Discriminant Graph Structures for Facial Expression Recognition

Stefanos Zafeiriou and Ioannis Pitas, *Fellow, IEEE*

***Abstract*—In this paper, a series of advances in elastic graph matching for facial expression recognition are proposed. More specifically, a new technique for the selection of the most discriminant facial landmarks for every facial expression (discriminant expression-specific graphs) is applied. Furthermore, a novel kernel-based technique for discriminant feature extraction from graphs is presented. This feature extraction technique remedies some of the limitations of the typical kernel Fisher discriminant analysis (KFDA) which provides a subspace of very limited dimensionality (i.e., one or two dimensions) in two-class problems. The proposed methods have been applied to the Cohn–Kanade database in which very good performance has been achieved in a fully automatic manner.**

***Index Terms*—Elastic graph matching, expandable graphs, Fisher's linear discriminant analysis, Kernel techniques.**

I. 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, video-conferencing, user profiling, customer satisfaction studies for broadcast and web services and smart environments construction require efficient facial expression recognition in order to achieve the desired results [1], [2].

Several research efforts have been made 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) [3]. The interested reader may refer to [4]–[6] and in the references therein for facial expression recognition methods. A more recent survey for facial expression recognition can be found in [7]. Fully automatic facial expression recognition is a difficult task, since it requires robust face and facial landmarks detection and tracking of the specific facial landmarks that participate in the development of the various facial expressions. That is, recognition performance highly depends on the robust detection and/or tracking of certain landmarks upon the facial area (e.g., eye, lip tracking, etc.). In many cases, in order to reduce the effect of false facial landmarks detection and their erroneous tracking, one or more parts of the preprocessing are performed manually. In [6], manual facial landmark annotation of the Candide grid [8] is performed in neutral images. The preselected facial landmarks on the neutral image are tracked until the image reaches its highest expression intensity. Then, the deformation of these landmarks with respect to the neutral state, throughout the facial expression evolvement, is used for facial expression recognition. In [9], manual facial landmark selection has been performed in every facial expression image. In other cases facial landmark detection has been performed using special equipment [10], for instance when infrared cameras have been used for robust eye detection. A method that could achieve fully automatic facial expression recognition is the elastic graph matching (EGM) algorithm [11].

EGM [12] has been initially proposed for arbitrary object recognition from images and has been a very popular topic of research for various facial image characterization applications. In EGM, a reference object graph is created by overlaying a rectangular elastic sparse graph on the object image and then calculating a Gabor wavelet bank response at each graph node. This way, a feature vector is assigned to every node, the so-called *jet*. The graph matching process is implemented by a stochastic optimization of a cost function which takes into account both jet similarities and grid deformations. A two stage coarse-to-fine optimization procedure suffices for the minimization of such a cost function.

A lot of research has been conducted in order to boost the performance of EGM for face recognition, face verification, facial expression recognition and sex determination [13]–[25]. In [14], the graph structure has been enhanced by introducing a stack like structure, the so-called *bunch graph*, and has been tested for face recognition. For every node in the bunch graph structure, a set of jets has been measured for different instances of a face (e.g., with open or closed mouth, open or shut eyes). This way, the bunch graph representation could cover a variety of possible changes in the appearance of a face. In [15], the bunch graph structure has been used for determining facial characteristics, such as beard, the presence of glasses, or even a person's sex.

In [17], EGM has been proposed and tested for frontal face verification. A variant of the typical EGM, the so-called *morphological elastic graph matching* (MEGM), has been proposed for frontal face verification and tested for various recording conditions [18]–[20]. In [18], [20], the standard coarse-to-fine approach proposed in [17] for EGM has been replaced by a simulated annealing method that optimizes a cost function of the jet similarity measures subject to node deformation constraints. Another variant of EGM has been presented in [21], where morphological signal decomposition has been used instead of the

Manuscript received October 10, 2007; revised July 07, 2008. Current version published December 10, 2008. This work was supported by the 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. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Wen Gao.

S. Zafeiriou is with the Department of Electrical and Electronic Engineering, Imperial College London, London, U.K. (e-mail: dralbert@aiia.csd.auth.gr).

I. Pitas is with the Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece (e-mail: pitas@aiia.csd.auth.gr).

Digital Object Identifier 10.1109/TMM.2008.2007292

1520-9210/$25.00 © 2008 IEEE

standard Gabor analysis [17]. EGM with Gabor jets for facial expression recognition has been proposed in [11] and [26]–[29].

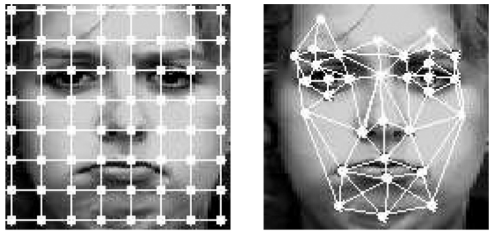
Discriminant techniques have been employed in order to enhance the classification performance of EGM. The use of linear discriminant techniques at the feature vectors for the extraction of the most discriminating features has been proposed in [17], [18], and [20]. Several schemes that aim at weighting the graph nodes according to their discriminatory power have also been proposed in [18], [20], [25], and [30]. A combined discriminant scheme has been proposed in [22], where discriminant analysis has been employed in every step of elastic graph matching for face verification. The use of Fisher's Linear Discriminant Analysis (FLDA) for discriminant feature selection in the graphs for facial expression recognition has been proposed in [11]. In [11], [29], FLDA has been applied in a graph-wise manner (i.e., the feature vectors that have been used in FLDA were the set of graph jets), contrary to the methods in [18], [20], and [22] where node-specific discriminant transforms have been calculated. Moreover, a series of discriminant techniques in graph-based representations with Gabor features have been proposed in [9]. The methods in [9] have some resemblance with EGM but have not implemented an elastic graph matching procedure since landmark selection and matching has been manually performed. In [23] and [24], novel robust Gabor-based features have been proposed and novel wrapping elastic graph matching procedure has been introduced which is robust against rotation and scaling. Moreover, in [23] a novel kernel-based method for feature extraction has been proposed and used for face recognition.

Finally, in [31], a method for face recognition has been proposed which follows a similar strategy to the one used in this paper. That is, face recognition is treated as a two class problem in order to extract discriminant Gabor-based features using Adaboost. To apply the AdaBoost they have introduced the intra-face and extra-face difference space in the Gabor feature space and converted the multiclass problem to a corresponding two-class. In addition, to deal with the imbalance between the amount of the positive samples and that of the negative samples, a re-sampling scheme has been adopted to choose the negative samples.

Although a lot of research has been conducted for feature selection and discriminant node weighting in elastic graphs, not much have been done concerning the type of graphs that is more appropriate for face recognition, face verification and facial expression recognition. The sparse graph that has been used for face representation in the literature is:

- either an evenly distributed graph placed over a rectangular image region [17], [18], [20], [21], [30]
- or a graph that is placed on preselected nodes that correspond to some fiducial facial points (e.g., nose, eyes, etc.) [11], [14], [15], [26]–[29].

Intuitively, one may think that graphs with nodes placed at pre-specified facial landmarks may perform better than the rectangular graphs. However, such graphs are more difficult to be automatically applied, since they require a detection module to find the precise coordinates of the facial landmarks in the reference images or, in many cases, manual landmark annotation [9], [11], [14]. On the contrary, an evenly distributed rectangular graph is easier to be handled automatically, since only a face detection algorithm is required to find an initial approximation of the rectangular facial region [17], [18], [20], [21], [30]. Fig. 1 shows the two different types of facial graphs used in elastic graph matching algorithms. In [32], an algorithm that finds the optimal discriminant graph structure has been proposed (optimal according to a discriminant criterion). The graphs proposed in [32] have nodes placed at discriminant facial landmarks. It has been shown in [32] that these graphs can be found in a fully automatic manner and have better performance than the typical rectangular graphs in face verification.



(a) (b)

Fig. 1. Different types of facial graphs: (a) rectangular graph and (b) graph with nodes at fiducial landmarks.

In this paper, we meticulously study the use of EGM for facial expression recognition. More specifically, the contributions of this paper are the following.

- The motivation and application of morphological filters in order to deal with the problem of facial expression recognition.
- The application of expression-specific graphs with nodes placed at discriminant landmarks. In order to apply such graphs, we introduce a discriminant analysis that produces a graph whose nodes correspond to the most discriminant facial landmarks for a particular expression.
- The introduction of a novel kernel-based method for both graph-wise and node-wise discriminant feature selection and its application for facial expression recognition. The main contribution of the proposed kernel-technique, is that it tries to remedy some of the limitations of the kernel methods based on the Fisher's discriminant criterion that provide very limited number of features in two class problems (i.e., the so-called kernel direct discriminant analysis (KDDA) provides only one discriminant projection [33] and the so-called Complete kernel Fisher discriminant analysis (CKFDA) [34] only two discriminant dimensions in two class problems). These spaces of very limited number of dimensions may prove to be insufficient for correctly representing the samples. The proposed approach discovers a low dimensional space with the number of dimensions to be proportional to the number of training samples.

The proposed method, unlike the methods in [6] and [9], is fully automatic. That is, there is no need to manually locate the face and/or manual annotate facial landmarks. The facial expression recognition problem is a challenging one because different individuals display the same expression differently. Selecting the most relevant features and ignoring unimportant features is a key

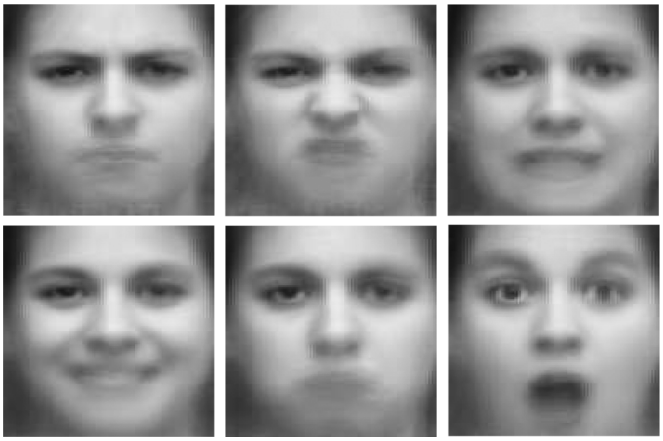


Fig. 2. Mean images of each facial expression for the posers of the Cohn–Kanade database. From left to right, the mean image of anger, disgust, fear, happiness, sadness, and surprise are depicted.

step for the solution of this problem. The proposed method, selects automatically the best facial landmarks for every facial expression. That is, the discriminant analysis learns automatically the discriminant landmarks for every facial expression, unlike the method in [11], where the fiducial grid has been found by manually locating various landmarks of each facial image.

The rest of the paper is organized as follows. In Section II, the application of elastic graph matching algorithm for facial expression recognition is discussed. In Section III, the algorithm for learning discriminant expression-specific graphs structures is proposed. In Section IV, the novel discriminant analysis with kernels for feature extraction is introduced. Experimental results using the Cohn–Kanade database [35] are described in Section V. Finally, conclusions are drawn in Section VI.

II. Elastic Graph Matching for Facial Expression Recognition

A. Graph Selection for Facial Expression Representation

In the first step, of the EGM algorithm, a sparse graph that is suitable for facial expression representation is selected [11], [14], [17], [18], like the ones depicted in Fig. 1. Afterwards, the reference facial image or images are selected in order to build the reference samples for every facial expression. Two types of reference graphs have been considered in this work. The first graph uses only the mean facial expression image as the reference facial expression graph. The mean facial expression image for each of the expressions is depicted in Fig. 2.

