function F depends only on position and first derivatives of the solution T, the resulting equation is a static Hamilton-Jacobi equation.

- If the speed function F depends only on the position (x, y), then the resulting equation is the familiar Eikonal equation.

- In any case, the solution T typically is multi-valued; although it is required that the speed function F be strictly positive (or negative), this in itself does not ensure that the solution T only reflects a single crossing of the point (x, y). In fact, our solution is restricted to the so-called viscosity solution which limits the solution to the first crossing time T [44].

In order to approximate the equations of motion, the key idea is to select an approximation to the gradient operator ∇T which correctly chooses this correct limiting weak solution [17]. The approximate solution of Eq. 3.10 on a 2D grid is given as

$$\left[\max\left(D_{ij}^{-x}T, 0\right)^2 + \min\left(D_{ij}^{+x}T, 0\right)^2 + \max\left(D_{ij}^{-y}T, 0\right)^2 + \min\left(D_{ij}^{+y}T, 0\right)^2 \right] = \frac{1}{F_{ij}^2} \qquad (3.11)$$

where $D_{ij}^{-x}T = \dfrac{T_{i,j} - T_{i-1,j}}{\Delta x}, D_{ij}^{+x}T = \dfrac{T_{i+1,j} - T_{i,j}}{\Delta x}, D_{ij}^{-y}T = \dfrac{T_{i,j} - T_{i,j-1}}{\Delta y}, D_{ij}^{+y}T = \dfrac{T_{i,j+1} - T_{i,j}}{\Delta y}$

and F_{ij} is the speed value at the (i, j) position. As seen in Eq. 3.11, it is a boundary value problem which states that the time difference between neighbor points cannot exceed the inverse of the speed. It also guarantees the one-way evolution of the curve, from smaller values of T to larger values.
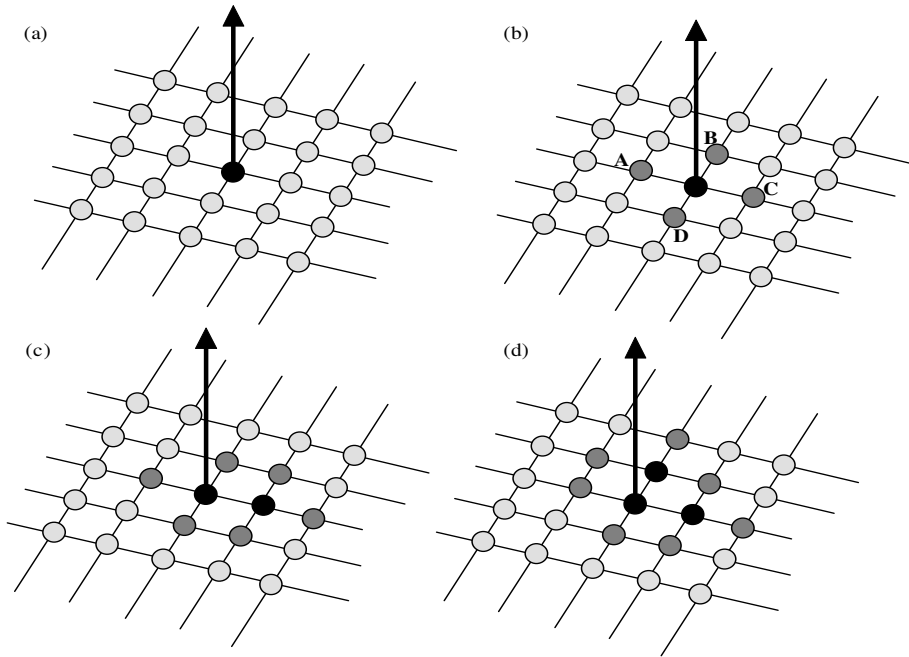
Figure 3.2 : Update procedure for Fast Marching Method.

3.2.1 Fast Marching Algorithm

Figure 3.2 represents the basic steps of the Fast Marching method. Let the beginning boundary value be at the origin as in Figure 3.2.a. The light gray points are the unknown "far away" points. The new time values for the 4-neighboring grid points (dark gray points) are calculated using Eq. 3.11 (Figure 3.2.b).The time value of any grid point is calculated as adding the neighboring time values and the time difference between the grid point and neighbor point. The time difference is simply multiplication of speed value with the distance between the grid point and neighbor point. Then the grid point with a minimum time value is selected. Let point C have the minimum time. Point C can change its color from gray to black. Now, 4-neighbors of C are updating their time values according to Eq. 3.11 (Figure 3.2.c). Suppose point B has minimum time values in this iteration. It changes its color to black and its neighbors update the time values (Figure 3.2.d).
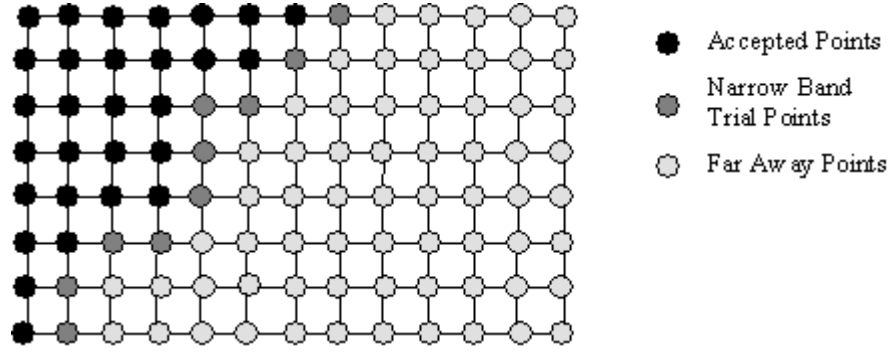
Figure 3.3 : Progress of Fast Marching Method on 2-D grid with specific types of points.

Algorithm of the Fast Marching Level Set progress can be summarized as follows (See Figure 3.3):

A. Initialization Step

    a. Initialize the front and set all initial points A_{ij} as **Accepted Points**. Assign $T_{ij} = 0$.

    b. Assign all 4-neighbors of the initial points (on the propagation way) A as **Narrow Band Trial Points**. Set $T_{ij} = dy / F_{ij}$. (dy is 1 for 4-neighborings and $\sqrt{2}$ for 8-neighborings)

    c. Set all other points as **Far Away Points**. Assign $T_{ij} = \infty$ for the points on propagation way, assign $T_{ij} = -\infty$ otherwise.

B. Marching Step

    a. Begin loop: Find the point that has minimum time values among trial points (P_{min})

    b. Add the point P_{min} to accepted points and remove from trial points.

    c. Look the 4-neighboring points of P_{min} one by one, if it is a far away points, remove from there and add to trial points.

    d. Recompute the time values for all neighbors according to Eq. 3.11, selecting the largest possible solution to the quadratic equation.

    e. Return to a, if trial points are not empty.

3.2.2 Modeling the Speed Function

As seen in Eq. 3.11 the arrival time values of each grid point are absolutely dependent on the speed function F. Therefore the Fast Marching system can be driven only by selecting the speed values. Since the aim of this study is to capture the objects in images, F should be decreased as the front gets near the object boundaries.

Mostly, the object boundaries can be differentiated from other regions with their high gradients. Then, a speed function that is inversely proportional with the gradient magnitude can be defined:

$$F = \beta \frac{1}{1 + \alpha |\nabla G_\sigma * I|}$$

(3. 12)

Here $\nabla G_\sigma * I$ denotes the gradient of the input image, which is obtained by convolving the image with a 2-D Gaussian filter with zero mean and σ variance. β is the maximum speed of the front when the gradient goes to zero. And α is the scaling factor of the gradient magnitude.

Another version of the gradient based speed function can be expressed as follows;

$$F = \beta \exp(-\alpha |\nabla G_\sigma * I|)$$

(3. 13)

Malladi et al. proposed some useful Level Set speed functions, that is dependent on the geometry of the front or local gradients of the image [45]. They separated the speed function F into two components:

$$F = F_A + F_G.$$

(3. 14)

The term F_A, referred to as the advection term, is independent of the moving front's geometry. The front uniformly expands or contracts with speed F_A depending on its sign and inflation forces [45]. The second term F_G, is the part that depends on the geometry of the front, such as its local curvature κ which is defined as

$$\kappa = \nabla \cdot \frac{\nabla \psi}{|\nabla \psi|} = -\frac{\psi_{xx} \psi_y^2 - 2\psi_x \psi_y \psi_{xy} + \psi_{yy} \psi_x^2}{\left(\psi_x^2 + \psi_y^2\right)^{3/2}}$$

(3. 15)

where ψ is the level set function. This diffusion term smoothes out the high curvature regions of the front and has the same regularization effect on the front as the internal deformation energy term in thin-plate membrane splines.

4. THE PROPOSED GRADIENT BASED SHAPE DESCRIPTORS

4.1 Introduction

Shape representation and description have been playing important roles in many areas of computer vision, pattern recognition, and robotics. They include character recognition, fingerprint matching and industrial inspection [19]. There are two recent tutorials on the shape description and matching techniques. Veltkamp and Hagedoorn [30] investigated the shape matching methods in four parts: global image transformations, global object methods, voting schemes and computational geometry. They also worked on shape dissimilarity measures. Another review on shape representation methods is accomplished by Zhang and Lu [31]. They first classified the solutions into two categories: contour-based and region-based methods, also referred to as external and internal techniques, respectively. The classification is based on whether shape features are extracted only from the contour or they are extracted from the whole shape region. Under each class, the different methods are further divided as structural approaches and global approaches. This sub-classification is based on whether the shape is represented as a whole or by segments/sections called primitives.

Our study falls into global contour-based shape description class utilizing only object boundaries. We assumed that the object is adequately segmented from the background and the boundaries are extracted. Two approaches most related to this study are shape signatures and shape matching. A shape signature represents a shape by a function extracted from object boundary points. Examples of shape signatures include complex coordinates, centroid distance and curvature. Shift matching is required to compensate the rotation variations between two shapes. Shape signatures are also sensitive to noise, local shape deformations and occlusion. Spectral transformations such as the Fourier Transform and the Wavelet Transform are applied to the signatures to decrease the sensitivity to noise and local shape deformations, and the Fourier descriptors and the Wavelet Descriptors are extracted. Matching is done in transform space in this case.

In contrast to the previous approach, shape matching works in the spatial domain and measures the point-to-point similarity between two shapes. One example of shape matching methods utilizes Hausdorf distance [46]. One advantage of the Hausdorf distance is that it can make a partial match. On the other hand, it is not translation, scale, and rotation invariant. Belongie et al. [47] proposed a shape matching approach attaching a feature called shape context to each point on the boundary. Then they solved one-to-one correspondence problem assuming that corresponding points have the similar shape context. Zhuowen and Alan [48] presented an algorithm for shape matching based on generative model to show one shape can be generated by the other. The matching process is formulated by the expectation maximization. Although shape matching algorithms yield precise retrieval results, their computational cost due to matching is unacceptable for online shape retrieval systems.

In this study, our purpose is to close the gap between signature-based descriptors and shape matching using gradient-based local description by increasing the recognition performance of the signature-based approaches. The local descriptors are increasingly used in image recognition due to their robustness to occlusions and geometrical deformations. Yokono and Poggio [49] investigated the performance of local descriptors using various combinations of Gaussian derivatives with different orientations and scales for an object recognition task. They compared the performances in terms of selectivity and invariance to several affine transformations such as rotation, scale changes, brightness changes, and viewpoint changes. They reported that the Gaussian derivative descriptor outperformed other Gauss-like filter descriptors. In this work, we propose two contour-based global shape description schemes using responses to a set of steerable filters along the object boundary. On contrary to Yokono and Poggio's study [49] we use the gradient information to describe object shape instead of object texture.

In the proposed schemes, in order to capture the shape information, we extract local image gradient on the boundary while we trace it. The gradient feature is a two-dimensional vector which may have various orientation and magnitude depending on the local intensity distribution. Magnitude and orientation are of high importance when they are used as shape clues. The gradients are arranged into a shape clue matrix. This shape clue satisfies a rule called shifting property (see appendix A). The

proposed affine-invariant shape signatures are explained in Section 2. The steerable G-filter kernels are employed to obtain the local image gradient data in this study. Steerable filters are frequently used to detect local shape features in many region-based texture and shape description techniques. They can provide orientation information, which are sensitive to the presence of edges, bars and other simple image structures [50, 51, 52, 53]. It has been shown that the generalized edge detector based on G-filter can deliver improved results when compared to first order Gaussian derivative [54]. In addition, G-filter is capable of producing different edge detectors including the first order Gaussian derivative as we change its two parameters (i.e. λ and τ). We are interested in the filter responses only along object boundaries and these responses are aggregated into a one-dimensional shape signature. The Fourier Transform is then applied to the proposed shape signature in order to provide a representation invariant to starting point and to have compact description. Fourier Descriptor (FD) is one of the most widely used shape descriptor due to its low computation complexity, clarity and coarse to fine description capability [32, 33, 34, 35].

Different image signatures are reported in the literature based on "color", "texture" or "shape". Since we are only interested in shape based signatures, we consider the well-known "shape" signatures in our investigation such as complex coordinates, centroid distance, curve bending angles and curvature signature. Although boundary-based shape signatures described above provide a means by which the shape of the object is described, they have limited selectivity feature especially when used with the content-based image retrieval systems. In order to solve this problem, we suggest an approach where we can manage the selectivity by scale and direction. Another advantage of the proposed method is that it can be applied to both grayscale and binary images. This is especially important when it has been used in content-based image retrieval systems due to the fact that the databases are mostly comprised of color images.

