A Statistical Framework for Scale, Rotation and Translation Invariant Object Recognition

Muhammad Faisal Hayat
mf_hayat@hotmail.com

Syed Muhammad Ahsan
s_m_ahsan@hotmail.com

,

University of Engineering and Technology Lahore, Pakistan

Abstract *The process of feature extraction is fundamental to object recognition process, The paper proposes techniques for computing boundary and region descriptors, which are efficient and work well even in the presence of noise. For boundary descriptors, shape is extracted through morphological algorithms and boundary of the object is traced, after they have been suitably binarized and noise has been removed. Boundary obtained in this way is manipulated to obtain boundary descriptors like chain code, shape numbers, Fourier descriptors applied on signatures (both distance from centroid and coordinates), moments and other scalar descriptor. Similarly region descriptors like moments, skeleton and scalar descriptors are calculated. The statistical patterns obtained from these descriptors are matched against already calculated patterns stored in database for recognizing object, where pattern is a set of descriptors. The patterns proposed in the paper are invariant to scaling, rotation and translation transformation.*

Keywords Boundary Descriptors, Morphological algorithms, Binarization, chain codes, shape numbers, scalar descriptors

1. Introduction

There are two important issues in recognition of the object. First is how to find it out what an object is. A kid asks his mother about an object after seeing an object, when he is told, he records it for future recognition. We can say that object is being learnt. Second is how the object is matched with the learnt object and how the brain stores the learnt object. Different models have been proposed for this purpose. Most important of them is the mathematical model of the object. The brain stores the mathematical model of the object and when it sees an object it calculates the mathematical model and matches the models to determine what an object is. This leads us to an important question as to what a mathematical model is like. It is vital that mathematical model should be independent of any distortion of the image for example if a chair's leg is broken brain still interpret it as a chair, also it should be insensitive to rotation, scale and translation changes. So recognizing an object generally converges to finding suitable mathematical model of an object. Such a mathematical model is proposed in this paper.

Since shape is a fundamental property of an object and an important low level feature to human perception, an effective mathematical model contains shape descriptors. There are generally two types of shape descriptors. Boundary based shape descriptors and region based shaped descriptors. Boundary based shape descriptors such as fourier descriptors, curvature scale shape and shape signature exploits only boundary information they cannot deal with disjoint shapes where boundary may not be available, therefore, they have limited applications but they are extremely computational efficient since small number of points are to be processed. If small number of points can help in recognizing object, other points can easily be ignored. On the other hand in region based techniques, all the points within a shape region are taken into account to obtain the region descriptors. Common region based descriptors employs moment descriptors, texture analysis and skeletonizing techniques. Grid method to describe shape is attracting interest for its simplicity in representation. Since region based shape representation combine information across an entire object rather than exploiting information just at boundary points, they can capture interior information in a shape. Other advantages of region based methods are that they can be employed to describe disjoint shape and robust to shape distortions but it may requires more computation as compared to boundary based shape descriptors, a property not required for real time systems.

In this paper different boundary and region based descriptors involve in object recognition process and object recognition process itself is discussed. Section 2 describes contour based descriptors, Section 3 relates to region based descriptors, Section 4 proposes the process of object recognition, Section 5 shows the experimental results, Section 6 concludes the paper. In section 7, future works are discussed and in section 8, references are given.

2. Contour based shape descriptors

This section describes four contour based shape descriptors i.e 1. Chain codes, 2. Signatures, 3. Fourier descriptors, 4. Scalar descriptors.

