# Salient Feature and Reliable Classifier Selection for Facial Expression Classification

*Marios Kyperountas*[*,a] , Anastasios Tefas[a], and Ioannis Pitas[a,b]*

[a] Aristotle University of Thessaloniki

Department of Informatics

Artificial Intelligence and Information Analysis Laboratory

Box 451, 54006 Thessaloniki, Greece

Email: {mkyper@aiia.csd.auth.gr, pitas@aiia.csd.auth.gr }

[b] Informatics and Telematics Institute,

Center of Research and Technology Hellas,

Thessaloniki, Greece

* Corresponding author:

Present address:

175H San Angelo Avenue

Santa Barbara, CA 93111

Tel. +1 (805) 259-6725

Fax +1 (805) 968-2315

Email: mkyper@aiia.csd.auth.gr

**Abstract**

A novel facial expression classification (FEC) method is presented and evaluated. The classification process is decomposed into multiple two-class classification problems, a choice that is analytically justified, and unique sets of features are extracted for each classification problem. Specifically, for each two-class problem, an iterative feature selection process that utilizes a class separability measure is employed to create Salient Feature Vectors (SFVs), where each SFV is composed of a selected feature subset. Subsequently, two-class discriminant analysis is applied on the SFVs to produce Salient Discriminant Hyper-planes (SDHs), which are used to train the corresponding two-class classifiers. To properly integrate the two-class classification results and produce the FEC decision, a computationally efficient and fast classification scheme is developed. During each step of this scheme, the most reliable classifier is identified and utilized, thus, a more accurate final classification decision is produced. The JAFFE and the MMI databases are used to evaluate the performance of the proposed Salient-Feature-and-Reliable-Classifier Selection (SFRCS) methodology. Classification rates of 96.71% and 93.61% are achieved under the leave-one-sample-out evaluation strategy, and 85.92% under the leave-one-subject-out evaluation strategy.

**Index Terms**: Facial expression classification, salient feature selection, classifier selection, two-class classification.

## 1. INTRODUCTION

Facial expression classification (FEC) is a type of non-verbal communication that has attracted significant attention from the scientific community over the last few years [1, 2, and 3]. This resulted from the need to develop automatic and, in some cases, real-time human centered interfaces, where the face plays a crucial role [4, 5]. Examples of applications that use FEC are facial expression cloning in virtual reality applications, video-conferencing, and user profiling, indexing, and retrieval from image and video databases. Facial expressions play a very important role in human face-to-face interpersonal interaction [6]. In fact, facial expressions represent a direct and naturally preeminent means of communicating emotions [7, 8, and 9]. The most commonly used facial expression model was an outcome of extensive studies by Ekman and Friesen [10, 11]. According to this model, there are six "universal facial expressions" representing happiness, sadness, anger, fear, surprise and disgust.

### *1.1 Previous Work*

In general, the human brain can recognize facial expressions just by observing the shape of facial features [12, 13]. When dealing with the FEC problem, the feature extraction process can be appearance-based or geometry-based. For appearance-based processes, the fiducial points of the face are selected either manually or automatically [14]. Then, feature extraction filters, which are usually Gabor filters, are convolved with the facial images and vectors are formed by the responses of the filtering processes. Alternatively, these filters can be applied to the entire face image. For geometry-based processes, the positions of a set of fiducial points form a feature vector that represents the facial geometry [14].

Approaches for FEC can be divided into spatial and spatio-temporal. In spatial approaches, temporally disjoint images are processed in order to extract spatial facial features that are then used for the classification of facial expressions. Some state-of-the-art spatial FEC methods are [14, 15, 16, 17, 18, and 19]. In [14], two hybrid FEC systems are proposed that employ the 'one-against-all' classification strategy. The first system decomposes the facial images into linear combinations of several basis images using Independent Component Analysis (ICA). Subsequently, the corresponding coefficients of these combinations are fed into Support Vector Machines (SVMs) that carry out the classification process. In addition, the performance of baseline techniques that combine ICA with either two-class Cosine Similarity Classifiers (CSC) or two-class Maximum Correlation Classifiers (MCC) is investigated. The second system performs feature extraction via a set of Gabor Wavelets (GWs). The resulting features are then classified using CSC, MCC, or SVMs that employ various kernel functions. In [15], the FEC problem is undertaken using Kernel Canonical Correlation Analysis (KCCA). Initially, for each facial

