Texture and Shape Information Fusion for Facial Expression and Facial Action Unit Recognition

Irene Kotsia, Stefanos Zafeiriou and Ioannis Pitas

Aristotle University of Thessaloniki
Department of Informatics
Box 451
54124 Thessaloniki, Greece
Tel. ++ 30 231 099 63 04
Fax ++ 30 231 099 63 04
email: {ekotsia, dralbert, pitas}@aiia.csd.auth.gr

Preprint submitted to Elsevier *9 July 2007*

Abstract

A novel method based on fusion of texture and shape information is proposed for facial expression and Facial Action Unit (FAU) recognition from video sequences. Regarding facial expression recognition, a subspace method based on Discriminant Non-negative Matrix Factorization (DNMF) is applied to the images, thus extracting the texture information. In order to extract the shape information, the system firstly extracts the deformed Candide facial grid that corresponds to the facial expression depicted in the video sequence. A Support Vector Machine (SVM) system designed on an Euclidean space, defined over a novel metric between grids, is used for the classification of the shape information. Regarding FAU recognition, the texture extraction method (DNMF) is applied on the differences images of the video sequence, calculated taking under consideration the neutral and the expressive frame. An SVM system is used for FAU classification from the shape information. This time, the shape information consists of the grid node coordinate displacements between the neutral and the expressed facial expression frame. The fusion of texture and shape information is performed using various approaches, among which are SVMs and Median Radial Basis Functions (MRBFs), in order to detect the facial expression and the set of present FAUs. The accuracy achieved in the Cohn-Kanade database is 92.3% when recognizing the seven basic facial expressions (anger, disgust, fear, happiness, sadness, surprise and neutral), and 92.1% when recognizing the 17 FAUs that are responsible for facial expression development.

Key words: Facial expression recognition, Facial Action Unit Recognition, Discriminant Non-negative Matrix Factorization, multi-dimensional embedding, Support Vector Machines, Radial Basis Functions, Fusion.

1 Introduction

During the past two decades, facial expression recognition has attracted a significant interest in the scientific community, as it plays a vital role in human centered interfaces. Many applications such as virtual reality, videoconferencing, user profiling and customer satisfaction studies for broadcast and web services, require efficient facial expression recognition in order to achieve the desired results [1], [2]. Therefore, the impact of facial expression recognition on the above mentioned application areas, is constantly growing. Several research efforts have been performed regarding facial expression recognition. The facial expressions under examination were defined by psychologists as a set of six basic facial expressions (anger, disgust, fear, happiness, sadness and surprise) plus the neutral state [3]. In order to make the recognition procedure more standardized, a set of muscle movements known as $Facial\ Action\ Units$ ($FAUs$) that produce each facial expression, was cre-

ated, thus forming the so called *Facial Action Coding System* (*FACS*) [4]. These FAUs are combined in order to create the rules governing the formation of facial expressions, as proposed in [5]. A survey on the research made concerning facial expression recognition can be found at [6], [7]. Many approaches have been reported regarding facial expression recognition (direct or based on FAU recognition). These approaches can be distinguished in two main directions, those that use texture information (e.g. pixels intensity) and the rest that use geometrical or shape-based information (e.g. feature node displacements).

The most frequently used texture features are Gabor filter output [8]-[10], pixel intensities [11]-[16], Discrete Cosine Transform (DCT) features [15] and skin color information [17]-[19]. Accordingly, feature extraction methods based on Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) [13,20] have been used in order to enhance the performance of texture information. The classification of the texture information was performed using Neural Networks (NNs) [7,17,21], empirical classification rules [6,17,19], Bayes or Adaboost classifiers [9,10,18], SVMs [9,18,22].

The most frequently used shape features are facial features lines [23], motion information [8], [24,25], or facial action units [26]. These features are extracted using 2-D or 3-D facial models [5,27,28]. The classification of the shape information was performed using NNs [17], empirical classification rules [5,17], dynamic bayesian networks [8,24], dynamic time warping [26], template matching [29], Hidden Markov Models [30], manifold embedding [31], Bayes or Adaboost algorithms [32] or SVMs [9,33,34].

In [34], a technique for facial expression recognition has been proposed. The method employed considers the geometrical information of the Candide nodes, acquired as the coordinates differences, to use them as an input to SVM systems in order to achieve facial expression classification for the 6 basic facial expressions. The method in [34] had the following limitations:

- it is based on node displacements from the neutral state in order to recognize a basic expression, therefore, the recognition of the neutral state is necessary as a preprocessing step applied prior to the classification method
- texture information is not taken under consideration as only shape information is used.

In the current paper, a novel method for video based facial expression and FAU recognition is proposed that exploits both the texture and shape information. The recognized facial expressions are the seven basic ones (anger, disgust, fear, happiness, sadness, surprise and neutral), while the recognized FAUs are the ones contained in the rules proposed in [5] (FAUs 1, 2, 4, 5, 6, 7, 9, 10, 12, 15, 16, 17, 20, 23, 24, 25 and 26). The features of the facial texture are obtained by

applying a subspace representation method based on a discriminant extension of the Non-negative Matrix Factorization (NMF) algorithm [35] (the so-called DNMF algorithm [36]) on the images derived from the video sequence. In the case of facial expression recognition, the DNMF algorithm is applied directly on the expressive facial images, while in the case of FAU recognition, it is applied on the differences images. The differences images are calculated by subtracting the neutral frame of the video sequence from the fully expressed one. The differences images are used instead of the original facial expression images, due to the fact that they emphasize the facial regions in motion and reduce the variance related to the identity specific aspects of the facial image [37]. We should note that the differences images are only used in the FAU recognition process and not in the facial expression recognition one. In the case of FAU recognition, the neutral state is not taken under consideration, as no FAUs are present in it. Thus, the calculation of the differences images is, in that case, feasible. The neutral state can be found by using the results of the proposed facial expression recognition process, in order to derive the differences images.

The recognition of facial expressions and FAUs when using only either texture or shape information has certain drawbacks. When only texture is used, misclassification cases appear due to the lack of shape information in some specific facial expressions. For example, anger and fear differences images are not significantly different, implying that they cannot be discriminated well using only texture information. Such a problem can be solved with the introduction of shape information. On the other hand, when only shape information is used, subtle facial movements lead to facial expression misclassifications. For example, the mouth/lip movement can lead to a wrong facial expression recognition when either fear or happiness is recognized. By introducing texture information, these facial expressions are better separated. Thus, the fusion of texture and shape information is expected to provide superior results. Various methods were used to achieve the fusion of the two independent sources of information. The method that provided the best results was the Median Radial Basis Function (MRBF) NNs and thus it will be the only one described below.

The use of the DNMF algorithm for facial expression recognition has been motivated by the fact that it can achieve a discriminant decomposition of faces, as noted in [36]. In the frontal face verification problem [36], the DNMF method achieves a decomposition of the facial images, whose basis images represent salient facial features, such as eyes, eyebrows or mouth. We believe that the preservation of these salient features in the learning process of DNMF is caused by the class information taken into account by the algorithm, since these features are of great importance for facial identity verification. We also believe that the extension of the DNMF algorithm to facial expressions and FAU recognition problem is well motivated, since the algorithm is capable

of decomposing the images into facial parts that play a vital role to facial expression and FAU recognition [38],[39]. In the facial expression recognition problem, the class is composed of the images that belong to the same facial expression. Hence, there is a correlation between the features discovered by DNMF algorithm and the facial expression classification framework. This is indeed shown in the Section 6, where it is demonstrated that the DNMF basis images are salient facial parts that preserve discriminant information for every facial expression, like smile, lowered eyebrows etc, in contrast to the NMF basis images that do not display spacial locality of such high quality and Local-NMF (LNMF) basis images [40] that do not correspond directly to facial parts, even though they have better spacial localization than the equivalent basis images of NMF algorithm.

In the case of facial expression recognition, the shape information is calculated extracting the deformed Candide facial grid that corresponds to the facial expression depicted in the video sequence [34]. A space is created (via multidimensional scaling [41]-[43]) taking under consideration the distances calculated for every node to node correspondence between the training and testing grids. An SVM system is then used for the classification of the extracted shape information. In the case of FAU recognition, the shape information is extracted by calculating the Candide node displacements between the neutral and the expressive frame [34] that forms the facial expression. The FAU classification is obtained using a bank of two-class SVM systems. For facial expression recognition, both the texture and shape information extraction subsystems have as output the facial expression class whose center has the least distance from the test sample expression under examination. For FAU recognition, the set of FAUs that are adequate for facial expression representation are detected [5]. The experiments performed using the Cohn-Kanade database indicate a recognition accuracy of 92.3% when recognizing the seven basic facial expressions and 92.1% when recognizing the 17 basic FAUs. The FAU recognition is almost 10% better than the corresponding FAU recognition rate achieved when this set of FAUs and the Candide grid were used in [34].

Summarizing, the contributions of this study are:

- The extension of the DNMF algorithm presented in [36] for facial expression and FAU recognition.
- The introduction of a novel classification framework for facial grids that involves the definition of a new Euclidean space, based on metric multidimensional scaling, and its application to the Candide grids for facial expression recognition. This framework constitutes the recognition of seven facial expression feasible unlike [34] where the neutral state could not be recognized.
- The combination of texture and shape information for facial expression and FAU recognition.

The proposed method is different to the method in [34] since:

- facial expression recognition involves 7 facial expressions, the 6 basic ones (anger, disgust, fear, happiness, sadness and surprise) plus the neutral state. In [34], only the recognition of the 6 basic facial expressions is feasible since the knowledge of the neutral state is mandatory. In the proposed system the whole Candide grids are used instead of the nodes coordinates differences that were used in [34].
- A novel classification framework for grids that is comprised of two parts, an initial Euclidean embedding and a following multiclass SVM system, is proposed.
- Texture information is also used and its results are fused with shape information results to achieve better classification rates.

The rest of the paper is organized as follows: The systems used for facial expression and FAU recognition are outlined in Section 2. The DNMF algorithms for facial expression and FAU recognition are described in Section 3. The methods used for shape information extraction is presented in Section 4. The procedure followed in order to achieve the fusion of the extracted texture and shape information, is described in Section 5. The database used for the experiments and some observations regarding the results are described in Section 6.1. The recognition accuracy rates achieved for facial expression and FAU recognition are presented in Sections 6.2 and 6.3, respectively. Conclusions are drawn in Section 7.

2 System description

The system is composed of three subsystems: texture information extraction, shape information extraction and their fusion for final classification. A facial expression image database is created for the experiments. Regarding facial expression recognition, for each image sequence the fully expressive image from every video sequence is taken under consideration. In the case of FAU recognition, the difference images (see Figure 1), created by subtracting the neutral image intensity values from the corresponding values of the fully expressive image, are used for the texture information extraction subsystem. The differences images are used instead of the original facial expression images, due to the fact that they emphasize the facial regions in motion and reduce the variance related to the identity specific aspects of the facial image for FAU recognition [37]. The same image sequences are also used as input to the shape extraction information subsystem.

The grid tracking system used was the one described in [44]. An example of the Candide grids for every facial expression can be seen in Figure 2.

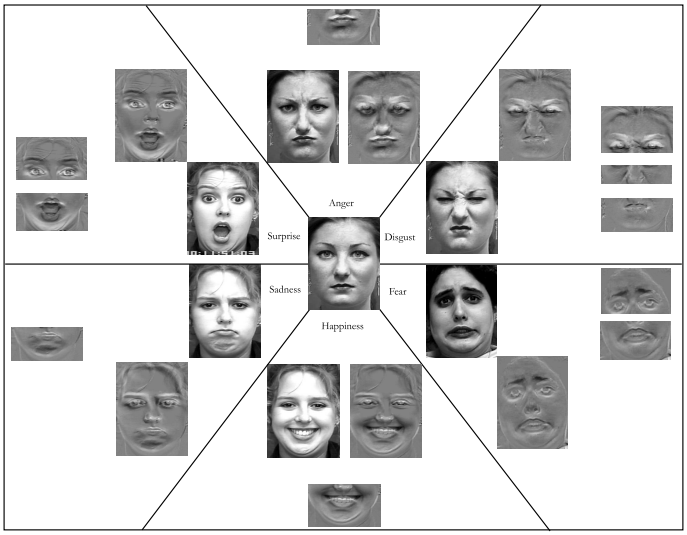


Fig. 1. Differences images between neutral pose and fully expressive one. They are split into facial regions containing the most expressive difference information.

