4th Canadian Conference on Computer and Robot Vision, Montreal, 2007

Automatic Detection and Clustering of Actor Faces based on Spectral Clustering Techniques

S. Foucher, L. Gagnon
R-D Department, CRIM, 550 Sherbrooke West, Suite 100, Montréal, QC, CANADA, H3A 1B9
{samuel.foucher, langis.gagnon}@crim.ca

Abstract

We describe a video indexing system that aims at indexing large video files in relation to the presence of similar faces. The detection of near-frontal view faces is done with a cascade of weak classifier. Face tracking is done through a particle filter and generate trajectories. Face clusters are found based on a spectral clustering approach. We compare the performance of various spectral clustering techniques based on 2DPCA features. The system performance is evaluated against a public face database as well as on a real full-length feature movie.

1. Introduction

Face recognition is one of the best ways to directly extract high-level semantic information for video summarization, video indexing or automatic casting. While automatic face recognition aims at matching observed faces against a face database [8][9][12][17][21], automatic casting consists in forming groups of similar faces represented by key-faces (i.e. principal actors) [12]. A typical full-length movie contains a relatively small number (5-10) of principal actors but with many face instances and sometimes many faces of walk-on actors. Thus, the main challenge in automatic casting is to efficiently form clusters of potentially thousands of face images.

Here, we present a key-face clustering technique that can be used for automatic casting. Key-face identification is also important for the production of descriptive video. Descriptive video, also known as audiovision, is a narration added to the movie audio track that describes visual elements for the blind and seeing-impaired people. This industry is growing due to the imposition of regulations to increase broadcasting of programs with descriptive narration. The work we present here is part of a project targeting the development of software tools for computer-assisted descriptive video [25].

There have been very few works on automatic casting. In [17], faces in news videos are recognized and retrieved on the basis of visual and textual information from transcripts. In [20], a spectral clustering algorithm, called the Ng algorithm [5], is used to cluster a relatively small set of faces (39 faces for 6 peoples) described by their eigenface projection. In [21], faces are described by a complex 2D-HMM model and the resulting features are simply clustered using the k-means algorithm. In [7], an affine invariant affinity measure is proposed to describe each face followed by a hierarchical k-medoids clustering technique. In [6], frame-based face detections are combined with temporal tracking in order to extract faces exemplars which are then represented and compared on the basis of a SIFT signature.

One of the main step of our system consists in detecting and tracking near-frontal view faces in order to form trajectories. The tracking algorithm is a particle filter (bootstrap filter) for which the likelihood of each particle is based on the response of a frontal view detector. For each trajectory, we keep a set of representative faces which are then encoded using 2DPCA techniques. Finally, a clustering algorithm is applied in order to form clusters of similar faces. The choice of the optimal number of clusters (validation problem) is not critical for our application. Over-clustering of the dataset is not an issue as long as it produces several homogeneous clusters per actor faces. We compare the performances of different 2DPCA representations and spectral clustering techniques on the ORL databases. Preliminary performance results on a full-length feature movie are also presented.

2. System overview

The high-level system architecture is represented on Figure 1. The inputs are the video file and the shot boundary time positions. We automatically detect shot transitions (cut and smooth transitions) based on the mutual color information [22]. Within each shot, we automatically detect and track face candidates resulting in a set of trajectories. For each trajectory, a representative sample composed of the best observed frontal face views are stored. Finally, similar faces belonging to different trajectories are clustered using a spectral clustering technique. False-alarms are rejected. The clustering can be iterated for clusters that are difficult to identify. Audiovision for actor identification can be produced when the user identifies each cluster.

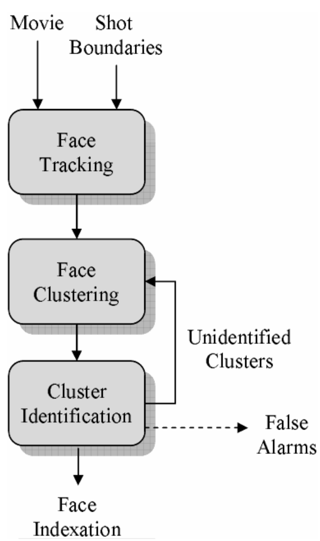

Figure 1. Overview of the proposed system

3. Key-faces extraction

3.1 Face detection

The near-frontal view face detector is based on a cascade of weak classifiers [10] [11] available in the OpenCV library. The tracking is based on a particle filter technique (bootstrap filter) where the particle weight for a given ROI depends on the face classifier response [4]. For a given ROI x_k^i, we take the classifier response as the maximum level reached in the weak classifier cascade (the maximum being 24).