Another alternative for building the reference graph for every facial expression, is the bunch graph setup. In the reference bunch graph instead of one jet per node, a set (bunch) of jets is stored that can model different instances of jets for a facial expression. In Fig. 3, the two alternatives regarding the choice of the reference facial expression graph are pictorially described. The reference graph that is constructed using the mean images has been significantly outperformed in our experiments by the bunch graph thus, we will refer only to the bunch graph from now onwards.

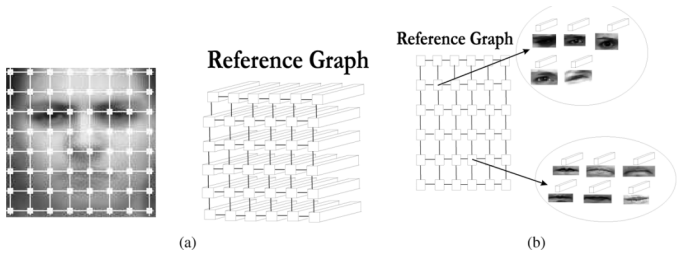


Fig. 3. (a) Reference anger graph has one jet per node and is built using only the mean facial expression image. (b) Reference graph contains a bunch of jets per node for different instances of anger.

B. Multiscale Image Analysis

The facial image region is analyzed and a set of local descriptors is extracted at each graph node. Analysis is usually performed by building an information pyramid using scale-space techniques. In the standard EGM, a 2-D Gabor based filter bank has been used for image analysis [12]. The output of multiscale morphological dilation-erosion operations or the morphological signal decomposition at several scales is a nonlinear alternative of the Gabor filters for multiscale analysis. Both methods have been successfully used for facial image analysis [18], [20], [21], [36]. In the morphological EGM, this information pyramid is built using multiscale morphological dilation-erosions [37]. Given an image $f(\mathbf{x}) : \mathcal{D} \subseteq \mathcal{Z}^2 \rightarrow \Re$ (where $\mathcal{Z}$ is the set of integers and $\Re$ is the set of real numbers) and a structuring function $g(\mathbf{x}) : \mathcal{G} \subseteq \mathcal{Z}^2 \rightarrow \Re$, the dilation of the image $f(\mathbf{x})$ by $g(\mathbf{x})$ is denoted by $(f \oplus g)(\mathbf{x})$. Its complementary operation, the erosion, is denoted by $(f \ominus g)(\mathbf{x})$ [18]. The multiscale dilation-erosion pyramid of the image $f(\mathbf{x})$ by $g_\sigma(\mathbf{x})$ is defined in [37], where σ denotes the scale parameter of the structuring function. In [18] it was demonstrated that the choice of the structuring function does not lead to statistically significant changes in the classification performance. However, it affects the computational complexity of feature calculation.

Such morphological operations can highlight and capture important information for key facial features such as eyebrows, eyes, nose tip, nostrils, lips, face contour, etc. but can be affected by different illumination conditions and noise [18]. To compensate for these conditions, the normalized multiscale dilation-erosion is proposed for facial image analysis. It is well known that the different illumination conditions affect the facial region in a non uniform manner. However, it can be safely assumed that the illumination changes are locally uniform inside the area of the structuring element used for multiscale analysis. The proposed morphological features are calculated by subtracting the mean value of the intensity of the image f inside the area of the structuring element from the corresponding maximum (dilation) or minimum (erosion) of the area. Formally, the normalized multiscale morphological analysis is given by

$$(f \star g_\sigma)(\mathbf{x}) = \begin{cases} (f \oplus g_\sigma)(\mathbf{x}) - m_-(f, \mathbf{x}, G_\sigma), & \text{if } \sigma > 0 \\ f(\mathbf{x}), & \text{if } \sigma = 0 \\ (f \ominus g_{|\sigma|})(\mathbf{x}) - m_+(f, \mathbf{x}, G_\sigma), & \text{if } \sigma < 0 \end{cases} \quad (1)$$

where $m_-(f,\mathbf{x},\mathcal{G}_\sigma)$ and $m_+(f,\mathbf{x},\mathcal{G}_\sigma)$ are the mean values of the image $f(\mathbf{x}-\mathbf{z})$, $\mathbf{x}-\mathbf{z}\in\mathcal{D}$ and $f(\mathbf{x}+\mathbf{z})$, $\mathbf{x}+\mathbf{z}\in\mathcal{D}$ inside the support area of the structuring element $\mathcal{G}_\sigma=\{\mathbf{z}\in\mathcal{G}:\|\mathbf{z}\|<\sigma\}$, respectively. Another implementation for the operators $m_+(f,\mathbf{x},\mathcal{G}_\sigma)$ and $m_-(f,\mathbf{x},\mathcal{G}_\sigma)$ would be the median of the values of the image inside the support area of the structuring element. The output of these morphological operations forms the jet $\mathbf{j}(\mathbf{x}^l)$ at the graph node l that is located in image coordinates $\mathbf{x}^l$

$$\mathbf{j}(\mathbf{x}^l)=((f\star g_{\sigma_\Lambda})(\mathbf{x}^l),\ldots,(f\star g_{\sigma_1})(\mathbf{x}^l),f(\mathbf{x}^l),(f\star g_{\sigma_{-1}})(\mathbf{x}^l),\ldots,(f\star g_{\sigma_{-\Lambda}})(\mathbf{x}^l)). \quad (2)$$

where Λ is the number of different scales used. The various scales of normalized morphological multiscale analysis (NMMA) can highlight various facial characteristics that are particular important for facial expression development, like the shape of the mouth, teeth, eyebrows, furrows, etc. Some examples that verify the above statement can be found in Fig. 4, where the different scales of NMMA are shown for different facial parts and facial expressions.

C. Matching Procedure

The next step of EGM is to match the reference graph on the test facial expression image in order to find the correspondences of the reference graph nodes on the test image. This is accomplished by minimizing a cost function that employs node jet similarities while preserving at the same time the node neighborhood relationships. Let the subscripts t and r denote a test and a reference facial image (or graph), respectively. The L_2 norm between the feature vectors at the l-th graph node of the reference and the test graph is used as a similarity measure between jets, i.e.:

$$C_f\left(\mathbf{j}\left(\mathbf{x}_t^l\right),\mathbf{j}\left(\mathbf{x}_r^l\right)\right)=\left\|\mathbf{j}\left(\mathbf{x}_r^l\right)-\mathbf{j}\left(\mathbf{x}_t^l\right)\right\|. \quad (3)$$

Let $\mathcal{V}$ be the set of all graph vertices of a certain facial image. For the rectangular graphs, all nodes, except from the boundary nodes, have exactly four connected nodes. Let $\mathcal{H}(l)$ be the four-connected neighborhood of node l. In order to quantify the node neighborhood relationships using a metric, the local node deformation is used

$$C_d\left(\mathbf{x}_t^l,\mathbf{x}_r^l\right)=\sum_{\xi\in\mathcal{H}(l)}\left\|\left(\mathbf{x}_t^l-\mathbf{x}_r^l\right)-\left(\mathbf{x}_t^\xi-\mathbf{x}_r^\xi\right)\right\|. \quad (4)$$

The objective is to find a set of vertices $\{\mathbf{x}_t^l(r),l\in\mathcal{V}\}$ in the test image that minimizes the cost function

$$C\left(\{\mathbf{x}_t^l\}\right)=\sum_{l\in\mathcal{V}}\left\{C_f\left(\mathbf{j}\left(\mathbf{x}_t^l\right),\mathbf{j}\left(\mathbf{x}_r^l\right)\right)+\lambda C_d\left(\mathbf{x}_t^l,\mathbf{x}_r^l\right)\right\}. \quad (5)$$

The jet of the l-th node that has been produced after the matching procedure of the reference facial expression graph $r(r=\{\text{anger, disgust, fear, happiness, sadness, surprise}\})$ to the facial expression image t is denoted as $\mathbf{j}(\mathbf{x}_t^l(r))$. This notation is used due to the fact that different facial expressions r result to different test jets $\mathbf{j}(\mathbf{x}_t^l(r))$. Thus, the jet of the l-th node of the test facial expression graph t is a function of the reference facial expression graph r. The notation $\mathbf{j}(\mathbf{x}_r^l)$ is used only when the l-th node is in a preselected position of a facial image.

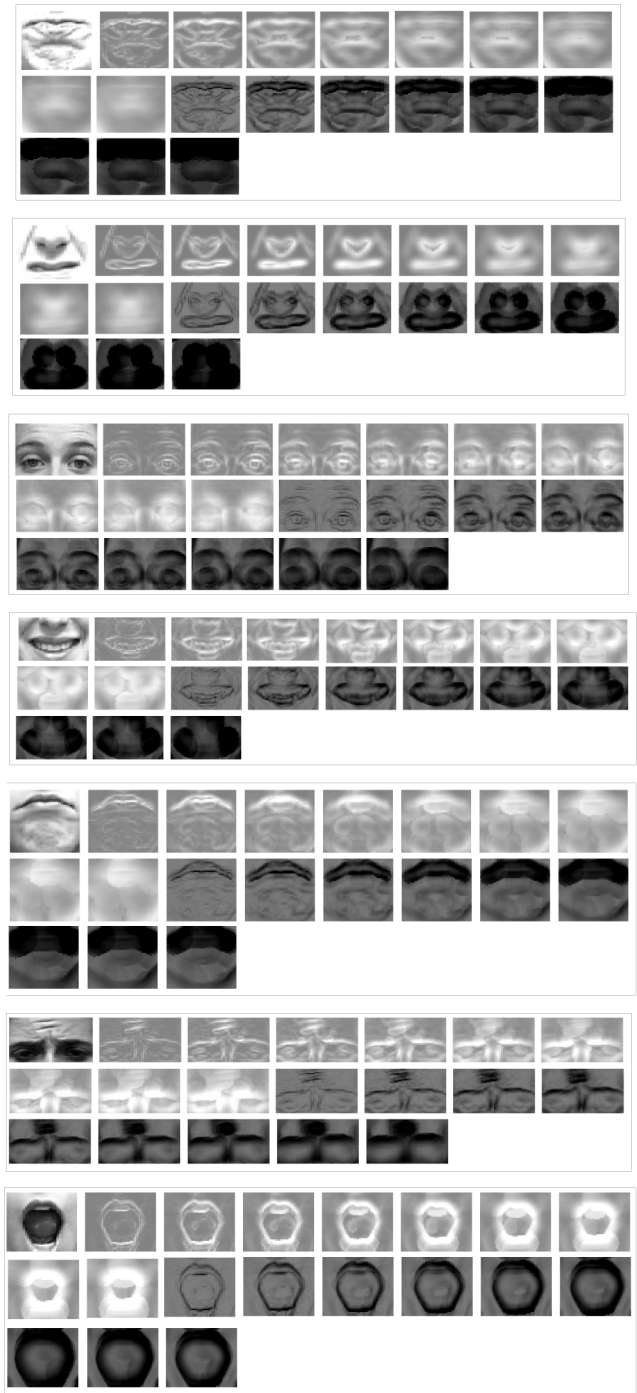


Fig. 4. NMMA of various facial parts in expressed facial images. The upper left image of every block is a facial part from the original image extracted from the Cohn–Kanade database. The first nine images of every block starting from the left corner, apart from the upper left one, are the normalized dilated images and the remaining nine are the normalized eroded images. As can be seen, NMMA captures and highlights important facial features like mouth, eyebrows, eyes and furrows that participate in facial expression development.

