

OBJECT SHAPE RECOGNITION

*A thesis submitted in partial fulfillment of the requirements for the award of the
degree of*

Bachelor of Technology

in

Electronics and Communication Engineering

by

Manas Ranjan Das (108EC026)

Sunil Barla (108EC029)



DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

OBJECT SHAPE RECOGNITION

A thesis submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Electronics and Communication Engineering

by

Manas Ranjan Das (108EC026)

Sunil Barla (108EC029)

Under the guidance of

Prof. S MEHER



DEPARTMENT OF ELECTRONICS AND COMMUNICATION

ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA



NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

CERTIFICATE

This is to certify that the thesis titled, " **Object Shape Recognition**" submitted by **Manas Ranjan Das** (108EC026) and **Sunil Barla** (108EC029) in partial fulfillment of the requirements for the award of **Bachelor of Technology Degree in Electronics and Communication Engineering** at **National Institute of Technology, Rourkela**(Deemed University) is an authentic work carried out by them under my supervision and guidance.

Date:

Prof. S MEHER

Department of Electronics and Communication Engineering

ACKNOWLEDGEMENT

On the submission of our thesis report on, “OBJECT SHAPE RECOGNITION” we would like to extend our gratitude and sincere thanks to our supervisor Prof. S MEHER for his constant motivation and support during the entire course of our work in the last one year. We truly appreciate and value his esteemed guidance and encouragement from the beginning to the end of this project. We are very grateful and indebted to him for his kind help in our successful completion of the project.

We also want to thank all our teachers Prof. S.K PATRA, Prof. POONAM SINGH, Prof. K.K MOHAPATRA, Prof. S ARI and Prof. L.P ROY for providing a solid background for our studies and research thereafter.

Above all we want to thanks all our friends whose direct and indirect support helped us in completion of our project

ABSTRACT

Objects around us make our environment; in day to day life we tend to classify each of the objects visible to us. We tend to classify each object like a ball is spherical; a notebook is rectangular and so on using our senses. A machine like computer does not have senses to recognize or even detect an object. We have to train or develop an algorithm for a machine like computer to do so. The approach here is to classify some of the common objects around us and decide whether they belong to any geometric shape or not. The shape of the objects can be represented by some feature space which may be used for recognizing shape of the objects. We use the corner detection method, signature method and chain code method to achieve a good recognition. The corner detection method is based on detecting corners on the boundary and then deriving the feature vector from the distance between the corners. The signature method is based on the distance of the boundary points from the centre of the object and all those distances from the feature vector. The chain code method is based on finding the chain code of the object and then finds the histogram of it, which forms the feature vector. The purpose of this thesis is to use all the three methods of recognition and visualize their performances.

Table of Contents**Page No.**

1. INTRODUCTION.....	9
1.1 What is object shape recognition?	9
1.2 PRE-PROCEESSING	10
1.3 OBJECT BOUNDARY EXTRACTION	11
1.4 MORPHOLOGICAL OPERATIONS	12
1.5 CORNER DETECTION	13
1.6 SIGNATURE.....	18
1.7 FREEMAN CHAIN CODE	19
2. EXPERIMENTAL WORK	21
2.1. CORNER DETECTION METHOD	21
2.2 SIGNATURE METHOD	30
2.3 CHAIN CODE APPROACH	37
3. CONCLUSION	42
4. REFERENCES.....	43

List of FiguresPage No.

Fig...1: A circle	9
Fig...2 Showing 3 points on a plane	14
Fig...3 Showing chain code directions for 4 connectivity	19
Fig...4 Showing chain code directivity for 8 connectivity	19
Fig...5 Showing histogram of the chain code of an object	20
Fig...6 Reference image 1	22
Fig...7 Test image1 for recognition by corner detection method	22
Fig...8 Plot of square of determinants at different boundary points for test image 1	23
Fig...9 Showing corner points in test image 1	23
Fig...10 Reference image 2	25
Fig...11 Test image 2 for recognition by corner detection method	25
Fig...12 Plot of Square of determinants at different boundary points for test image 2	26
Fig...13 Showing corner points in test image 3	26
Fig...14 Test image 3 for recognition by corner detection method	28
Fig...15 Showing corner points in test image 3	29
Fig...16 Test image 4 for recognition by signature method	31
Fig...17 Plot of signature of test image 4	31
Fig...18 Test image 5 for recognition by signature method	32
Fig...19 Plot of signature for test image 5	33
Fig...20 Reference image 3	34
Fig...21 Test image 6 for recognition by signature method	34
Fig...22 Plot of signature of reference image 3	35
Fig...23 Plot of signature of test image 6	35
Fig...24 Plot of error for comparison of signature values of Fig..20 and Fig..21	36
Fig...25 Reference image 4	38
Fig...26 Test image 7 for recognition by chain code method	38
Fig...27 Reference image 5	40
Fig...28 Test image 8 for recognition by chain code method	40

List of TablesPage No

Table 1: Showing the description vectors of reference image 1 and test image 1 by the corner detection method , error and MSE	24
Table 2: Showing the description vectors of reference image 2 and test image 2 by the corner detection method , error and MSE	27
Table 3: Showing the description vector of test image 3 by corner detection method	29
Table 4: Showing the chain code histogram of reference image 4 and test image 7, Error values and MSE	39
Table 5: Showing the chain code histogram of reference image 5 and test image 8, Error values and MSE	41

1. INTRODUCTION

1.1 What is object shape recognition?

In our everyday life we come across various objects of various shapes and with the help of our senses we can easily classify and distinguish among them but for a computer, which does not have any sense, we need to train or program it for such a classification or recognition of different objects.

For example:

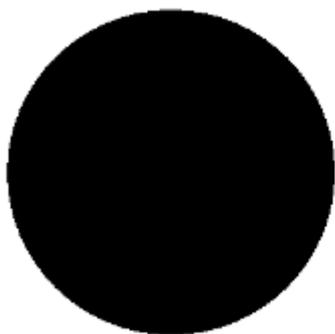


Fig...1

From our intuition we can easily say that the object in fig.1 is a circle or a sphere. But what about a computer, how will it recognize this object .We have to provide it with an artificial intuition to do so. The following chapters provide us with one of the approaches to train a machine like computer to recognize objects.

1.2 PRE-PROCEESSING

The image before it is actually processed needs pre-processing. The image taken from a camera includes noise which introduces some distortions in the image which is needed to be eliminated; so we use some filtering techniques like low-pass filtering for Gaussian noise, median filtering for salt-pepper noise, notch filtering for periodic noise etc.

The next step is to convert the RGB image to gray scale image; after that the gray scale image is converted to binary image by use of thresholding technique.

The process of filtering out the noise involves deriving the properties of the noise i.e. to determine what kind of noise is present in the image. One way of determining the property of noise is to take Fourier transform of the image and then study the spectra. Noises present in the image in general have higher frequency than that of original image. There are different types of noises that distort an image which classified according to their probability density function. Although the salt and pepper noise is present almost in every image acquisition systems.

The process of converting of gray scale to binary image is also important as it depends upon the application i.e. which type of image is being converted to binary; whether it only a certain amount of pixels which are higher gray level to be segmented or a significant amount of pixels which are high gray levels but with variable intensity in different areas of the image.

For the first case we can use global thresholding and for the second we can use adaptive thresholding by processing blocks by blocks in the thresholding operation in the image.

1.3 OBJECT BOUNDARY EXTRACTION

Object boundary detection lies in a very trivial concept that at the boundary of an object there is a sudden abrupt change in the intensity value as we just cross the boundary.

