A MUTUAL INFORMATION BASED FACE CLUSTERING ALGORITHM FOR MOVIES

N. Vretos, V. Solachidis and I. Pitas

Department of Informatics
University of Thessaloniki, 54124, Thessaloniki, Greece
phone: +30-2310996304, fax: +30-2310996304
email:vretos,vasilis,pitas@aiia.csd.auth.gr

ABSTRACT

In this paper a new approach for face clustering is developed. Mutual information and joint entropy are exploited in order to create a metric for the clustering process. The way the joint entropy and the mutual information are calculated gives some interesting properties to the aforementioned metric, which guarantees some robustness against standard noisy transformation such as scaling, cropping and pose changes. A slight preprocessing of the input face images is done in order to undertake problems that arise from detector's known errors.

1. INTRODUCTION

Face clustering is a very important application for movie's semantic extraction. It can contribute in many ways, like determining the primordial actors or the creation of databases' references or dialog detection and many others. Until now some interesting algorithms have been proposed in [1]-[3], but most of them are based in calibrated face images from news or face recognition databases like [1]. In [4], a more close to our approach algorithm is proposed, which proposes a solution for the same problem as the one we are trying to solve. It is based on a preprocessing of the input face images and then it uses kernel PCA methods in order to extract features and classifies the processed images. Our approach limits the preprocessing phase and exploits the capabilities of joint entropy and mutual information in order to classify face images exported from a Haar like features detector like the one proposed in [5].

Mutual information (MI) is a novel and useful tool in order to find similarities between information. More concretely, MI is defined as the information that is shared between two distributions. Until now, MI is much exploited in bioinformatics application and serves many purposes in that field from DNA sequences categorization [6] to classification of proteins [7]. In image processing MI is used, in many reprise, in image registration for medical images [7] and gives relatively good results. In this paper we will investigate the possibility of using this tool for classification of face images in a more realistic content such as movies, where difficulties arise from the fast variations of illumination, scale, pose etc.

The remainder of this paper is organized as follows : in Section 2 a mathematical presentation of the mutual information and its normalized version are presented and the way that we are using it in our approach. In Section 3 the detection process and the fuzzy c-means clustering algorithm is presented, as well as how we define the clustering objects. In Section 4 we demonstrate results for a real movie case. Finally in Section 5 further work and conclusions are discussed.

2. MUTUAL INFORMATION FOR FACE CLUSTERING

Mutual information is defined as the information shared between two distribution. Let X and Y be two distributions. We define the joint entropy as:

$$H(X,Y) = -\sum(p(x,y)\log(p(x,y))) \tag{1}$$

where $p(x,y)$ the normalized (summed to one) probability density function of the common information of distribution X and Y. In the same way, we define the Shannon's entropy for X and Y as :

$$H(X) = -\sum(p(x))\log(p(x)) \tag{2}$$

$$H(Y) = -\sum(p(y))\log(p(y)) \tag{3}$$

Therefore we can define the mutual information as:

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{4}$$

where this give use the final equation of mutual information:

$$I(X;Y) = \sum_x \sum_y p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{5}$$

$I(X;Y)$ is a quantity that measures the mutual dependence of two random variables. If we use a logarithm with base 2, then the measure is in bit. This quantity needs to be somehow normalized in order to create a uniform metric between different images and therefor be used as a similarity measure. For this reason, we use the normalized MI, which is defined as the quotient of the sum of two entropies with the joint entropy of those two distribution.

$$NMI(X;Y) = \frac{H(X)+H(Y)}{H(X,Y)} \tag{6}$$

Is is also useful to notice that:

$$NMI(X;Y) = \frac{H(X)+H(Y)}{H(X,Y)} \tag{7}$$

$$NMI(Y;X) = \frac{H(X)+H(Y)}{H(Y,X)} \tag{8}$$

But as we know from (1) :

$$H(X,Y) = H(Y,X) \tag{9}$$

So,

$$NMI(X;Y) = NMI(Y;X) \tag{10}$$

A very detailed explanation of how this normalizes the mutual information can be found in [8].

In our approach we use the intensity images and we define for every image the distribution density function as the histogram of the intensities of that image summed to one. In order to calculate the joint entropy between the two images we construct a 2D histogram of 256 bins which take in account the relative positions of intensities so that similarity occurs between two images, when same intensities are located in same spacial locations. Less literarily, the 2D joint histogram is calculated as follow: Let A and B be the first and the second image respectively of size $N_1 \times N_2$. And $i, j \in [0, 255]$ then:

$$Hist(i,j) = |\{(k,l) \in N_1 \times N_2 \mid A(k,l) = i \text{ and } B(k,l) = j\}| \tag{11}$$

where $|\cdot|$ denotes the cardinality of a set.

