Engineering Letters, 14:1, EL_14_1_24 (Advance online publication: 12 February 2007)

K-Mutual Nearest Neighbour Approach for Clustering Two-Dimensional Shapes Described by Fuzzy-Symbolic Features

H.S.Nagendraswamy and D.S.Guru

Abstract— **In this paper, a new method of representing two-dimensional shapes using fuzzy-symbolic features and a similarity measure defined over fuzzy-symbolic features useful for clustering shapes is proposed. A *k*-mutual nearest neighborhood approach for clustering two-dimensional shapes is presented. The proposed shape representation scheme is invariant to similarity transformations and the clustering method exploits the mutual closeness among shapes for clustering. The feasibility of the proposed methodology is demonstrated by conducting experiments on a considerably large database of shapes and also, its validity is tested by comparing with the well known clustering methodologies.**

Index Terms—**Fuzzy-Symbolic features, *k*-mutual nearest neighbour, Symbolic data analysis, Shape representation, Shape clustering.**

I. INTRODUCTION

Clustering plays a significant role in several exploratory pattern analysis, grouping, decision-making, machine learning situations, including data mining, document retrieval, image segmentation and pattern classification. In general, cluster analysis is of great importance in classifying a heap of information, which is in the form of data patterns. Clustering of planar objects or images of objects, according to the shapes of their boundaries along with the inner shape details can significantly improve database searches in systems with shape-based queries. For instance, testing a query against prototypes of different clusters to select a cluster and then testing against shapes only in that cluster is much more efficient than testing exhaustively. In order to cluster shapes, we need to characterize shapes in terms of features, which best captures the shape information. The concept of symbolic data sets provides a natural way of capturing shape information and the concept of fuzzy sets efficiently handles the uncertainty in the data. Many clustering algorithms have been proposed for clustering conventional [12], [14], [15], [16] as well as symbolic data [5], [6], [8], [9], [10], [11], [14] and shown that the approaches based on fuzzy-symbolic data outperforms conventional approaches. The reason is that the fuzzy-symbolic data provides a natural and realistic analysis to problems as symbolic data are more unified by means of relationships [2] and the fuzzy theory handles efficiently the impreciseness. Even though the concept of fuzzy-symbolic approach is much explored in data analysis and clustering, its inherent capabilities in preserving shape properties of an object have not yet been explored. To the best of our knowledge, there has been no attempt in proposing an approach based on fuzzy-symbolic data for shape analysis. This motivated us to think of clustering two dimensional shapes effectively by describing shapes in terms of fuzzy-symbolic features. However, there are several approaches for shape analysis based on conventional data types. Moment based features [13], signature based centroidal profile [3], Fourier Descriptors [19], Curvature Scale Space (CSS) [18], Editing shock graphs [21], polar representation of the contour points about the geometric centre of the shape [1] have been proposed. A detailed survey on shape analysis can be found in [4], [17].

It has been observed from the literature survey that most of the conventional agglomerative clustering techniques use similarity proximity matrix for clustering patterns. The agglomerative clustering merges two patterns, which possess high similarity value at each level and continue the merging process until the desired number of clusters are obtained or all the patterns are put into one cluster. However, it has been shown that about 62% of the total numbers of individuals in random artificial and natural populations are in mutual pairs. The concept of "Mutual Nearest Neighbourhood" has been introduced [7] and successfully used for agglomerative and disaggregate clustering. Thus, it is very appropriate to cluster patterns by looking at their mutual nearness than just by looking at their proximity matrix. In the proposed methodology, the concept of mutual nearest neighbourhood has been extended such that k patterns are clustered into one class if they are k-mutually nearest neighbours.

With this backdrop, in this paper, we propose a method of representing two-dimensional shapes using fuzzy-symbolic (multi interval membership values) features. Unlike other existing shape representation schemes, the proposed scheme is capable of utilizing both contour as well as region information. A similarity measure defined over the fuzzy-

Manuscript received on April 24, 2006. This work is a part of the UGC sponsored major research project No. F.30-259/2004 (SR). The UGC support is highly appreciated.

H.S.Nagendraswamy, Author, Research Scholar in the Department of Studies in Computer Science, University of Mysore, Mysore-570006, INDIA (E-mail: swamy_hsn@yahoo.com)

D.S.Guru, Author with the Department of studies in Computer Science, University of Mysore, Mysore-570006, INDIA (E-mail: guruds@lycos.com)

symbolic features for clustering shapes is also described. The concept of mutual nearest neighbourhood is extended for realistic and effective clustering of shapes. The proposed shape representation scheme is shown to be invariant to similarity transformations (rotation, translation, scale and reflection) and the clustering methodology is effective. Several experimentations have been conducted to demonstrate the success of the proposed methodology. Moreover, the proposed methodology is validated by comparing with the well known clustering methodologies in literature.

The paper is organized as follows: Section 2 presents the proposed symbolic shape representation scheme and the proposed similarity measure. Section 3 describes a novel shape clustering procedure based on the concept of *k*-mutually nearest neighbours. Experimental results are presented in section 4. A comparative study of the proposed methodology with the well known clustering techniques is given in section 5 followed by conclusions in section 6.

II. Proposed Methodology

This section explains in detail the proposed method of representing shapes and a similarity measure for estimating the degree of similarity between two shapes useful for clustering.

A. Shape representation

The proposed shape representation scheme extracts fuzzy-symbolic features from a shape with reference to the axis of least inertia, which is unique to the shape. The axis of least inertia of a shape is defined as the line for which the integral of the square of the distances to points on the shape boundary is a minimum. It serves as a unique reference line to preserve the orientation of the shape. Preserving the orientation of a shape is essential for extracting features, which are invariant to similarity transformations. However, for symmetrical shapes (star like shapes) more than one axis of least inertia can be expected. Because of shape symmetry, in spite of multiple axis of least inertia, the extracted features from a shape will not change drastically. In addition, variation in feature values due to rotation and limitations of finite precision arithmetic is recorded in our feature extraction methodology. But if the shape is major deformed or occluded then the axis of least inertia may change its orientation and hence the extracted feature vector may not agree with the feature vector of the original shape. Therefore, the proposed method of shape representation is not intended to handle major occlusions or deformations. However, the method copes well with minor occlusions and deformations.

In order to extract fuzzy-symbolic features from a shape, first, the slope angle θ of the axis of least inertia of a shape curve is estimated as described in [22]. Once the slope angle θ is calculated, each point on the shape curve is projected on to the axis of least inertia.

Let $S=\{s_1, s_2, s_3, ..., s_L\}$ represent L points on the shape curve and (x_i, y_i) denote the Cartesian co-ordinates of the point s_i for i = 1, 2, 3,…, L, Let $C=(C_x, C_y)$ denote the centroid of S. Given a point (x_i, y_i) on a shape curve, the co-ordinates of its projected point $(x_i^/, y_i^/)$ on the axis of least inertia can be found as follows:

Since, the points (C_x, C_y) and $(x_i^/, y_i^/)$ are lying on the axis of least inertia, we have

$$\frac{(C_y - y_i^/)}{(C_x - x_i^/)} = \tan\theta \Rightarrow C_y - y_i^/ = C_x \tan\theta - x_i^/ \tan\theta. \quad (1)$$

Similarly, as the point $(x_i^/, y_i^/)$ is the projection of the point (x_i, y_i), we have

$$\frac{(y_i^/ - y_b)}{(x_i^/ - x_b)} = \tan(90+\theta) \Rightarrow y_i^/ - y_b = x_i^/ \tan(90+\theta) - x_b \tan(90+\theta). \quad (2)$$