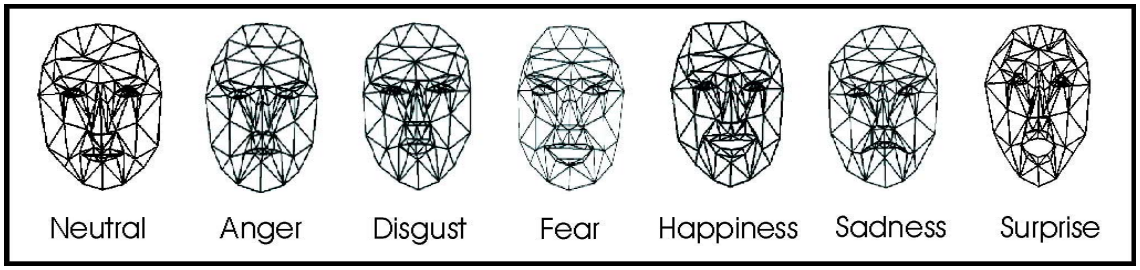


Fig. 2. An example of the Candide grid for every facial expression.

In the case of facial expression recognition, the extracted information is used as an input to the information processing subsystem that includes an Euclidean embedding. Finally, the information classification subsystem consists of a 7-class SVM system that classifies the embedded deformed grid into one of the 7 facial expression classes under examination. The subsystem used for facial expression recognition is shown in Figure 3.

For facial expression recognition, the output information from both the texture and shape classifiers consists of the distances of the test video sample from the winning class. These distances are fed to the fusion subsystem to provide the final classification result, i.e. the facial expression class the video sequence belongs to.

Facial expressions can also be described as combinations of FAUs, as proposed in [5]. As can be seen from the rules (Table 1), the FAUs 1, 2, 4, 5, 6, 7, 9, 10, 12, 15, 16, 17, 20, 23, 24, 25 and 26 are necessary for fully describing all facial expressions (see Figure 4). Therefore, we concentrate on the detection of these 17 FAUs. The operators +, *or* in Table 1 refer to the logical AND,

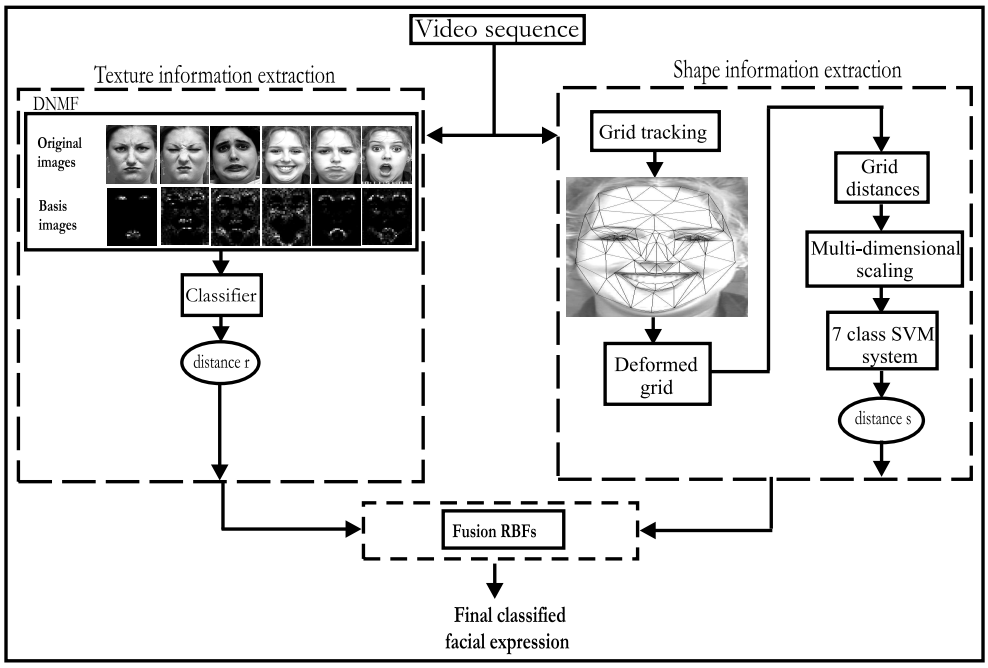


Fig. 3. System architecture for facial expression recognition in facial videos.

Table 1
The FAU to facial expressions rules as proposed in [5].

Expression	FAU coded description [5]
Anger	$4 + 7 + (((23 \text{ or } 24) \text{ with or not } 17) \text{or } (16 + (25 \text{ or } 26)) \text{ or } (10 + 16 + (25 \text{ or } 26)))$ with or not 2
Disgust	$((10 \text{ with or not } 17) \text{ or } (9 \text{ with or not } 17)) + (25 \text{ or } 26)$
Fear	$(1 + 4) + (5 + 7) + 20 + (25 \text{ or } 26)$
Happiness	$6 + 12 + 16 + (25 \text{ or } 26)$
Sadness	$1 + 4 + (6 \text{ or } 7) + 15 + 17 + (25 \text{ or } 26)$
Surprise	$(1 + 2) + (5 \text{ without } 7) + 26$

OR operations, respectively.

When FAU recognition is attempted, the extracted information obtained from the grid tracking system is used to calculate the Candide nodes differences between the neutral and fully expressive frame. The nodes differences are used as an input to a bank of 17 two-class SVM systems, each one corresponding to a FAU to be detected. Each SVM system is able to recognize if the FAU under examination is present or absent in the video sequence being examined.

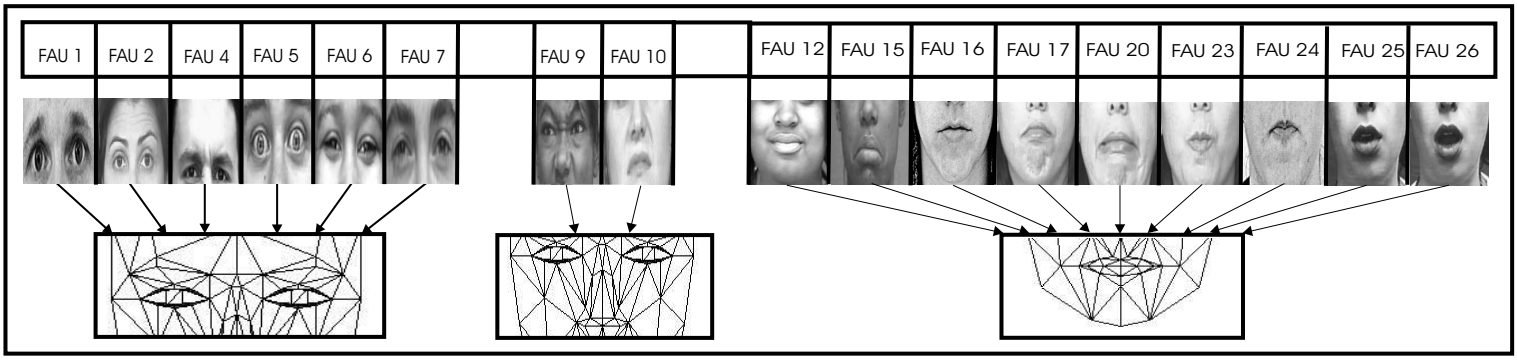


Fig. 4. Set of FAUs to be recognized and the corresponding part of the facial grid.

The subsystem used for FAU recognition is shown in Figure 5.

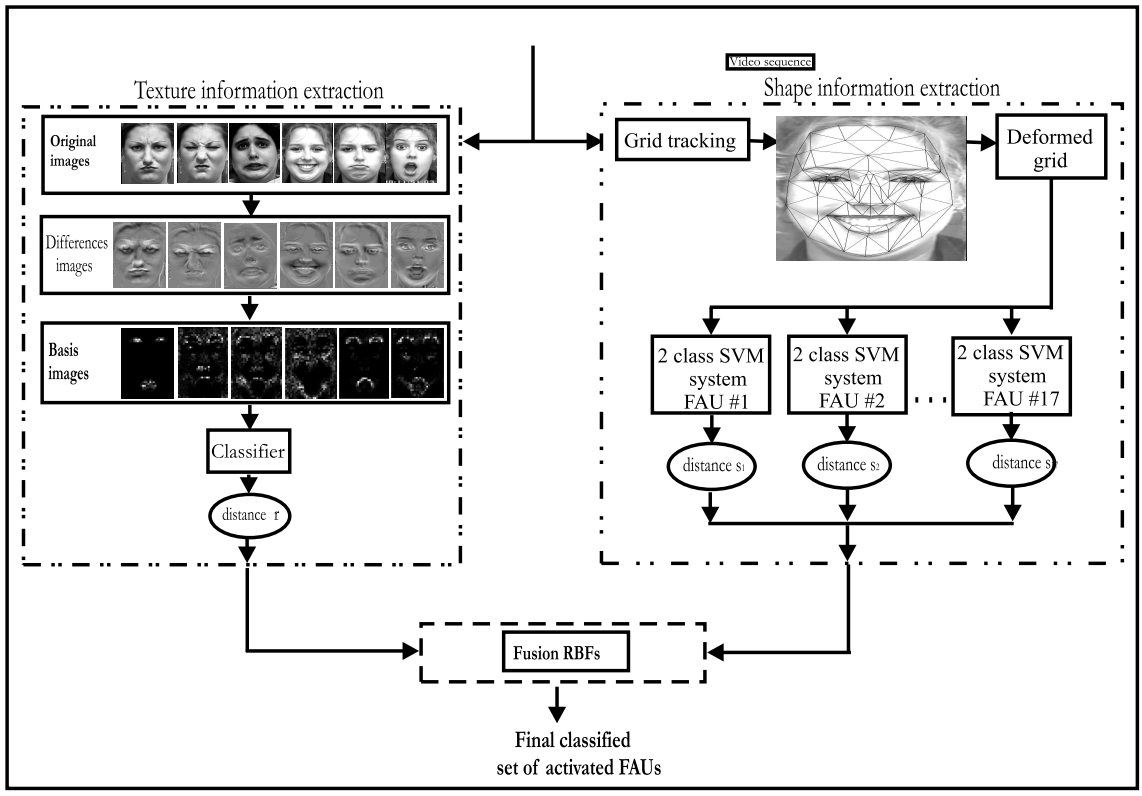


Fig. 5. System architecture for FAU recognition in facial videos.

For FAU recognition, the output information from both the texture and shape classifiers consists of a set of activated FAU in the examined video sequence. This set is fed to the fusion subsystem to provide the final classification result, i.e. the set of activated FAUs in the examined video sequence.

3 Texture information extraction and classification

In this Section, the extension of DNMF for facial expression and FAU recognition will be provided, starting by revisiting the NMF algorithm.

3.1 Facial expression recognition using texture information

For facial expression recognition, each expressive image $\mathbf{y} \in \mathcal{Y}$ belongs to one of the 7 basic facial expression classes $\{\mathcal{Y}_1, \mathcal{Y}_2, \ldots, \mathcal{Y}_7\}$. The facial image is scanned row-wise to form a vector $\mathbf{x} \in R_+^F$.

Let $\mathbf{x} = [x_1, \ldots, x_F], \mathbf{q} = [q_1, \ldots, q_F]$ be positive vectors $x_i > 0,\ q_i > 0$, then the Kullback-Leibler (KL) Divergence (or relative entropy) between $\mathbf{x}$ and $\mathbf{q}$ is defined [45] as:

$$KL(\mathbf{x}||\mathbf{q}) \triangleq \sum_i (x_i \ln \frac{x_i}{q_i} + q_i - x_i). \tag{1}$$

NMF tries to approximate the facial expression image $\mathbf{x}$ by a linear combination of the elements of $\mathbf{h} \in \Re_+^M$ such that $\mathbf{x} \approx \mathbf{Zh}$, where $\mathbf{Z} \in \Re_+^{F \times M}$ is a non-negative matrix, whose columns sum to one. In order to measure the error of the approximation $\mathbf{x} \approx \mathbf{Zh}$ the $KL(\mathbf{x}||\mathbf{Zh})$ divergence can be used [35]. In order to apply NMF, the matrix $\mathbf{X} \in \Re_+^{F \times L} = [x_{i,j}]$ should be constructed, where $x_{i,j}$ is the i-th element of the j-th image. In other words, the j-th column of $\mathbf{X}$ is the $\mathbf{x}_j$ image. NMF aims at finding two matrices $\mathbf{Z} \in \Re_+^{F \times M} = [z_{i,k}]$ and $\mathbf{H} \in \Re_+^{M \times L} = [h_{k,j}]$ such that :

$$\mathbf{X} \approx \mathbf{ZH}. \tag{2}$$

After the NMF decomposition, the image $\mathbf{x}_j$ can be written as $\mathbf{x}_j \approx \mathbf{Zh}_j$, where $\mathbf{h}_j$ is the j-th column of $\mathbf{H}$. Thus, the columns of the matrix $\mathbf{Z}$ can be considered as basis images and the vector $\mathbf{h}_j$ as the corresponding weight vector. The $\mathbf{h}_j$ vectors can also be considered as the projected vectors of lower dimensionality representing the original facial expression vector $\mathbf{x}_j$.