4.2 Directional Gradient Extraction Using Steerable Filters

In this study, we are only interested in boundary based shape descriptors and assume that objects always have closed boundaries. Many shape descriptors exist in the literature and most of these descriptors are not able to address varieties of shape

variations in nature such as rotation, scale, skew, stretch, and noise. We propose an affine invariant shape descriptor in this study, which handles rotation, scale and skew transformations. The basic idea of the proposed approach is to use gradient information at the boundaries rather than using the boundary locations alone. We used 2D Generalized Edge Detector [54] to obtain object boundary, but other well-known edge operators like Canny's edge detector [55] can also be used. We then trace the detected boundary pixels along the clockwise direction to attain the locations of the neighboring boundary pixels denoted as (x_i, y_i). So the object boundary (Γ) is easily obtained as an N by 2 matrix

$$\Gamma = \{[x_i, y_i]; i = 1, 2, ..., N\}, \tag{4. 1}$$

where N is the length of the contour, i.e. $N = |\Gamma|$. Since we are interested in the directed gradients at these boundary locations, we utilize steerable G-Filters to obtain the gradient at certain directions and scales as

$$D_{(\lambda, \tau)}^{\theta}(I, x_i, y_i) = \left(I * G_{(\lambda, \tau)}^{\theta}\right)(x_i, y_i), \tag{4. 2}$$

where I is the image intensity. Steerable $G_{(\lambda, \tau)}^{\theta}$ filter is defined in terms of $G_{(\lambda, \tau)}^{\theta=0}$ as

$$G_{(\lambda, \tau)}^{\theta}(x', y') = G_{(\lambda, \tau)}^{\theta=0}(x, y)$$
$$[x', y']^T = R(\theta)[x, y], \tag{4. 3}$$

where $R(\theta)$ is the rotation matrix, i.e.

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{4. 4}$$

Detailed analysis of these filters $G_{(\lambda, \tau)}^{\theta=0}$ is given in [54]. In case G is not available, one can use the first derivative of Gaussian as G filter. Let us denote the response matrix as $F(\Gamma) = \begin{bmatrix} f_{k,m} \end{bmatrix}$ where $f_{k,m}$ is equal to $D_{(\lambda, \tau)}^{\theta_m}(I, x_k, y_k) = \left(I * G_{(\lambda, \tau)}^{\theta_m}\right)(x_k, y_k)$ as

$$F(\Gamma) = \begin{bmatrix} D_{(\lambda,\tau)}^{\theta_0}(x_1, y_1) & \cdots & D_{(\lambda,\tau)}^{\theta_{M-1}}(x_1, y_1) \\ D_{(\lambda,\tau)}^{\theta_0}(x_2, y_2) & \cdots & D_{(\lambda,\tau)}^{\theta_{M-1}}(x_2, y_2) \\ \vdots & \ddots & \vdots \\ D_{(\lambda,\tau)}^{\theta_0}(x_N, y_N) & \cdots & D_{(\lambda,\tau)}^{\theta_{M-1}}(x_N, y_N) \end{bmatrix}$$

(4. 5)

assuming that we use M number of steerable filters whose directions are multiple of π/M such that $\theta_m = m\pi/M$. In this case, the size of F is $M|\Gamma|$. The local shape becomes more accurately described as M, the number of filters, increases.

An alternative to the previous gradient-based signature definition is to treat the gradients in the perpendicular directions as a complex number rather than treating them as independent data. In this case, the response matrix takes the following form

$$H(\Gamma) = \begin{bmatrix} \left(D^{\theta_0} + jD^{\theta_0 + \pi/2}\right)(x_1, y_1) & \cdots & \left(D^{\theta_{M-1}} + jD^{\theta_{M-1} + \pi/2}\right)(x_1, y_1) \\ \left(D^{\theta_0} + jD^{\theta_0 + \pi/2}\right)(x_2, y_2) & \cdots & \left(D^{\theta_{M-1}} + jD^{\theta_{M-1} + \pi/2}\right)(x_2, y_2) \\ \vdots & \ddots & \vdots \\ \left(D^{\theta_0} + jD^{\theta_0 + \pi/2}\right)(x_N, y_N) & \cdots & \left(D^{\theta_{M-1}} + jD^{\theta_{M-1} + \pi/2}\right)(x_N, y_N) \end{bmatrix}$$

(4. 6)

where $\theta_m = \dfrac{m\pi}{2M}$.

When the object is rotated α degrees around the center of gravity, the columns of the matrix F is circularly shifted to left or right depending on whether the objects rotate clockwise or counterclockwise direction. The relationship between the rotation angle and amount of shifting can be stated as follows assuming that the rotation angle is a multiple of π/M,

$$F^{\Gamma^\alpha} = \left[f'_{k,m} \right], \alpha = s\frac{\pi}{M},$$
$$f'_{k,(m+s)\bmod M} = f_{k,m}.$$

(4. 7)

where $\Gamma^\alpha = \left\{ R(\alpha)[x_i, y_i]^T \right\}$. This property is called as perfect shifting-property.

In Eq. 4.5, $F(\Gamma)$ is an N by M matrix while in Eq. 4.6 $H(\Gamma)$ is an N by $M/2$ matrix. On the other hand, $F(\Gamma)$ is a real valued matrix while $H(\Gamma)$ is a complex

valued matrix. Note that both matrices $F(\Gamma)$ and $H(\Gamma)$ satisfy the perfect shifting rule under the rotations when θ is exact multiple of π / M.

In order to demonstrate the circular-shifting rule, we obtain the filter response matrix (F) at different steering angles both for the object (the digit "5") and its rotated version. We first apply the filter at each pixel on the boundary of the sample object given in Fig. 4.1(a) and then apply it on the boundary of its $90°$ rotated version given in Fig 4.1(d). The steering angle is changed from $0°$ to $180°$ with $1°$ increments. The response matrices are plotted in Fig. 4.1(b) and in Fig. 4.1(e) for $0°$ and $90°$ respectively. Fig. 4.1(c) shows how the steerable filter response changes with the rotation angle at one pixel on the boundary. In other words, Fig. 4.1(c) is obtained by taking the vertical cross-section of Fig. 4.1(b). The cross-section of F for the rotated object on the same pixel is shown in Fig. 4.1(f). It can be easily verified from the figures Fig. 4.1(c) and Fig. 4.1(f) that they satisfy the equation (4.7) where $s = 90$ and $M = 180$. The experimental results are consistent with the circular-shifting rule.

We also analyze how the circular-shifting rule is affected when the rotation angle is not exact multiple of π / M. Let us assume that rotation angle is $s\dfrac{\pi}{M} + \phi$ where $0 \leq \phi \leq \dfrac{\pi}{M}$ and we have M steerable filters $\left\{ G^{\theta_k} ; k = 0, 1, ..., M-1 \right\}$. Any steerable filter response in one direction can be expressed as a linear combination of M different G^{θ_k} filter responses as [49]

$$G^{\varphi} = \sum_{k=0}^{M-1} c_k(\varphi) G^{\theta_k}$$

$$c_k(\varphi) = \cos\left(\varphi - \frac{k\pi}{M} \right) = \cos(\varphi - \theta_k)$$

(4. 8)

32
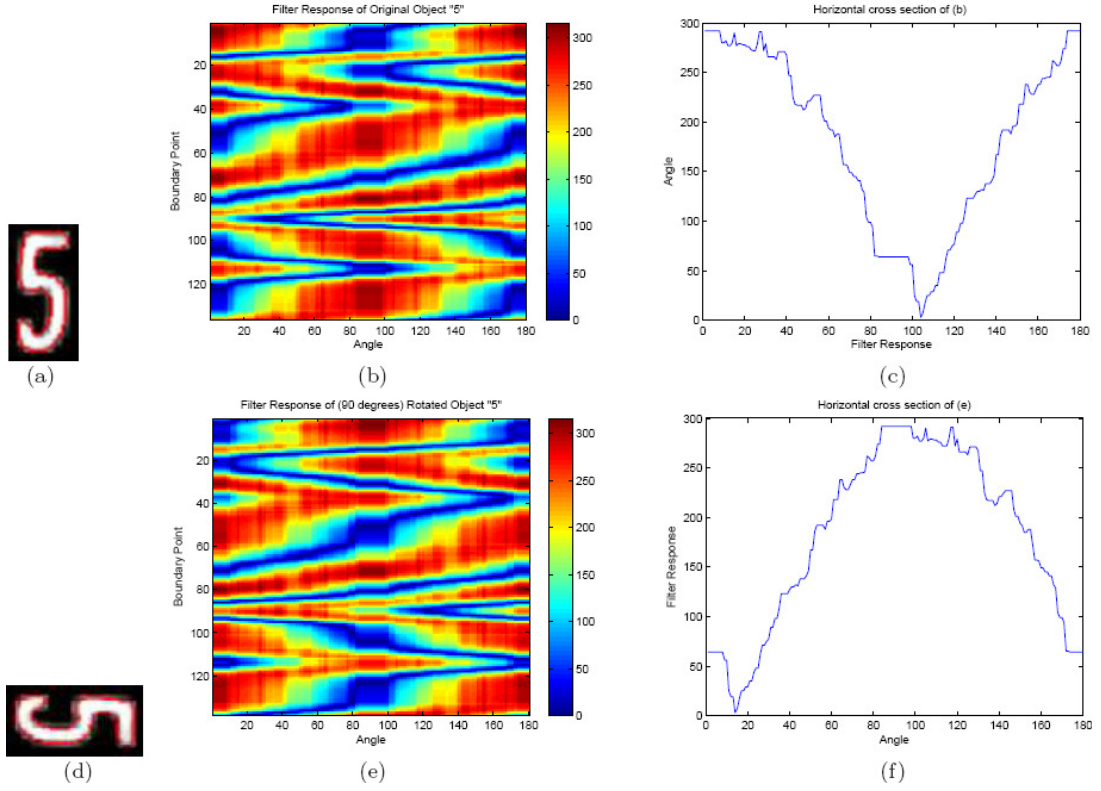
Figure 4.1 : (a) Sample digit "5", (b) Filter responses on object boundary pixels for each direction, (c) Steerable filter response at one pixel on the curve with respect to steering angle, which corresponds to a vertical cross section of (b). (d) 90 degrees rotated digit (e) Filter responses on object boundary pixels for each angle,(f) Steerable filter response at the same pixel on the curve with respect to rotation angle, which corresponds to a vertical cross section of (e).

The difference between the signatures obtained for the same object, one is rotated $\alpha = s\dfrac{\pi}{M}$ degrees and the other is rotated $\alpha + \phi = s\dfrac{\pi}{M} + \phi$ degrees can be found as

$$\left\| f_{k,m} \right\|^2 = \left\| I^{\alpha} * G^{\theta_m} \right\|^2$$

$$\left\| \hat{f}_{k,m} \right\|^2 = \left\| I^{\alpha+\phi} * G^{\theta_m} \right\|^2 = \left\| I^{\alpha} * G^{\theta_m+\phi} \right\|^2$$

$$\left\| f_{k,m} - \hat{f}_{k,m} \right\|^2 = \left\| I^{\alpha} * G^{\theta_m} - I^{\alpha} * G^{\theta_m+\phi} \right\|^2 \qquad \textbf{(4. 9)}$$

$$\left\| f_{k,m} - \hat{f}_{k,m} \right\|^2 = \left\| I^{\alpha} * \left(G^{\theta_m} - G^{\theta_m+\phi} \right) \right\|^2$$

$$\left\| f_{k,m} - \hat{f}_{k,m} \right\|^2 = \left\| I^{\alpha} \right\|^2 \left\| G^{\theta_m} - G^{\theta_m+\phi} \right\|^2 .$$

Now Eq. 4.8 could be used to compute the difference between the responses obtained for θ_m and $\theta_m + \phi$ degrees rotated filter kernels as

$$\left\| G^{\theta_m} - G^{\theta_m + \phi} \right\|^2 = \left\| G^{\theta_m} - \sum_{k=0}^{M-1} \cos\left(\theta_m + \phi - \theta_k\right) G^{\theta_k} \right\|^2 \qquad \textbf{(4. 10)}$$

This formulation helps determine the number of directions and consequently the size of the descriptor. Our aim is to minimize $\varepsilon = \left\| f_{k,m} - \hat{f}_{k,m} \right\|$. For a given ε, one can easily obtain M using the above equation. ε can be considered as desired description level. Shape is better described as we decrease ε. There is a tradeoff between description level and computational cost. The computation cost increases as the descriptor size increases. Experimental results reveal that we do not need to use large numbers of filters for $F(\Gamma)$, and even $M = 2$ is sufficient when $H(\Gamma)$ is used as signature. Next section will discuss how to use Fourier Descriptor applied to the gradient signature that we discussed above and how to define a distance function between two descriptors.

4.3 Fourier Based Shape Descriptor

Fourier Descriptors (FDs) are mostly employed for boundary shape description. Zhang and Lu [20] compared shape retrieval using FDs derived from different shape signatures and from different Fourier invariants in terms of computational complexity, robustness, convergence speed, and retrieval performance. They reported that the centroid distance shape signature outperforms other signature methods in terms of above criteria. Selecting the shape signature is the most critical step for FDs. Various signature models were proposed in the literature such as complex boundary coordinates, centroid distance and boundary curvatures. These signatures are summarized Section 2.1.

