DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

HAND GESTURE RECOGNITION

by
Bekir CAN

November, 2012
İZMİR

HAND GESTURE RECOGNITION

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science
in Electrical and Electronics Engineering

by

Bekir CAN

November, 2012

İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"HAND GESTURE RECOGNITION"** completed by **BEKİR CAN** under supervision of **ASST. PROF. DR. METEHAN MAKİNACI** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

ASST. PROF. DR. METEHAN MAKİNACI

Supervisor

Asst. Prof. Dr. Müntaz YILMAZ

(Jury Member)

Asst. Prof. Dr. Özgür TAMER

(Jury Member)

Prof.Dr. Mustafa SABUNCU

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

I would like to thank my adviser Asst. Prof. Dr. Metehan MAKİNACI for his support throughout my master study. His guidance and experiences have expanded my vision.

I also thank to my parents Raif CAN and Sevim CAN for their support and understanding. I thank my elder brother Mustafa for programming tips and advices. I thank my cousins Ayşe CAN and Hüseyin CAN for their contribution to the hand gesture database.

I dedicate this master thesis to my newborn niece Zeynep CAN.

Bekir CAN

HAND GESTURE RECOGNITION

ABSTRACT

In this master study, the purpose is to classify different hand gestures in our database. The database consists of 6 types of gesture class and each gesture class has 50 hand images. Each type of gesture symbolizes a number from 0 to 5. The hand gesture recognition system consists of four main stages: Image enhancement, segmentation, feature extraction and classification. In the image enhancement stage, median filter is used to get rid of high frequency components. After the image enhancement stage, hand region in the image needs to be separated for the next stage. In order to extract hand data from the image, regions that are similar to skin color are found using a color threshold process, and then contour data of the hand which will represent the hand region is selected by finding the longest inner contour of the longest outer contour in the existing regions that are similar to skin color. In the feature extraction stage, useful features for the classification stage are obtained using of the shape features such as convexity defects of the contour. Classifier of the system consists of simple conditional expressions and intersection arc. Depending on the features, the classifier decides which gesture corresponds to the input of the system. The system has a ninety nine percent success rate.

Keywords: Hand gesture recognition, computer vision, digital image processing

EL İŞARETİ TANIMA

ÖZ

Bu yüksek lisans çalışmasında amaç veri tabanımızdaki farklı el işaretlerinin sınıflandırılmasıdır. Veri tabanı 6 çeşit el işareti sınıfından oluşmaktadır ve her bir işaret sınıfı 50 el görüntüsüne sahiptir. Her bir işaret 0 'dan 5 'e kadar bir sayıyı simgelemektedir. El işareti tanıma sistemi dört ana kısımdan oluşmaktadır: Görüntü geliştirme, bölütleme, öznitelik çıkarma ve sınıflandırma. Görüntü geliştirme kısmında, median filtre yüksek frekanslı bileşenlerden kurtulmak için kullanılır. Görüntü geliştirme kısmından sonra, görüntüdeki el alanı sonraki kısım için ayrılması gerekmektedir. Görüntüden el bilgisini çıkartmak için, el cildi benzeri bölgeler renkli eşikleme işlemi kullanılarak bulunur ve el bölümünü temsil edecek elin kontür bilgisi mevcut el cildi benzeri bölgelerde en uzun dış konturün en uzun iç konturü seçilerek bulunur. Öznitelik çıkarma kısmında, sınıflandırma için işe yarar öznitelikler konturün dışbükeylik defekleri gibi biçim özellikleri kullanılarak elde edilir. Sistemin sınıflandırıcısı basit koşulsal ifadeler ve kesişim yayından oluşur. Özniteliklere bağlı olarak sınıflandırıcı sistemin girişiyle hangi işaretin uyuştuğuna karar verir. Sistem yüzde doksan dokuz başarı oranına sahiptir.

Anahtar sözcükler: El işareti tanıma, bilgisayarla görme, sayısal görüntü işleme

CONTENTS

Page

M.Sc THESIS EXAMINATION RESULT FORM ii

ACKNOWLEDGMENTS iii

ABSTRACT iv

ÖZ v

CHAPTER ONE - INTRODUCTION 1

1.1 Structure of the Image System 3

CHAPTER TWO - IMAGE ENHANCEMENT 4

CHAPTER THREE - SEGMENTATION 6

3.1 Color Thresholding 6

3.2 Binarization 7

3.3 Contour Process 9

CHAPTER FOUR - FEATURE EXTRACTION 11

4.1 Center Of Mass 11

4.2 Ratio of Extreme Distances 11

4.3 Convex Hull 13

4.4 Convexity Defect Process 13

CHAPTER FIVE - CLASSIFICATION 20

5.1 Pre-Classification 21

5.2 Classification Arc 22

5.3 Tracking Process 23

CHAPTER SIX - RESULTS 25

CHAPTER SEVEN - CONCLUSION 28

7.1 Comparison 29

REFERENCES 32

CHAPTER ONE
INTRODUCTION

Main goal of the study is to develop a recognition system which recognizes hand gestures in our database. The database consists of 6 types of classes. Each gesture class symbolizes a number from 0 to 5. One sample image from each class can be seen in Figure 1.1. Each class in the database has 50 sample images. These sample images have fixed background, nonuniform illumination and undesired high frequency components caused by background texture.

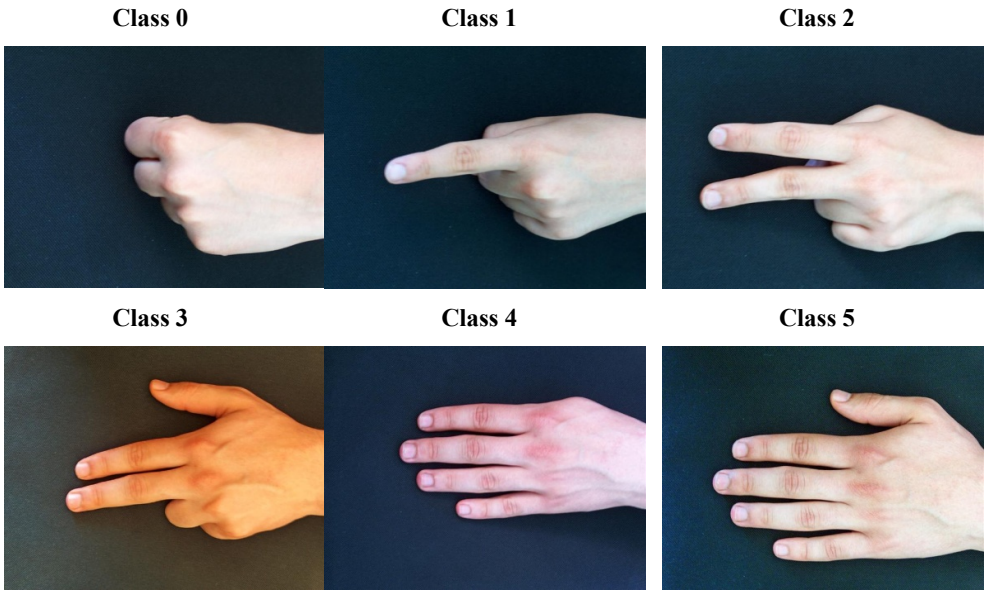


Figure 1.1 Hand gesture classes from 0 to 5

Block diagram of the hand gesture recognition system can be seen in Figure 1.2. Main stages of the hand gesture recognition system are listed below:

- Image enhancement
- Segmentation
- Feature extraction
- Classification

Image enhancement stage is needed to reduce effects of high frequency

components in the image. The images have undesired high frequency components caused by background texture and this type of components effects success of segmentation results negatively. Median type of filter is used in order to reduce effects of these high frequency components.

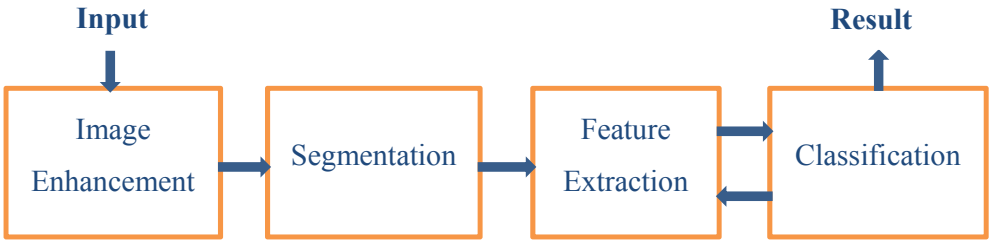


Figure 1.2 Block diagram of the hand gesture recognition system.

In segmentation stage, background information is removed from the image. Thanks to the fixed background, there is an easy way to segment hand from the background. Red color value is a dominant feature of the hand region. Therefore, first step of segmentation stage is red color thresholding process to obtain regions that are similar to skin color. Then, binary image is obtained by applying adaptive thresholding to gray level image that contains these regions. Segmentation stage is completed by choosing the longest inner contour of the longest outer contour in the binary image as the hand.

