Frontal view recognition in multiview video sequences

I. Kotsia[1,2], N. Nikolaidis[1,2] and I. Pitas[1,2]
[1]Aristotle University of Thessaloniki, Department of Informatics, Box 451, 54124, Greece
[2]Informatics and Telematics Institute, CERTH, Greece

***Abstract*—In this paper, a novel method is proposed as a solution to the problem of frontal view recognition from multiview image sequences. Our aim is to correctly identify the view that corresponds to the camera placed in front of a person, or the camera whose view is closer to a frontal one. By doing so, frontal face images of the person can be acquired, in order to be used in face or facial expression recognition techniques that require frontal faces to achieve a satisfactory result. The proposed method firstly employs the Discriminant Non-Negative Matrix Factorization (DNMF) algorithm on the input images acquired from every camera. The output of the algorithm is then used as an input to a Support Vector Machines (SVMs) system that classifies the head poses acquired from the cameras to two classes that correspond to the frontal or non frontal pose. Experiments conducted on the IDIAP database demonstrate that the proposed method achieves an accuracy of 98.6% in frontal view recognition.**

I. Introduction

The aim of the proposed method is to take advantage of existing face or facial expression recognition techniques that require frontal faces. The scenario under consideration includes multiple cameras that are placed at certain known angles in a convergent setup in order to properly capture the movements of a person. In such a scenario, the proposed algorithm can be used to identify the view that corresponds to the camera placed in front of a person, or the camera whose view is closer to a frontal one. By doing so, frontal face images of the person can be acquired and fed to a face or facial expression recognition technique that requires frontal faces. The face or facial expression recognition problem task is thus approached in a multi-view environment, leading to view-independent face or facial expression recognition.

Two cases can be handled by such an approach. The first case assumes that the person's head pose remains the same throughout the video sequence, whereas the second one assumes that the person's head pose changes through time. In the latter case, the camera that provides a frontal view should be detected at each frame. It should be noted that the proposed technique can be also used in an analogous manner for the utilization of existing frontal face or facial expressions recognition techniques in a multi-view environment.

For the proposed method, the images (frames) acquired from each camera are used as an input to the Discriminant Non-Negative Matrix Factorization (DNMF) algorithm. The DNMF algorithm is a matrix decomposition algorithm that is an extension of the Non-negative Matrix Factorization (NMF) algorithm. The NMF algorithm is an unsupervised algorithm that allows only additive combinations of non negative components. DNMF was the result of an attempt to introduce discriminant information to the NMF decomposition in a supervised manner. The NMF and DNMF algorithms will be presented analytically below. DNMF decomposes an image into a linear combination of basis images. The DNMF's output, namely the decomposition coefficients, is then inserted into a Support Vector Machines (SVMs) system that performs the final classification into the two desired classes (frontal or non frontal facial images). A diagram of the proposed system is depicted in Figure 1.

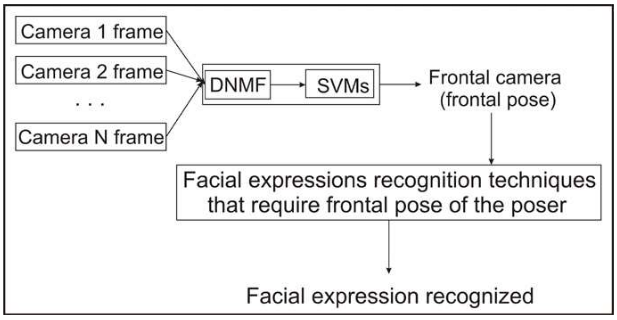
Camera 1 frame
Camera 2 frame
. . .
Camera N frame
DNMF
SVMs
Frontal camera
(frontal pose)
Facial expressions recognition techniques
that require frontal pose of the poser
Facial expression recognized

Fig. 1. Diagram of the proposed system

II. Discriminant Non-Negative Matrix Factorization Algorithms

In this Section, the Non-Negative Matrix Factorization (NMF) algorithm and the procedure followed to formulate its variant, the DNMF approach [1], are briefly presented.

