A Comparative Study of Hierarchical Matching Algorithms for Face Recognition

Hajar Momeni
h_momeni@stu.yazduni.ac.ir

Mohammad T Sadeghi
m.sadeghi@yazduni.ac.ir

Hamid R Abutalebi
habutalebi@yazduni.ac.ir

Signal Processing Research Laboratory, Department of Electrical and Computer Engineering,
Yazd University, Yazd, Iran

Abstract

The similarity measure based nearest neighbour classifier is commonly used in object recognition or retrieval systems. The result of a query in such a system is in general the images from the database whose descriptors are closest to the descriptor of the query image. An important issue in large scale recognition systems is the computational burden caused by measuring the similarity of the features derived from the query image with the templates of the database objects. Hierarchical Agglomerative Clustering (HAC) algorithms are considered as a solution to this problem. In these algorithms, a dendrogram is formed down to top in the training stage. In the recognition phase, the query image moves on the dendrogram from the highest level to lower levels in order to find the best matched object. One of the most affecting issues in these algorithms is the strategy used for building the dendrogram. In this paper, different techniques adopted for this purpose are studied and compared within the framework of a face recognition system. Our experimental results demonstrate that using an appropriate merging technique, the average recognition time reduces while the performance of the system is not highly degraded.

1 Introduction

Content-Based Visual Information Retrieval (CBVR) or Content-Based Image Retrieval (CBIR) has been among the most vivid research areas in the computer vision community. In recent years, the volume of digital images has increased dramatically. As a result, a crisis is now taking place within a broad range of disciplines that need and use visual materials. Whilst the current storage and capture technologies are able to cope with the huge numbers of images, the image retrieval systems are in danger of poor quality because of the difficulty of access. It is the job of an image retrieval system to produce images that a user wants. In response to a user's query the system must offer images that are similar in some user-defined senses [1] [2].

The similarity measure based nearest neighbour classifier is commonly used in object recognition or retrieval systems where only a few training samples are usually available. The result of a query in such a system is in general the images from the database whose descriptors are the closest to the descriptor of the query image. An important issue in large scale recognition systems is the computational burden caused by measuring the similarity of the features derived from the query image with the templates of the database objects.

There are three general algorithmic techniques for reducing the computational load in nearest-neighbour searches: computing partial distances, pre-structuring and editing the stored prototypes [3]. In partial distance method, a subset of features, say r out of d dimensions, are first used for calculating the distance. If this partial distance is too great, the computation is not further performed.

In pre-structuring, we create some form of search tree in which prototypes are selectively linked. During classification, we compute the distance of the test point to one or a few stored "entry" or "root" prototypes. The associated object prototypes are then considered and the closest one to the test point is found. If the tree is properly structured, we will reduce the total number of prototypes that need to be searched. In this method, we are no longer guaranteed to find the closest prototypes. A careful trade-off between the search complexity and the accuracy of the system is an important issue here. Sophisticated search trees such as k-d tree, SS tree, SR tree have been proposed for this purpose [4] [5].

The third method of reducing the complexity of nearest-neighbour search is to eliminate "useless" prototypes during training, a technique known as editing. Two important editing algorithms are pruning and condensing [3]. A simple method to reduce the space complexity is to eliminate prototypes which are surrounded by training points of the same category. We can also combine these three complexity reduction methods.

Applying a hierarchical structure in process of prototypes editing is an active research area. A dendrogram is constructed using Hierarchical Agglomerative Clustering (HAC) algorithm on the prototypes of the database. A few representatives less than the database prototypes are then chosen by cutting the dendrogram from a predefined level. These representatives play the roll of the prototypes so that in classification stage, a test pattern is compared with these representatives instead of the original prototypes. In [6], an automated HAC based system has been proposed which automatically organises a collection of music files according to musical surface characteristics.