After image segmentation stage, the image is ready for feature extraction. Rotation, size and location invariant hand features based on hand shape are extracted in feature extraction stage. These features are obtained using center of mass and shape descriptors such as contours and convexity defects.

Last stage of the recognition system is classification. This stage considers the features which come from the feature extraction stage. If these features are not appropriate, classification stage sends a feedback to the feature extraction stage to tune itself and generate new feature values. If everything is OK, an arc which intersects all of open fingers with a minimum angle is created. Center point of this arc is assumed as the center of mass of the hand. Radius of the arc is chosen such that it intersects all open fingers in any condition. These intersected fingers are counted

and the count gives the recognition result.

1.1 Structure of the Image System

In this study, top-left corners of the images are taken as the origin points of the images. Figure 1.3 shows an image with its origin and axes.

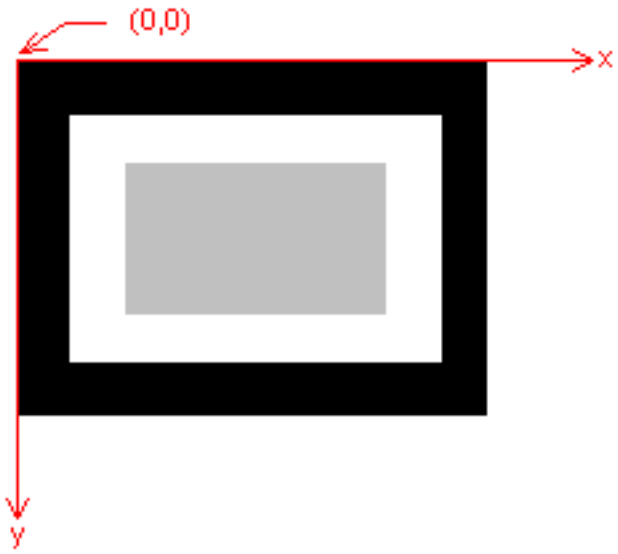


Figure 1.3 Origin of an image and its (x,y) coordinate system

Color images consist of three channels, gray level and binary images consist of one channel. Each element is represented by 8 bits. Each channel is assumed as an N x M array:

$$f(x,y) = \begin{bmatrix} f(0,0) & \cdots & f(N-1, & 0) \\ \vdots & \ddots & \vdots & \\ f(0,M-1) & \cdots & f(N-1,M-1) \end{bmatrix} \quad (1.1)$$

The database images are color and resolution of the images is 640 x 482 pixels. The images are in RGB color space.

CHAPTER TWO
IMAGE ENHANCEMENT

The database images have to be enhanced in order to increase success rate of the next stages. Aim of this stage is to emphasize the hand in the image.

If the database images are analyzed, it can be observed that due to the nonuniform illumination and texture of the background, some pixel values of the background may have same color values with the hand region. These types of regions have high frequency components and these regions have to be eliminated. Texture of the background and nonuniform illumination can be seen in Figure 2.1.



Figure 2.1 Nonuniform illumination and background texture

In order to eliminate these high frequency components, median type of low-pass filter is used. Median type of filter is applied to raw image two times for each color channel to reduce these high frequency terms. Median filter can be expressed as follows:

$$k \in \mathbb{N}, \qquad 2\mathrm{k}+1=\mathrm{n}$$

$$S = \left\{x - \frac{n-1}{2}, \dots, x-1, x, x+1, \dots, x + \frac{n-1}{2}\right\} \tag{2.1}$$

$$R = \left\{y - \frac{n-1}{2}, \dots, y-1, y, y+1, \dots, y + \frac{n-1}{2}\right\} \tag{2.2}$$

$$dst(x,y) = median\{\, src(s,r) \mid s \in \mathrm{S}, \qquad r \in \mathrm{R}\} \tag{2.3}$$

where

n defines size of the filter. n is equal to 5 in this study.

Shape of the hand is an important parameter for the recognition system. While median filter provides impulse noise reduction, it also provides less blurring than linear smoothing filters of same size (Gonzales & Woods, 2008). Effect of applying median filter to a sample image can be seen in Figure 2.2.

Raw image | Median low-pass filter result

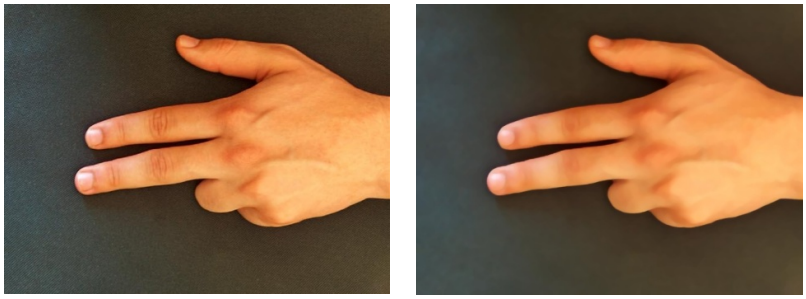


Figure 2.2 Raw image and its median filter result

Resulting images after applying adaptive tresholding to unfiltered and filtered images are shown in Figure 2.3. Adaptive thresholding will be used in the (next) segmentation stage.

Unfiltered | Using median filter

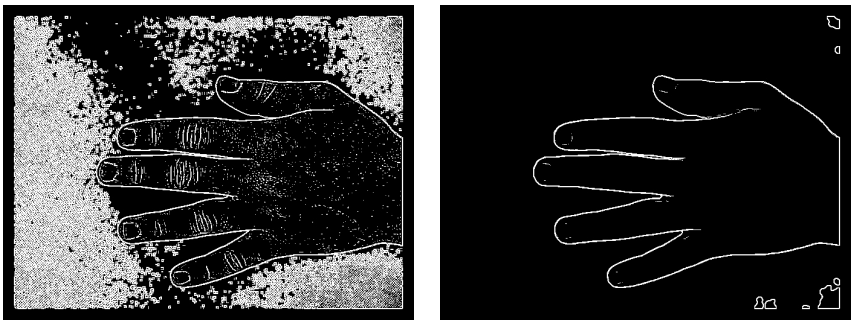


Figure 2.3 Effect of the median filtering to the next stage

CHAPTER THREE
SEGMENTATION

Aim of the segmentation stage is to extract the hand from the background and eliminate unnecessary hand details such as fingernails for the next stages. Segmentation stage consists of 3 parts as follows:

- Color thresholding
- Binarization
- Contour process

3.1 Color Thresholding

Thanks to fixed background color of the database images, there is an easy way to separate regions that are similar to skin color from the background. If the background and the hand region color values are analyzed, it can be seen that red values of the image is a dominant feature to specify regions that are similar to skin color. This analysis shows that red values of skin color pixels are generally greater than or equal to 75.

Red values greater than or equal to 75 are assumed as regions that are similar to skin color, values below 75 are assumed as background. In order to obtain these regions, a simple thresholding process is performed by using only the red channel of the RGB image. A color extraction result and its filtered image are shown in Figure 3.1. This color thresholding process is defined as follows:

$$dst(x,y) = \begin{cases} src(x,y) & if\ src(x,y)_{Red} \geq 75 \\ 0 & otherwise \end{cases} \quad (3.1)$$

In spite of the fact that the color thresholding process tries to extract hand region from background, it can be seen in Figure 3.2 that some background regions can also pass from the color thresholding process. On the other hand, since some parts of the hand region, such as fingernails, effect success of the next stages negatively, these

regions should be eliminated.

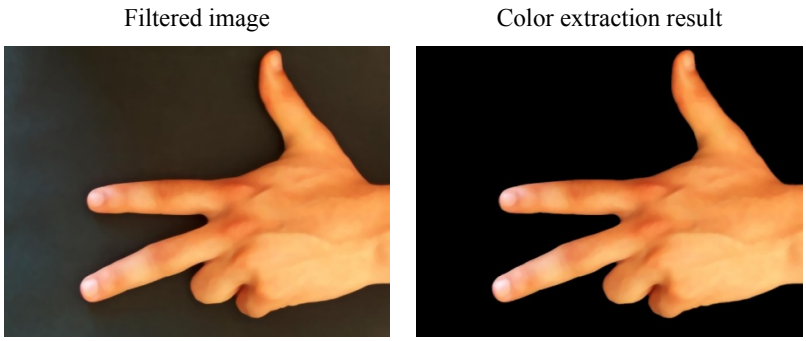


Figure 3.1 Filtered image and its color extraction result



Figure 3.2 Some background regions after the color extraction

3.2 Binarization

Before performing thresholding to obtain binary image, the RGB color image has to be transformed to a gray level image. Luminance component (Y) is calculated using red, green and blue parts of the color image for each pixel in order to obtain gray level image. Figure 3.3 shows a color image and its gray level image. Luminance component (Y) is calculated as below:

$$Y = 0.2989 \times Red + 0.5870 \times Green + 0.1140 \times Blue \qquad (3.2)$$