In the proposed shape descriptors we apply Fourier transform to the gradient based shape signatures described in the previous section in order to obtain a compact descriptor. We denote the descriptors as $\tilde{F} = \left[\left\| \tilde{f}_{k,m} \right\| \right]$ and $\tilde{H} = \left[\left\| \tilde{h}_{k,m} \right\| \right]$ where the coefficients are computed as follows

$$\tilde{f}_{k,m} = \frac{1}{N} \sum_{n=0}^{N-1} f_{k,m} \exp\left(\frac{-j2\pi kn}{N}\right)$$
$$\tilde{h}_{k,m} = \frac{1}{N} \sum_{n=0}^{N-1} h_{k,m} \exp\left(\frac{-j2\pi kn}{N}\right)$$

(4. 11)

where N is the length of the curve (Γ) and $k \in [1..L]$. We take the magnitude of the Fourier coefficients to keep the descriptor invariant to starting point on the boundary ([57]). Scale invariance of the descriptor is achieved by dividing the magnitudes by the DC component, i.e., $\tilde{f}_{0,m}$ ([57]). Note also that both $\tilde{F}$ and $\tilde{H}$ still satisfy the circular-shifting rule when we rotate the object as

$$\tilde{F}(\Gamma) = \left[\tilde{f}_{k,m}\right]$$
$$\tilde{F}(\Gamma_\alpha) = \left[\tilde{f}'_{k,m}\right], \alpha = s\frac{\pi}{M}$$
$$\tilde{f}'_{k,m} = \tilde{f}'_{k,(m+s)\bmod M,}$$

(4. 12)

which also applies to $\tilde{H}$ and $\tilde{h}_{k,m}$. The proof is given in the Appendix A.

Finally, we present a distance function to compare two descriptors. Assume that we are given two descriptors such as $\tilde{f}_{k,m}$ and $\tilde{p}_{k,m}$. Then we define the distance between two descriptors considering the shifting-property as

$$SD(\tilde{f}, \tilde{p}) = \min_{r \in \{0,\ldots,M-1\}} \left\| \tilde{f}_{k,(m+r)\bmod M} - \tilde{p}_{k,m} \right\|_2$$

(4. 13)

The distance between two descriptors (e.g. $\tilde{f}_{k,m}$ and $\tilde{f}'_{k,m}$) that belong to one object and its s-rotated form (corresponding to the angle $s\frac{\pi}{M}$) is given by

$$SD(\tilde{f}, \tilde{f}') = \min_{r \in \{0,\ldots,M-1\}} \left\| \tilde{f}_{k,(m+r)\bmod M} - \tilde{f}'_{k,m} \right\|_2 = 0$$

(4. 14)

where $\tilde{f}'_{k,m} = \tilde{f}_{k,(m+s)\bmod M}$ due to circular-shifting rule. In addition, an estimate of rotation can be computed as

$$\Theta\left(\tilde{f},\tilde{p}\right)=\left(\arg\min_{r\in\{0,\ldots,M-1\}}\left\|\tilde{f}_{k,(m+r)\bmod M}-\tilde{p}_{k,m}\right\|_2\right)\frac{\pi}{M} \tag{4.15}$$

assuming that $\tilde{f}$ and $\tilde{p}$ are in the same object class. The estimation error upper bound is $\frac{\pi}{2M}$. Several experiments are performed again on the same digit "5" to study the actual estimation error for various M values (2, 4, 6, 8, 10, 12, 14, 16). The results are given in Table 4.1. It shows that the actual error is much smaller than the theoretical upper-bound.

The computational complexity of the equation 4.19 is $M\times(M\times L)$ where M is the number of filter kernels used and L is the number of Fourier coefficients used.

Table 4.1 : Rotation angle estimation error $\Theta(\ )$ for various M and L = 15.

M	2	4	6	8	10	12	14	16
$\Theta(\ )$	15.18°	11.45°	7.61°	5.76°	4.69°	3.95°	3.50°	3.05°

4.4 Experimental Work and Results

The performance of the proposed descriptors is evaluated over two databases; one containing digits taken from vehicle license plates and the other containing MPEG-7 Core Experiment shape data set. First, we demonstrate the performance of our algorithm on the license plate characters. Fig. 4.2 shows typical segmented license plate characters selected from the database, which contains 8321 grayscale digit characters segmented from the real vehicle license plates. About the half of the characters in the database, namely 4121 digits, is used in training, and the remaining 4200 digits are kept for testing. In practice, although there are 10 digits appear on the license plate images, we only labeled 9 classes because one can not distinguish between 6 and 9 under rotation assuming that no prior information is available.
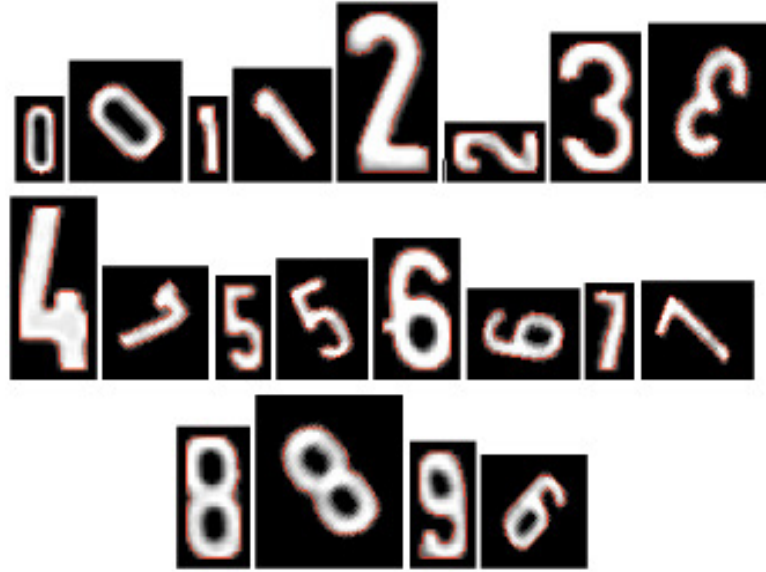
Figure 4.2 : Typical segmented license plate characters selected from the database.

In the first group of experiments, we explored how the recognition rate changes with the size of descriptor. The descriptors have two dimensions: number of kernel filters denoted as M and the number of Fourier coefficients denoted as L. The results are summarized in Table 4.2. The proposed descriptor $\tilde{F}$ results in 99% recognition rate for $M > 6$ and $L > 3$. So the optimum value of L can be selected as 4. When compared to centroid distance with a performance of at most 90% for $L = 15$, this is a considerable improvement. Another observation is that the performance of centroid distance drops dramatically when L gets smaller. For example, the recognition performance is 60% for $L = 3$ whereas the performance of the proposed descriptor which is 95% is much higher, for $M = 8$ and $L = 3$. In Fig. 4.3, we plot the recognition rates, as a function of L for four methods, i.e. the proposed descriptors ($\tilde{F}(M = 8)$, $\tilde{H}(M = 2)$), centroid distance, boundary curvature, and complex coordinates. It shows a considerable improvement achieved by the proposed methods. We presented only one result for the descriptor $\tilde{H}(M = 2)$ in Table 4.2 since the recognition rate performance obtained for $M > 2$ is very close to the performance obtained for $M = 2$ as shown in Table 4.3.

Table 4.2 : Recognition rate with respect to descriptor size (M,L).

Method	Number of Coefficients					
	15	**10**	**7**	**5**	**3**	**2**
Centroid distance	90.21	90.14	84.54	76.11	59.90	52.95
Curvature	89.5	88.88	88.45	85.45	67.54	55.28
Complex Coor.	93.66	89.28	91.54	73.19	50.42	35.90
$\tilde{F}(M=2)$	57.40	54.04	47.59	44.97	32.54	24.02
$\tilde{F}(M=3)$	82.69	81.40	76.83	73.40	59.26	42.61
$\tilde{F}(M=4)$	91.71	89.52	86.11	82.19	67.71	54.38
$\tilde{F}(M=5)$	96.14	95.64	93.57	90.35	81.81	71.45
$\tilde{F}(M=6)$	97.66	97.69	96.02	95.21	88.88	79.42
$\tilde{F}(M=8)$	99.28	99.47	99.14	98.73	95.73	88.00
$\tilde{F}(M=12)$	99.04	99.09	99.19	99.23	98.00	94.97
$\tilde{F}(M=16)$	99.16	99.42	99.33	99.11	98.00	96.00
Complex Gradient	98.54	96.76	93.4	68.61	34.64	20.23

Table 4.3 : Recognition rates obtained for the descriptor $\tilde{H}$.

Method	Number of Coefficients					
	15	**10**	**7**	**5**	**3**	**2**
Centroid distance	90.21	90.14	84.54	76.11	59.90	52.95
Curvature	89.5	88.88	88.45	85.45	67.54	55.28
Complex Coor.	93.66	89.28	91.54	73.19	50.42	35.90
$\tilde{H}(M=2)$	98.54	96.76	93.4	68.61	34.64	20.23
$\tilde{H}(M=4)$	98.61	96.59	93.23	68.26	34.71	19.76
$\tilde{H}(M=6)$	98.64	96.45	93.73	68.02	34.4	20.69
$\tilde{H}(M=8)$	98.59	96.35	93.09	68.42	33.8	19
$\tilde{H}(M=12)$	98.67	96.54	93.66	68.04	34.23	20.9
$\tilde{H}(M=16)$	98.54	96.42	93.23	68.57	34.88	20.8

The fact that the performance of the descriptor $\tilde{H}$ does not change considerably as M increases is very important in several aspects. First of all, for $M=2$ the descriptor $\tilde{H}$ has only one column. In this special case, the corresponding signature vector can be easily computed by using horizontal and vertical components of gradient operator. Secondly the distance measure in (Eq. 4.13) reduces to Euclidean distance since shifting and minimum operations are not required in this case. Another advantage is that the distance measure is a metric now.

The recognition performance of the descriptor $\tilde{F}$ is more sensitive to M and nearly constant along the dimension L. On the other hand, the descriptor $\tilde{H}$ is sensitive to

L and almost constant along the dimension M. Since the computational complexity is given by $M^2 L$, the descriptor $\tilde{H}$ better manages the tradeoff between the descriptor size (ML) and the computational cost.

We now consider the case that the edges are detected somewhat far away from their ideal locations. We demonstrate the recognition results over the same data set, but this time we use poorly detected edges that are obtained at the early stages of a fast marching algorithm [58] without using global shape information (Fig. 4.4). The recognition results are given in Table 4.4. With regard to the results shown for the ideal edges and poorly detected edges in Table 4.2 and Table 4.4, the superiority of the proposed descriptor over the centroid distance and curvature is now much more evident than the case with the ideal edges. The results show that the proposed method is more robust to the errors introduced during the contour detection stage as compared to other methods. In the second group of experiments, we explored how the recognition rate changes with the scale parameters, i.e. (λ, τ), of the filter kernel.
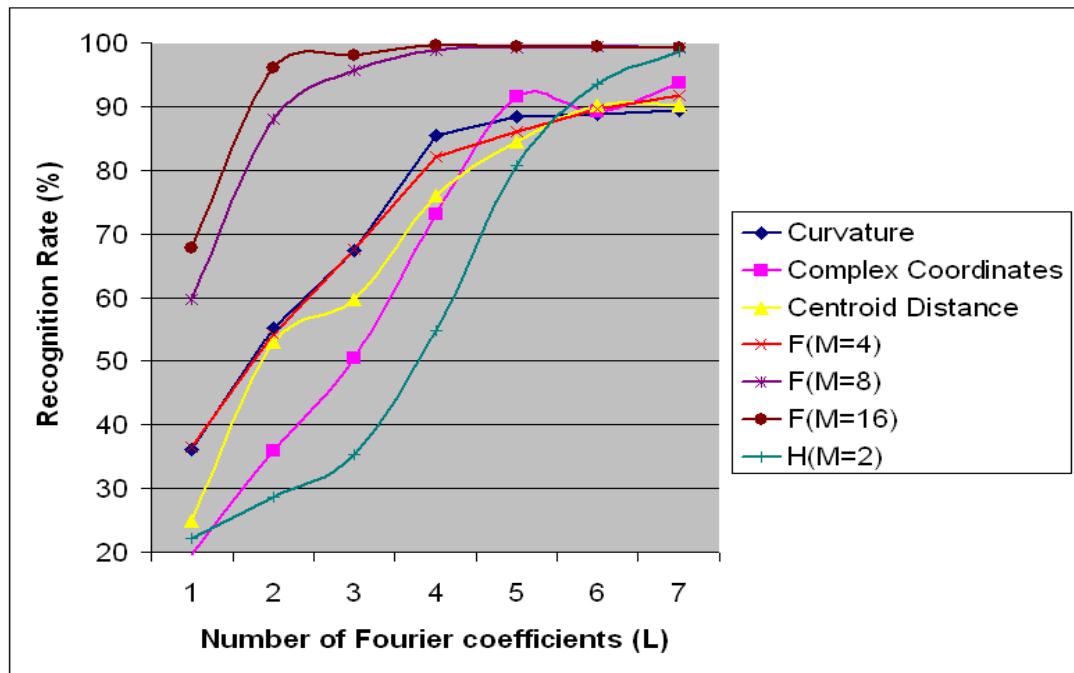
Figure 4.3 : Recognition rates for four methods, the proposed descriptor (M = 8), centroid distance, boundary curvature, complex coordinates with respect to L where L is the number of Fourier coefficients used.
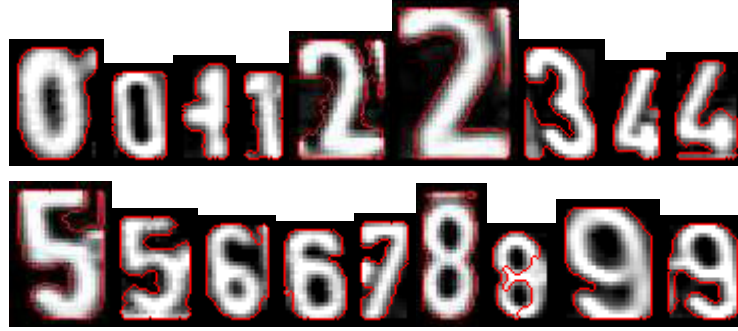
Figure 4.4 : Poorly segmented license plate characters that are selected from the database.