By combining equations (1) and (2), we get

$$x_i^/ = \frac{(C_y - y_b + x_b \tan(90+\theta) - C_x \tan\theta)}{(\tan(90+\theta) - \tan\theta)}.$$

From eq. (1) we have $y_i^/ = C_y + (x_i^/ - C_x)\tan\theta$

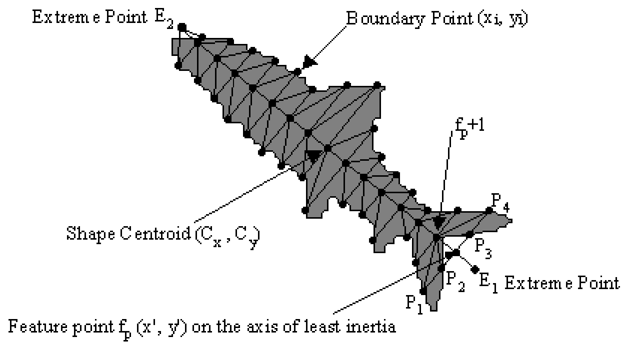

Fig. 1. Shape of an object with its axis of least inertia and extreme points

The two farthest projected points say *E1* and *E2* on the axis of least inertia are chosen as extreme points as shown in Fig. 1. The Euclidean distance between these two extreme points defines the length of the axis of least inertia.

Once the axis of least inertia of a shape is obtained, keeping it as a unique reference line, fuzzy-symbolic features are extracted from the shape. A fixed number say *n*, of equidistant points called feature points on the axis of least inertia are found (Fig. 1). The number of feature points defines the dimension of the feature vector associated with a shape. At every feature point chosen, an imaginary line perpendicular to the axis of least inertia is drawn. It is interesting to note that this perpendicular line intersects the shape curve at several points. Generally, the number of such intersecting points is even when the drawn perpendicular line is not tangential to a shape curve at any point. The number of such intersecting points is exactly two if the shape is fully convex without having holes inside. In the case of shapes with holes or with concave parts, the perpendicular line intersects the shape curve at more than two points and the number of such intersecting points is always even.

Once the intersecting points on the shape curve are identified, we extract fuzzy-symbolic features by using the concept of fuzzy equilateral triangle membership function. In this work, we have considered equilateral triangle membership function formed by any three non-collinear points. Let θ_1, θ_2 and θ_3 be the inner angles of a triangle, in the order $\theta_1 \geq \theta_2 \geq \theta_3$, and let U be the universe of triangles; i.e.,

$$U = \left\{(\theta_1, \theta_2, \theta_3) / \theta_1 \geq \theta_2 \geq \theta_3 \geq 0;\ \theta_1 + \theta_2 + \theta_3 = 180^o\right\} \quad (3)$$

We can define an approximate equilateral triangle for any triplet of angles fulfilling the constraints given in equation (3) using the following membership function [20]

$$\mu\ (\theta_1, \theta_2, \theta_3) = 1 - \tfrac{1}{180}(\theta_1 - \theta_3) \quad (4)$$

Thus the membership function $\mu\ (\theta_1, \theta_2, \theta_3)$ assigns a value between 0 and 1 to a fuzzy set of equilateral triangles. For instance, the line drawn perpendicular to the axis of least inertia at the feature point f_p intersects the shape curve at four points P_1, P_2, P_3 and P_4 as shown in Fig. 1. A triangle is formed by considering the first two farthest intersecting points P_1, P_4 and the feature point (f_p+1), then it is approximated to an equilateral triangle and its membership value (μ_1) is calculated using Eq. (4). Similarly, we form a triangle by considering the next two farthest intersecting points P_2, P_3 and the feature point (f_p+1). The equilateral triangle membership value (μ_2) associated with this triangle is also calculated. Thus $\{\mu_1, \mu_2\}$ is considered to be the feature value at the feature point f_p. The reason behind considering the farthest intersecting points while forming a triangle is that it measures the span length of the intersecting points and also the problem of sequencing feature values in case of multiple values at any particular feature point can be avoided. Notice that the feature value is not necessarily a singleton (crisp) but multivalued

The set of all extracted multivalued type features (because of n feature points chosen on the axis of least inertia) forms the feature vector of dimension n to represent the shape. In our work the value for n is set to 15 and is empirically chosen. Unlike conventional shape representation schemes, where a shape feature vector is a collection of single valued data, the proposed shape feature vector is a collection of multivalued type data. The proposed representation scheme is capable of capturing the internal details of shapes in addition to capturing the external boundary details.

Let S be the shape to be represented and n is the number of feature points chosen on its axis of least inertia. Then the feature vector F representing the shape S, is in general of the form $F = [f_1, f_2, f_3, ..., f_n]$, where $f_l = \{\mu_{l_1}, \mu_{l_2}, \mu_{l_3}, ..., \mu_{l_k}\}$ for some $l_k \geq 1$. It shall be observed that as the feature extraction process is based on the axis of least inertia and fuzzy equilateral triangle membership value, the feature vector F representing the shape S is invariant to shape similarity transformations. The angles of the triangle formed by the intersecting points at a feature point f_p with respect to feature point (f_p+1) on the axis of least inertia are scaling invariant. Thus the membership value does not change even if the shape is scaled up or scaled down uniformly. The rotational invariance is automatically achieved, as the axis of least inertia is invariant to rotation. However, due to rotation the sequence in which the features are extracted may be different. A way of alleviating this problem is to sequence the features such that the first feature corresponds to the extreme point, which is nearer to the centroid of the shape. An added advantage of the proposed scheme is that unlike other contour traversal technique based approaches, the proposed method is also invariant to flipping transformation. Generally, flipping invariance is difficult to achieve in shape analysis if a method is sensitive to the direction of contour traversal because the direction of contour traversal changes when an object/shape is flipped over. Since the axis of least inertia of a shape flips along with the shape about the same line by preserving the starting point, the proposed representation method is invariant to flipping.

In practice, however, feature values defined at a feature point on the axis of least inertia are not exactly same but they lie within certain range due to finite precision arithmetic. Thus, each component of the feature value defined at a feature point is approximated by interval data type as $f_l = \{[\mu_{l_1}^- ... \mu_{l_1}^+], [\mu_{l_2}^- ... \mu_{l_2}^+], ..., [\mu_{l_k}^- ... \mu_{l_k}^+]\}$, where μ^- and μ^+ represent, respectively, the minimum and the maximum variations of the respective component. These minimum and maximum values shall be computed by examining a shape in all possible orientations and with different scaling factors. Thus, in a more general format, the symbolic feature vector F associated with a shape S is a collection of n multi-interval-valued data type against the crisp data type in case of conventional shape analysis techniques. In practice, for the purpose of creating a shape database, we generated a large number of samples for each shape with different orientations and scaling factors. However, only one representative vector, which is an aggregation of all, is stored in the database.

B. A Similarity measure

Similarity is the fundamental notion in clustering shapes, which defines the closeness or farness of two shapes. A measure of similarity between two shapes drawn from the same feature space is essential to most of the clustering techniques. In this subsection, we present a similarity measure for estimating the degree of similarity between two shapes described by multi interval valued features, which is used for clustering of shapes.

Let $f_{Ak} = \left\{\left[\mu_{Ak_1}^-, \mu_{Ak_1}^+\right], \left[\mu_{Ak_2}^-, \mu_{Ak_2}^+\right], ..., \left[\mu_{Ak_p}^-, \mu_{Ak_p}^+\right]\right\}$ and $f_{Bk} = \left\{\left[\mu_{Bk_1}^-, \mu_{Bk_1}^+\right], \left[\mu_{Bk_2}^-, \mu_{Bk_2}^+\right], ..., \left[\mu_{Bk_q}^-, \mu_{Bk_q}^+\right]\right\}$ be the k^{th} feature component of two shapes A and B described by n multi interval valued feature vectors $F_A = [f_{A1}, f_{A2}, f_{A3}, ..., f_{An}]$ $F_B = [f_{B1}, f_{B2}, f_{B3}, ..., f_{Bn}]$ respectively. Here μ^- is the lower limit and μ^+ is the upper limit of the interval.