In [18], the optimization of (5) has been implemented as a simulated annealing procedure that imposes global translation of the graph and local node deformations. In this paper, in order

to deal with face translation, rotation and scaling, the following optimization problem:

$$D_t(r) = \sum_{l\in\mathcal{V}} \left\{C_f\left(\mathbf{j}\left(\mathbf{x}_t^l\right), \mathbf{j}\left(\mathbf{x}_r^l\right)\right)\right\} \text{ subject to}$$
$$\mathbf{x}_t^l = \mathbf{A}\mathbf{x}_r^l + \boldsymbol{\delta}_l, \|\boldsymbol{\delta}_l\| \leq \delta_{\max} \quad (6)$$

is solved using simulated annealing, as well. The matrix $\mathbf{A}$ is an Euclidean transformation matrix and can be expressed as $\mathbf{A} = \mathbf{TRS}$, assuming, before the initialization of the optimization procedure, that the center of the mass of the graph coincides with the center of the coordinate system axes. The matrix $\mathbf{S} = \text{diag}\{a_1, a_2, 1\}$ is a scaling matrix with $1 - a < a_1 < 1 + a$, $1 - a < a_2 < 1 + a$ and $a > 0$ is a scalar that controls the maximum and minimum scaling of the graph. The matrix $\mathbf{R}$ is the rotation matrix

$$\mathbf{R} = \begin{bmatrix} \cos\theta & -sin\theta & 0 \\ \sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where $-\theta_1 < \theta < \theta_1$ with θ_1 is a scalar term that controls the maximum degree of rotation of the graph. Finally, $\mathbf{T}$ is a translation matrix

$$\mathbf{T} = \begin{bmatrix} 0 & 0 & T_1 \\ 0 & 0 & T_2 \\ 0 & 0 & 1 \end{bmatrix}$$

for the graph with $-T < T_1 < T, -T < T_2 < T$ and $T > 0$ is a scalar that controls the maximum translation of the graph. Finally, $\boldsymbol{\delta}_l$ denotes a local perturbation of the graph nodes. The choices of λ in (5) and $\delta_{\max}$ in (6) control the rigidity/plasticity of the graph [17], [18]. Another alternative for handling rotation and scaling using Gabor based features is the method proposed in [24], [23].

The optimization problems in (5) and (6) are valid only when the reference facial expression graph contains one jet per node. When the reference facial expression graph contains more than one jet per node, i.e., the case of bunch graph, the cost function in (5) should be reformulated as

$$C_B\left(\{\mathbf{x}_t^l\}\right) = \sum_{l\in\mathcal{V}} \min_m \left\{C_f\left(\mathbf{j}\left(\mathbf{x}_t^l\right), \mathbf{j}_{B_m}\left(\mathbf{x}_r^l\right)\right)\right\} + \lambda \sum_{l\in\mathcal{V}} C_d\left(\mathbf{x}_t^l, \mathbf{x}_r^l\right) \quad (7)$$

where $\mathbf{j}_{B_m}(\mathbf{x}_r^l)$ is the jet of m-th jet of the bunch of the l-th node for the facial expression graph r. In the same manner, the constrained optimization in (6) can be reformulated to

$$D_t(r) = \sum_{l\in\mathcal{V}} \min_m \left\{C_f\left(\mathbf{j}\left(\mathbf{x}_t^l\right), \mathbf{j}_{B_m}\left(\mathbf{x}_r^l\right)\right)\right\} \text{subject to}$$
$$\mathbf{x}_t^l = \mathbf{A}\mathbf{x}_r^l + \boldsymbol{\delta}_l, \|\boldsymbol{\delta}_l\| \leq \delta_{\max}. \quad (8)$$

In order to avoid a time-consuming elastic matching procedure, we first initialize the graph position using a face detector and afterwards we study for small scaling, rotation and translation changes for a finest matching. After the matching procedure, the distance $D_t(r)$ can be used as a quantitative measure for the similarity of a reference facial expression graph with a test image.

III. Finding Discriminant Expression-Specific Graph Structures

As has been already been discussed, the graphs that have been used for face representation in EGM algorithms have been either rectangular graphs or graphs with nodes manually placed at prespecified landmarks. It cannot be proven that these graphs are optimal for the examined applications i.e., face recognition/verification and facial expression recognition. In [32], it has been shown that graphs with nodes placed at the discriminant facial landmarks for every person perform significantly better than the rectangular graphs. We will try to find the optimal graph setup for facial expression recognition tasks (optimal under some criterion optimization).

A. Measuring the Significance of Each Node

In the following, $\mathbf{m}(\mathcal{X})$ denotes the mean vector of a set of vectors $\mathcal{X}$ and $N(\mathcal{X})$ its cardinality. When $\mathcal{X}$ is a set of scalar values their mean will be denoted as $m(\mathcal{X})$ and their variance as $\sigma^2(\mathcal{X})$. Let $\mathcal{F}_l(r)$ and $\tilde{\mathcal{F}}_l(r)$ be the sets for the jets of the l-th node that correspond to facial expression intra-class matchings (i.e., the graph jets that have been produced by matching the reference facial expression graph r to all images of the same facial expression class) and to facial expression inter-class matchings (i.e., the graph jets that have been produced by matching the reference facial expression graph r to the images of the other facial expressions), respectively. In order to define the similarity of a test jet $\mathbf{j}(\mathbf{x}_t^l(r))$ to the class of jets of the facial expression r for the same node, we use the following norm:

$$c_t^l(r) = \left\|\mathbf{j}\left(\mathbf{x}_t^l(r)\right) - \mathbf{m}(\mathcal{F}_l(r))\right\|^2 \quad (9)$$

which is actually the Euclidean distance of a sample to the mean of the facial expression class r and is one of most commonly employed measures in pattern recognition applications.

Let $\mathcal{C}_l(r)$ and $\tilde{\mathcal{C}}_l(r)$ be the sets of local similarity values $c_t^l(r)$ that correspond to the facial expression intra-class and inter-class samples, respectively. A possible measure for the discriminant power of the l-th node for the facial expression r is the Fisher's discriminant ratio [38]

$$p_1^l(r) = \frac{(m(\mathcal{C}_l(r)) - m(\tilde{\mathcal{C}}_l(r)))^2}{\sigma^2(\mathcal{C}_l(r)) + \sigma^2(\tilde{\mathcal{C}}_l(r))}. \quad (10)$$

In [18] and [20], it has been proposed to weight the graph nodes after the elastic graph matching using the coefficients $p_1^l(r)$ in order to form a similarity measure between graphs. Another possible measure of the discriminant power of a graph node is the following:

$$p_2^l(r) = \frac{\frac{1}{N(\tilde{\mathcal{C}}_l(r))} \sum_{c_t^l(r)\in\tilde{\mathcal{C}}_l(r)} c_t^l(r)}{\frac{1}{N(\mathcal{C}_l(r))} \sum_{c_t^l(r)\in\mathcal{C}_l(r)} c_t^l(r)}. \quad (11)$$

The measure (11) increases when the inter-class similarity measures for the l-th graph node are of high values and/or the local similarity measures for the intra-class similarity measures are

of low values. The measures defined in (10) and in (11) are heuristic indicators for the discriminant power of every node.

Now, by summing the discriminant coefficients for a certain graph setup g, we have

$$E_g(r) = \frac{1}{L}\sum_{l=1}^{L} p^l(r) \tag{12}$$

where L is the total number of nodes. This is the mean of all the discriminant measures and is a characteristic measure for a particular graph setup of the facial expression r. As described in [32], the previous analysis leads to an optimization procedure in order to find the graph g that has the maximum $E_g(r)$. The desired properties (constraints) of the graph g apart from having maximum $E_g(r)$ are:

- the graph should have a relatively small number of nodes so that the elastic graph matching procedure has low computational cost;
- the nodes should not be very close to each other in order to avoid redundant use of the same discriminant information.

Formally, the above optimization problem can be written as

$$\begin{aligned} \acute{g} &= \arg\max_{g} E_g(r) \text{ subject to} \\ \left\|\mathbf{x}_r^l - \mathbf{x}_r^j\right\| &\geq \Delta, \forall\, l, j \text{ nodes with } l \neq j \\ L &= \text{constant} \end{aligned} \tag{13}$$

where Δ is a preselected threshold that controls the density of the graph.

In order to solve the constraint optimization procedure we assume that the optimal solution is a sub-graph of the Δ-rectangular graph (i.e., the graph with nodes placed at every Δ pixels). An iterative algorithm that uses expandable graphs is proposed in order to find the discriminant graph. We assume that the nodes that have high discriminant values should be placed in facial regions that are indeed discriminant for the specific facial expression. These facial regions should be better represented. This can be achieved by expanding certain nodes that possess the highest discriminant power. In the following, the steps of the proposed algorithm are described in more detail. This procedure should be repeated for all six facial expressions in order to find the most discriminant graph for each one.

Let the initial graph that contains L vertices at the first iteration $i \leftarrow 1$. Let $\mathcal{B}_i$ be the set of graph vertices at the i-th iteration. The algorithm has the following steps.

Step 1. Match the reference graph of the facial expression r to all intra-class and inter-class images.

Step 2. For each node l, calculate the measure $p^l(r)$.

Step 3. Select a subset of the nodes with the higher discriminant value that have not been already expanded and expand them. The nodes that lie in the perimeter of the graph can be expanded only inside the facial region. Fig. 5 describes pictorially this step for the rectangular graph of anger.

Step 4. Verify that the inserted nodes do not violate the graph sparseness criterion. That is, erase the new nodes that violate the criterion $\|\mathbf{x}_r^l - \mathbf{x}_r^j\| < \Delta$, $\forall l, j$ (for the rectangular graphs used in this work, this is equivalent with checking if some of the inserted nodes have already been examined). The set of the final inserted nodes in the i-th iteration is denoted as $\mathcal{A}_i$.

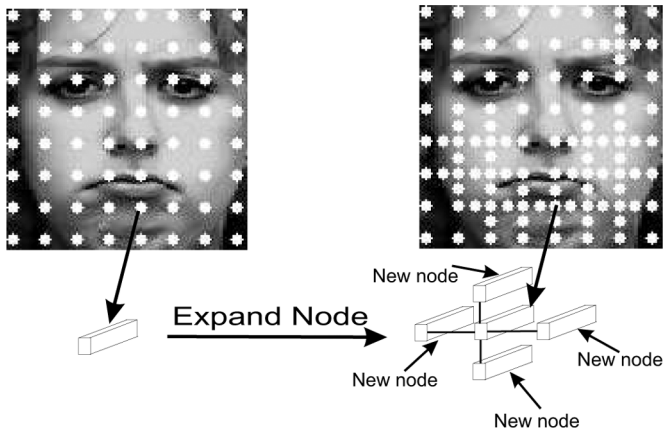


Fig. 5. Expanding the graph.

Step 5. Match locally the nodes of $\mathcal{A}_i$ in all the intra-class and inter-class facial expression images. Let $k \in \mathcal{A}_i$ be an inserted node and $\hat{\mathbf{x}}_t^k$ be the initial coordinate vector for the node k in a test image t. The local matching procedure is the outcome of the local search

$$\begin{aligned} \dot{\mathbf{x}}_t^k(r) = \arg\min_{\mathbf{x}_t^k} C_f\left(\mathbf{j}\left(\mathbf{x}_t^k\right), \mathbf{j}\left(\mathbf{x}_r^k\right)\right) hfill \\ \text{subject to} \|\mathbf{x}_t^k - \hat{\mathbf{x}}_t^k\| \leq \delta_{\max} \end{aligned} \tag{14}$$

where $\dot{\mathbf{x}}_t^k(r)$ is the final coordinate vector that gives the jet $\mathbf{j}(\dot{\mathbf{x}}_t^k(r))$.

