HIERARCHICAL FACE CLUSTERING USING SIFT IMAGE FEATURES

Panagiotis Antonopoulos, Nikos Nikolaidis and Ioannis Pitas

Dept. of Informatics, Aristotle University of Thessaloniki, Box 451, 54124 Thessaloniki, Greece
e-mail: {pantonop, nikolaid, pitas}@aiia.csd.auth.gr

ABSTRACT

In this paper an algorithm to cluster face images found in video sequences is proposed. A novel method for creating a dissimilarity matrix using SIFT image features is introduced. This dissimilarity matrix is used as an input in a hierarchical average linkage clustering algorithm, which yields the clustering result. Three well known clustering validity measures are provided to asses the quality of the resulting clustering, namely the F measure, the overall entropy (OE) and the Γ statistic. The final result is found to be quite robust to significant scale, pose and illumination variations, encountered in facial images.

1. INTRODUCTION

Clustering could be considered as a form of unsupervised classification imposed over a finite set of objects. Its goal is to group sets of objects into classes, such that similar objects are placed in the same cluster, while dissimilar objects are placed in different clusters.

Human faces are some of the most important and frequently encountered entities in videos and can be considered as high-level semantic features. Face clustering in videos can be used in many applications such as video indexing and content analysis [1], as a preprocessing step for face recognition [2], or even as a basic step for extracting the principal cast of a feature length movie, as described in [3] and [4]. Furthermore, face clustering is of great importance when it comes to video based facial expression recognition applications, that deal with more than one persons. Such systems, that can be applied in virtual reality, human centered interfaces or user profiling, should be able to detect the presence of each person in the image sequence, cluster the face images and afterwards perform facial expression recognition for each person separately.

A limited number of face clustering algorithms have been reported in the literature. Fitzgibbon and Zisserman [4] have proposed an approach for face clustering in video that involves the so called Joint Manifold Distance (JMD). In the proposed method, each subspace represents a set of faces of the same person detected in contiguous frames. The clustering uses the sequence to sequence distance and follows an agglomerative strategy. In [3] the same authors proposed another distance metric for clustering and classification algorithms, called Affine Invariant Distance Measure (AIDM). This distance function, which is invariant to affine transformations, is used in combination with partitioning based algorithms, for face clustering. On the other hand, Eickeler et al. [1] have proposed a face clustering method, called Hidden Markov Models-clustering (HMM-clustering), which is a K-means clustering, that uses Hidden Markov Models to represent a cluster prototype. Finally, Czirjek, et al. [2] have proposed a semi-supervised method for automatically detecting and clustering human faces in generic video sequences. The described clustering approach differs from classical, totally unsupervised clustering approaches, in the sense that it makes use of a number of pre-existing face clusters each corresponding to a specific newscaster commonly observed in the news programs. Then the method assigns each face sequence from the test set to a pre-existing cluster or starts a new one.

In this paper a new technique for clustering human faces, detected in videos that contain faces, is proposed. The clustering is achieved by using a dissimilarity matrix, constructed with the aid of SIFT image features [6], [7], that is fed into a hierarchical average linkage clustering algorithm. The method is tested on feature length video sequences, providing very encouraging results.

The rest of this paper is organized as follows. Section 2 presents the proposed face clustering method. In more detail, subsection 2-A describes the hierarchical agglomerative clustering, whereas subsection 2-B describes a novel method for computing a dissimilarity matrix using SIFT descriptors. Additionally, in section 3 experimental results of the clustering algorithm, using three clustering validity measures, are given. Finally, in section 4 conclusions are drawn.

2. METHOD DESCRIPTION

The data, that we wish to cluster consist of face images obtained by a face detection algorithm. The faces are detected using the Boosted Cascade method, described in [5]. This method uses the Adaboost algorithm to select and combine a set of appropriate features that resemble Haar basis functions in image areas, so as to train efficient classifiers. A combination of successively more complex such classifiers in a cascade allows the early rejection of non-face regions,

thus allowing for more computation to be spent on more promising areas. The face images, generated by the face detection algorithm, are unlabel , thus an unsupervised clustering procedures should be applied.

The proposed method consists of two parts: First a dissimilarity matrix is created using SIFT image features, derived from the images generated by the face detector and then, a hierarchical average linkage face clustering algorithm is applied on the aforementioned dissimilarity matrix. The following sections describe, in more detail, each part of the method.

2-A. Hierarchical Clustering

A hierarchical clustering method is a procedure that transforms a dissimilarity matrix into a sequence of nested partitions [8]. A dissimilarity matrix **D** is a square and symmetric matrix that contains all the pairwise dissimilarities between the samples, that should be clustered. If the n objects to be clustered are defined by the set O:

$$O = \{o_1, o_2, ..., o_n\} \tag{1}$$

the elements of **D** are defined as $D_{ij} = dissimilarity(o_i, o_j)$, with $i, j = 1 \dots n$. Obviously, $D_{ii} = 0$ and $D_{ij} = D_{ji}$.