By defining the joint histogram that way, we have to admit that in order to calculate it, images have to be of same size.This means that one has to resize one image to the other's dimensions. In our approach, and in order to equalize big scaling interpolation issues, we define a mean bounding box which is calculated from all bounding boxes that the face detector provides to us. This approach shows better results than if we scale every pair of images forward the bigger or the smaller of them. So every image is scaled towards this mean bounding box before the mutual information calculation.

Another issue is the fact of anisotropic scaling. Detector's results are bounding boxes where typically width and height are not equal. In order to scale forward a mean bounding box problems arise when the two dimensions are not equal. To override this, we calculate the bigger dimension of the bounding box and then we are taking the square box that equals this dimension centered to the original's bounding box center. Less literally, let $B = \{x_1, y_1, x_2, y_2\}$ be a bounding box. We define the width as $x_2 - x_1$ and the height as $y_2 - y_1$. From the two dimensions we take the bigger one and stretch the other at that size. The resulting bounding box for B for width bigger than height (resp. height bigger than width), will be:

$$\begin{aligned} B_{new} &= \{x_1, y_1 - k, x_2, y_2 + k\} \\ (resp.\ B_{new} &= \{x_1 + k, y_1, x_2 - k, y_2\}) \end{aligned} \tag{12}$$

where k equals $\frac{(x_2-x_1)-(y_2-y_1)}{2}$.

We have noticed that problems arise from scaling issues that involves detectors inaccuracy. This means that if the face is not correctly detected and the face image contains a big amount of the background then scaling is mismatching the two face images and results are inaccurate. In order to override this bottleneck, another processing step is made, which is inspired from registration algorithms and aim in maximizing the accuracy of our results.

Once we have put the detector's outputs in the same scale we calculate the NMI for different frames of the target face image. We vary the bounding box's width and height from 80% to 120% of the initial mean bounding box, with a step of 5%. The aforementioned values are calculated experimentally. In this way, we are trying to eliminate scaling problems due to detector's errors. In figure 1 one can see two images which show the aforementioned case. Finally, we take the maximum of the calculated NMIs between the two images.

As mentioned before, the movies' context is dominated from several difficulties in order to extract content information. The way our approach is using the mutual information is undertaking those problems. By using the scale variance within the detectors results and the point-to-point approach of the joint entropy we have succeeded to provide good results in a very complicated task. In [4] the problem is tackled based on a preprocessing of the image. Our approach is trying to avoid the preprocess and goes deeper in the mutual information properties to that end.

Fig. 1. In this image one can see that images are of different scales but faces are practically of same size .

2.1. Mutual Information Vectors

Our algorithm consists of creating a vector of MIs for every image. The dimension of that vector is equal to the size of the face detection results' data set. For every face image in the results set we calculate the NMI between this image and any other, and therefore we create a vector $\mathbf{v}$. All those vectors results in an $M \times M$ matrix (where M the cardinal of the set of all detections from a video sequence) where every row i of that matrix will be the NMI of the i-th detection with all other images.

$$S(i,j) = NMI(FaceImage_i, FaceImage_j) \tag{13}$$

It is obvious that the elements of the diagonal will have value one, which is the normalized mutual information of a face image with itself and also the matrix will be symmetric w.r.t the main diagonal. The diagonal property of the matrix is a forward effect of the MI symmetry shown in eq. (10). Those properties are very helpful because they drastically intervene in the time complexity of the algorithm. By using those properties the time complexity is minimized by a multiplicative factor of 0.5 and an additive factor of -M. In figure 2 one can see the image of a matrix S for a 253 detections set. In this figure a test of consecutive appearances of two different actors is shown. One has to notice the square regions that appear in that image and that way we can understand that same persons appear. The thin lines that appears are in most cases detectors false results which are very different from the face pattern.

3. FUZZY C-MEANS CLUSTERING

In order to cluster our results we use the fuzzy c-means algorithm. This method has been proven that in situation where we have a light mixture of classes' elements, it performs better than the simple k-means algorithm. Other more sophisticated methods for clustering where abandoned because there where very consuming in time and did not give better results in this context. Therefore Fuzzy c-means (FCM) looked like the optimal solution in our case.

In order to use this algorithm we define every row of the aforementioned matrix M as a different vector in an M-dimensional L^2-normed vector space over $\mathbb{R}$. In figure 3 and 4 one can see how those vectors are formed for two examples of 709-dimensional vectors.