3.2 Face tracking

Given a set of face bounding box $\{x_{k-1}^i\}_{i=1}^N$ (particles), an iteration k of the tracking algorithm consists in 4 steps:

1. For each particle i, draw a new particle $\mathbf{x}_k^i \sim p(\mathbf{x}_k \mid \mathbf{x}_{k-1}^i)$.
2. Assign a weight $\tilde{\omega}_k^i = 1 - 0.8^{z_k^i}$ derived from the response z_k^i (cascade level) of the face detector.
3. Normalize the weight as $\omega_k^i = \tilde{\omega}_k^i \Big/ \sum_i \tilde{\omega}_k^i$.
4. Generate a new set of random samples $\{x_k^{i*}\}_{i=1}^N$ by resampling $\{x_k^i\}_{i=1}^N$ (with replacement) from the discrete distribution $\{\omega_k^i\}_{i=1}^N$. An example of result is given on Figure 2 and Figure 3

On Figure 2, a tracking result is shown for a shot containing two individuals side by side. A spatio-temporal display of the recorded tracks is shown on Figure 3. Tracks are fragmented due to the frontal view of the faces in the middle of the shot.

Figure 2. Tracking of two actors walking and talking in an alley. The red curves are the face tracks. The green boxes are the result of the particle filter.

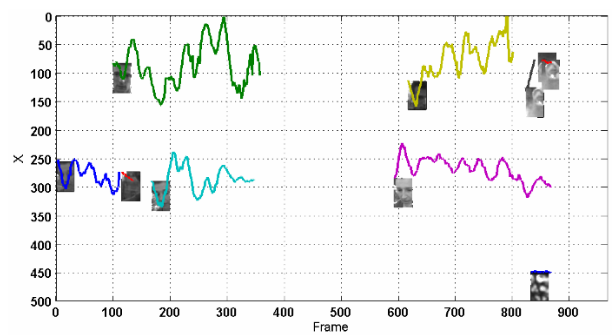

Figure 3. Resulting tracks for the shot shown on Figure 2. The vertical axis is the horizontal position X of the tracked face.

3.3 Frontal views identification

The response of the near-frontal view detector is very sensitive to changes in face pose. The detector response map shows tight compact peak for a frontal face and a uniform response otherwise (Figure 4). In order to quantify this qualitative change, we propose to quantify the compactness of the peak response using the coefficient

$$Q = \frac{|R|}{\bar{W} \times \bar{H}} \quad (1)$$

where R is the largest image segment corresponding to the peak. This segment is determined by first thresholding the detection map and then applying a connected components analysis. $\bar{W}$ and $\bar{H}$ are the average face dimensions determined only from the positions associated to R.

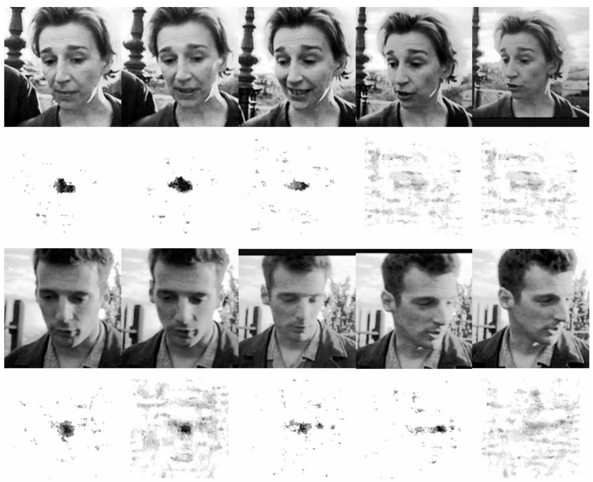

Figure 4. Examples of detected faces along with their respective probability map (dark values are for high probability values).

3.4 Face normalization

The face is simply normalized by determining the center of the face based on the detector response map to correct for translation and scaling variations. We do not correct for in-plane or in-depth rotations which would require the additional detection of several face features (eyes, mouth, etc.). The normalized image is 84x64 pixels and covers most of the face (from the tip of the chin to the forehead) while avoiding too much contamination by the background.

