

## USE OF DYNAMIC TIME WARPING FOR OBJECT SHAPE CLASSIFICATION THROUGH SIGNATURE

Santosh K.C.

Institut National de Recherche en Informatique et en Automatique (INRIA), LORIA -  
Campus Scientifique -BP 239 - 54506 Vandoeuvre-les-Nancy Cedex, France

Corresponding author: Santosh.KC@loria.fr

*Received 21 June, 2009; Revised 15 January, 2010*

### ABSTRACT

This paper expresses an application of similarity matching of the signatures through DTW. Fundamental aspect of classification is template matching. The classification is robust to noise, scaling, and rotation. Feature includes radius plus angle along the boundary points with respect to center of gravity. The classification automatically and confidently discloses the shape of every object at once throughout page from top to bottom. The paper expresses its promising results within an average of a few seconds (cheaper classification) for an object. A series of tests is done with all possible configurations of geometrical shapes.

**Keywords:** Signature, Dynamic Time Warping, Uniform Scaling

### INTRODUCTION

Shape is one of the salient features of visual content and is popularly used in visual information retrieval. In other words, the shape of an object is equally important like color, texture etcetera in the context of image processing and pattern recognition (Goshtasby *et al.*, 1998 and Bibriesca, 1982). It reveals a concept of object's structure to learn object and even identifies those learned components. This brings images easier to handle. Under image retrieval scope, shape description is the first step to move to further processing (Bimbo, 2001) where the basic problem is to define the similarity between the shapes. In many cases, the similarity between the shapes should obey the human perception and broadly, images can be classified based on the shape context using different types of shape descriptors (Costa *et al.*, 2001). Human Visual System (HVS) (Elliffe *et al.*, 2002 and Stringer *et al.*, 2000) is generally far superior from artificial as it is possible to discriminate thousands of different shapes, colors, textures in a variety of lightening conditions.

Globally speaking, shape of the graphical object varies from one to other and thereby it is easier to classify objects in scene understanding but sometimes, there have not been sufficient parameters/features from the shape to be dealt for distinguishing. Moreover, a variety of potential errors arising from the randomness, incompleteness and vagueness, limits the efficacy of the classifier. Classifiers are very sensitive to the shape as well as the position, and become easily affected by these potential errors. This eventually affects the global confidence in recognition. The paper does not cover all about how to handle noisy objects but, it presents a prototype model for shape classification and attempts to demonstrate its potential towards the flexibility and extension of the paper.

The structure of this paper is as follows. The paper starts with detailing literature review in the section – Context: State-of-the-Art. The learning module is developed in section - System Design: it includes pre-processing, feature selection and template management. Then, classification module follows in section – Similarity Measurement and Classification (it tells strictly about an algorithm for classification), and section - Experimental Setup gives an overview of the way of examining object shapes. Assessment of the methodology is explained in the section – Assessment and Discussions where analyzing part of the assessment is also carried out. Conclusion section ends the paper along with few upcoming tasks to be headed in the section – Conclusions and Future Directions.

### **CONTEXT: State-of-the-Art**

Pattern recognition studies the operation and design of systems that recognizes the patterns (Gonzalez *et al.* 1993). Comprehensive idea on arithmetic operations on shapes is explored in (Bribiesca, 1981). Pattern here, refers to the shape of an object. Shape analysis is useful in a number of applications: machine vision including medical image analysis, aerial image analysis and manufacturing. Techniques (Loncaric, 1998) vary depending on the applications. Methods employed in object recognition are global and local. Global methods are Fourier descriptors and moments (Arbter *et al.*, 1990, Brown, 1992), while local techniques include features such as critical/interest points. Local techniques are preferred especially in the region of noisy environment and where more complexity in the studied objects presents. Region and contour/boundary based techniques are popular (Gonzalez *et al.* 1993, Freeman *et al.*, 1974). Boundary based techniques trace out only the boundary (Freeman *et al.*, 1977, Freeman *et al.*, 1970, Persoon *et al.*, 1977), while ignoring the interior of the object. In such a case, regular object with holes are treated as solid object, since holes are not so important and the boundary only explores shape of the overall object. Very often, shape boundary is described by the set of points. Therefore, high computational cost associated with every point must be reduced. Those points can be selected based on the maximal curvature (Super, 2004), distance from the center of gravity (Zhang *et al.*, 2003) and any criteria deemed appropriate for the applications. Equal spacing of points can be defined in terms of either perimeter distance or radial angle (McNeill *et al.*, 2005). Besides, use of landmark points along the outline gives better results but load increases since it uses expensive optimization algorithm (Wang *et al.*, 2004).

Popularly, Bribiesca *et al.*, 1978 and 1980 used the model invariant to scale, translation, rotation and reflection under 2D images. In human visual perception, the shapes are processed in a multiple resolutions and therefore multi-scale shape representations is essential (Kunttu *et al.*, 2003) in the shape based image classification and retrieval. This multi-resolution representation provides additional accuracy to the shape classification. Besides, Kunttu *et al.*, 2003 positioned the concept of multi-scale Fourier descriptor for shape classification that includes an idea of curvature Fourier, radius Fourier, contour Fourier (Kauppinen *et al.*, 1995), and A-invariant methods for Fourier-based shape representation (Arbter *et al.*, 1990). Moreover, (Kunttu *et al.*, 2003) added that contour Fourier and A-invariant methods were best approaches in shape classification. The Contour Fourier method transforms the Fourier directly for the complex coordinate function of the object boundary in which both positive and negative frequency axis descriptors are taken. Fundamentally, size does not effect the shape of objects. In a similar manner, rotation and spatial organization of the objects in a complex scene does not change the shape. With these ideas, 2D simple

objects were experimented under HVS scope (Vonikakis *et al.*, 2005).

