THE PUBLISHING HOUSE
OF THE ROMANIAN ACADEMY

PROCEEDINGS OF THE ROMANIAN ACADEMY, Series A,
Volume 7, Number 1/2006, pp. 000-000

AN AUTOMATIC UNSUPERVISED PATTERN RECOGNITION APPROACH

Tudor BARBU

Institute for Computer Sciences of the Romanian Academy, Iaşi branch, Romania
Corresponding author: tudbar@iit.tuiasi.ro

In this paper, we propose an automatic unsupervised classification technique. The method works successfully for any kind of feature vectors, therefore we insist on classification step of recognition process only. First we propose an semiautomatic unsupervised classification approach, based on region-growing method. Then, by eliminating the condition of knowing the number of classes, we obtain an automatic clustering procedure. This method can be successfully applied in various domains which use pattern recognition. Thus, it can be used to perform classification of the components of a large media database, where the number of classes cannot be set through interactivity.

Key words: pattern recognition; feature vectors; unsupervised classification; automatic clustering; region-growing, distance, object, media entities.

1. INTRODUCTION

Our work approaches the pattern recognition domain in the general case. As we know, any pattern recognition system consists of two main processes: the feature vector extraction and the feature vector classification ([1]-[5]). Considering a set of entities to be recognized, these two operations are successively applied on that set.

The first step of recognition, feature detection, consists of computing the feature vector of each entity from the set ([2]). Obviously, the form of the obtained vector depends on the type of object we want to recognize. Thus, these feature vectors could be unidimensional vectors, matrices or multidimensional vectors.

We do not insist on feature extraction in this work, focusing on the second step of the pattern recognition process instead. The pattern classification consists in grouping these previously obtained feature vectors in several classes of similarity. Depending on the feature vector type, various linear or nonlinear metrics can be used in the classification process ([1],[2],[5]-[7]).

Depending on the existence of a training set, a classification technique can be either supervised or unsupervised ([1],[5],[6]). This paper approaches the unsupervised classification (recognition), known also as *clustering*, only ([2]-[5]). Semiautomatic unsupervised classification implies the human interactivity in the feature vector clustering process. Most unsupervised recognition systems require some *a priori* knowledge about the entities they operate on. Therefore, some essential parameters, like the number of classes or the threshold for distances between vectors, have to be set by the user.

Data clustering algorithms can be *hierarchical* or *partitional* ([2],[5]). With hierarchical algorithms, successive clusters are found using previously established clusters, whereas partitional algorithms determine all clusters in one go. Hierarchical algorithms can be agglomerative (bottom-up) or divisive (top-down). As examples of such procedures we mention the *region-growing* methods ([3],[6],[8],[9]). The partitional clustering methods include algorithms like *K-means*, which requires knowledge about the number of the set partitions (*K*), *QT CLust* algorithm, requiring a distance threshold setting, or *Fuzzy C-means* clustering ([1],[2],[5],[6]).

Recommended by Horia-Nicolae TEODORESCU, member of the Romanian Academy

The automatic clustering approaches does not utilize any previous knowledge about the classes to be obtained. Therefore, automatic unsupervised classification does not need interactivity, the user being not involved at all. It is useful in performing recognition over very large sets of objects, where the semiautomatic techniques fail to set the real number of classes. Automatic methods are faster than the semiautomatic ones, but usually they obtain smaller recognition rates.

We provide such an automatic unsupervised classification approach, working for entities of any type. It uses an region-growing type algorithm ([3]). Thus, in the next section we propose a region-growing based semiautomatic clustering technique which uses knowledge about the number of classes. Then, in the third chapter we use and extend this technique, obtaining a classification method which needs no previous knowledge about number of clusters or any other parameters. The two unsupervised classification approaches represent the main contributions of this paper.

The obtained automatic clustering procedure have been tested on many data sets. Some testing results of our experiments are described in the fourth section of this work. Also, various possible applications, mainly in multimedia recognition domains, are discussed in the same section. Our work ends with a chapter of conclusions and a references section.

2. SEMIAUTOMATIC UNSUPERVISED CLASSIFICATION METHOD

In this chapter, we will describe a semiautomatic clustering technique, based on a region-growing algorithm. Region growing general scheme usually starts with a number of regions and, at each step, several regions are somehow merged into larger regions, until the desired number of regions is achieved ([3],[6]).