Step 6. For each node $k \in \mathcal{A}_i$, calculate its discriminant value $p^k(r)$.

Step 7. Let $\mathcal{C}_i = \mathcal{A}_i \cup \mathcal{B}_i$. Order the nodes in $\mathcal{C}_i$ according to their discriminant power and obtain a graph g_{i+1} by keeping only the L nodes with the highest discriminant power. The set $\mathcal{B}_{i+1}$ contains the nodes of g_{i+1}.

Step 8. If $(E_{g_{i+1}}(r) - E_{g_i}(r)) > \tau$ then $i \leftarrow i+1$ and goto Step 4 else stop.

Fig. 6 shows the optimal graphs derived from the proposed procedure for the Cohn–Kanade database [35] using Normalized morphological features, respectively. All images have been aligned for visualization purposes. As can be seen from these images, the nodes of the optimal graphs for the morphological features are upon facial areas that correspond to furrows, frows, lips etc. which are landmarks that are considered discriminant for the classification of facial expressions. The main reason for the graphs not being symmetric is that in many cases facial expressions are not symmetric in many persons. The asymmetry clue of facial expressions has been commented and used for face recognition in [39]. The interested reader may refer to [40] and [39] and in references therein for more detail concerning the asymmetry of facial expressions. The other reason is that the method is applied in a fully automatic manner, thus this procedure may have introduced additional asymmetries to the graphs. Moreover, we can incorporate symmetric constraints in the proposed algorithm in order to satisfy symmetry in the derived

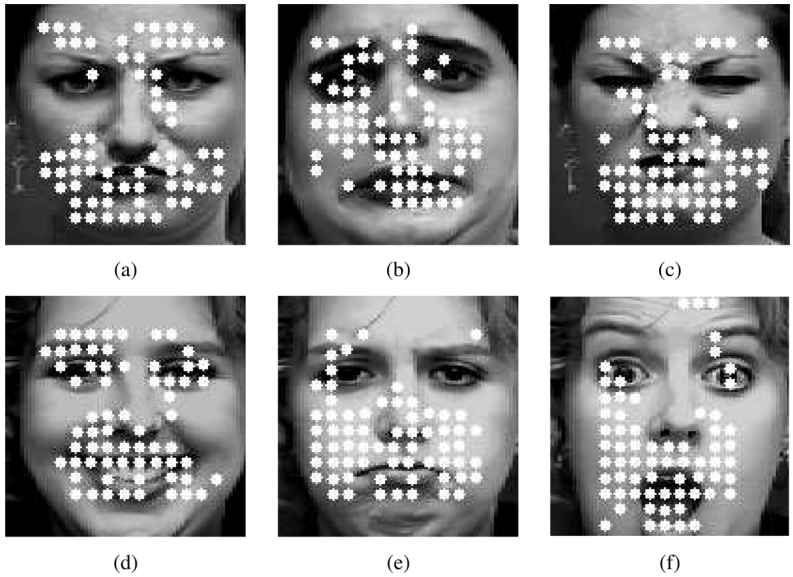


Fig. 6. Optimal graphs found using Normalized Morphological-based features for (a) anger, (b) disgust, (c) fear, (d) happiness, (e) sadness, and (f) surprise.

graphs. This can be accomplished, when expanding a node, by expanding its symmetric in the left or right part of the face, as well. The same procedure can be followed when erasing a node. In the fully automatic scenario, that has been followed in this paper, the best results has been produced by graphs like the ones shown in Fig. 6

The elastic graph matching procedure of the new graphs is performed using the minimization procedure indicated in the optimization problem (6). The optimization problem (6) uses global Euclidean transformation (i.e., using rotation, translation, and scaling) of the graph. That is, the components cannot be transformed independently but only as part of the entire graph. In the second step, every node can be locally matched (deformed) independently as imposed by the optimization problem (6).

IV. A Novel Two-Class Kernel Discriminant Analysis for Feature Extraction

We can improve the performance of the new graph by imposing discriminant analysis to the jets of the nodes or to the whole graphs as in [18], [20], [11]. To do so, two strategies can be considered.

1) *Graph-wise feature extraction.* In this case, the feature vector is the whole graph that is comprised of all jets and the algorithm learns for every facial expression one discriminant transform. Such a strategy has been applied in [11] for facial expression recognition.
2) *Node-wise feature extraction.* In this case, the algorithm learns node-specific discriminant transforms for every facial expression. This strategy is motivated by the fact that every graph node is a local expert that contains its own discriminant power. Such a strategy has been followed in [18], [22], where person and node-specific discriminant transforms have been learned for every node.

In the following we will formulate a novel nonlinear discriminant feature extraction method that can be applied in both strategies. An optimization procedure is used that is inspired from the optimization of Fisher's criterion in [41]. The advantage of choosing a similar optimization procedure to [41] is that it does not require matrix inversions contrary to other optimization procedures [42], [34]. The main limitation of the discriminant analysis based on Fisher's optimization problem in [41], [42], [34] is that for two class problems it produces only one discriminant direction contrary to the proposed criterion that provides a set of discriminant directions with its number to be proportional to the number of training samples.

Our aim is to find a discriminant feature extraction transform $\mathbf{\Psi}$ (in most cases $\mathbf{\Psi}$ serves as a dimensionality reduction matrix). In order to make use of kernel techniques the original input space is projected to an arbitrary-dimensional space $\mathcal{F}$ (the space $\mathcal{F}$ usually has the structure of a Hilbert space [43], [44]). To do so, let $\phi : \mathbf{y} \in \Re^M \longrightarrow \phi(\mathbf{y}) \in \mathcal{F}$ be a nonlinear mapping from the input space $\Re^M$ to the Hilbert space $\mathcal{F}$. In the Hilbert space, we want to find linear projections to a low-dimensional space with enhanced discriminant power. The discriminant power of the new space is often defined in respect to a discriminant optimization criterion. This discriminant criterion defines an optimization problem which gives a set of linear projections in $\mathcal{F}$ (linear in $\mathcal{F}$ is nonlinear in $\Re^M$).

A linear subspace transformation of $\mathcal{F}$ onto a K-dimensional subspace, which is isomorphic to $\Re^K$, is a matrix $\mathbf{\Psi} = [\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_K]$ with $\mathbf{x}_i \in \mathcal{F}$. The new projected vector $\acute{\mathbf{y}} \in \Re^M$, of the vector $\mathbf{y}$, is given by

$$\acute{\mathbf{y}} = \mathbf{\Psi}^T \phi(\mathbf{y}) = \left[\boldsymbol{\psi}_1^T \phi(\mathbf{y}), \ldots, \boldsymbol{\psi}_K^T \phi(\mathbf{y})\right]^T. \quad (15)$$

The dimensionality of the new space is usually much smaller than the dimensionality of $\mathcal{F}$ and the dimensionality of the input space $\Re^M$ (i.e., $K \ll M$). The matrix multiplication in (15) is computed indirectly (i.e., without explicit calculation of ϕ) using dot-products in the Hilbert space $\mathcal{F}$ [33], [34], [45].

Prior to developing the new optimization problem, we will introduce some notation that will be used throughout this Section. Let that the training set be separated into two disjoint classes $\mathcal{Y}$

and $\tilde{\mathcal{Y}}$. In our case, the class $\mathcal{Y}$ represents the facial expression intra-class samples and the class $\tilde{\mathcal{Y}}$ denotes the facial expression inter-class samples. For notation compactness, let $n = N(\mathcal{Y} \cup \tilde{\mathcal{Y}})$. The intra-class vectors $\mathbf{y}_i$ be denoted as $\boldsymbol{\rho}_i (\mathbf{y}_i \in \mathcal{Y})$, while the inter-class samples $\mathbf{y}_i$ be denoted as $\boldsymbol{\kappa}_i (\mathbf{y}_i \in \tilde{\mathcal{Y}})$. Let also $\bar{\boldsymbol{\rho}} = 1/(N(\mathcal{Y})) \sum_{i=1}^{N(\mathcal{Y})} \phi(\boldsymbol{\rho}_i)$, $\bar{\boldsymbol{\kappa}} = 1/(N(\tilde{\mathcal{Y}}) \sum_{i=1}^{N(\tilde{\mathcal{Y}})} \phi(\boldsymbol{\kappa}_i)$ and $\bar{\mathbf{m}} = 1/(n) \sum_{i=1}^{n} \phi(\mathbf{y}_i)$ be the mean vectors of $\mathcal{Y}$, $\tilde{\mathcal{Y}}$ and total mean of vectors in the Hilbert space $\mathcal{F}$. Any function k satisfying the Mercer's condition can be used as a kernel. The dot product of $\phi(\mathbf{y}_i)$ and $\phi(\mathbf{y}_j)$ in the Hilbert space can be calculated without having to evaluate explicitly the mapping $\phi(\cdot)$ as $k(\mathbf{y}_i, \mathbf{y}_j) = \phi(\mathbf{y}_i)^T \phi(\mathbf{y}_j)$ (this is also known as the kernel trick [43], [44]). Typical kernels are the polynomial and radial basis function (RBF) kernels

$$\begin{aligned} k(\mathbf{x}, \mathbf{y}) &= \phi(\mathbf{x})^T \phi(\mathbf{y}) = (\mathbf{x}^T \mathbf{y} + 1)^d \\ k(\mathbf{x}, \mathbf{y}) &= \phi(\mathbf{x})^T \phi(\mathbf{y}) = e^{-\gamma (\mathbf{x}-\mathbf{y})^T (\mathbf{x}-\mathbf{y})} \end{aligned} \tag{16}$$

where d is the degree of the polynomial and γ controls the spread of the Gaussian kernel.

The criterion that is used in this paper, will be formed using a simple similarity measure in the Hilbert space $\mathcal{F}$. This measure quantifies the similarity of a given feature vector $\mathbf{y}$ to the class r in the subspace spanned by the columns of the matrix $\boldsymbol{\Psi} = [\boldsymbol{\psi}_1 \dots \boldsymbol{\psi}_K]$, with $\boldsymbol{\psi}_i \in \mathcal{F}$. The L_2 norm in the reduced space spanned by the columns of $\boldsymbol{\Psi}$ is used as a similarity measure

$$\begin{aligned} \hat{c}(\mathbf{y}) &= \|\boldsymbol{\Psi}^T (\phi(\mathbf{y}) - \bar{\boldsymbol{\rho}})\|^2 \\ &= \mathrm{tr}[\boldsymbol{\Psi}^T (\phi(\mathbf{y}) - \bar{\boldsymbol{\rho}})(\phi(\mathbf{y}) - \bar{\boldsymbol{\rho}})^T \boldsymbol{\Psi}] \end{aligned} \tag{17}$$

which is actually the Euclidean distance of a projected sample to the projected mean of the reference class $\mathcal{Y}$. This distance should be small for the samples of the class $\mathcal{Y}$ and big for the samples of the class $\tilde{\mathcal{Y}}$.