Since we use the concept of mutual similarity between shapes for clustering, we first estimate the degree of similarity $(S_{A \to B})$ from the shape A to the shape and B the degree of similarity $\left(S_{B\to A}\right)$ from the shape B to the shape A. Then the mutual similarity value (MSV) between the shapes A and B is defined to be the average of $S_{A\to B}$ and $S_{B\to A}$.

Since the k^{th} feature component in a feature vector is of type multi interval valued, the degree of similarity from A to B with respect to their l^{th} interval is estimated based on degree of overlapping between their lower and upper limits of their intervals. If the interval $\left[\mu^{-}_{Ak_l}, \mu^{+}_{Ak_l}\right]$ describing the l^{th} interval of the k^{th} feature of A is contained in the interval $\left[\mu^{-}_{Bk_l}, \mu^{+}_{Bk_l}\right]$ describing the l^{th} interval of the k^{th} feature of B then the degree of similarity from A to B is taken as 1, otherwise it is given by the average degree of similarity between their respective lower and upper limits.

Thus, the degree of similarity from A to B with respect to the l^{th} interval of their k^{th} feature is given by

$$S^{l}_{Ak\to Bk} = \begin{cases} 1 & if \ \left(\mu^{-}_{Ak_l} \geq \mu^{-}_{Bk_l}\right) and \ \left(\mu^{+}_{Ak_l} \leq \mu^{+}_{Bk_l}\right) \\ \frac{1}{2}\left[\eta^{-} + \eta^{+}\right] & otherwise \end{cases} \tag{5}$$

Similarly, degree of similarity from B to A with respect to the l^{th} interval of their k^{th} feature is given by

$$S^{l}_{Bk\to Ak} = \begin{cases} 1 & if \ \left(\mu^{-}_{Bk_l} \geq \mu^{-}_{Ak_l}\right) and \ \left(\mu^{+}_{Bk_l} \leq \mu^{+}_{Ak_l}\right) \\ \frac{1}{2}\left[\eta^{-} + \eta^{+}\right] & otherwise \end{cases} \tag{6}$$

where $\eta^{-} = \dfrac{1}{1 + \left|\mu^{-}_{Ak_l} - \mu^{-}_{Bk_l}\right| * \beta}$, and

β is the normalizing factor, which is set to 10.

The mutual similarity value (MSV) between A and B with respect to l^{th} interval is given by

(7)

It shall be noticed that if the intervals are one and the same then the MSV between them is 1 (maximum); otherwise the similarity value depends on the extent to which the intervals are separated. More the extent to which they are separated less shall be the degree of similarity.

It may so happen that in some cases, the number of intervals (l) in the k^{th} feature of A and B are not the same. Let the number of intervals present in the k^{th} feature of A be p and the number of feature values present in the k^{th} feature of B be q and $p<q$. In such situations we choose the first p components out of q from B for the computation of similarity; for the remaining (p-q) components of B, (for which there are no corresponding components in A for the computation of similarity), we assume the degree of similarity to be zero. Thus the overall degree of similarity between the shape A and the shape B with respect to their k^{th} feature component is given by

(8)

Thus the total degree of similarity between the shape A and B with respect to all the n features is given by

(9)

where n is the number of feature points chosen on the axis of least inertia. In our approach, feature values describing the shape are in the range 0 and 1. Since, the k^{th} feature component similarity between shape A and B is in the range 0 and 1, the total degree of similarity between the shape A and the shape B is also in the range 0 and 1.

III. Clustering of Shapes

In order to cluster shapes described by multi interval-valued features, we first compute the degree of mutual similarity among all the shapes using the proposed similarity measure as explained in section II*B* and a similarity matrix is obtained. A similarity position matrix of size (r x r) for all r shapes is created based on the descending order of their similarities. The position matrix gives the nearness position of shapes in terms of their similarity rank. The concept of k-mutual nearest neighbor, based on their position in the position matrix, is employed to cluster shapes. In a position matrix, if the shape S_i is the k^{th} nearest neighbor of the shape S_j, and the shape S_j is also the k^{th} nearest neighbor of the shape S_i, then S_i and S_j are said to be k-mutually nearest neighbors. This idea of k-mutually nearest neighbors is successfully applied in our clustering procedure. According to this idea, if m shapes are put into one cluster then all those m shapes must be k-mutually nearest neighbors. The value of k is set to 2 initially and incremented by 1 each time until we get the desired number of clusters or all shapes are put into a single cluster. Thus the proposed method of clustering shapes can be algorithmically expressed as follows.

Algorithm: *Clustering-of-Shapes*

Input : Shapes described by fuzzy-symbolic features ($S_1, S_2, S_3, \ldots, S_r$).

Output : Clusters of shapes ($C_1, C_2, \ldots, C_m$).

Method:

1. Obtain similarity matrix of size r x r using the proposed similarity measure.
2. Create a similarity position matrix of size (r x r) for all the shapes based on the descending order of their similarities.
3. Let $C=\{C_1, C_2, \ldots, C_r\}$, initially contain r number of clusters each with an individual shape.
4. Set the number of clusters (*noc*) to r.
5. Set k to 2.
6. Merge two clusters C_u and C_v, if all the shapes in C_u and C_v are k-mutual nearest neighbors.
7. Decrement the number of clusters *noc* by 1.
8. Increment k by 1.
9. Repeat steps 6 to 8 until the desired number of clusters is obtained or all the shapes are put into single a cluster.

Algorithm Ends

IV. EXPERIMENTAL RESULTS

We have conducted several experiments on variety of data sets to validate the feasibility of the proposed methodology. In this section we present the clustering results on a considerably large database of shapes.

Experiment 1: In order to illustrate the proposed method of shape clustering, we first consider a sample database of shapes consisting 9 shapes of 3 classes. The shapes to be clustered are first subjected to feature extraction process as described in section II*A* and then the similarity measure described in section II*B* is applied on the extracted features to obtain a similarity matrix. Based on the similarity matrix, a similarity position matrix is obtained. The similarity position matrix gives the closeness of the shapes in terms of their position. The proposed k-mutual nearest neighbor approach is then applied on the similarity position matrix to obtain clusters as explained in the algorithm *Clustering-of-Shapes*. Fig. 2 shows the shapes and their similarity values, Table I shows the similarity position matrix and Fig. 3 shows various stages of clustering for different values of k.