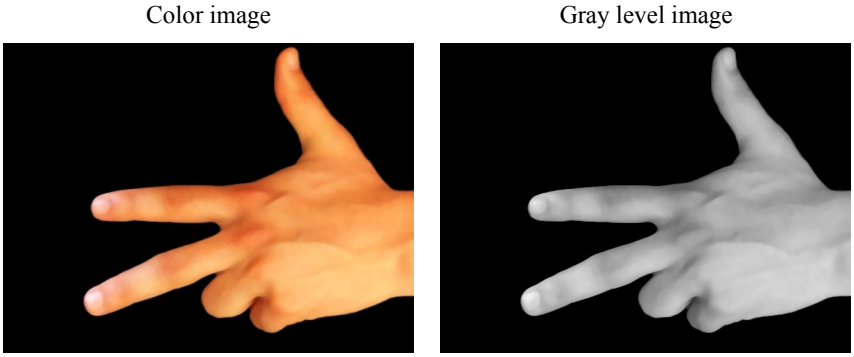


Figure 3.3 Color image and its gray level image

After the gray level transform, the image is ready for binarization. There are many methods in image processing to obtain a binary image from a gray level image. However, due to illumination of the images being nonuniform, using fixed thresholding methods are useless because intensity values of the hand region vary from one point to another point. Adaptive thresholding is used to binarize the gray level image. Adaptive thresholding method is defined as follows:

$$dst(x,y) = \begin{cases} 1, & if\ src(x,y) > T(x,y) \\ 0, & otherwise \end{cases} \tag{3.3}$$

Using (2.1) and (2.2)

$$Neighborhood\ [src(x,y)] = \{src(s,r) |\ s \in S, r \in R\} \tag{3.4}$$

$$T(x,y) = \frac{\sum_{\substack{s \in S \\ r \in R}} src(s,r)}{n \times n} - C \tag{3.5}$$

where:

n defines size of a pixel neighborhood that is used to calculate a threshold value for the pixel: 3, 5, 7, and so on. n is equal to 5 in this study.

C is a constant value. C is equal to 5 in this study.

Before applying adaptive thresholding to the image, zeros are padded to the image. Length of the padding is 20 pixels and the padding is applied to left, right, top and bottom of the image. Result of this process can be seen in Figure 3.4.

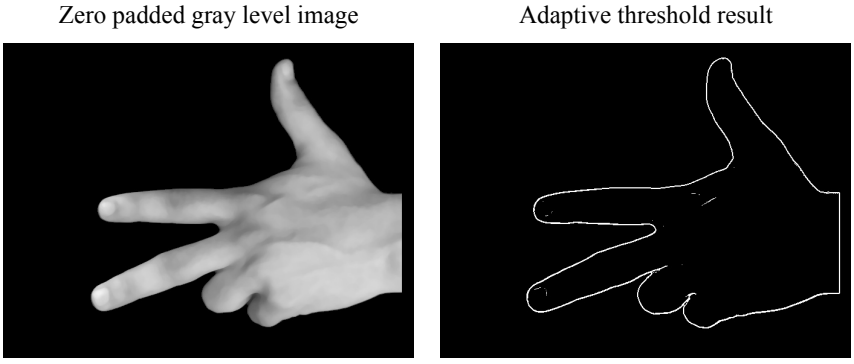


Figure 3.4 Zero padded gray level image and its binary image

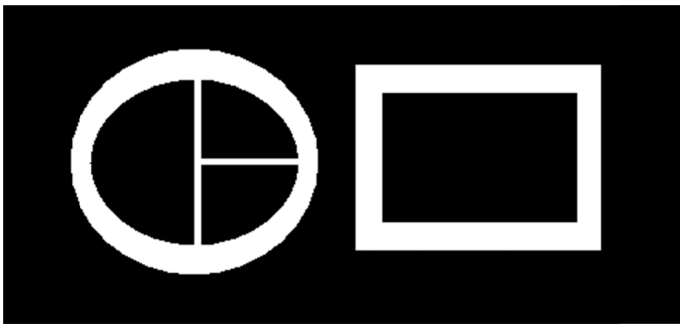
3.3 Contour Process

Contour is a list of points which represents boundary of a line or a curve in a binary image (Bradski & Kaehler, 2008). It can be assumed that the binary image obtained from the binarization stage consists of lines and curves. These lines and curves have to be represented by a contour in order to process hand shape.

In the study, contours are separated into two types: inner contours and outer contours. An inner boundary of a line or a curve is represented by related inner contour and an outer boundary of itself is represented by related outer contour. Contours are retrieved from the binary image using Suzuki and Abe's algorithm (Suzuki & Abe, 1985). Figure 3.5 shows inner and outer contours of a binary image.

Some small background regions may exist after the color thresholding process. First outer contours of the hand and these regions are in first contour level. The longest first level outer contour in the binary image is selected as hand and the other outer contours are ignored in order to eliminate these small background regions. The longest inner contour of the hand has more details than the outer contour. Since hand shape is the most important feature, the longest inner contour of the hand is used for next stages. A new RGB image is created with zeros and contour of the hand is drawn in red RGB(128, 0, 0) in order to emphasize the boundary. Inside of the drawing is filled with white color RGB(255, 255, 255) to define hand region. Results of all contour process stages are shown in Figure 3.6.

Binary image



Contours of the binary image

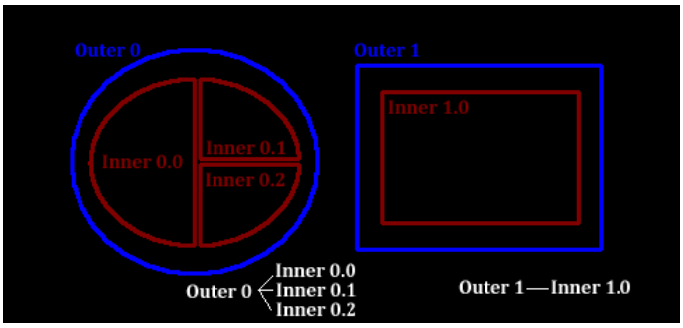
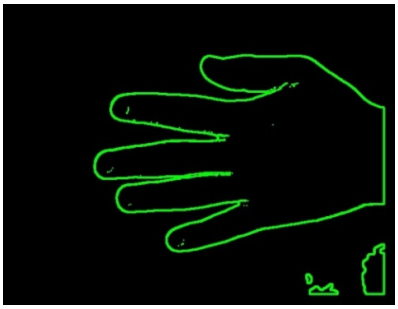
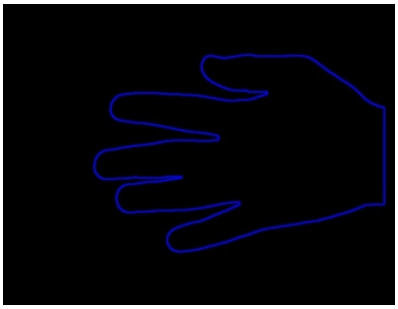


Figure 3.5 A binary image and its contours

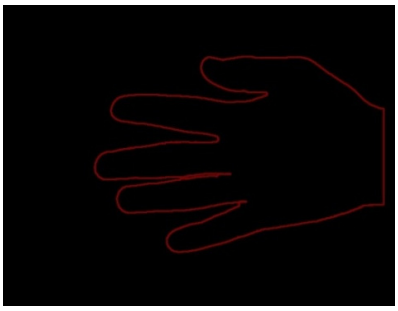
All contours in the image



Outer hand contour



Inner hand contour



Result of the contour process



Figure 3.6 All contours, inner and outer hand contours and contour process result of an image

CHAPTER FOUR
FEATURE EXTRACTION

In the feature extraction stage, the representative features of the hand gesture are obtained by using shape analysis techniques and empirical expressions which obtained with our observation, try and error experiments.

4.1 Center of Mass

Center of mass of the segmented image is calculated in order to determine a reference point. Non-zero values of the image are assumed as '1' and the center of mass is calculated only using red channel as follow:

Spatial moments $m_{ji}\ of\ src(x,y)$:

$$m_{ji} = \sum_{x,y} (src(x,y).x^j.y^i) \tag{4.1}$$

where $O = (\mu, \xi)$ is the center of mass of $src(x,y)$:

$$\mu = \left(\frac{m_{10}}{m_{00}}\right), \xi = \left(\frac{m_{01}}{m_{00}}\right) \tag{4.2}$$

Calculated center of mass of a segmented image is shown in Figure 4.1.

4.2 Ratio of Extreme Distances

Max and min euclidean distances of the hand contour to the center of mass and the max distance / the min distance ratio are calculated as follows:

$$\mathbb{N}_n = \{0, 1, 2, \dots, n\}$$
$$P = \{P(t) | t \in \mathbb{N}_n\}$$

$$P(t) = (x, y)$$

$$\delta = \max\{\| P(t) - O \| | p(t) \in P\} \quad (4.3)$$

$$\gamma = \min\{\| P(t) - O \| | p(t) \in P\} \quad (4.4)$$

$$\beta = \frac{\delta}{\gamma} \quad (4.5)$$

Extreme points of a hand contour are shown with green points in Figure 4.2. Aim of calculating β value is to give information whether hand is open or not.