Now, let us consider the following recognition problem. Suppose we have a set of entities (objects) to be recognized (classified), and that set is formally written as $S = \{Ob_1,...,Ob_n\}$. After performing a proper feature extraction process on *S*, we obtain the feature vector set $\{V(Ob_1),...,V(Ob_n)\}$. Clustering these feature vectors results in obtaining a classification for the objects from *S*.

The classes we try to obtain have to satisfy the main class property: for any object $Ob_i \in S$, the object from *S* which is most similar to Ob_i must belong to the same class with it. The region-growing based classification procedure proposed by us uses the concept of distance between clusters. We try to define this distance such as the resulted clusters to satisfy the specified property.

There are many ways to define the distance between feature vector sets (classes). Thus, one solution is to consider the distance between classes as being the distance between their *centers of gravity* (*centroides*). Another way is to use the Hausdorff metric for sets ([7]-[9]). Maximum distance between elements of each cluster (*complete linkage clustering*) ([2]), the sum of all intra cluster variance or the increase in variance for the cluster being merged (Ward's criterion) represent other possible defining ways.

We consider two methods as being most proper for computing the distances between our clusters of objects. The first metric is the *single linkage clustering* ([2]), computed as the minimum distance elements of each cluster:

$$d(C_1,C_2) = \min_{x\in C_1, y\in C_2} dist(x,y), \tag{1}$$

where C_1 and C_2 represent feature vector classes and *dist* is a proper metric for comparing feature vectors *x* and *y*. The second possible distance to be used is the *average linkage clustering*, computed as the mean distance between the elements of each cluster:

$$d(C_1,C_2) = \frac{\sum_{x\in C_1}\sum_{y\in C_2} dist(x,y)}{card(C_1)\cdot card(C_2)} \tag{2}$$

The region-growing classification procedure proposed by us is characterized by the following main processing steps:

1. Sets (interactively) the desired number of clusters, $K \le n$.
2. Starts the process with *n* clusters: $C_1 = \{V(Ob_1)\},...,C_n = \{V(Ob_n)\}$.
3. At each iteration computes the overall minimum distance between clusters and merges those being at that distance from each other:

$$\forall i < j,\ d(C_i, C_j) = d_{\min} \Rightarrow C_i := C_i \cup C_j, C_j := \phi, \tag{3}$$

where

$$d_{\min} = \min_{i \neq j \in [1,n]} d(C_i, C_j), \tag{4}$$

and distance *d* is computed from (1) or (2).

4. Stops when the *optimization criterion* is achieved: the number of clusters becomes *K*.

After clustering the feature vectors in these *K* classes, we obtain easily the *K* classes corresponding to the objects of initial set *S*. The presented method becomes automatic when this value *K* is not set interactively, being *a priori* known. When semiautomatic, this technique produces satisfactory recognition rates for not very large *n* values.

5. AUTOMATIC UNSUPERVISED CLASSIFICATION TECHNIQUE

In this section we extend the presented semiautomatic classification method and obtain an automatic variant of it which does not require any previous knowledge about number of clusters. This means it does not need any interactivity also. We use the same set of objects to be classified, *S*, and the same feature set as in the previous case.

The clustering algorithm we propose consists of two main parts. The first one operates on the feature vectors, while the second operates on the distances between them. Thus, first part resembles the region-growing algorithm described in the second chapter. It starts, like the previous one, with all feature vectors as clusters. Then, it unifies these clusters using the same method, but eliminating the optimization criterion condition. While the first region-growing algorithm stops the iterations when it reaches a certain number of classes, this one is not stopped by any *K* value condition. Thus, after performing clustering process using single linkage clustering, all regions are finally merged into a single cluster.

The second part of the procedure analyzes the computed minimum distances between clusters. The previous region-growing algorithm is thus applied to them, these distance values being clustered in two categories: "*large*" distances and "*small*" distances. The small distances are those who determine the feature vector classes, and the corresponding object classes. Thus, this automatic classification procedure is described by the following steps:

1. Initialize a distance set: $D := \phi$.
2. Starts the classification process with the *n* initial clusters: $C_1 = \{V(Ob_1)\},...,C_n = \{V(Ob_n)\}$.
3. At each iteration computes the overall minimum distance between clusters and merges those being at that distance from each other. The formulas (3) and (4) are applied again here, but distance *d* can be computed from (1) only. Minimum distance is then registered: $D := D \cup \{d_{\min}\}$.
4. When a single cluster remains, a new clustering process is performed on the distance set *D*, using the previous region-growing algorithm with parameter *K* = 2. Two classes containing distance values are thus obtained.
5. One element from each class is randomly selected and the two distance values are then compared. The class corresponding to the greater value represents the set of large distances, let it be D_l. The smaller value belongs to the set of small distances, D_s. Obviously, $D = D_l \cup D_s$.
6. Each object receives its order number as an initial class label: $\forall i \in [1,n], C(Ob_i) := i$.
7. For any small distance, it searches for all pairs of vectors corresponding to it and the objects related to the feature vectors from each pair will be inserted in the same class:

$$\forall dis \in D_s, \forall i < j \in [1,n],\ if\ d(V(Ob_i),V(Ob_j)) = dis \Rightarrow C(Ob_j) := i \quad (5)$$

Thus, this clustering algorithm assigns a class label to each object of set *S*. The classification performed this way is totally automatic, no interactivity being present. Although the region-growing algorithm described in last section is used here for distance classification, the process remains automatic because the number of classes, *K*, is already known. The provided automatic unsupervised classification (recognition) approach works successfully on large sets of objects, therefore for great *n* values too.

8. EXPERIMENTS AND APPLICATION AREAS

We have performed a lot of recognition experiments using the proposed clustering approach. We present now such a practical test. Thus, we consider a set of objects to be recognized, $S = \{Ob_1,...,Ob_8\}$. Let us suppose that, after performing a feature extraction process, eight feature vectors in the bidimensional feature space are obtained. Let them be as follows: $V(Ob_1) = [0,5]$, $V(Ob_2) = [1,5]$, $V(Ob_3) = [1,7]$, $V(Ob_4) = [2,6]$, $V(Ob_5) = [2,2]$, $V(Ob_6) = [3,6]$, $V(Ob_7) = [3,3]$, $V(Ob_8) = [5,3]$, $V(Ob_9) = [7,5]$.

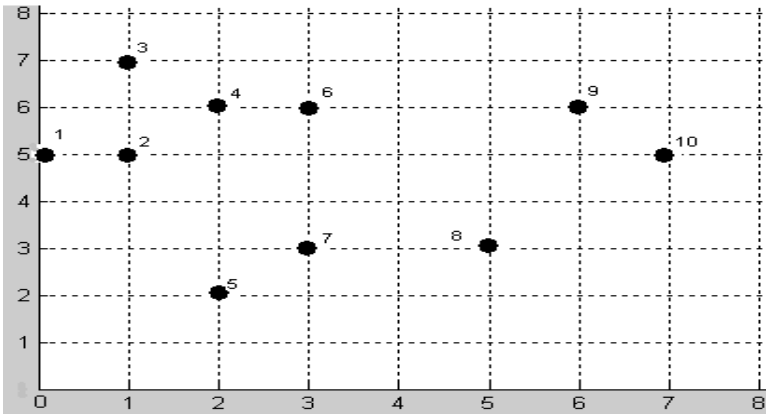

Fig.1. Representation of feature vectors as points in 2*D* space

All these feature vectors are displayed in Fig.1, being represented as points in the bidimensional feature space. Each vector $V(Ob_i)$ is marked through its *i* value in the above figure. Running the newly proposed automatic clustering procedure on this graphical data set, we build first a circuitless connected graph. Thus, for each minimum distance between clusters, the corresponding graphical points have to be unified by a segment in the picture.

Thus, the first minimum distance between any two points belonging to different clusters is $d(V(Ob_1),V(Ob_2)) = d(V(Ob_4),V(Ob_6)) = 1$, where metric *d* is the Euclidean distance. Therefore, there are obtained the new clusters {1,2} and {4,6}, their points being linked by segments. Next minimum distance value is $d(V(Ob_2),V(Ob_4)) = d(V(Ob_3),V(Ob_4)) = d(V(Ob_5),V(Ob_7)) = d(V(Ob_9),V(Ob_{10})) = \sqrt{2}$, therefore new edges are added to the graph and new classes of points result: {1,2,3,4,6}, {5,7}, {9,10} and {8}. Next minimum distance is $d(V(Ob_7),V(Ob_8)) = 2$, therefore edge (7, 8) is added and cluster {5,7,8} result. Last computed minimum distance value is $d(V(Ob_2),V(Ob_7)) = d(V(Ob_8),V(Ob_{10})) = \sqrt{8}$, the edges (2,7) and (8,10) being included in the graph. Thus all the points become connected by edges and the desired cluster, represented by their tree, is obtained as in Fig.2.