	Dog-1	Dog-2	Dog-3	Hand-1	Hand-2	Hand-3	Man-1	Man-2	Man-3
Dog-1	1.00000	0.82171	0.78039	0.63386	0.60306	0.63626	0.61334	0.65243	0.62352
Dog-2	0.82171	1.00000	0.79573	0.58355	0.65329	0.63984	0.61794	0.64026	0.62866
Dog-3	0.78039	0.79573	1.00000	0.57178	0.68986	0.62634	0.64252	0.66400	0.64919
Hand-1	0.63386	0.58355	0.57178	1.00000	0.81063	0.90650	0.63791	0.66495	0.64083
Hand-2	0.60306	0.65329	0.68986	0.81063	1.00000	0.87621	0.61034	0.63773	0.61624
Hand-3	0.63626	0.63984	0.62634	0.90650	0.87621	1.00000	0.64035	0.66731	0.64470
Man-1	0.61334	0.61794	0.64252	0.63791	0.61034	0.64035	1.00000	0.86419	0.97281
Man-2	0.65243	0.64026	0.66400	0.66495	0.63773	0.66731	0.86419	1.00000	0.85297
Man-3	0.62352	0.62866	0.64919	0.64083	0.61624	0.64470	0.97281	0.85297	1.00000

Fig. 2 Similarity matrix

TABLE I
SIMILARITY POSITION MATRIX

Shapes	Similarity Positions (k)								
	1	2	3	4	5	6	7	8	9
Dog-1	Dog-1	Dod-2	Dog-3	Man-2	Man-3	Man-1	Hand-3	Hand-2	Hand-1
Dog-2	Dog-2	Dog-1	Dog-3	Man-2	Hand-3	Hand-2	Man-3	Man-1	Hand-1
Dog-3	Dog-3	Dog-1	Dog-2	Hand-2	Man-2	Hand-3	Man-3	Man-1	Hand-1
Hand-1	Hand-1	Hand-3	Hand-2	Man-2	Dog-2	Dog-3	Dog-1	Man-1	Man-3
Hand-2	Hand-2	Hand-3	Hand-1	Dog-3	Man-2	Dog-2	Dog-1	Man-3	Man-1
Hand-3	Hand-3	Hand-1	Hand-2	Man-2	Dog-1	Dog-3	Man-3	Man-1	Dog-2
Man-1	Man-1	Man-3	Man-2	Dog-1	Hand-3	Dog-3	Dog-2	Hand-2	Hand-1
Man-2	Man-2	Man-1	Man-3	Dog-1	Hand-3	Dog-3	Dog-2	Hand-1	Hand-2
Man-3	Man-3	Man-1	Man-2	Dog-1	Hand-3	Dog-3	Dog-2	Hand-2	Hand-1

In the example, we have considered 3 categories of shapes viz., *Dog*, *Hand* and *Man*. One can observe a significant variation among shapes belonging to the same category. The similarity matrix (Fig. 2) shows that the shapes belong to the same category possess more similarity value than the shapes belong to the outside category. Similar observation can be made from the similarity position matrix (Table 1) that the shapes belong to the same category are very near and are mutual nearest neighbors.

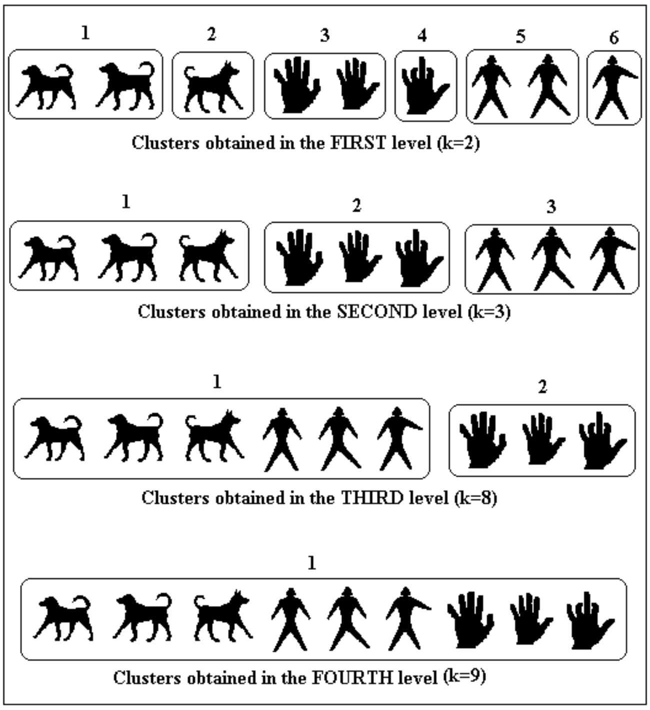

Fig. 3. Various stages of clustering

From Fig. 3, we can observe that the shapes *Dog-1* and *Dog-2*, the shapes *Hand-1* and *Hand-3* and the shapes *Man-1* and *Man-3* are *2*-mutually nearest neighbors (k = 2), hence, they are clustered in the first level itself. We can also observe that they are visually more similar than the other shapes. In the second level, the shapes *Dog-1*, *Dog-2* and *Dog-3* are put into cluster 1, the shapes *Hand-1*, *Hand-2* and *Hand-3* are put into cluster 2 and the shapes *Man-1*, *Man-2* and *Man-3* are put into cluster 3 as they are *3*-mutually nearest neighbors (k = 3). No clusters are formed for k = 4, 5, 6 and 7, this shows the clear margin between shapes belonging to same category and shapes belonging to outside the category. Shapes *Dog-1*, *Dog-2*, *Dog-3*, *Man-1*, *Man-2* and *Man-3* are clustered into one group when k = 8 and finally all the shapes are put into one cluster when k = 9.

Experiment 2: In order to demonstrate the success of the proposed shape clustering methodology, we have conducted an experiment on a considerably large database of shapes. Fig. 4 shows the 350 shapes categorized into 35 classes, each class with 10 example shapes. So it is expected to obtain 35 clusters from the proposed clustering methodology.

Figure 5 (a-d) shows the clusters obtained for the shapes shown in Fig. 4. We have shown only the last 4 levels of the clusters obtained by the proposed methodology. An ideal system is the one, which produces all the 35 true clusters for k=10 as there are 10 shapes in each category. But because of large intra class variation among shapes, we cannot expect all the true clusters for k=10. We can observe from Fig. 5(a) that 26 true clusters are formed for k=73. *APPLE* and *HEAD*, *FISH* and *LEAF* are merged together at this level to produce 2 clusters instead of 4 clusters. This is because the shapes in these categories are visually very similar to each other. That is APPLE shapes are similar to *HEAD* shapes and *FISH* category shapes are similar to *LEAF* category shapes. This can be observed in Fig. 4. Shapes in the categories *PLANE*, *HAND*, *HUMAN*, *MOMENTO* and *DOG* are not completely formed their clusters at this level because shapes in these categories possess large intra class variations as observed in Fig. 4. Totally, we obtained 38 clusters at this level. In the next level, we have obtained 37 clusters for k=82. Shape categories *BELL* and *BUTTERFLY* are merged at this level and formed single group as shown in Fig. 5(b). No changes were observed in the clusters belongs to *PLANE*, *HAND*, *HUMAN*, *MOMENTO* and *DOG* categories. In the next level, for k=84, 36 clusters are realized. At this level *CHOPPER* and *RABBIT* are merged together to form a single cluster as they appear to be similar and they are all mutually nearest neighbors for k=84 as shown in Fig. 5(c). Finally, we obtained 34 clusters for k=88 as shown in Fig. 5(d). Shape categories *HAND* and *DOG*, respectively, formed their true clear clusters. However, shape categories *PLANE*, *HUMAN* and *MOMENTO* still could not form their complete clusters. *PLANE* category results with 2 clusters with shapes {101, 102, 103, 108, 109} and {104, 105, 106, 107, 110}. *HUMAN* category results with 2 clusters with shapes {141, 142, 143, 144, 145, 146, 147, 148} and {149, 150}. Shapes {149, 150} are quite more deformed with widespread legs. Since our method of feature extraction measures such span information, these shapes are not merged with all the other shapes in its category. However, these shapes are not merged with any other shape categories. This shows that the proposed method of clustering is effective and realistic in nature. Similarly, shapes in the category MOMENTO formed 2 clusters with shapes {281, 282, 283, 284, 289, 290} and {285, 286, 287, 288}. Similar argument is true for this category also.