The defined cost for the decomposition (2) is the sum of all KL divergences for all images in the database. This way the following metric can be formed :

$$D_N(\mathbf{X}||\mathbf{ZH}) = \sum_j KL(\mathbf{x}_j||\mathbf{Zh}_j) = \sum_{i,j} (x_{i,j} \ln(\frac{x_{i,j}}{\sum_k z_{i,k} h_{k,j}}) + \sum_k z_{i,k} h_{k,j} - x_{i,j}) \tag{3}$$

as the measure of the cost for factoring $\mathbf{X}$ into $\mathbf{ZH}$ [35]. The NMF factorization is the outcome of the following optimization problem :

$$\min_{\mathbf{Z},\mathbf{H}} D_N(\mathbf{X}||\mathbf{ZH}) \text{ subject to} \tag{4}$$
$$z_{i,k} \geq 0,\ h_{k,j} \geq 0,\ \sum_i z_{i,j} = 1,\ \forall j.$$

NMF has non-negative constraints on both the elements of $\mathbf{Z}$ and of $\mathbf{H}$. These nonnegativity constraints permit the additive combination of multiple basis images in order to represent a facial expression. In contrast to PCA, no basis images subtractions can occur. For these reasons, the nonnegativity constraints correspond better to the intuitive notion of combining facial parts in order to create a complete expressive face. By using an auxiliary function and the Expectation Maximization (EM) algorithm [35], the following update rules for $h_{k,j}$ and $z_{i,k}$ guarantee a non increasing behavior of (3):

$$h_{k,j}^{(t)} = h_{k,j}^{(t-1)} \frac{\sum_i z_{i,k}^{(t-1)} \frac{x_{i,j}}{\sum_l z_{i,l}^{(t-1)} h_{l,j}^{(t-1)}}}{\sum_i z_{i,k}^{(t-1)}} \tag{5}$$

$$\acute{z}_{i,k}^{(t)} = z_{i,k}^{(t-1)} \frac{\sum_j h_{k,j}^{(t)} \frac{x_{i,j}}{\sum_l z_{i,l}^{(t-1)} h_{l,j}^{(t)}}}{\sum_j h_{k,j}^{(t)}} \tag{6}$$

$$z_{i,k}^{(t)} = \frac{\acute{z}_{i,k}^{(t)}}{\sum_l \acute{z}_{l,k}^{(t)}}. \tag{7}$$

where t is the iteration number. Since $\mathbf{x}_j \approx \mathbf{Z}\mathbf{h}_j$, a natural way to compute the projection of $\mathbf{x}_j$ to a lower dimensional feature space using NMF is $\acute{\mathbf{x}}_j = \mathbf{Z}^\dagger \mathbf{x}_j$ where $\mathbf{Z}^\dagger$ is the pseudo-inverse of $\mathbf{Z}$, given by $\mathbf{Z}^\dagger = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T$.

In order to incorporate discriminant constraints [36] in the NMF cost function and find the discriminant facial regions that are activated in the image for each different facial expression, let the vector $\mathbf{h}_j$ that corresponds to the jth column of the matrix $\mathbf{H}$ be the coefficient vector for the ρ-th facial image of the r-th facial expression class, which 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 facial expression class r is denoted by $\boldsymbol{\mu}^{(r)} = [\mu_1^{(r)} \dots \mu_M^{(r)}]^T$, the mean of all classes by $\boldsymbol{\mu} = [\mu_1 \dots \mu_M]^T$ and the cardinality of each facial class $\mathcal{Y}_r$ by N_r, respectively. Then, the within scatter for the coefficient vectors $\mathbf{h}_j$ is defined by:

$$\mathbf{S}_w = \sum_{r=1}^{7} \sum_{\rho=1}^{N_r} (\boldsymbol{\eta}_\rho^{(r)} - \boldsymbol{\mu}^{(r)})(\boldsymbol{\eta}_\rho^{(r)} - \boldsymbol{\mu}^{(r)})^T \tag{8}$$

whereas the between scatter matrix is defined as:

$$\mathbf{S}_b = \sum_{r=1}^{7} N_r (\boldsymbol{\mu}^{(r)} - \boldsymbol{\mu})(\boldsymbol{\mu}^{(r)} - \boldsymbol{\mu})^T. \tag{9}$$

A modified divergence can be constructed inspired by the minimization of the Fisher criterion. This is done by requiring tr[$\mathbf{S}_w$] to be as small as possible, while tr[$\mathbf{S}_b$] is required to be as large as possible. The new cost function is given by:

$$D_d(\mathbf{X}||\mathbf{Z}_D\mathbf{H}) = D_N(\mathbf{X}||\mathbf{Z}_D\mathbf{H}) + \gamma \text{tr}[\mathbf{S}_w] - \delta \text{tr}[\mathbf{S}_b], \tag{10}$$

where γ and δ are positive constants. Following the same EM approach used by NMF [35] and LNMF [40] techniques, the following update rules for the weight coefficients $h_{k,j}$ that belong to the r-th facial expression class are, $j \in F_r = \{\sum_{\rho=1}^{r-1} N_\rho + 1, \cdots, \sum_{\rho=1}^{r} N_\rho\}$ [36]:

$$h_{k,j}^{(t)} = \frac{T_1 + \sqrt{T_1^2 + 4(2\gamma - (2\gamma + 2\delta)\frac{1}{N_r}) h_{k,j}^{(t-1)} \sum_i z_{i,k}^{(t-1)} \frac{x_{i,j}}{\sum_l z_{i,l}^{(t-1)} h_{l,j}^{(t-1)}}}}{2(2\gamma - (2\gamma + 2\delta)\frac{1}{N_r})}, \tag{11}$$

where T_1 is given by:

$$T_1 = (2\gamma + 2\delta)(\frac{1}{N_r} \sum_{\lambda, \lambda \neq l} h_{k,\lambda}) - 2\delta\mu_k - 1. \tag{12}$$

The update rules for the bases $\mathbf{Z}_D$ are the same as in NMF and can be given by (6) and (7). The above decomposition is a supervised non-negative matrix factorization method that decomposes the facial expression images into parts, while enhancing class separability. The matrix $\mathbf{Z}_D^\dagger = (\mathbf{Z}_D^T\mathbf{Z}_D)^{-1}\mathbf{Z}_D^T$, which is the pseudo-inverse of $\mathbf{Z}_D$, is then used for extracting the discriminant features as $\acute{\mathbf{x}} = \mathbf{Z}_D^\dagger\mathbf{x}$. It is interesting to note here that there is no restriction on how many dimensions we may keep for $\acute{\mathbf{x}}$ and that the DNMF bases are common for all the different facial expression classes in the database, contrary to the DNMF algorithm applied for FAU recognition, where the extracted bases are class specific.

In order to make a decision about the facial expression class the test image belongs to, the image is projected to the lower dimensional feature space derived from applying the DNMF algorithm. The Euclidean distance between the projection $\acute{\mathbf{x}} = \mathbf{Z}_D^\dagger\mathbf{x}$ and the center of each facial expression class $\mathbf{m}_i$ is calculated and the image is classified to the closest facial expression class:

$$r_{\mathbf{x}} = \min_{i=1,\ldots,7} \|\mathbf{Z}_D^\dagger(\mathbf{x} - \mathbf{m}_i)\|. \tag{13}$$

3.1.1 FAU recognition using texture information

For FAU recognition, the differences images of each video sequence, calculated by subtracting the neutral frame from the expressive one, are used. Each differences image belongs to one of the 2 classes representing the presence/absence

of k-th FAU $\{\mathcal{Y}_k^{(1)}, \mathcal{Y}_k^{(2)}\}$. Each differences image calculated is initially normalized. The smallest intensity value for every image is defined and its absolute value is added to each pixel, resulting that way in a positive image. In both cases, the input image is afterwards scanned row-wise to form a vector $\mathbf{x}^\delta \in \Re_+^F$ of dimension F. As in the DNMF for facial expression recognition, we form the matrix $\mathbf{X}^\delta$ that has as columns the $\mathbf{x}^\delta$ images. The corresponding weight matrix $\mathbf{H}$ has columns the vectors $\boldsymbol{\eta}_i^{(1)}$ for the presence of the FAU and $\boldsymbol{\eta}_i^{(2)}$ for its absence.

For a two-class problem (like the k-th FAU recognition problem), we should define the within class scatter matrix of the training set as:

$$\mathbf{S}_w^k = \sum_{\boldsymbol{\eta}_i^{(1)} \in \mathcal{Y}_k^{(1)}} (\boldsymbol{\eta}_i^{(1)} - \boldsymbol{\mu}_k^{(1)})(\boldsymbol{\eta}_i^{(1)} - \boldsymbol{\mu}_k^{(1)})^T + \sum_{\boldsymbol{\eta}_i^{(2)} \in \mathcal{Y}_k^{(2)}} (\boldsymbol{\eta}_i^{(2)} - \boldsymbol{\mu}_k^{(2)})(\boldsymbol{\eta}_i^{(2)} - \boldsymbol{\mu}_k^{(2)})^T \quad (14)$$

where $\boldsymbol{\mu}_k^{(1)}$ and $\boldsymbol{\mu}_k^{(2)}$ are the mean vectors of the classes $\mathcal{Y}_k^{(1)}$ and $\mathcal{Y}_k^{(2)}$ (i.e., the presence and absence of the k-th FAU), respectively. The between scatter matrix is defined as:

$$\begin{aligned} \mathbf{S}_b^k &= \sum_{\boldsymbol{\eta}_i \in \mathcal{Y}_k^{(1)}} N_k^{(1)} (\boldsymbol{\mu}_k^{(1)} - \boldsymbol{\mu})(\boldsymbol{\mu}_k^{(1)} - \boldsymbol{\mu})^T + \sum_{\boldsymbol{\eta}_i \in \mathcal{Y}_k^{(2)}} N_k^{(2)} (\boldsymbol{\mu}_k^{(2)} - \boldsymbol{\mu})(\boldsymbol{\mu}_k^{(2)} - \boldsymbol{\mu})^T \\ &= N_k^{(1)} N_k^{(2)} (\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)})(\boldsymbol{\mu}^{(1)} - \boldsymbol{\mu}^{(2)}) \end{aligned} \quad (15)$$

where $N_k^{(1)}$ and $N_k^{(2)}$, are the cardinalities of the presence and absence of the k-th FAU classes, respectively. The DNMF cost function to be minimized is given by:

$$D_d(\mathbf{X}^\delta || \mathbf{Z}_D^k \mathbf{H}) = D_N(\mathbf{X}^\delta || \mathbf{Z}_D^k \mathbf{H}) + \gamma \mathrm{tr}[\mathbf{S}_w^k] - \delta \mathrm{tr}[\mathbf{S}_b^k]. \quad (16)$$

where γ and δ are positive constants. Following the same EM approach used by NMF [35] and LNMF [40] techniques, the following update rules for the weight coefficients $h_{k,j}$ that belong to one of the two classes (existence or absence of a FAU) are derived from:

$$h_{k,j}^{(t)} = \frac{T_2 + \sqrt{T_2^2 + 4(2\gamma - (2\gamma + 2\delta)\frac{1}{N_k^{(i)}}) h_{k,j}^{(t-1)} \sum_i z_{i,k}^{(t-1)} \frac{x_{i,j}}{\sum_l z_{i,l}^{(t-1)} h_{l,j}^{(t-1)}}}}{2(2\gamma - (2\gamma + 2\delta)\frac{1}{N_k^{(i)}})}, \quad (17)$$

where T_2 is given by:

$$T_2 = (2\gamma + 2\delta)(\frac{1}{N_k^{(i)}} \sum_{\lambda, \lambda \neq l} h_{k,\lambda}) - 2\delta\mu_k - 1. \quad (18)$$

The update rules for the bases $\mathbf{Z}_D^k$ are the same as in NMF and can be given by (6) and (7). It is interesting to note here that the extracted DNMF bases are now class specific (different bases for each FAU).