Curvature scale space (CSS) technique (Mokhtarian *et al.*, 2003) is found robust to describe complex shapes (Kopf *et al.*, 2005) where curvature of points is taken from the inflection points. But the major drawback is CSS method draws poor representation of convex segments of the shape. This is because of the missing important remaining points along the shape. Therefore, extra interpretations have to be done.

It is evident that a string of coordinates along the boundary of any object represents its shape. Kept this in mind, the paper takes center of gravity as a reference point and calculates angle values along the contour of the object with the inclusion of the radius. It is comparable to few existing methods in some respects, but it comes to be different in the use of boundary points (Kang *et al.*, 2004) and the use of principal axis (An *et al.*, 2005). In addition to the shape, the values of radius from the center of gravity over the boundary suggest the actual size of the object. Moreover, minimum and maximum distance can be possibly taken from the center of gravity. Point to remember here is, for a complex object, center of gravity does not always fall in the center of the object since it is entirely depend on distribution of the density of pixels over the region. It should also be kept in mind that use of interest points (inflection points) (Kopf *et al.*, 2005) or landmark points (Wang *et al.*, 2004) may not represent the actual shape of an object. But, it yields faster access to the users.

The aim of the paper is to provide a concept of importance of shape for identifying/learning components of any objects in a complex scene. This presents a prototype method for classifying simple 2D object's shape. Very specially, it focuses on geometric shapes. Comprehensive idea of template based methodologies is presented where Dynamic Programming (DP) has been applied in the classification module. Dynamic Time Warping (DTW) algorithm has been popularly used in all areas under the scope of image processing and pattern recognition, for instance from symbol/object recognition to speech.

Roughly, the proposed model uses normalized signature of every shape from every possible rotation over 360 degree. Therefore, each object's shape gives rise to many signatures as it receives from every orientation. Remember, it employs size normalization of every object shape. In order to decrease the load (time complexity), the paper proposes to resize every object to (0, 0) to (1, 1) window without interrupting the actual shape. This has been done with the help of Minimum Boundary Rectangle (MBR) [Freeman *et al.*, 1975, Papadiasi *et al.*, 1995, K.C. *et al.*, 2009]. In classification module, test signature is aligned with the templates in order to find the best match. Best match here, refers to the template from which it produces lowest matching score. Use of DTW for similarity assessment leaves an advantage that any size of the signature can be applicable to the classifier. Connecting with those previous works (K.C. *et al.*, 2006), Dynamic Programming (DP) now pours the flexibility of its worthy extensions.

## **SYSTEM DESIGN**

Very basic structure of template-based classification model is shown in Figure 1. It consists of both training and testing modules. In template based classification system, both the modules contain the same elements: pre-processing and feature selection. The only difference is: templates are stored in training module, whilst test features are aligned with the stored

templates in testing module. What it says, training module ends in template management step, but testing module extends up to similarity matching in order to receive the best match.

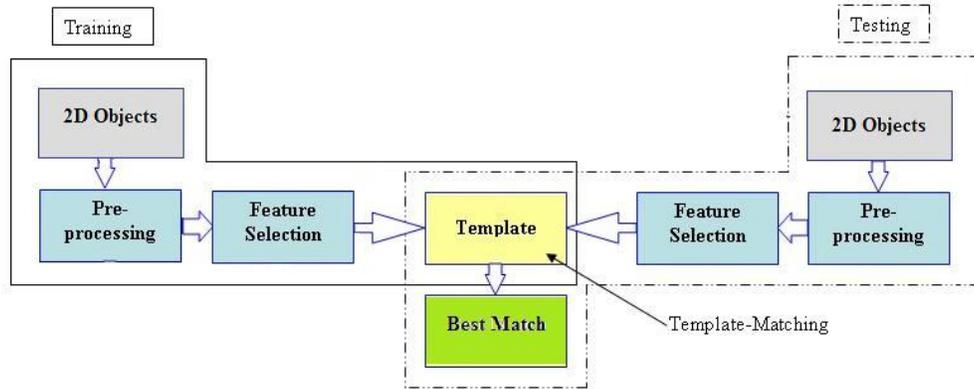


Figure 1 A simple template based classification model

### PRE-PROCESSING

Inputs are often incomplete, noisy and inconsistent, which are needed to be pre-processed before applying to the system in order to receive confidence in correct classification. All the ways (techniques) to refine the data suitable for analyzing are included under the pre-processing technique. Sampling, noise elimination, discretization, integration and transformation are the basic techniques of pre-processing. Different systems use a variety of different techniques. This proves that techniques are not always usual and standard. In other words, it brings a concept that the pre-processing techniques are problem dependent. This paper just focuses on noise elimination in binary images and it uses MATLAB image processing tool box. Noise here, refers to unnecessary dots over the region within the image.

In addition, size normalization has been done without loss of the actual shape. In order to use variable size of the object's shape, this is the most for consistency examine. The paper uses size normalization, i.e. the conversion of original size of every object to the new standard size. The new standard size is designated as,

$$P_{new} = \left( \frac{P - P_{min}}{P_{max} - P_{min}} \right) (P_{max}' - P_{min}') + P_{min}'$$

where,  $p_{max}$  and  $p_{min}$  are the maximum and minimum coordinate points on the object's boundary.  $P_{max}' = (1,1)$  and  $p_{min}' = (0,0)$  give the size of new standard window. This is how size normalization has been done. This aids that the classifier is robust to uniform scaling.