4. Face clustering

4.1 Face representation

PCA techniques have been successful in face recognition. However, the concatenation of 2D images into 1D vectors leads to high-dimensional vector space with the following typical problems: 1) the estimation of the covariance matrix is difficult due to the *small sample size* (SSS) problem; 2) computing the eigenvectors is very time-consuming; 3) feature vectors lies in a very high-dimensional space which brings the *curse of dimensionality* dilemma. On the opposite, the 2DPCA techniques [1][3][2] directly seek the optimal projective vectors from face images without preliminary image-to-vector transformation. These methods have the following advantages over PCA: 1) the size of the covariance matrix is either ($H \times H$) or ($W \times W$) for a $H \times W$ image; 2) because the input are factually the rows or the columns, the feature set is significantly enlarged which reduces the SSS problem; 3) the 2D information is better preserved. The first two advantages lead to faster and more robust eigenvector estimation. However, one disadvantage is that more coefficients are needed to represent the image (i.e. the K first eigenvectors generate $H \times K$ coefficients).

Variants of the original 2DPCA have been proposed recently:

- the Dia2DPCA based on diagonal images [3], which better captures correlation between the image rows and columns;
- a Bilateral-projection-based 2DPCA (B2DPCA) that constructs simultaneously two subspaces to encode the row and columns vectors [2];
- 2DPCA can be performed along the columns (2DPCACOLS) or the rows (2DPCAROWS).

To calculate the distance between two feature matrices $\mathbf{A}$ and $\mathbf{B}$, we use the *Assembled Distance Metric* (AMD) [27] with $p=0.125$:

$$d_{AMD}(\mathbf{A},\mathbf{B}) = \left(\sum_{j=1}^{K} \left(\sum_{i=1}^{m} \left(\mathbf{A}_{ij} - \mathbf{B}_{ij} \right)^2 \right)^{\frac{1}{2}p} \right)^{1/p}, \ p > 0 \quad (2)$$

In the following sections, we assume N face trajectories $\{\mathbf{I}^{(i)}\}_{i=1,\ldots,N}$. Each face trajectory is composed of M normalized face samples $\{\mathbf{I}^{(i,m)}\}_{m=1,\ldots,M}$. A pairwise distance matrix $\mathbf{D}'$ is formed by calculating the minimum AMD distance between faces in the training sets. For instance, for 2DPCACOLS:

$$\mathbf{D}'_{ij} = \min_{m,n \in \{1,\ldots,M\}} \left\{ \begin{matrix} d_{AMD}(\left[\mathbf{I}^{(j,m)} - \bar{\mathbf{I}}^{(i)}\right]\mathbf{W}^{(i)}, \\ \left[\mathbf{I}^{(i,n)} - \bar{\mathbf{I}}^{(i)}\right]\mathbf{W}^{(i)}) \end{matrix} \right\} \quad (3)$$

where $\mathbf{W}^{(i)}$ is the 2DPCACOLS projection matrix derived from the training sample of the ith trajectory with the mean image $\bar{\mathbf{I}}^{(i)}$. The matrix $\mathbf{D}'$ is not symmetric so we need to symmetrize the distance matrix $\mathbf{D} = \min\left\{\mathbf{D}', \mathbf{D}'^{T}\right\}$.

4.2 Spectral clustering techniques

Spectral clustering [5][14][18][19] refers to a class of techniques which is dependent on the eigenstructure of an affinity matrix (i.e. the matrix of pairwise affinities between points of the datasets) to partition the data objects into disjoint clusters. They differ from central grouping techniques (e.g. k-medoids, k-means, EM, etc.) which are based on distances from cluster prototypes (centroïds) usually assuming Gaussian distributions. Spectral clustering has been shown to be optimal for affinity matrices that are block stochastics [13]. Recently, many applications have been published in the context of video indexation, scene representation and video summarization [2][15][16][20]. The first step in spectral clustering is the estimation of an affinity matrix $\mathbf{S}$ usually based on the exponential of the pairwise distance matrix $\mathbf{D}$:

$$\mathbf{S}_{ij} = e^{-\mathbf{D}_{ij}/(2\sigma^2)} \tag{4}$$

An important parameter is the Gaussian kernel size σ which greatly influences the clustering result. Most often, a global parameter is estimated from the statistics of the pairwise distances. However, it is intuitive to think that the value for σ should be class-dependent. Recently, local estimation techniques have been proposed [19][23] where the scale factor σ_i is estimated for each line *i*. In [19], the scale factor is estimated so that the sum $\sum_j S_{ij}$ is a fixed constant $\tau = 10$ (method AAM). In [23], σ_i is set to be equal to the *k*-nearest distance (*k=7*); this is called the *Local Scaling* (LS) approach. The impact of the value of *k* and τ is shown on Figure 6.