2.1 Chain Codes

Rather than storing the absolute information of all the points in the object, chain code stores relative information of all the points, which yields translational invariant representation, [1][2][4]. If contour is 4-connected we only need four values to encode the direction of next pixel but if contour is 8 connected we need eight numbers. The representation would only help in object recognition if same starting point is selected. If starting point is different we have to rotate the code to different starting point. Small variations in contour give different chain codes. Also the representation is not rotational invariant. This rotational variation can be dealt with encoding not only relative direction, but difference in the successive directions i.e differential chain codes. Similarly starting point rotates the code. If we rotate an n-element code so that it has the smallest value when viewed as an n digit integer, which would be invariant to the selection of starting point. Such normalized differential chain code is called as shape number. Figure 1 (a) shows the actual object, (b) shows the boundary extracted of the object, the boundary extraction algorithms are discussed later in this paper. The object is drawn on grid as shown in (c), in (d) the boundary points are expanded to cover full cell area in order to account for the noise in the object's boundary. In (e), new set of coordinates are selected based upon filling of the entire cell. Now chain codes, differential chain codes and shape numbers can extracted from (e). But still chain codes obtained will be variant to scale. We can try to normalize away these effects by re-sampling grid along some of the principal axis of the shape. This is shown in figure 1 (f) which shows the basic rectangle around the object. The figure 1 (g) shows the grid generated within the basic rectangle. This dynamically generated grid can account for rotation, resizing and minor point distortion. The points extracted in this way might not be connected as show in (h), so application of chain codes and shape numbers might lead to complex algorithms. The grid shape description approach [3] is used, in which grid cells are assigned the value of 1 if they are covered by the shape and 0 if they are outside the shape. A shape number consisting of binary sequence is created by scanning the grid left-right and top-bottom order and this binary sequence is used as shape descriptors to index the shape. Grid description is a straight forward shape representation which may be suitable for shape coding as is adopted in MPEG-4. However, for retrieval purposes, it is unquestionable, because a slight shape distortion, such as affine transform can cause very big differences in the similarity measurement. Since normalization are mainly based on major axis and eccentricity, shapes otherwise similar may be treated as different due to normalization.

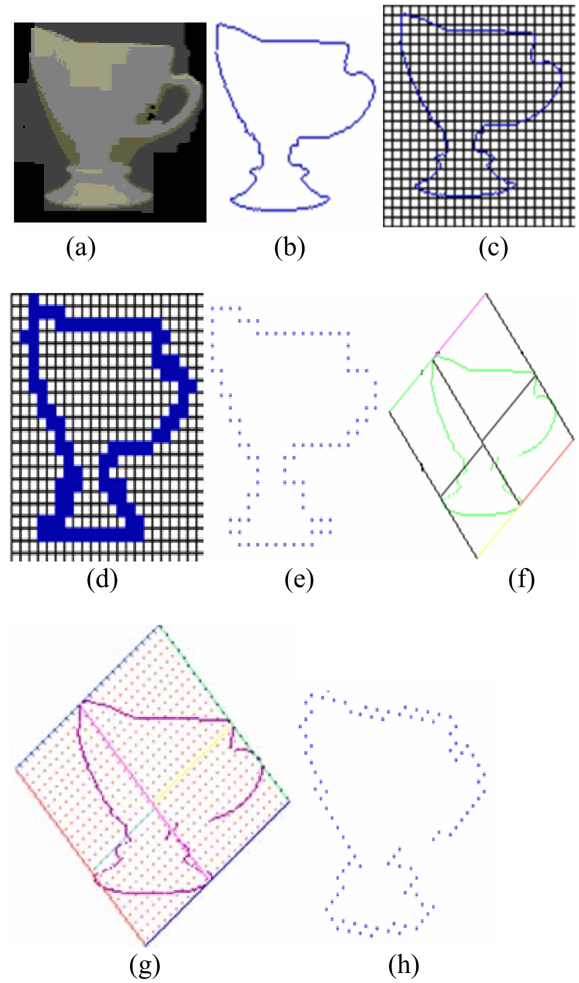


Figure 1

The algorithm proceeds as follows

- Calculate the major and minor axis of the object
- Calculate the basic rectangle of the shape
- Generate the 20*20 grid inside the rectangle
- Calculate chain code, chain code difference and shape numbers now or scan through the rectangle's cells, mark it 0 if object is not present in the cell else mark it as 1.

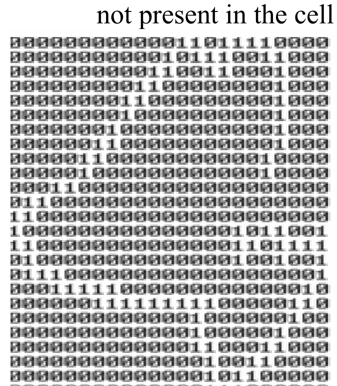

00000000000011011110000
00000000000101110011000
00000000001100110001000
00000000011000000001000
00000000110000000001000
00000000100000000001000
00000001000000000001000
00000011000000000001000
00000110000000000010000
00000100000000000010000
00011000000000000000000
01100000000000000000000
11000000000000000000000
10000000000000001011001
11000000000000001101111
01000000000000001001001
01110000000000000000001
00011111000000000000010
00000011111111100000110
00000000000001000000100
00000000000001000001000
00000000000001100011000
00000000000000100110000
00000000000000101100000
00000000000000111000000