### FEATURE – Signature

Very often, system has been dominated by the selection of features. Therefore, sufficient feature selection is necessary from the limited provided input such that it is enough to distinguish from one another even though inputs look identical in human eyes. Elegant feature selection can greatly decrease the workload and simplify the subsequent design process of the classifier. In other words, features should contain information required to

distinguish between the classes, be sensitive to irrelevant variability of the input, and also be limited to permit efficient computation of discriminant functions and to limit the amount of training data required. This brings readers attention to how sensitive the feature is to the system.

This paper proposes to use signature as the feature. Consider a centroid of an object 'C' from which radius can be computed along the boundary points. In addition, in every point, angle with respect to centroid is calculated. Now, a string containing both radius and angle gives the feature of the object's shape. This is the fundamental idea behind use of feature. Figure 2 demonstrates how the feature is extracted by taking a simple 2D rectangular object. No matter, how many holes are there in the object, but it only deals with the boundary/contour points since there are many evidences (Freeman *et al.*, 1977, Freeman *et al.*, 1970, Persoon *et al.*, 1977, Super, 2004, Zhang *et al.*, 2003 etcetera) that boundary reveals the shape of the object. In such a feature, a sequence of only angle does not represent the idea of the shape but also radius in every point aids distinguishing parameters from one shape to another. The major clue behind how to distinguish one shape to another is just the immediate/sudden change of radius and angle along the boundary from one point to the next with respect to centroid. For instance, four corners are really important to separate rectangle from triangle. But on the other hand, those four corners are not sufficient for rectangle and square separation. That is why, the feature takes all boundary points not only corners.

Along the boundary points 'Pk', a feature string for *i*-th object's shape from the center Ci can be expressed as,

$$S_i = [(R_{i,1}, f(C_{i,1}, P_{i,1})), (R_{i,2}, f(C_{i,2}, P_{i,2})), \dots, (R_{i,k}, f(C_{i,k}, P_{i,k}))]$$

where,  $R_{i,k}$  denotes radius of *k*-th point on the boundary, which can be easily calculated by the use of simple Euclidean distance metric technique (see section – Similarity Measurement) and function  $f(C,P)$  denotes angle of boundary point 'P' with respect to centroid 'C' which can be further expressed as,

$$f(C,P) = \arctan\left(\frac{y_P - y_C}{x_P - x_C}\right)$$

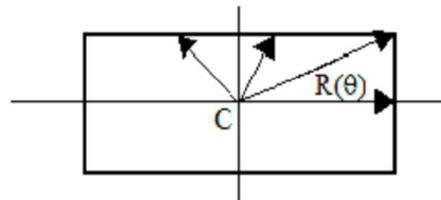


Figure 2 Signature model

How does the signature look is shown in Figure 3 for a few simple 2D objects with regular shapes: a triangle, a square, a circle and a rectangle respectively. The rise and fall of the signature gives idea of number of corners with respect to reference centroid and every valley in the signature pours a concept of number of sides. In addition, visual size of each valley

determines how big is the side of an object (Figure 3 (Entity 4)). It seems that the size of the valley is proportional to the length of the side of an object shape.

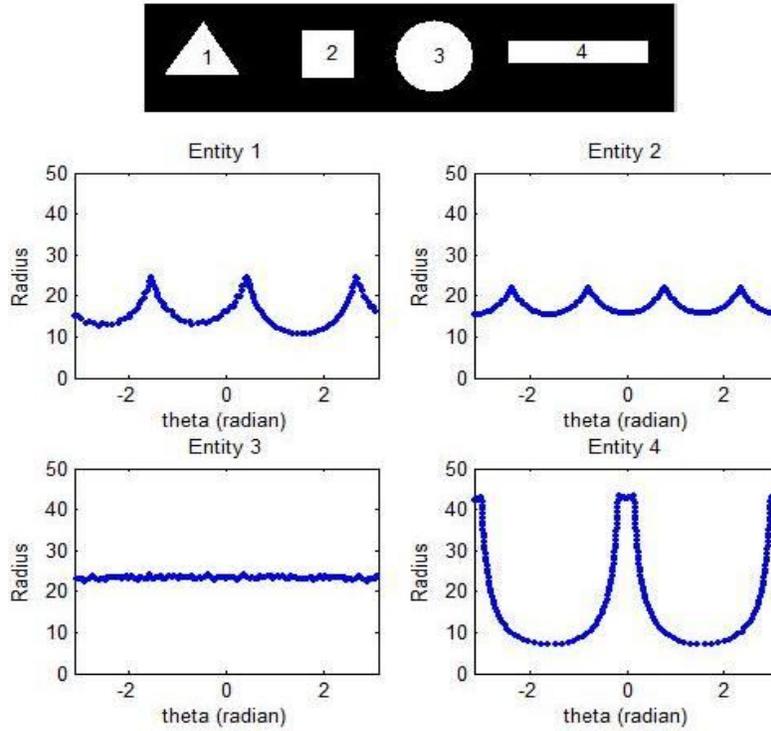


Figure 3 Signatures

**TEMPLATES AND THEIR MANAGEMENT**

Templates for all orientations over one cycle (360 degree) for each type of shape are needed. What it says, the paper takes every signature from every orientation. In this paper, each object is rotated at an angle of 5 degree in each step from the initial orientation until it reaches the original orientation. It is to remind that the interval of rotation can be any values depending on the necessity.