We have not encountered any miss grouping at any level. This is because all the shapes to be clustered into one group must be mutually nearest neighbors. So only those shapes, which belong to the same category, agree in their structure and they only can be mutually nearest neighbors. Thus the proposed mutually nearest neighbor clustering technique avoids possible miss grouping, which might arise in case of simple agglomerative clustering technique. In simple agglomerative clustering technique, we could not get clear clusters for *APPLE*, *HEAD*, *GREEBLE*, *BELT1*, *BELT2*, *FISH*, *LEAF* and *MOMENTO* categories.

Since the proposed method of shape description is based on the axis of least inertia, which depends on the global structure of the shape, more shape deformation or occlusion might change the orientation of the axis of least inertia yielding feature vectors, which are entirely different. Therefore the proposed method of shape clustering is not intended to consider major shape deformations or occlusions. But one of the strength of the proposed shape description is that it takes care of internal details of shapes such as holes, which plays a significant role in most of the shape analysis applications.

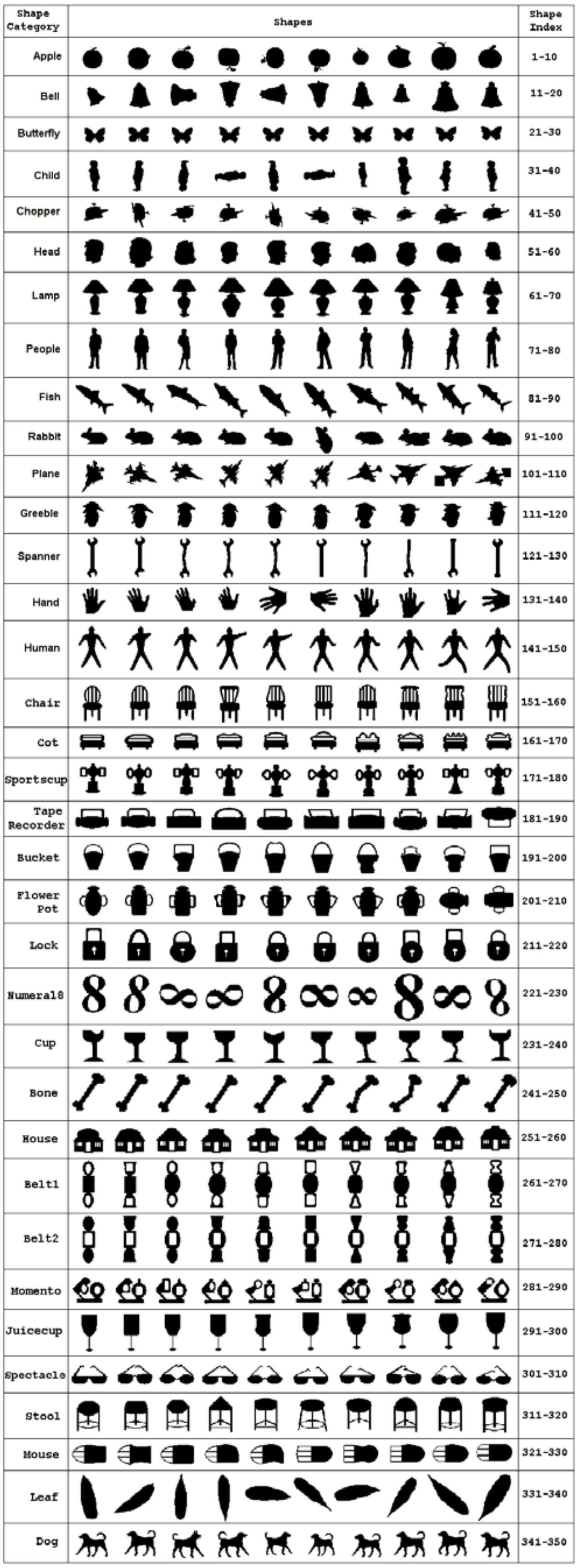

Fig. 4. Set of 35 categories of shapes each with 10 examples

Number of Clusters : 38 KMNN Value : 73		
Categories	Shape Numbers	Clusters
APPLE HEAD	1 2 3 4 5 6 7 8 9 10 51 52 53 54 55 56 57 58 59 60	1
BELL	11 12 13 14 15 16 17 18 19 20	1
BUTTERFLY	21 22 23 24 25 26 27 28 29 30	1
CHILD	31 32 33 34 35 36 37 38 39 40	1
CHOPPER	41 42 43 44 45 46 47 48 49 50	1
RABBIT	91 92 93 94 95 96 97 98 99 100	1
LAMP	61 62 63 64 65 66 67 68 69 70	1
PEOPLE	71 72 73 74 75 76 77 78 79 80	1
FISH LEAF	81 82 83 84 85 86 87 88 89 90 331 332 333 334 335 336 337 338 339 340	1
PLANE	101 102 103 108 109 104 105 106 107 110	2
GREEBLE	111 112 113 114 115 116 117 118 119 120	1
WRENCH	121 122 123 124 125 126 127 128 129 130	1
HAND	131 134 135 137 139 140 132 133 136 138	2
HUMAN	141 142 143 144 145 146 147 148 149 150	2
CHAIR	151 152 153 154 155 156 157 158 159 160	1
COT	161 162 163 164 165 166 167 168 169 170	1
SPORTSCUP	171 172 173 174 175 176 177 178 179 180	1
TAPERECORDER	181 182 183 184 185 186 187 188 189 190	1
BUCKET	191 192 193 194 195 196 197 198 199 200	1
FLOWERPOT	201 202 203 204 205 206 207 208 209 210	1
LOCK	211 212 213 214 215 216 217 218 219 220	1
NUMERAL8	221 222 223 224 225 226 227 228 229 230	1
CUP	231 232 233 234 235 236 237 238 239 240	1
BONE	241 242 243 244 245 246 247 248 249 250	1
HOUSE	251 252 253 254 255 256 257 258 259 260	1
BELT1	261 262 263 264 265 266 267 268 269 270	1
BELT2	271 272 273 274 275 276 277 278 279 280	1
MOMENTO	281 282 283 284 289 290 285 286 287 288	2
JUICECUP	291 292 293 294 295 296 297 298 299 300	1
SPECTACLE	301 302 303 304 305 306 307 308 309 310	1
STOOL	311 312 313 314 315 316 317 318 319 320	1
MOUSE	321 322 323 324 325 326 327 328 329 330	1
DOG	341 343 345 346 350 342 344 347 348 349	2

(a)