Hierarchical clustering methods exhibit a deterministic nature, in the sense that they produce always the same output, regardless of their initialization. A partition P of the n objects splits the set O into subsets $\{S_1, S_2, ..., S_m\}$ that satisfy the following rule [8]:

$$\begin{aligned} S_i \cap S_j &= \oslash \text{ , } for\ i, j \in [1, m],\ i \neq j \\ S_1 \cup S_2 \cup ... \cup S_m &= O \end{aligned} \tag{2}$$

A partition P_1 is nested into partition P_2 if every component of P_1 is a subset of a component of P_2. In this way, a partition can be formed by merging its nested partitions.

An agglomerative, or bottom-up, hierarchical clustering algorithm [8], has been used in our case. In such algorithms the procedure starts with n singleton clusters (each of the n objects are placed in individual clusters) and a sequence of partitions is formed by successively merging clusters. The notion of distance among clusters plays a major role in the merging of clusters. The most frequently used inter-cluster merging techniques are single linkage (clusters are merged based on the shortest distance between objects in the two clusters), complete linkage (merging is based on the largest distance between objects) and average linkage (based on the average distance between objects). The average distance D_{RQ} between two clusters, is defined as the mean value of all distances among each object in cluster R and each object in cluster Q [9]:

$$D_{RQ} = \frac{\sum_{i \in R} \sum_{j \in Q} D_{ij}}{|R| \cdot |Q|} \tag{3}$$

where $|\cdot|$ denotes the cluster cardinality.

Our algorithm utilizes the average linkage merging approach, because it takes into account information from all objects (faces) in a cluster.

2-B. Computing The Dissimilarity Matrix Using SIFT Image Features

The scale invariant feature transform (SIFT) algorithm is a method for extracting highly distinctive invariant features from images, that can be used to perform reliable matching between different views of an object or a scene [7]. In our case the SIFT features were used for matching the face images and creating the dissimilarity matrix used in agglomerative clustering algorithm, described in the previous section.

SIFT algorithm, [6], [7], has four major stages:

- *scale-space exrema detection*
- *keypoint localization*
- *orientation assignment*
- *keypoint descriptor*

SIFT evaluates characteristic keypoints on an image and constructs a canonical view for each keypoint, which is invariant to significant levels of local shape distortion, scale, camera viewpoint and illumination changes. Each keypoint is assigned a 128 element vector, that expresses the orientation, scale and location of a region of pixels around the keypoint. This makes it a very useful tool for fast matching of a large quantity of face images.

In order to construct the dissimilarity matrix **D** of size $N \times N$, where N is the total number of face images we wish to cluster, the following procedure is used for evaluating the dissimilarity between facial images A_i, A_j i.e. the element D_{ij} of the matrix.

First, SIFT keypoints, along with their corresponding feature vectors, are extracted from images A_i and A_j. Then, matching is accomplished by finding candidate matching keypoints based on the Euclidean distance of their feature vectors, as proposed in [7]. A match between two keypoints in A_i and A_j is accepted only if the distance of their feature vectors is less than threshold distRatio (defined in [7]) times the distance to the second closest match. The result is a number of keypoint matches found for this pair of face images.

Due to the fact that matching A_i against A_j doesn't produce the same matching result to matching A_j against A_i, whereas the dissimilarity matrix should be symmetric, we perform the matching twice; once for the pair (A_i, A_j) and once for the pair (A_j, A_i). The maximum number of keypoint matches found in the two matches is the final matching result for the specific pair of images. Finally, the above number of keypoint matches is transformed to a dissimilarity ratio (DR_{ij}) between the two compared images using the formula:

$$DR_{ji} = DR_{ij} = 100(1 - \frac{M_{ij}}{min(K_i, K_j)}) \tag{4}$$

where M_{ij} is the maximum number of keypoint matches found between the pairs (A_i, A_j), (A_j, A_i) and K_i, K_j are the numbers of keypoints found in A_i, A_j respectively. $DR_{ij} \in [0, 100]$ and high DR_{ij} values indicate large dissimilarity between face images. DR_{ij} is considered as the element D_{ij} of the dissimilarity matrix, constructed for the N facial images. Figure 1 shows the dissimilarity matrix for the test set used in our experiments. In this point, it should be noted that performing the matching twice for the same pair, does not increases significantly the calculation time, since the time consuming calculation of the SIFT image features is done only once.