In this paper, seventy-two signatures have been taken in every *i-th* object shape. Mathematically,

$$S_i = [S_{i,1}, S_{i,2}, \dots, S_{i,72}]$$

This makes that the system is robust to any orientation of the test shape to be distinguished.

Template management has taken an important place since it reduces the load on the classifier because it makes system easier to handle. Interactive users take an advantage of the correct template management as they do not like to wait a long for the output. In this paper, each frame is created for each type of the object shape where it contains seventy-two signatures. Therefore, frame numbering depends on how many types of shapes are going to be used.

## SIMILARITY MEASUREMENT

In order to find the similarity between one feature to the other, the most fundamental technique comes behind distance measurement. This suggests how similar/dissimilar is the studied features. In the context of pattern recognition, it is usual that the features are similar once it results the least distance compared to other pairs. The opposite holds for dissimilarity measurement. This is a very fundamental definition.

Application of distance metric techniques varies from one to another. Looking into the cheapest distance metric models, families of Euclidean distance metric comes to be entirely common for all learners under the same scope. K.C. *et al.*, 2006 gives comprehensive idea about commonly used distance metric types along with the realistic application of DTW algorithm. Connecting with K.C. *et al.*, 2006 ideas, this section presents an idea of what type of distance metric is preferable on which problem/application. The following paragraphs deal with the techniques and present rough concept in a straightforward manner.

The presence of the pixel grid makes distance measurement for all sequences possible. For those fixed length sequences, families of *Euclidean* distance metric are common and faster. The distance between two sequences X and Y is,

$$d_{Euclidean} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

where,

$$X = [x_1, x_2, \dots, x_i, \dots, x_n], Y = [y_1, y_2, \dots, y_i, \dots, y_n]$$

while, *Euclidean Squared* does not take the square root and as a result clustering process is faster. But in some cases, it affects. For instance, hierarchical clustering is likely to change. Manhattan (City Block) comes together with families of faster distance metric,

$$d_{Manhattan} = \sum_{i=1}^n |x_i - y_i|$$

On the way, Pearson Correlation can be expressed as,

$$d_{Pearson} = 1 - r$$

where,  $r$  is the dot product of the  $z$ -scores of the vector  $x$  and  $y$ . It is expressed as,

$$r = \frac{z(x) \cdot z(y)}{n}$$

and

$$z(x) = \frac{x - \bar{x}}{\sigma}$$

The Chebychev distance takes the maximum distance between the elements. The distance between the sequences  $X$  and  $Y$  is computed using the formula,

$$d_{Chebychev} = \max_i |x_i - y_i|$$

where,  $x_i$  and  $y_i$  are the values of the  $i$ -th elements in both the sequences  $X$  and  $Y$  respectively. The Chebychev distance may be appropriate if the difference between the sequences is reflected more by differences in individual dimensions rather than all the dimensions considered together. It is noted that this distance measurement is very sensitive to outlying measurements. It is also called chess-board distance metric. A very simple graphical demonstration for simple Euclidean City-Block and Chess-board distance metric models is shown in Figure 4.

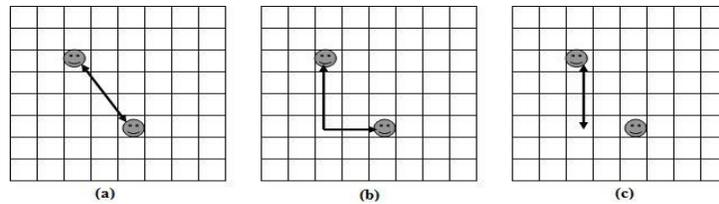


Figure 4 (a) Euclidean, (b) City-Block, and (c) Chess-Board distance measure models.

Spearman Rank Correlation measures the correlation between two sequences. The two sequences are ranked separately and the differences in rank are calculated at each position,  $i$ . The distance between sequences is computed by,

$$d_{Sp\_Rank} = 1 - \frac{6 \sum_{i=1}^n (\text{rank}(x_i) - \text{rank}(y_i))^2}{n(n^2 - 1)}$$

where,  $x_i$  and  $y_i$  are the values of the  $i$ -th elements in both the sequences  $X$  and  $Y$  respectively. The range of Spearman correlation is from -1 to 1. Spearman correlation can detect certain linear and non-linear correlations.

The models recently discussed can be applied where two studied profiles are of same length. That is why, those are straightforward for calculating distance between the pairs. If the studied profiles are of variable lengths then how can we measure the similarity?

### DTW

It is very easy to align two sequences having equal lengths but, how is it possible to align two non-linear sequences? The question is answered by the DTW itself. DTW overcomes the shortcoming of Euclidean distance measure and other common and simple techniques that are used for determining the distance between two sequences having equal length.

In both – speech and character – recognition areas, such a case (sequences having different lengths) occurs. Keogh *et al.*, 1999 provide a concrete difference between the simple Euclidean distance and DTW. It focuses on DTW in case of massive datasets.