The discriminant measure used is the following:

$$\begin{aligned} J(\boldsymbol{\Psi}) &= \frac{1}{N(\tilde{\mathcal{Y}})} \sum_{\mathbf{y} \in \mathcal{Y}} \hat{c}(\mathbf{y}) - \frac{1}{N(\mathcal{Y})} \sum_{\mathbf{y} \in \tilde{\mathcal{Y}}} \hat{c}(\mathbf{y}) \\ &= \frac{1}{N(\tilde{\mathcal{Y}})} \sum_{\boldsymbol{\kappa}} \|\boldsymbol{\Psi}^T (\phi(\boldsymbol{\kappa}) - \bar{\boldsymbol{\rho}})\|^2 \\ &\quad - \frac{1}{N(\mathcal{Y})} \sum_{\boldsymbol{\rho}} \|\boldsymbol{\Psi}^T (\phi(\boldsymbol{\rho}) - \bar{\boldsymbol{\rho}})\|^2 \\ &= \frac{1}{N(\tilde{\mathcal{Y}})} \mathrm{tr} \left[\sum_{\boldsymbol{\kappa}} \boldsymbol{\Psi}^T (\phi(\boldsymbol{\kappa}) - \bar{\boldsymbol{\rho}})(\phi(\boldsymbol{\kappa}) - \bar{\boldsymbol{\rho}})^T \boldsymbol{\Psi} \right] \\ &\quad - \frac{1}{N(\mathcal{Y})} \mathrm{tr} \left[\sum_{\boldsymbol{\rho}} \boldsymbol{\Psi}^T (\phi(\boldsymbol{\rho}) - \bar{\boldsymbol{\rho}})(\phi(\boldsymbol{\rho}) - \bar{\boldsymbol{\rho}})^T \boldsymbol{\Psi} \right] \\ &= \mathrm{tr}[\boldsymbol{\Psi}^T \mathbf{W}^\Phi \boldsymbol{\Psi}] - \mathrm{tr}[\boldsymbol{\Psi}^T \mathbf{B}^\Phi \boldsymbol{\Psi}] \\ &= \mathrm{tr}[\boldsymbol{\Psi}^T (\mathbf{W}^\Phi - \mathbf{B}^\Phi) \boldsymbol{\Psi}] \end{aligned} \tag{18}$$

where $\mathrm{tr}[.]$ is the trace operator and the matrices $\mathbf{W}$ and $\mathbf{B}$ are given by

$$\mathbf{W}^\Phi = \frac{1}{N(\tilde{\mathcal{Y}})} \sum_{\boldsymbol{\kappa}} (\phi(\boldsymbol{\kappa}) - \bar{\boldsymbol{\rho}})(\phi(\boldsymbol{\kappa}) - \bar{\boldsymbol{\rho}})^T$$

and

$$\mathbf{B}^\Phi = \frac{1}{N(\mathcal{Y})} \sum_{\boldsymbol{\rho}} (\phi(\boldsymbol{\rho}) - \bar{\boldsymbol{\rho}})(\phi(\boldsymbol{\rho}) - \bar{\boldsymbol{\rho}})^T. \tag{19}$$

The discriminant measure (18) increases when the samples of the class $\tilde{\mathcal{Y}}$ are far from the center of the class $\mathcal{Y}$ and/or when the samples of the class $\mathcal{Y}$ are close to their center.

By additional requiring $\boldsymbol{\psi}_m^T \boldsymbol{\psi}_m = 1$, we can formulate the discriminant criterion for feature extraction as

$$\begin{aligned} &\max J(\boldsymbol{\Psi}) = \mathrm{tr}[\boldsymbol{\Psi}^T (\mathbf{W}^\Phi - \mathbf{B}^\Phi) \boldsymbol{\Psi}] \\ &= \sum_{m=1}^{K} \boldsymbol{\psi}_m^T (\mathbf{W}^\Phi - \mathbf{B}^\Phi) \boldsymbol{\psi}_m \\ &\text{subject to } \boldsymbol{\psi}_m^T \boldsymbol{\psi}_m - 1 = 0 \; m = 1, \cdots, K. \end{aligned} \tag{20}$$

The optimal $\boldsymbol{\Psi}$ can be found by the saddle point of the Lagrangian

$$\begin{aligned} L(\boldsymbol{\psi}_m, \lambda_m) = &\sum_{m=1}^{K} \boldsymbol{\psi}_m^T (\mathbf{W}^\Phi - \mathbf{B}^\Phi) \boldsymbol{\psi}_m \\ &- \lambda_m \left(\boldsymbol{\psi}_m^T \boldsymbol{\psi}_m - 1 \right) \end{aligned} \tag{21}$$

with λ_m be the Lagrangian multipliers. According to the KKT conditions, we have

$$\begin{aligned} &\nabla L(\boldsymbol{\psi}_m, \boldsymbol{\lambda})|_{\boldsymbol{\psi} = \boldsymbol{\psi}_o} = 0 \Leftrightarrow \\ &((\mathbf{W}^\Phi - \mathbf{B}^\Phi) - \lambda_m \mathbf{I}) \boldsymbol{\psi}_m = 0, m = 1, \cdots, K \Leftrightarrow \\ &(\mathbf{W}^\Phi - \mathbf{B}^\Phi) \boldsymbol{\psi}_m = \lambda_m \boldsymbol{\psi}_m \end{aligned} \tag{22}$$

which means that the Lagrangian multipliers λ_m are the eigenvalues of $\mathbf{W}^\Phi - \mathbf{B}^\Phi$ and the vectors $\boldsymbol{\psi}_m$ are the corresponding eigenvectors. By substituting (22) to (20), the criterion $J(\boldsymbol{\Psi})$ can be reformulated as

$$\begin{aligned} J(\boldsymbol{\Psi}) &= \sum_{m=1}^{K} \boldsymbol{\psi}_m^T (\mathbf{W}^\Phi - \mathbf{B}^\Phi) \boldsymbol{\psi}_m \\ &= \sum_{m=1}^{K} \lambda_m \boldsymbol{\psi}_m^T \boldsymbol{\psi}_m = \sum_{k=1}^{K} \lambda_m. \end{aligned} \tag{23}$$

Thus, $J(\boldsymbol{\Psi})$ is maximized when the columns of the matrix $\boldsymbol{\Psi}$ are composed of the d largest eigenvectors of $\mathbf{W}^\Phi - \mathbf{B}^\Phi$.

Since the matrices $\mathbf{W}^\Phi$ and $\mathbf{B}^\Phi$ are of arbitrary dimension, it is not possible to calculate them directly in practice. We will combine the theory in [41], [34], [46] to find a robust and time efficient solution of the defined optimization problem. Fist, let us define the matrix $\mathbf{S}^\Phi$ as

$$\begin{aligned} \mathbf{S}^\Phi &= N(\tilde{\mathcal{Y}}) \mathbf{W}^\Phi + N(\mathcal{Y}) \mathbf{B}^\Phi \\ &= \sum_{\boldsymbol{\kappa}} (\phi(\boldsymbol{\kappa}) - \bar{\boldsymbol{\rho}})(\phi(\boldsymbol{\kappa}) - \bar{\boldsymbol{\rho}})^T \\ &\quad + \sum_{\boldsymbol{\rho}} (\phi(\boldsymbol{\rho}) - \bar{\boldsymbol{\rho}})(\phi(\boldsymbol{\kappa}) - \bar{\boldsymbol{\rho}})^T \\ &= \sum_{\mathbf{y} \in \mathcal{Y} \cup \tilde{\mathcal{Y}}} (\phi(\mathbf{y}) - \bar{\boldsymbol{\rho}})(\phi(\mathbf{y}) - \bar{\boldsymbol{\rho}})^T \\ &= \sum_{i=1}^{L} \tilde{\boldsymbol{\mu}}_i \tilde{\boldsymbol{\mu}}_j^T = \boldsymbol{\Phi}_s \boldsymbol{\Phi}_s^T \end{aligned} \tag{24}$$

where $\tilde{\boldsymbol{\mu}}_i = \phi(\mathbf{y}_i) - \bar{\boldsymbol{\rho}}$ and $\boldsymbol{\Phi}_s = [\tilde{\boldsymbol{\mu}}_1 \dots \tilde{\boldsymbol{\mu}}_n]$. It can be easily proven that the matrix $\mathbf{S}^{\Phi}$ is compact, self-adjoint and positive operator in $\mathcal{H}$, thus its eigenvector system forms a basis of $\mathcal{H}$[1]. Let the two complementary spaces $\mathcal{B}$ and $\mathcal{B}^{\perp}$ spanned by the orthonormal eigenvectors that correspond to the non-null and the null eigenvalues of $\mathbf{S}^{\Phi}$ (the columns of the matrices $\boldsymbol{\Phi}_s$ and $\tilde{\boldsymbol{\Phi}}_s$, respectively). Every vector $\boldsymbol{\psi}_m$ can be written, in a unique manner, as $\boldsymbol{\psi}_m = \boldsymbol{\gamma}_m + \boldsymbol{\delta}_m$, or equivalently as $\boldsymbol{\psi}_m = \boldsymbol{\Phi}_s\boldsymbol{\zeta}_m + \tilde{\boldsymbol{\Phi}}_s\boldsymbol{\eta}_m$, where $\boldsymbol{\gamma}_m = \boldsymbol{\Phi}_s\boldsymbol{\zeta}_m \in \mathcal{B}$ and $\boldsymbol{\delta}_m = \tilde{\boldsymbol{\Phi}}_s\boldsymbol{\eta}_m \in \mathcal{B}^{\perp}$. Moreover, the space $\mathcal{B}$ is isomorphic to $\Re^{n-1}$ (i.e., $\boldsymbol{\zeta}_m \in \Re^{n-1}$), and the $\mathcal{B}^{\perp}$ to the $\mathcal{H}$ minus the $n-1$ dimensions. Equation (22) can be expanded as

$$(\mathbf{W}^{\Phi} - \mathbf{B}^{\Phi})\boldsymbol{\Phi}_s\boldsymbol{\zeta}_m = \lambda_m(\boldsymbol{\Phi}_s\boldsymbol{\zeta}_m + \tilde{\boldsymbol{\Phi}}_s\boldsymbol{\eta}_m). \tag{25}$$

By multiplying with $\boldsymbol{\Phi}_s^T$, we have

$$\left(\boldsymbol{\Phi}_s^T\mathbf{W}^{\Phi}\boldsymbol{\Phi}_s - \boldsymbol{\Phi}_s^T\mathbf{B}^{\Phi}\boldsymbol{\Phi}_s\right)\boldsymbol{\zeta}_m = \lambda_m\boldsymbol{\zeta}_m \tag{26}$$

and by multiplying with $\tilde{\boldsymbol{\Phi}}_s^T$, we have

$$0 = \lambda_m\boldsymbol{\eta}_m. \tag{27}$$

Thus, the $\boldsymbol{\eta}_m$ do not play any role in the optimization problem. The above analysis is similar to the one presented in [34], where it has been shown that the space of the vectors $\boldsymbol{\eta}_m$ does not play any role in the optimization of the Fisher discriminant ratio with kernels.