Once the affinity matrix has been computed, a *spectral representation* is computed based on the eigenvectors of the affinity matrix according to the following. Given a similarity matrix **S**:

1. Normalize the similarity matrix $\mathbf{S} \to \tilde{\mathbf{S}}$.
2. Compute the *k* first eigenvectors $\mathbf{v}_i$ of $\tilde{\mathbf{S}}$ to form the spectral feature matrix $\mathbf{Y} = \left[\mathbf{v}_1, ..., \mathbf{v}_k\right]$
3. Normalize each row of **Y** to unit vector:

Many variants have been proposed that differ mainly in the way the affinity matrix is normalized and the application of step 3. The effect of the normalization is to enhance the block structure of the similarity matrix. In Table I, we summarize the 5 different normalization techniques that we have considered: the Multiway NCUT (MNCUT) [26], the Ng-Jordan-Weiss algorithm (NJW) [5], the conductivity method (COND) [19] and no normalization (NN).

Table 1. Normalizations used for comparison

Algorithm	Normalization	Ref
MNCut	$\tilde{\mathbf{S}} = \mathbf{D}^{-1}\mathbf{S}$	[26]
NJW	$\tilde{\mathbf{S}} = \mathbf{D}^{-1/2}\mathbf{S}\mathbf{D}^{-1/2}$	[5]
NN	$\tilde{\mathbf{S}} = \mathbf{S}$	
COND	$\tilde{\mathbf{S}} = Conductivity(\mathbf{S})$	[19]

The last step of spectral clustering is the actual unsupervised clustering where data points are now represented by their spectral feature **Y**. A simple k-means algorithm can be applied initialized with some orthogonal elements of **Y** [5]. Fischer and Poland have proposed a k-lines clustering from the observation that clusters may have linear shapes [19]. Yu and Shi [26] use a discretisation procedure.

5. Results

5.1 Evaluation on the ORL database

We use the ORL database in order to evaluate the different spectral clustering methods. The ORL database is composed of 40 different individuals with 10 grayscale images (110×92) per individual. There are no variations in the lighting conditions and the background is uniform. However, variations in pose and expression are significant. We split each sequence into 3 training sets of *M=3* images each, simulating samples from a face tracking trajectory. From the 400 images, we obtain *N=120* trajectories. Histogram equalization is performed before the computation of the 2DPCA features. The k-means algorithm is iterated 120 times, with different orthogonal initialization. For each clustering result we compute the standard silhouette value *s(i)* for each data point i belonging to cluster C_l:

$$s(i) = \frac{b(i) - a(i)}{1 - \min\left\{a(i), b(i)\right\}}, \text{ with}$$
$$a(i) = \frac{1}{\left|C_l\right|}\sum_{j \in C_l} \mathbf{S}_{ij} \text{ and } b(i) = \max_{k \neq l}\left\{\frac{1}{\left|C_k\right|}\sum_{j \in C_k} \mathbf{S}_{ij}\right\} \tag{5}$$

We keep the data partition that is maximizing the average silhouette.

We evaluate the clustering results in terms of minimum Clustering Error among all permutations of the cluster numbers [13]. The influence of the 2DPCA subspace dimension K is shown on Figure 5 when LS+NN+K-MEANS is applied with 40 classes. The optimal subset dimension is between 10 and 20 for all the 2DPCA methods. 2DPCACOLS gives the lowest error rate at 4.0% with $K=10$, followed by Dia2DPCA at 4.5% with $K=20$.

The effect of the scaling technique on the clustering error is shown on Figure 6 for NN+k-means. The Local Scaling (LS) technique [23] performs better than the AAM method [19] for the optimal parameters values used in those references ($k=7$ and $\tau = 10$).

On Figure 7, we compare the different affinity matrix normalizations presented in Table I assuming the LS method is applied on the distance matrix and k-means is used for the clustering. For comparison, we give also the results obtained when the direct grayscale correlation coefficient between images is employed as a similarity measure. We can see that No Normalization (NN) combined with the 2DPCACOLS features gives the best results (see Table II).