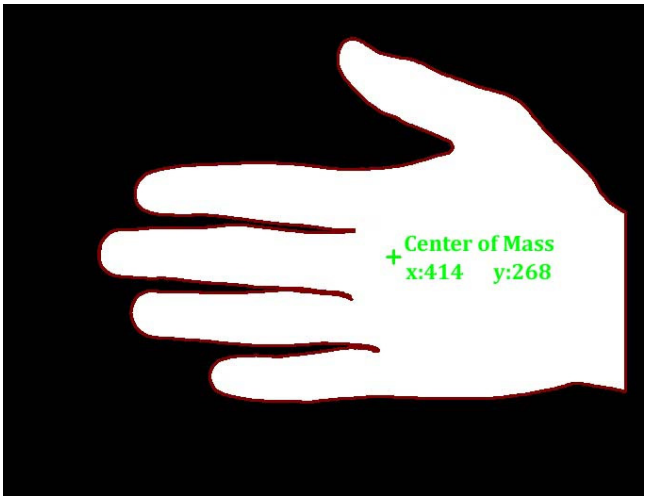


Figure 4.1 Center of mass of the hand image and its coordinates

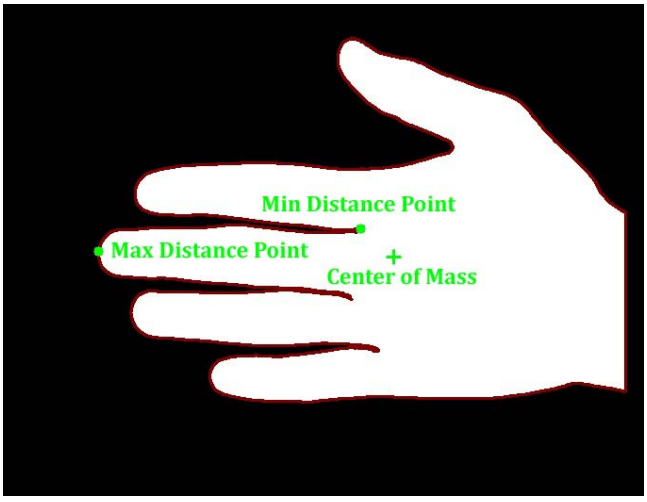


Figure 4.2 Max and min points to the center of mass

4.3 Convex Hull

Convex hull of the hand is found by using the contour data. The Sklansky's algorithm (Sklansky, 1975) is used to obtain of the convex hull. Convex hull of a hand contour is shown in Figure 4.3.

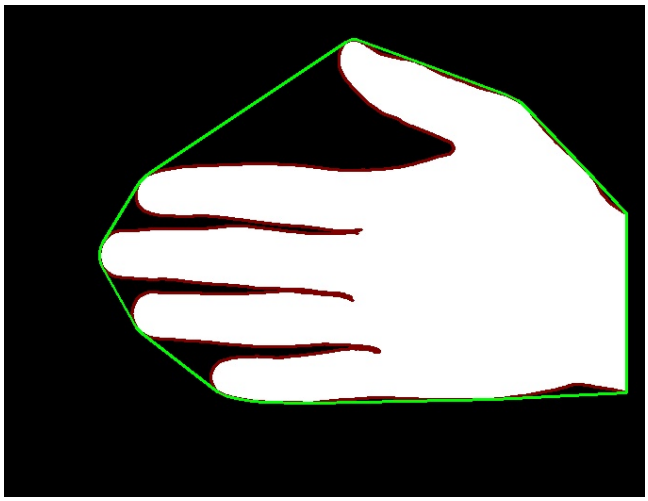


Figure 4.3 Hand with its convex hull

4.4 Convexity Defect Process

Convexity defect is a useful way for extracting data from the hand shape. Convexity defects of the hand are obtained by evaluating the hand contour and its convex hull using Homma and Takenaka's convexity defects algorithm (Homma & Takenaka, 1985). This algorithm gives information about start and end points of defects on the convex hull, the farthest defect points from related edge of the convex hull and distance of the farthest points.

$$\mathrm{A} = \{E_i | \, i \in \{0,1,2,\dots,n\}\} \tag{4.6}$$
$$O = (\mu, \xi)$$
$$for \; \forall \, E_i \in A,$$

S_i: Defect start point

T_i: Defect end point

d_i: The farthest distance of a defect

G_i: The farthest distance point of a defect

The defects which have its farthest distances less than or equal to 6 are eliminated and ignored for next computations. Figure 4.4 shows the remaining defects after this elimination in orange color. This process can be expressed as follows:

$$A^I \subset A$$
$$if\ d_i \leq 6, E_i\ \notin\ A^I, otherwise\ E_i \in\ A^I$$
$$A^I = \{E_i | d_i > 6\} \subset A \tag{4.7}$$

Edge lengths between the start points and the end points of a convexity defects are shown in Figure 4.5. Edge lengths of the convex hull are calculated using euclidean distance:

$$for\ \forall\ E_i \in A^I, e_i = \| S_i - T_i \| \tag{4.8}$$

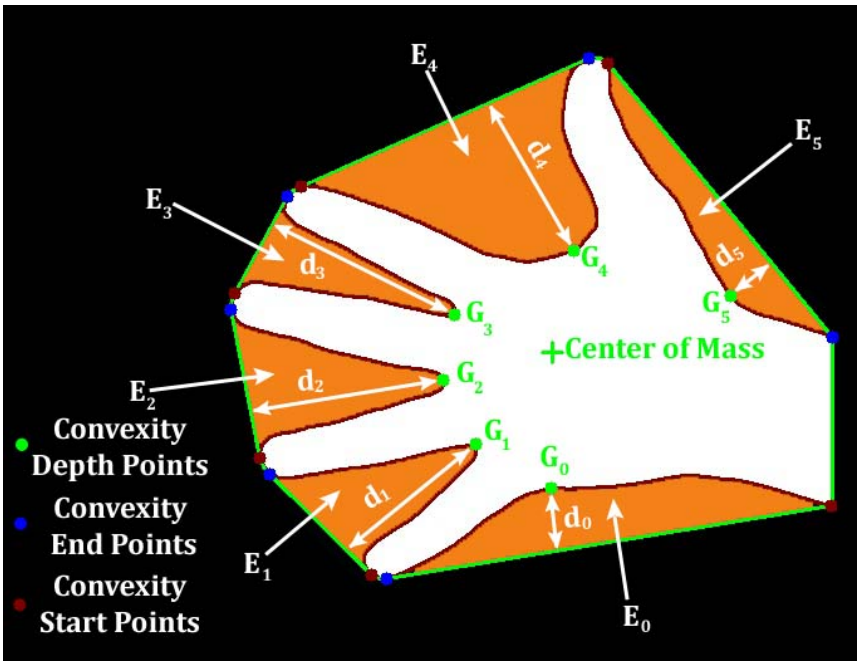


Figure 4.4 Convexity defect concept for the hand image

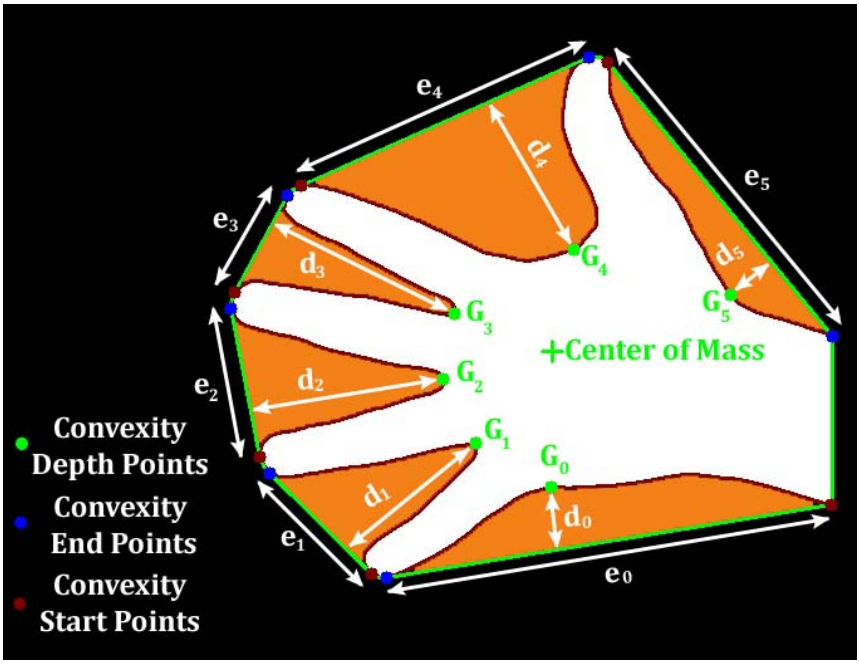


Figure 4.5 Lengths of the edges of the convex hull are calculated for each defect using the start and the end points