Summarizing, it has been shown that only the first L (with $L \leq n-1$) positive eigenvalues of $\mathbf{S}^{\Phi}$ are of interest to us. These eigenvectors can be indirectly derived from the eigenvectors of the matrix $\boldsymbol{\Phi}_s^T\boldsymbol{\Phi}_s (L \times L)$. Let λ_i^s and $\mathbf{c}_i (i = 1 \dots L)$ be the i-th eigenvalue and the corresponding eigenvector of $\boldsymbol{\Phi}_s^T\boldsymbol{\Phi}_s$, sorted in ascending order of eigenvalues. It is true that $(\boldsymbol{\Phi}_s\boldsymbol{\Phi}_s^T)(\boldsymbol{\Phi}_s\boldsymbol{\varpi}_i) = \lambda_i^s(\boldsymbol{\Phi}_s\mathbf{c}_i)$. Thus, $\boldsymbol{\varpi}_i = \boldsymbol{\Phi}_s\mathbf{c}_i$ are the eigenvectors of $\mathbf{S}^{\Phi}$. In order to remove the null space of $\mathbf{S}^{\Phi}$, the first $L \leq n-1$ eigenvectors (given in the matrix $\boldsymbol{\Pi} = [\boldsymbol{\varpi}_1 \dots \boldsymbol{\varpi}_L] = \boldsymbol{\Phi}_s\mathbf{C}$, where $\mathbf{C} = [\mathbf{c}_1 \dots \mathbf{c}_L]$), whose corresponding eigenvalues are non zero, should be calculated. Thus, $\boldsymbol{\Pi}^T\mathbf{S}^{\Phi}\boldsymbol{\Pi} = \boldsymbol{\Lambda}_s$, with $\boldsymbol{\Lambda}_s = \mathrm{diag}[{\lambda_1^s}^2 \dots {\lambda_L^s}^2]$, a $L \times L$ diagonal matrix. The orthonormal eigenvectors of $\mathbf{S}^{\Phi}$ are the columns of the matrix

$$\boldsymbol{\Pi}_1 = \boldsymbol{\Phi}_s\boldsymbol{\Pi}{\boldsymbol{\Lambda}_{\mathbf{s}}}^{-1/2}. \tag{28}$$

After projecting all the training vectors to $\boldsymbol{\Pi}_1$, the optimization problem reduces to finding the eigenvectors of $\mathbf{W} - \mathbf{B}$, where $\mathbf{W} = \boldsymbol{\Pi}_1^T\mathbf{W}^{\Phi}\boldsymbol{\Pi}_1$ and $\mathbf{B} = \boldsymbol{\Pi}_1^T\mathbf{B}^{\Phi}\boldsymbol{\Pi}_1$.

A. Feature Extraction From the Two-Class Kernel Procedure

We can now summarize the training procedure of the proposed algorithm:

[1]The matrix $\mathbf{S}^{\Phi}$ is not to be confused with the total scatter matrix. In $\mathbf{S}^{\Phi}$ the intra class mean is subtracted from all the training vectors, while in the total scatter matrix case the total mean vector is subtracted from the training vectors

Step 1) Calculate the nonzero eigenvalues and the eigenvectors of $\boldsymbol{\Phi}_s^T\boldsymbol{\Phi}_s$ and project each facial vector $\mathbf{y}$ as

$$\begin{aligned}
\boldsymbol{\Pi}_1^T\phi(\mathbf{y}) &= (\boldsymbol{\Pi}{\boldsymbol{\Lambda}_{\mathbf{s}}}^{-1/2})^T\boldsymbol{\Phi}_s^T\phi(\mathbf{y}) \\
&= (\boldsymbol{\Pi}{\boldsymbol{\Lambda}_{\mathbf{s}}}^{-1/2})^T[\tilde{\boldsymbol{\mu}}_1 \dots \tilde{\boldsymbol{\mu}}_n]^T\phi(\mathbf{y}) \\
&= (\boldsymbol{\Pi}{\boldsymbol{\Lambda}_{\mathbf{s}}}^{-1/2})^T([\phi(\mathbf{y}_1) \dots \phi(\mathbf{y}_n)]^T\phi(\mathbf{y}) \\
&\quad - [\bar{\boldsymbol{\rho}} \dots \bar{\boldsymbol{\rho}}]^T\phi(\mathbf{y})) \\
&= (\boldsymbol{\Pi}{\boldsymbol{\Lambda}_{\mathbf{s}}}^{-1/2})^T([\phi(\mathbf{y}_1) \dots \phi(\mathbf{y}_n)]^T\phi(\mathbf{y}) \\
&\quad - \frac{1}{N(\mathcal{Y})}\mathbf{1}_{nN(\mathcal{Y})}[\phi(\boldsymbol{\rho}_1) \dots \phi(\boldsymbol{\rho}_{N(\mathcal{Y})})]^T\phi(\mathbf{y}_i)) \\
&= (\boldsymbol{\Pi}{\boldsymbol{\Lambda}_{\mathbf{s}}}^{-1/2})^T([k(\mathbf{y}_1, \mathbf{y}) \dots k(\mathbf{y}_n, \mathbf{y})]^T \\
&\quad - \frac{1}{N(\mathcal{Y})}\mathbf{1}_{nN(\mathcal{Y})}[k(\boldsymbol{\rho}_1, \mathbf{y}) \dots k(\boldsymbol{\rho}_{N(\mathcal{Y})}, \mathbf{y})])
\end{aligned} \tag{29}$$

where $\mathbf{1}_{n_1n_2}$ is a $n_1 \times n_2$ matrix of ones.

Step 2) In the new space, calculate $\mathbf{W}$ and $\mathbf{B}$. Perform eigenanalysis to $\mathbf{W} - \mathbf{B}$ and obtain a set of K orthonormal eigenvectors. The eigenvectors are stored in a matrix $\boldsymbol{\Xi} \in \Re^{(n-1)\times K}$.

After following these steps, the discriminant projection for a test vector $\mathbf{y}$ is given by

$$\begin{aligned}
\acute{\mathbf{y}} = (\boldsymbol{\Pi}{\boldsymbol{\Lambda}_{\mathbf{s}}}^{-1/2}\boldsymbol{\Xi})^T([k(\mathbf{y}_1, \mathbf{y}) \dots k(\mathbf{y}_n, \mathbf{y})]^T \\
- \frac{1}{N(\mathcal{Y})}\mathbf{1}_{nN(\mathcal{Y})}[k(\boldsymbol{\rho}_1, \mathbf{y}) \dots k(\boldsymbol{\rho}_{N(\mathcal{Y})}, \mathbf{y})])
\end{aligned} \tag{30}$$

and the number of dimensions of the discriminant vectors $\acute{\mathbf{y}} \in \Re^K$ is $K \leq n-1$.

V. Experimental Results

The Cohn–Kanade database [35] was used for the facial expression recognition in six basic facial expressions classes (anger, disgust, fear, happiness, sadness, and surprise). This database is annotated with FAUs. These combinations of FAUs were translated into facial expressions according to [47], in order to define the corresponding ground truth for the facial expressions.

For learning the reference bunch graphs and discriminant transforms we have used 80% of the data while the remaining 20% have been used for testing. More specifically, all image sequences contained in the database are divided into six classes, each one corresponding to one of the six basic facial expressions to be recognized. Five sets containing 20% of the data for each class, chosen randomly, are created. One set containing 20% of the samples for each class is used for the test set, while the remaining sets form the training set. After the classification procedure is performed, the samples forming the testing 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. The average classification accuracy is the mean value of the percentages of the correctly classified facial expressions. In order to decide which of the six facial classes a test facial image belongs to, the six reference facial expression graphs are matched to the test

TABLE I
RECOGNITION RATE FOR THE TESTED FEATURES

Features	Graph Structure	Recognition Rate
Gabor-based	Rectangular	84.1%
Normalized Morphological	Rectangular	85.5%

TABLE II
RECOGNITION RATE FOR THE TESTED FEATURES

Features	Graph Structure	Recognition Rate
Gabor-based	Rectangular	84.1%
Gabor-based	Discriminant Graph Structure	90.5%
Normalized Morphological	Rectangular	85.5%
Normalized Morphological	Discriminant Graph Structure	91.8%

images and the one with the minimum distance is the winner class.

We have conducted three series of experiments. In the first, we wanted to measure the way the choice of the multiscale analysis affects the recognition performance. In the second, we evaluated the use of grids placed in discriminant facial landmarks for both Gabor-based graphs and Morphological graphs. Finally, we have explored the contribution of the proposed discriminant analysis with respect to the performance.

A. Experiments With Different Multiscale Analysis

We have implemented a Gabor-based elastic graph matching approach similar to the one applied in [11] and [29], as well as the proposed elastic graph matching procedure with the normalized morphological features. For the Gabor features, six orientations and five scales have been considered giving a total of 30 features per node, while for the morphological features nine scales have been considered with a total of 19 features. The graphs that have been used have been 8×8 evenly distributed graphs, like the ones depicted in Fig. 1. Table I summarizes the experimental results. As can be seen, the normalized morphological features perform better than the Gabor-based.

B. Experiments With Different Graph Structures

We have conducted experiments using the optimal graph structures that have been derived from the algorithm in Section III. All the tested approaches are fully automatic since only a face detector is needed in order to initialize the graph. Afterwards, elastic graph matching is applied for a finest matching. Thus, we have conducted no experiments with graphs placed at manually selected facial landmarks like the ones proposed in [11]. The facial expression recognition rates for the rectangular graph structures and the discriminant structures learned from the proposed approach for both Gabor-based and morphological-based graphs are summarized in Table II. As can be seen, the proposed discriminant graphs outperform the rectangular graphs for both Normalized Morphological features and Gabor-based features. Hence, these graphs have indeed captured landmarks that are discriminant for facial expression recognition.

C. Experiments With Various Discriminant Analysis and Strategies

Finally, we have conducted experiments in order to evaluate the performance of facial expression recognition using the proposed discriminant analysis. For comparison reasons, we have implemented a Gabor-based elastic graph matching approach with the typical FLDA classifier in the same manner as proposed in [11] and [29]. The main limitation of the FLDA discriminant analysis and its kernel alternatives [33], [34], [41] is that it gives only one or two discriminant projection in two class problems. That is, the so-called direct kernel discriminant analysis (DKDA) gives only one discriminant vector [33] and the so-called complete kernel discriminant analysis (CFKDA) [34] two discriminant vectors. This is obviously a limitation in the search for discriminant features. The proposed discriminant analysis gives up to n discriminant dimensions, where n is the number of training samples.

We applied the proposed discriminant analysis, described in Section IV, and KFDA [41] using both node-wise and graph-wise discriminant transforms. Fig. 7(a) shows the recognition performance of the proposed discriminant analysis for both Gabor-based and morphological features using polynomial kernels with degrees from 1 to 4 for the graph-wise strategy. We have experimented with polynomial kernels with degrees more than 4, but we have seen no further improvement in performance. As can be seen, the proposed discriminant analysis outperforms the KFDA.

The corresponding results for the node-wise strategy are shown in Fig. 7(b). It can be seen that the node-wise strategies (i.e., learning a discriminant transform for every node and for every facial expression) have better performance than the graph-wise strategies. The best performance that the proposed system achieved was 97.1% and has been measured when we selected graphs with normalized morphological filters using the optimal graph structures that have derived from the algorithm in Section III and applying the proposed discriminant analysis in a node-wise manner.

D. Comparison With State-Of-The-Art

Recently, [6] a facial expression recognition system has been described that has been tested in Cohn–Kanade dat abase using a similar experimental protocol. The system in [6] has shown superior performance and has achieved 99.7% recognition rate. The drawback of the system in [6] is that it requires the Candide grid to be manually placed upon the facial area and moreover it requires the manual detection of the neutral state in a video sequence. On the other hand, the proposed method is fully automatic and does not require the detection of the neutral state. Moreover, it can be used for both image and video based facial expression recognition contrary to [6] that requires the whole video of facial expression development from the neutral state to the fully expressive image. Finally, the proposed method can be adjusted to EGM-based tracking systems like the ones proposed in [48] and [49] in order to achieve facial expression recognition in video sequences.