Table 4.4 : Recognition rate with respect to descriptor size (M,L) obtained for the license plate digits where poorly detected edges are used.

Method	Number of Coefficients					
	15	**10**	**7**	**5**	**3**	**2**
Centroid distance	88.66	87.74	85	76.72	61.13	48.12
Curvature	86.3	85.28	83.6	79.31	69.25	57.23
Complex Coor.	88.61	68.88	49.09	33.31	24.12	20.91
$\tilde{F}(M=2)$	93.77	93.2	91.61	88.72	72.49	55.74
$\tilde{F}(M=3)$	96.07	95.68	94.88	92.63	83.39	71.12
$\tilde{F}(M=4)$	96.81	96.7	96.38	95.45	89.53	80.73
$\tilde{F}(M=5)$	97.15	96.94	96.62	96.13	93.18	87.81
$\tilde{F}(M=6)$	97.47	97.3	97.09	96.7	95	91.31
$\tilde{F}(M=8)$	97.32	97.33	96.98	96.58	94.64	91.54
$\tilde{F}(M=12)$	97.40	97.53	96.87	96.38	94.41	91.52
$\tilde{F}(M=16)$	97.17	96.98	96.83	96.36	94.41	91.29
Complex Gradient	90.88	81.6	59.8	47.63	30.15	23.84

Table 4.5 : Recognition rate with respect to filter scale.

	(Filter Size, λ, τ)			
	(7,1.5,0.225)	**(9,2,0.268)**	**(11,2.5,0.3)**	**(13,3,0.33)**
The proposed descriptor (M=8, L=5)	97.19	98.73	99.59	99.54

We change the filter size from 7 to 15, hence the scale λ from 1.5 to 3.0. The results are summarized in Table 4.5. In this experiment, the highest recognition rate is achieved for $\lambda = 2.5$.

In the last group of experiments over the license plate characters, we explored how M and ϕ effect the shifting property of the descriptor. We rotate the same object by 1° angles and compute the distance between the original object and its rotated

versions by using Eq. 4.13. For the sake of brevity, only the average distances are presented in Table 4.6. We apply the same method to both grayscale and binary objects. The first row of the Table 4.6 holds the average distances for 9 different kernel sets applied to binary objects. As M increases, we describe the shape much better as indicated by Eq. 4.10. Results from grayscale objects are given in the second row of this table. The results are collectively plotted in Fig. 4.5 for different values of M {2, 4, 5, 6, 8, 10, 12, 14, 16}. As M increases, the rotation invariance property of the defined distance measure becomes more evident.

Table 4.6 : Average distances between the object and its 1^0 rotated versions (Binary object results are given in the first row, and Grayscale object results are given in the second row).

M=2	M=4	M=5	M=6	M=8	M=10	M=12	M=14	M=16
0.018	0.013	0.013	0.013	0.013	0.012	0.012	0.012	0.012
0.012	0.007	0.007	0.006	0.006	0.005	0.005	0.004	0.004
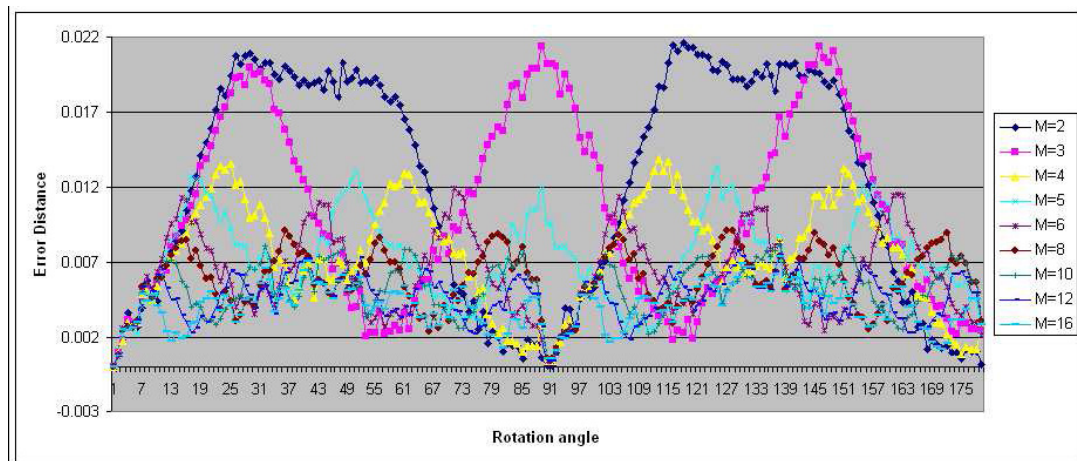
Figure 4.5 : Distance error between the object and its rotated versions for M = 2, 4, 5, 6, 8, 10, 12, 14, 16.

We have tested our algorithm on a subset of MPEG-7 Core Experiments Shape-1 Part B which contains 20 images for each 42 binary objects. Table 4.7 summarizes the recognition rates with respect to both rank and descriptor size for the proposed descriptors and the well-known Fourier-based shape descriptors including centroid distance and boundary curvature. As seen, the proposed method with $M \geq 4$ outperforms the other methods for all rank values. In the experiments, we keep 10 images for training and 10 images for testing. We have also performed another series of experiments to assess the effect of training size on the recognition rate. The results

are given in Table 4.8 for $\tilde{F}(M=7)$. We present the rank 10 query results in Fig. 4.6 where the first image is the query image and the remaining images are the first 10 images retrieved from the database. We used the same descriptor $\tilde{F}(M=7)$ again in this experiment. It is seen that the recognition rate increases as the training size increases for this data set.

Table 4.7 : Recognition rates for the MPEG-7 Core Experiments Shape data set with respect to rank and descriptor size.

Method	Rank									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Centroid	83.77	87.58	90.69	92.36	93.31	93.79	94.27	94.51	94.74	95.36
Curvature	67.78	72.31	74.94	77.56	79.71	81.62	82.81	84	85.91	87.81
Complex Coor.	84.48	89.26	91.88	92.12	93.31	93.55	95.94	96.18	96.42	96.65
$\tilde{F}(M=2, L=10)$	77.8	82.81	84.96	86.39	87.35	89.22	90.21	90.69	90.93	92.36
$\tilde{F}(M=3, L=10)$	84.96	89.26	90.69	92.84	93.07	93.79	94.27	94.99	95.22	95.22
$\tilde{F}(M=4, L=10)$	88.06	93.55	94.74	95.7	95.94	96.18	96.18	96.42	97.13	97.13
$\tilde{F}(M=5, L=10)$	88.3	93.31	94.03	95.22	95.94	96.42	96.89	97.37	97.85	97.85
$\tilde{F}(M=6, L=10)$	89.26	93.07	95.22	97.61	97.61	97.85	98.09	98.32	98.32	98.32
$\tilde{F}(M=7, L=10)$	90.93	95.22	96.65	97.61	98.09	98.32	98.32	98.32	98.56	98.56
$\tilde{F}(M=8, L=10)$	89.73	93.79	95.94	96.65	96.89	97.61	97.61	98.09	98.32	98.32
$\tilde{F}(M=9, L=10)$	90.69	95.22	96.42	97.13	97.37	98.09	98.56	98.8	98.8	98.8
$\tilde{F}(M=10, L=10)$	89.02	94.98	96.65	97.85	98.09	98.32	98.32	98.32	98.32	98.32
$\tilde{F}(M=11, L=10)$	90.93	94.98	97.13	97.61	98.09	98.09	98.32	98.32	98.32	98.32
$\tilde{F}(M=12, L=10)$	89.97	94.27	96.89	97.61	97.85	97.85	98.09	98.32	98.32	98.56
$\tilde{F}(M=13, L=10)$	90.45	94.03	96.42	97.61	97.85	97.85	98.09	98.56	98.56	98.56
$\tilde{F}(M=14, L=10)$	88.78	93.31	95.22	96.89	97.37	97.61	97.85	97.85	98.32	98.32
$\tilde{F}(M=15, L=10)$	89.73	94.03	95.7	97.13	97.61	97.85	97.85	97.85	98.09	98.09
$\tilde{F}(M=16, L=10)$	89.26	94.27	96.65	97.61	98.32	98.32	98.32	98.32	98.32	98.32
$\tilde{H}(M=2, L=10)$	89.97	94.03	95.22	95.7	95.94	95.94	96.18	96.65	96.89	97.13

Table 4.8 : The effect of training size on recognition rates for the MPEG-7 shape data set.

	Rank

Training Size	1	2	3	4	5	6	7	8	9	10
1	74.15	80.92	84.81	87.57	89.83	91.96	93.47	94.1	94.1	94.47
3	80.36	86.53	90.32	92	92.84	94.24	94.95	95.09	95.51	95.93
5	85.53	91.41	93.64	94.75	95.86	96.5	96.82	96.82	96.97	96.97
7	88.44	92.66	94.31	95.96	96.33	96.88	97.43	97.61	97.61	97.61
9	90.02	95.01	96.52	97.18	98.04	98.04	98.04	98.48	96.89	98.69

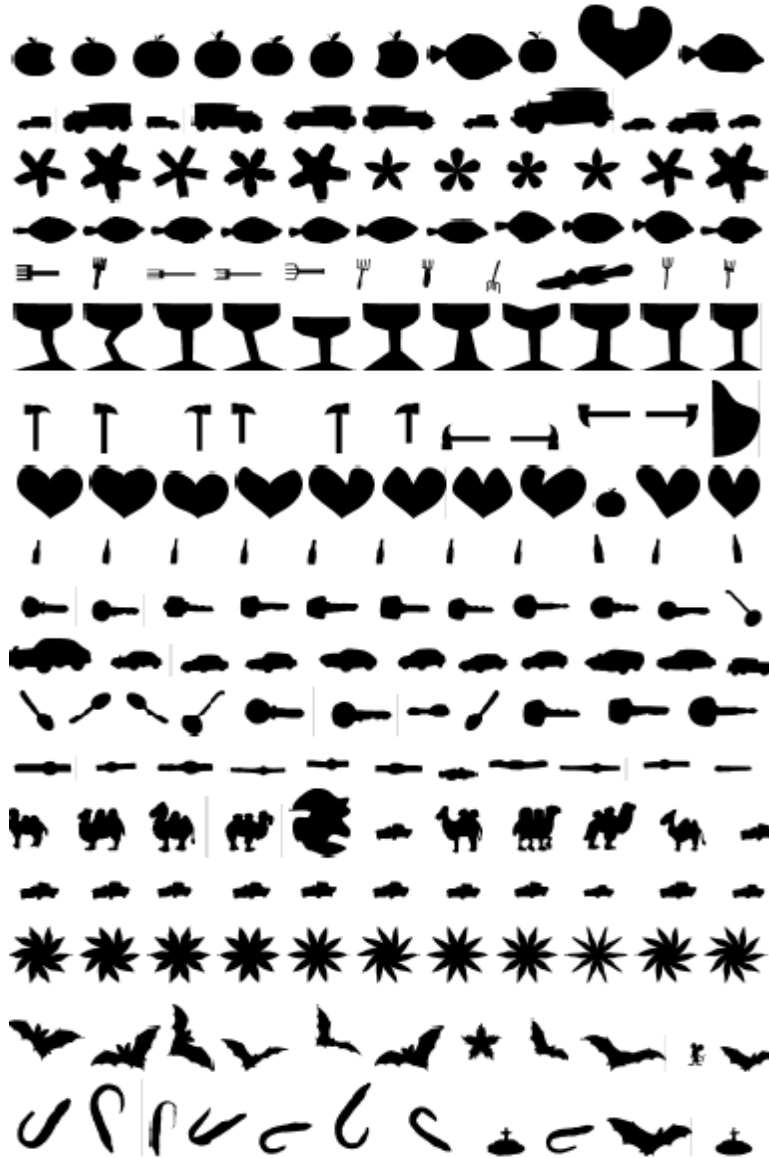**Figure 4.6 :** Content-based image retrieval performance of the proposed method. Each line contains a query where the first image is the query image and the remaining images are the query results.

In addition to these data sets, we evaluate the performance of the proposed descriptors on the Kimia data set [48] containing 99 binary images of 9 objects. The numbers of correctly retrieved objects are given in Table 4.9 for the proposed

descriptors ($\tilde{H}$, $\tilde{F}$) and the other contour-based methods. As seen, the proposed descriptors outperform the other contour-based methods on this database as well.

Table 4.9 : Recognition performance for the Kimia Data Set with respect to rank and descriptor size. Results are given in terms of the number of correctly retrieved shapes.