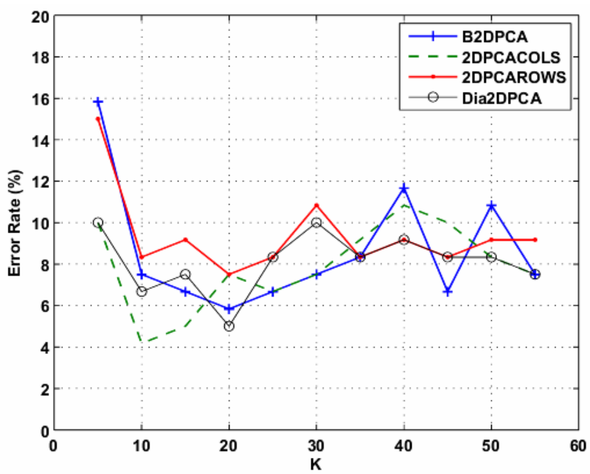

Figure 5. Influence of feature dimension on the error rate (LS+NN+k-means).

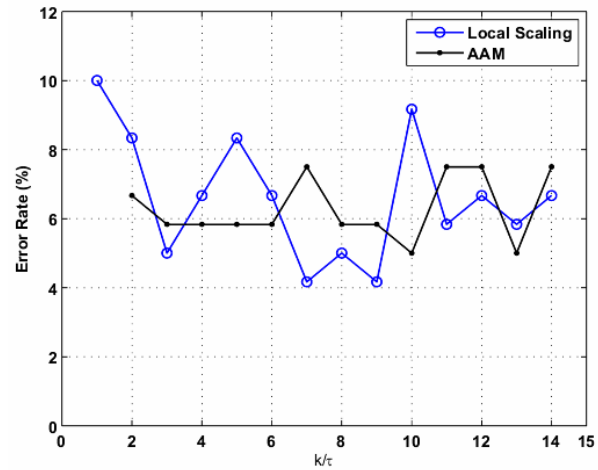

Figure 6. Influence of the k and τ parameters on the error rate.

Table II. Clustering Error Rates (%) for different features and normalization techniques (LS and k-means is used).

	COND	NJW	NCUT	NN
2DPCACOLS	*69.17*	*7.50*	*7.50*	*4.17*
2DPCAROWS	*69.17*	*10.00*	*9.17*	*8.33*
Dia2DPCA	*65.50*	*6.67*	*6.67*	*6.67*
B2DPCA	*65.83*	*6.67*	*7.50*	*5.83*
CORRELATION	*58.33*	*14.17*	*13.33*	*12.5*

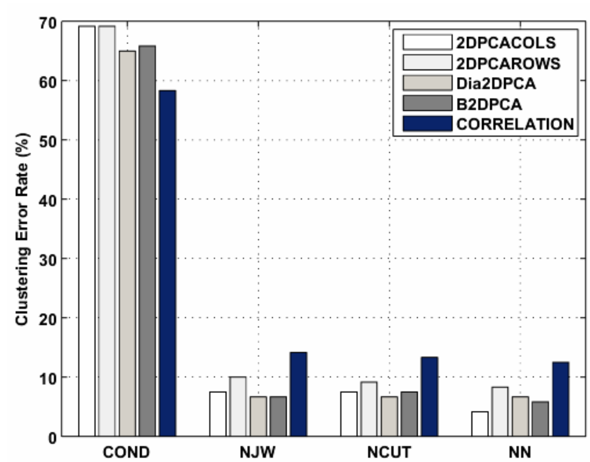

Figure 7. Clustering error rate for each normalization technique and different features.

5.2 Evaluation on a full-length feature film

The system was tested on the full-length French movie "Le Fabuleux Destin d'Amélie Poulain" that last 1.5 hours. About 1,280 shots have been automatically detected. The face tracking resulted in 1,287 trajectories (Figure 8).

Around 36.6% of the trajectories are false alarms and 10% are unknown faces (Figure 9). The actor 3 is the most observed (23.0%) whereas the other principal actors are between 2% and 5%.