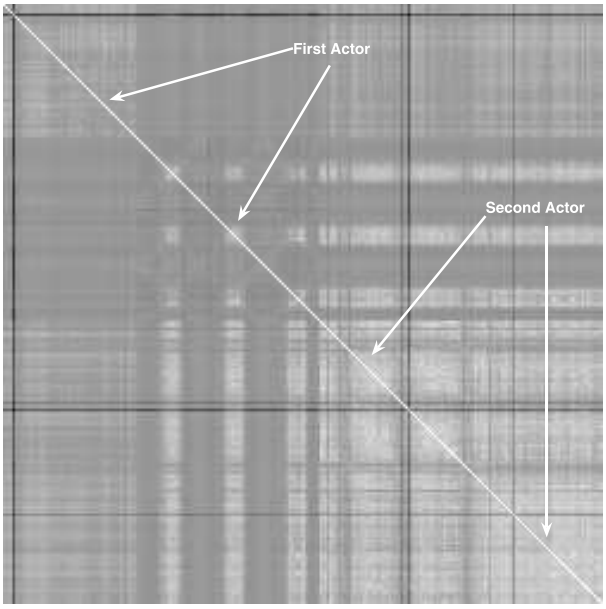

Fig. 2. Darker regions belongs to the first actor and clearer ones to the second actor. The video sequence has four consecutive shots in the order FA-FA-SA-SA where FA and SA first and second actor respectively.

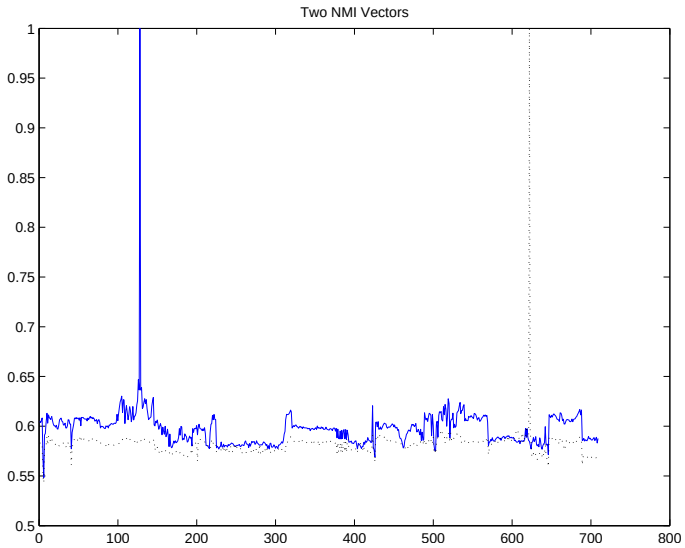

Fig. 3. Two vectors which belong to different clusters from an experiment of 709 detections. The picks at 128 and 622 define the mutual information of the images with themselves.

Therefore, we use the Euclidian distance to calculate distances between the vectors

$$dist(\mathbf{v_i}, \mathbf{v_j}) = \sqrt{\sum_{k=1}^{M} (v_{i_k} - v_{j_k})^2} \tag{14}$$

and by those means to calculate a predefined number of clusters' centers. A detailed implementation of the FCM algorithm can be found in [9].

We have seen that initialization has a significant role for FCM. So in order to provide better results the first centers are manually selected in a way that faces that corresponds in different actors constitutes a different initial center. A random selection of initial centers vary the results of a factor of 0.5% of false classification.

For our detection method we have chosen a detector based in Haar-like features [5] which we boost with a color optimization and also an ellipsis fitting algorithms in order to tune up its results. Errors from the detection process are count in the evaluation of the face clustering algorithm.

4. EXPERIMENT SCENARIO AND RESULTS

In order to test our algorithm we have conducted the following experiment. From a movie called the "Two weeks notice" we have extracted a set of 709 detections from 2000 frames. The frames that we have used are 6500 to 8500 in a 23 fps version of that film. First we execute the detection algorithm in every frame and then we have used the detector's results in order to evaluate our algorithm. In the detector's results set we have two main actors and some wrong detections as well. We consider two cases of results: one where detector's faults belongs to the set and another where we have manually removed false detections.

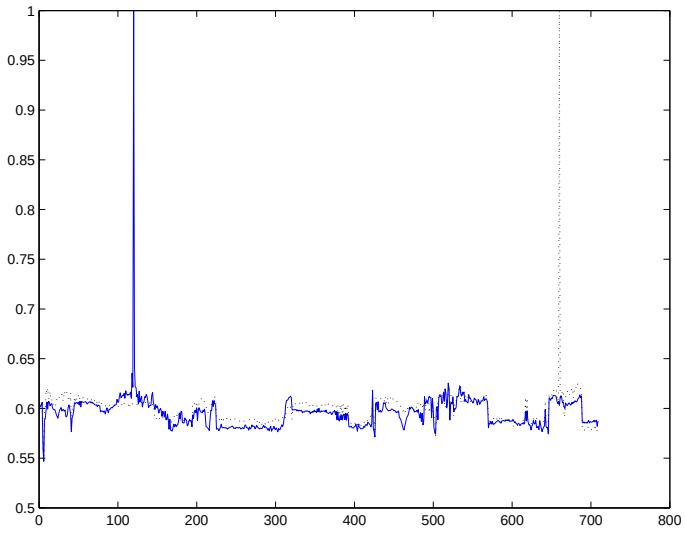

Fig. 4. Two vectors which belong to the same cluster from an experiment of 709 detections. The picks at 120 and 660 define the mutual information of the images with themselves.