In order to find if a FAU is activated in the facial differences image $\mathbf{x}$, the image is projected to the lower dimensional feature space derived from the DNMF algorithm. The distance used for the classification of the k-th FAU is given by:

$$u_{\mathbf{x}}^{k} = \min_{i=1,2} \|\mathbf{Z}_{D}^{k\,\dagger}(\mathbf{x}^{\delta} - \mathbf{m}_{k}^{(i)})\| \tag{19}$$

where $\mathbf{m}_{k}^{(1)}$ and $\mathbf{m}_{k}^{(2)}$ are the mean differences images of the first and second class (presence and absence of k-th FAU), respectively.

4 Shape information extraction and classification

4.1 Shape information extraction subsystem

The shape information extraction system is composed of two subsystems: one for Candide grid node information extraction and another one for grid node information classification. The grid node information extraction is performed by a tracking system. Candide node tracking is performed by a pyramidal variant of the well-known Kanade-Lucas-Tomasi (KLT) tracker [46]. The loss of tracked features is handled through a model deformation procedure that increases the robustness of the tracking algorithm. The algorithm, initially fits and subsequently tracks the Candide facial wireframe model in video sequences containing the formation of a dynamic human facial expression from the neutral state to the fully expressive one. The facial features are tracked in the video sequence using a variant of the KLT tracker [46]. If needed, model deformations are performed by mesh fitting at the intermediate steps of the tracking algorithm. Such deformations provide robustness against node losses and increase tracking accuracy. The algorithm automatically adjusts the grid to the face and then tracks it through the image sequence, as it evolves over time. The grid initialized in semi-automatic way. That is, elastic graph matching [47] is applied and afterwards some nodes that may have been misplaced are corrected manually. At the end, the grid tracking algorithm produces the deformed Candide grid that corresponds to the formed facial expression. A poser with the corresponding grid for the six basic facial expressions plus the neutral state is shown in Figure 6.

4.2 Facial expression recognition using shape information

The extracted grids are afterwards normalized. The normalization procedure ensures the common scaling, orientation and coordinates system, so that their

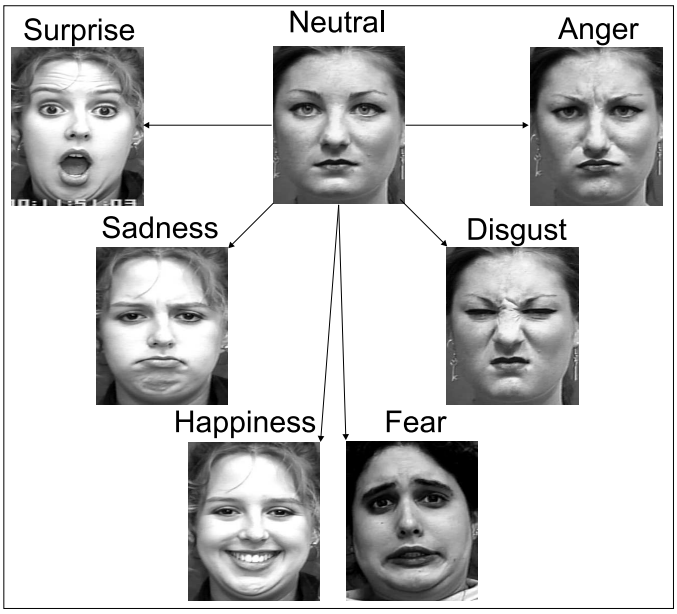


Fig. 6. An example of each facial expression for a poser from the Cohn-Kanade database.

comparison is feasible. The grids are initially moved so that the tip of the nose for every grid is the center of the coordinates system. Afterwards, their scaling is processed in such a way that the length and the width of the grid is constant. Finally, the angle that is defined using the horizontal line that joins the inner eyes corners and the vertical line that joins the center of the forehand with the tip of the nose, is checked so that is also common for all grids. The normalized grids are then used as an input to the shape extraction information subsystem where a metric-multidimensional scaling is performed in order to create a new Euclidean feature space. The projection of the input data on that new space is then used as an input to a SVM classifier for the classification of the shape information.

4.2.1 Metric-multidimensional scaling

Given two Candide grid point sets: $\mathcal{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_p\}$ and $\mathcal{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_p\}$, we propose the following metric in order to measure the similarity between deformed grids :

$$d_H(\mathcal{A}, \mathcal{B}) = \frac{1}{p} \sum_{a_i \in A, b_i \in B} \|\mathbf{a}_i - \mathbf{b}_i\|. \tag{20}$$

It can be easily proven that the proposed measure satisfies the following properties:

- reflectivity i.e., $d_H(\mathcal{A}_i, \mathcal{A}_i) = 0$
- positivity i.e., $d_H(\mathcal{A}_i, \mathcal{A}_j) > 0$ if $\mathcal{A}_i \neq \mathcal{A}_j$

- symmetry i.e., $d_H(\mathcal{A}_i, \mathcal{A}_j) = d(\mathcal{A}_j, \mathcal{A}_i)$
- triangle inequality i.e., $d_H(\mathcal{A}_i, \mathcal{A}_j) \leq d_H(\mathcal{A}_i, C) + d_H(C, \mathcal{A}_j)$ where A_i, A_j, C grids.

Thus, the proposed distance as a proper similarity measure [48]. We will use this similarity measure in order to define a metric multidimensional scaling [41]-[43].

Let $\{\mathcal{A}_1, \ldots, \mathcal{A}_N\}$ be the set of training facial grid database. The similarity matrix of the training is defined as:

$$[\mathbf{D}]_{i,j} = d_H(\mathcal{A}_i, \mathcal{A}_j). \tag{21}$$

We will use the dissimilarity matrix $\mathbf{D}$ in order to define an embedding $\mathbf{X} \in \Re^{k \times N}$, where $k \leq N$ is the dimensionality of the embedding and the i-th column of $\mathbf{X}$, denoted as $\mathbf{x}_i$, corresponds to the feature vector of the facial grid $\mathcal{A}_i$ in the new Euclidean space. In order to find the embedding $\mathbf{X}$, the matrix $\mathbf{B}$ is defined as:

$$\mathbf{B} = -\frac{1}{2}\mathbf{JDJ} \tag{22}$$

where $\mathbf{J} = \mathbf{I}_{N \times N} - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T \in \Re^{N \times N}$ is the centering matrix, where $\mathbf{I}_{N \times N}$ is the $N \times N$ identity matrix and $\mathbf{1}_N$ is the N-dimensional vector of ones. The matrix $\mathbf{J}$ projects the data so that the embedding $\mathbf{X}$ has zero mean. The eigen-decomposition of the matrix $\mathbf{B}$ will give us the desired embedding. The matrix $\mathbf{B}$ is positive semi-definite (i.e., it has real and non-negative eigenvalues), since the distance matrix $\mathbf{D}$ is Euclidean. Let p be the number of positive eigenvalues of matrix $\mathbf{B}$. Then, the matrix $\mathbf{B}$ can be written as:

$$\mathbf{B} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T = \mathbf{Q}\mathbf{\Lambda}^{\frac{1}{2}} \begin{bmatrix} \mathbf{M} & \\ & \mathbf{0} \end{bmatrix} \mathbf{\Lambda}^{\frac{1}{2}}\mathbf{Q}^T = \mathbf{G}^T\mathbf{M}\mathbf{G} \tag{23}$$

where $\mathbf{\Lambda}$ is a diagonal matrix with the diagonal consisting of the p positive eigenvalues, which are presented in the following order: first, positive eigenvalues in decreasing order and finally the zero values. The matrix $\mathbf{Q}$ is an orthogonal matrix of the corresponding eigenvectors. The matrix $\mathbf{M}$ is equal to $\mathbf{M} = \mathbf{I}_{p \times p}$ where $\mathbf{I}_{p \times p}$ is the identity $p \times p$ matrix. The matrix $\mathbf{G}$ is the embedding of the set of facial grids in the Euclidean space $\Re$ [48]:

$$\mathbf{G} = \mathbf{\Lambda}_k^{\frac{1}{2}}\mathbf{Q}_k^T \tag{24}$$

where $\mathbf{\Lambda}_k$ contains only the non-zero diagonal elements of $\mathbf{\Lambda}$ and $\mathbf{Q}_k$ is the matrix with the corresponding eigenvectors.

In this case, the new embedding is:

$$\mathbf{G}_\rho = \mathbf{\Lambda}_\rho^{\frac{1}{2}}\mathbf{Q}_\rho^T \tag{25}$$

where $\mathbf{\Lambda}_\rho$ is a diagonal matrix having as diagonal elements the magnitude of the diagonal elements of $\mathbf{\Lambda}_l$, in descending order. The matrix $\mathbf{Q}_\rho$ contains the corresponding eigenvectors. For the dimensionality ρ of the new embedding, the following inequality holds: $\rho \leq p \leq N$. As already mentioned, the vector $\mathbf{g}_i^l$, i.e. the i-th column of the matrix $\mathbf{G}_l$ corresponds to the feature vector of the grid $\mathcal{A}_i$ in the Euclidean space.

4.2.2 Multiclass Support Vector Machines in the new space

For every facial expressive grid $\mathcal{A}_i \in \Re^\rho$, a feature vector $\mathbf{g}_i^\rho$ is created. The feature vectors $\mathbf{g}_i^\rho$ labelled properly with the true corresponding facial expressions are used as an input to a multi-class SVM. SVMs were chosen due to their good performance in various practical pattern recognition applications [34][49]-[52] and their solid theoretical foundations. A brief presentation of the optimization problem of the multi-class SVMs will be given below. The interested reader can refer to [53]-[56] and the references therein for formulating and solving multi-class SVM optimization problems.

The training data are $(\mathbf{g}_1^\rho, l_1), \ldots, (\mathbf{g}_N^\rho, l_N)$, where $\mathbf{g}_j^\rho \in \Re^\rho$ $l_j \in \{1, \ldots, 7\}$ are the corresponding facial expression class labels. The multi-class SVM solves only one optimization problem [55]. It constructs 7 facial expressions rules, where the $k-$th function $\mathbf{w}_k^T \phi(\mathbf{g}_j) + b_k$ separates training vectors of the class k from the rest of the vectors, by minimizing the objective function:

$$\min_{\mathbf{w},\mathbf{b},\boldsymbol{\xi}} \qquad \frac{1}{2}\sum_{k=1}^{7} \mathbf{w}_k^T \mathbf{w}_k + C \sum_{j=1}^{N} \sum_{k \neq l_j} \xi_j^k \tag{26}$$

subject to the constraints:

$$\begin{aligned} \mathbf{w}_{l_j}^T \phi(\mathbf{g}_j) + b_{l_j} &\geq \mathbf{w}_k^T \phi(\mathbf{g}_j) + b_k + 2 - \xi_j^k \\ \xi_j^k \geq 0, \quad j = 1, \ldots, N, \quad k &\in \{1, \ldots, 7\} \backslash l_j. \end{aligned} \tag{27}$$

C is the term that penalizes the training errors. $\mathbf{b} = [b_1 \ldots b_7]^T$ and $\boldsymbol{\xi} = [\xi_1^1, \ldots, \xi_i^k, \ldots, \xi_N^7]^T$ are the bias and slack variable vectors, respectively. For the solution of the optimization problem (26), subject to the constraints (27), the reader can refer to [53,55,56].

The nonlinear mapping ϕ has been used for a high dimensional feature mapping for obtaining a linear SVM 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's condition [53,57]:

$$h(\mathbf{g}_i, \mathbf{g}_j) = \phi(\mathbf{g}_i)^T \phi(\mathbf{g}_j). \tag{28}$$

The functions used as SVM kernels were the d degree polynomial function:

$$h(\mathbf{g}_i, \mathbf{g}_j) = (\mathbf{g}_i{}^T \mathbf{g}_j + 1)^d \tag{29}$$

and the Radial Basis Function (RBF) kernel:

$$h(\mathbf{g}_i, \mathbf{g}_j) = \exp(-\gamma \parallel \mathbf{g}_i - \mathbf{g}_j \parallel^2). \tag{30}$$

where γ is the spread of the Gaussian function.

The decision function is:

$$p(\mathbf{g}) = \underset{k=1,\ldots,7}{\operatorname{argmax}}(\mathbf{w}_k^T \phi(\mathbf{g}) + b_k). \tag{31}$$

The question that remains is how novel grids can be classified using the proposed embedding and multiclass SVM.