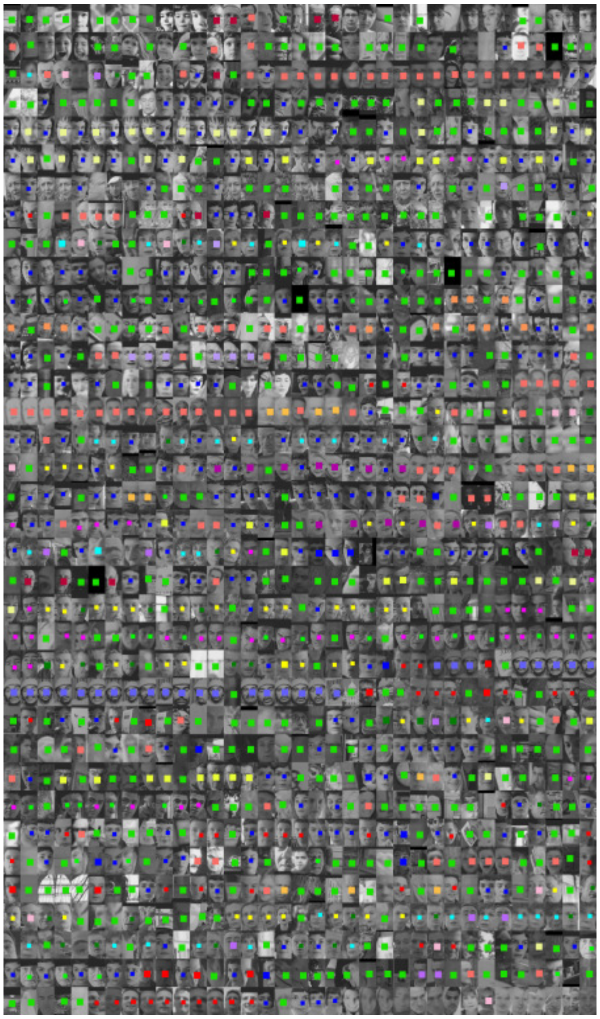

Figure 8. The 1,287 extracted faces (one face per trajectory) of a full-length movie. Each face is manually labeled (different color patches).

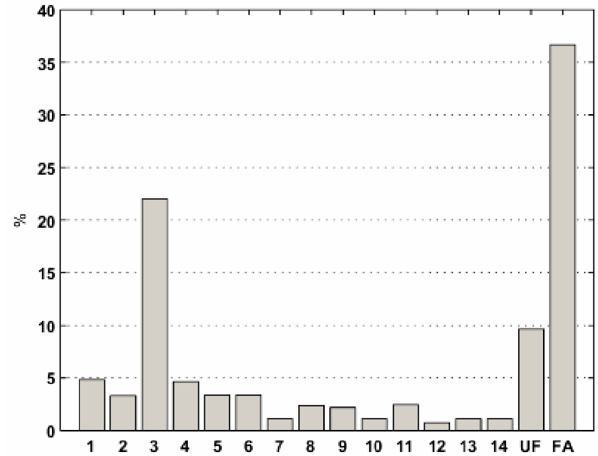

Figure 9. True face classes distribution of the full-length movie (UF= Unknown Faces, FA= False Alarms).

Figure 10. True clusters for the two main actors.

For each face trajectory, we keep the 3 best faces from which we compute the 2DPCACOLS eigenvectors after the images have been equalized. The affinity matrix is computed and then scaled using the LS method. The spectral clustering (NN + k-means) is applied with 1,000 iterations for k-means and 60 clusters. We keep the best partition according to the maximum average silhouette value criterion. The clustering result is presented to the user in the following way: clusters are ranked according to their average cluster silhouette. Within each cluster, tracks are ranked according to their decreasing silhouette value (5) so that outliers are displayed in last positions (see Figure 11).

A good cluster is a homogeneous cluster with no outliers that the user can labeled directly without further processing. However, some clusters have some degree of contamination as it is the case for the last cluster shown on Figure 11 for example. In that case, the cluster is still considered acceptable if the top tracks (largest silhouette values) in the cluster are homogeneous. Consequently, the clustering result is evaluated by counting the number of contiguous tracks from the same class on top of each cluster (i.e. the most similar tracks). The result is shown on Figure 12. Only 270 tracks on the 1,092 that have been labeled have been correctly identified or 24.7%. If we exclude the false alarms and the unknown faces, we have 180 tracks identified on 587 (30.7%). This recognition rate reaches 35.6% when we consider only the top 4 actors. The principal actor (actor 3) is recognized at 42.5%. The clustering method can then be repeated only on the unidentified tracks.