Let an image scanned row-wise so as to form a vector $\mathbf{x} = [x_1 \dots x_F]^T$ for the NMF algorithm. The basic idea behind NMF is to approximate (with small approximation error) the image $\mathbf{x}$ by a linear combination of a set of basis images in $\mathbf{Z} \in \Re_+^{F \times M}$, whose coefficients are the elements of $\mathbf{h} \in \Re_+^M$ such that $\mathbf{x} \approx \mathbf{Z}\mathbf{h}$. In order to train the NMF, the matrix is constructed, where x_{ij} is the i-th element of the j-th image vector. In other words, the j-th column of $\mathbf{X}$ is the facial image x_j. NMF aims at finding two matrices and such that:

$$\mathbf{X} \approx \mathbf{Z}\mathbf{H}. \tag{1}$$

Obviously, the application of NMF requires the evaluation of the basis images in $\mathbf{Z}$. This is done by a training phase that requires a set of training images $x_1 \dots x_T$.

After the NMF decomposition, the facial image x_j can be written as $\mathbf{x}_j \approx \mathbf{Z}\mathbf{h}_j$, where $\mathbf{h}_j$ is the j-th column of $\mathbf{H}$. Thus, the M columns of the matrix $\mathbf{Z}$ can be considered as the M basis images and the vector $\mathbf{h}_j$ as the weight vector that corresponds to image $\mathbf{x}$. The vector $\mathbf{h}_j$ can be also considered as the projection of $\mathbf{x}_j$ in a lower dimensional space.

The cost for the decomposition (1) can be defined as the sum of all KL divergences for all images in the database:

$$\begin{aligned} D(\mathbf{X}||\mathbf{Z}\mathbf{H}) &= \textstyle\sum_j KL(\mathbf{x}_j||\mathbf{Z}\mathbf{h}_j) \\ &= \textstyle\sum_{i,j} \left(x_{i,j} \ln(\frac{x_{i,j}}{\sum_k z_{i,k} h_{k,j}}) + \sum_k z_{i,k} h_{k,j} - x_{i,j} \right). \end{aligned} \tag{2}$$

The NMF factorization is the outcome of the following optimization problem:

$$\min_{\mathbf{Z},\mathbf{H}} D(\mathbf{X}||\mathbf{Z}\mathbf{H}) \text{ subject to} \tag{3}$$

$$z_{i,k} \geq 0, \; h_{k,j} \geq 0, \; \sum_i z_{i,j} = 1, \; \forall j.$$

In order to formulate the DNMF algorithm, let the matrix $\mathbf{X}$ that contains all the facial images that are organized in two classes $r = \{1, 2\}$. The first class consists of the frontal images while the second one of the non frontal images. The j-th column of $\mathbf{X}$ is the ρ-th image of the r-th image class. Thus, $j = \sum_{i=1}^{r-1} N_i + \rho$, where N_i is the cardinality of the image class i. It should be noted that the frontal image class consists of the images corresponding to one camera view provided that the person does not move (pan, roll) his head during the acquisition. If this is not the case, the images should be assigned to the two classes manually. In this case, the images from the other cameras make up the non-frontal image class.

The columns of the matrix H are divided to two sets, each set containing the vectors $\mathbf{h}_j$ corresponding to each class r. The vector $\mathbf{h}_j$ that corresponds to the j-th column of the matrix $\mathbf{H}$, is the coefficient vector for the ρ-th facial image of the r-th class and will be denoted as $\boldsymbol{\eta}_\rho^{(r)} = [\eta_{\rho,1}^{(r)} \dots \eta_{\rho,M}^{(r)}]^T$. The mean vector of the vectors $\boldsymbol{\eta}_\rho^{(r)}$ for the class r is denoted as $\boldsymbol{\mu}^{(r)} = [\mu_1^{(r)} \dots \mu_M^{(r)}]^T$ and the mean of all classes as $\boldsymbol{\mu} = [\mu_1 \dots \mu_M]^T$. Then, the within-class scatter matrix for the coefficient vectors $\mathbf{h}_j$ is defined as:

$$\mathbf{S}_w = \sum_{r=1}^{K} \sum_{\rho=1}^{N_r} (\boldsymbol{\eta}_\rho^{(r)} - \boldsymbol{\mu}^{(r)})(\boldsymbol{\eta}_\rho^{(r)} - \boldsymbol{\mu}^{(r)})^T \tag{4}$$

whereas the between-class scatter matrix is defined as:

$$\mathbf{S}_b = \sum_{r=1}^{K} N_r (\boldsymbol{\mu}^{(r)} - \boldsymbol{\mu})(\boldsymbol{\mu}^{(r)} - \boldsymbol{\mu})^T. \tag{5}$$

The matrix $\mathbf{S}_w$ defines the scatter of the sample vector coefficients around their class mean. The dispersion of samples that belong to the same class around their corresponding mean should be as small as possible. A convenient measure for the dispersion of the samples is the trace of $\mathbf{S}_w$.