4.2.3 Classifying Novel Grids

For testing, let $\{\mathcal{G}_1, \ldots, \mathcal{G}_n\}$ be a set of n testing (novel) facial grids. The matrix $\mathbf{D}_n \in \Re^{n \times N}$ is created, with $[\mathbf{D}_n]_{i,j} = d_H(\mathcal{G}_i, \mathcal{A}_j)$. The matrix $\mathbf{D}_n$ represents the similarity between the n test facial grids and all the training facial grids. The matrix $\mathbf{B}_n \in \Re^{n \times N}$ of inner products that relates all the new (test) facial grids to all facial grids from the training set is then found as follows:

$$\mathbf{B}_n = -\frac{1}{2}(\mathbf{D}_n \mathbf{J} - \mathbf{U}\mathbf{D}\mathbf{J}) \tag{32}$$

where $\mathbf{J}$ is the centering matrix and $\mathbf{U} = \frac{1}{N}\mathbf{1}_n \mathbf{1}_N^T \in \Re^{n \times N}$. The embedding $\mathbf{G}_n \in \Re^{\rho \times n}$ of the test facial grids is defined as:

$$\mathbf{G}_n = \mathbf{\Delta}_\rho^{-\frac{1}{2}} \mathbf{Q}_\rho^T \mathbf{B}_n^T. \tag{33}$$

The columns of the matrix $\mathbf{G}_n$ are the features used for classification. Let $\mathbf{g}_{i,n} \in \Re^\rho$ be the i-th column of the matrix $\mathbf{G}_n$, i.e. the vector that contains the features of the grid $\mathcal{G}_i$. A test grid deformation feature vector is classified to one of the seven facial expressions using (34). Once the seven-class SVM system is trained, it can be used for testing, i.e., for recognizing facial expressions on new facial videos. $\mathcal{G}_i$ to one of the seven facial expression classes is performed by the decision function:

$$f(\mathcal{G}_i) = \arg \max_{k=1,\ldots,7}(\mathbf{w}_k^T \phi(\mathbf{g}_{i,n}) + b_k), \tag{34}$$

where $\mathbf{w}_k$ and b_k have been found during training. The distance that defines the facial expression class the grid deformation vector belongs to is given by:

$$s_{\mathbf{g}} = \max_{k=1,\ldots,7}(\mathbf{w}_k^T \phi(\mathbf{g}) + b_k) \tag{35}$$

which is the distance from the class separating hyperplane.

4.3 FAU recognition using shape information

For FAU recognition, the shape information produced from the j-th video sequence is the Candide node displacements $\mathbf{d}_j^i$ of the Candide grid nodes, defined as the difference between coordinates of this node in the neutral and expressive frame [34]:

$$\mathbf{d}_j^i = [\Delta x_j^i \;\; \Delta y_j^i]^T \quad i \in \{1, \ldots, 104\}. \tag{36}$$

where $\Delta x_{i,j}$, $\Delta y_{i,j}$ are the x, y coordinate displacement of the i-th node in the j-th image respectively. This way, for every facial image sequence in the training set, a feature vector $\mathbf{g}_j$ is created, called *grid deformation feature vector* containing the geometrical displacement of every grid node:

$$\mathbf{g}_j^\delta = [\mathbf{d}_{1,j} \quad \mathbf{d}_{2,j} \; \ldots \; \mathbf{d}_{E,j}]^T, \;\; j = 1, \ldots, N \tag{37}$$

having $Q = 104 \cdot 2 = 208$ dimensions. We assume that each grid deformation feature vector $\mathbf{g}_j^\delta \;\; j = 1, \ldots, N$.

Let $\mathcal{V}$ be the database that is consisted of the differences of grids between the neutral and expressive states as extracted from the video sequences. For the k-th FAU recognition, the database is clustered into 2 different classes $\{\mathcal{V}_k^{(1)}, \mathcal{V}_k^{(2)}\}$ each one representing one possible k-th FAU state (presence or absence). The grid deformation feature vector $\mathbf{g}_j^\delta \in \Re^Q$ is used as an input to 17 two class SVM systems, each one detecting a specific FAU (the FAU set includes FAUs 1, 2, 4, 5, 6, 7, 9, 10, 12, 15, 16, 17, 20, 23, 24, 25 and 26). Each SVM system, uses the Candide node geometrical displacements to decide whether a specific FAU is activated for the test grid under examination or not. The k-th SVM, $k = 1, \ldots, 17$ is trained with the examples in $\mathcal{V}_k^{(1)} = \{(\mathbf{g}_j^\delta, y_j^k), \quad j = 1, \ldots, N, \quad y_j^k = 1\}$ as positive ones and all other examples $\mathcal{V}_k^{(2)} = \{(\mathbf{g}_j, y_j^k), \quad j = 1, \ldots, N, \quad y_j^k = -1\}$ as negative ones. The feature vectors $\mathbf{g}_j \in \Re^Q$ labelled properly with the correct label ($l_j = 1$ when the FAU under examination is activated and $l_j = -1$ when it is not activated) are used as an input to a set of two-class SVM systems.

Two class SVM systems are used in order to detect the activated FAUs. The grid deformation feature vector $\mathbf{g}_j^\delta \in \Re^Q \;\; j = 1, \ldots, N$ is used as an input to 17 two-class SVM systems, each one detecting a specific FAU from the ones depicted in Figure 4. Each SVM system uses the grid node geometrical displacements to decide whether a specific FAU is activated at the grid under examination or not. In order to train the k-th SVMs network, the following

minimization problem has to be solved [54]:

$$\min_{\mathbf{w}_k, b_k, \boldsymbol{\xi}^k} \quad \frac{1}{2}\mathbf{w}_k^T\mathbf{w}_k + C_k \sum_{j=1}^{N} \xi_j^k \tag{38}$$

subject to the separability constraints:

$$\begin{aligned} y_i^k(\mathbf{w}_k^T\phi(\mathbf{g}_j^\delta) + b_k) &\geq 1 - \xi_j^k, \\ \xi_j^k &\geq 0, \quad j = 1, \ldots, N, \end{aligned} \tag{39}$$

where b_k is the bias for the k-th SVM, $\boldsymbol{\xi}^k = [\xi_1^k, \ldots, \xi_N^k]$ is the slack variable vector and C_k is the term that penalizes the training errors.

After solving the optimization problem (38) subject to the separability constraints (39) [53,57], the function that decides whether the k-th FAU is activated by a test displacement feature vector $\mathbf{g}^\delta$ is:

$$f_k(\mathbf{g}) = \text{sign}(\mathbf{w}_k^T\phi(\mathbf{g}^\delta) + b_k). \tag{40}$$

The distance of the grid to the decision surface of the k-th FAU ($k = 1, \ldots, 17$) in the test video that produced the grid deformation vector is :

$$v_{\mathbf{g}}^k = \mathbf{w}_k^T\phi(\mathbf{g}^\delta) + b_k. \tag{41}$$

5 Fusion of texture and shape information

5.1 Fusion for facial expression recognition

Various methods were used in our study to achieve fusion of the texture and shape information results (SVMs and many variations of NNs). However, only the one that provided the best results (MRBF NN) will be described below, due to space limitations. The DNMF algorithm, applied to the image $\mathbf{x}$, produces the distance $r_{\mathbf{x}}$ as a result, while SVMs applied to the Candide grid $\mathbf{g}$, produce the distance $s_{\mathbf{g}}$ as the equivalent result. The distances $r_{\mathbf{x}}$ and $s_{\mathbf{g}}$, defined in (13) and in (34), respectively, were normalized in $[0, 1]$ using Gaussian normalization [58]. Thus, a new feature vector $\mathbf{c}$, $\mathbf{c} = [r_{\mathbf{x}} \quad s_{\mathbf{g}}]^T$, is defined containing information from both texture and shape information sources. The feature vector $\mathbf{c}$ was used as an input to a RBF NN system. The output of this system is the facial expression class label l. Many variations of RBF NNs

were tested in our experiments, such as the general RBF NNs, Generalized Regression NNs (GRNNs) and MRBFs.

5.2 Fusion for FAU recognition

The DNMF algorithm, applied to the difference image $\mathbf{x}^\delta$, produces a score $u^k_{\mathbf{x}}$ (defined in (19)) as a result, which specifies whether the k-th FAU examination was activated in the image $\mathbf{x}^\delta$. The SVM application to the vector of geometrical displacements $\mathbf{g}^\delta$, produces the score $v^k_{\mathbf{g}^\delta}$ (defined in (41)) as the equivalent result. A new feature vector $\mathbf{c}^k = [u^k_{\mathbf{x}} \quad v^k_{\mathbf{g}^\delta}]^T$ is defined containing information from both texture and shape information sources was created. The feature vector $\mathbf{c}$ was used as an input to a MRBF NNs system to produce the final decision on FAU recognition.

5.3 Median Radial Basis Function Neural Networks for Fusion

In this Section, we shall describe the best solution found in our experiments for fusing the scores of the texture and shape classifiers. The best solution has been an RBF NN [59] based on robust statistics, the so-called MRBF. The use of the MRBF for fusing the scores has been motivated by its successful application in fusing the scores of various modalities in the person identification problem [60].

An RBF network is a two-layer feed-forward neural network, in which various clusters are grouped together in order to describe classes, thus making it appropriate for nonlinear functional approximation [61]. The inputs of the RBF network are the previously described vectors $\mathbf{c}$. Each hidden unit implements a Gaussian function which models a cluster:

$$\phi_j(\mathbf{c}) = exp[-(\mathbf{c} - \mathbf{p}_j)^T \mathbf{S}_j^{-1} (\mathbf{c} - \mathbf{p}_j)] \tag{42}$$

where $\mathbf{c}$ is the entry vector, $\mathbf{p}_j$ is the mean vector, $\mathbf{S}_j$ is the covariance matrix, and $j = 1, \ldots, L$, where L is the total number of hidden units. Each hidden unit models the location and the spread of a cluster. The output unit consists of a weighted sum of hidden unit outputs, which are fed into a sigmoidal function:

$$\psi(\mathbf{c}) = \frac{1}{1 + exp[-\sum_{j=1}^{P} \lambda_j \phi_j(\mathbf{c})]} \tag{43}$$

where λ_j are the output weights associated with the hidden units. The output consists of a decision function $\psi(\mathbf{c}) \in (0, 1)$.

A very common approach for estimating the parameters of an RBF network consists of an adaptive implementation of the $k-$means clustering algorithm [62]. Another approach is to use hybrid SVMs plus a RBF system, where the centers of the classes are estimated using initially a SVM system (i.e., we use as the RBF centers the learned SVs (Support Vectors) [63]). In [59], a robust statistics algorithm was proposed for estimating the parameters of the RBF networks. It was proven that this algorithm provides better parameter estimates when the clusters are overlapping or in the presence of outliers [59]. MRBF assigns an incoming data vector to a cluster which has the smallest Euclidean distance:

$$\|\mathbf{c}_i - \mathbf{p}_j\| = \min_k \|\mathbf{c}_i - \mathbf{p}_k\|. \tag{44}$$

After assigning a set of vectors to the same cluster, we calculate the center of the cluster using the marginal median algorithm

$$\mathbf{p}_j = \text{med}\{\mathbf{c}_{j,0}, \mathbf{c}_{j,1}, ..., \mathbf{c}_{j,n}\} \tag{45}$$

where $\mathbf{c}_{j,i}$ for $i = 0, \ldots, n$ are the data samples assigned to the hidden unit j. In order to limit the computational complexity, we consider only a limited set of data samples and the formula (45) is calculated from a running window. For the dispersion estimation we employ the median of the absolute deviations from the median algorithm:

$$\mathbf{S}_j = \frac{\text{med}\{|\mathbf{c}_{j,0} - \mathbf{p}_j|, \ldots, |\mathbf{c}_{j,n} - \mathbf{p}_j|\}}{0.6745}. \tag{46}$$

The covariance matrix $\mathbf{S}_j$ is considered to be diagonal. The output weights are calculated from the back-propagation algorithm:

$$\lambda_j = \sum_{i=0}^{n} [H(\mathbf{c}_i) - \psi(\mathbf{c}_i)]\psi(\mathbf{c}_i)[1 - \psi(\mathbf{c}_i)]\phi_j(\mathbf{c}_i) \tag{47}$$

where $H(\mathbf{c}_i)$ is the decision function associated with each data sample in the training set (i.e, $H(\mathbf{c}_i)$ is the label of $\mathbf{c}_i$).

MRBF networks use the second order statistics. The radial basis functions modelling the clusters are not influenced by the presence of outliers in the MRBF training algorithm, due to the use of the robust median operators [64]. Therefore, MRBF networks are expected to have good classification performance.

6 Experimental results

6.1 Database description

The Cohn-Kanade database has been used in the experiments. This database is annotated with FAUs. These combinations of FAUs were translated into facial expressions according to [5], in order to define the corresponding ground truth for the facial expressions. All the available subjects and videos were taken under consideration to form the database for the experiments.