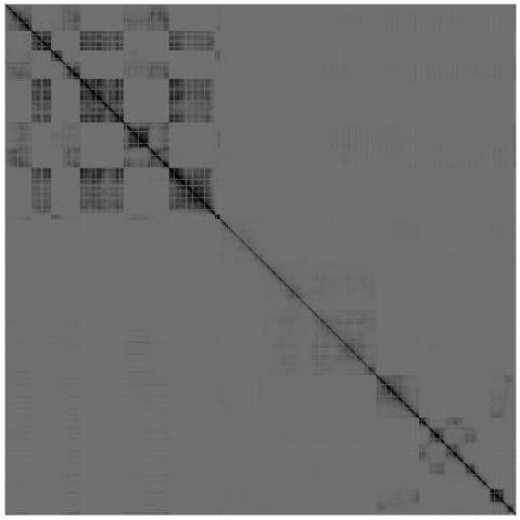

Fig. 1. Dissimilarity matrix created using SIFT image features for 941 face images extracted from a part of the feature length film "Two weeks notice".

3. EXPERIMENTAL RESULTS

Three measures are employed in order to asses the performance of the clustering in an objective fashion, namely the F measure, the overall entropy and the Γ statistic. Let

- N be the total number of patterns (face images);
- N_f be the number of classes, according to the ground truth;
- N_c be the total number of clusters created by the clustering algorithm;
- n_{ij} be the number of patterns from class j in cluster i;
- n_i be the number of patterns that belong to cluster i;
- n_j^c be the number of patterns that belong to class j;
- M_C be the number of combinations of two patterns, that can be derived from the input data set;

The validity measures mentioned above, are described as follows:

3-A. F measure

The F-measure for a cluster i and class j is [11]:

$$F_{ij} = \frac{2 \cdot \frac{n_{ij}}{n_j^c} \cdot \frac{n_{ij}}{n_i}}{\frac{n_{ij}}{n_j^c} + \frac{n_{ij}}{n_i}} \tag{5}$$

whereas for each class over the entire hierarchy is:

$$F_j = \max_i F_{ij} \tag{6}$$

The maximum refers to all clusters at all levels. Finally, the F measure for the whole clustering hierarchy is defined by combining (5), (6) in:

$$F = \sum_j \frac{n_j^c}{N} \cdot F_j \tag{7}$$

Its major advantages are that the average is evaluated over the classes rather than the clusters, it compares an entire hierarchy with a flat partition and it tries to capture how well the clusters of the investigated partition match those of the ground truth. F measure values range between 0 and 1, with 1 indicating perfect clustering.

3-B. Overall entropy (OE)

The OE is defined in [12] as:

$$OE = \beta \cdot E_c + (1 - \beta) \cdot E_l \tag{8}$$

where E_c is the *overall cluster entropy* and E_l is the *overall class entropy* and $\beta \in [0, 1]$ functions as a weight parameter that balances those two entropies. More specifically, the *overall cluster entropy* E_c is given by the weighted sum of the individual cluster entropies:

$$E_c = \frac{1}{N} \cdot \sum_{i=1}^{N_c} n_i \cdot E_{c_i} \tag{9}$$

where for each cluster, c_i, the individual cluster entropy E_{c_i} is:

$$E_{c_i} = -\sum_{j=1}^{N_f} \frac{n_{ij}}{n_i} \log \frac{n_{ij}}{n_i} \tag{10}$$

The *overall class entropy* E_l is computed as the weighted sum of the individual cluster entropies, using an analogous procedure.

The cluster entropy reflects the homogeneity of the patterns in a cluster and its values range between 0 and 1, with low values indicating high homogeneity. OE makes use of the advantages of both E_c and E_l with β being a balancing factor between them. In our experiments $\beta = 0.5$ was chosen in (8).

3-C. Γ statistic

The Γ statistic is a correlation coefficient, which follows the idea of partitional structure validity and is a special case of Hubert's Γ statistic [8]. It expresses the correlation between the clustering produced by the clustering algorithm and the perfect clustering (i.e. the classes) given by the ground truth. Γ statistic is defined by the following formula:

$$\Gamma = \frac{(M_c \cdot a - m_1 \cdot m_2)}{[m_1 \cdot m_2 \cdot (M_c - m_1) \cdot (M_c - m_2)]^{\frac{1}{2}}} \qquad (11)$$

where,
$a = \frac{1}{2}\sum_{i=1}^{N_c}\sum_{j=1}^{N_f} n_{ij}^2 - (N/2)$;
$b = \frac{1}{2}\sum_{j=1}^{N_f}(n_j^c)^2 - \frac{1}{2}\sum_{i=1}^{N_c}\sum_{j=1}^{N_f} n_{ij}^2$;
$c = \frac{1}{2}\sum_{i=1}^{N_c} n_i^2 - \frac{1}{2}\sum_{i=1}^{N_c}\sum_{j=1}^{N_f} n_{ij}^2$;
$m_1 = a + b$, $m_2 = a + c$;

Its values ranges between -1 to 1, with high values indicating a good clustering result.