Method	Rank									
	1	2	3	4	5	6	7	8	9	10
Centroid	84	92	94	94	95	96	96	97	97	98
Curvature	52	62	65	70	75	77	77	79	82	87
Complex Coor.	89	90	93	93	93	93	93	94	96	98
$\tilde{F}(M=2,L=10)$	70	79	83	84	86	91	91	92	94	95
$\tilde{F}(M=3,L=10)$	78	84	87	90	92	92	92	92	93	93
$\tilde{F}(M=4,L=10)$	82	89	92	94	97	98	98	98	98	98
$\tilde{F}(M=5,L=10)$	86	90	93	96	98	98	99	99	99	99
$\tilde{F}(M=6,L=10)$	84	94	95	97	98	98	98	98	98	98
$\tilde{F}(M=7,L=10)$	89	95	97	97	99	99	99	99	99	99
$\tilde{F}(M=8,L=10)$	87	93	94	95	97	97	97	98	98	99
$\tilde{F}(M=9,L=10)$	89	93	93	94	96	98	98	98	98	99
$\tilde{F}(M=10,L=10)$	89	94	94	96	98	98	98	99	99	99
$\tilde{F}(M=11,L=10)$	89	94	94	94	96	97	98	99	99	99
$\tilde{F}(M=12,L=10)$	88	93	96	96	98	98	98	99	99	99
$\tilde{F}(M=13,L=10)$	88	94	95	98	98	98	98	99	99	99
$\tilde{F}(M=14,L=10)$	88	92	94	97	97	97	98	99	99	99
$\tilde{F}(M=15,L=10)$	90	93	95	96	98	98	98	99	99	99
$\tilde{F}(M=16,L=10)$	87	91	95	97	98	98	99	99	99	99
$\tilde{H}(M=2,L=10)$	90	93	95	96	97	97	97	97	97	97

We also studied performance of the descriptors under shearing transformation, which is defined as the following

$$T = \begin{bmatrix} 1 & 0 & 0 \\ \alpha & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
(4. 16)

One digit for each class is selected from the license plate digit database. 64 different shearing transformations obtained by changing the parameter α in [-1..1] are then applied to each digit in the training set and these transformed images are used in testing (see Fig. 4.7).

Figure 4.7 : Images of the digit "5" under different shear transformations.

Recognition performances with respect to varying L are given in Table 4.10 for the proposed descriptors ($\tilde{H}$ and $\tilde{F}$) and the other contour-based methods. The descriptor $\tilde{F}$ is almost invariant under shearing for $L \geq 4$ when compared to $\tilde{H}$ and other descriptors.

Table 4.10 : Recognition rates under shearing transformation with respect to number of coefficients (L).

Method	*Number of Coefficients (L)*						
	2	**3**	**5**	**7**	**10**	**15**	**25**
Centroid	42.12	45.77	51.9	52.57	54.22	54.39	54.39
Curvature	43.94	44.61	59.36	64.17	63.34	62.85	57.37
Complex Coor.	44.61	48.92	68.15	73.46	76.11	74.62	74.29
$\tilde{F}\,(M=9)$	91.21	95.68	99.17	99.66	100	100	100
$\tilde{H}\,(M=2)$	37.18	62.18	76.28	85.07	89.05	92.2	95.35

Finally we demonstrate the ability of our algorithm to deal with occlusion on a database comprised of synthetically occluded objects. The objects are selected from MPEG-7 Shape data set that we used in the previous experiments and the occluded version of an object is obtained by erasing the boundary at various rates (e.g. 1%-20%). The recognition performance for the synthetically generated objects is given in Table 4.11. The proposed descriptor can tolerate occlusion at most 5% where the recognition rate is still greater than 80%. On the other hand, the other methods can tolerate occlusion at most 2%. It is known that boundary-based shape descriptors have limited performance under occlusion. When this is considered, it is a significant improvement. We also present the rank 10 query results in Fig. 8, where the first image is the synthetically produced query image and the remaining images are the first 10 images retrieved from the database. In the experiments, the query objects have occlusion rates of 6% on the average.

Table 4.11 : Recognition rates for the occluded objects that are synthetically produced from the MPEG-7 Core Experiments shape data set at various occlusion rates.

	Occlusion rate (%)

Method	0	1	2	5	8	10	15	20
Centroid	83.77	83.53	82.33	68.97	53.69	43.67	21.47	13.84
Curvature	67.78	67.54	63	48.44	31.74	26.97	13.84	6.66
Complex Coor.	84.48	84.48	83.77	73.5	63.48	57.27	33.17	20.28
$\tilde{F}(M=7)$	90.93	90.69	91.4	82.34	62.52	55.88	34.41	20.76
$\tilde{F}(M=8)$	89.73	89.74	89.49	82.33	59.9	51.79	32.69	19.57
$\tilde{H}(M=2)$	89.97	90.69	90.69	79.71	63.48	51.07	22.43	11.93

Our analyses indicate that the proposed descriptors outperform several commonly used descriptors in the selected binary and grayscale databases. Even though shape matching approaches can achieve higher performance on the Kimia database as reported by [48], with the expense of high matching cost, a cost-efficient invariant descriptors, such as given here, with a high recognition rate will probably be more preferable for most of the content-based retrieval systems.
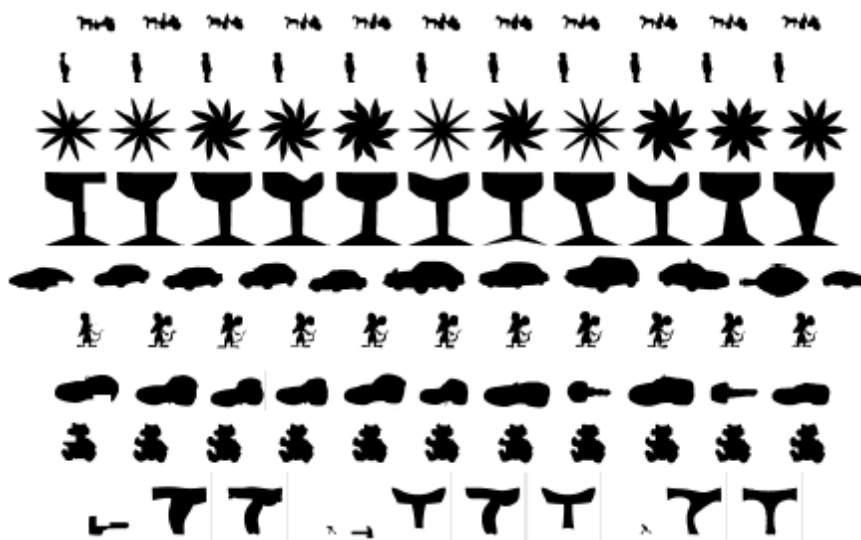
Figure 4.8 : CBIR performance of the proposed method under occlusion. Each line contains a query where the first image is the occluded query image and the remaining images are the query results.

4.5 Discussions

In this study, we present two new gradient based object shape descriptors. The proposed system utilizes not only the boundary point coordinates of the objects, but also the filter responses along the boundaries. We compare the recognition

performance of new shape descriptors with well-known boundary based shape descriptors on the databases including grayscale license plate characters and binary objects. The experimental results show that the proposed descriptors drastically outperform other shape descriptors. Another advantage of the proposed descriptors is that one can use them on both grayscale and binary images whereas other shape descriptors can only be applied to binary objects. The proposed algorithm takes the advantage of local distribution information found in the gradient cue to achieve this high performance. Even though the performance of the proposed descriptors $\left(\tilde{F} \ and \ \tilde{H} \right)$ are comparable, the compactness and the computational efficiency of the descriptor $\tilde{H}$ make this descriptor very attractive for content-based image retrieval systems.

5. SEGMENTATION AND RECOGNITION WITH SHAPE DRIVEN FAST MARCHING METHODS

5.1 Introduction

In this part of the thesis, a Fast Marching – Shape Description integrated system that is capable of both extracting object boundaries and recognizing shapes is represented.

One of the challenges in the field of image segmentation is the incorporation of prior knowledge on the shape of the segmenting contour. Several methods of incorporating prior shape information into object location determination have been developed. In [21] a statistical model of shape variation is established from a set of corresponding points across the training images, and then, a Bayesian formulation based on this prior knowledge and edge information of the image is employed to find the object boundary. In [22] an elliptic Fourier decomposition of the boundary is utilized to incorporate the global shape information into the segmentation.

Integration of statistical shape variation into the level set methods was first proposed by Leventon et.al. [23]. They compute a statistical shape model over a training set of curves implicitly. The segmentation process embeds an initial curve as a level set of a higher dimensional surface and evolves the surface locally based on image gradients and curvature, and globally toward a maximum a posteriori (MAP) estimate of shape and pose. The MAP estimate is computed at each step of the surface evolution, based on the prior shape and the image information. Since training shapes are embedded by the signed distance function dimension of the input space increases drastically and it is unclear in what way the surface representation affects the shape learning, since only the zero level set of the surface corresponds to a perceivable shape.

Several researchers worked on this area [24, 25, 29]. Cremers et al. [26,27] present a variational integration of nonlinear shape statistics into a Mumford–Shah based segmentation process [28]. The nonlinear statistics are derived from a set of training silhouettes by a novel method of density estimation which can be considered as an

extension of kernel PCA to a stochastic framework. They applied the proposed algorithm to find boundaries of specific objects, such as human hand and license plate characters. They presented good results for segmentation of plate characters with nonlinear shape prior statistics. But again only object boundary segmentation is the aim of the study and recognition issues were not concerned by the authors.

Rousson and Paragios [29] developed an approach consisting of two stages. The first stage is for shape modeling, built directly on the level set space using a collection of samples. Then, this model is used as a basis to introduce the shape prior in an energetic form. This prior aims at minimizing the non-stationary distance between the evolving interface and the shape model in terms of their level set representations. The limitations are similar with [23], that is embedding the shapes into a signed distance level set map increases the complexity and. There is indefiniteness about the effects of this shape representation to the shape learning because of the contour is defined only on the zero level sets.

There are main differences between these studies and the proposed system as listed below:

i.   In these studies, the evolving front is always forced to have the prior. However, we stop the front near true object boundaries

ii.   It is stated that, the proposed method in [29] does not work when number of prior object classes is more than one. But our system is capable of segmenting and recognizing different classes of characters.

iii. Previous researchers obtained the shape statistics from the whole map of level set values; however, only the front itself is employed here for shape description.

iv.   Previously proposed systems need high calculation power because they have two optimization stages, one is for minimization of image energies, and the other is for minimizing shape similarity energies. On the other hand, our system has one optimization step for minimizing both energies.
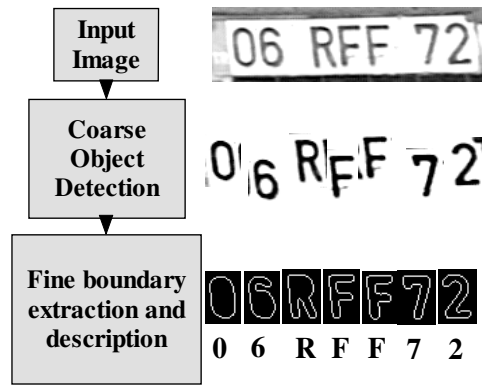
Figure 5.1 : Proposed segmentation – recognition system.

The proposed algorithm consists of two parts as seen in Figure 5.1: Coarse detection of objects and fine boundary extraction with classification. The first part is designed to separate objects individually. The boundaries of the objects need to be extracted and recognized. It is very hard to locate the true object boundaries globally with FM methods. The second part is designed to treat the objects one by one for fine tuning of boundary contour and classification.

5.2 Coarse Object Detection

The aim of this module is to isolate and separate object candidates by each other. A big Fast Marching (FM) front is initialized around the border of the input scene including all interested objects and the front starts shrinking through the objects using image gradient based speed functions (Eq. 3.12). Level Set methods like FM, naturally have very efficient contour splitting and merging abilities. Hence, the active front would be easily split up to capture the individual shapes. Each front segment is processed one by one. The Coarse object detection phase is finalized when a front segment wraps a single object. The next step, "Fine boundary extraction and description", runs individually for each piece.

Deciding whether a front piece includes only one object or not is another issue to be solved. A minimum surrounding rectangle is found for each piece and sizing properties of the rectangle are utilized for determining the number of objects in this region. If the rectangle is too small for an object candidate, then the front piece is removed from the list.

The algorithm for coarse object detection can be summarized as follows:

a) Initialize the front at the borders of the image (see first image in Figure 5.2)

b) Make Fast Marching iterations with speed function in Eq. 3.12.

c) Evolve the front

d) Find 8-connected front pieces, examine size of each piece blob if it can be an object candidate.

e) Start fine extraction and description for these object regions (red color contours in Figure 5.2). Remove the red contours from the FM trial point list.

f) If there is no "object sized" piece go to (a)

g) If there is no remaining front nodes in Fast Marching set, finish iterations.
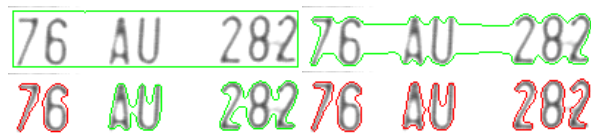
Figure 5.2 : Coarse segmentation algorithm.

5.3 Fine Boundary Extraction and Description

Fast Marching based object detection methods (Section 3) have a main drawback in that, it is difficult to stop the front around true object boundaries. It is an especially challenging task while using image gradient based non-zero speed functions. There are several sources of this problem:

i. The Fast Marching front evolves in only one direction. That means that if the front misses the true boundary location once, it cannot come back again.

ii. The gradient intensities along the shape boundary may not be uniform. That causes different marching time values on boundary points of the same shape. (see Figure 5.3 (a))

iii. There may be edge discontinuities or gaps on the object boundary. It means that some true boundary points may have very small gradient intensities. (see Figure 5.3 (b))

<center>(a)   (b)</center>

Figure 5.3 :   Example shape images for corrupted boundary .
a) Nonuniform gradient levels along boundary points,
b) Shape discontinuties.