image, 34 landmark points are manually located and then converted into a Labeled Graph (LG) vector using the Gabor Wavelet Transformation (GWT) method to represent the facial features. Also, for each training facial image, the semantic ratings describing the basic expressions are combined into a six-dimensional semantic expression vector, by utilizing class-label information. KCCA is employed to implement the learning of the correlation between the LG vector and the semantic expression vector. This correlation contributes to the estimation of the associated semantic expression vector of a given test image that is used to perform FEC. The method in [16] uses Supervised Locally Linear Embedding (SLLE) to perform feature extraction. Then, a minimum-distance classifier is used to classify the various expressions. SLLE computes low dimensional, neighborhood-preserving embeddings of high dimensional data and is used to reduce data dimension and extract features. The basic idea of LLE is the global minimization of the reconstruction error of the set of all local neighbors in the data set. This technique maps its inputs onto a single global coordinate system of lower dimension, attempting to discover nonlinear structure in high dimensional data by exploiting the local symmetries of linear reconstructions. It expects the construction of a local embedding from a fixed number of nearest neighbors to be more appropriate than from a fixed subspace. The Supervised-LLE algorithm uses class label information when computing neighbors to improve the performance of classification. The work in [17] introduces the ICA-FX feature extraction method that is based on ICA and is supervised in the sense that it utilizes class information for multi-class classification problems. Class labels are incorporated into the structure of standard ICA by being treated as input features, in order to produce a set of class-label-relevant and a set of class-label-irrelevant features. The learning rule being used applies the stochastic gradient method to maximize the likelihood of the observed data. Then, only class-label-relevant features are retained, thus reducing the feature space dimension in line with the principle of parsimony. This improves the generalization ability of the nearest-neighbor classifier that is used to perform FEC. In [18], the Locality-Preserved Maximum Information Projection (LPMIP) technique produces linear projections used to identify the underlying manifold structure of a data set. LPMIP takes within-locality and between-locality into account simultaneously in the modeling of the manifold. LPMIP seeks to find a balance between the global and local structures, so as to find a subspace that detects the intrinsic manifold structure for classification tasks. To perform FEC, facial expression features are extracted using Gabor wavelet filters at 34 manually marked fiducial points on each face. Then, once the facial expression projection subspace is learned via LPMIP, a nearest-neighbor classifier is used to produce the FEC decision. In [19], a FEC system is developed that classifies frontal images to one of 7 basic facial expressions. Initially,

features are extracted using a bank of Gabor filters of 8 orientations and 5 scales. Then, the AdaBoost method of [20] is applied for feature selection. AdaBoost sequentially selects the feature that gives the most information about classification, given the features that have been already selected. The selected features are fed to 7 SVM classifiers that are trained to discriminate each expression from everything else. The expression classification decision is made by choosing the classifier with the maximum margin for the test example.

Unlike spatial approaches, spatio-temporal methods allow the modeling of the dynamics of facial expressions by taking into consideration facial features that are extracted from multiple frames of a facial expression video sequence. Some recent state-of-the-art spatio-temporal FEC methods are [21, 22, 23, 24, and 25]. Spatial approaches were shown to achieve good FEC performance, even though, in some cases, temporal changes can provide critical information about expressions that can lead to improved recognition performance. Specifically, evidence suggests that facial dynamics enhance recognition accuracy under suboptimal viewing conditions [26]. This is because motion information can serve to normalize variations due to lighting, skin color, and other static facial variations [27]. Nevertheless, spatio-temporal algorithms can only be employed in a limited number of applications, since they usually demand high computational load and also because spatial data are more readily available than temporal data. As a result, we focus our effort on developing a spatial-only FEC method that can provide good classification results.