In [7], a system has been designed based on the HAC algorithm in order to automatically extract representative face samples, namely exemplars from a set of

images for each person in the associated database for multishot face recognition. Experimental results confirm the effectiveness and flexibility of their method. Jain et.al also have proposed a HAC based technique for automatically selecting the prototypes related to fingerprint templates from a given set of fingerprint impressions in a biometric authentication application [8].
In this work, the Hierarchical Agglomerative Clustering (HAC) algorithm is considered as a solution to the problem of computational complexity of the nearest-neighbour classifier. We have entered the hierarchical structure in the process of prestructuring of the prototypes by building a dendrogram instead of a search tree. In the proposed method, a dendrogram is formed in the training stage considering all the prototypes from down where the original prototypes are available to top where the final representatives are present. A representative is chosen for each node of the dendrogram. Then, in the recognition phase, the query image moves on the dendrogram from the highest level towards the lower levels and is compared with the representatives of related clusters in order to find the best matched object.
If the dendrogram is properly structured, we are waitfullness to reduce the total number of comparisons which have to be performed. As mentioned earlier, a good trade-off between the complexity of the system and its performance is crucial. In this paper, different techniques adopted for building the dendrogram are studied and compared within the framework of a face recognition system. Our experimental results demonstrate that by using an appropriate dendrogram, the average recognition time reduces by a factor of about 0.5 while the performance of the system is not highly degraded. However, dendrograms which highly reduce the average recognition time, have a poor performance.

This paper is organised as follows. In section 2, the Hierarchical Clustering Analysis is briefly reviewed. The adopted method of constructing a dendrogram is also described. A thresholding technique is then proposed in section 3 in order to speed up the recognition process. The experimental results are presented in section 4, followed by conclusions in section 5.

2 Hierarchical Clustering Analysis

The techniques proposed in this work are based on Hierarchical Clustering Analysis(HCA). HCA is a method for exploring the underlying structure of a set of objects through an iterative process which associates (agglomerative methods(HAC)) or dissociates (divisive methods) objects together. The iterative scheme is halted when all the objects have been processed.

The similarity measure which is used for merging the objects and/or groups of objects and the linkage technique are two important choices in the HCA based algorithms. In this study, Euclidean distance was used as the similarity measure. Some of the most used linkage algorithms are Complete (or furthest-neighbour), Single (or nearest-neighbour), Average, Centroid and Ward's techniques [3].

The structure obtained by hierarchical clustering is often presented in the form of a dendrogram where each linkage step in the clustering process is represented by a connection line. A comparison study of merits and flaws of the above mentioned linkage techniques can be found in [9].

We adopted the use of the HAC algorithm for constructing the search dendrogram in the training stage. An appropriate representative is chosen for every level of the dendrogram which is further used in the recognition phase. One of the most affecting factors in this algorithm is the strategy used for building the dendrogram. In the following sections, different techniques adopted for this purpose are studied. These algorithms are compared together from the computational point of view. In fact, if the nearest neighbour rule is normally carried out, the total number of required comparisons, in the recognition phase, would be equal to the number of the original prototypes, n. It is important to find out how much the average number of comparisons is reduced using each adopted technique.

2.1 Centroid dendrogram ('Cent')

The centroid dendrogram is built using the HAC algorithm along with the centroid linkage rule. When a new group is formed, the Euclidean distance between the mean of the objects within the group and mean of the objects of any of the other available groups is considered as the distance between these two groups. Since the mean of the objects within the new formed group is considered as it's representative, the centroid linkage is meaningful. The centroid linkage can produce a non-monotonic dendrogram such as the one shown in figure 1. However, it does not affect our recognition procedure.
Applying this method to different data leads to dendrograms with different structures. From the computational burden point of view in the worst case, a special form of dendrogram such as the one shown in figure 2 is produced. Considering this plot, in such a case, the average number of required comparisons in the recognition phase can be calculated as follows: if the search results are objects 1,2,... (as figure shows) then 2×1, 2×2,... comparisons are respectively needed. Considering n as the number of objects, this form of dendrogram, always involves $n-1$ levels. So, if the search result is the most left point of the dendrogram (object 6 or 7 in figure 2), $2 \times (n-1)$ comparisons have to be performed. Therefore, the average number of required comparisons is:

$$\frac{(2 \times 1) + (2 \times 2) + (2 \times 3) + ... + (2 \times (n-1))}{n-1} =$$
$$\frac{2 \times (1 + 2 + ... + n - 1)}{n-1} = \frac{2 \times (n \times (n-1)/2)}{n-1} = n \quad (1)$$

As one can see, in such a case, the computational complexity is not reduced compare to the basis nearest neighbour search. In order to avoid generation of such a dendrogram, the most similar objects can be coupled together two by two in each level.

2.2 Binary dendrogram ('Bin')

In a Binary dendrogram, the most similar two objects are first grouped. Then, considering the remaining object, the most similar ones are grouped. This process is continued until the most similar objects are coupled together two by two. The mean value of the

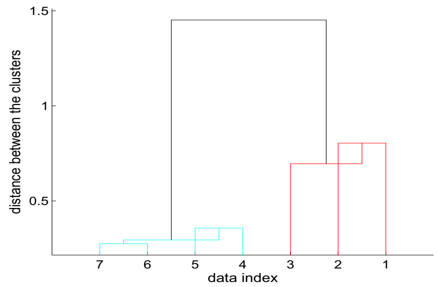

Figure 1: Non monotonic dendrogram

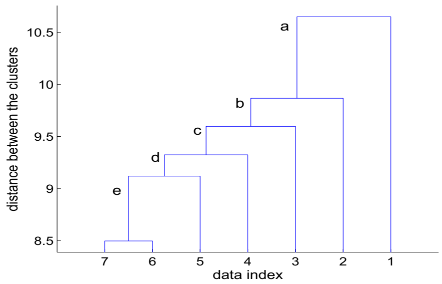

Figure 2: Computationally no efficient dendrogram

data in each group is computed as the group representative. This process is repeated until the highest level cluster is formed.

In the recognition phase, a query image goes through $log_2 n$ levels of the dendrogram where n is the total number of objects. So, the average number of required comparisons is equal to $2 \times log_2 n$. Therefore, the computational burden is highly reduced compare to the nearest neighbour method specially when n is large.

This approach can be generalised by grouping the objects in each level three by three and so on. For example using a Trinary dendrogram ('Tri'), the average number of required comparison is $3 \times log_3 n$ which is slightly less than $2 \times log_2 n$ associated to the Binary one.

3 Thresholding in Recognition Phase

As we mentioned, using a centroid dendrogram, the computational complexity is not reduced if the bad case of plot 2 happens. The search process can be speed up if an appropriate threshold is assigned to each node of the dendrogram. Considering the threshold associated to each node, distance between the query image and the object in the right hand side of the node (see figure 2) is calculated and compared with the threshold. If the distance is less than the threshold the query is assigned to the object. Otherwise, the process is continued considering the left hand side node.

In this work, the Threshold value has been determined in the training process considering evaluation data. This is performed using two different techniques: Global Thresholding (GT) and Level Specific Thresholding (LST).

In the GT approach, a same value is considered as the threshold for all nodes. In the evaluation stage, the optimum threshold is determined by choosing the value which minimises the recognition error. In the LST technique, a specific threshold is assumed for each node. The initial value of the threshold is determined considering the distance between the two elements of the node. A constant value is then added to the nodes threshold and the associated evaluation error rate is determined. The constant value which minimises the error rate is finally chosen and added to the initial values for determining the final thresholds.

The average number of comparisons is halved by applying the thresholding technique($\frac{n}{2}$ instead of n).

4 Experimental Results

In this section, results of applying different method of constructing of dendrogram are presented within the framework of a face recognition system. These results are compared with the results obtained from the basic nearest neighbour classifier.