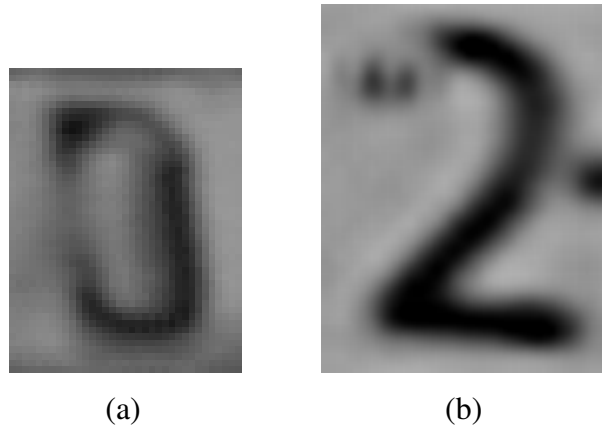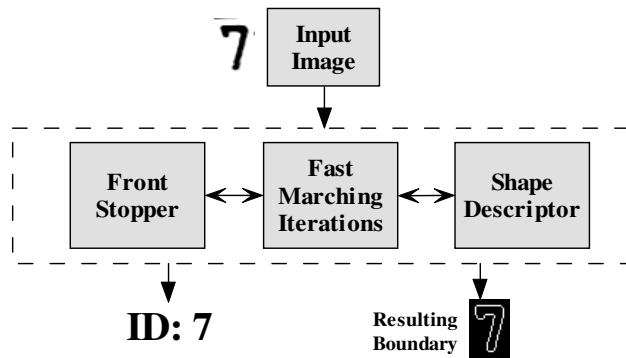**Figure 5.4 :** Fine boundary extraction & recognition phase.

A front stopping and recognition algorithm to solve these problems (Figure 5.4) has been developed. The proposed algorithm uses a fast marching evolved contour and extracts local and global information around the contour. The extracted information is utilized for stopping the surface.

5.3.1 New Speed Formula

The basic objective is simply locating the real object boundaries with an evolving active front. Image gradient information is utilized to model the physical boundaries which are formulated in the FM speed function (Eq. 3.12). This is a common formula which assumes that the gradients spread up homogeneous around the image. This assumption is mostly invalid in the real world, due to the physical corruption of objects and imaging noise. Consider Figure 5.5 that illustrates the gradient of two cross-sections on the same shape boundary. The blue cross-section is on an edge that has big intensity transition while the red one lies on an edge that has small intensity transition. If only the gradient magnitudes are used, the evolving front never passes

<center>53</center>

on these two regions at the same time because of the gradient imbalance on the edge pixels. This is a critical problem of the Level Sets method locating real object boundaries.
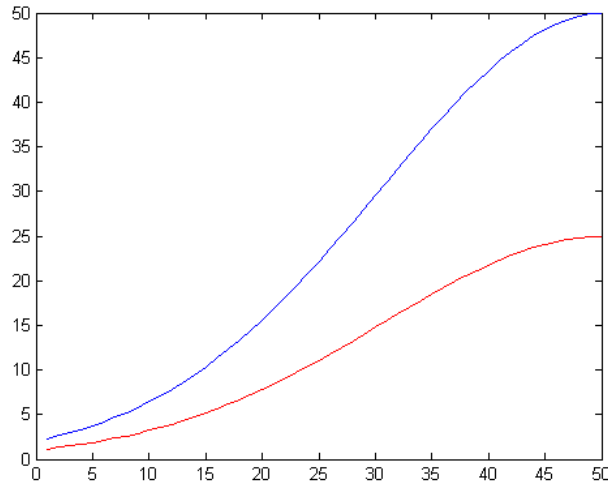
Figure 5.5 : Two gradient cross-sections on an image. y-axis shows gradient magnitude and x-axis shows cross-section pixel location.

Real object boundaries are standing on real edges. Real edge points have not only big gradients but also have small Laplacians (second order derivatives). Perona and Malik [62] utilized first order (gradient) and second order (Laplacians) derivatives of images in their anisotropic diffusion theory for smoothing images while preserving real edges.

In this study, to overcome this limitation, second derivatives (Laplacians) are also embedded into the front speed formula [2]. With the help of Laplacians, the location of gradient peaks, which are probable edges, can be determined. Recall that, Laplacian goes to zero on an optimum edge.

Gradients and Laplacians are utilized to formulate the front speed as

$$F = \begin{cases} \dfrac{\beta\left|\nabla^2 G_\sigma * I\right|}{1+\alpha\left|\nabla G_\sigma * I\right|} & \left|\nabla G_\sigma * I\right| > T \\[4mm] \dfrac{1}{1+\alpha\left|\nabla G_\sigma * I\right|} & Otherwise \end{cases} \qquad \textbf{(5. 1)}$$

where ∇G_σ and $\nabla^2 G_\sigma$ show the gradient and Laplacian operators respectively. Recall that, the formula in the second row of the equation is same as the generalized gradient based speed term in Eq. 3.12. The Laplacian term provides speeding up the

front on the regions with high gradient changes when gradients do not reach peaks. It also slows down the front around the gradient peaks.

The proposed speed formulation shows that the fast marching grid has similar time values on real object boundaries, which is a desirable property.

5.3.2 Local Front Stopping

Contrast differences along shape boundaries (Figure 5.5) prevent the global fitting of the fast marching contour on true shape boundary points. That is because different gradient magnitudes produce different speed values and different speed values cause different arrival time values $T(x, y)$ (see Section 3.2). Lankton and Tannenbaum proposed a localization framework to solve this general problem of active contour methods [61]. They suggested narrowing the energy minimization around neighboring region of interested curve point, instead of global energy minimization. However, energy minimization for each active point is not cost effective and not suitable for Fast Marching method. Then a local node freezing method is proposed to handle these differences and fixing the active contour near the true object boundary points.

A promising local front stopping algorithm which tests each FM trial point whether it should be fixed or not [2] was devised. The fixed front nodes are called "fix trial points", that is a new type of FM points (gray colored points in Figure 5.6). These special points are still on the list of trial points but they have no chance to march anymore.

A trial point should be tagged as "fix" if it is near the real object edges. There are two main properties of a real edge: it has a big gradient magnitude and its gradient is a local maximum toward edge normal [55]. These properties can be adapted into front stopping algorithm as follows: The proposed algorithm tends to freeze the nodes one by one when the node is on a grid with a high-enough gradient (hysteresis thresholding) and when the gradient is a local maximum toward the normal of the evolving curve (Figure 5.7). There is also a smoothness condition to stop the node: If the number of fix trial points in the 8-neighborhood of the node is more than two then it is labeled a fix trial point.

55

5.3.3 FM – Shape Descriptor Integration

Concurrent segmentation and recognition is one of the main contributions of the thesis. Integration of Fast Marching and Shape Description schemes is accomplished during the front stopping part.

Figure 5.6 : Fine segmentation iterations. Black points are the moving trial points and gray points are the fix trial points.
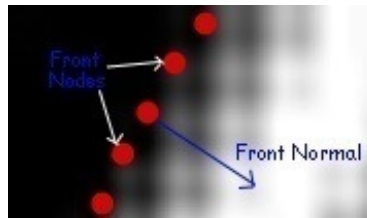
Figure 5.7 : Demonstration of evolving nodes and the front normal on image gradient map.

The proposed front stopping algorithm (Section 5.3.2) gives us the opportunity of threading the Fast Marching front as a true shape boundary. Trial points constitute the front contour and trial points $x_t(k)$ have two types: "fix" trial points $x_f(k)$ and "moving" trial points $x_m(k)$. (Figure 5.6)

$$x_t(k) = x_f(k) \bigcup x_m(k) \tag{5.2}$$

Shape signatures $u(t)$ are obtained by a linear process over the boundary points, then

$$u_t(t) = u_f(t) \bigcup u_m(t). \tag{5.3}$$

Let us use Function 2.9 to obtain the shape descriptors as

$$a_n = \frac{1}{N} \sum_{t=0}^{N-1} u_t(t) \exp\left(\frac{-j2\pi nt}{N}\right)$$

$$= \frac{1}{N}\left[\{U_f\} + \{U_m\}\right] \tag{5. 4}$$

$$= \frac{1}{N}\left[\left\{\sum_{t=0}^{F-1} u_f(t) \exp\left(\frac{-j2\pi nt}{N}\right)\right\} + \left\{\sum_{t=F}^{N-1} u_m(t) \exp\left(\frac{-j2\pi nt}{N}\right)\right\}\right]$$

where F and N show the number of fix trial points and all trial points.

The formulation in Eq. 5.4 has the advantage that calculation complexity is decreased since the fix term is stable and does not change rapidly. New fix points are simply added to U_f and calculation for old fix points is not repeated.

Another point is determining the starting time of shape description as the front is getting closer to the real shape boundary. The number of the fix trial points is a good parameter to start description. Description is needed to make near true object boundary and fix trial points are natural courier for real object edges. The shape description process is started when the number of fix trial points is greater than the number of moving points.

5.3.4 Classification with Fusion

Recognition methods mostly suffered from segmentation errors, which are especially important for boundary-based classifiers. A general segmentation system produces only one result for a single shape; so there is only one possibility to recognize the object. However, the proposed FM based object segmentation and recognition technique is able to create more than one object boundary outcome, which can be utilized to improve classification performance [2]. It is also useful while working with a limited number of train samples. The number of train samples can be increased with utilizing different boundary contours for the same shape.

The classifier flow chart is sketched at Figure 5.8. Description starts when number of fix points F exceeds the number that is half of the number of trial points N. There are two conditions to end FM iterations: i) all trial points are fixed and ii) the classification confidence is high enough.

Consider M shape descriptor vectors $\left[a^0, a^1, ..., a^{M-1}\right]$, M decision results $\left[r^0, r^1, ..., r^{M-1}\right]$ and M confidence values $\left[c^0, c^1, ..., c^{M-1}\right]$, $0 < c^i < 1$ have been collected at the end of FM iterations .
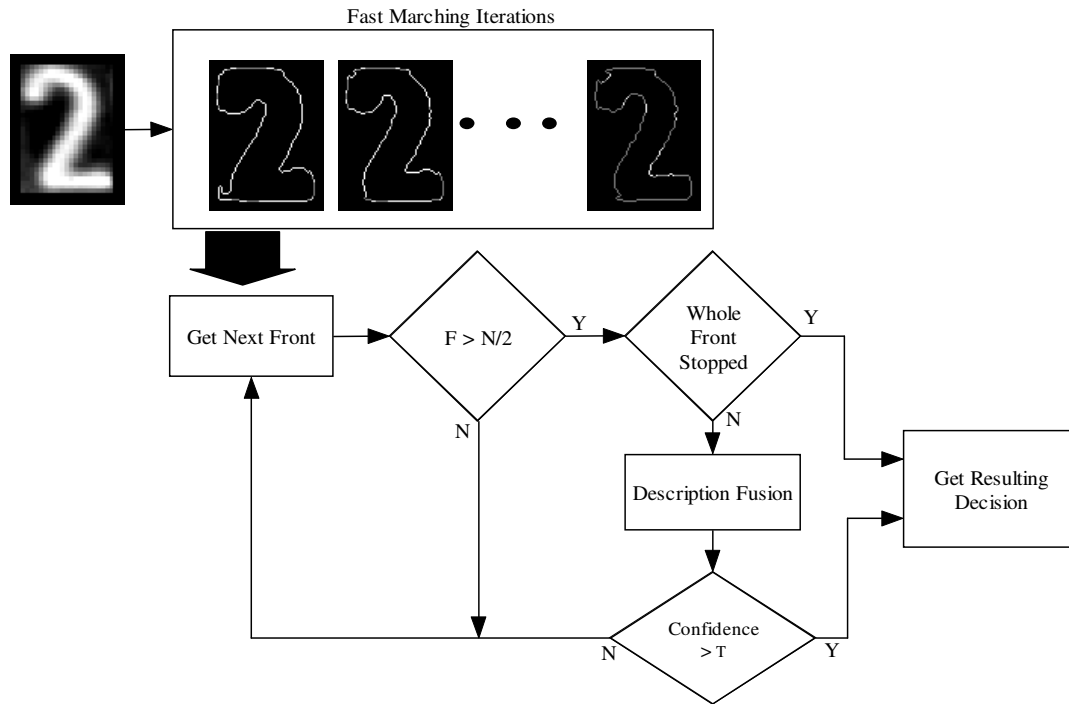
Figure 5.8 : Proposed Classification Scheme.

"Description fusion" is the base step that combines the iterated front data together. These shape decisions are fused with decision voting which constructs the decision label histogram of r^i and selects the maximum number as a final decision (majority voting). The experimental works on classification and performance reports can be found in Section 5.4. The 1 - nearest neighbor method is selected as a classifier.

5.4 Experimental Work and Results

The two main aims of the proposed system are to capture the object boundaries and recognize the objects. The license plate character dataset is utilized to measure the recognition performance of the proposed concurrent segmentation – recognition system. The Fast Marching front stopping algorithm (Section 5.3.2) is applied to prepare the boundary contours in the training and testing set. Evolving boundary contours are collected through FM iterations. There are a large number of contours for each character (Figure 5.9); therefore, a certain number of contours need to be selected. Naturally, the contour samples, which include larger number of fix trial points (Section 5.3.2), tend to have higher recognition performance. For that reason, the selection starts when the fix point percentage exceeds 50% of the trial points and

continues until global front freeze. Ten contour samples have been selected uniformly in this defined evolving period for each character.

Figure 5.9 : Evolving object boundary contour samples.