Number of Clusters : 37 KMNN Value : 82		
Categories	Shape Numbers	Clusters
APPLE HEAD	1 2 3 4 5 6 7 8 9 10 51 52 53 54 55 56 57 58 59 60	1
BELL BUTTERFLY	11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30	1
CHILD	31 32 33 34 35 36 37 38 39 40	1
CHOPPER	41 42 43 44 45 46 47 48 49 50	1
RABBIT	91 92 93 94 95 96 97 98 99 100	1
LAMP	61 62 63 64 65 66 67 68 69 70	1
PEOPLE	71 72 73 74 75 76 77 78 79 80	1
FISH LEAF	81 82 83 84 85 86 87 88 89 90 331 332 333 334 335 336 337 338 339 340	1
PLANE	101 102 103 108 109 104 105 106 107 110	2
GREEBLE	111 112 113 114 115 116 117 118 119 120	1
WRENCH	121 122 123 124 125 126 127 128 129 130	1
HAND	131 134 135 137 139 140 132 133 136 138	2
HUMAN	141 142 143 144 145 146 147 148 149 150	2
CHAIR	151 152 153 154 155 156 157 158 159 160	1
COT	161 162 163 164 165 166 167 168 169 170	1
SPORTSCUP	171 172 173 174 175 176 177 178 179 180	1
TAPERECORDER	181 182 183 184 185 186 187 188 189 190	1
BUCKET	191 192 193 194 195 196 197 198 199 200	1
FLOWERPOT	201 202 203 204 205 206 207 208 209 210	1
LOCK	211 212 213 214 215 216 217 218 219 220	1
NUMERAL8	221 222 223 224 225 226 227 228 229 230	1
CUP	231 232 233 234 235 236 237 238 239 240	1
BONE	241 242 243 244 245 246 247 248 249 250	1
HOUSE	251 252 253 254 255 256 257 258 259 260	1
BELT1	261 262 263 264 265 266 267 268 269 270	1
BELT2	271 272 273 274 275 276 277 278 279 280	1
MOMENTO	281 282 283 284 289 290 285 286 287 288	2
JUICECUP	291 292 293 294 295 296 297 298 299 300	1
SPECTACLE	301 302 303 304 305 306 307 308 309 310	1
STOOL	311 312 313 314 315 316 317 318 319 320	1
MOUSE	321 322 323 324 325 326 327 328 329 330	1
DOG	341 343 345 346 350 342 344 347 348 349	2

(b)

Number of Clusters : 36 KMNN Value : 84		
Categories	Shape Numbers	Clusters
APPLE HEAD	1 2 3 4 5 6 7 8 9 10 51 52 53 54 55 56 57 58 59 60	1
BELL BUTTERFLY	11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30	1
CHILD	31 32 33 34 35 36 37 38 39 40	1
CHOPPER RABBIT	41 42 43 44 45 46 47 48 49 50 91 92 93 94 95 96 97 98 99 100	1
LAMP	61 62 63 64 65 66 67 68 69 70	1
PEOPLE	71 72 73 74 75 76 77 78 79 80	1
FISH LEAF	81 82 83 84 85 86 87 88 89 90 331 332 333 334 335 336 337 338 339 340	1
PLANE	101 102 103 108 109 104 105 106 107 110	2
GREEBLE	111 112 113 114 115 116 117 118 119 120	1
WRENCH	121 122 123 124 125 126 127 128 129 130	1
HAND	131 134 135 137 139 140 132 133 136 138	2
HUMAN	141 142 143 144 145 146 147 148 149 150	2
CHAIR	151 152 153 154 155 156 157 158 159 160	1
COT	161 162 163 164 165 166 167 168 169 170	1
SPORTSCUP	171 172 173 174 175 176 177 178 179 180	1
TAPERECORDER	181 182 183 184 185 186 187 188 189 190	1
BUCKET	191 192 193 194 195 196 197 198 199 200	1
FLOWERPOT	201 202 203 204 205 206 207 208 209 210	1
LOCK	211 212 213 214 215 216 217 218 219 220	1
NUMERAL8	221 222 223 224 225 226 227 228 229 230	1
CUP	231 232 233 234 235 236 237 238 239 240	1
BONE	241 242 243 244 245 246 247 248 249 250	1
HOUSE	251 252 253 254 255 256 257 258 259 260	1
BELT1	261 262 263 264 265 266 267 268 269 270	1
BELT2	271 272 273 274 275 276 277 278 279 280	1
MOMENTO	281 282 283 284 289 290 285 286 287 288	2
JUICECUP	291 292 293 294 295 296 297 298 299 300	1
SPECTACLE	301 302 303 304 305 306 307 308 309 310	1
STOOL	311 312 313 314 315 316 317 318 319 320	1
MOUSE	321 322 323 324 325 326 327 328 329 330	1
DOG	341 343 345 346 350 342 344 347 348 349	2

(c)

Number of Clusters : 34 KMNN Value : 88		
Categories	Shape Numbers	Clusters
APPLE HEAD	1 2 3 4 5 6 7 8 9 10 51 52 53 54 55 56 57 58 59 60	1
BELL BUTTERFLY	11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30	1
CHILD	31 32 33 34 35 36 37 38 39 40	1
CHOPPER RABBIT	41 42 43 44 45 46 47 48 49 50 91 92 93 94 95 96 97 98 99 100	1
LAMP	61 62 63 64 65 66 67 68 69 70	1
PEOPLE	71 72 73 74 75 76 77 78 79 80	1
FISH LEAF	81 82 83 84 85 86 87 88 89 90 331 332 333 334 335 336 337 338 339 340	1
PLANE	101 102 103 108 109 104 105 106 107 110	2
GREEBLE	111 112 113 114 115 116 117 118 119 120	1
WRENCH	121 122 123 124 125 126 127 128 129 130	1
HAND	131 132 133 134 135 136 137 138 139 140	1
HUMAN	141 142 143 144 145 146 147 148 149 150	2
CHAIR	151 152 153 154 155 156 157 158 159 160	1
COT	161 162 163 164 165 166 167 168 169 170	1
SPORTSCUP	171 172 173 174 175 176 177 178 179 180	1
TAPERECORDER	181 182 183 184 185 186 187 188 189 190	1
BUCKET	191 192 193 194 195 196 197 198 199 200	1
FLOWERPOT	201 202 203 204 205 206 207 208 209 210	1
LOCK	211 212 213 214 215 216 217 218 219 220	1
NUMERAL8	221 222 223 224 225 226 227 228 229 230	1
CUP	231 232 233 234 235 236 237 238 239 240	1
BONE	241 242 243 244 245 246 247 248 249 250	1
HOUSE	251 252 253 254 255 256 257 258 259 260	1
BELT1	261 262 263 264 265 266 267 268 269 270	1
BELT2	271 272 273 274 275 276 277 278 279 280	1
MOMENTO	281 282 283 284 289 290 285 286 287 288	2
JUICECUP	291 292 293 294 295 296 297 298 299 300	1
SPECTACLE	301 302 303 304 305 306 307 308 309 310	1
STOOL	311 312 313 314 315 316 317 318 319 320	1
MOUSE	321 322 323 324 325 326 327 328 329 330	1
DOG	341 342 343 344 345 346 347 348 349 350	1

(d)

Fig. 5 Last four levels of clusters obtained by the proposed methodology for shapes shown in Fig. 4

V. DISCUSSIONS AND COMPARATIVE STUDY

In order to show the feasibility of the proposed similarity measure for symbolic patterns and the proposed mutually nearest neighbourhood clustering technique, we have conducted several experiments on variety of data sets. In this section we present the clustering results on a few well-known data sets.

A. Experiment 1

The first experiment is conducted on Ichino's fat oil data, which has been used by several researchers as a typical example of a data set involving interval-valued features. It is composed of eight patterns described by four interval-valued features [14]. The proposed similarity measure is employed on the data set to estimate the degree of similarity and the similarity matrix is obtained (see Table II). Based on the similarity matrix, a position matrix, which gives the relative nearness of patterns in the descending order of similarity, is created (see Table III). The concept of *k*-mutual nearest neighborhood is employed on the position matrix and the dendrogram representation of the cluster formed is shown in the Fig. 6. A dendrogram is a special type of tree structure that provides a convenient picture of a hierarchical clustering [15]. It consists of layers of nodes, each representing a cluster and lines connecting nodes to represent clusters which are nested into one another. Cutting dendrogram horizontally creates a clustering.