A brief explanation of DTW with an example is described in the following paragraphs. Consider two tests  $T$  and reference  $R$  sequences, of length  $n$  and  $m$  respectively.

$$T = [t_1, t_2, \dots, t_p, \dots, t_n], R = [r_1, r_2, \dots, r_i, \dots, r_m]$$

In order to align two variable length sequences a matrix of size  $n \times m$  is constructed. An element of that matrix contains the distance of two points  $t_i$  and  $r_j$ . Euclidean distance between two points can be expressed as,

$$d(i, j) = \sqrt{(t_i - r_j)^2}$$

Each matrix element  $(i, j)$  corresponds to the alignment between the points  $t_i$  and  $r_j$ . A warping path  $Wp$ , now can be defined in the set of matrix elements that defines between  $T$  and  $R$ . The  $k$ -th element of  $W$  is,  $w_k = (i, j)_k$ . Now, we have,

$$W = w_1, w_2, \dots, w_k, \dots, w_K$$

with the condition that  $\max(m, n) \leq K < m + n - 1$ .

Following constraints are considered in warping path:

### Boundary Condition

$w_1 = (1, 1)$  and  $w_k = (n, m)$ , is simply stated. This is necessary the warping path to start and finish diagonally in opposite corner cells in the matrix.

### Continuity

Given  $w_k = (a, b)$  then  $w_{k-1} = (a', b')$ , where  $a-a' \leq 1$  and  $b-b' \leq 1$ . This makes the path run diagonally not in other adjacent cells.

### Monotonicity

Given  $w_k = (a, b)$  then  $w_{k-1} = (a', b')$ , where  $a-a' \geq 0$  and  $b-b' \geq 0$ . This forces the points in  $W$  to be monotonicity space time.

On the other hand, the warping path can be efficiently determined by using *Dynamic Programming (DP)*. It can be expressed as,

$$D(i, j) = \min[D(i-1, j-1), D(i-1, j), D(i, j-1)] + d(t_i, r_j)$$

$D(i, j)$  is the cumulative distance between two sequence  $T$  and  $R$  from three adjacent cells and Euclidean distance in the  $(i, j)$ , which is shown in Figure 5. The element at the end of the matrix gives the minimum distance between the sequences. This is referred to as Matching score in classification module. Euclidean distance is the special case of DTW where the  $k$ -th element of  $W$  is constrained such that  $w_k = (i, j)_k, i = j = k$ . However, it is defined only in case the sequences are of same length. One of the main properties of DTW is time complexity. The time complexity is defined as  $O(nm)$ .

		$B_m$					
		1.578	0.000	-1.560	1.500	0.000	0.120
$A_n$	1.578	0.000	1.578	4.716	4.794	6.372	7.830
	1.278	0.300	1.278	4.716	4.338	5.616	6.774
	0.000	1.878	0.300	1.860	3.360	3.360	3.480
	-1.345	4.801	1.645	0.515	3.360	4.705	4.825
	1.678	4.901	3.323	3.753	0.693	2.371	3.929
	1.340	5.139	4.663	6.223	0.853	2.033	3.253
	0.000	6.717	4.663	6.223	2.353	0.853	0.973
	1.200	7.095	5.863	7.423	2.653	2.053	1.933
	1.450	7.223	7.313	8.873	2.703	3.503	3.263
	1.670	7.315	8.893	10.54	2.873	4.373	4.813
	0.120	8.773	7.435	9.115	4.253	2.993	2.993

Figure 5 Alignment of two non-linear sequences  $A_n$  and  $B_m$  with the use of DTW (K.C. *et al.*, 2006). The element at the end of the matrix is regarded as the minimum distance i.e.,  $D(A_n, B_m) = D(N, M) = 2.993$ . The shaded elements along the diagonal represents Dynamic Warping path.

Typically, two situations are faced in matching the sequences.

### Whole Matching

Matching takes place between two sequences having identical lengths. In such a case, Euclidean distance can be possible and is preferred for faster processing.

### Sub-sequence Matching

Consider a test sequence  $T$  is smaller in length in comparison to reference sequence  $R$ . The test sequence slides along the subsections of every possible subsection of  $R$  to find the best match. It is time consuming to match two sequences having different lengths.

### CLASSIFICATION

Classification process starts with test feature matching the templates stored by using Dynamic Programming (DP). Test feature is the best match with the template from which it gives the lowest matching score (see section - Similarity Measurement). The matched template reveals the recognized shape. The classification procedure is very similar to K.C. *et al.*, 2006. The paper proposes to use a string of radius and angle values for a shape of the object.

Correct management of templates in pattern recognition can greatly reduce the load to the classifier. Considering the issue, the paper focuses on how one can use it for achieving optimum classifier's performance. In order to store templates, the paper uses one frame for one shape. One shape consists of seventy-two signatures, as mentioned earlier (see section - Template and their Management). Therefore, each frame reserves seventy-two signatures. Depending on the number of shapes of the objects, frame numbering varies. This can be

$$Frame = [F_1, F_2, \dots, F_n]$$

expressed as,

In each test signature, there are seventy-two matching scores from every  $i$ -th frame (remember that there are seventy-two signatures in each  $i$ -th frame),

$$F_i = [D_{i,1}, D_{i,2}, \dots, D_{i,72}]$$

From all frames, test feature looks for a best match. The best matching score can be achieved by,

$$Best_{MS} = \min[F_1, F_2, \dots, F_n]$$

Very specifically, the paper proposes threshold in order to collect those similar features from every frame during feature matching. Thereby, similar distance matching scores are reserved for later correct recognition. Threshold is now calculated by adding fixed constant to Best match value.