Some example characters and their selected 10 contours are illustrated in Figure 5.10 In traditional classification schemes, there is only one opportunity to classify a pattern in contrast to proposed system. There are 10 samples for each pattern in training and testing. Table 5.1 exposes the recognition rates with a single training sample and a single testing sample for each character pattern as in traditional methods.
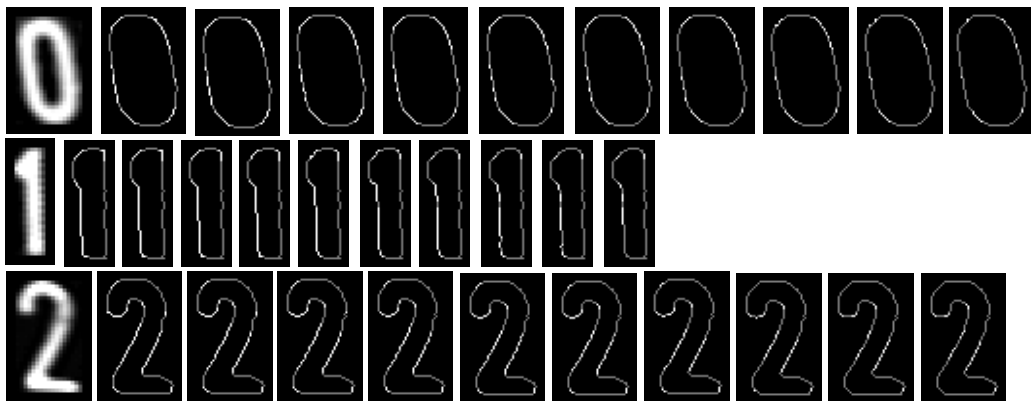

Figure 5.10 : 10 Selected boundary contours for three different license plate character.

Let i and j indicates the selected sample index of training and testing patterns respectively. In Table 5.1 it can be easily seen that success rates increase while $|i - j|$ decreases. As can be seen in the table, rates are bigger along the left diagonals, where $|i - j|$ is near zero.

Table 5.1 : Recognition rates when using single train and single test samples for each character. Numbers 1,2,..,10 indicates the selection order index (See Fig. 5.10).

Train index	Test index									
	10	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
10	**96.82**	96.67	95.96	95.28	94.08	91.29	88.38	86.52	85.92	84.75
9	96.8	**96.77**	96.16	95.81	94.29	91.83	89.39	87.11	86.45	85.43
8	96.7	96.34	**96.22**	95.83	94.95	92.87	90	87.03	85.94	85.48
7	96.42	96.37	96.04	95.86	95.05	93.4	91.67	89.26	88.02	87.34
6	96.42	96.42	96.14	**95.91**	95.78	94.82	93.15	90.56	89.21	88.48
5	96.6	96.27	96.19	96.09	**96.01**	95.86	94.85	93.25	91.29	90.81
4	96.29	96.22	96.11	96.06	95.89	**95.99**	95.94	95.33	94.16	93.53
3	96.29	96.09	95.89	95.76	95.81	95.56	**96.19**	**95.86**	95.81	95.73
2	96.39	96.11	95.96	95.37	95.02	94.19	94.64	95.38	**95.94**	95.91
1	96.39	96.06	95.81	95.3	94.67	93.5	93.04	94.19	95.91	**96.09**

Another result, extracted from Table 5.1, is that recognition rates are better for bigger sample indexes i and j. That is because these sample contours have more fix nodes than others, so they are near the real object boundaries.

Now there is more than one sample for each pattern both in training and testing set.s Therefore these information sources need to be merged. This is accomplished by three test mechanisms as follows:

- This test is to measure the effect of the augmented train set with evolving contour samples. All 10 training samples were utilized and test is made on single test samples. Recognition rates are shown at Table 5.2. Recognition performance is obviously better than the results in Table 5.1, which means that the training set augmentation property of the proposed technique increases the success rate.

Table 5.2 : Testing with augmented training set.

Train	Test index									
	10	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
Use All	97.08	96.87	96.7	96.67	96.62	96.42	96.75	96.42	96.47	96.49

- The second test is to show the advantage of test set augmentation. We use single training samples and test on whole 10 testing samples. A character is accepted to be classified right when any sample is right. Recognition rates are listed in Table

5.3. Notice that, this type of test cannot be used in real problems, it is only for demonstrating the classification power in these augmented test samples.

Table 5.3 : Testing with augmented testing set.

	Train index									
Test	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
Any true	98.75	98.8	98.68	98.5	98.5	98.65	98.65	98.65	98.63	98.63

- The third test is actually is a special combination of the first two tests. In this test, each test sample classification result is threaded as a different source of decision, and a simple decision voting is accomplished. Notice that, this type of test can be used in any real world classification problem. The recognition performance has been investigated while changing the sample number for each training and testing pattern (Table 5.4). Naturally, the performance is getting better while the sample number is increasing. A recognition rate of 97.56% is obtained when all 10 samples are combined. Notice that, the results in this voting test is much better than the single sample number test in Table 5.1, which reaches 96.82 % at maximum. It proves that the proposed segmentation and recognition system is more successful than the traditional single sample recognition systems.

Table 5.4 : Recognition rates for using majority voting.

	Sample Number (Both in train & test set)								
Test	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**
Voting	97.56	97.51	97.28	97.2	96.98	97.13	96.77	96.82	96.8

Segmentation of broken characters is one of the most difficult tasks for any character recognition system. The proposed system had a promising performance on this problem thanks to the proposed front stopping algorithm (Section 5.3.2) using global shape constrains and preserving the smoothness. Figure 5.11 represents some examples of broken license plate characters and the segmentation results.

Figure 5.11 : Segmentation results on broken characters.

Another problem of the character segmentation is degradation of the images by noise or physical effects. As an example, paints or mugs on the license plate surface cause such problems. Segmentation results of the Canny Edge Detection [55] method and the proposed method are illustrated in Figure 5.12. As seen in the results, our system can capture the object boundaries in spite of the noise around the characters.
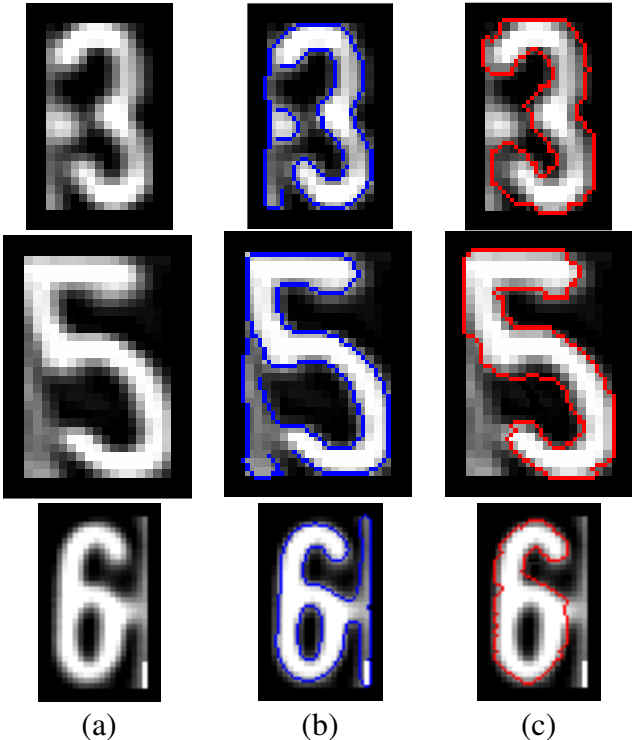

| (a) | (b) | (c) |

Figure 5.12 : Segmentation result on corrupted characters, a) input images, b) Canny edge detection results, c) results of the proposed system.

6. CONCLUSION AND RECOMMENDATIONS

Shape identification systems mostly suffer from the object segmentation problems in the real world. Moreover, segmentation errors are more dangerous for boundary based shape recognizers than for other type of recognizers. This study proposes a new approach to set up a joint shape segmentation and identification system, which is based on image gradient information.

The system utilizes the variational Fast Marching active contours to capture the object boundaries. A new Laplacian based speed function and a promising local front stopping algorithm is proposed as contributions of the thesis. In this study, shape description is defined as a continuous process while the active contour closes up to real object boundaries. On the contrary of traditional shape identification systems, the proposed scheme allows multiple shape recognition chances. These recognition results are combined with the majority voting decision fusion method to increase recognition reliability. Experimental results in Section 5.4 show that the proposed decision fusion result is more successful than any single recognition result. This new idea can be employed for any active contour based shape capturing systems.

The most critical contribution of the thesis is the new boundary based shape descriptors, called Gradient Based Shape Descriptors (GBDS) [1]. The descriptors utilize the local image gradient based features along the shape boundaries as shape signatures. The shape descriptors are defined as Discrete Fourier Transform (DFT) of the shape signatures. GBSD can be applied to both binary and gray-scale shape images. It is experimentally proved that GBSD is less sensitive to miss-localizations of shape boundaries since it uses not only the boundary coordinates but also the filter responses along the boundary contour. The recognition performance of new gradient based shape descriptors are compared with other well-known boundary based shape descriptors (centroid distance shape signature, curvature shape signature, complex coordinate shape signature) on the databases including gray-scale license plate characters and binary objects. The experimental results show that the proposed gradient-based shape descriptors drastically outperform other boundary-based shape

descriptors. Another advantage of the proposed descriptors is that one can use them on both gray-scale and binary images whereas other shape descriptors can only be applied to binary images of objects. The proposed algorithm takes the advantage of local distribution information found in the gradient cue to achieve this high performance. Even though the performance of the proposed descriptors (F and H) is comparable, the compactness and the computational efficiency of the descriptor make this descriptor very attractive for content-based image retrieval systems.

REFERENCES

[1] **Çapar, A., Kurt, B., and Gökmen, M.,** 2009: Gradient-based Shape Descriptors, *Machine Vision and Applications*, Vol. 20, no. 6, pp. 365-378.

[2] **Çapar, A., and Gökmen, M.,** 2009: Shape Recognition by Voting on Fast Marching Iterations, *Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS)*, pp. 379-388.

[3] **Sangwine, S.J., and Horne, R.E.N.,** 1998: *The Color Image Processing Handbook*, Chapman & Hall, London.

[4] **Fogel, I., and Sagi, D.,** 1989: Gabor Filters as Texture Discriminator, *Biological Cybernetics*, Vol. 61, no. 2, pp. 103-113.

[5] **Jain, A.K., and Farrokhnia, F.,** 1991: Unsupervised texture segmentation using Gabor filters, *Pattern Recognition*, Vol. 24, no. 12, pp. 1167-1186.

[6] **Randen, T., and Husoy, J.H.,** 1999: Filtering for texture classification: a comparative study, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 21, no. 4, pp. 291-310.

[7] **Jain, A.K., and Karu, K.,** 1996: Learning texture discrimination masks, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 18, no. 2, pp. 195-205.

[8] **Kim, K.I., Jung, K.C., Park, S.H., and Kim, H.J.,** 2002: Support vector machines for texture classification, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 24, no. 11, pp. 1542-1550.

[9] **Chen, J., Pappas, T.N., Mojsilovic, A., and Rogowitz, B.E.,** 2005: Adaptive perceptual color–texture image segmentation, *IEEE Transactions on Image Processing*, Vol. 14, no. 10, pp. 1524-1536.

[10] **Deng, Y., and Manjunath, B.S.,** 2001: Unsupervised segmentation of color-texture regions in images and video, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 23, no. 8, pp. 800-810.

[11] **Mirmehdi, M., and Petrou, M.,** 2000: Segmentation of color textures, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 22, no. 2, pp. 142-159.

[12] **Shafarenko, L., Petrou, M., and Kittler, J.,** 1997: Automatic watershed segmentation of randomly textured color images, *IEEE Transactions on Image Processing*, Vol. 6, no. 11, pp. 1530-1544.

[13] **Chanussot, J., and Lambert, P.,** 1999: Watershed approaches for color image segmentation, *Proceedings of IEEE Workshop on Nonlinear Signal and Image Processing*, pp. 129-133.

[14] **Ma, W., and Manjunath, B.S.,** 2000: Edge Flow: A technique for boundary detection and image segmentation, *IEEE Transactions on Image Processing*, Vol. 9, no. 8, pp. 1375-1388.

[15] **Kato, Z., and Pong, T.C.,** 2006: A Markov random field image segmentation model for color textured images, *Image and Vision Computing*, Vol. 24, no. 10, pp. 1103-1114.

[16] **Panjwani, D.K., and Healey, G.,** 1995: Markov random field models for unsupervised segmentation of textured color images, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 17, no. 10, pp. 939-954.

[17] **Sethian, J.A.,** 1996: A Fast Marching level set method for monotonically advancing fronts, *Proceedings of the National Academy of Sciences*, Vol. 93, no. 4, pp. 1591-1595.

[18] **Osher, S., and Sethian, J.A.,** 1988: Fronts propagating with curvature-dependent speed: Algorithms based on the Hamilton-Jacobi formulation, *Journal of Computational Physics*, Vol. 79, no. 1, pp. 12-49.

[19] **Costa, L. F., and Cesar, Jr. R. M.,** 2001: *Shape Analysis And Classification: Theory And Practice*, CRC Press, New York.

[20] **Zhang, D., and Lu, G.,** 2005: Study and evaluation of different Fourier methods for image retrieval, *Image and Vision Computing*, Vol. 23, no. 1, pp. 33-49.

[21] **Wang, Y., and Staib, L.,** 1998: Boundary finding with correspondence using statistical shape models, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 338-345.

[22] **Staib, L., and Duncan, J.,** 1992: Boundary finding with parametrically deformable contour methods, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 14, no. 11, pp. 1061-1075.

[23] **Leventon, M., Grimson, W., and Faugeras, O.,** 2000: Statistical Shape Infuence in Geodesic Active Contours, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 316-323.

[24] **Chen, Y., Tagare, H., Thiruvenkadam, S., and Huang, F.,** 1992: Boundary finding with parametrically deformable contour methods, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 14, no. 11, pp. 1061-1075.

[25] **Gastaud, M., Barlaud, M., and Aubert, G.,** 2004: Combining Shape Prior and Statistical Features for Active Contour Segmentation, *IEEE Transactions On Circuits And Systems for Video Technology*, Vol. 14, no. 5, pp. 726-734.