Next, these nine obtained edges have to be clustered into two classes. It is enough to classify only their four length values (corresponding to minimum distances), $\{1, 2, \sqrt{2}, \sqrt{8}\}$, using the absolute difference as metric and the region-growing algorithm.

First minimum distance between any two of these values is $\sqrt{2} - 1 \cong 0.41$, therefore we obtain three clusters $\{1, \sqrt{2}\}, \{2\}, \{\sqrt{8}\}$. Next minimum distance is computed as $2 - \sqrt{2} \cong 0.59$, thus the first two clusters are merged and we obtain the two final classes: $\{1, \sqrt{2}, 2\}$ and $\{\sqrt{8}\}$. Obviously, $\sqrt{8} \geq 2$, which means the edges having the length 1, $\sqrt{2}$ or 2 represent *small* distances, marked in the figure by blue color, while the edges having the length $\sqrt{8}$ represent the *large* distances, marked in red in the same figure.

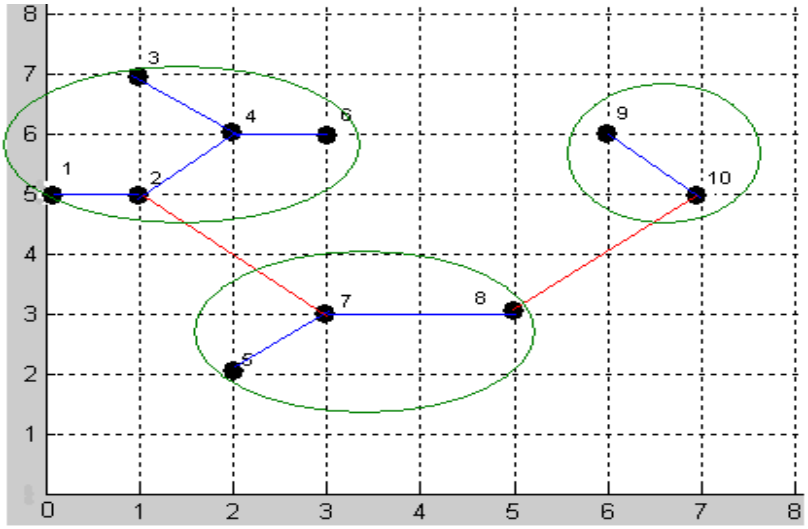

Fig. 2. Feature vector unsupervised classification results

Therefore, the following three clusters of points are obtained: {1,2,3,4,6}, {5,7,8} and respectively {9,10}. They are evidenced by the green colored circles in Fig.2. These classes of points represent the classes of feature vectors, which correspond to the classes of objects. Thus, the final recognition result over S consists of the following classes: $\{Ob_1, Ob_2, Ob_3, Ob_4, Ob_6\}$, $\{Ob_5, Ob_7, Ob_8\}$, $\{Ob_9, Ob_{10}\}$. Representing this result in terms of object similarity, we have $Ob_1 \approx Ob_2 \approx Ob_3 \approx Ob_4 \approx Ob_6$, $Ob_5 \approx Ob_7 \approx Ob_8$ and $Ob_9 \approx Ob_{10}$, respectively.

There are many application areas of this proposed automatic classification technology. One important application domain is multimedia analysis. Unsupervised region-growing based classification could be successfully used in media object recognition ([6]-[9]).

In our previous works we utilized the semiautomatic clustering approach for classification of images, image objects ([6]), audio sequences ([7],[8]), videoclips or videoshots ([9]). In these media pattern recognition cases the number of classes is set interactively. The users visualize or listen the media entities (having the same media type) from a given set, thus becoming able to set the number of their similarity classes. If the set contains a very large number of media objects, then the human interaction in the recognition process becomes useless. Therefore, there is need for an automatic clustering process.

Classification in large media sets is often used by important processes, such as multimedia indexing and retrieval ([2]). Thus, multimedia databases may contain many tables, each table corresponding to a media type and registering a great number of media objects. An indexing structure is required to facilitate the retrieval of these registered entities. Any *relevance-based* retrieval process needs a cluster-based index at each table level, because it uses object *similarity* instead of object *identity*.