Figure 2

The result of the above algorithm is show in figure 2, which shows the bit sequence of 0's and 1's, which can be encoded to obtain boundary descriptor. For example bit sequence 00000000 00001101 11100000…… is the descriptor of mug shown above. This can also be a valuable region descriptor. The technique of obtaining sequence of number obtained in this way is gaining popularity.

2.2 Signatures

In general, a shape signature is any 1-D function representing 2-D area or boundaries, [1][6]. There are a number of ways to generate signatures. One of the simplest is to plot the distance from the centroid to the boundary as a function of angle. Similarly complex coordinates (position coordinates), curvature and cumulative angular function can be used as signatures. In this paper, we have used signatures based on distance from the centroid to the boundary as a function of angle. The centroid distance approach is expressed by the distance of the boundary points from the centroid (xc,yc) for the shape.

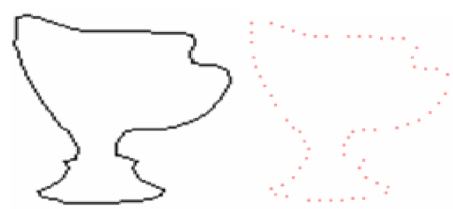
$$R(t) = [x(t) - xc]^2 + [y(t) - yc]^2$$

Due to the subtraction of the centroid, which represents the position of the shape, from boundary coordinates, the centroid distance representation is invariant to translation. Signatures generated by this approach are obviously dependent on size and starting point. Size normalization can be achieved simply by normalizing distance curve to say, unit maximum value. The starting point problem can be solved by always selecting the same point for example on the major axis. Once signature has been obtained, we are still faced with the problem of describing it in a way that will allow us to differentiate between signatures corresponding to different boundary shapes. This problem, however, is generally easier because we are now dealing with 1-D function. An approach often used to characterize a signature is to compute its moments.

2.3 Fourier Descriptors

Fourier transformation can be applied on the signatures obtained in section 2.2, [1][2][3][6]. The Fourier transformation applied on the signatures would yield Fourier descriptors of the shape. These coefficients actually describe shape of the object in the frequency domain contrary to the signatures which describe the shape of the object in the time domain. The lower frequency domain descriptors describe the general shape of the object while higher frequency domain descriptor describes the finer details of the shape of the object. The number of descriptors can vary as a small number of descriptors might be enough for discriminating and recognizing shape. The descriptors obtained in this way are still variant to translation, scale and rotation but the descriptors can be easily manipulated to obtain translation, scale and rotation invariants. In the time domain actual object data need to be changed before calculation of the descriptors which was difficult.

Before applying Fourier transformation on the shape's signature, shape need to be first sampled, i.e it should consist of some fixed number of points for matching in recognition process, [6]. As the object's descriptors are to be matched to model's descriptors as described in introduction, both should consist of same number of data points. In this paper, number of points is reduced to 64, generally power of 2 is selected for applying FFT. The result of reducing number of points to 64 is shown in figure 3 (b), while (a) shows the actual shape of the object. The process not only normalizes the size of the shapes but also has the effect of smoothing the shape which has the effect of eliminating the noise in the shape boundary and also the small details along the object boundary.



(a) (b)

Figure 3

Equal point sampling is used, to obtain sampled set 64 points on which Fourier transformation is applied. The equal point sampling selects candidate points spaced at equal number of points along the shape boundary. Other techniques can be applied for example equal angle sampling or equal arc length encoding etc.

After sampling points are obtained, one can apply Fourier transformation in two ways, one is to apply Fourier transformation over signature of the points i.e distance of each point from the centroid as explained in last section. In the paper DFT by correlation is used to calculate Fourier descriptors, [8].

$$Re[a(u)] = \sum_{k=0}^{K-1} s(k)\cos(2\pi ku/K)$$

$$Im[a(u)] = -\sum_{k=0}^{K-1} s(k)\sin(2\pi ku/K)$$

where s(k) is time domain signal and a(u) is frequency domain signal.