From Fig. 6, we can observe that the patterns {1, 2}, {3, 4}, {5, 6} and {7, 8} are 2-mutually nearest neighbors (k=2). That means, the pattern 2 is in the second position according to similarity rank for the pattern 1. Similarly, the pattern 1 is in the second position according to similarity rank for the pattern 2. One can notice this fact from the similarity matrix shown in Table II and the position matrix shown in Table III. Thus the patterns {1, 2} are clustered in the first level itself (dendrogram). The above argument is true for the patterns {3, 4}, {5, 6} and {7, 8}. The Patterns {3, 4} and {5, 6} are 5-mutually nearest neighbors (See Table III) and hence they are clustered in the second level (for *k*=5) as shown in the dendrogram (Fig. 6). As there are no patterns, which are mutually nearest neighbors for *k*=3 and *k*=4, no patterns are merged for *k*=3 and *k*=4. The patterns {3, 4, 5, 6} and {1, 2} are 6-mutually nearest neighbors and are clustered at the level 3. No clusters are formed for (*k*=7) and finally we get a single cluster at the level 4 for *k*=8. We can cut the dendrogram at any level and realize the clusters depending on the desired number of clusters.

TABLE II
SIMILARITY MATRIX FOR FAT OIL

	1	2	3	4	5	6	7	8
1	1.00000	0.69303	0.47567	0.60109	0.67750	0.54414	0.33697	0.32880
2	0.69303	1.00000	0.66609	0.66711	0.55824	0.58490	0.42265	0.41518
3	0.47567	0.66609	1.00000	0.80402	0.71845	0.74422	0.47229	0.47131
4	0.60109	0.66711	0.80402	1.00000	0.63999	0.68102	0.40771	0.41510
5	0.67750	0.55824	0.71845	0.63999	1.00000	0.84983	0.42711	0.47161
6	0.54414	0.58490	0.74422	0.68102	0.84983	1.00000	0.45556	0.48231
7	0.33697	0.42265	0.47229	0.40771	0.42711	0.45556	1.00000	0.71059
8	0.32880	0.41518	0.47131	0.41510	0.47161	0.48231	0.71059	1.00000

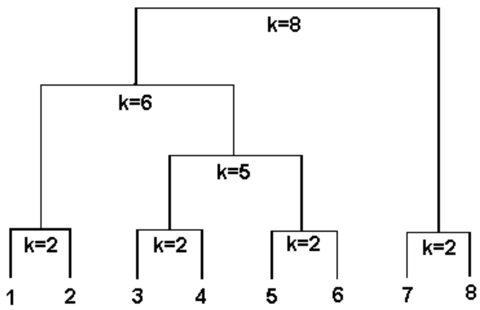

Fig. 6.Dendrogram representation of the clusters formation at various levels for Fat-Oil data by the proposed methodology

TABLE III
POSITION MATRIX OBTAINED BASED ON THE SIMILARITY RANK

Pattern Number	Similarity Positions							
	1	2	3	4	5	6	7	8
P_1	P_1	P_2	P_5	P_4	P_6	P_3	P_7	P_8
P_2	P_2	P_1	P_4	P_3	P_6	P_5	P_7	P_8
P_3	P_3	P_4	P_6	P_5	P_2	P_1	P_7	P_8
P_4	P_4	P_3	P_6	P_2	P_5	P_1	P_8	P_7
P_5	P_5	P_6	P_3	P_1	P_4	P_2	P_8	P_7
P_6	P_6	P_5	P_3	P_4	P_2	P_1	P_8	P_7
P_7	P_7	P_8	P_3	P_6	P_5	P_2	P_4	P_1
P_8	P_8	P_7	P_6	P_5	P_3	P_2	P_4	P_1

B Experiment 2

We have also conducted an experiment on the data set given in [14] on microcomputers. The data set describes a group of microcomputers consisting of 12 patterns. Each pattern has five features. Two of the features are qualitative (Display and Microprocessor) and the rest are quantitative (RAM, ROM and Keys). The similarity proximity matrix for this data set is given in the Table IV. The dendrogram representation of the clusters formed at various levels is shown in Fig. 7. The proposed algorithm resulted in two clusters {1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 12} and {7}.

In order to validate the correctness, we have compared the results of the proposed method with that of other available methodologies. For this purpose we have considered six methodologies which are listed in Table V.
Table V summarizes the results obtained through the applications of all the seven methodologies including the proposed methodology on two different data sets viz., Fats and Oils data and Microcomputer data, as the results on these two data sets are studied by all the six methodologies.

It can be noticed in Table V that the fats and oils patterns are either grouped into 2 clusters or into 3 clusters. The methods [6], [9], [11], [14] have grouped the patterns into 2 clusters ({1,2,3,4,5,6},{7,8}) and the methods [5], [8] have grouped the patterns into 3 clusters ({1,2}, {3,4,5,6}, {7,8}) based on their own cluster indicator function which acts as a stopping criterion. The entries not available in the Table V denote that the corresponding result has not been shown in the respective research work. We have not computed the same during experimentation as those methodologies require a prior knowledge of the number of samples in each pattern, which is indeed a real drawback of those approaches. Authors [14]

TABLE IV
SIMILARITY MATRIX FOR MICROCOMPUTER DATA

	1	2	3	4	5	6	7	8	9	10	11	12
1	1.00000	0.91429	0.70804	0.70595	0.42657	0.48797	0.29192	0.52454	0.83926	0.96234	0.79567	0.58724
2	0.91429	1.00000	0.73916	0.79167	0.45768	0.51908	0.29979	0.61026	0.80718	0.93025	0.76359	0.61835
3	0.70804	0.73916	1.00000	0.55669	0.50429	0.54676	0.43697	0.45559	0.83807	0.71499	0.60941	0.62372
4	0.70595	0.79167	0.55669	1.00000	0.63959	0.62860	0.34804	0.71978	0.59884	0.72192	0.84136	0.70882
5	0.42657	0.45768	0.50429	0.63959	1.00000	0.91765	0.43089	0.69314	0.40420	0.43351	0.55682	0.76388
6	0.48797	0.51908	0.54676	0.62860	0.91765	1.00000	0.43281	0.77549	0.46561	0.49491	0.60444	0.80192
7	0.29192	0.29979	0.43697	0.34804	0.43089	0.43281	1.00000	0.28654	0.41715	0.29407	0.34471	0.46404
8	0.52454	0.61026	0.45559	0.71978	0.69314	0.77549	0.28654	1.00000	0.51120	0.54051	0.65003	0.57741
9	0.83926	0.80718	0.83807	0.59884	0.40420	0.46561	0.41715	0.51120	1.00000	0.87692	0.71026	0.47110
10	0.96234	0.93025	0.71499	0.72192	0.43351	0.49491	0.29407	0.54051	0.87692	1.00000	0.83333	0.59418
11	0.79567	0.76359	0.60941	0.84136	0.55682	0.60444	0.34471	0.65003	0.71026	0.83333	1.00000	0.75132
12	0.58724	0.61835	0.62372	0.70882	0.76388	0.80192	0.46404	0.57741	0.47110	0.59418	0.75132	1.00000

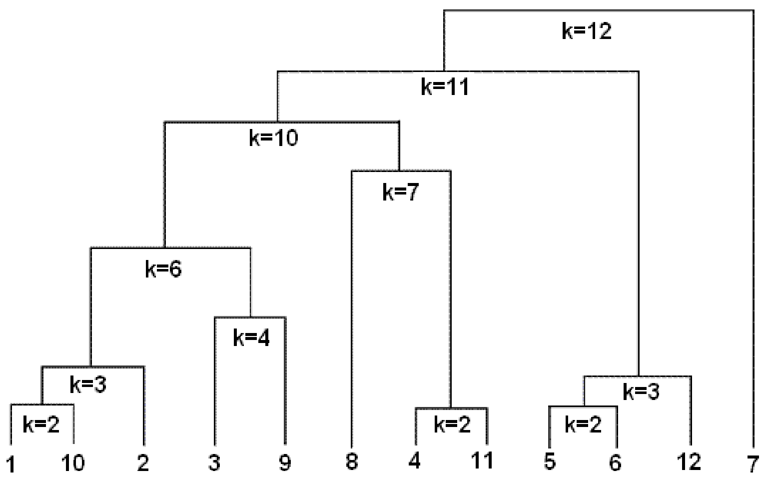

Fig. 7. Dendrogram representation of the clusters formed at various levels for Microcomputer data by the proposed methodology

have given the clustering of samples grouped into 2 clusters and as well as the samples grouped into 3 clusters. When our method is employed on the fats and oils data and the dendrogram (Fig. 6) is cut at the level of 3 clusters, the results are same as that of all the methods which yield 3 clusters [5], [8], [11], [14] and when agglomeration is allowed to continue up to 2 clusters then the result obtained is exactly same as that of the methods which yield 2 clusters [6], [9], [11], [14]. This shows consistency in the results of all the considered and our method on the fats and oils data.