$$Thsld = Best_{MS} + Const.$$

This then leads to find weighting matching score for every frame. Weighting matching score is the ratio of weight of the frame to the normal matching score. Simply for each frame, weight can be fixed by counting the number of similar features retains under threshold. Therefore, weight for  $i$ -th frame is,

$$W_i = \sum_{j=1}^{72} C(D_{i,j}, Thsld)$$

where,

$$C(x, y) = \begin{cases} 1 & \text{if } (x < y) \\ 0 & \text{otherwise} \end{cases}$$

Now, consider the  $i$ -th frame, weighting matching score can be obtained by dividing the weight of their own frame. Weighting matching score taken only from those matching scores below threshold. Suppose, there are  $t$  matching scores below the threshold. Mathematically, weighting matching scores can be expressed as,

$$\overline{D}_{i,j} = \frac{1}{W_i} [D_{i,1}, D_{i,2}, \dots, D_{i,t}]$$

One must be careful that every frame may not be included in order to find weighting matching scores. This is what it propagates until now. In other words, some frames which consist of quite different shapes are removed and few similar will be remaining. Lastly, the similar procedure for getting the best match is explored in the following equation, only difference is the weighting matching scores.

$$\overline{Best}_{MS} = \arg \min \left[ \sum_{i=1}^n \overline{F}_i \right]$$

This weighted best match gives the index of the shape of the object. Index refers to the placement of template which ultimately reveals the frame numbering. In depth, it even reserves the exact orientation of the shape in that particular frame. Remember, the index of the normal best matching may not be identical to the weighted best matching score.

### EXPERIMENTAL SETUP

The paper follows very fundamental procedures as covered in other projects under object recognition. It simply includes, noise removal, labeling, interpreting the object's shape by signatures from every orientation and automatic shape classification with the help of DTW algorithm for similarity measurement.

In order to keep concrete stand point of the idea under object shape recognition, many samples of geometrical shapes were tested (see section – Assessment and Discussions). This explores the potential toward shape recognition and its extension under the same scope.

All experiments were done by using MATLAB 7.0.1 on a 2.40GHz, 3.00GB RAM and 32 bit operating system, running Windows Vista.

### DATASET

Very few 2D objects' shape have been taken for training the classifier. In a similar way, simple shapes are taken for testing. Figure 6 and 7 demonstrate twenty-seven classes of objects' shape for training and few samples of testing objects out of many respectively. Very specifically, only those testing samples having variable scaling or orientation are demonstrated in Figure 7.

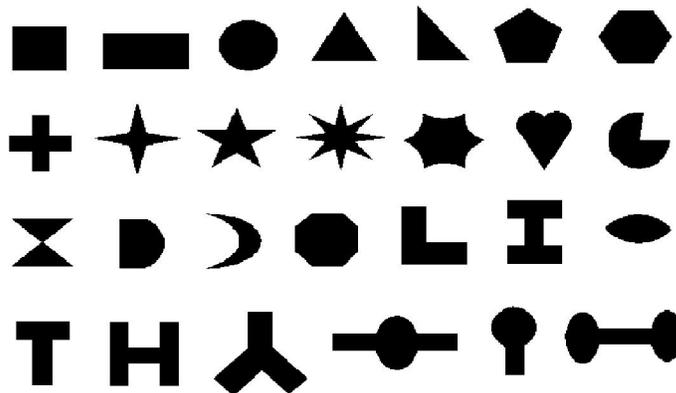


Figure 6 Simple 2D shapes for training

### ASSESSMENT AND DISCUSSIONS

This section brings readers to a strict and straightforward idea on how and why object shapes are mis-classified and rejected as well. 2D objects are randomly tested page by page. Each page consists of many objects with similar/dissimilar shapes, and variable sizes and orientations. For a quick understanding, the paper only discloses Figure 7 where, it illustrates

one page with twenty-seven tested 2D objects. As discussed in the section – Classification, the labeling (indexing) at the output declares the shape of the object according the template placement.

There are obviously several causes (starting from vagueness of the inputs, pre-processing, feature selection, and classification algorithm), but the project brings two globally accepted criteria for mis-led results while testing 2D object shapes. They are,

**Mis-recognition:** It says different index than what it should be. For instance, 'star' is matched with 'red-cross shape'.

**Rejection:** It does not convey any indexing at the output. It may be either due to noise (input vagueness), insufficient feature selection or crude classification algorithm. It only says there is no similar template with the test shape.