The second is to view the boundary as complex plane, then each 2-D boundary point (x,y) is reduced to the 1-D complex number i.e x+ij. The sequence of points along the boundary forms a function on which complex discrete fourier transform know as complex DFT can be applied to get the fourier descriptors. In the paper complex DFT by correlation is used, [1][8].

If $s(k) = x(k) + jy(k)$, DFT of $s(k)$ is $a(u)$ i.e

$$a(u) = 1/K \sum_{k=0}^{K-1} s(k)\, e^{-j\,2\pi uk/K} \quad u,k = 0,1,2,\ldots..K-1$$

Since signatures used in Fourier transformation process are invariant to translation, the resulting Fourier descriptors obtained will also be invariant to translation. Rotation invariance in both cases can be achieved by ignoring the phase information and by taking only the magnitude of the values in Fourier descriptors. In case of complex coordinate signature, all the N descriptors obtained except the first one (DC component) are needed to index the shape. The DC component depends only on the position of the shape, it is not useful in describing shape therefore can be discarded. Scale normalization is achieved by dividing the magnitude values of all the other descriptors, by the magnitude value of the second descriptor. For centroid distance signatures, there are only N/2 different frequencies in the Fourier transform, therefore only half of the fourier descriptors are needed to index the shape, scale invariance is achieved by dividing the magnitude of values of first half of frequency descriptors by the DC component.

2.4 Scalar descriptors

Apart from these descriptors, length, diameter, eccentricity, chord distribution, curvature and bending energy are also useful descriptors involved in object recognition process, [2].

3. Region based shape descriptors

In this section, three region based shape descriptors i.e 1. Moments, 2. Skeleton, 3. Others are discussed.

3.1 Moments

Region moment representation interprets a normalized gray level image function as a probability density of a 2D function, [1][2]. Properties of the random variable can be described using statistical characteristics moments. Assuming that non zero pixel values represent regions, moments can be used for binary or gray level region description. A moment of order (p+q) is dependent on scaling, translation, rotation and even on gray level transformation and is given by

$$m_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q f(x,y)\, dxdy$$

in a digitized image we evaluate sums

$$m_{pq} = \sum_x \sum_y x^p y^q f(x,y)$$

where x,y,i,j are the region point coordinate in digitized images. The central moments are defined as

$$u_{pq} = \sum_x \sum_y (x-x')^p (y-y')^q f(x,y)$$

where $x' = m_{10}/m_{00}$ and $y' = m_{01}/m_{00}$.

Translation invariance can be achieved if use the central moments. The central moments of order up to 3 are:

$$U_{00} = m_{00}$$
$$U_{02} = m_{02} - y'm_{01}$$
$$U_{10} = 0$$
$$U_{30} = m_{30} - 3x'm_{20} + 2x'^2 m_{10}$$
$$U_{01} = 0$$
$$U_{03} = m_{03} - 3y'm_{02} + 2y'^2 m_{01}$$
$$U_{11} = m_{11} - y'm_{10}$$
$$U_{20} = m_{20} - x'm_{10}$$
$$U_{21} = m_{21} - 2x'm_{11} - y'm_{20} + 2x'^2 m_{01}$$
$$U_{12} = m_{12} - 2y'm_{11} - x'm_{02} + 2y'^2 m_{10}$$

The normalized centralized moments denoted by n_{pq} are defined as

$$n_{pq} = U_{pq} / U_{00}^{\gamma} \quad \text{where}$$

$$\gamma = (p+q)/2 + 1 \qquad \text{for } p+q = 2,3,\ldots\ldots$$

A set of seven invariant moments can be derived from the second and third moments

$$O_1 = n_{20} + n_{02}$$
$$O_2 = (n_{20} - n_{02})^2 + 4n^2_{11}$$
$$O_3 = (n_{30} - 3n_{12})^2 + (3n_{21} - n_{03})^2$$
$$O_4 = (n_{30} + n_{12})^2 + (n_{21} + n_{03})^2$$
$$O_5 = (n_{30} - 3n_{12})(n_{30} + n_{12})[(n_{30} + n_{12})^2 - 3(n_{21} + n_{03})^2] + (3n_{21} - n_{03})(n_{21} + n_{03})$$
$$O_6 = (n_{20} - n_{02})[(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2] + 4n_{11}(n_{30} + n_{12})(n_{21}+n_{03})$$
$$O_7 = (3n_{21} - n_{03})(n_{30} + n_{12})[(n_{30} + n_{12})^2 - 3(n_{21} + n_{03})^2] + (3n_{12} - n_{30})(n_{21} + n_{03}) [3(n_{03} + n_{12})^2 - (n_{21} + n_{03})^2]$$

These set of moments are invariant to translation, rotation and scale change.