It can also be noticed from Table V that the results obtained on Microcomputer data through all the 6 approaches are entirely different except the results of the methods [9], [11] and [14]. In the work [6], it is stated that no consistency can be expected on Microcomputer data. However, our method has resulted with 2 clusters, which are same as that of the methods [9], [11] and [14] encouraging their results.

VI CONCLUSIONS

In this paper, a new similarity measure useful for clustering shapes described by fuzzy-symbolic features is proposed. The concept of *k*-mutually nearest neighbours used for clustering shapes provides a realistic insight into the closeness among the shapes in a cluster. One can accept the idea that two set of shapes are grouped together to form a single cluster only when all the shapes in both the clusters are *k*-mutually nearest neighbours and this fact can be revealed by the results of the proposed methodology on the shape database and also on the standard data sets. The efficacy of the proposed methodology is experimentally established and its validity is tested by comparing with the well-known methodologies.

TABLE V
RESULTS BASED COMPARISION

Methodology	Fats and oils		Microcomputer	
	Description at 2 Clusters level	Description at 3 Clusters level	Description at 2 cluster level	Description at level more than or equal to 3 clusters
Ichino and Yaguchi (1994)	{1,2,3,4,5,6} {7,8}	{1,2} {3,4,5,6} {7,8}	{1,2,3,4,5,6,8,910,11,12} {7}	{1,2,3,9,10} {4,5,6,8,11,12} {7}
Gowda and Ravi (1995(a))	Not available	{1,2} {3,4,5,6} {7,8}	Not available	{1,2,4,6,8,9,10,11,12} {3} {7} {5}
Gowda and Diday (1991)	Not available	{1,2} { 3,4,5,6} {7,8}	Not available	{ 1,2,4,10,11} {7} {3,9} {5,6,12} {8}
Gowda and Diday (1992)	{1,2,3,4,5,6} {7,8}	Not available	Not available	{1,2,10,11} {7} {3,9} {4,5,6,8,12}
Gowda and Ravi (1995(b))	{1,2,3,4,5,6} {7,8}	Not available	{1,2,3,4,5,6,8,9,10,11,12} {7}	Not available
Guru et al. (2004)	{1,2,3,4,5,6} {7,8}	{1,2} {3,4,5,6} {7,8}	{1,2,3,4,5,6,8,9,10,11,12} {7}	{1,2,3,4,9,10,11} {4,5,6,12} {7} {8}
Proposed method	{1,2,3,4,5,6} {7,8}	{1,2} {3,4,5,6} {7,8}	{1,2,3,4,5,6,8,910,11,12} {7}	{1,2,3,4,8,9,10,11} {7} {5,6,12}

REFERENCES

[1] Bernier T and J. A. Landry, 2003. "A new method for representing and matching shapes of natural objects". Pattern Recognition, Vol. 36, pp. 1711-1723.

[2] Bock H.H and E. Diday (Eds.), 2000. "Analysis of symbolic data". Springer Verlag.

[3] Davies E.R., 1990. Machine Vision: Theory, Algorithms and Practicalities. Academic press, New York

[4] Dengsheng Zhang and Guojun Lu, 2004. "Review of shape representation and description techniques". Pattern Recognition, Vol. 37, pp. 1-19.

[5] Gowda K.C and E. Diday, 1991. "Symbolic Clustering using a new dissimilarity measure". Pattern Recognition, Vol 24, No 6, pp 567-578.

[6] Gowda K.C and Diday E., 1992. "Symbolic Clustering using a new similarity measure". IEEE Transactions on System, Man and Cybernetics. Vol. 22, No 2, pp 368-378.

[7] Gowda K.C. and G. Krishna, 1977. “Agglomerative clustering using the concept of Mutual Nearest Neighborhood”. Pattern Recognition, Vol. 10, pp. 105-112.

[8] Gowda K.C. and Ravi T.V., 1995 (a). “Agglomerative clustering of Symbolic Objects using the concepts of both similarity and dissimilarity”. Pattern Recognition Letters. 16, pp 647-652.

[9] Gowda K.C. and Ravi T.V., 1995 (b). “Divisive Clustering of Symbolic Objects using the concepts of both similarity and dissimilarity”. Pattern Recognition, Vol. 28, No 8, pp 1277-1282.

[10] Guru D.S., and Bapu B. Kiranagi, 2005. “Multivalued type dissimilarity measure and the concept of mutual dissimilarity for clustering symbolic patterns". Pattern Recognition. Vol. 38, No. 1, pp. 151-156.

[11] Guru D.S., Bapu B. Kiranagi and P. Nagabhushan, 2004. “Multivalued type proximity measure and concept of mutual similarity value useful for clustering symbolic patterns". Pattern Recognition Letters. Vol. 25, No. 10, pp. 1203-1213.

[12] Hartigan J.A., 1995, "Clustering algorithms". Wiley New York.

[13] Hu M.K., 1962. Visual Pattern Recognition by moment invariants, IRE Transaction on Information Theory Vol. 8, No. 2, pp.179-187.

[14] Ichino M and H. Yaguchi, 1994. “Generalized Minkowski metrices for mixed feature type data analysis”. IEEE Transactions on System, Man and Cybernetics. Vol. 24, No. 4, April.

[15] Jain A. K. and C.R. Dubes, 1988. “Algorithms for Clustering Data”. Prentice Hall, Engle Wood Cliffs.

[16] Jain A. K., Murthy M. N. and Flynn P. J., 1999. "Data Clustering: A Review". ACM Computing Surveys, Vol. 31, No. 3, pp 264-324.

[17] Loncaric S., 1998. "A survey of shape analysis techniques". Pattern Recognition, Vol. 31, No. 8, pp. 983-1001.

[18] Mokhtarian F., and Mackworth A., 1986. "Scale-based description and recognition of planar curves and two-dimensional shapes". IEEE Pattern Analysis and Machine Intelligence. Vol. 8, pp. 34-43.

[19] Persoon E. and Fu K.S., 1977. Shape discrimination using Fourier Descriptors. IEEE Transactions. Systems Man Cybernatics, Vol. 7, pp. 170-179.

[20] Ross T.J., 2004, "Fuzzy logic with engineering applications". Second Edition, John Wiley & Sons Ltd.

[21] Sebastian T. B., Klein P.N. and Kimia B.B., 2001. "Recognition of shapes by editing shock graphs". Proc. 8th IEEE International conference on computer Vision (ICCV), Vol. 1, pp. 755-762.

[22] Tsai D.M. and Chen M.F., 1995. "Object Recognition by linear weight classifier". Pattern Recognition Letters, Vol. 16, pp. 591-600.