If the ratio between depth distance and related convex hull edge is below a certain 'c' value, related defect is ignored in the next computations. Default value of 'c' is equal to 0.25. If the classifier sends a feedback, this defect elimination and rest of the feature extraction process are reperformed, but this time elimination is performed by taking 'c' as 0.16 . Figure 4.6 shows one of the elimination results. The ratio process can be expressed as:

$$A^{II} \subset A^{I}$$

$$for\ any\ c\ value:$$

$$if\ \frac{d_i}{e_i} < c, \qquad E_i \notin A^{II}$$

$$if\ \frac{d_i}{e_i} \geq c, \qquad E_i \in A^{II}$$

$$A^{II} = \left\{ E_i \middle| \frac{d_i}{e_i} \geq c \right\} \subset A^{I} \tag{4.9}$$

$$\sigma = s(A^{II}) \tag{4.10}$$

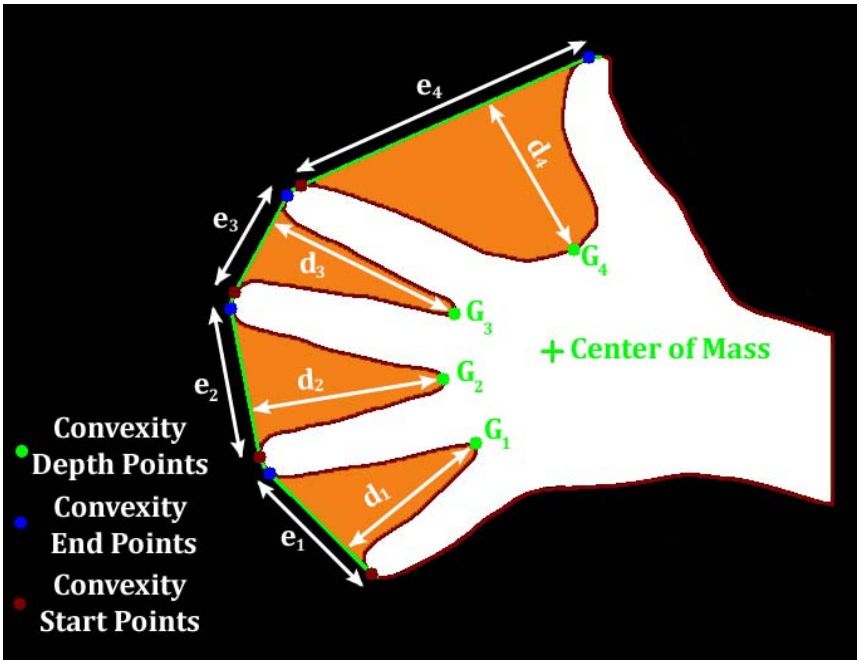


Figure 4.6 Result for c = 0.25 threshold elimination. Remaining defects from the elimination are shown in orange color.

σ value gives information whether there are any open fingers or not.

Distances between start points and the center of mass, and between end points and the center of mass are shown in Figure 4.7. These distances are calculated using euclidean distance:

$$for\ \forall E_i \in A^{II}, \qquad s_i = \| S_i - O \| \ and\ t_i = \| T_i - O \| \tag{4.11}$$

and max value of these distances is calculated:

$$u_i = \max\{s_i, t_i\} \tag{4.12}$$

$$u_m = \max\{u_i | E_i \in A^{II}\} \quad 0 \leq m \leq n \tag{4.13}$$

If any u_i value is less than 0.55 x u_m, related defect is eliminated and ignored in next computations:

$$A^{III} \subset A^{II}$$
$$if\ u_i < (0.55 \times u_m), \quad E_i \notin A^{III}$$
$$if\ u_i \geq (0.55 \times u_m), \quad E_i \in A^{III}$$
$$A^{III} = \{E_i | u_i \geq (0.55 \times u_m)\} \subset A^{II} \tag{4.14}$$

and min value of these distances is found and radius of an arc is calculated:

$$r = 0.925 \times \min\{u_i | E_i \in A^{III}\} \tag{4.155}$$

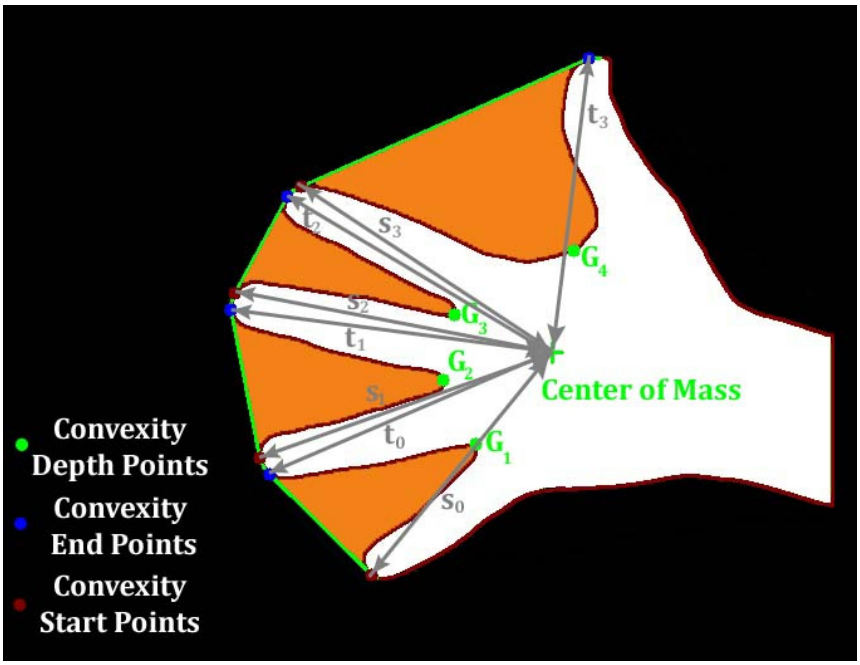


Figure 4.7 Distances from the start points and the end points to the center of mass are shown in gray arrows.

The center of mass is assumed as orgin of the hand. Angles for start and end point of each defect is calculated as follows:

$$for\ \forall\ E_i^I \in A^{III}$$
$$S_i^I = (x_i, y_i)$$
$$T_i^I = (z_i, t_i)$$
$$O = (\mu, \xi)$$

$$\widehat{s_i} = \arctan\left(\frac{y_i - \xi}{x_i - \mu}\right) \tag{4.16}$$

$$\widehat{t_i} = \arctan\left(\frac{t_i - \xi}{z_i - \mu}\right) \tag{4.17}$$

and an arc which generally intersects most of open fingers' defects with min angle is obtained:

$$s(A^{III}) = p$$

If the elements of A^{III} are renamed as $E_0^I, E_1^I, E_2^I, \dots, E_{p-1}^I$ respectively, A^{III} will become: $A^{III} = \{E_0^I, E_1^I, E_2^I, \dots, E_{p-1}^I\}$

$$i \in \{0,1,2, \dots, p-1\}, \;\; j \in \{0,1,2, \dots, p-1\}$$

$$if\ i \neq 0;$$

$\widehat{C_i D_{i-1}}$ is an arc with angle θ_i and radius r

$$for\ \forall\ E_j^I \in A^{III}, \widehat{C_i D_{i-1}} \cap E_j^I \neq \emptyset$$
$$then\ \theta_i = \widehat{s_i} - \widehat{t_{i-1}}$$
$$for\ \exists\ E_j^I \in A^{III}, \widehat{C_i D_{i-1}} \cap E_j^I \neq \emptyset$$
$$then\ \theta_i = 360 - (\widehat{s_i} - \widehat{t_{i-1}})$$

$$if\ i = 0;$$

$\widehat{C_0 D_{p-1}}$ is an arc with angle θ_0 and radius r

$$for\ \forall\ E_j^I \in A^{III}, \widehat{C_0 D_{p-1}} \cap E_j^I \neq \emptyset$$
$$then\ \theta_O = \widehat{s_O} - \widehat{t_{p-1}}$$
$$for\ \exists\ E_j^I \in A^{III}, \widehat{C_0 D_{p-1}} \cap E_j^I \neq \emptyset$$
$$then\ \theta_O = 360 - (\widehat{s_O} - \widehat{t_{p-1}})$$

$$\theta_k = \min\{\theta_i | i \in \{0,1,2,\dots,p-1\}\} \quad (4.18)$$
$$k \neq 0 \; then \; C_k\widehat{D_{k-1}}$$
$$k = 0 \; then \; C_0\widehat{D_{p-1}}$$

Result of an arc process is shown in Figure 4.8.