3.2 Skeletons

An important approach for representing the structural shape of a plane region is to reduce it to a graph. This is often accomplished by obtaining the skeleton of the region via a thinning (also called skeletonizing) algorithm, [1]. Thinning procedures play a central role in a broad range of problems in computer vision, ranging from automated inspection of printed circuit boards to counting of asbestos fiber on air filters. In this paper we have used morphological algorithm for extracting the skeleton of the shape. The algorithms uses medial axis transformation technique (MAT).

(a) (b)

Figure 4

The result of applying MAT algorithm on figure 4 (a) is show in figure 4 (b). How can the skeleton of the image helps in shape description? Skeleton of image can account for some distortion in the shape, a change in the appearance of the object would not change the structure of the skeleton too much. So skeleton can act as valuable descriptor, but it is dependent upon scale, translation and rotation transformation. The problem can be solved by considering skeleton as the object itself and applying descriptors over the skeleton to fine tune the recognition process, the technique has a shortcoming that considering only skeleton will ignore certain details in object, on the other hand considering both skeleton and object will not be computational efficient.

3.3 scalar descriptors

Apart from these descriptors, some scalar descriptors like area, perimeter, compactness, dispersion, Euler numbers and eccentricity can be very useful in object recognition process, [1][2].

The area of the region is defined as the number of pixels in the region. Each pixel may be taken into consideration to get the real size of a region. If an image is represented as a rectangular raster, simply counting of region's pixels will provide its area. In general, the area of a region in the plane is defined as

$$A(s) = \sum_{x}\sum_{y} I(x,y)\,\Delta A$$

where I(x,y) = 1, if the pixel is within a shape, and 0 otherwise. Area changes with scale changes, however, it is invariant to rotation, small errors in the computation of the area will appear when applying a rotation transformation due to discretization of image.

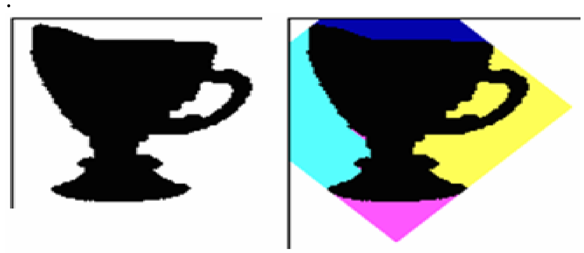
Similarly perimeter of a region is the length of its boundary. Compactness is a dimensionless quantity and thus is insensitive to uniform scale changes and is minimal for a disk shaped region. Compactness is insensitive to orientations. It is square of perimeter divided by area. Dispersion is measured as the ratio major chord length area, defined as

$$I(s) = \frac{\prod \max[(x_i - x')^2 + (y_i - y')^2]}{A(s)}$$

A property useful for region descriptor is the number of connected components. The number of holes H and connected components C can be used to define euler number E: i.e $E = C - H$

The value of diameter and the orientation of a line segment connecting the two extreme points that comprises the diameter (this line is called the major axis of the boundary) are useful descriptors. The minor axis of a boundary is defined as the line perpendicular to the major axis, and of such length that a box passing through the outer four points of intersection of the boundary with the two axes completely encloses the boundary. The box is called basic rectangle and the ratio of major to minor axis is called the eccentricity of the boundary. In this paper, morphological convex hull algorithm is used, to get the basic rectangle of the object, [1]

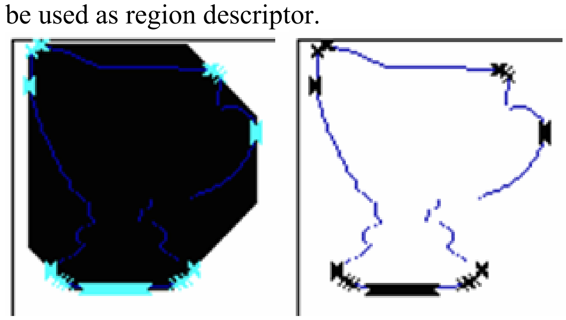
.