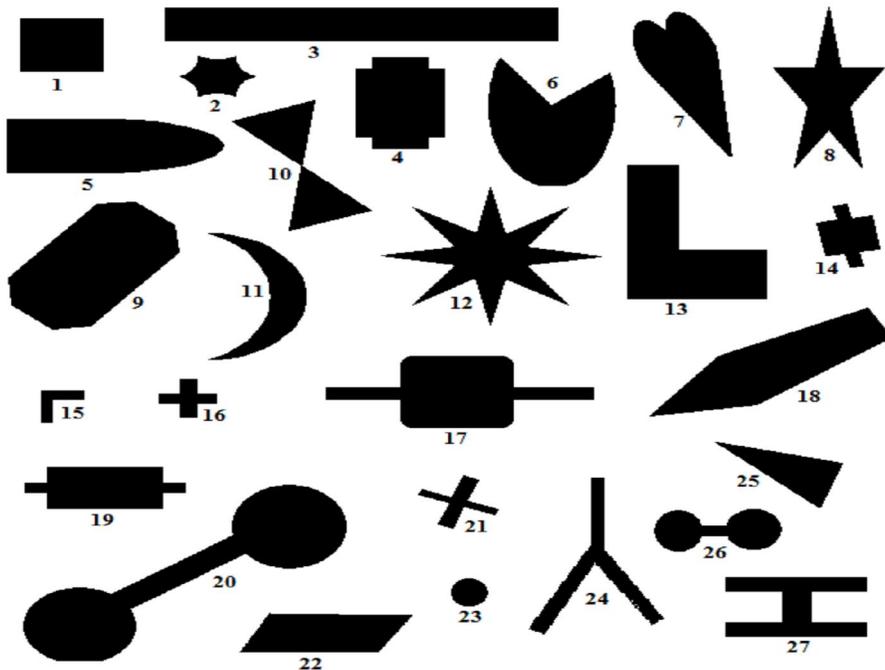


Figure 7 Few testing samples

Very specifically, the paper does not differentiate the shape of the rectangle with the parallelepiped due to feature selection – it only uses number of sides plus its size. This tells that the use of center of gravity for referencing the object in order to calculate signature comes to be crude for even a simple 2D objects. This is not always the worst case. Moreover, if the center of gravity does not fall within the shape of the object, then it is not possible to examine the shape of the 2D objects from outside. For instance, Figure 7 (15 and 24) tells that center of gravity does not fall within the boundary. The paper is fully dependent on the center of gravity, which must be within the object. But, one should remember that it works well when the system has been trained with such kinds of object. Moreover, scaling in

particular region – non uniform scaling (scaling in one side not on the other side Figure 7 (9, 14, 18 and 19) within the object does not confidently convey correct indexing. Besides, there is no problem in uniform scaling over the region (Figure 7 (2, 3, 6, 8, 11, 12, 13, 16, 17, 20, 23, 24, 26, 27)).

Considering the time complexity of the classifier, it must be superior to the existing ones as it compares only with the templates with limited possible orientations over one cycle. One might think its load as it linearly increases with the number of signatures from every orientation of each object shape. In other words, use of signatures from every orientation in each type of object's shape increases the load on to the classifier. As it yields preciseness in recognizing the object shapes, increasing by a few seconds is not really a bottleneck to its performance. Preciseness refers to the shape recognition plus its orientation. It even fits the demand of any interactive users. It results within a couple of minutes for a one page of objects from top to bottom and it takes a few couple of seconds for an object in average.

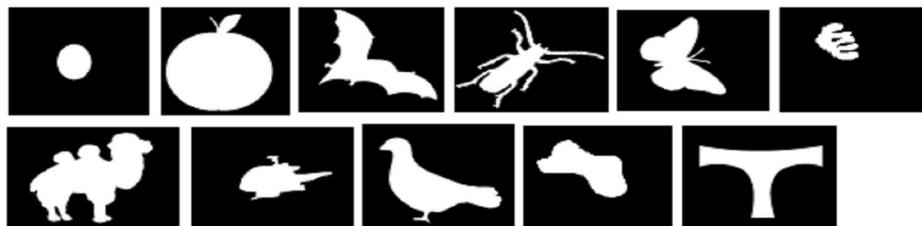


Figure 8 An extension of assessment

Interestingly, recognition of ball, apple, bat, etcetera as shown in Figure 8 has been done by using the same procedure. Due to this, it now discloses its robustness towards learning any kinds of shapes. This phenomenon draws an attention and excitement for further improvement under object recognition. Remember, shape can classify object but it may not noticeably contain sufficient features such that one can reconstruct the other side.

## CONCLUSIONS AND FUTURE DIRECTIONS

The paper presents an idea in the context of 2D object shape recognition. It expresses an application of similarity matching of the signatures. It brings a concept of the best match by aligning two non-linear features' sequences through DTW. The classification is robust to noise, scaling, and rotation. Feature includes radius plus angle along the boundary points with respect to center of gravity. The faster processing time is the superiority of the project among others. The paper expresses its promising results within an average of a few seconds for an object. A series of experiments is done with all possible configurations of geometrical shapes. This is how the paper discloses its potential and flexibility. An extension of the paper pours its flexibility, recently discussed by taking an example of Figure 8.

The use of principal axis of the object in feature selection rather than taking signatures from every orientation over a cycle can greatly reduce the time complexity associated to classifier. But, it does not really convey its orientation of the shapes. With this, the methodology is likely to be extended up to the level of object recognition through shape configurations.

## REFERENCES

1. A. Goshtasby, "Registration of Images with Geometric Distortions", IEEE Transaction on Geo Science Remote Sensing, 26, pp. 60-64, 1998.
2. B. J. Super, "Fast Correspondence – System for Shape-Retrieval", Pattern Recognition Letters, Vol. 25, pp. 217-225, 2004.
3. A.D. Bimbo, "Visual Information Retrieval", Morgan Kaufmann Publishers, SanFrancisco, 2001.
4. D. Papadias, T. Sellis, Y. Theodoridis, M. J. Egenhofer, Topological Relations in the world of Minimum Bounding Rectangles: a Study with R-trees, in: Internal Conference on Management Data, 1995, pp. 92–103.
5. E. Persoon and K. Fu, "Shape Discrimination Using Fourier Descriptors", IEEE Transaction on Systems, Man, and Cybernetics, Vol. 7, pp. 170-179. 1977.
6. E. Bribiesca and A. Guzman, "Shape Description and Shape Similarity Measurement for Two Dimensional Rrgions", 4<sup>th</sup> International Conference on Pattern Recognition, pp. 608-612, 1978.
7. E. Bribiesca and A. Guzman, "How to Describe Pure Form and How to Measure Differences in Shapes Using Shape Numbers", Pattern Recognition, Vol. 2, No. 1, pp. 101-112, 1980.
8. E. Bribiesca, "Arithmetic Operation Among Shapes Using Shape Numbers", Pattern Recognition, Vol. 13, No. 2, pp. 123-137, 1981.
9. E. Bribiesca, "Shape Classification on Digital Images", International Society of Photogrammetry and Remote Sensing, Commission IV, Symposium: Environmental Assessment and Resource Management, 1982.
10. E. Bribiesca, "Unsupervised and Supervised-classification on Digital Images Using Shape and Color", Advances in Information Sciences and Technology, Pattern Recognition and Digital Technique, Vol. 1, 1982.
11. F. Mokhtarian and Mirosław Bober, "Curvature Scale Space Representation: Theory, Applications, and MPEG-7 Standardization (Computational Imaging and Vision, 25)", Kluwer Academic Publishersm Dordrecht, The Netherlands, 2003.
12. G. McNeill and S. Vijayakumar, "2D Shape Classification and Retrieval", International Joint Conference on Artificial Intelligence, pp. 235-242 , 2005.
13. H. Freeman and L.S. Davis, "A Corner Finding Algorithm for Chain Coded Curves", IEEE Transactions on Computers, Vol. 26, pp. 297-303, 1977.