4.1 The XM2VTS database

In this work, we use XM2VTS database [1]. The XM2VTS database is a multi-modal database consisting of face images, video sequences and speech recordings taken of 295 subjects at one month intervals. Since the data acquiring was distributed over a long period of time, significant variability of appearance of subjects, e.g. change of hair style, facial hair, shape and presence or absence of glasses, is presented in the recordings. The XM2VTS image database contains eight images for each subject from 4 recording sessions.

4.2 Experimental setup and results

By applying the cross-validation method, 50 percent of the image data (4 images for each subject) are randomly taken as the training data. The remaining data is then randomly partitioned into two parts, one part consists of 2 images for each subject assumed to be in the evaluation set and the other 2 in the test set.

The training set is used for constructing the prototype models. It is performed by averaging of the images of a person. The evaluation set is selected to define the threshold discussed in section 3. The threshold is set either globally (GT) or using the level specific thresholding (LST) technique. The test set is also selected to simulate realistic recognition tests where test's identity is unknown to the system.

The original resolution of the image data is 720×576. The experiments were performed with a relatively low resolution face images, namely 64×48. The result reported in this article have been obtained by applying a geometric face registration based on manually annotated eyes positions. Histogram equalisation was used to normalise the registered face photometrically. Although, the face images have been taken by colour camera, but we use the associated gray level images.

In the face identification process, the face image data is projected into the fisherface space by applying the Linear Discriminant Analysis (LDA) technique. So, the number of dimensions are reduced from 3072 to 294. Euclidean distance has been also used as the distance metric.

In addition to the basic nearest-neighbour classifier (1-NN), the above mentioned hierarchical structuring approaches, i.e. 'cent', 'bin', 'tri' and 'cent' with thresholding have been implemented. The recognition performance was evaluated considering the problem of person identification (PID).

The PID aspect of face recognition is connected with the possible application of the system to verifying or

[1]http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb/

Table 1: Identification results(Average Success Rate%)

	1-NN	Cent	Bin	Tri	GT	LST
Mean	84.6	79.1	53.5	56.7	57.9	66.3
Std	1.88	2.28	3.07	1.33	0.99	1.11
A.N.C	295	363	17.4	17.1	213	190
S.F	1	0.81	16.9	17.26	1.38	1.56

authenticating person's identity based on their facial images. All images in the database have to be labeled and in response to the query, the identity of the corresponding person is to be returned. The intuitive measure evaluating the performance of the recognition system from the PID point of view is the Average Success Rate (ASR) defined by following equation.

$$ASR = \frac{1}{P} \times \sum_{p=1}^{P} SR(p), \qquad ASR \in [0,1] \tag{2}$$

where $SR(p)$ is the person success rate calculated as:

$$SR(p) = \frac{NS(p)}{NQ(p)} \tag{3}$$

where $NQ(p)$ is the number of query images for person with identity of p and $NS(p)$ is the number of successful queries for this person. By using cross-validation technique, the experiments were repeated 5 times. Their mean and standard deviation of the ASR values were reported in the first tow lines of table 1(per percent). Also, the Average Number of Comparisons ($A.N.C$) which has been computed experimentally and the Speed-up Factor ($S.F$) defined by equation 4 have been shown in the table. The $S.F$ value declares how much the adopted system performs faster than the basis 1-NN system.

$$S.F = \frac{A.N.C\ of\ 1-NN\ system}{A.N.C\ of\ proposed\ system} \tag{4}$$

As one can see, using the centroid dendrogram, the performance of the system is slightly reduced compare to the basic NN classifier. However, as discussed earlier, the method is not computationally efficient compare to the other HAC algorithms. It can be seen that in this case the $A.N.C$ value is even worst that the expected average value. Although, the 'Bin' and 'Tri' dendrograms are computationally much more efficient, they are not very reliable. The performance of the system using the Centroid dendrogram along with the Level Specific Thresholding technique is not as good as the Centroid method, but it is computationally more efficient.