The matrix $\mathbf{S}_b$ denotes the between-class scatter matrix and defines the scatter of the mean vectors of all classes around the global mean $\boldsymbol{\mu}$. Each class must be as far as possible from the other classes. Therefore, the trace of $\mathbf{S}_b$ should be as large as possible.

To formulate the DNMF method [2], discriminant constraints have been incorporated in the NMF decomposition inspired by the minimization of the Fisher's criterion [2]. The DNMF cost function is given by:

$$D_d(\mathbf{X}||\mathbf{Z}\mathbf{H}) = D(\mathbf{X}||\mathbf{Z}\mathbf{H}) + \gamma \text{tr}[\mathbf{S}_w] - \delta \text{tr}[\mathbf{S}_b] \tag{6}$$

where γ and δ are non-negative constants. The update rules that guarantee a non-increasing behavior of (6) for the weights $h_{k,j}$ and the bases $z_{i,k}$, under the constraints of (2), can be found in [2].

Once the basis images have been calculated by the application of DNMF on the training face images, the facial image acquired from a certain camera is projected to the derived lower dimensional feature space $\tilde{\mathbf{g}} = \mathbf{Z}^T\mathbf{x}$ and is later inserted to a SVMs system that decides if the facial image under examination is frontal or not. A brief description of the SVMs system used [2] will be presented below.

III. Support Vector Machines Classifier

In order to decide if the facial image under examination is frontal or not, the output of the DNMF algorithm is used as an input to a two class SVMs system. The SVMs is trained with the frontal pose images in the set $\mathcal{U}^1 = \{(\mathbf{g}_j, y_j), j = 1, \dots, M, y_j = 1\}$ as positive examples and all non-frontal pose images in $\mathcal{U}^2 = \{(\mathbf{g}_j, y_j), j = 1, \dots, K, y_j = -1\}$ as negative examples where $\mathbf{g}_i$ is the output of the DNMF algorithm and y_j is the image label.

The SVMs used for our experiments were proposed in [3] and are a variant of the typical maximum margin SVMs. They have been inspired by the optimization of the Fisher's discriminant ratio and incorporate statistic information about the classes under examination. The typical maximum margin SVMs as well as the variant that was used for the experiments will be presented below in detail.

A. Maximum margin SVMs

In order to train the SVMs network, the following minimization problem has to be solved [4]:

$$\min_{\mathbf{w}_k, b_k, \boldsymbol{\xi}^k} \quad \frac{1}{2}\mathbf{w}_k^T\mathbf{w}_k + C_k \sum_{j=1}^{N} \xi_j^k \tag{7}$$

subject to the separability constraints:

$$y_i^k(\mathbf{w}_k^T \phi(\mathbf{g}_j) + b_k) \geq 1 - \xi_j^k, \xi_j^k \geq 0, \quad j = 1, \dots, N \tag{8}$$

where b_k is the bias for the k-th SVM, $\boldsymbol{\xi}^k = [\xi_i^k, \ldots, \xi_w^k]$ is the slack variable vector and C_k is the term that penalizes the training errors.

After solving the optimization problem (7) subject to the separability constraints (8) ([5], [6]), the function that decides whether the facial image corresponds to a frontal pose is:

$$f_k(\mathbf{g}) = \mathrm{sign}(\mathbf{w}_k^T \phi(\mathbf{g}) + b_k) \tag{9}$$

where $\mathcal{G}$ is an arbitrary dimensional Hilbert space [7] and $\phi : \Re^L \rightarrow \mathcal{G}$. In this formulation, a nonlinear mapping ϕ has been used for a high dimensional feature mapping for obtaining a linear SVMs system in which it should be $\phi(\mathbf{g}) = \mathbf{g}$. This mapping is defined by a positive kernel function, $h(\mathbf{g}_i, \mathbf{g}_j)$, specifying an inner product in the feature space and satisfying the Mercer condition [5], [6]:

$$h(\mathbf{g}_i, \mathbf{g}_j) = \phi(\mathbf{g}_i)^T \phi(\mathbf{g}_j). \tag{10}$$

The function used as the SVMs kernel was the d degree polynomial function:

$$h(\mathbf{g}_i, \mathbf{g}_j) = ({\mathbf{g}_i}^T \mathbf{g}_j + 1)^d. \tag{11}$$

and the Radial Basis Function (RBF) kernel:

$$h(\mathbf{g}_i, \mathbf{g}_j) = \exp(-\gamma \parallel \mathbf{g}_i - \mathbf{g}_j \parallel^2). \tag{12}$$

where γ is the spread of the Gaussian function.