We make use of this elegant concept. What we do is take any pixel and traverse to its corresponding neighbor pixel and see the change in the intensity values during such travel. If the change in intensity is too high, we assign the pixels from which we traverse as the boundary pixels. We can use the derivative function for getting such a result. If the derivative at a point is too high we can assign it as a boundary pixel.

The boundary of an object serves in our purpose to be a major and vital factor for recognition and hence it's efficient and precise detection is highly a necessity. Hereafter the boundary has been detected, now we again apply some morphological operations to make the boundary pixels to be of unit width to avoid any further complications in the processing of these boundary elements.

1.4 MORPHOLOGICAL OPERATIONS

Our objective here is to recognize the shape of the object and hence we are least bothered of the region inside the boundary of the object or outside it. Or it may happen that two or more objects may be present touching each other's boundaries. In those cases we use some morphological operations to separate the objects boundaries or may be filling the holes or any other discontinuities within the region enclosed by the boundaries or outside it to enhance our boundary detection capabilities.

These morphological operations basically used are

1.Dilation

Dilation is generally used for thickening of the object.

2.Erosion

Erosion is generally used for thinning of the object.

3.Opening

Opening is generally used for smoothening the contour of an object or elimination of thin protrusions.

4.Closing

Closing is generally used for filling gaps between close objects.

5.Hit or Miss Transform

Hit or Miss Transform is generally used for detecting particular pattern in the image.

1.5 CORNER DETECTION

1.5.1 HISTORY OF CORNER DETECTION:

The concept of corner detection was first proposed by Moravec [1980]. It actually considers a local window which computes average intensity and the corner is detected if the change in average intensity in every direction high; if the change in average intensity in only one direction is high then it is an edge or if the change in average intensity in any direction is small then the area there is flat.

The average change in intensity proposed $E(x, y)$ is given by

$$E(x, y) = s^2 \cdot W * Fx^2 + 2 \cdot s \cdot t \cdot W * Fx \cdot Fy + t^2 \cdot W * Fy^2$$

Where Fx is the derivative along X-direction and Fy is the derivative along Y-direction and W is averaging window of size $s \times t$.

1.5.2 DETERMINANT METHOD FOR CORNER DETECTION:

Having detected the boundary of an object, the next step that we adopt here is to detect the corners (if any) present in the boundary. The heart of this project lies in the efficient detection of the corners of an object.

Corners may be described as those points in a boundary where the curvature of the boundary is very high. Various elegant techniques are available for a corner detection like ‘Harris-Stephens corner detection technique’, but we here use a different technique to detect the corners.

The corner detection we use is based on basic geometry of a 2D-plane. The process is as follows ,consider three points placed at equal distance from each other on a plane. Let us consider here a figure for a better understanding

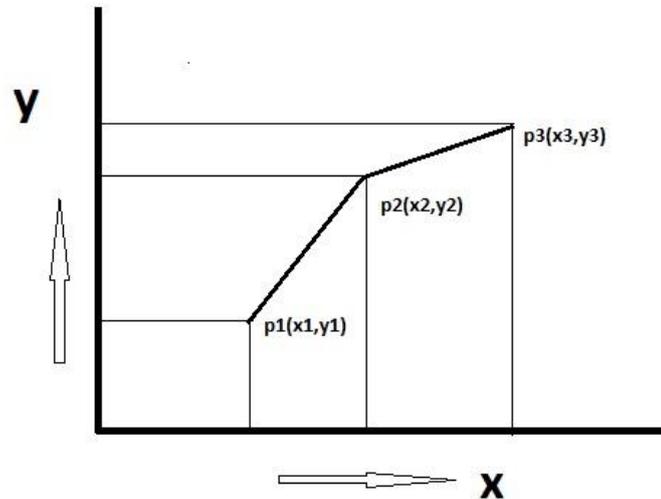


Fig....2

In the above figure the slope of the line p_1p_2 (the line joining the points p_1 and p_2) is given by the following expression:

$$\tan(\Theta_1) = (y_2 - y_1) / (x_2 - x_1).$$

Similarly the slope of the second line p_2p_3 (the line joining the points p_2 and p_3) is given by the following expression:

$$\tan(\Theta_2) = (y_3 - y_2) / (x_3 - x_2).$$

Where Θ_1 and Θ_2 are the angles of the lines p_1p_2 and p_2p_3 respectively with respect to the x-axis.

Now suppose that the two lines are collinear then the slopes are equal and the condition can be extended as determinant of $[x_2 - x_1 \quad y_2 - y_1; x_3 - x_2 \quad y_3 - y_2]$ should be equal to zero.

And if they are not collinear then determinant of $[x_2-x_1 \ y_2-y_1; x_3-x_2 \ y_3-y_2]$ is a non zero value. So the point x_2 is a corner point in case of a regular image like a square or a rectangle etc.

We will use this concept to determine corners for an object given in an image. Let us define a function that is the square of the determinant that is to be calculated in the above mentioned method. We will take boundary points or pixels 3 at a time and then compute the determinant as described below. We will not use the consecutive points but use a specified separation between the points and move across every point on the boundary. This separation is the specified connected distance between pixels in the boundary. We then assign a point to be a corner if the function as described above for a set of three points is a non-zero value and above certain threshold to detect all possible corner points.

Note : Here prior to the corner detection down sampling of the image can serve to be a boon to the detection process, as it would remove any kind of discontinuities or distortions in the boundary of the image. Down sampling is a technique to reduce the size of an image. It is achieved in a simple yet powerful method by rejecting alternate rows and columns of an image if the down sampling is done by an integer factor. Down sampling should be such that it does not tend to reduce any considerable amount of information in the image but remove unwanted discontinuities. Down sampling can also be very useful to normalize an image or say an object as we will see in case of object recognition by chain code approach. Since, without down sampling the chain code will differ for objects of same shape but different size. Thus, down sampling should be used to obtain scaling invariance.

1.5.2 FEATURE EXTRACTION:

A simple question is how to make use of these detected corners for object shape recognition. Various approaches can be made for such a task to perform.

Here we use a simple technique.

Deriving a feature vector from the corner points for recognition of a proper shape or an approximation of an object to proper shape.

Suppose we have 'n' corners detected, then we form a feature vector of nC_2 dimension. The elements of this vector are the all possible distances between any two corner points. Then what we sort this feature vector elements in decreasing order. Now the first element is the length between corners which are the most farthest from each other and the last element is the distance between the corners which are nearest to each other.

Now the challenge is to use the feature vector for a powerful recognition technique.

The first approach is to match the feature vector obtained from the above said algorithm of the test image with the feature vector of the reference image stored in database.

The second approach we follow is a non-matching technique which is helpful in recognition of proper shapes as described earlier or in a proper shape approximation to an object.

We will use some of the examples to have an insight on the use of feature vector.

Suppose the number of corners detected is 4. Now the feature vector would contain 6 elements. Now if all the elements are unequal with some error value permitted, then the object is a quadrilateral. (The assumption here is that the object is a convex polygon). Again if the first two elements are equal i.e. the diagonals are equal and all other elements are equal then it is a square, of course some percentage of error permitted. Similarly if first two

elements are equal and fourth and fifth elements are unequal then it is a rectangle and lastly if first two elements are not equal but all others are equal then it is a rhombus. The use is completely based on the geometrical intuitive of any convex polygon.

Since, the feature vector consists only the Euclidean distance between any two corner points of the object; thus the feature doesn't get affected on rotation and translation; however it does while scaling. To make it rotationally invariant the feature vector is normalized.

Note: A practical problem arises for a circle by corner detection method. By the theory we have presented the number of corners detected in a circle should be ideally zero. But that is not the case; we get a large number of corners for a circle. The reason is quite obvious; we take the determinant method for corner detection. Since none of the set of three points given a separation on the boundary of a circle is collinear, hence the determinant is non zero for a great number of points. So we will apply other techniques like signature and chain code method specifically for ellipse or a circle.