Apart from the method proposed in [6], a comparison of the recognition rates achieved for each facial expression with the state-of-the-art [40], [50]–[55] when six or seven facial expressions were examined (the neutral state is the seventh facial expression) is depicted at Table III. The total facial expression recognition of the proposed method has been 97.1% for the six facial expressions. Unfortunately, there is not a direct method to

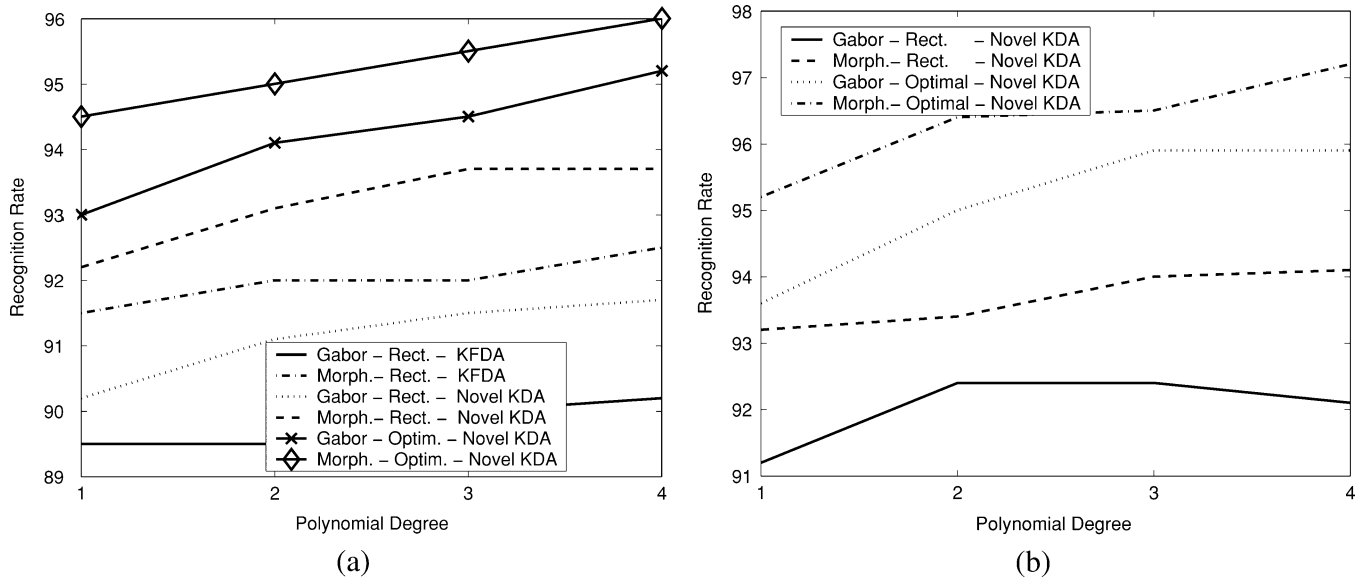


Fig. 7. Recognition rates for various polynomial kernels: (a) graph-wise strategies and (b) node-wise strategies.

TABLE III
COMPARISON OF FACIAL EXPRESSION RECOGNITION ALGORITHMS IN THE COHN–KANADE DATABASE

Method	Number of Sequences	Number of Classes	Recognition Rate
[50]	320	7(6)	88.4(92.1)%
[51]	313	7	86.9%
[52]	313	7	93.8%
[53]	375	6	93.8%
[54]	-	6	90.9%
[40]	284	6	93.66%
[55]	374	6	96.26%
Proposed	374	6	**97.1**%

compare the rates achieved by other researchers [56], [55], since their is not standard protocol (every one uses his own testing protocol). Nevertheless, the proposed method has achieved the best recognition rate among the recent state-of-the-art methods. The second best method has been the one proposed in [55], where a total 96.7% recognition rate has been achieved using a method based on local binary patterns and SVM classifiers. The main drawback of the method in [55] is that it is only tested in perfect manually aligned image sequences and no experiments in fully automatic conditions have been presented.

VI. CONCLUSION

We have meticulously studied the use of elastic graph matching for facial expression recognition and motivated the use of morphological features for facial expression representation. We have applied a discriminant analysis that learns the optimal graph structure for every facial expression. Finally, we have proposed a novel nonlinear discriminant analysis for both graph-wise and node-wise feature selection. We have experimented on both Gabor-based and Normalized Morphological elastic graph matching architectures and we have been applied them in a fully automatic manner to achieve facial expression recognition. The experimental results show that the proposed methods significantly increase the performance of both Gabor and morphological EGM in facial expression recognition.

APPENDIX A
COMPUTATION OF $\mathbf{\Phi}_s^T\mathbf{\Phi}_s$

Before proceeding to the expansion, we should define the following matrices:

$$\begin{aligned}
[\mathbf{K}_1]_{i,j} &= \phi(\boldsymbol{\rho}_i)^T\phi(\boldsymbol{\rho}_j) = k(\boldsymbol{\rho}_i,\boldsymbol{\rho}_j) \\
&\qquad i=1\ldots N(\mathcal{Y}) \text{ and } j=1\ldots N(\mathcal{Y}) \\
[\mathbf{K}_2]_{i,j} &= \phi(\boldsymbol{\kappa}_i)^T\phi(\boldsymbol{\rho}_j) = k(\boldsymbol{\kappa}_i,\boldsymbol{\rho}_j) \\
&\qquad i=1\ldots N(\tilde{\mathcal{Y}}) \text{ and } j=1\ldots N(\mathcal{Y}) \\
[\mathbf{K}_3]_{i,j} &= \phi(\boldsymbol{\rho}_i)^T\phi(\boldsymbol{\kappa}_j) = k(\boldsymbol{\rho}_i,\boldsymbol{\kappa}_j) = \mathbf{K}_2^T \\
[\mathbf{K}_4]_{i,j} &= \phi(\boldsymbol{\kappa}_i)^T\phi(\boldsymbol{\kappa}_j) = k(\boldsymbol{\kappa}_i,\boldsymbol{\kappa}_j) \\
&\qquad i=1\ldots N(\tilde{\mathcal{Y}}) \text{ and } j=1\ldots N(\tilde{\mathcal{Y}}) \qquad (31)
\end{aligned}$$

the matrix $\mathbf{K}$ as the total kernel function

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_4 & \mathbf{K}_2 \\ \mathbf{K}_3 & \mathbf{K}_1 \end{bmatrix} \qquad (32)$$

and the $\mathbf{E}$ as

$$\mathbf{E} = \begin{bmatrix} \mathbf{K}_2 \\ \mathbf{K}_3 \end{bmatrix}. \qquad (33)$$

The $\mathbf{\Phi}_s^T\mathbf{\Phi}_s$ is expanded as

$$\mathbf{\Phi}_s^T\mathbf{\Phi}_s = [\tilde{\boldsymbol{\mu}}_1\ldots\tilde{\boldsymbol{\mu}}_n]^T[\tilde{\boldsymbol{\mu}}_1\ldots\tilde{\boldsymbol{\mu}}_n] = [\tilde{\boldsymbol{\mu}}_i^T\tilde{\boldsymbol{\mu}}_j] \qquad (34)$$

where

$$\begin{aligned}
&\tilde{\boldsymbol{\mu}}_i^T \tilde{\boldsymbol{\mu}}_j \\
&\quad = \phi(\mathbf{y}_i)^T \phi(\mathbf{y}_j) - \phi(\mathbf{y}_i)^T \bar{\boldsymbol{\rho}} - \bar{\boldsymbol{\rho}}^T \phi(\mathbf{y}_j) + \bar{\boldsymbol{\rho}}^T \bar{\boldsymbol{\rho}} \\
&\quad = [\mathbf{K}]_{i,j} - \frac{1}{N(\mathcal{Y})} \sum_{m=1}^{N(\mathcal{Y})} \phi(\mathbf{y}_i)^T \phi(\boldsymbol{\rho}_m) \\
&\qquad - \frac{1}{N(\mathcal{Y})} \sum_{m=1}^{N(\mathcal{Y})} \phi(\boldsymbol{\rho}_m)^T \phi(\mathbf{y}_j) \\
&\qquad + \frac{1}{N(\mathcal{Y})^2} \sum_{m=1}^{N(\mathcal{Y})} \phi(\boldsymbol{\rho}_m)^T \sum_{m=1}^{N(\mathcal{Y})} \phi(\boldsymbol{\rho}_m) \\
&\quad = [\mathbf{K}]_{i,j} - \left[\frac{1}{N(\mathcal{Y})} \mathbf{E} \mathbf{1}_{N(\mathcal{Y})n}\right]_{i,j} - \left[\frac{1}{N(\mathcal{Y})} \mathbf{1}_{nN(\mathcal{Y})} \mathbf{E}\right]_{i,j} \\
&\qquad + \left[\frac{1}{N(\mathcal{Y})^2} \mathbf{1}_{nN(\mathcal{Y})} \mathbf{K}_1 \mathbf{1}_{N(\mathcal{Y})n}\right]_{i,j} \\
&\quad = \left[\mathbf{K} - \frac{1}{N(\mathcal{Y})} \mathbf{E} \mathbf{1}_{N(\mathcal{Y})n} - \frac{1}{N(\mathcal{Y})} \mathbf{1}_{nN(\mathcal{Y})} \mathbf{E} \right. \\
&\qquad \left. + \frac{1}{N(\mathcal{Y})^2} \mathbf{1}_{nN(\mathcal{Y})} \mathbf{K}_1 \mathbf{1}_{N(\mathcal{Y})n}\right]_{i,j}. \qquad (35)
\end{aligned}$$

REFERENCES

[1] A. Pentland and T. Choudhury, "Face recognition for smart environments," *IEEE Comput.*, vol. 33, no. 2, pp. 50–55, Feb. 2000.

[2] M. Pantic and L. J. M. Rothkrantz, "Toward an affect-sensitive multimodal human-computer interaction," *Proc. IEEE*, vol. 91, no. 9, pp. 1370–1390, Sep. 2003.

[3] P. Ekman and W. V. Friesen, *Emotion in the Human Face*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[4] M. Pantic and L. J. M. Rothkrantz, "Automatic analysis of facial expressions: The state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1424–1445, Dec. 2000.

[5] B. Fasel and J. Luettin, "Automatic facial expression analysis: A survey," *Pattern Recognit.*, vol. 36, no. 1, pp. 259–275, 2003.

[6] I. Kotsia and I. Pitas, "Facial expression recognition in image sequences using geometric deformation features and support vector machines," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 172–187, Jan. 2007.

[7] Z. Zeng, M. Pantic, G. Roisman, and T. Huang, "A survey of affect recognition methods: Audio, visual, and spontaneous expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, accepted for publication.

[8] M. Rydfalk, CANDIDE: A Parameterized Face Linkoping Univ., Linkoping, Sweden, 1978, Tech. Rep..

[9] G.-D. Guo and C. R. Dyer, "Learning from examples in the small sample case: Face expression recognition," *IEEE Trans. Syst., Man, Cybern. B: Cybern.*, vol. 35, no. 3, pp. 479–488, Jun. 2005.

[10] Y. Zhang and J. Qiang, "Active and dynamic information fusion for facial expression understanding from image sequences," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 699–714, May 2005.

[11] M. J. Lyons, J. Budynek, and S. Akamatsu, "Automatic classification of single facial images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 12, pp. 1357–1362, Dec. 1999.

[12] M. Lades, J. C. Vorbrilggen, J. Buhmann, J. Lange, C. V. D. Malsburg, R. P. Würtz, and W. Konen, "Distortion invariant object recognition in the dynamic link architecture," *IEEE Trans. Comput.*, vol. 42, no. 3, pp. 300–311, Mar. 1993.