These experiments were repeated considering the FERET database [2] and a combination of the XM2VTS and FERET databases. The experimental results associated to the FERET data were almost similar to the XM2VTS one. Based on our experiments, the adopted Centroid dendrogram performs better on the larger scale recognition system, i.e. the XM2VTS + FERET data. In this case, the designed system was 4 times faster than the 1-NN system with only 3 percent reduction in the performance compare to the 1-NN system.

The proposed algorithms can be applied for other recognition systems as well. This problem is a matter of interest in the future studies.

[2] http://face.nist.gov/colorferet/

5 Conclusions

An important issue in large scale recognition systems is the computational burden caused by measuring the similarity between a query image and the database objects. Hierarchical Agglomerative Clustering (HAC) algorithms are considered as a solution to this problem. One of the most affecting issues in these algorithms is the strategy used for building the associated dendrogram. In this paper, different techniques adopted for this purpose were studied and compared within the framework of a face recognition system. Our experimental results on the XM2VTS demonstrates that using the Centroid dendrogram along with the Level Specific Thresholding technique, the average recognition time reduces by a factor of about 0.5. The performance of the system is not highly degraded compare to the other time efficient algorithms. Our primary study on a large scale database confirms that the proposed Centroid dendrogram performs more efficiently in such a case.

Acknowledgments

The financial support from the ITRC, Iran Telecommunication Research Center, is gratefully acknowledged.

References

[1] G. Guo, A. K. Jain, W. Ma, H. Zhang: "Learning similarity measure for natural image retrieval with relevance feedback," *IEEE Transactions on Neural Networks*, vol.13, pp.811-820, 2002.

[2] F. W. M. Stentiford: "An attention based similarity measure with application to content based information retrieval", *In Proceedings of the Storage and Retrieval for Media Databases conference, SPIE Electronic Imaging*, 2003.

[3] R. O. Duda, P. E. Hart, D. G. Stork: *Pattern Classification*, John Wiley, 2001.

[4] C. Bohm, S. Berchtold, D. A. Keim: "Searching in High Dimensional Spaces: Index Structures for Improving the Performance of Multimedia Databases," *ACM Computing Surveys* , vol.33, no.3, pp.322-373, 2001.

[5] N. Katayama, S. Satoh: "The SR Tree: An Index Structure for High Dimensional Nearest Neighbor Queries", *In Proceedings of ACM SIGMOD Int'l Conf. Multimedia and Design*, pp.369-380, May 1997.

[6] A. S. Lampropoulos, G. A. Tsihrintzis: "Agglomerative Hierarchical Clustering For Musical Database Visualization and Browsing", *3rd Hellenic Conference on Artificial Intelligence*, Samos, Greece, May 2004.

[7] W. Fan, D-Y Yeung: "Face Recognition with Image Sets Using Hierarchically Extracted Exemplars from Appearance Manifolds", *Proceedings of the Seventh IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, Southampton, UK, pp.177-182, April 2006.

[8] A. Jain, U. Uludag, A. Ross: "Biometric Template Selection: A Case Study in Fingerprints", *In Proc. of 4th Int'l Conference on Audio- and Video-Based Person Authentication (AVBPA)*, Guildford, UK, pp.335-342, June 2003.

[9] J. A. S. Almeida, L. M. S. Barbosa, A. A. C. C. Pais, S. J. Formosinho: "Improving hierarchical cluster analysis: A new method with outlier reduction and automatic clustering", *Chemometrics and Intelligent Laboratory Systems 87*, pp.208-217, 2007.

[10] W. Skarbek, K. Kucharski, M. Bober: "Dual LDA for Face Recognition," *Visual Information Laboratory, Mitsubishi Electric Fundamenta Informaticae XXI*, vol.61, no.3-4, pp.303-334, IOS Press, 2003.