B. SVMs proposed in [3]

In order to form the optimization problem of the SVMs proposed in [3] we should define the within class scatter matrix of the training set:

$$\mathbf{S}_w^k = \sum_{\mathbf{g}_i \in \mathcal{U}_k^1} (\mathbf{g}_i - \boldsymbol{\mu}_k^1)(\mathbf{g}_i - \boldsymbol{\mu}_k^1)^T + \sum_{\mathbf{g}_i \in \mathcal{U}_k^2} (\mathbf{g}_i - \boldsymbol{\mu}_k^2)(\mathbf{g}_i - \boldsymbol{\mu}_k^2)^T \tag{13}$$

where $\boldsymbol{\mu}_k^1$ and $\boldsymbol{\mu}_k^2$ are the mean vectors of the classes $\mathcal{U}_k^1$ and $\mathcal{U}_k^2$, respectively. It is assumed that the within scatter matrix $\mathbf{S}_w^k$ is invertible (which is true in our case, since the dimensionality of the vector $\mathbf{g}_i$ is classically smaller than the number of available training examples). The optimization problem of the modified SVMs is [3]:

$$\min_{\mathbf{w}_k, b_k, \boldsymbol{\xi}^k} \qquad \mathbf{w}_k^T \mathbf{S}_w^k \mathbf{w}_k + C_k \sum_{j=1}^{N} \xi_j^k \tag{14}$$

subject to the separability constraints (8) (here we refer to the linear case where $\phi(\mathbf{g}) = \mathbf{g}$).

The linear decision function that decides whether the facial image under examination corresponds to a frontal pose or not, is:

$$f_k(\mathbf{g}) = \mathrm{sign}(\mathbf{w}_k^T \mathbf{g} + b_k) = \mathrm{sign}(\frac{1}{2} \sum_{j=1}^{N} y_i^k a_i^k \mathbf{g}_j^T {\mathbf{S}_w^k}^{-1} \mathbf{g} + b_k). \tag{15}$$

IV. Experimental Results

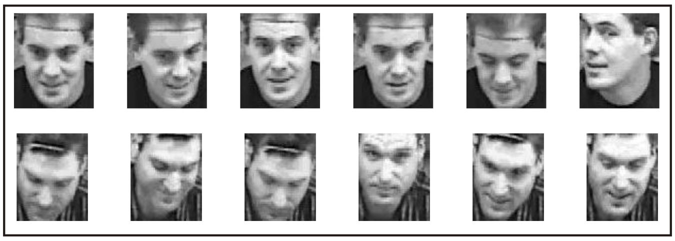

Fig. 2. An example of the IDIAP database

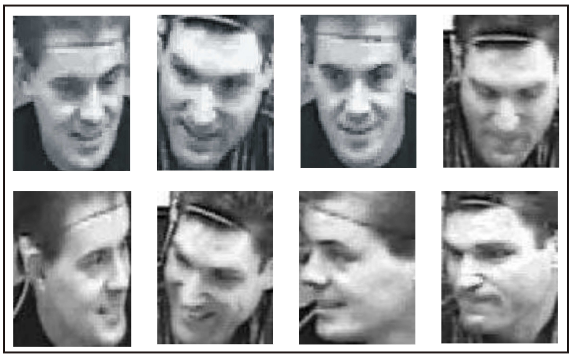

Fig. 3. Examples of frontal (upper row) and non frontal (lower row) facial images from the IDIAP database

Due to the lack of multiview data with accompanying ground truth, experiments were performed in the IDIAP database [8]. The database comprises of 23 video sequences involving people engaged in natural activities. In total, 16 different subjects participate in the video database. The database contains head pose ground truth in the form of pan, tilt and roll angles (i.e. Euler angles with respect to the camera coordinate system) for each frame of the video sequences.

Face detection and tracking were applied on the images acquired from the video cameras and the resulting Regions Of Interest (ROI) were inserted in the DNMF algorithm. An example of the results of a face tracker for a video from the IDIAP database is shown in Figure 2.