The most frequently used approach for testing the generalization performance of a classifier is the leave-one cross-validation approach [65]. It was devised in order to make maximal use of the available data and produce averaged classification accuracy results. The term leave-one out cross-validation does not correspond to the classical leave-one-out definition, as a variant of leave-one-out was used (i.e., leave 20% of the samples out) for the formation of the test dataset in our experiments. However, the procedure followed will be called leave-one-out from now on for notation simplicity without loss of generalization. More specifically, all image sequences contained in the database are divided into 7 facial expression classes (or 17 FAU classes). Five sets containing 20% of the data for each class, chosen randomly, were created. One set containing 20% of the samples for each class is used as the test set, while the remaining sets form the training set. After the classification procedure is performed, the samples forming the test set are incorporated into the current training set, and a new set of samples (20% of the samples for each class) is extracted to form the new test set. The remaining samples create the new training set. This procedure is repeated five times. A diagram of the leave-one-out cross-validation method can be seen in Figure 7. The average classification accuracy is defined as the mean value of the percentages of the correctly classified facial expressions over all data presentations.

The accuracy achieved for each facial expression is averaged over all facial expressions and does not provide any information with respect to a particular expression. The confusion matrices [34] have been computed to handle this problem. The confusion matrix is a $n \times n$ matrix 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 number of facial expressions that are correctly classified, while the off-diagonal entries correspond to misclassifications. The abbreviations an, di, fe, ha, sa, su and ne represent anger, disgust, fear, happiness, sadness, surprise and neutral, respectively.

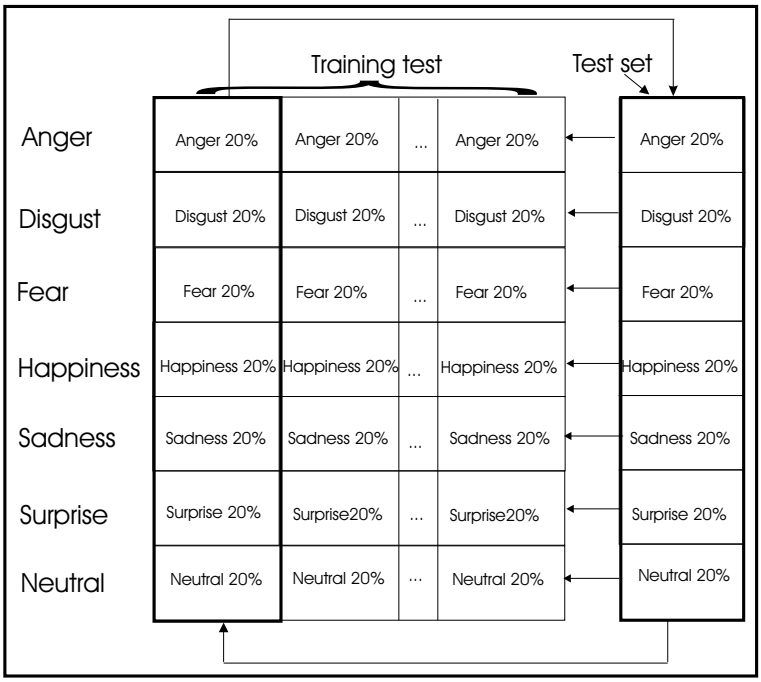


Fig. 7. Diagram of leave-one-out method used in classification assessment.

6.2 Facial expression recognition

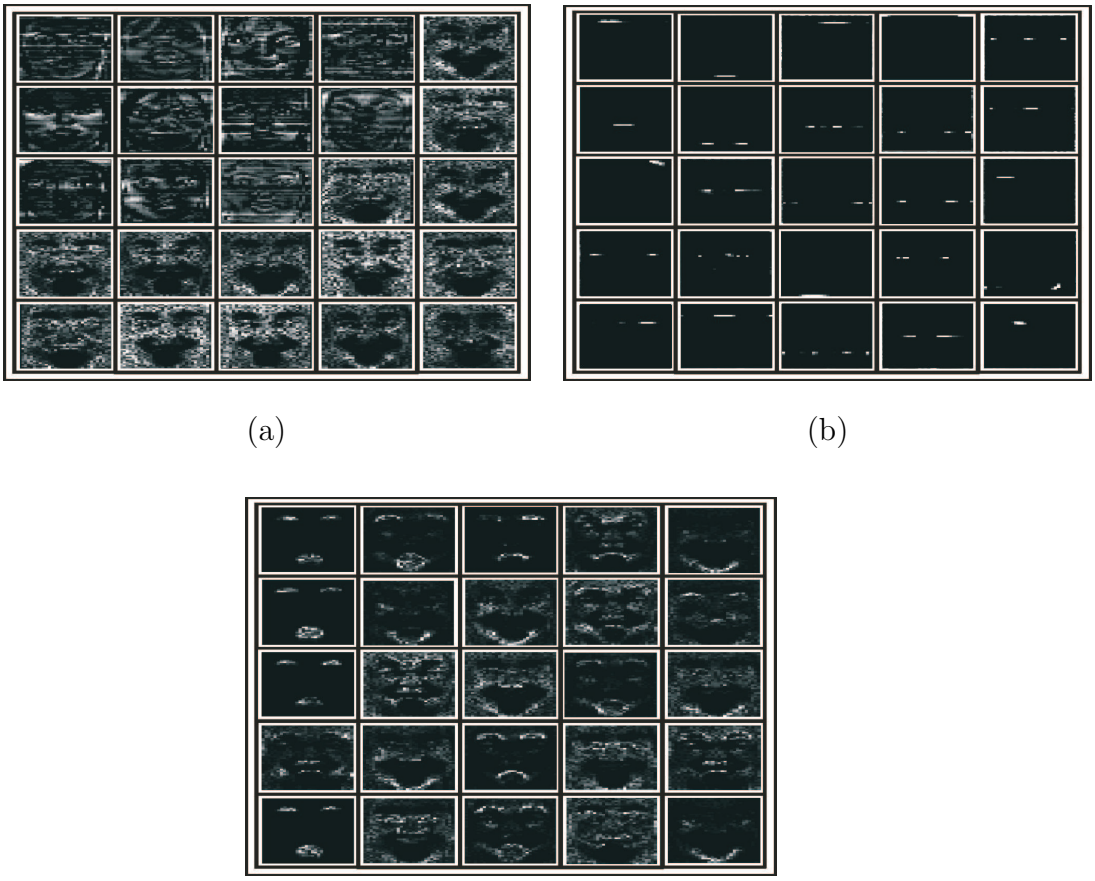
In this Section, facial expression recognition experiments are described. The facial expressions under examination are the six basic ones plus the neutral state. Only the best accuracies achieved for any method used are taken under consideration to make the final conclusions.

6.2.1 Facial expression recognition from texture

The basis images extracted when the NMF, LNMF and DNMF algorithms were applied are depicted in Figure 8. The accuracy rates obtained for facial expression recognition using texture information and applying several methods, such as PCA, PCA followed by LDA, NMF, LNMF and DNMF, are shown in Figure 9. DNMF clearly outperforms the rest image representations. The number of dimensions kept after applying PCA plus LDA, were equal to the number of facial expression classes minus 1, thus equal to 6. The confusion matrix obtained when using DNMF on texture information is presented in Table 2.a. The best accuracy achieved was equal to 74.3%.

As can be seen from the confusion matrix, sadness seems to be the most ambiguous facial expression. More specifically, it is misclassified the most as neutral and anger (23.3% and 7.7% of the cases, respectively). The facial expression that follows in misclassification rate is fear, which is mainly confused

with neutral (21% of the cases). The facial expression misclassification descending ordering continues with happiness (misclassified as neutral in 15.6% of the cases), disgust (misclassified as neutral in 20% of the cases), surprise (misclassified as neutral in 18.6% of the cases), anger (misclassified as neutral in 14% of the cases) and neutral (misclassified as anger in 5.5% of the cases).



(a) (b)

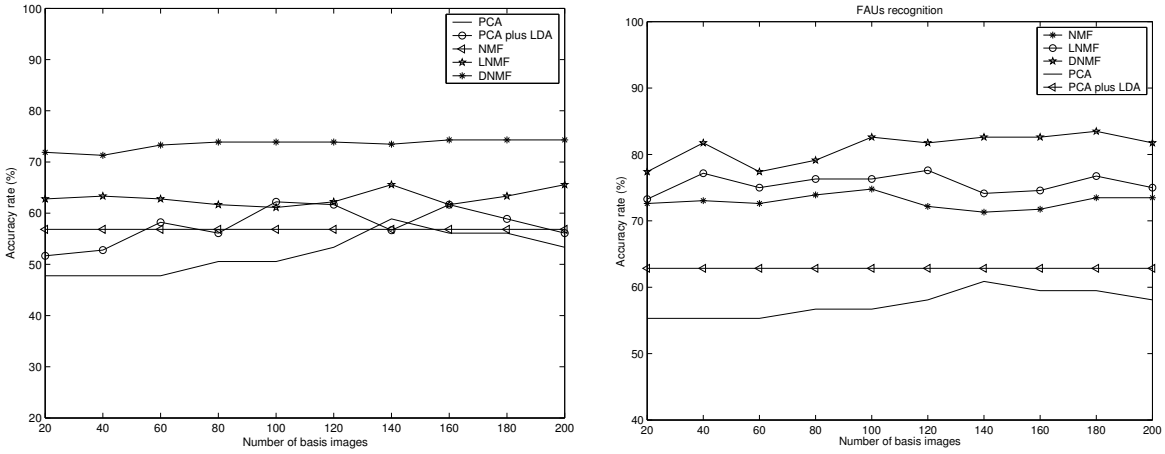
(c)

Fig. 8. Basis images extracted for (a) NMF, (b) LNMF and (c) DNMF algorithms.

6.2.2 Shape information extraction using SVMs

The confusion matrix obtained when using SVMs on shape information using the method described in Section 4, is presented in Table 2.b. The accuracy achieved was equal to 84.8%. In Figure 10, the accuracy rates achieved for facial expression recognition when using SVMs with polynomial and RBF kernels are shown.

As can be seen from the confusion matrix, fear seems to be the most ambiguous facial expression. More specifically, fear is misclassified the most as happiness, followed by disgust and neutral (11.8%, 7.3% and 7.2% of the cases, respec-



(a) (b)

Fig. 9. Recognition accuracies obtained for (a) facial expression and (b) FAU recognition using NMF, LNMF and DNMF algorithms.

tively). The facial expression that follows in misclassification rate is sadness, which is mainly confused with anger (10.8% of the cases). The facial expression misclassification descending ordering continues with neutral (misclassified as surprise in 5.8% of the cases), disgust (misclassified as anger in 14.3% of the cases), anger and happiness (misclassified as disgust and neutral in 6% and 9% of the cases, respectively) and surprise (misclassified as fear in 7.1% of the cases).

Table 2
Confusion matrices when using (a) texture (74.3%) and (b) shape (84.8%)information, respectively.