(a) (b)

Figure 5

The result of applying morphological convex hull algorithm over object in figure 5 (a) is show in figure 5 (b). The concept of convex hull is not only helpful in finding out the eccentricity of the object, but it is equally useful for describing an entire region. But the ability of convex hull to grow beyond minimum dimensions required to guarantee convexity, can be shortcoming if convex hull is to be used as region descriptor.



(a) (b)

Figure 6

One simple solution is to reduce this effect by limiting growth so that it does not extend past the vertical and horizontal dimensions of the original set of points. The result of applying the technique over object in figure 5(a) is shown in figure 6(a), figure 6(b) shows critical points in the object that can also act as useful descriptors.

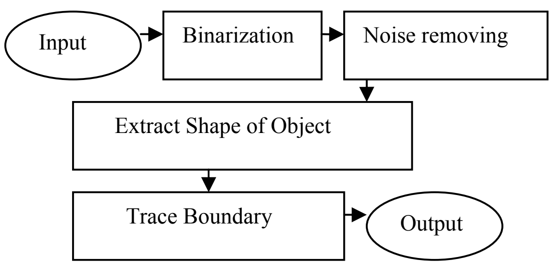
4. Object Recognition

The process of object recognition generally constitutes three phases, i.e pre-processing, feature extraction and feature matching. These phases are described below.

4.1 Pre-Processing

The shapes input to the object recognition system (ORS) might be real world objects or user drawn shapes, before a shape is recognized it need to be first converted into form suitable for application of algorithms in feature extraction phase. The main goal of pre processing is to convert the input object into set of coordinates that describe the shape of the object i.e to extract boundary information or coordinates of boundary from the shape.

The block diagram can be show as



The first step in the preprocessing is to binarize the shape image, this can be done by selecting a proper threshold by analyzing the histogram of the image. In reality, shape images are often corrupted with noise, as a result, the shape obtained from the thresholding usually has noise around the shape boundary, therefore proper algorithm need to be applied to remove this noise which eliminates the isolated pixels and small regions or segments. After the object has been binarized and the noise has been removed, shape extraction algorithms are employed. The shape extraction procedure is important as later phases use the outcome of this procedure to calculate the descriptors related to the shape, which later are use in object matching. In this paper, morphological algorithms for shape extraction are used. The algorithm first calculates dilated image of the object followed by the calculation of eroded image, and calculates the object boundary by subtracting eroded image from dilated image (or from original image). The result of applying dilation and erosion on object in figure 7(a) is shown in figure 7(b) and (c) respectively.

(a) (b) (c)

Figure 7

The algorithm for calculating the shape proceeds as follows:

1. Read image information (all coordinates) output from preprocessing phase in A.
2. Calculate eroded image from A, that is
   B = erosion(A)
3. Calculate C as
   C = A – B
4. where C will contain the boundary of image

The result of applying algorithm over the object in figure 7(a) is shown in figure 8. Shape consists of two boundaries, outside shown in blue and inside boundary shown in black.

Figure 8

Next the boundary is traced, which is used in feature extraction phase i.e boundary based shape descriptors extractions. Various boundary extraction algorithms have been proposed in literature, in this paper, following techniques to compute the boundary from the shape are used.

1. Select any coordinate of the shape output from previous algorithm (shown in figure 8)
2. Mark coordinate as current pixel
3. Check out all neighboring pixels of current pixel
4. If any pixel in the neighbor of current pixel is in shape, add current pixel to the list of boundary coordinates, make that neighboring pixel the current pixel
5. Go to step 1.

The output of the algorithm is shown in figure 9 (a), the algorithm cannot operate on shape that is not connected, this shortcoming is removed by using m-connectivity technique for gap filling between the boundary points before the application of above algorithm.

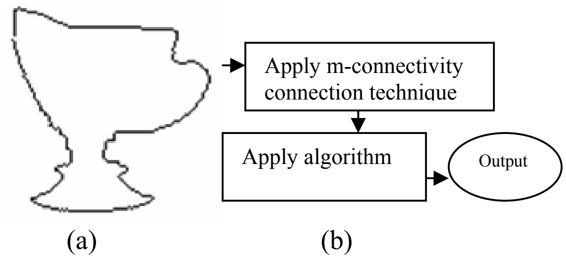


Figure 9

The process can be summarized as shown in figure 9(b).