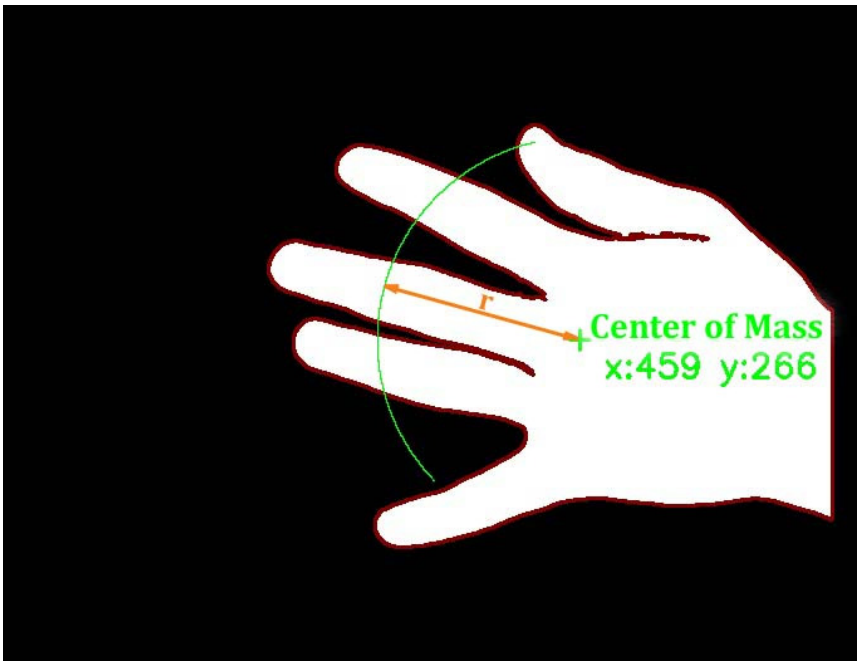


Figure 4.8 The arc θ_k is seen in green color. Its center is the center of mass and it is radius is r.

CHAPTER FIVE
CLASSIFICATION

Aim of this stage is classifying hand gesture of the input image by using the data obtained from feature extraction. Segmented images are processed with the features. These feature parameters are shown Table 5.1. Figure 5.1 shows one of segmented images.

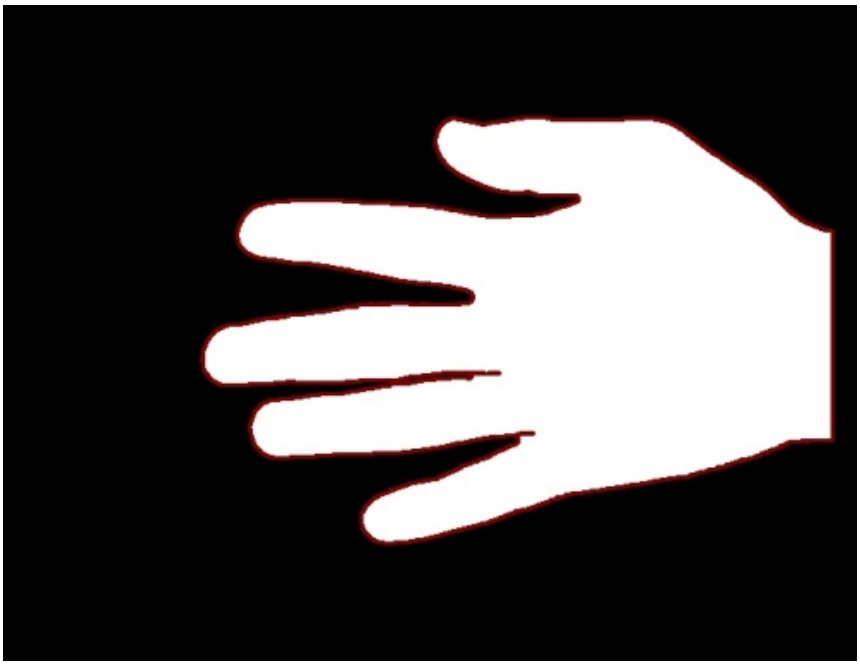


Figure 5.1 Segmented images are used as input images in the classification stage.

Table 5.1 Hand feature parameters obtained from the feature extraction stage

Feature	Description
β	$\beta = \frac{\delta}{\gamma}$
σ	$\sigma = s(A^{II})$
θ_k	Its radius is r, its center is $O = (\mu, \xi)$

5.1 Pre-Classification

Conditional expressions are used to evaluate whether the hand is open or not and the gesture is known or not.