### *1.2 The Salient Feature and Reliable Classifier Selection (SFRCS) Methodology*

The proposed SFRCS spatial FEC methodology attempts to classify any random facial image to one of the following $C = 7$ basic facial expression classes: happiness (E1), sadness (E2), anger (E3), fear (E4), surprise (E5), disgust (E6), and neutral state (E7). To do so, the best subset of extracted features is identified and these selected features are concatenated to produce the Salient Feature Vectors (SFVs). This is done individually for all $\frac{C(C-1)}{2}$ pair-wise comparisons between the $C$ facial expression classes. For each pair of expression classes, the (two) SFVs are produced by first convolving the facial images with a set of 2-D Gabor filters, which are associated with different scales and orientations, to extract the features and then utilizing a class separability measure in an iterative process to select a subset of salient features. This measure is based on Fisher's criterion [28] and quantifies the separation, produced in a Linear Discriminant Analysis (LDA) subspace, between two facial expression classes, when these classes are represented by a specific feature set. As a result, we can identify the combination of features that produces the largest class separation for a specific pair of expression classes, i.e.

the salient feature set. By utilizing this criterion, the feature selection process can reject features that produce the need to have strongly non-linear separating surfaces between the classes, which drastically weaken the classification performance of a LDA-based algorithm. Moreover, it prevents selecting features that are similar to one another, i.e. that are, or are close to being, linearly dependent. Then, two-class LDA is applied to the SFVs in order to produce the Salient Discriminant Hyper-planes (SDHs), which are essentially projections onto which large class separations are attained. The SDHs are used to train the $\frac{C(C-1)}{2}$ two-class classifiers, and the corresponding two-class separations are measured. This completes the training phase of our algorithm.

Next, during the test phase where the FEC decision is produced, the class separability measurements are utilized to develop a novel classification scheme that is tailored to the feature selection process. A reference Interim Classification Scheme (ICS) is presented, which integrates the two-class classification results and has certain useful properties that can lead to a fast and accurate decision. The first property is that the classification decision is produced by utilizing results from only *C*-1 two-class classifiers and, in this set of classifiers, all *C* classes – one of which is the true class – are considered as possible outcomes. Secondly, if a class is rejected by any of the *C*-1 classifiers, none of the remaining classifiers considers the rejected class as a possible outcome. The ICS is used to illustrate the steps of the proposed Reliable Classifier Selection (RCS) classification scheme, which can be initialized on the same classifier topology and to show that both share the same useful properties. The RCS consists of $C-1$ classification steps, where at each step the most reliable classifier is identified, which is associated with the largest class separation and, as such, is expected to solve the easiest two-class problem. The classification decision that is produced by this classifier is used to designate the facial expression class that matches the least to the expression of the test face. This facial expression class, which represents the current top mismatch to the true class, is no longer a candidate to being the true class. By association, all two-class problems, or classifiers, that consider this class as a possible outcome are removed from the remaining classification process. The RCS classification scheme produces computationally efficient and accurate classification decisions, since it reduces the classification process to solving only $C-1$ two-class classification problems, the ones associated with the most reliable classifiers.

### *1.3 Paper Outline*

The outline of this paper is as follows: Section 2 illustrates how the multi-class and the one-against-all LDA approaches compare against the two-class LDA approach. Here, we justify why the proposed SFRCS method

chooses to incorporate two-class LDA in order to tackle the FEC problem. Section 3 presents the feature extraction process, as well as the iterative process that is used to select the set of features that comprise each SFV and create the SDHs. Section 4 introduces the reference ICS and the proposed RCS classification scheme that is developed in order to produce a more accurate FEC decision by identifying the most reliable classifiers. In section 5, the proposed methodology is tested on the JAFFE and MMI facial expression databases under common evaluation strategies in order to assess its FEC performance on standard data sets and compare it against competing state-of-the-art algorithms. Lastly, conclusions are drawn in section 6.