[13] J. Zhang, Y. Yan, and M. Lades, "Face recognition: Eigenface, elastic matching, and neural nets," *Proc. IEEE*, vol. 85, no. 9, pp. 1423–1435, Sep. 1997.

[14] L. Wiskott, J. Fellous, N. Krüger, and C. V. D. Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 775–779, Jul. 1997.

[15] L. Wiskott, "Phantom faces for face analysis," *Pattern Recognit.*, vol. 30, no. 6, pp. 837–846, 1997.

[16] R. P. Wurtz, "Object recognition robust under translations, deformations, and changes in background," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 769–775, Jul. 1997.

[17] B. Due, S. Fischer, and J. Bigün, "Face authentication with Gabor information on deformable graphs," *IEEE Trans. Image Process.*, vol. 8, no. 4, pp. 504–516, Apr. 1999.

[18] C. Kotropoulos, A. Tefas, and I. Pitas, "Frontal face authentication using discriminating grids with morphological feature vectors," *IEEE Trans. Multimedia*, vol. 2, no. 1, pp. 14–26, Mar. 2000.

[19] C. Kotropoulos, A. Tefas, and I. Pitas, "Morphological elastic graph matching applied to frontal face authentication under well-controlled and real conditions," *Pattern Recognit.*, vol. 33, no. 12, pp. 31–43, Oct. 2000.

[20] C. Kotropoulos, A. Tefas, and I. Pitas, "Frontal face authentication using morphological elastic graph matching," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 555–560, Apr. 2000.

[21] A. Tefas, C. Kotropoulos, and I. Pitas, "Face verification using elastic graph matching based on morphological signal decomposition," *Signal Process.*, vol. 82, no. 6, pp. 833–851, 2002.

[22] S. Zafeiriou, A. Tefas, and I. Pitas, "Exploiting discriminant information in elastic graph matching," in *Proc. IEEE Int. Conf. Image Processing (ICIP 2005)*, Geneva, Italy, Sep. 11–14, 2005.

[23] H.-C. Shin, J. H. Park, and S.-D. Kim, "Combination of warping robust elastic graph matching and kernel-based projection discriminant analysis for face recognition," *IEEE Trans. Multimedia*, vol. 9, no. 6, pp. 1125–1136, Oct. 2007.

[24] H.-C. Shin, S.-D. Kim, and H.-C. Choi, "Generalized elastic graph matching for face recognition," *Pattern Recognit. Lett.*, vol. 28, no. 9, pp. 1077–1082, July 2007.

[25] N. Krüger, "An algorithm for the learning of weights in discrimination functions using A priori constraints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 764–768, Jul. 1997.

[26] H. Hong, H. Neven, and C. V. D. Malsburg, "Online facial expression recognition based on personalized gallery," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Nara, Japan, 14–16, 1998, pp. 354–359.

[27] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Nara, Japan, 14–16, 1998, pp. 200–205.

[28] Z. Zhang, M. Lyons, M. Schuster, and S. Akamatsu, "Comparison between geometry-based and gabor-wavelets-based facialexpression recognition using multi-layer perceptron," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Nara, Japan, 14–16, 1998, pp. 454–459.

[29] M. J. Lyons, J. Budynek, A. Plante, and S. Akamatsu, "Classifying facial attributes using a 2-d gabor wavelet representation and discriminant analysis," in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Mar. 28–30, 2000, pp. 202–207.

[30] A. Tefas, C. Kotropoulos, and I. Pitas, "Using support vector machines to enhance the performance of elastic graph matching for frontal face authentication," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 7, pp. 735–746, Jul. 2001.

[31] P. Yang, S. Shan, W. Gao, S. Z. Li, and D. Zhang, "Face recognition using ada-boosted gabor features," in *Proc. 6th IEEE Int. Conf. Automatic Face and Gesture Recognition*, Seoul, Korea, May 17–19, 2004.

[32] S. Zafeiriou, A. Tefas, and I. Pitas, "Learning discriminant person specific facial models using expandable graphs," *IEEE Trans. Inform. Forensics and Security*, vol. 2, no. 1, pp. 50–55, Mar. 2007.

[33] L. Juwei, K. N. Plataniotis, and A. N. Venetsanopoulos, "Face recognition using kernel direct discriminant analysis algorithms," *IEEE Trans. Neural Netw.*, vol. 14, no. 1, pp. 117–126, 2003.

[34] J. Yang, A. F. Frangi, J. Yang, D. Zhang, and Z. Jin, "KPCA plus LDA: A complete kernel Fisher discriminant framework for feature extraction and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 230–244, Feb. 2005.

[35] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proc. IEEE Int. Conf. Face and Gesture Recognition*, Mar. 2000, pp. 46–53.

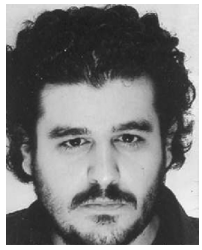
[36] S. Zafeiriou, A. Tefas, and I. Pitas, "Elastic graph matching versus linear subspace methods for frontal face verification," in *Proc. Int. Workshop on Nonlinear Signal and Image Processing*, 2005.

[37] P. T. Jackway and M. Deriche, "Scale-space properties of the multiscale morphological dilation-erosion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 1, pp. 38–51, Jan. 1996.

[38] K. Fukunaga, *Statistical Pattern Recognition*. San Diego, CA: Academic, 1990.

[39] S. Mitra, M. Savvides, and B. V. K. V. V. Kumar, "Face identification using novel frequency-domain representation of facial asymmetry," *IEEE Trans. Inform. Forensics and Security*, vol. 1, no. 3, pp. 350–359, 2006.

[40] S. P. Aleksic and K. A. Katsaggelos, "Automatic facial expression recognition using facial animation parameters and multi-stream hmms," *IEEE Trans. Inform. Forensics and Security*, vol. 1, no. 1, pp. 3–11, 2006.
[41] H. Li, T. Jiang, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 157–165, Jan. 2006.
[42] L. Juwei, K. N. Plataniotis, and A. N. Venetsanopoulos, "Face recognition using LDA-based algorithms," *IEEE Trans. Neural Netw.*, vol. 14, no. 1, pp. 195–200, 2003.
[43] K.-R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–201, 2001.
[44] B. Scholkopf, S. Mika, C. J. C. Surges, P. Knirsch, K.-R. Muller, G. Ratsch, and A. J. Smola, "Input space vs. feature space in kernel-based methods," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1000–1017, 1999.
[45] A. Scholkopf, B. Smola, and K. R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, pp. 1299–1319, 1998.
[46] J. Liu, S. Chen, X. Tan, and D. Zhang, "Comments on "efficient and robust feature extraction by maximum margin criterion"," *IEEE Trans. Neural Netw.*, vol. 6, no. 18, pp. 1862–1864, Nov. 2007.
[47] M. Pantic and L. J. M. Rothkrantz, "Expert system for automatic analysis of facial expressions," *Image and Vis. Comput.*, vol. 18, no. 11, pp. 881–905, Aug. 2000.
[48] G. Stamou, N. Nikolaidis, and I. Pitas, "Object tracking based on morphological elastic graph matching," in *Proc. IEEE Int. Conf. Image Processing (ICIP 2005)*, Geneva, Sep. 11–14, 2005.
[49] B. Li and R. Chellapa, "Face verification through tracking facial features," *J. Opt. Soc. Amer. A*, vol. 18, pp. 2969–2981, 2001.
[50] C. Shan, S. Gong, and P. W. McOwan, "Robust facial expression recognition using local binary patterns," in *Proc. ICIP 2005*, 2005, pp. 370–373.
[51] M. S. Bartlett, G. Littlewort, I. Fasel, and J. R. Movellan, "Real time face detection and facial expression recognition: Development and applications to human computer interaction," in *Proc. Conf. Computer Vision and Pattern Recognition Workshop*, Madison, WI, Jun. 16–22, 2003, vol. 5, pp. 53–58.
[52] G. Littlewort, M. Bartlett, I. Fasel, J. Susskind, and J. Movellan, "Dynamics of facial expression extracted automatically from video," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, Workshop on Face Processing in Video*, 2004.
[53] Y. Tian, "Evaluation of face resolution for expression analysis," in *Proc. IEEE Workshop Face Processing in Video 2004*, 2004.
[54] M. Yeasin, B. Bullot, and R. Sharma, "Recognition of facial expressions and measurement of levels of interest from video," *IEEE Trans. Multimedia*, vol. 8, no. 3, pp. 500–508, Jun. 2006.
[55] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 915–928, Jun. 2007.
[56] I. Cohen, N. Sebe, S. Garg, L. S. Chen, and T. S. Huanga, "Facial expression recognition from video sequences: Temporal and static modelling," *Comput. Vis. Image Understand.*, vol. 91, pp. 160–187, 2003.



Stefanos Zafeiriou was born in Thessaloniki, Greece, in 1981. He received the B.Sc. degree in informatics (with highest honors) in 2003 and the Ph.D. degree in informatics in 2007, both from the Aristotle University of Thessaloniki.

During 2007–2008, he was a Senior Researcher in the Department of Informatics, Aristotle University of Thessaloniki. Currently, he is a Senior Researcher in the Department of Electrical and Electronic Engineering, Imperial College London, London, U.K. He has co-authored over 30 journal and conference publications. His current research interests lie in the areas of signal and image processing, computational intelligence, pattern recognition, machine learning, computer vision and detection and estimation theory.

Dr. Zafeiriou has received various scholarships and awards during his studies.

Ioannis Pitas (F'07) received the Dipl. Elect. Eng. in 1980 and the Ph.D. degree in electrical engineering in 1985, both from the Aristotle University of Thessaloniki, Thessaloniki, Greece.

Since 1994, he has been a Professor in the Department of Informatics, Aristotle University of Thessaloniki. From 1980 to 1993, he served as Scientific Assistant, Lecturer, Assistant Professor, and Associate Professor in the Department of Electrical and Computer Engineering. He served as a Visiting Research Associate at the University of Toronto, Toronto, ON, Canada, University of Erlangen-Nuernberg, Germany, Tampere University of Technology, Tampere, Finland, as Visiting Assistant Professor at the University of Toronto and as Visiting Professor at the University of British Columbia, Vancouver, BC, Canada. He was a Lecturer in short courses for continuing education. He has published over 600 journal and conference papers and contributed in 22 books in his areas of interest. He is the co-author of the books *Nonlinear Digital Filters: Principles and Applications* (Norwell, MA: Kluwer, 1990), *3-D Image Processing Algorithms* (New York: Wiley, 2000), *Nonlinear Model-Based Image/Video Processing and Analysis* (New York, Wiley, 2001), and author of *Digital Image Processing Algorithms and Applications* (New York: Wiley, 2000). He is the editor of the book *Parallel Algorithms and Architectures for Digital Image Processing, Computer Vision and Neural Networks* (New York: Wiley, 1993). His current interests are in the areas of digital image and video processing and analysis, multidimensional signal processing, watermarking, and computer vision.

Dr. Pitas has been an invited speaker and/or member of the program committee of several scientific conferences and workshops. He has served as Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, IEEE TRANSACTIONS ON NEURAL NETWORKS, IEEE TRANSACTIONS ON IMAGE PROCESSING, and the *EURASIP Journal on Applied Signal Processing*, and as co-editor of *Multidimensional Systems and Signal Processing*. He was General Chair of the 1995 IEEE Workshop on Nonlinear Signal and Image Processing (NSIP95) and the IEEE ICIP 2001 and Technical Chair of the 1998 European Signal Processing Conference.