1.6 SIGNATURE

Signature of an object can be described to be a one dimensional mapping of a 2-D object. It is actually the distance profile or the loci of the distance of any point on the boundary of the object to the centre of the object.

Considering this definition of signature, the signature of a perfect circle should be a horizontal straight line or variance of the signature being zero and the mean of the signature is the radius of the circle. However if some error value is allowed then we check for the variance of the signature to be less than a small value and if the condition is satisfied then the object is classified as a circle.

We first find the boundary of the object. Then to make the no of points taken to be standardized for every object we take 360 points each 1 degree apart from each other. Then the distance of ea point from the centre is calculated and plotted. To make it rotationally invariant what we do is the largest value of the signature i.e. the farthest point from the centre in the sequence of signature values is made the starting point without affecting the sequence.

Then to make it scaling invariant we normalize the values of signature.

We can use this for recognition of objects either by matching with the signature of reference objects stored in database or by using the properties of signature for simple objects.

1.7 FREEMAN CHAIN CODE

The chain code of an image is usually calculated from the boundary by sub sampling the boundary and then assigning a specific number i.e.(0-3)for 4 connectivity and (0-7)for 8 connectivity as we travel from one point to the other on the boundary.

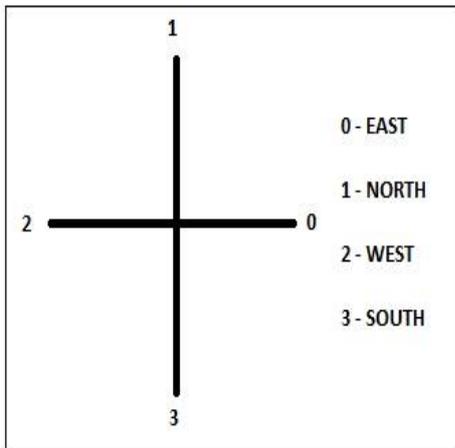


Fig....3

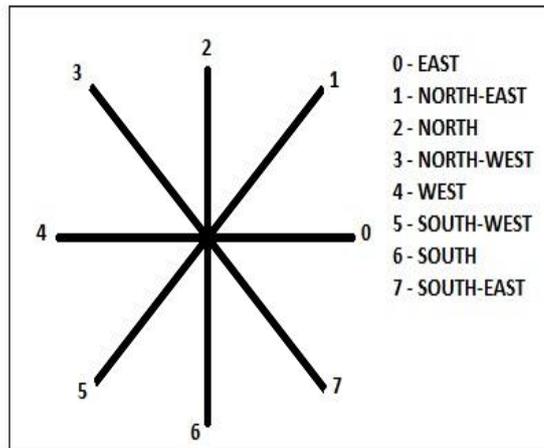


Fig....4

Thus chain code is nothing but the direction of traversal from one point to the other, each direction being specified by a number.

1.7.1 USING CHAIN CODES FOR SHAPE RECOGNITION:

Various techniques are there for such an application like shape number which is a translation, scaling and rotational invariant descriptor (invariance only for multiples of 90 degree rotations.)

But here we use chain code histogram for recognition of shapes.

$$P_k = L_k / N$$

Where $k = 0, 1, \dots, 7$; L_k is the no. of kth code and N is the order of the chain code.

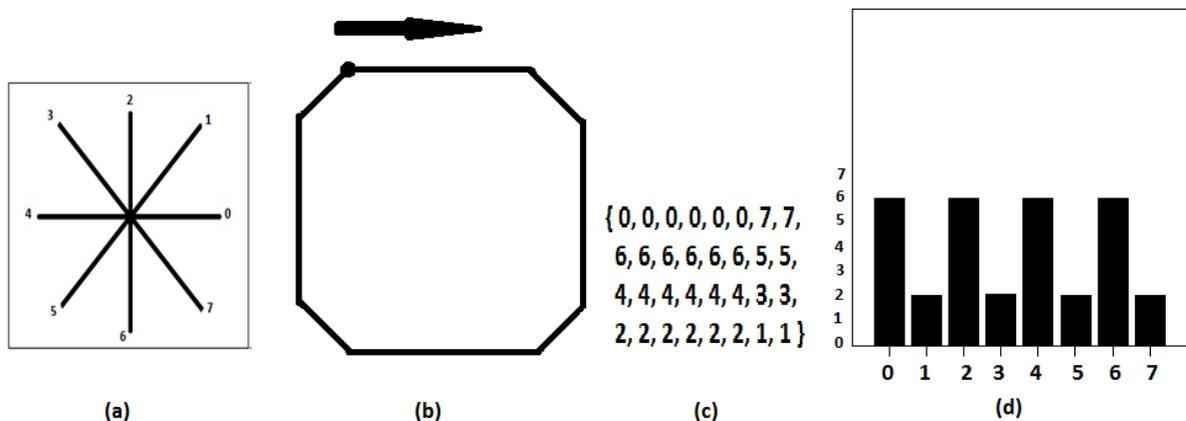


Fig....5

- Representation of directions for 8-connectivity
- Image
- Freeman chain code of the boundary
- Chain code histogram

In this method, we try to match the histogram of the FCC of the test image with the histogram of the FCC of the stored image in database. This method is scaling and translation invariant; however it is not rotational invariant.

2. EXPERIMENTAL WORK

2.1. CORNER DETECTION METHOD

2.1.1 ALGORITHM:

- RGB Image is converted to binary image
- Morphological opening for removal of unwanted objects
- Boundary extraction from the image
- Finding the determinant function for each of the boundary points
- Converting the boundary into image and replacing the gray level with determinant function values
- Converting the Image to binary
- Morphological closing to connect proximal discontinuous pixels
- Extraction of corner elements
- Morphological Hit or Miss to extract corner points
- Deriving the feature vector using the corner points
- Store the feature vector in database for a reference image
- Test for matching of the feature vector of the test image with that of the reference image
- Deriving the relationship between elements of the feature vector to test for simpler shapes

2.1.2 RESULTS AND OUTPUTS:

- The figure shown below is the reference image whose description vector is stored in the database.

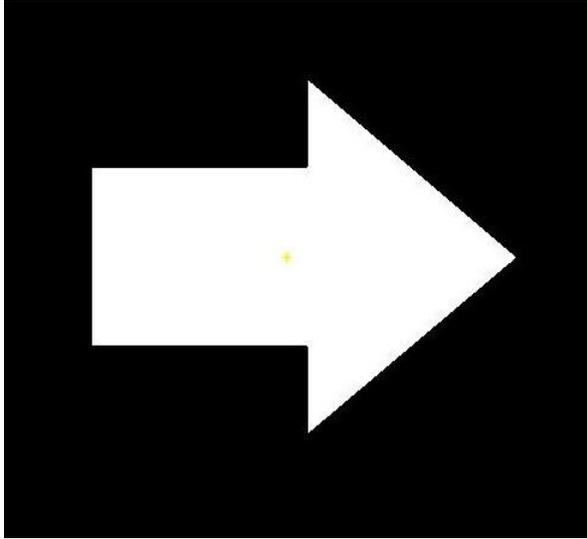


Fig....6



- The image shown below is the test image

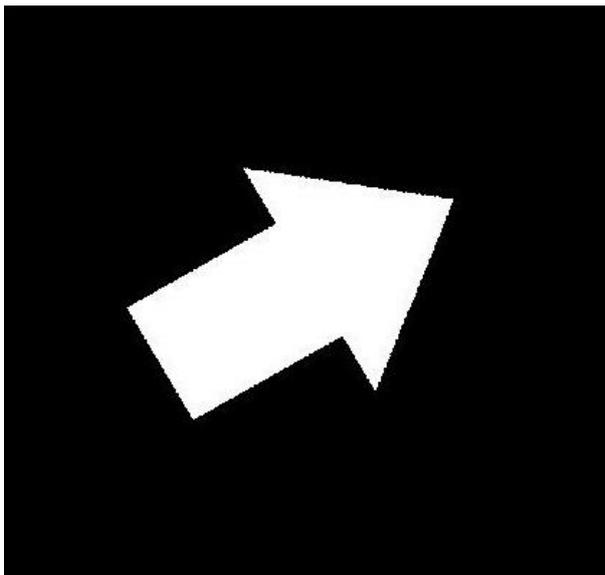


Fig....7



- The figure shown below displays square of determinant function at different boundary points