The other technique is move set of tangents from left to right, right to left, bottom to top and top to bottom, if a tangent encounters an object's pixel, mark that as boundary. The result of algorithm can be seen in figure 10, the boundary has been shown in four color, as they have detected by tangents moving in four directions. Tangents from right to left has detected pink coordinates, form left to right have detected green, top to bottom have detected cyan and bottom to up have detected yellow. The technique will give broken boundary, this shortcoming can be solved by applying m-connectivity connection technique to fill gap between boundary points or simply traversing in more than four directions. The result of traversing in eight directions have been shown in figure 11(b), four direction traversing is shown in figure 11(a).


Figure 10

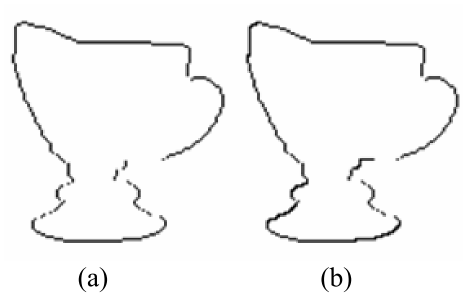
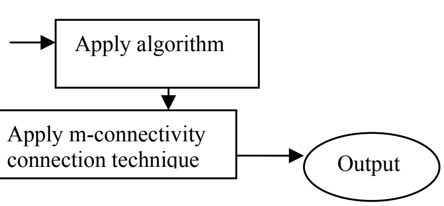
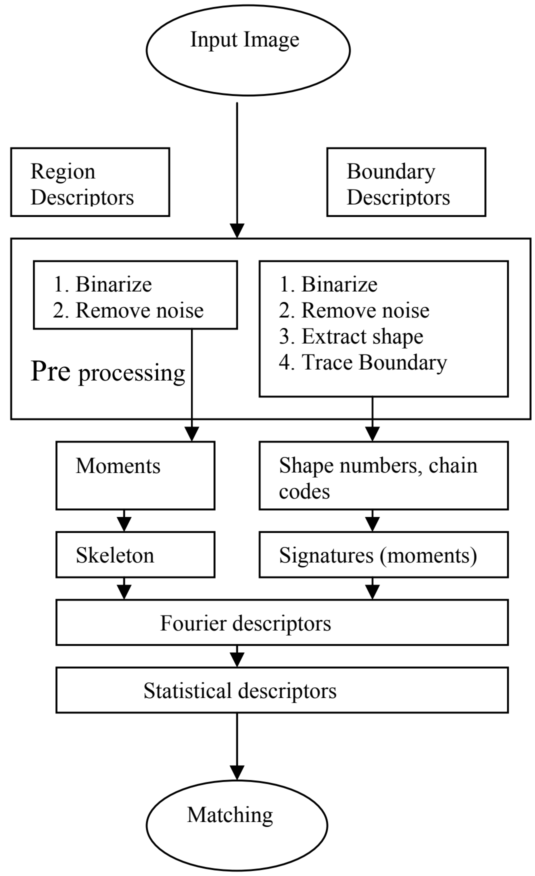


Figure 11

The process can be summarized as



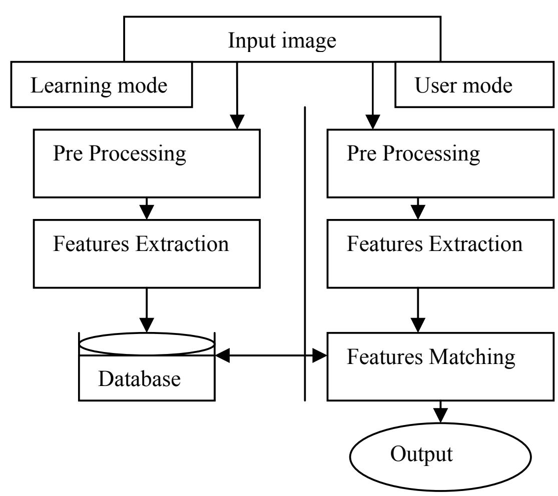
4.2 Feature Extraction



The output of pre processing phase is used to extract features helpful in matching phase, the features are basically descriptors as described in detail in section 2 and 3, consisting both boundary and region based shape descriptors. In this paper, following model of features extraction is proposed

4.3 Object Recognition

In this paper, following model for object matching and recognition is used.