	Cluster 1	Cluster 2	Wrong Detections
Cluster 1	260	40	18
Cluster 2	7	346	38

Table 1. In this scenario we achieve an average of 85.4% of correct classifications and a 6.6% of uncorrect. An error of 7.9% is introduced from the detector.

	Cluster 1	Cluster 2
Cluster 1	260	39
Cluster 2	8	355

Table 2. In this scenario we achieve an average of 93% of correct classifications and a 7% of uncorrect. Detectors faults are manually removed. Notice the correction in cluster 2 from 346 to 355 after we have remove the detector's faults.

The frames where selected so that light conditions and actor's poses vary within the scene. In the detector's results set we end up with a variety of face images in many poses and light conditions. With this approach we ensure the robustness of our algorithm in those kind of attacks. In order to calculate the percentages of good and bad classifications we use the following procedure: We calculate all the good and bad classifications in all the clusters and then we divide this number with the number of detections from the detector's output set. In other words, let suppose that we have N_i good and L_i bad classifications for cluster i. Then if M is the cardinal of the detector's output the percentage of good classifications is

$$\frac{\sum_i N_i}{M} \cdot 100\%$$

, while the percentage of the bad ones is

$$\frac{\sum_i L_i}{M} \cdot 100\%$$

Finally, and for the first case only, the calculation of the percentage of wrong detections, follows the same principal. In table 1 and 2 one can see the results for the two cases respectively.

It is important to notice that the remove operation is applied before the whole clustering operation, and that way, some correction of the results is happening because data are more uniform. In other word the cluster centers are not substituted in changes because of wrong detections where this can introduce wrong calculation of them.

5. CONCLUSIONS AND FUTURE WORK

We have developed a method for clustering face images within a very complex context such as movies. Results, as shown before, are rather promising for this difficult task if one considers the big variations that arise, w.r.t. light conditions, pose changes, emotions changes etc. in such a context. As face clustering has a lot of application in multimedia development, image processing and content based image retrieval applications (CBIR) we will investigate this problem further and we will concentrate our effort in integrating information from trackers in order to boost results in the classification process. Tracker will provide us with arrays of image faces, which belong to the same person and that way we will have to calculate the mutual information between groups of face images which will correct errors for some cases of no frontal faces for instance.

The proposed method is a novel approach of the use that one can make of the mutual information in image analysis, and give good results in a hard task like the one we are solving. Exploration of the joint entropy and the mutual information on image data is shown to be a very good similarity criterion which can help in many other image processing application as well.

On the other hand, with our approach we minimize time complexity because of less preprocessing on the face images. This is an advantage for applications who needs fast clustering process, like interactive TV applications.

6. ACKNOWLEDGEMENT

The work presented was developed within NM2 (New media for a New Millennium), a European Integrated Project (http://www.ist-nm2.org), funded under the European Commission IST FP6 program.

7. REFERENCES

[1] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Y. W. Teh, E. Learned-Miller, and D. A. Forsyth, "Names and faces in the news," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR'04)*. IEEE, 2004, vol. 2nd, pp. 848–854.

[2] Mark Everingham and Andrew Zisserman, "Automated person identification in video.," in *CIVR*, 2004, pp. 289–298.

[3] A. Fitzgibbon and A. Zisserman, "On affine invariant clustering and automatic cast listing in movies," ECCV, 2002.

[4] O. Arandjelovic and A. Zisserman, "Automatic face recognition for film character retrieval in feature-length films," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego*, 2005, pp. 860–867.

[5] Paul Viola and Michael Jones, "Robust real-time object detection," *International Journal of Computer Vision - to appear*, 2002.

[6] Xiaobo Zhou, Xiaodong Wang, Edward R. Dougherty, Daniel Russ, and Edward Suh, "Gene clustering based on clusterwide mutual information.," *Journal of Computational Biology*, vol. 11, no. 1, pp. 147–161, 2004.

[7] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, "Mutual-information-based registration of medical images: a survey," *Medical Imaging, IEEE Transactions on*, vol. 22, no. 8, pp. 986–1004, 2003.

[8] Zengyou He, Xiaofei Xu, and Shengchun Deng, "K-anmi: A mutual information based clustering algorithm for categorical data," 2005.

[9] R L Cannon, J V Dave, and J C Bezdek, "Efficient implementation of the fuzzy c-means clustering algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 2, pp. 248–255, 1986.