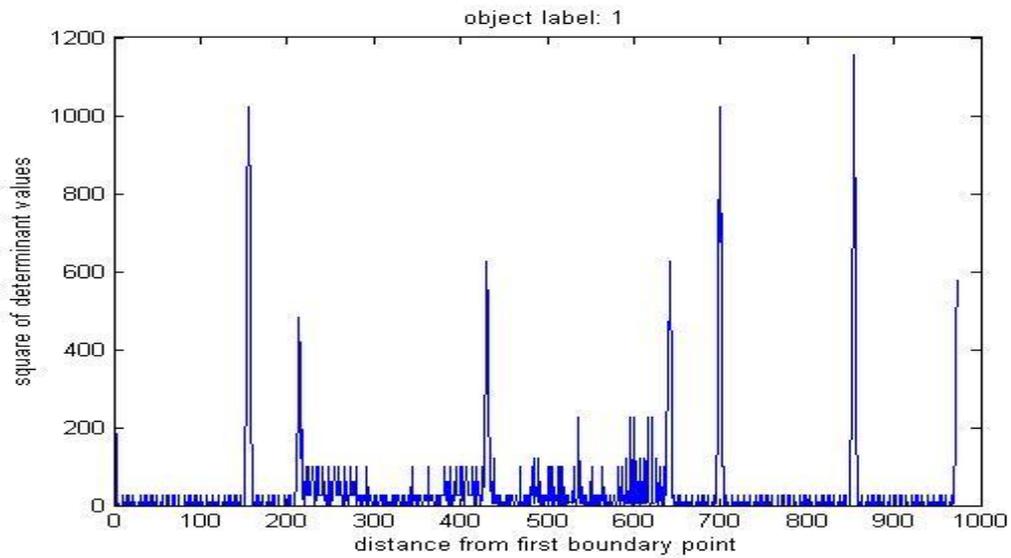


Fig....8

- The major peaks in the graph estimates the no of corners in the object; in this figure the no of corners estimated is 7.

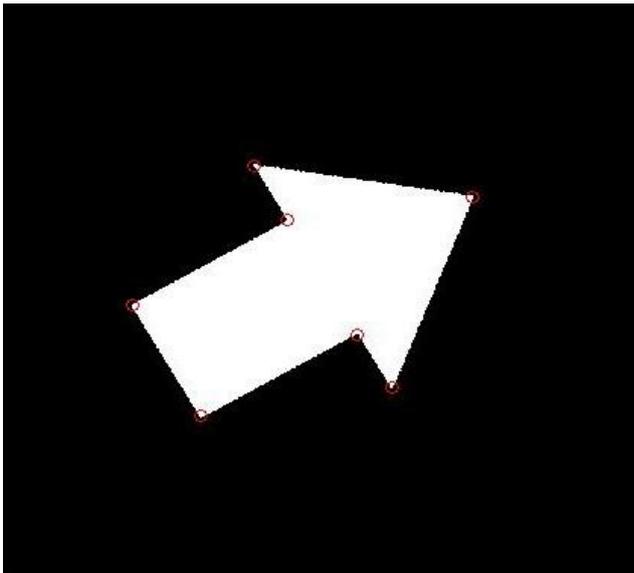


Image showing corner points of the object identified by the small circles

Fig....9

NAME	VALUES
Description vector of Reference Image	[8.1900 8.1800 6.2800 6.2600 6.2600 5.1800 5.1800 5.0200 4.9900 4.7500 4.7500 4.4200 4.4200 4.2700 4.2600 4.0800 4.0800 3.2300 3.1400 1.5200 1.5200]
Description vector of Test Image	[8.2063 8.1833 6.2754 6.2598 6.2278 5.2073 5.1755 5.0309 4.9950 4.7481 4.7449 4.4260 4.4036 4.2688 4.2518 4.1055 4.0998 3.2331 3.1299 1.5152 1.5119]
Error	[0.0163 0.0033 -0.0046 -0.0002 -0.0322 0.0273 -0.0045 0.0109 0.0050 -0.0019 -0.0051 0.0060 -0.0164 -0.0012 - 0.0082 0.0255 0.0198 0.0031 -0.0101 -0.0048 -0.0081]
Mean Square Error	1.8533 X 10 ⁻⁴

Table 1

INFERENCE:

Since the mean square error is very small of the order 10^{-4} , therefore the object best matches with the reference object taken.

- The figure shown below is the reference image whose description vector is stored in the database.

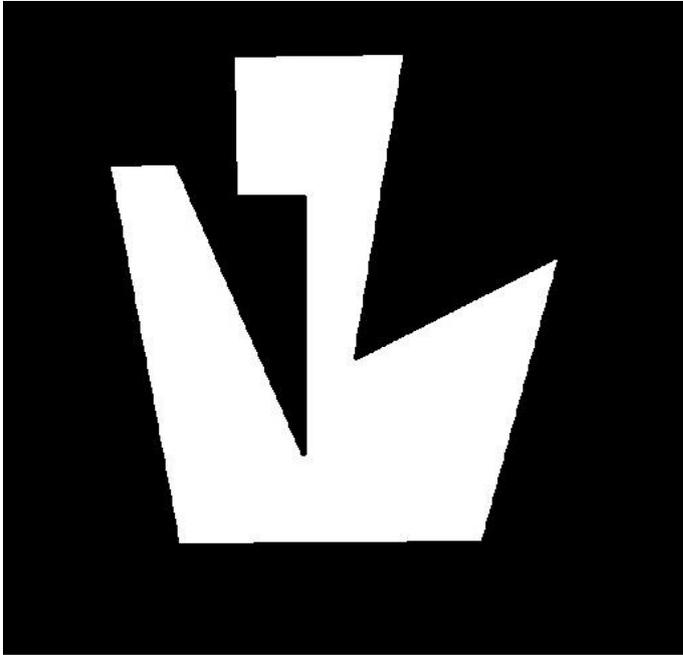


Fig....10

- The image shown below is the test image



Fig....11

The figure shown below displays square of determinant function at different boundary points