An information retrieval system based on object relevance receives an input object and try to extract all the recordings from a database [5], which correspond to the entities similar to the input. To detect all the relevant objects from a table of the information base [5], the retrieval system must utilize an index structure composed of several classes. Therefore, all feature vectors of the objects stored in that table must be classified first, using the automatic clustering method. The input object has to be associated with the cluster whose center of gravity is the closest to its feature vector. The objects of the detected class are the ones what are relevant to the input entity.

For example, we may consider a large multimedia database containing distinct tables for the graphical, audio and video entities. The fields of each table must correspond to the main features of that media type. Such a table usually stores hundreds or thousands images, sounds, or other media. All these media entities cannot be properly displayed to the user, so their number of clusters cannot be set interactively. The automatic unsupervised classification procedure provided by us produces satisfactory results in this case. The

metric d used by our clustering algorithm depends here on the feature vector form. Euclidean distance is often used, mainly in the image classification. Some nonlinear metrics, such as the Hausdorff–based distance introduced in our previous works ([7]-[9]), can be utilized in the classification process of the vectors in the same feature set but having different forms (for example matrices having distinct dimensions), like some sound feature vectors. Next, if the input media object represents an image, its feature vector must be compared with the centroides of the clusters from the index of the table of images, if we want to retrieve the information data of all the stored images similar with it. If the input is a sound or a clip, the relevant objects will be retrieved the same way from the audio or video tables.

9. CONCLUSIONS

In this paper, we presented two unsupervised recognition technologies: a semiautomatic and an automatic one. The proposed unsupervised classification automatic algorithm represents the main contribution of this work. Some practical experiments, whose results prove the effectiveness of our procedure, are also described.

This classification technique is however restricted by some conditions. Our method does not handle properly the degenerated classification cases. The first situation is when any two objects from set S are quite different each other. In this case each class must contain a single object, while the clustering procedure provided here creates classes having at least two elements.

So, some modifications should be made to the clustering algorithm described in chapter 3. Thus, at each step the computed minimum distance has to be compared to a given threshold. If $d_{\min} \geq T$, then the points corresponding to that distance cannot be merged. The only problem here would be how the threshold T should be chosen, only.

The second degenerated case is the situation when all the objects to be recognized are very similar each other. That means only one class, containing all entities, must be obtained. First part of our clustering algorithm produces a single large cluster, therefore the second part should not be performed in this case. Introducing a threshold can be a solution here, too. Thus, after clustering all points (feature vectors) in a single class, the overall maximum of the previously computed minimum distances must be compared with that threshold. If condition $d_{\max} \leq T$ is satisfied, then the distance classification should not be performed.

As we mentioned in the previous chapter, our automatic clustering method could be applied in database indexing and retrieval domains. Multimedia indexing represents the main application area of the proposed technique. Classification in large media sets or sound, image and video, and database indexing using automatic clustering represent tasks to be approached in our future works.

REFERENCES

1. DUDA, R., HART, P., STORK, D., G., *Pattern Classification*, John Wiley & Sons, 2000.
2. JAIN, A.,K., MURTY, M. N., FLYNN, P. J., *Data Clustering: a review*, ACM Computing Surveys (CSUR), 31(3), 264-323, 1999.
3. KURITA, T., *An efficient agglomerative clustering algorithm for region growing*, PATREC: Pattern Recognition, Pergamon Press, 1991.
4. ROMESBURG, H. C., *Cluster Analysis for Researchers*, Reprint of 1990 edition published by Krieger Pub. Co., 2004.
5. ZAIANE, O., *CMPUT 690: Principles of Knowledge Discovery in Databases*, Online Courses at www.cs.ualberta.ca/%7Ezaiane/courses/cmput690/materials.shtml , 1999.
6. BARBU, T., *A Pattern Recognition Approach to Image Segmentation*, Proceedings of the Romanian Academy, Series A, Volume 4, Number 2, May-August 2003, pp.143-148.
7. BARBU, T., *Discrete Speech Recognition Using a Hausdorff Based Metric*, Proceedings of the 1st International Conference of E-Business and Telecommunication Networks, ICETE 2004, Setubal, Portugal, Aug. 2004, Vol. 3, pp.363-368.
8. BARBU, T., COSTIN, M., *Comparing Various Automatic Speaker Recognition Approaches*, Proceedings of Symposium of Electronics and Telecommunications, ETC 2004, Sixth Edition, Oct. 2004.
9. BARBU, T., *Content-based Video Recognition Technique using a Nonlinear Metric*, Proceedings of the 47th International Symposium ELMAR-2005 focused on Multimedia Systems and Applications, June 2005.

Received October 29, 2005