$lab_{ac}\% \backslash {}^{lab_{cl}\%}$	an	di	fe	ha	sa	su	ne
an	77	0	0	5.6	7.7	1.4	5.5
di	0	74	3.6	3.3	1.5	0	0
fe	3	0	68.2	2.2	1.5	1.5	1.1
ha	0	3	3.6	73.3	1.5	1.4	1.1
sa	6	0	0	0	61.5	1.4	0
su	0	3	3.6	0	3	75.7	2.2
ne	14	20	21	15.6	23.3	18.6	90.1

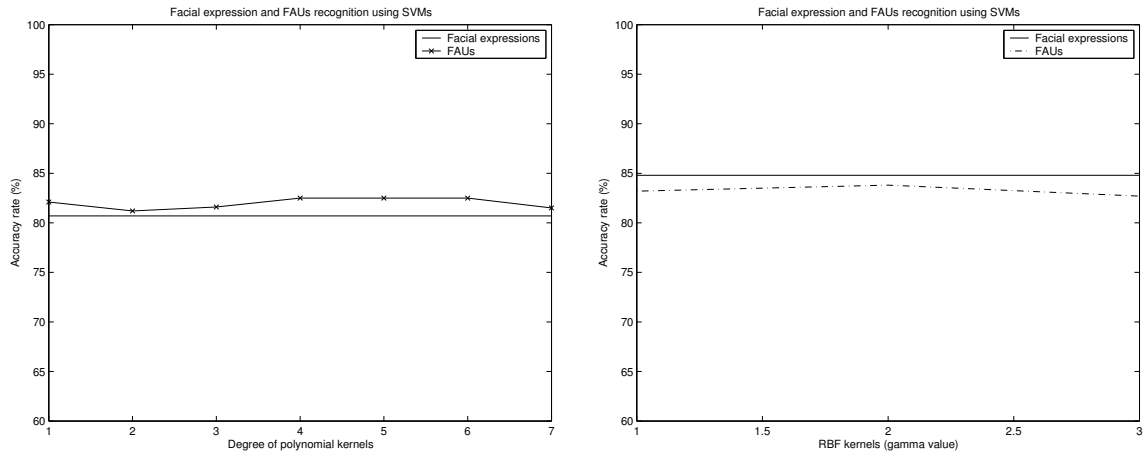
(a)

$lab_{ac}\% \backslash {}^{lab_{cl}\%}$	an	di	fe	ha	sa	su	ne
an	91	14.3	0	0	10.8	0	4.8
di	6	85.7	7.3	0	0	0	0
fe	0	0	68.2	0	0	7.1	2.4
ha	0	0	11.8	91	4.6	0	0
sa	0	0	5.5	0	80	0	2.4
su	3	0	0	0	0	92.9	5.8
ne	0	0	7.2	9	4.6	0	84.6

(b)

6.2.3 Fusion of texture and shape information for facial expression recognition

The confusion matrix obtained when fusion using MRBF NNs is presented in Table 3. The accuracy achieved when MRBF NNs were used for the fusion of the texture and shape results, was equal to 92.3%, which is better than using either texture or shape information alone. The combination of texture and



(a) (b)

Fig. 10. Facial expression and FAU recognition accuracies using shape and SVMs for various kernels (a) polynomial kernels (b) RBF kernels.

shape information increases the classification rate for all facial expressions. More specifically:

- For anger, the final accuracy achieved when fusion is applied is equal to 93.6%, while the equivalent ones before fusion were 77% and 91% for texture and shape information classification, respectively. The confusion of anger with fear and neutral that appears when only texture information is used no longer exists, while the confusion of anger with sadness remains but is significantly reduced when fusion is introduced. Regarding shape information, the confusion of anger with surprise no longer exists when fusion is introduced, while the confusion of anger with disgust remains but is significantly reduced.
- For disgust, the final accuracy achieved when fusion is applied is equal to 89.5%, while the equivalent ones before fusion were 74% and 85.7% for texture and shape information classification, respectively. The confusion of disgust with surprise and neutral that appears when only texture information is used no longer exists. Regarding shape information, the confusion of disgust with anger no longer exists when fusion is introduced.
- For fear, the final accuracy achieved when fusion is applied is equal to 84.3%, while the equivalent ones before fusion were equal to 68.2% both for texture and shape information classification. The confusion of fear with disgust, happiness, surprise and neutral that appears when only texture information is used no longer exists. Regarding shape information, the confusion of fear with disgust, happiness and neutral no longer exists when fusion is introduced, while the confusion of disgust with sadness remains but is significantly reduced.
- For happiness, the final accuracy achieved when fusion is applied is equal to 97.5%, while the equivalent ones before fusion were 73.3% and 91% for

texture and shape information classification. The confusion of happiness with anger, disgust and fear that appears when only texture information is used no longer exists, while the confusion of happiness with neutral remains but is significantly reduced. Regarding shape information, the confusion of happiness with neutral remains but is significantly reduced.

- For sadness, the final accuracy achieved when fusion is applied is equal to 94.3%, while the equivalent ones before fusion were 61.5% and 80% for texture and shape information classification. The confusion of sadness with anger, disgust, fear, surprise and neutral that appears when only texture information is used no longer exists. Regarding shape information, the confusion of sadness with anger and neutral is now absent.
- For surprise, the final accuracy achieved when fusion is applied is equal to 95.6%, while the equivalent ones before fusion were 75.7% and 92.9% for texture and shape information classification. The confusion of surprise with anger, fear and happiness that appears when only texture information is used no longer exists, while the confusion of surprise with neutral remains but is significantly reduced. Regarding shape information, the confusion of surprise with fear is now absent.
- For neutral, the final accuracy achieved when fusion is applied is equal to 91.3%, while the equivalent ones before fusion were 90.1% and 84.6% for texture and shape information classification. The confusion of neutral with fear and happiness that appears when only texture information is used no longer exists. Regarding shape information, the confusion of neutral with anger, fear and sadness is now absent, while the confusion of neutral with surprise remains but is significantly reduced.

As can be seen from the confusion matrix (Table 3), all facial expressions are correctly recognized in more cases when texture and shape information are used. This is due to the fact that all facial expressions depend to a great extent on the posers' expressive ability. For example, anger can appear only with a gaze change rather than the equivalent mouth movement, something that can only be detected by the human eye (therefore being visible as a change in texture information), while disgust includes a frown that can not be perfectly represented by the Candide grid due to the lack of enough grid vertices that should be placed at the wider nose area. Fear can include extremely minor facial movements in combination with gaze changes, thus making it difficult to recognize (also being visible as changes in texture information). Sadness may be expressed as a difference in gaze and a subtle mouth movement and of course neutral does not include any movement at all. Thus all of the above mentioned facial expressions are greatly affected by the presence of texture information when it comes to their recognition. The remaining facial expressions (happiness and surprise) include more important changes in the form of facial movements. Their existence however results in major texture changes, e.g. when a person smiles a white area corresponding to his teeth appears, while when a person is surprised and opens his mouth a big black area appears. Thus, the recognition

of happiness and surprise can be also improved when texture information is available.

Table 3
Confusion matrix achieved fusing texture and shape information using MRBF NNs for seven facial expressions. The facial expression recognition rate has been 92.3%.

$lab_{ac}(\%) \backslash lab_{cl}(\%)$	an	di	fe	ha	sa	su	ne
an	93.6	0	0	0	0	0	6.7
di	1.6	89.5	0	0	0	0	0
fe	0	0	84.3	0	0	0	0
ha	2.6	10.5	0	97.5	5.7	0	0
sa	2.2	0	15.7	0	94.3	2.5	0
su	0	0	0	0	0	95.6	2.0
ne	0	0	0	2.5	0	1.9	91.3

A comparison of the recognition rates achieved for each facial expression with the state of the art [65]-[68], when six facial expression were examined (the neutral state was not taken under consideration) is depicted in Figure 11. The total facial expression recognition of the proposed fused architecture has been 94.5% for the six facial expressions. Unfortunately, their is no direct method to compare the rates achieved by other researchers [65]-[68], since their is not standard protocol (every one use his own testing protocol). Moreover, some of the methods like [66–68] have been tested only of the six facial expressions therefore, the performance of these methods in case the seventh facial expression (ie.neutral) had been included remains unknown. Only the method in [65] has been tested for the seven facial expressions and their recognition rate has been 78.52% which is significantly lower than the performance of the proposed method that achieved 91.3% for neutral.

6.3 FAU recognition

In this Section, FAU recognition is described. We expected the FAUs 1, 2, 4, 5, 6, 7, 9, 10, 12, 15, 16, 17, 20, 23, 24, 25 and 26, as proposed in the facial expression recognition rules in [5] (17 FAUs in total).

6.3.1 FAU recognition using texture information

The accuracy rates obtained for FAU recognition using texture information and by applying several methods, such as PCA, PCA followed by LDA, NMF, LNMF and DNMF are shown in Figure 9.b. Only one dimension was kept after applying PCA and LDA, this number being equal to the number of classification classes (presence or absence of a FAU) minus 1. Only the DNMF method that provided the best accuracies is taken in consideration for the fusion experiments. The total classification accuracy achieved was equal to 84.4%.

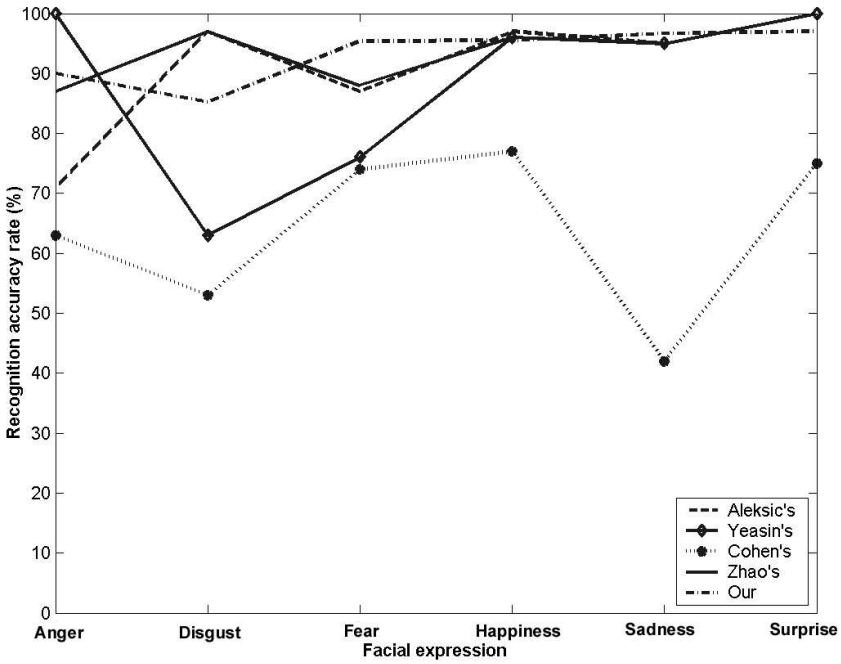


Fig. 11. Comparison with recent dynamic facial expression recognition methods.

6.3.2 FAU recognition using shape information

The total accuracy achieved was equal to 86.7%. In Figure 10, the accuracy rates achieved for FAU recognition when using SVMs are shown. The functions used as SVM kernels were the polynomial and RBF functions.

6.3.3 Fusion of texture and shape information for FAU recognition

The total accuracy achieved for both cases was equal to 92.1%, which is significantly better than the one obtained when using either texture or shape information. The accuracy rate was increased due to the use of both texture and shape information. The introduction of texture eliminates some of the confusions observed when using shape information only. This happens as in many FAUs, the shape information is not enough to fully describe its presence. In many cases, the available grid nodes fail to describe all possible texture characteristics, such as furrows and wrinkles that may appear on the face. To be more specific, when FAU 12 is observed (see Figure 12), some vertical furrows appear between the nose and the corners of the mouth (emphasized with a cloud of black dots). These furrows cannot be fully described by the Candide grid deformation due to the absence of properly placed grid nodes. The same happens with FAU 23 (also shown in Figure 12), where horizontal furrows appear between the chin and mouth (emphasized with a cloud of black dots).

Texture can capture all the necessary information where the shape description would fail, thus making the fusion of the two kinds of information more powerful. For FAU 9, the accuracy rate achieved when using texture information was equal to 86.4%, the equivalent one when using shape information was 91.7%. Fusion produced an accuracy of 95.8%. The proposed method increased the accuracy by more than 12% when compared to the accuracy achieved when only shape information is used (82.7%) [34].

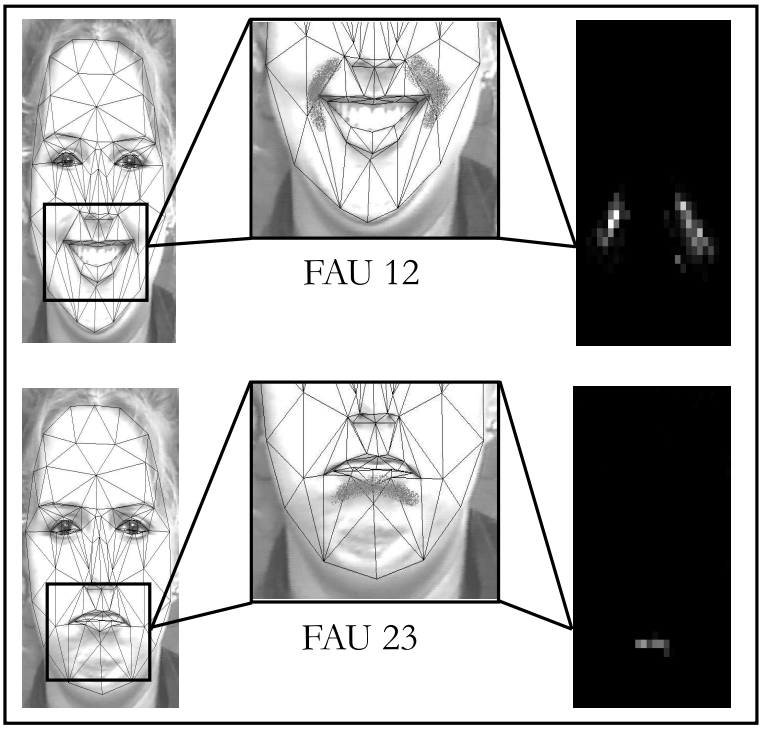


Fig. 12. Furrows that appear when FAUs 12 and 23 are observed and two of the sparse DNMF bases that correspond to the furrows.