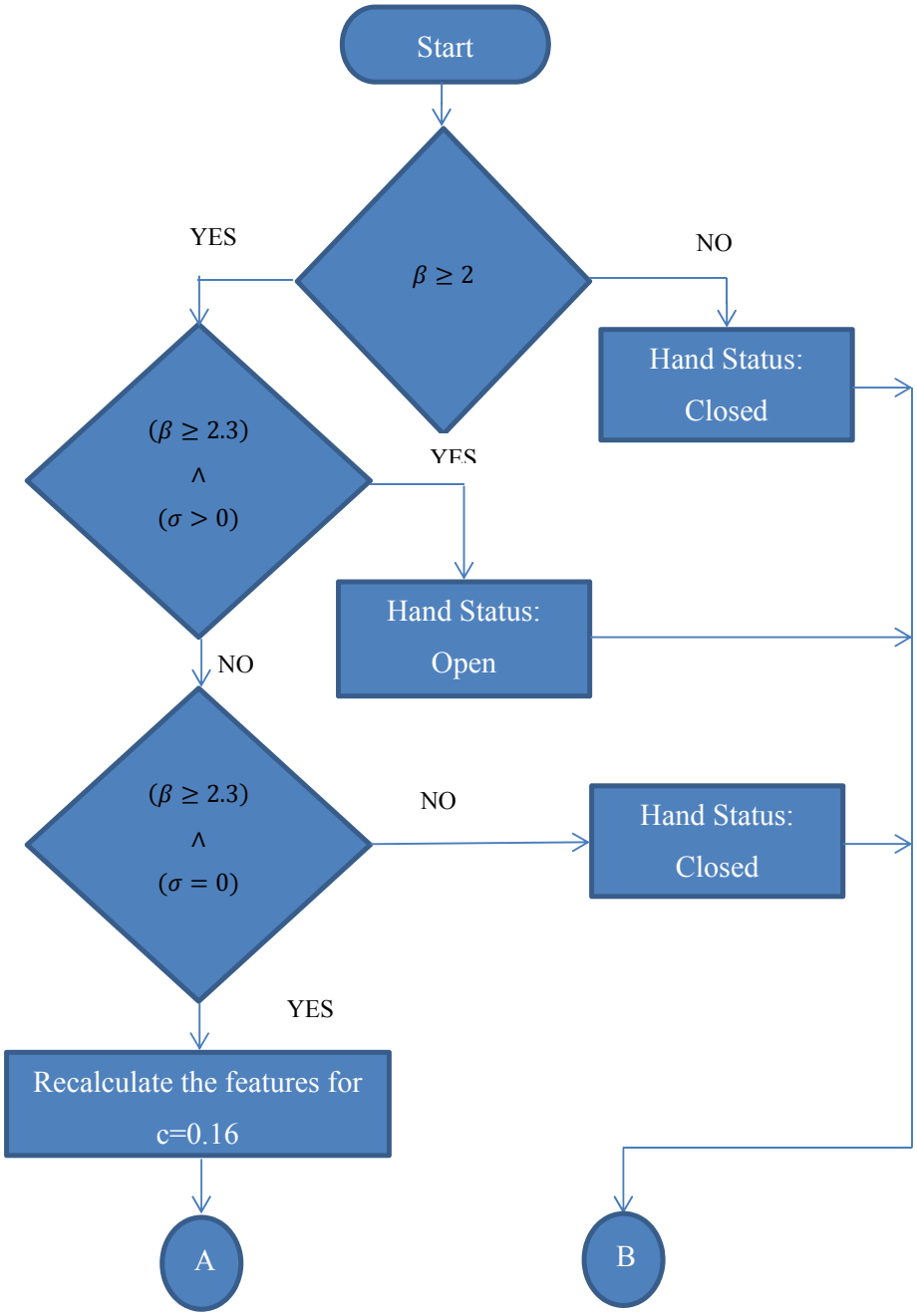


Figure 5.2 First part of pre-classification flow chart

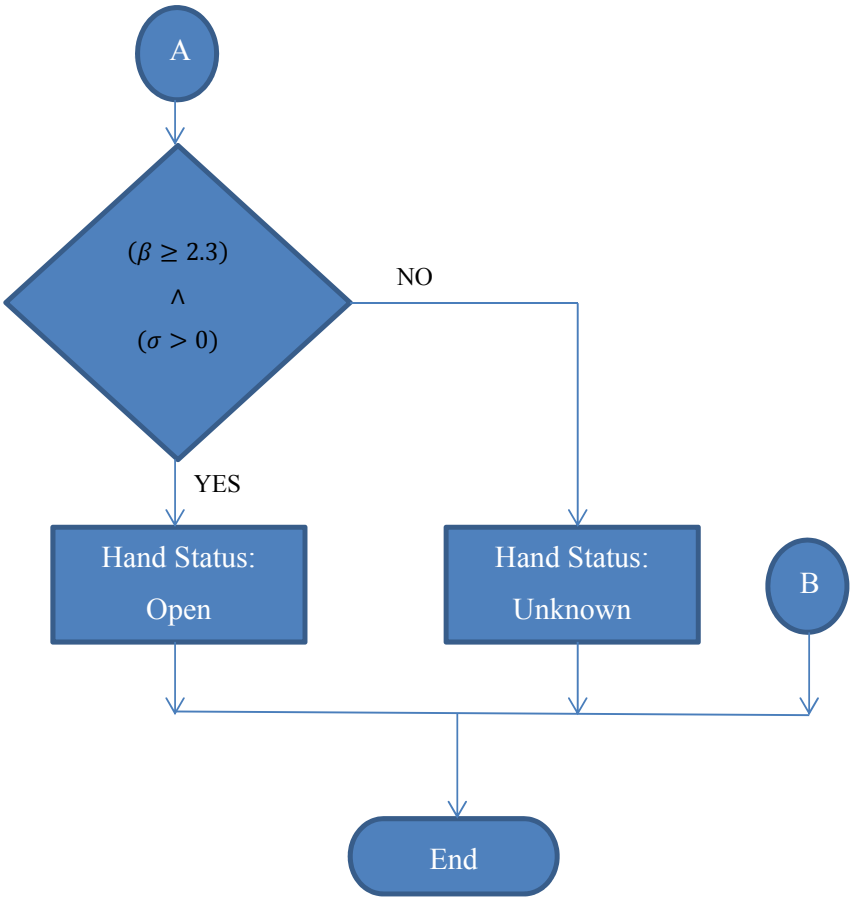


Figure 5.3 Second part of pre-classification flow chart

Figure 5.2 and Figure 5.3 show flow chart of the pre-classification. If the result of pre-classification is "open", the classification process continues. If the pre-classification result is "closed", result of the recognition process is gesture 0.

5.2 Classification Arc

In the classification extraction stage, in order to be sure that arc θ_k intersects with all open fingers, θ_k is extended from its end points by 15 degrees. This new arc is called θ_c. Figure 5.4 shows θ_k and θ_c of a segmented image.



Figure 5.4 θ_k is extended by 15 degrees from its end points in order to intersect with all open fingers.

5.3 Tracking Process

Using only the red color channel of the segmented image and by tracking θ_c from one of its end points to another in 0.1 degree steps, a one dimensional signal in the form shown in Figure 5.5 is obtained.

There are three types of points in this signal:

- Black Points: Red color value of the point is 0
- Contour Points: Red color value of the point is 128
- White Points: Red color value of the point is 255

Number of transitions from contour points to white points gives the type of the hand gesture. Each recognition result is written to top left of the image. Figure 5.6 shows one of the classification results and its classification input.

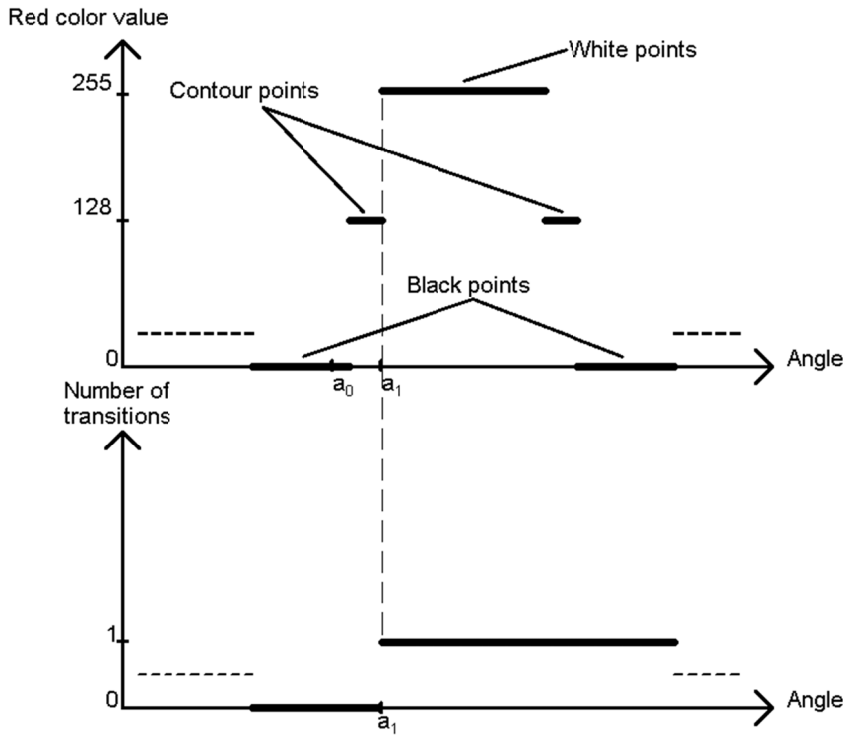


Figure 5.5 Top Figure is 1D signal and bottom Figure shows number of transitions. At a_1 angle, from contour to white point transition occurs.

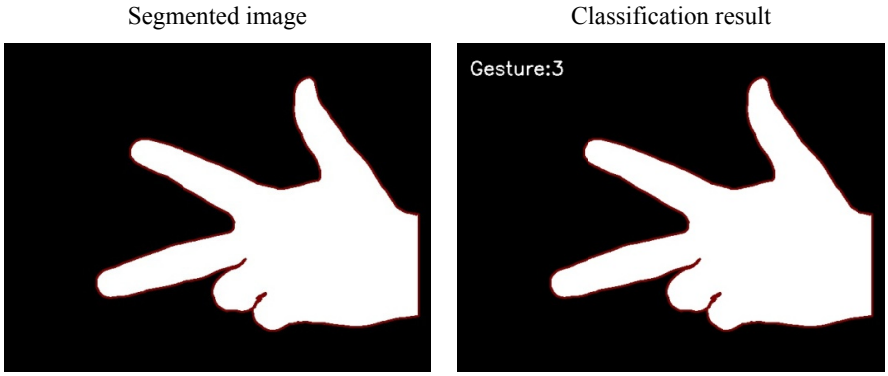


Figure 5.6 Input image of the classification stage and result of the classification

CHAPTER SIX
RESULTS

The hand gesture recognition system is tested using the database images. Each class has 50 images. All classes except class 1 are recognised with zero error. Three images from class 1 are recognised as unknown gesture. Overall accuracy of the system is 99%. Recognition results are shown Table 6.1.

Table 6.1 Classification result of the system

		Classification Result							Error %
	Gesture Class	0	1	2	3	4	5	Unknown	
Input	0	50	0	0	0	0	0	0	0
	1	0	47	0	0	0	0	3	6
	2	0	0	50	0	0	0	0	0
	3	0	0	0	50	0	0	0	0
	4	0	0	0	0	50	0	0	0
	5	0	0	0	0	0	50	0	0

Two unknown results are shown in Figure 6.1 and related raw images are shown in Figure 6.2. Although these images belong to class 1, they are classified as unknown gesture. If these three images are analysed, it can be seen that $\frac{d_i}{e_i}$ ratio of these images are below threshold value 'c' (even for 0.16) of the feature extraction stage.

It can be seen in Figure 6.2 that there are some semi-closed fingers. These fingers effect $\frac{d_i}{e_i}$ ratio and recognition result. Features and max $\frac{d_i}{e_i}$ value of unknown results are shown in Table 6.3. Table 6.2 shows feature sets, max $\frac{d_i}{e_i}$ values and related recognition results of various input images.

Image result 023 Image result 027

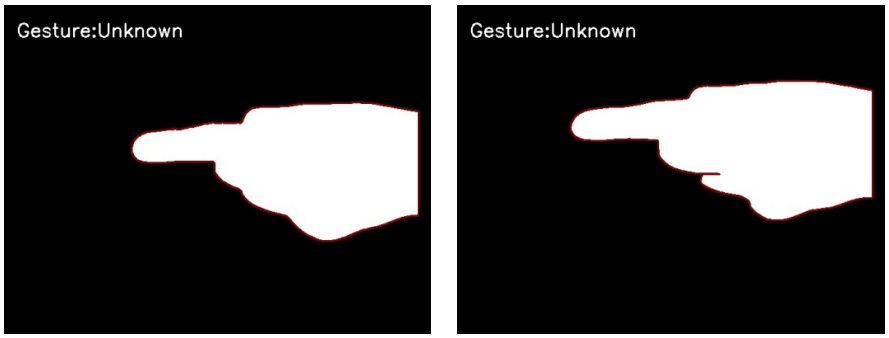


Figure 6.1 These images are recognized as unknown gesture

Raw image 023 Raw image 027



Figure 6.2 Raw hand images of Figure 6.1

Table 6.2 Feature sets, max $\frac{d_i}{e_i}$ values and results of various input images

Input Image Class	β	θ_k	O		r	σ	$\max(\frac{d_i}{e_i})$	Result
			μ	ξ				
0	1.7	-	484	250	-	0	0.2418	0
1	2.53	117	480	287	263	1	0.1707	1
2	4.93	77	410	232	291	2	1.9444	2
3	4.54	83	452	295	194	3	1.7891	3
4	9.17	139	419	287	200	3	2.5982	4
5	5.63	128	440	273	221	4	2.2216	5

Table 6.3 Features and max $\frac{d_i}{e_i}$ value of unknown results

Image	β	θ_k	O		r	σ	$\max(\frac{d_i}{e_i})$
			μ	ξ			
Image result 023	3.06	-	484	287	-	0	0.1502
Image result 025	3.08	-	482	253	-	0	0.1535
Image result 027	3.74	-	476	215	-	0	0.1440

CHAPTER SEVEN
CONCLUSION

In this study, a hand gesture recognition system is implemented. Image enhancement, segmentation, feature extraction and classification techniques are introduced for the hand gesture recognition system.