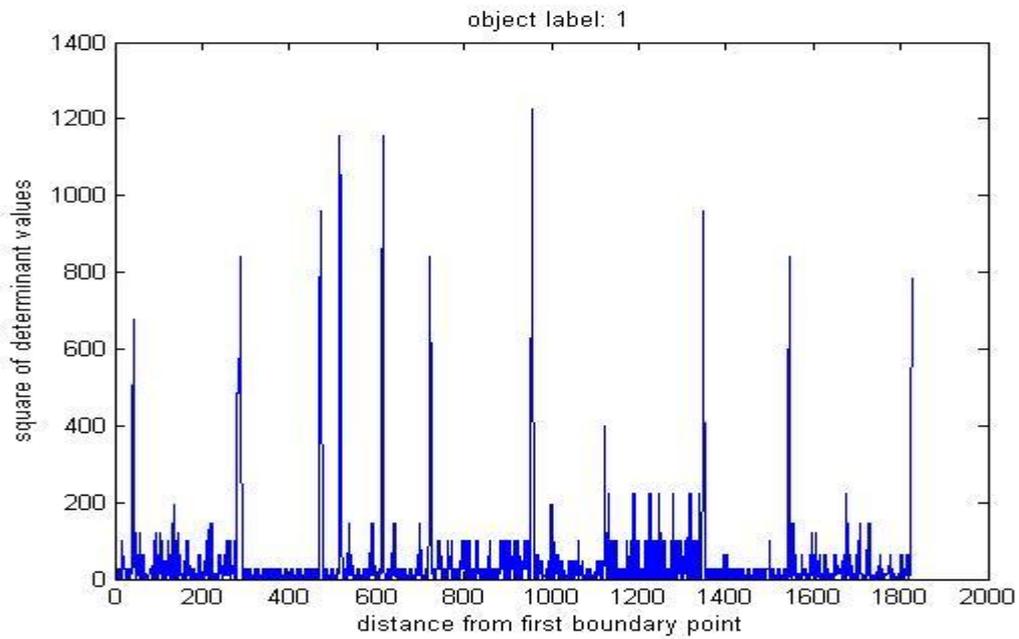


Fig....12

- The major peaks in the graph estimates the no of corners in the object; in this figure the no of corners estimated is 11.



Corner points are shown by
small circles

Fig....13

NAME	VALUES
Description vector of Reference Image	[3.4552 3.4295 3.2835 3.1837 3.1664 3.0427 2.9041 2.7364 2.6802 2.6607 2.6106 2.4787 2.4693 2.4438 2.3713 2.3708 2.3221 2.3024 2.2012 2.0732 2.0293 1.9816 1.9735 1.9491 1.9213 1.8821 1.8750 1.7946 1.7492 1.6824 1.6520 1.6224 1.5955 1.5627 1.5410 1.4003 1.3582 1.3155 1.2799 1.1933 1.1774 1.0927 1.0802 1.0316 0.9859 0.9855 0.9310 0.8734 0.8083 0.7992 0.7873 0.7058 0.4271 0.3933 0.3765]
Description vector of Test Image	[3.4549 3.4244 3.2763 3.1733 3.1709 3.0448 2.9096 2.7380 2.6812 2.6469 2.6090 2.4695 2.4681 2.4483 2.3812 2.3752 2.3253 2.3053 2.1920 2.0817 2.0303 1.9830 1.9675 1.9546 1.9175 1.8817 1.8698 1.8027 1.7476 1.6825 1.6583 1.6201 1.5895 1.5587 1.5345 1.3942 1.3684 1.3094 1.2870 1.1920 1.1845 1.1022 1.0639 1.0420 0.9868 0.9816 0.9426 0.8746 0.8169 0.7997 0.7836 0.6891 0.4263 0.4040 0.3770]
Error	[-0.0003 -0.0051 -0.0072 -0.0104 0.0045 0.0021 0.0055 0.0016 0.0010 -0.0138 -0.0016 -0.0092 -0.0012 0.0045 0.0099 0.0044 0.0032 0.0029 -0.0092 0.0085 0.0010 0.0014 -0.0060 0.0055 -0.0038 -0.0004 -0.0052 0.0081 -0.0016 0.0001 0.0063 -0.0023 -0.0060 -0.0040 -0.0065 -0.0061 0.0102 -0.0061 0.0071 -0.0013 0.0071 0.0095 -0.0163 0.0104 0.0009 -0.0039 0.0116 0.0012 0.0086 0.0005 -0.0037 -0.0167 -0.0008 0.0107 0.0005]
MSE	4.6030 X 10 ⁻⁵

Table 2

INFERENCE:

Since the MSE is very small of the order 10^{-5} , therefore the object best matches with the reference image.

- An approach for recognition of simple images without matching:

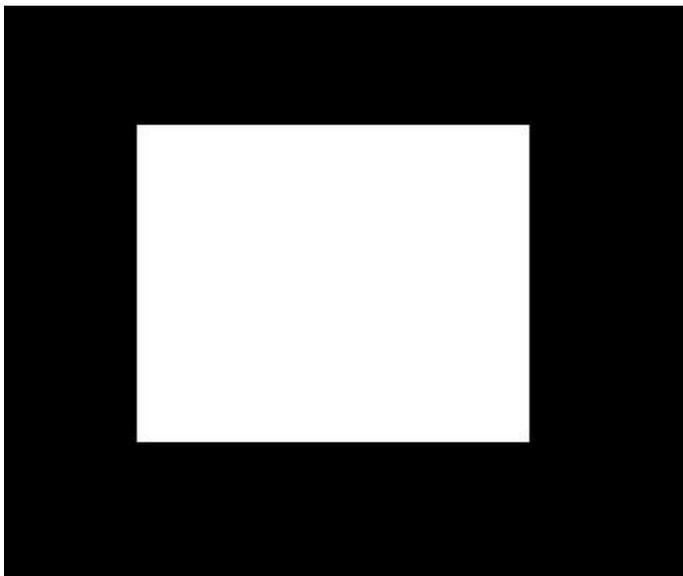
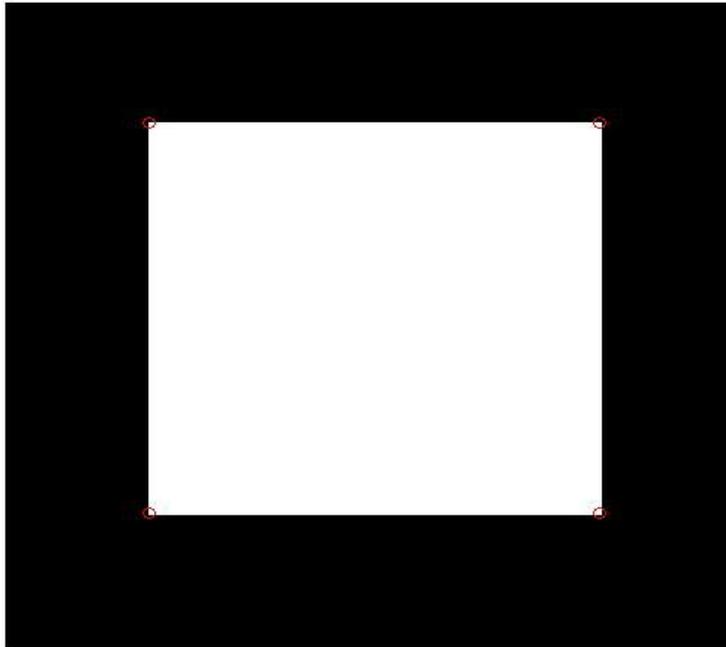


Fig....14

- Image showing corners of the object



Corners can be seen by small circles

Fig....15

Name	value
Test image description vector	[20.7107 20.7107 14.6689 14.6689 14.6204 14.6204]

Table 3

INFERENCE:

As it can be seen that the first two elements which represent the diagonals are equal and the last four elements which represent the sides are also almost equal, the test image is a square by the fundamental properties of a square.

2.2 SIGNATURE METHOD

2.2.1 ALGORITHM:

- RGB Image is converted to binary image
- Morphological opening for removal of unwanted objects
- Find the centre of mass of the object
- Boundary extraction from the image
- Find the distance of boundary points from the centre of mass which are 1 degree apart from each other and form a vector of it
- Normalize the vector formed to make signature scaling invariant
- The maximum value in the sequence of vector is shifted to the starting point while maintaining the sequence to make it rotationally invariant
- Using the loci of distance above mentioned as the descriptor for image recognition either by matching this descriptor with the descriptors stored in the database or using the properties of simple geometrical shapes without matching.