For the experiments, appropriate ground-truth data were extracted from the IDIAP database. The images regarded as frontal facial images included images with a slight pan and roll movement, taking under consideration that in a multiview environment the camera positions might be such that no camera captures a perfectly frontal image. In this case the view that is closer to a frontal one should be detected. Examples of facial images that were assigned to the frontal facial pose class (allowing a head displacement of $10^{\mathbf{o}}$ in all axes) are depicted in the first row of Figure 3, while in the second row examples from the non frontal facial class (head rotation more than $10^{\mathbf{o}}$ in all axes) are shown.

The most usual approach for testing the generalization performance of a SVMs classifier, is the leave-one-out cross validation approach [9] which enables the maximal use of the available data and evaluates averaged classification accuracy on the test dataset. A variant of this approach was used in our case. More specifically, all facial images contained in the database were divided into 2 classes, each one corresponding to frontal and non frontal poses, according to the range of degrees we defined as an acceptable head rotation in each axis. Five sets containing 20% of the images contained in each class, chosen randomly, were created. One such set was used as test set, while the remaining four sets formed the training set. After the classification procedure is performed, the samples forming the testing set were incorporated into the current training set, and a new set of samples (20% of the samples for each class) was extracted to form the new test set. The remaining samples create the new training set. This procedure was repeated five times. The average classification accuracy was calculated as the mean value of the percentages of the correctly classified facial images.

The confusion matrix has also been computed. The confusion matrix is a $n \times n$ matrix (n being the number of classes) containing information about the actual class label lab_{ac} in its columns and the label obtained through classification lab_{cl} in its rows. The diagonal entries of the confusion matrix are the percentages of facial images that are correctly classified, while the off-diagonal entries are the percentages corresponding to misclassification rates.

The accuracies achieved when head rotation for frontal images was within $5^{\mathbf{o}}$, $10^{\mathbf{o}}$, $15^{\mathbf{o}}$ and $20^{\mathbf{o}}$ in each axis, were equal to 98.6%, 98.2%, 95.6% and 94.9%, respectively. The confusion matrix of the experiments when the acceptable head rotation for a frontal pose was within $10^{\mathbf{o}}$ is presented in Table 1. All the above results were achieved using a RBF kernel with $\gamma = 0.1$.

V. Future work

The proposed frontal view detection algorithm will be combined with existing facial expression recognition algorithms that require frontal view images in order to judge their performance as a single system on multi-view data. Research towards facial expressions recognition algorithms that work on multi-view data and exploit all available views will be also conducted. Later on, when 3D reconstructions of persons in a scene become available, methods that operate on such data will be researched.

VI. Conclusion

Frontal view recognition in multiview video sequences has been investigated in this paper. A novel method that uses the DNMF algorithm in combination with an SVMs system in order to detect the frontal pose from an image acquired from a camera has been proposed. Experiments performed in the IDIAP database yielded an accuracy rate equal to 98.6% in frontal view recognition.

Acknowledgments.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n 211471 (i3DPost).

References

[1] S. Zafeiriou, A. Tefas, I. Buciu, and I. Pitas, "Exploiting discriminant information in non-negative matrix factorization with application to frontal face verification," *IEEE Transactions on Neural Networks*, vol. 17, no. 3, pp. 683–695, 2006.

[2] ——, "Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification," *IEEE Transactions on Neural Networks*, vol. 17, no. 3, pp. 683 – 695, 2006.

[3] A. Tefas, C. Kotropoulos, and I. Pitas, "Using support vector machines to enhance the performance of elastic graph matching for frontal face authentication," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 7, pp. 735–746, 2001.

[4] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass Support Vector Machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, March 2002.

[5] V. Vapnik, *Statistical learning theory*. New York: Wiley, 1998.

[6] C. J. C. Burges, "A tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge discovery*, vol. 2, no. 2, 1998.

[7] B. Scholkopf, S. Mika, C. Burges, P. Knirsch, K.-R. Muller, G. Ratsch, and A. Smola, "Input space vs. feature space in kernel-based methods," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1000–1017, 1999.

[8] S. Ba and J.-M. Odobez, "Evaluation of multiple cues head pose estimation algorithms in natural environments," in *IEEE International Conference on Multimedia and Expo (ICME)*, Amsterdam, 2005.

[9] I. Cohen, N. Sebe, S. Garg, L. S. Chen, and T. S. Huanga, "Facial expression recognition from video sequences: temporal and static modelling," *Computer Vision and Image Understanding*, vol. 91, pp. 160–187, 2003.