The first stage of the study is image enhancement. High frequency components of database images effect success of the next stages negatively. Median type of filter is applied to the input raw image two times in order to eliminate effects of these high frequency components. Since hand shape is an important factor for the hand recognition system, using median filter has an advantage.

In the segmentation stage, background and some unnecessary hand data of the filtered image are eliminated for the next stages. After the color thresholding process, some background regions may exist. However, thanks to the image enhancement stage, these regions are very small. These small regions are successfully eliminated in the contour process. Since nonuniform illumination makes fixed thresholding techniques useless, adaptive thresholding is used in order to neutralize the effect of the nonuniform illumination. Adaptive threshold calculates its threshold value individually for each pixel. Binary image obtained from adaptive threshold method may carry small background regions. These regions and some unnecessary hand details such as fingernails may effect the result of the next stages negatively. The contour process eliminates these regions and details.

In the feature extraction stage, some representative data about the hand gesture is obtained by using empiric expressions which obtained with our observation, try and error experiments. There are two aims of this stage. One of them is extracting data to find out whether the hand is open or not and the gesture is known or not. The other one is extracting information about open fingers.

In the classification stage, first step is to find out hand status whether the hand is

open or not and the gesture is known or not. This process is evaluated by using β and σ features. If the hand status is open, using extended arc θ_c, all intersected fingers in the segmented image are counted. Number of the intersections gives the type of the hand gesture. Computational expressions of the stage are simple and results of the classifications are successful.

All classes except class 1 are recognised with zero error. Error rate of the class 1 is 6%. Three images of class 1 recognised as unknown image. Overall accuracy of the system is 99%.

Our recognition system has been implemented using OpenCV computer vision library. This implementation has been tested also as a real time hand gesture recognition system. Our pc configuration is Intel Pentium M 740 1.73 GHz processor, 2 GB RAM and our webcam resolution is 640 x 480. Processing speed of the system is 6-8 frames per second. It is well enough to work as a real time hand gesture system. However, segmentation stage of the system needs improving for general use because it can only segment the hand when the background has fixed black color and also it depends color of the skin. In the other hand, the recognition system just tries to count open fingers but the system does not give any information about which fingers are open. Feature extraction stage and classification stage of the system have to be reconsidered if locations of open fingers are important or different gestures but same number of open fingers is wanted to give same results. Figure 7.1 shows same results of different gestures.

7.1 Comparison

There are a lot of studies on hand gesture recognition. Comparison table of some of these (Huang & Hu & Chang, 2009; Malima & Özgür & Çetin, 2006; Yun & Peng, 2009) and our study is shown in Table 7.1.

All studies in table 7.1 are used different databases. Also, number of target classes and used features for each study are different. First study can work with uniform as

well as cluttered backgrounds as long as there are not too many skin - like color pixels in the background. Database images of second study have white wall as the background but real time system of this study has hand detection algorithm in order to detect hand in even cluttered backgrounds. Third study uses same colored backgrounds in its database in order to test classification performance. However, real time version of the system has skin color segmentation algorithm. Our study does not aim to work with complicated backgrounds. A black cloth is used as the background.

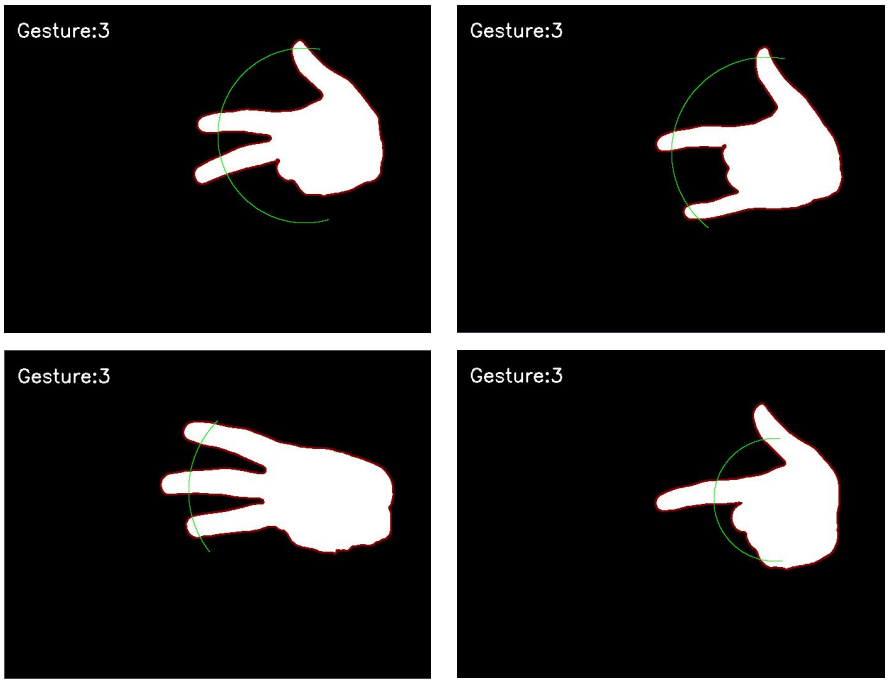


Figure 7.1 These gestures are recognized as class 3. Top left gesture only belongs to our database classes. Bottom right gesture has two open fingers but it is recognized as class 3.

In table 7.1, second and third studies uses support vector machine to classify hand gestures. In these two studies, the study which uses Hu invariant moments has better success rate than the study which uses Gabor filter and principal component analysis. The best success rate in the table belongs to our study with 99%.

It can be seen in Table 7.1 that hand gesture recognition systems have good success rates. According to the results, hand gesture recognition systems might be

able to become an alternative for conventional systems such as TV remote controls or computer mouse in the near future.

Table 7.1 Comparison table of various hand gesture recognition studies

	Author	Database & Background	Used Features	Classification Method	Success rate
1.	Asanterabi Malima, Erol Özgür, Müjdat Çetin, Sabancı University	5 classes 105 images & uniform as well as cluttered backgrounds	Intersection circle	1D binary signal tracking	91%
2.	Liu Yun, Zhang Peng, Qingdao University of Science and Technology	3 classes 390 test images & white wall as the background	Hu invariant moments	Support vector machine	96.2%
3.	Wu-Chih Hu, National Penghu University Deng-Yuan Huang, Sung-Hsiang Chang Da-Yeh University	11 classes 660 images & same colored backgrounds	Gabor filter with principal component analysis	Support vector machine	95.2%
4.	Bekir Can, Metehan Makinacı, Dokuz Eylül University	6 classes 300 images & black cloth as the background	Intersection arc, some empirical expressions	1D color signal tracking, conditional expressions	99%

REFERENCES

Bradski G., & Kaehler A. (2008). *Learning OpenCV* (1st ed.). O'Reilly Media, Inc.

Gonzalez R. C., & Woods R. E. (2008). *Digital Image Processing* (3rd ed.). Pearson Prentice Hall.

Homma K., & Takenaka E.-I. (1985). An image processing method for feature extraction of space-occupying lesions. *Journal of Nuclear Medicine, 26*, 1472 – 1477.

Huang Y.D., & Hu W.C., & Chang S.H. (2009). Vision-based Hand Gesture Recognition Using PCA+Gabor Filters and SVM, *2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP '09.*

Malima A., & Özgür E., & Çetin M. (2006). A Fast Algorithm For Vision-Based Hand Gesture Recognition For Robot Control. *2006 IEEE 14th Signal Processing and Communications Applications*.

Sklansky J.(1982). Finding the Convex Hull of a Simple Polygon. *Pattern Recognition Letters, 1(*2), 79-83

Suzuki S., & Abe K.(1985). Topological Structural Analysis of Digitized Binary Images by Border Following. *Computer Vision, Graphics, and Image Processing, 30*(1), 32-46

Yun L., & Peng Z. (2009). An Automatic Hand Gesture Recognition System Based on Viola-Jones Method and SVMs, *Computer Science and Engineering, WCSE '09, Second International Workshop. 2*, 72-74