14. H. Freeman and J.M. Glass, "Computer Processing of Line Drawing Images", *Computing Surveys*, Vol. 6, No. 1, pp. 57-97, 1974.
15. H. Freeman and J.M. Glass, "Boundary Encoding and Processing", In *Picture Processing and Psychopictorics*, Academic Press, pp. 241-263, 1970.
16. H. Freeman, R. Shapira, Determining the Minimum-Area Encasing Rectangle for an Arbitrary Closed Curve, *Communications of the ACM* Vol. 18 No. 7 (1975) 409–413.
17. H. Kauppinen, T. Seppanen, and M. Pietikainen, "An Experimental Comparison of Auto-regressive and Fourier-based Descriptors in 2D Shape Classification", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 2, pp. 201-207, 1995.
18. I. Kunttu, L. Lepisto, J. Rauhamaa, and A. Visa, "Multiscale Fourier Descriptor for Shape Classification", *International Conference on Image Analysis and Processing (ICIAP)*, pp.536, 2003.
19. J. Zhang, X. Zhang, X. Krim, and G.G. Walter, "Object Representation and Recognition in Shape Spaces", *Pattern Recognition*, Vol. 36, pp. 1143-1154, 2003.
20. K. Arbter, W.E. Synder, H. Burkhardt, and G. Hirzinger, "Application of Affine-Invariant Fourier Descriptors to Recognition of 3D Objects", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 7, pp. 640-647, 1990.
21. K.C. Santosh and C. Nattee, "Structural Approach on Writer Independent Nepalese Natural Handwriting Recognition", *IEEE International Conference on Cybernetics Intelligent Systems*, pp. 1-6, 2006.
22. K.C. Santosh and C. Nattee, "Stroke Number and Order Free Handwriting Recognition for Nepali", *Lectures Notes in Computer Science – PRICAI: Trends in Artificial Intelligence*, pp. 990-994, 2006.
23. K.C. Santosh, Laurent Wendling and Bart Lamiroy "New Ways to Handle Spatial Relations through Angle plus MBR Theory on Raster Documents", *Graphics Recognition*, 2009. (To be appeared)
24. L.G. Brown, "A Survey of Image Registration Techniques" , *Computer Survey*, Vol. 24, No. 4, pp. 325-376, 1992.
25. L.F. Costa and R.M. Cesar, "Shape Analysis and Classification", *Theory and Practice*, CRC Press, Boca Raton, Florida, 2001.
26. M.C.M. Elliffe, E.T. Rolls, and S.M. Stringer, "Invariant Recognition of Feature Combinations in the Visual System", *Biological Cybernetics*, Vol. 86, pp. 59-71, 2002.

27. R.C. Gonzalez and R.E. Woods, "Digital Image Processing", Addison Wesley, 1993.
28. S..K. Kang, M.B. Ahmed, J.H. Chun, P.K. Kim, and J.A. Park, "Modified Radius-Vector Function for Shape Contour Description", Lecture Notes in Computer Science in Computational Science and Its Applications, pp. 940-947, 2004.
29. S. Kopf, T. Haenselmann, and W. Effelsberg, "Enhancing Curvature Scale Space Features for Robust Shape Classification", IEEE International Conference on Multimedia and Expo, pp. 4, 2005.
30. S.M. Stringer and E.T. Rolls, "Position Invariant Recognition in the Visual System with Cluttered Environment", Neural Networks, Vol. 13, pp. 305-315, 2000.
31. S. Loncaric, "A Survey of Shape Analysis Techniques", Pattern Recognition, Vol. 31, No. 8, pp. 983-1001, 1998.
32. S. Wang, T. Kubota, and T. Richardson, "Shape Correspondence Through Landmark Sliding", IEEE Conference on Comp. Vis. and Pattern Recognition, Vol. 1, pp. 143-150, 2004.
33. Y.E. An, G.W. Kang, M.H. Chang, S.J. Park, and J.A. Park, "Distances and Angles using Modified Radius Vector as Features for Shape Matching", ACIT-Signal and Image processing, 2005.
34. V. Vonikakis, I. Andreadis, and A. Gasteratos, "Simple-Shape Classification Based on the Human Visual System", In Proceedings of Visualization, Imaging, and Image Processing, pp. 162-167, 2005.