Our implementation was done in C++. The computation time is about 24 hours for the detection and tracking of faces (can be run on several computers), 1 hour for the calculation of the affinity matrix and 45 minutes for the clustering (Pentium IV, 2.2 Ghz, 512 Mb RAM). Note that the clustering procedure can be repeated without the need to recalculate the affinity matrix. The peak memory load was around 100 Mb

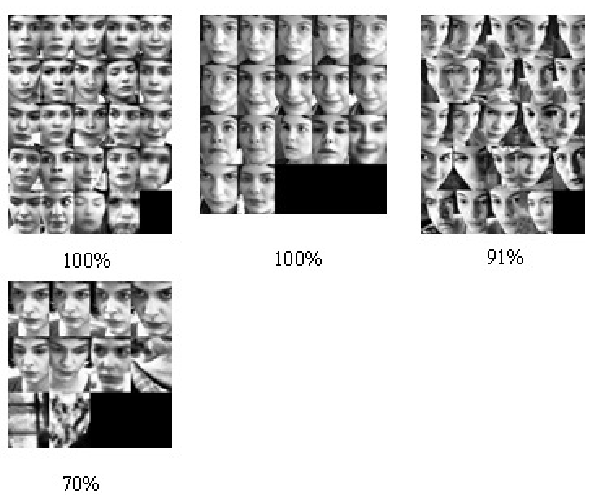

Figure 11. Top four clusters for Actor 3. The good classification rate for each cluster is given.

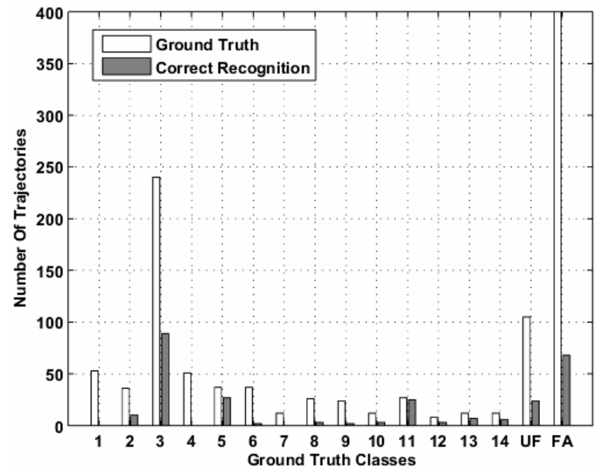

Figure 12. Number of correctly identified tracks for each class (UF= Unknown faces, FA= False Alarm).

6. Conclusion

In this paper, we investigate the performance of spectral clustering techniques based on 2DPCA features for the clustering of actor faces in movies. A preliminary evaluation on the ORL database shows that a distance matrix based on 2DPCACOLS, the Local Scaling method with no normalization of the affinity matrix gives the lowest clustering error rates.

The clustering method is then applied on a large dataset of faces extracted from a full length feature movie. The overall clustering performance (one iteration) is only of 25%.

Future work will be focusing on reducing the number of false alarms (currently one third of the tracks). One way to achieve this is to increase the performance of the face detector as well as the face normalization. The current system is also unable to detect, track and recognize faces that have a non-frontal pose (e.g. profiles).

Acknowledgments

This work is supported in part by (1) the Department of Canadian Heritage (www.pch.gc.ca) through Canadian Culture Online, (2) the Natural Science and Engineering Research Council (NSERC) of Canada (www.nserc.ca) and (3) the Ministère du Développement Économique de l'Innovation et de l'Exportation (MDEIE) of Gouvernement du Québec..

References

[1] Yang, J., Zhang, D., Frangi, A. F., Yanf, J., "Two-Dimensional PCA: A New Approach to Appearance-Based