7 Conclusions

A novel and complete (i.e., uses both shape and texture information) method for facial expression recognition is proposed in this paper. The recognition is performed by fusing the texture and the shape information extracted from a video sequence using a subspace representation method and an Euclidean embedding in combination with a SVMs system, respectively. The results obtained from the above mentioned methods are then fused. Various methods are used for fusion, including SVMs and MRBF. The system achieves an accuracy of 92.3% when recognizing the seven basic facial expressions and 92.1% when recognizing the 17 basic FAUs. Conclusions regarding the most misclassified

facial expressions are drawn and the way fusion aids to their easier and most accurate recognition is indicated.

8 Acknowledgment

This work was supported by the "SIMILAR" European Network of Excellence on Multimodal Interfaces of the IST Programme of the European Union (www.similar.cc) for Ms. Kotsia and by project 03ED849 co-funded by the European Union and the Greek Secretariat of Research and Technology (Hellenic Ministry of Development) of the Operational Program for Competitiveness within the 3rd Community Support Framework for Mr. Zafeiriou.

References

[1] A. Pentland, T. Choudhury, Face recognition for smart environments, IEEE Computer 33 (2) (2000) 50–55.

[2] M. Pantic, L. Rothkrantz, Toward an affect-sensitive multimodal human-computer interaction, Proceedings of the IEEE 91 (9) (2003) 1370–1390.

[3] P. Ekman, W. V. Friesen, Emotion in the Human Face, Prentice Hall, New Jersey, 1975.

[4] T. Kanade, J. Cohn, Y. Tian, Comprehensive database for facial expression analysis, in: Proceedings of IEEE International Conference on Face and Gesture Recognition, 2000, pp. 46–53.

[5] M. Pantic, L. J. M. Rothkrantz, Expert system for automatic analysis of facial expressions, Image and Vision Computing 18 (11) (2000) 881–905.

[6] M. Pantic, L. J. M. Rothkrantz, Automatic analysis of facial expressions: The state of the art, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (12) (2000) 1424–1445.

[7] B. Fasel, J. Luettin, Automatic facial expression analysis: A survey, Pattern Recognition 36 (1) (2003) 259–275.

[8] H. Gu, Y. Zhang, Q. Ji, Task oriented facial behavior recognition with selective sensing, CVIU 100 (3) (2005) 385–415.

[9] M. S. Bartlett, G. Littlewort, I. Fasel, J. R. Movellan, Real time face detection and facial expression recognition: Development and applications to human computer interaction, in: Proceedings of Conference on Computer Vision and Pattern Recognition Workshop, Vol. 5, Madison, Wisconsin, 2003, pp. 53–58.

[10] G. Littlewort, M. Bartlett, I. Fasel, J. Susskind, J. Movellan, Dynamics of facial expression extracted automatically from video, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Workshop on Face Processing in Video, 2004.

[11] C. Shan, S. Gong, P. McOwan, Appearance manifold of facial expression, in: "IEEE International Workshop on Human-Computer Interaction, 2005.

[12] Y. Kosaka, K. Kotani, Facial expression analysis by kernel eigenspace method based on class features (kemc) using non-linear basis for separation of expression-classes, in: Proceedings of IEEE International Conference on Image Processing (ICIP 2004), Singapore, 2004.

[13] S. Dubuisson, F. Davoine, M. Masson, A solution for facial expression representation and recognition, Signal Processing: Image Communication 17 (9) (2002) 657–673.

[14] C. Thomaz, D. G. A. Feitosa, Using mixture covariance matrices to improve face and facial expression recognitions, Pattern Recognition Letters 24 (13) (2003) 2159 – 2165.

[15] L. Ma, K. Khorasani, Facial expression recognition using constructive feedforward neural networks, IEEE Transactions on Systems, Man, And Cybernetics-Part B: Cybernetics 34 (3) (2004) 1588–1595.

[16] Y. Gizatdinova, V. Surakka, Feature-based detection of facial landmarks from neutral and expressive facial images, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (1) (2006) 135–139.

[17] M. Matsugu, K. Mori, Y.Mitari, Y. Kaneda, Subject independent facial expression recognition with robust face detection using a convolutional neural network, Neural Networks 16 (5-6) (2003) 555–559.

[18] D. Liang, J. Yang, Z. Zheng, Y. Chang, A facial expression recognition system based on supervised local linear embedding, Pattern Recognition Letters 26 (2005) 2374–2389.

[19] T. Yabui, Y. Kenmochi, K. Kotani, Facial expression analysis from 3d range images; comparison with the analysis from 2d images and their integration, in: Proceedings of IEEE International Conference on Image Processing, 2003, pp. 879–882.

[20] X.-W. Chen, T. Huang, Facial expression recognition: A clustering-based approach, Pattern Recognition Letters 24 (9-10) (2003) 1295–1302.

[21] Y. L. Tian, T. Kanade, J. Cohn, Evaluation of Gabor wavelet-based Facial Action Unit recognition in image sequences of increasing complexity, in: Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, 2002, pp. 229–234.

[22] S. Roweis, L. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323– 2326.

[23] Y. Gao, M. Leung, S. Hui, M. Tananda, Facial expression recognition from line-based caricatures, IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans 33 (3) (2003) 407–412.

[24] Y.Zhang, Q.Ji, Active and dynamic information fusion for facial expression understanding from image sequences, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (5) (2005) 699–714.

[25] P. Michel, R. Kaliouby, Real time facial expression recognition in video using support vector machines, in: Proceedings of 5th international conference on Multimodal interfaces, Vancouver, British Columbia, Canada, 2003, pp. 258–264.

[26] F. Dornaika, F. Davoine, View- and texture-independent facial expression recognition in videos using dynamic programming, in: Proceedings of IEEE International Conference on Image Processing, Genova, Italy, 2005.

[27] F. Dornaika, F. Davoine, Online appearance-based face and facial feature tracking, in: Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, U.K., 2004.

[28] B. Abboud, F. Davoine, M. Dang, Facial expression recognition and synthesis based on an appearance model, Signal Processing: Image Communication 19 (8) (2004) 723–740.

[29] M. Malciu, F. Preteux, Tracking facial features in video sequences using a deformable model-based approach, in: Proceedings of the SPIE, Vol. 4121, 2000, pp. 51–62.

[30] Y. Zhu, L. C. D. Silva, C. C. Ko, Using moment invariants and HMM in facial expression recognition, Pattern Recognition Letters 23 (1-3) (2002) 83–91.

[31] Y. Chang, C. Hu, R. Feris, M. Turk, Manifold based analysis of facial expression, Image and Vision Computing 24 (2006) 605614.

[32] G.Guo, C.R.Dyer, Learning from examples in the small sample case: Face expression recognition, IEEE Transactions on Systems, Man, And Cybernetics-Part B: Cybernetics 35 (3) (2005) 477–488.

[33] S. B. Gokturk, C. Tomasi, B. Girod, J.-Y. Bouguet, Model-based face tracking for view-independent facial expression recognition, in: Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, Cambridge, U.K., 2002, pp. 287–293.

[34] I. Kotsia, I. Pitas, Facial expression recognition in image sequences using geometric deformation features and support vector machines, IEEE Transactions on Image Processing 16 (1) (2007) 172–187.

[35] D. Lee, H. Seung, Algorithms for non-negative matrix factorization, in: NIPS, 2000, pp. 556–562.

[36] S. Zafeiriou, A. Tefas, I. Buciu, I. Pitas, Exploiting discriminant information in non-negative matrix factorization with application to frontal face verification, IEEE Transactions on Neural Networks 17 (3) (2006) 683–695.

[37] G. Donato, M. Bartlett, J. Hager, P. Ekman, T. Sejnowski, Classifying facial actions, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (10) (1999) 974–989.

[38] I. Kotsia, S. Zafeiriou, N. Nikolaidis, I. Pitas, Multiclass support vector machines and metric multidimensional scaling for facial expression recognition, 2007.

[39] I. Buciu, I. Pitas, DNMF modeling of neural receptive fields involved in human facial expression perception, Journal of Visual Communication and Image Representation 17 (5) (2006) 958–969.

[40] S. Li, X. Hou, H. Zhang, Learning spatially localized, parts-based representation, in: CVPR, Kauai, HI, USA, 2001, pp. 207–212.

[41] M. F. Cox, M. A. A. Cox, Multidimensional Scaling, Chapman and Hall, 2001.

[42] I. Borg, P. Groenen, Modern Multidimensional Scaling: theory and applications, Springer-Verlag, New York, 1997.

[43] W. S. Torgerson, Theory and Methods of Scaling, Wiley, New York, 1958.

[44] S. Krinidis, I. Pitas, Statistical analysis of facial expressions for facial expression synthesis, submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence.

[45] M. Collins, R. E. Schapire, Y. Singer, Logistic regression, adaboost and bregman distances, Computational Learing Theory (2000) 158–169.

[46] J. Y. Bouguet, Pyramidal implementation of the Lucas-Kanade feature tracker, Tech. rep., Intel Corporation, Microprocessor Research Labs (1999).

[47] L. Wiskott, J. M. Fellous, N. Kuiger, C. von der Malsburg, Face recognition by elastic bunch graph matching, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (1997) 775–779.

[48] E. Pekalska, P. Paclik, R. Duin, A generalized kernel approach to dissimilarity-based classification, Journal of Machine Learning Research 2 (2001) 175–211.

[49] H. Drucker, W. Donghui, V. Vapnik, Support vector machines for spam categorization, IEEE Transactions on Neural Networks 10 (5) (1999) 1048 – 1054.

[50] A. Ganapathiraju, J. Hamaker, J. Picone, Applications of support vector machines to speech recognition, IEEE Transactions on Signal Processing 52 (8) (2004) 2348 – 2355.

[51] M. Pontil, A. Verri, Support vector machines for 3D object recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (6) (1998) 637–646.

[52] A. Tefas, C. Kotropoulos, 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 23 (7) (2001) 735–746.

[53] V. Vapnik, Statistical learning theory, Wiley, New York, 1998.

[54] C. W. Hsu, C. J. Lin, A comparison of methods for multiclass Support Vector Machines, IEEE Transactions on Neural Networks 13 (2) (2002) 415–425.

[55] J. Weston, C. Watkins, Multi-class Support Vector Machines, Tech. Rep. Techical report CSD-TR-98-04 (2004).

[56] J. Weston, C. Watkins, Multi-class Support Vector Machines, in: Proceedings of ESANN99, Brussels, Belgium, 1999.

[57] C. J. C. Burges, A tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge discovery 2 (2).

[58] R. Snelick, U. Uludag, A. Mink, M. Indovina, A. Jain, Large-scale evaluation of multimodal biometric authentication using state-of-the-art systems, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (3) (2005) 450 – 455.

[59] A. G. Bors, I. Pitas, Median radial basis function neural network, IEEE Transactions on Neural Networks 7 (1996) 1351–1364.

[60] V. Chatzis, A. Bors, I. Pitas, Multimodal decision-level fusion for person authentication, IEEE Transactions on Systems, Man and Cybernetics, Part A 29 (6) (1999) 674–680.

[61] C. C. Hung, Y. Kim, T. Coleman, A comparative study of radial basis function neural networks and wavelet neural networks in classification of remotely sensed data, in: Fourth Biannual World Automation Congress (WAC2002), Orlando, FL, USA, 2002.

[62] R. J. Schalkof, Pattern Recognition: Statistical, Structural and Neural Approaches, Wiley, New York, 1992.

[63] B. Scholkopf, K.-K. Sung, C. J. C. Burges, F. Girosi, P. Niyogi, T. Poggio, V. Vapnik, Comparing support vector machines with gaussian kernels to radial basis function classifiers, IEEE Transactions on Signal Processing 45 (11) (1997) 2758–2765.

[64] I. Pitas, A. N. Venetsanopoulos, Nonlinear Digital Filters: Principles and Applications, Kluwer Academic, 1990.

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

[66] S. Aleksic, K. Katsaggelos, Automatic facial expression recognition using facial animation parameters and multi-stream hmms, IEEE Transactions on Information Forensics and Security 1 (1) (2006) 3–11.

[67] M. Yeasin, B. Bullot, R. Sharma, Recognition of facial expressions and measurement of levels of interest from video, Vol. 8, 2006, pp. 500–508.

[68] G. Zhao, M. Pietikainen, Dynamic texture recognition using local binary patterns with an application to facial expressions, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (6) (2007) 915–928.