[26] **Cremers, D., Tischhaauser, F., Weickert, J., and Schnörr, C.,** 2002: Diffusion Snakes: Introducing Statistical Shape Knowledge into the Mumford-Shah Functional, *International Journal of Computer Vision*, Vol. 50, no. 3, pp. 295-313.

[27] **Cremers, D., Kohlberger, T., and Schnörr, C.,** 2003: Shape Statistics in Kernel Space for Variational Image Segmentation, *Pattern Recognition In Kernel and Subspace Methods for Computer Vision*, Vol. 36, no. 9, pp. 1926-1943.

[28] **Mumford, D., and Shah, J.,** 1989: Optimal approximations by piecewise smooth functions and associated variational problems, *Communications on Pure and Applied Mathematics*, Vol. 42, no. 5, pp. 575-685.

[29] **Paragios, N., and Rousson, M.,** 2002: Shape Priors for Level Set Representations, *Proceedings of the Seventh European Conference of Computer Vision*, pp. 78-93.

[30] **Veltkamp, R., and Hagedoorn, M.,** 1999: State-of-the-art in Shape Matching, *Technical Report* UU-CS-1999.

[31] **Zhang, D., and Lu, G.,** 2004: Review of shape representation and description techniques, *Pattern Recognition*, Vol. 37, no. 1, pp. 1-19.

[32] **Zhang, D., and Lu, G.,** 2003: A comparative study of curvature scale space and Fourier descriptors for shape-based image retrieval, *Journal of Visual Communication and Image Representation*, Vol. 14, no. 1, pp. 39-57.

[33] **Rafiei, D., and Mendelzon, A. O.,** 2002: Efficient retrieval of similar shapes, *The International Journal on Very Large Data Bases*, Vol. 11, no. 1, pp. 17-27.

[34] **Phokharatkul, P., Kimpan, C.,** 2002: Handwritten Thai Character Recognition Using Fourier Descriptors and Genetic Neural Network, *Computational Intelligence*, Vol. 18, no. 3, pp. 270-293.

[35] **Granlund, G.,** 1972: Fourier preprocessing for hand print character recognition, *IEEE Transactions on Computers*, Vol. 21, no. 2, pp. 195-201.

[36] **Nabout, A.A., and Tibken, B.,** 2008: Object Shape Description Using Haar-Wavelet Functions, *Proceedings of International Conference on Information and Communication Technologies*, pp. 1-6.

[37] **Wunsch, P., Laine, A.,** 1995: Wavelet descriptors for multi-resolution recognition of hand-printed characters, *Pattern Recognition*, Vol. 28, no. 8, pp. 1237-1249.

[38] **Kass, M., Witkin, A., Terzopoulos, D.,** 1987: Snakes: Active contour models, *International Journal of Computer Vision*, Vol. 1, no. 4, pp. 321-331.

[39] **Terzopoulos, D.,** 2003: Deformable Models: Classic, Topology-Adaptive and Generalized Formulations, In Geometric Level Set Methods in Imaging, *Vision and Graphics*, Springer-Verlag, New York.

[40] **Chan, T.F., and Vese, L.A.,** 2001: Active Contours Without Edges, *IEEE Transactions on Image Processing*, Vol. 10, no. 2, pp. 266-277.

[41] **Kulkarni, S., Chatterji, B.N.,** 2002: Accurate shape modelling with front propagation using adaptive level sets, *Pattern Recognition Letters*, Vol. 23, no. 13, pp. 1559-1568.

[42] **Osher, S., and Sethian, J.,** 1988: Fronts propagating with curvature-dependent speed: Algorithms based on the Hamilton-Jacobi formulation, *Journal of Computational Physics*, Vol. 79, no. 1, pp. 12-49.

[43] **Osher, S.,** 2003: Level Set Methods, In Geometric Level Set Methods in Imaging, *Vision and Graphics*, Springer-Verlag, New York.

[44] **Sethian, J.A.,** 1997: Adaptive Fast Marching And Level Set Methods For Propagating Interfaces, *Acta Mathematics Universal Comenianae*, Vol. 67, no. 1, pp. 3-15.

[45] **Malladi, R., Sethian, J.A., Vemuri, B.C.,** 1995: Shape modelling with front propagation: a level set approach, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 17, no. 2, pp. 158-175.

[46] **Huttenlocher, D.P., Klanderman, G.A., Rucklidge,W.A.,** 1993: Comparing images using the hausdorff distance, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 15, no. 9, pp. 850-863.

[47] **Belongie, S., Malik, J., Puzicha, J.,** 2002: Shape matching and object recognition using shape contexts, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 24, no. 24, pp. 509-522.

[48] **Zhuowen, T., Alan, Y.,** 2004: Shape Matching and Recognition Using Generative Models and Informative Features, *Proceedings of 8th European Conference on Computer Vision*, pp. 195-209.

[49] **Yokono, J., Poggio, T.,** 2004: Oriented filters for object recognition: An empirical study, *Proceedings of 6th IEEE International Conference of Automatic Face Gesture Recognition*, pp. 755-760.

[50] **Freeman, W.T., Adelson, E.H.,** 1991: The Design and Use of Steerable Filters, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 13, no. 9, pp. 891-906.

[51] **Balard, D.H., Wixson, L.E.,** 1993: Object recognition using steerable filters at multiple scales, *Proceedings of IEEE Workshop on Qualitative Vision*, pp. 2-10.

[52] **Talleux, S., Tavsanoglu, V., Tufan, E.,** 1998: Handwritten character recognition using steerable filters and neural networks, *Proceedings of IEEE International Symposium on Circuits and Systems*, pp. 341-344.

[53] **Li, S., Shawe-Taylor, J.,** 2005: Comparison and fusion of multiresolution features for texture classification, *Pattern Recognition Letters*, Vol. 26, no. 5, pp. 633-638.

[54] **Gökmen, M., Jain, A.K.,** 1997: $\lambda\tau$-Space representation of images and generalized edge detection, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 19, no. 6, pp. 545-563.

[55] **Canny, J.F.,** 1986: A computational approach to edge detection, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 8, no. 6, pp. 112-131.

[56] **Gonzalez, R.C., Woods, R.E.,** 2002: *Digital Image Processing 2nd Edition*, Prentice-Hall, Englewood Cliffs.

[57] **Sharvit, D., Chan, J., Tek,H., Kimia, B.,** 1998: Symmetry-based indexing of image database, *Journal of Visual Communication and Image Representation*, Vol. 9, no. 4, pp. 366-380.

[58] **Çapar, A., Gökmen, M.,** 2006: Concurrent segmentation and recognition with shape-driven fast marching methods, *Proceedings of 18th International Conference on Pattern Recognition (ICPR)*, pp. 155-158.

[59] **Cremers, D., Rousson, M., and Deriche, R.,** 2007: A Review of Statistical Approaches to Level Set Segmentation: Integrating Color, Texture, Motion and Shape, *International Journal of Computer Vision*, Vol. 72, no. 2, pp. 195-215.

[60] **Ayed, I.B., Li, S., and Ross, I.,** 2009: A Statistical Overlap Prior for Variational Image Segmentation, *International Journal of Computer Vision*, Vol. 85, no. 1, pp. 115-132.

[61] **Lankton, S., Tannenbaum, A.,** 2008: Localizing Region-Based Active Contours, *IEEE Transactions on Image Processing*, Vol. 17, no. 11, pp. 2029-2039.

[62] **Perona, P., Malik, J.,** 1990: Scale-space and edge detection using anisotropic diffusion, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 12, no. 7, pp. 629-639.

APPENDICES

APPENDIX A.1 : Proof of Circular-Shifting Property in Fourier domain

APPENDIX A.1

Assume that the descriptor obtained for an object is denoted as $f_{k,m}$ and the descriptor obtained for the α degrees rotated form is denoted as $f'_{k,m}$. By the definition of circular-shifting property the descriptors are related to each other as

$$f'_{k,(m+s)\bmod M} = f_{k,m} \tag{A. 1}$$

Assuming that the rotation angle is exact multiple of $\dfrac{\pi}{M}$, namely $\alpha = s\dfrac{\pi}{M}$. Taking the Fourier transform of the descriptors ($f_{k,m}$ and $f'_{k,m}$) yields

$$\tilde{f}_{k,m} = \frac{1}{N}\sum_{n=0}^{N-1} f_{n,m} \exp\left(\frac{-j2\pi kn}{N}\right) \tag{A. 2}$$

$$\tilde{f}'_{k,m} = \frac{1}{N}\sum_{n=0}^{N-1} f'_{n,m} \exp\left(\frac{-j2\pi kn}{N}\right) \tag{A. 3}$$

Substituting (Eq. A.1) into (Eq. A.2) results in

$$\tilde{f}_{k,m} = \frac{1}{N}\sum_{n=0}^{N-1} f'_{n,(m+s)} \bmod M \exp\left(\frac{-j2\pi kn}{N}\right) \tag{A. 4}$$

$$\tilde{f}_{k,m} = \tilde{f}'_{k,(m+s)} \bmod M$$

Which means that circular-shifting applies also to $\tilde{f}_{k,m}$ as well as to $f_{k,m}$.

CURRICULUM VITA

Candidate's full name: Abdulkerim ÇAPAR

Place and date of birth: Kadirli 10.12.1976

Permanent Address: ITU Institute of Informatics, Ayazaga, Istanbul

Universities and Colleges attended:

2001 – 2010 ITU Computer Engineering Department PHd.

1998 – 2001 ITU Electronics - Telecommunication Department MSc.

1993 – 1998 Cukurova University Electrical and Electronics Engineering Department BSc.

Publications:

1. Çapar A., Kurt B., Gökmen M., "Gradient-based shape descriptors", Machine Vision and Applications, Springer, Vol. 20, no. 6, pp. 365-378., 2009

2. Çapar, A., and Gökmen, M., Shape Recognition by Voting on Fast Marching Iterations, Advanced Concepts for Intelligent Vision Systems (ACIVS), Bordeaux, France, Lecture Notes in Computer Science, Vol. 5807, pp. 379-388, 2009.

3. Çapar A., Gökmen M., "Concurrent Segmentation and Recognition with Shape-Driven Fast Marching Methods", International Conference on Pattern Recognition ICPR'06, Hong Kong, pp. 155-158, 2006

4. Çapar A., Kurt B., Gökmen M., "Affine Invariant Gradient Based Shape Descriptor", International Workshop on Multimedia Content Representation, Classification and Security (IWMCRS), Istanbul, Turkiye, Lecture Notes in Computer Science, Vol. 4105, pp. 514-521, 2006.

5. Çapar A., M. Gökmen, "Object Region Modeling with Eigen-Density, SVM and MRFs", Image and Vision Computing New Zealand IVCNZ'05, Dunedin, New Zealand 2005

6. Abdulkerim Çapar, Muhittin Gökmen, "Öz-Uzay Dağılım Enerjili Markov Rasgele Alanlarıyla Nesne Yeri Saptama", 13. IEEE Conference on Signal Processing and Its ApplicationsIEEE SIU 2005 Kayseri

7. Fatih Kahraman, Abdulkerim Çapar, Alper Ayvacı, Hakan Demirel, Muhittin Gökmen "Elyazısı Verileri Üzerinde YSA ve DVM' nin Sınıflandırma Başarımlarının Karşılaştırılması" 12. IEEE Conference on Signal Processing and Its Applications IEEE SIU 2004, Kuşadası pp. 615-619.

8. Abdulkerim Çapar, Alper Ayvacı, Fatih Kahraman, Hakan Demirel, Muhittin Gökmen "El Yazısı Tanıma Sistemlerinde Doğrusal Boyut İndirgeme Yöntemleri" 12. IEEE Conference on Signal Processing and Its ApplicationsIEEE SIU 2004, Kuşadası pp. 611-615.

9. Abdulkerim Çapar, Kadim Taşdemir, Özlem Kılıç, Muhittin Gökmen, A Turkish Handprint Character Recognition System, ISCIS, Antalya, Turkiye, Lecture Notes in Computer Science, Vol. 2869, pp. 447-156, 2003.

10. Abdulkerim Çapar, Muhammet S. Beratoğlu, Kadim Taşdemir, Özlem Kılıç, Muhittin Gökmen, "İTÜ Araç Plaka Tanıma Sistemi", 11. Conference on Signal Processing and Its Applications SIU 2003 İstanbul pp. 371-374.

11. Abdulkerim Çapar, Muhittin Gökmen, "Yapay Sinir Ağları Temelli Araç Plaka Yeri Saptama"i 11. Conference on Signal Processing and Its Applications SIU 2003, İstanbul pp. 383-386.

12. Abdulkerim Çapar, Yaşar Erenler, "İstatiksel Araç Plaka Doğrulama-Düzeltme Sistemi", 11. Conference on Signal Processing and Its Applications SIU 2003, İstanbul pp. 817-820.

13. Abdulkerim Çapar,Muhittin Gökmen, "Öznitelik Çıkarma Yöntemlerinin Elyazısı Tanıma Sistemleri Üzerindeki Etkileri" 10. Conference on Signal Processing and Its Applications SIU 2002 Pamukkale pp.530-535.

14. Abdulkerim Çapar, Özlem Kılıç, Kadim Taşdemir, Muhammet Beratoğlu, Muhittin Gökmen "Elyazısı Tanıma Sistemlerinde Öznitelik Çıkarma Ve Sınıflandırma Yöntemlerinin Karşılaştırması" URSI TÜRKİYE 2002 pp. 210-213

15. Abdulkerim Çapar, İlhan K. Yalçın, Muhittin Gökmen "Çeklerdeki Manyetik Mürekkep Karakter Satırının Bulunması ve Tanınması", 10. Conference on Signal Processing and Its Applications SIU 2001 Gazimagosa