2.2.2 RESULTS AND OUTPUTS:

- The image given below is a test image:

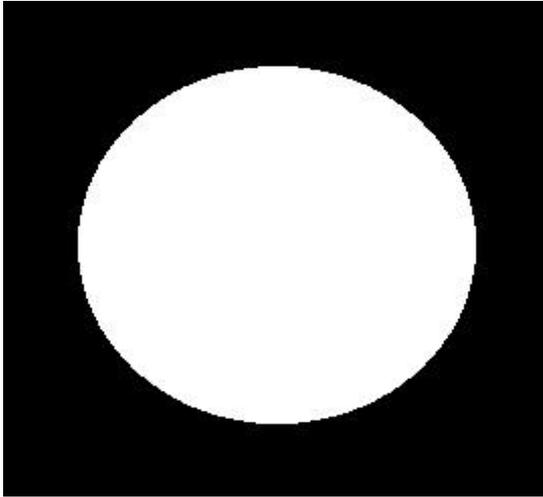


Fig....16

The plot of the signature of the image:

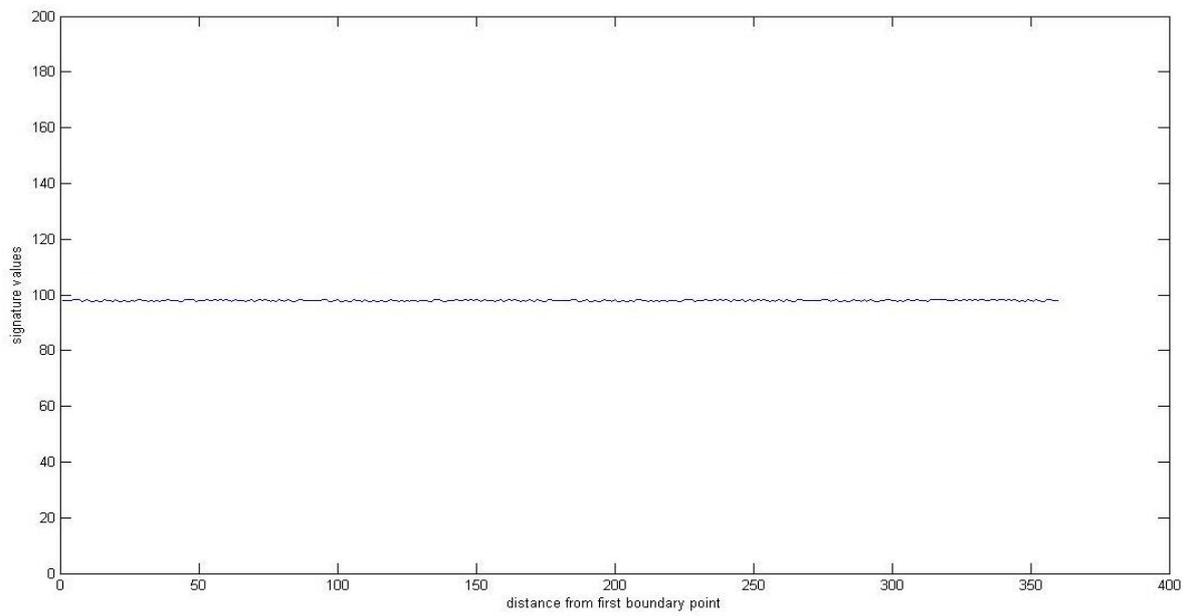


Fig....17

Variance of the signature is 0.0752.

INFERENCE:

The signature forms almost a straight line depicting that the distance of the boundary points from the centre of mass is almost constant; hence the object in the test image is a circle from the basic properties of a circle.

- The following image is another test image:

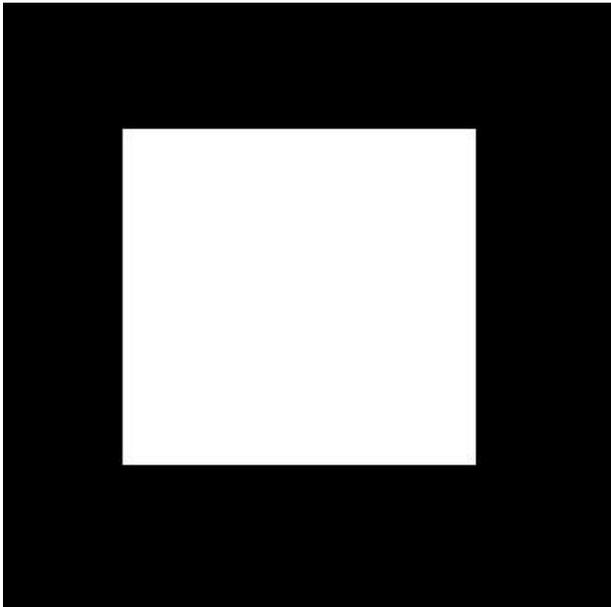


Fig....18

- The plot of signature for the test image

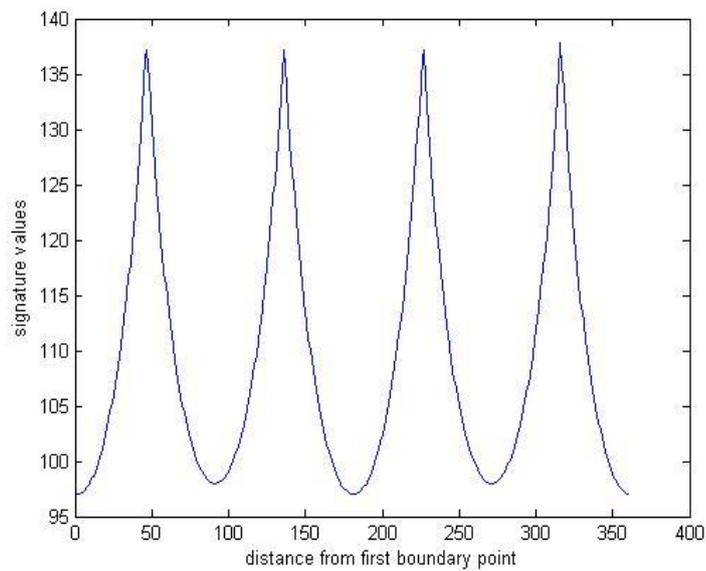


Fig....19

INFERENCE:

As can be seen from the plot of the signature that there are four peaks and all the maximum and minimum values are same which represent that there are four edges and the distance of the four corners from the centre is same (four same max values) and all the sides are equal (from the four same minimum values), therefore it is concluded that the object in the image is a square.

The image below is a reference image whose signature is stored in database:

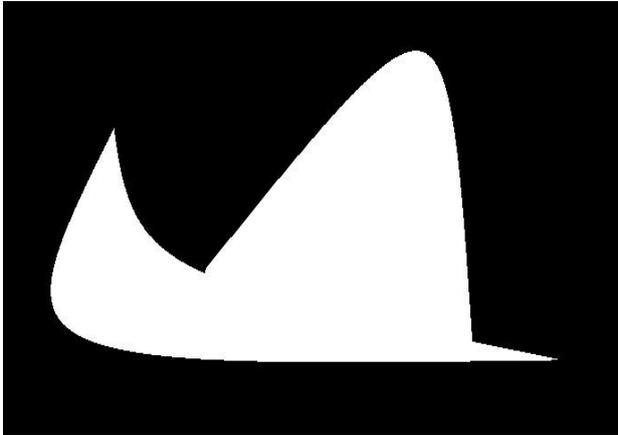


Fig....20

- The image below is a test image to be matched with the reference image

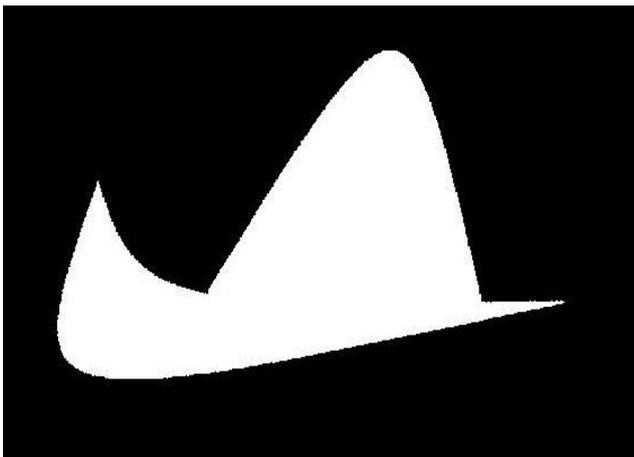


Fig....21

- The figure below shows the signature value at different angles from the centre for the reference image:

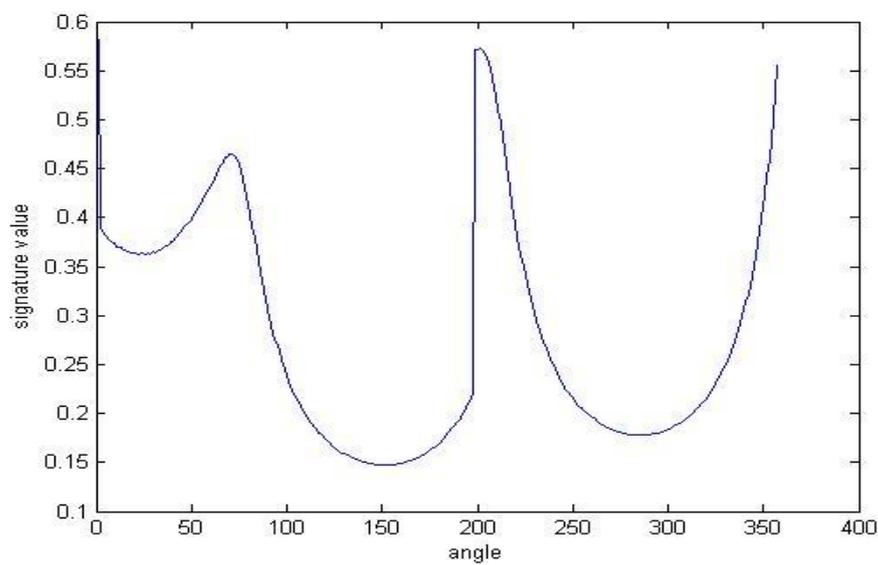


Fig....22

- The figure below shows the signature of the test image at different angles:

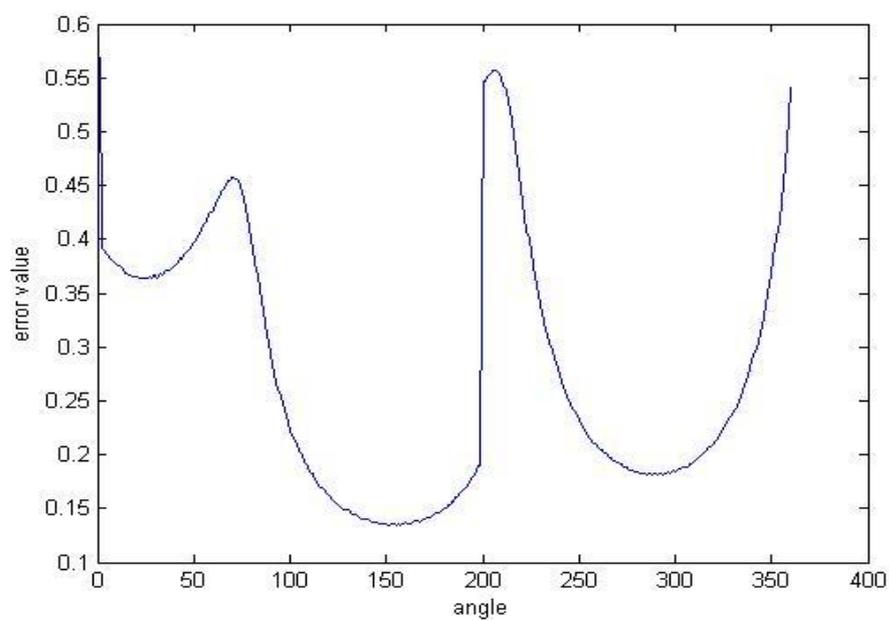


Fig....23

- The figure below shows the error value at different angles:

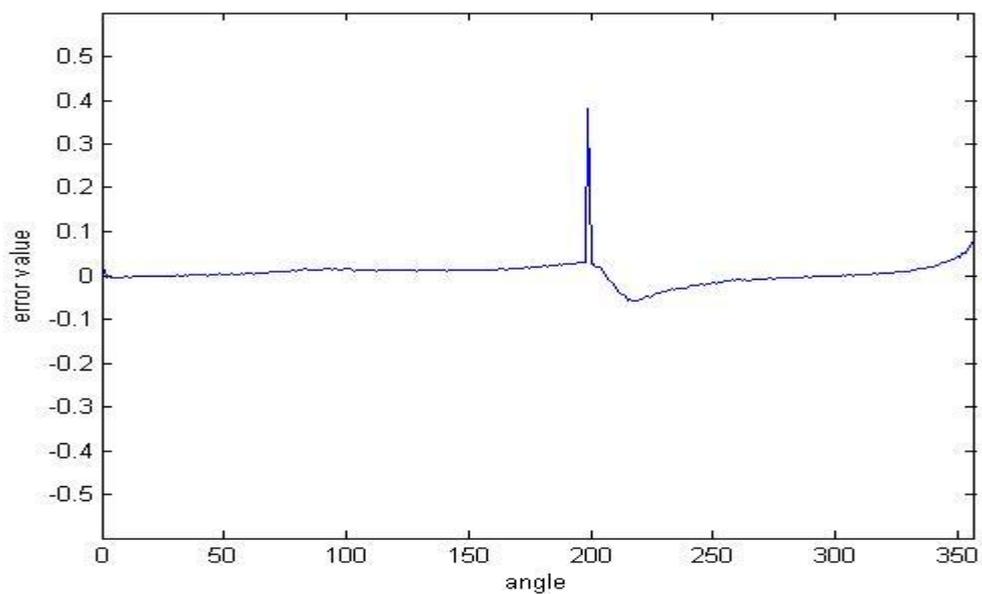


Fig....24

Variance of the error is 0.00078

MSE=6.1561X10⁻⁷

INFERENCE:

Since the variance and mean square error are very less, therefore the test image best matches with the reference image.

2.3 CHAIN CODE APPROACH

2.3.1 ALGORITHM:

- RGB Image is converted to binary image
- Morphological opening for removal of unwanted objects
- Boundary extraction from the image
- The major axis of the boundary is aligned with the x axis to make the chain code approach rotationally invariant.
- The boundary is then sub sampled to get fewer number of points on the boundary and then the sub sampled image is normalized so that the distance between each sub sampled boundary points is unity. Then the points are connected.
- The chain code is then calculated using 8 connectivity.
- The histogram of the chain code is calculated and then is compared and matched to the histogram of the chain codes of the reference images stored in the database.
- The error and mean square error is calculated .
- The object best matches with the object stored in the database for which the mean Square error is the least.