5. Experimental Results

The techniques in the paper are implemented using C language, no other graphics library except bgi of turbo C is used. The core purpose of using C is to achieve computational efficiencies. The techniques are tested with more that 50 object in database, the results even with template matching are exceptionally good, results are likely to fade away as the number of objects are increased, this shortcoming is discussed in section 7 (future works), but still sheer using of C as implementation have resulted in excellent system that can be used even to take real time decisions, making it suitable for real time and embedded systems. The results of computing descriptors on the shapes in figure 12 are shown in figure 13.

(a) (b) (c) (d)

Figure 12

Object Name	Human1	Human2	Human3	Human4
Center	91,86	95,83	91,85	118,75
Perimeter	528	562	502	559
Diameter	123.2274	132.0189	127.0118	129.0736
Area	5262	5564	5634	6013
Compactness	0.237068	0.221261	0.280801	0.241689
Compactness2	52.98062	56.76563	44.72915	51.96757
Elongatedness	0.086632	0.07981	0.087311	0.090231
Dispersion	0.073398	0.074493	0.070781	0.067364
Dispersion2	2.863564	3.18651	3.125577	3.035504
FD (coordinates)	1	1	1	1
	0.917134	0.709639	0.656819	0.224225
	1.100003	0.538509	1.12716	0.460868
	0.470844	0.467898	1.295069	0.400955
	0.234059	0.446728	0.253776	0.603098
	0.622752	0.376805	0.447117	0.334128
	0.714012	0.124204	0.326023	0.216422
	0.239625	0.27682	0.190268	0.271884
	0.576496	0.127886	0.726567	0.123396
	0.2972	0.16972	0.91496	0.2521
FD (singatures)	1	1	1	1
	0.53466	0.368515	0.709452	0.079167
	0.757098	0.186517	0.596357	0.39304
	1.418355	0.08597	0.751742	0.138416
	1.112033	0.156542	0.766023	0.631264
	0.230959	0.037443	1.405829	0.459717
	0.647908	0.06077	0.401832	0.322239
	0.751168	0.031668	0.38408	0.280337
	0.374024	0.046709	0.515208	0.166819
	0.81971	0.055716	0.2645	0.12257
Moments	58	55	58	55
	3799	3445	3874	3469
	269746	238297	281344	239683
	20578385	17820325	21971314	17879579
	0	0	0	0
	377	412	435	396

Figure 13

6. Conclusion and Future Work

In this paper, we have looked into the details of various descriptors that can be helpful in object recognition process. Since different descriptors have been described that appears to be useful, any one of the descriptor can be used for object recognition depending upon the application (i.e objects to be recognized), we have proposed a frame work that will help in development of such systems.

Searching entire database for matching each object is wasteful, in future works we will index the entries in the database and group them (in trees, or graph), for efficient retrieval of the results. Also a feedback loop need to be added, so that based on the output (if it is right or wrong) it learns, this would lead to development of neural network assigning weightage to different descriptors and adjusting weightage each time conclusion is drawn.

7. References

1.Rafeal C.Gonzalez. Richard E.Woods, Digital Image Processing, Prentice Hall second edition

2.Mark S.Nixon. Alberto S.Aguando, Feature Extraction and Image Processing, Newness production

3.Dengsheng Zhang. Guojun Lu, Content based shape retrieval using different shape descriptors: a comparative study, Monash University Australia

4.Dengsheng Zhang. Guojun Lu, a comparative study of three region shape descriptors, Monash University Australia

5.Andre Folkers, Institute for signal processing, Medical university of Lubeck, Germany, Hanan Samet, computer science department, university of Maryland. Content based image retrival using Fourier descriptors on a logo database.

6.Dengsheng Zhang. Guojun Lu, a comparative study on shape retrieval using fourier descriptors with different shape signatures, Monash University Australia

7.D.J.Lee, Sameer Antani, L.Rodney Long, similarity measurement using polygon curve representation and fourier descriptors for shaped based vertebral image retrieval, dept. of electrical and computer engineering Brigham Young University

8.Steven W.Smith, the scientist and engineer guide to digital signal processing, second edition California technical publishing