3-D. Experimental Results

The test corpus was a set of face images, obtained by applying the face detector described in section 2 on every fifth frame of a part of the feature length movie "Two Weeks Notice", consisting of approximately 40000 frames. This resulted in 941 face images of no specific size (on average they are of dimensions 100×100 pixels). No preprocessing has been performed on these images. The ground truth was extracted manually by inspecting these images and yielded three classes each containing a different face in different poses and illumination and a fourth class, that contained all the false face detections extracted by the face detector.

The clustering results for the 941 face images were very promising. The hierarchical average linkage clustering algorithm using the SIFT-based dissimilarity matrix for 4 clusters, yields results 0.8763 for F measure and 0.7977 for Γ statistic and an OE value of 0.094, which indicate well clustered data, in good agreement with the ground truth and forming homogeneous clusters. Examples of the results of the proposed face clustering algorithm are demonstrated in figure 2. Clusters 1, 2 and 4 contained only facial images from the same person. The third cluster contained the false face detections (non-facial images) as we expected, but it also included certain instances of the actor in cluster 1, due to a significant change in the person's pose.

It should be noted that, the hierarchical average linkage clustering algorithm is not designed to calculate automatically the natural grouping of the input data, i.e the number of clusters. Thus, the user has to provide the number of clusters himself. However, this is not considered a major drawback in most applications, i.e. when the number of actors in a film is known.

Fig. 2. Results of the proposed algorithm. The first 24 face images of every cluster are shown.

4. CONCLUSIONS AND FUTURE WORK

Clustering of face images in video sequences is a difficult task, since significant changes in lighting and viewpoint occur. This paper has introduced a novel method for creating a dissimilarity matrix for the face images using SIFT image features, which is fed to a hierarchical average linkage clustering algorithm. Experimental evidence for the clustering quality was provided by using the F measure, the OE and the Γ statistic as figures of merit. The assessment was conducted on a set of face images acquired by a part of a feature length film. The clustering results are very satisfactory.

In the future, we aim to use the proposed dissimilarity matrix as input in different clustering algorithms, not only hierarchical but partitional as well. Automatic calculation of the number of clusters sought, will be also pursued. Finally, the method will be tested on larger data sets.

ACKNOWLEDGMENT

This work has been conducted in conjunction with the SIMILAR European Network of Excellence on Multimodal Interfaces (http://www.similar.cc) of the IST Programme of the European Union. The SIFT image features calculation and matching, was achieved using the implementation provided by Lowe in [13] for research purposes.

5. REFERENCES

[1] S. Eickeler, F. Wallhoff, U. Iurgel, G. Rigoll. ”Content-based Indexing of Images and Videos using Face Detection and Recognition Methods”, IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP), Salt Lake City , Utah 2001.

[2] C. Czirjek, N. O’Connor,S. Marlow,N. Murphy. ”Face Detection and Clustering for Video Indexing Applications”, Advanced Concepts for Intelligent Vision Systems (Acivs), Ghent, Belgium, September 2003.

[3] A. W. Fitzgibbon and A. Zisserman. ”On affine invariant clustering and automatic cast listing in movies”. European Conference on Computer Vision (ECCV), vol. 3, pp. 304 - 320. Springer-Verlag, 2002.

[4] A. W. Fitzgibbon, A. Zisserman. ”Joint Manifold Distance: a new approach to appearance based clustering,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR ’03) - Vol. 1, pp. 26-36, 2003.

[5] P. Viola, M. Jones. “Robust real-time face detection,” International Journal of Computer Vision, vol. 57, pp. 137-154, May 2004.

[6] D. G. Lowe. ”Object recognition from local scale-invariant features”. In Proceedings of International Conference on Computer Vision, pp. 1150-1157, 1999.

[7] D. G. Lowe. ”Distinctive image features from scale-invariant keypoints,” International Journal of Computer Vision, vol. 60(2), pp. 91-110, June 2004.

[8] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Englewood Cliffs, N.J.: Prentice-Hall, 1988.

[9] R. R. Sokal, C. D. Michener. ”A statistical method for evaluating systematic relationships”. University of Kansas Science Bulletin, 38, pp. 1409-1438, 1958.

[10] K. Mikolajczyk and C. Schmid. ”A performance evaluation of local descriptors”. In Proceedings of Computer Vision and Pattern Recognition (CVPR), June 2003.

[11] B. Larsen and C. Aone. ”Fast and effective text mining using linear-time document clustering”. In Proc. 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1999.

[12] J.He, A. H. Tan. C. L. Tan., and S. Y. Sung, ”On quantitative evaluation of clustering systems”, in: W. Wu, H. Xiong, and S.Shekhar, eds., Clustering and Information Retrieval, Kluwer Academic Publishers, pp. 105-133, 2003.

[13] Online at: ”http://www.cs.ubc.ca/ lowe/keypoints/”.