2.3.2 RESULTS AND OUTPUTS:

- The image given below is a reference image whose histogram of the chain code is stored in the database:

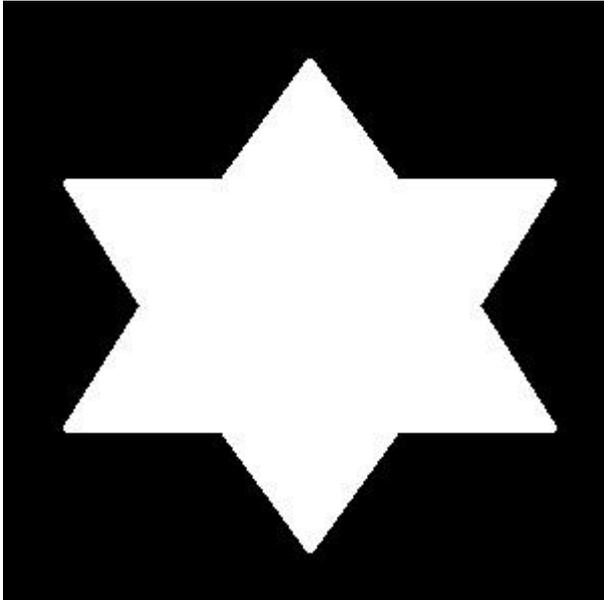


Fig....25



- The image given below is the test image:

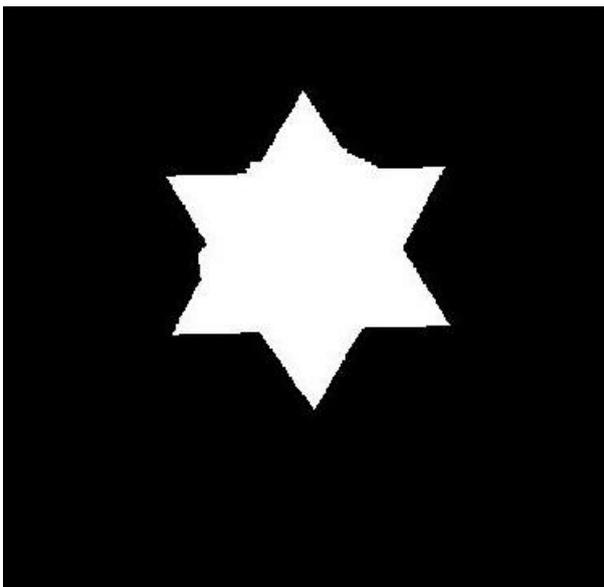


Fig....26



Name	values
Histogram of chain code of reference image	[0.4977 0.2884 0 0 0 0 0.0047 0.1023 0.1070]
Histogram of chain code of test image	[0.4706 0.2941 0.0049 0 0.0098 0.0049 0.0735 0.1422]
Error	[0.02 - 0.005 0.005 0 -0.01 0.002 0.02 - 0.035]
MSE	3.6943×10^{-4}

Table 4

INFERENCE:

Since the mean square error is very small of the order 10^{-4} , therefore the test image best matches with the reference image.

- The image is another image whose histogram of the chain code is stored in the database

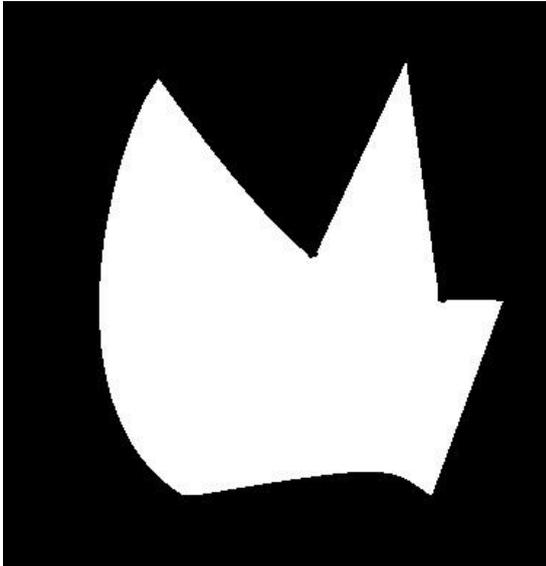


Fig....27



- The test image which is to be matched with the reference image

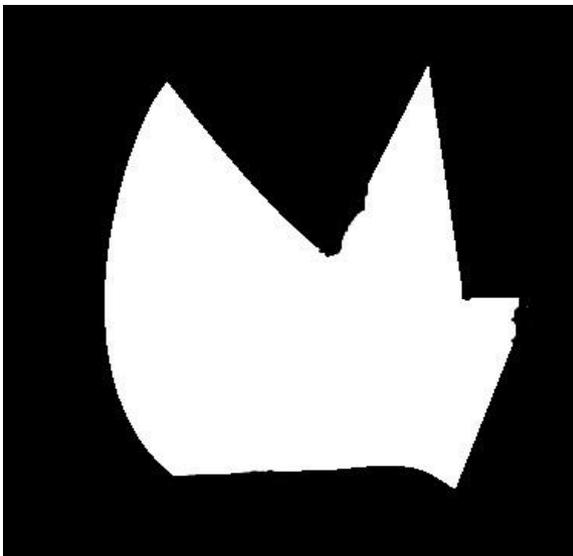


Fig....28



Name	Values
Histogram of chain code of reference image	[0.536 0.252 0.0045 0 0 0.0045 0.0811 0.1216]
Histogram of chain code of test image	[0.533 0.2578 0.0044 0 0.0044 0.0044 0.0756 0.1200]
Error	[0.0027 -0.0058 0.0001 0 -0.0044 0.0001 0.0044 0.0016]
MSE	1.0321×10^{-5}

Table 5

INFERENCE:

Since the mean square error is of order 10^{-5} , which is very low, the test image best matches with the reference image.

3. CONCLUSION

In this thesis, we studied and implemented successfully various techniques of object recognition. The techniques are briefly discussed and ample number of examples are given for proper understanding of the techniques. The corner detection method serves to give a good approximation for a number of objects but the constraint being that it can recognize objects by matching only if the number of corners detected in the test image and that of present in the reference image are same. If they are different then different results are obtained. However with these constraints the recognition is quite good with a very low mean square error of the order 10^{-4} to 10^{-5} .

The signature method can be aptly used for recognition even if the above constraints are not satisfied. With a 360 signature values feature vector the recognition was quite of a good quality.

The signature thus serves to be the best recognizer feature vector with an MSE of order 10^{-7} .

The chain code approach also gave a very good result. It has no constraints to be satisfied and yet gives good results with an MSE of order 10^{-5} .

All the approaches were independently and clearly studied and results are shown, which is in evidence of the above said statements.

4. REFERENCES

- Gonzalez and Woods.[2009]. Digital Image Processing, Pearson Publications.
- Gonzalez, Woods, and Eddins.[2010]. Digital Image Processing with Matlab, Tata McGraw Hill Publications.
- Sonka , Hlavac and Boyle.[1998]. Image Processing, Analysis and Machine Vision, CL-Engineering Publications.
- Jusoh, Nor Amizam and Zain , Jasni Mohamad.[2009]. “Application of Freeman Chain Codes: An Alternative Recognition Technique for Malaysian Car Plates,” IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.11
- B.Jahne, Digital Image Processing, Springer, Edition 6th, Berlin. 2005
- Freeman H,Computer Processing of Line-Drawing Images,CAN Computing Surveys,Vol.6,N o.1,1974.
- Rosin, P.L., 1996. Augmenting corner descriptors. Graphical Models and Image Processing 58, 286-294.
- Harris C and Stephens M J. A combined corner and edge detector. In Alvey Vision Conference, Manchester, 1988: 147-152.
- Sohn, K., Alexander, W.E., Kim, J.H., Snyder, W.E., 1994. A constrained regularization approach to robust corner detection. IEEE Transactions on Systems, Man, and Cybernetics 24 (5), 820-828.