Face Representation and Recognition", Trans. On Pattern Analysis and Machine Intelligence, 26(1): 131-137, 2004.
[2] Kong, H., Li, X., Wang, L., Teoh, E. K., Wang, J.-G., Venkateswarlu, R., "Generalized 2D Principal Component Analysis", IEEE International Joint Conference on Neural Networks (IJCNN), Montreal, Canada, 2005.
[3] Zhang, D., Zhou, Z.-H., Chen, S., "Diagonal principal component analysis for face recognition", Pattern Recognition, 2006, 39(1):140-142.
[4] Verma, R.C., Schmid, C., Mikolajczyk, K, "Face Detection and Tracking in a Video by Propagating Detection Probabilities", IEEE Trans. On PAMI, Vol. 25, No. 10, 2003.
[5] Ng, A.Y., Jordan, M., Weiss, Y., "On spectral clustering: Analysis and an Algorithm", In Proceedings of the Advances in Neural Information Processing Systems 14, 2001.
[6] Sivic, J., Everingham, M., Zisserman, A., "Person Spotting: Video Shot Retrieval for Face Sets". International Conference on Image and Video Retrieval (CIVR 2005), Singapore (2005)
[7] Fitzgibbon, A. W., Zisserman, A., "On Affine Invariant Clustering and Automatic Cast Listing in Movies", Proceedings of the 7th European Conference on Computer Vision, Copenhagen, Denmark (2002), Vol. 3, pp. 304-320.
[8] Everingham, M., Zisserman, A., "Automated Person Identification in Video", Proceedings of the International Conference on Image and Video Retrieval (2004), pp. 289-298.
[9] Everingham, M., Zisserman, A., "Automated Visual Identification of Characters in Situation Comedies", Proceedings of the International Conference on Pattern Recognition (2004), pp. 983-986.
[10] Viola, P., Jones, M. J., "Rapid object detection using a boosted cascade of simple features," IEEE CVPR, pp. 511-518, 2001.
[11] Lienhart, E., Maydt, J., "An extended Set of Haar-like Features for Rapid Object Detection", in IEEE ICME, 2002.
[12] Arandjelovic, O., Zisserman, A., "Automatic Face Recognition for Film Character Retrieval in Feature-Length Films", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego (2005), pp. 860-867.
[13] Verma, D., Meila, M., "A Comparison of Spectral Clustering Algorithms", UW CSE Technical report 03-05-01.
[14] Shi J., Malik, J., "Normalized cuts and image segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 8, pp. 888-905, Aug. 2000.
[15] Odobez, J.-M., Gatica-Perez, D., Guillemot, M., "Spectral structuring of home videos," in Proc. Int. Conf. Image and Video Retrieval, 2003, pp. 310-320.
[16] Rasheed, Z., Shah, M., "Detection and Representation of Scenes in Videos", IEEE Transactions on Multimedia, Vol. 7, No 6, Dec. 2005, pp. 1097-1105.
[17] Satoh, S., Nakamura, Y., Kanade, T., "Name-It: Naming and Detecing Faces in News Videos". IEEE Multimedia, 6(1):22-35, 1999.
[18] Fowlkes, C., Belongie, S. Chung F., Malik J., "Spectral Grouping Using the Nystrom Method", IEEE Trans. On PAMI, Vol. 26, no2, pp. 1-12, 2004.
[19] Fischer, I., Poland, Jan, "Amplifying the Block Matrix Structure for Spectral Clustering", In Proceedings of the 14th Annual Machine Learning Conference of Belgium and the Netherlands, pp. 21-28, 2005.
[20] Ekin, A., Pankanti, S., Hampapur, A., "Initialization-Independent Spectral Clustering with Applications to Automatic Video Analysis", ICASSP'2004.
[21] Eickeler, S., Wallhoff, F., Iurgel, U., Rigoll, G., "Content-based Indexing of Images of Video using Face Detection and Recognition Methods", ICASSP'2001.
[22] Cerneková, Z., Pitas, I., Nikou, C., "Information Theory-Based Shot Cut/Fade Detection and Video Summarization", IEEE Trans. On Circuits and Systems for Video Technology, Vol. 16, No. 1, 2006, pp. 82-91.
[23] Zelnik-Manor, L., Perona, P., "Self-Tuning Spectral Clustering", Eighteenth Annual Conference on Neural Information Processing Systems, (NIPS) 2004.
[24] Meila, M., Shi, J., "Learning Segmentation by Random Walks". In Advances in Neural Information Processing Systems, pp. 873-879, 2000.
[25] Gagnon, L., Foucher, S., Laliberté, F., Lalonde, M., Beaulieu, M., "Toward an Application of Content-Based Video Indexing to Computer-Assisted Descriptive Video", CRV, 2006.
[26] S. X. Yu, J. Shi, "Multiclass Spectral Clustering", International Conference on Computer Vision, Nice, France, 11-17 Oct 2003.
[27] Zuo, W.-M., Wang, K.-Q., Zhang, D., "Assembled Matrix Distance Metric for 2DPCA-Based Face and Palmprint Recognition", In Proc. Of ICMLS, pp. 4870-4875, 2005.