## 2. TWO-CLASS LDA VERSUS MULTI-CLASS AND ONE-AGAINST-ALL LDA

The purpose of this section is to justify our choice to use two-class LDA in order to tackle the FEC problem. In general, two-class LDA, or Fisher's Linear Discriminant (FLD) [28], accommodates our feature selection process well, since it allows forming a distinct set of features for each two-class classification problem. This is important and is expected to benefit the FEC process, since certain features are expected to be more useful when producing a separation between a specific pair of facial expression classes, than between a different pair. On the contrary, if multi-class LDA [28] was to be employed, which is the most common solution for recognition or classification problems where more than two classes are present, the same set of features would always be used to concurrently produce a separation between all classes. A third choice is one-against-all LDA, where a separation is produced between one class and all remaining classes [29]. Contrary to the classifier in the multi-class or one-against-all LDA case, the classifier in the two-class case is trained using a much smaller number of examples. This reduces the probability of including examples that may generate the need to produce a non-linear separation in order to discriminate between the classes. Therefore, the classifier in the two-class LDA case is expected to be more capable of producing a less complex decision boundary, in terms of linearity. This is because for two-class LDA, one class, e.g. $\mathcal{Y}_A$, is to be separated from a second class, e.g. $\mathcal{Y}_B$, whereas for one-against-all LDA, class $\mathcal{Y}_A$ is to be separated from all remaining classes, including class $\mathcal{Y}_B$, which should indeed be a more difficult problem. Even worse, for multi-class LDA, class $\mathcal{Y}_A$ is to be separated from all remaining classes and, concurrently, each of the remaining classes is to be separated from all other classes as well. Intuitively, we expect the feature selection and classification process for the multi-class or one-against-all LDA to be more susceptible to generalization problems. Next, an analysis is presented in order to verify these expectations and show the benefits of using two-class LDA. In subsection 2.1, multi-class LDA is compared against two-class LDA. In order to do

so, Fisher's criterion for the multi-class case is reformulated as the pair-wise expression that is shown in (9). We show that, under certain assumptions, (9) is equivalent to (1). Then cases are presented for which two-class LDA is expected to outperform multi-class LDA. In subsection 2.2, one-against-all LDA is compared against two-class LDA. Once again, certain assumptions are made and the alternative representation of Fisher's criterion for the one-against-all case is presented in (11). Then cases are presented for which two-class LDA is expected to outperform one-against-all LDA. In subsection 2.3, the three LDA-based classification approaches are evaluated in terms of computational cost and time complexity.

### *2.1 Multi-class vs. two-class LDA*

Given a set of $N$ feature vectors, realized by $\mathbf{x}_i \in \Re^d$, $i = 1,\ldots,N$, in the *d*-dimensional space, LDA is used to linearly reduce the dimension of these feature vectors by projecting them to the subspace defined by a discriminant matrix $\mathbf{W}$ that maximizes the ratio of the between-class scatter against the within-class scatter [28]:

$$J(\mathbf{W}) = \frac{trace\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_B\mathbf{W}\right)}{trace\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_W\mathbf{W}\right)}, \tag{1}$$

where $\mathbf{S}_B$ and $\mathbf{S}_W$ are the between-class and within-class scatter matrices, respectively. Alternatively, one can maximize the ratio of the determinants of the between-class scatter against the within-class scatter [30]. Assuming that the $N$ feature vectors correspond to $C$ different classes, $\Upsilon_l$, $l = 1,\ldots,C$, then $\mathbf{S}_B$ and $\mathbf{S}_W$ can be defined as:

$$\mathbf{S}_B = \sum_{l=1}^{C} N_l (\boldsymbol{\mu}_l - \boldsymbol{\mu})(\boldsymbol{\mu}_l - \boldsymbol{\mu})^{\mathrm{T}}, \tag{2}$$

$$\mathbf{S}_W = \sum_{l=1}^{C} \sum_{\mathbf{x}_i \in \Upsilon_l} (\mathbf{x}_i - \boldsymbol{\mu}_l)(\mathbf{x}_i - \boldsymbol{\mu}_l)^{\mathrm{T}}, \tag{3}$$

where $\boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$ is the mean of all samples, $\boldsymbol{\mu}_l = \frac{1}{N_l}\sum_{\mathbf{x}_i \in \Upsilon_l}\mathbf{x}_i$ is the mean of the feature vectors in class $\Upsilon_l$, and $N_l$ is the number of feature vectors in class $\Upsilon_l$.

In order to investigate on the expected performance of multi-class LDA versus a set of two-class LDA problems that compare between all possible pairs of classes, we will reformulate $\mathbf{S}_B$ as a pair-wise expression. Let us firstly assume that the classes are homoscedastic [30]. That is, they follow the same probability distribution and thus they have the same within-class scatter matrix $\mathbf{S}_W$ given in (3). It is straightforward to show that the whitening transform [30] can be used to transform all the classes to have within-class scatter matrix equal

to the identity matrix. The whitening process transforms the samples $\mathbf{x}_i \in \Re^d$, $i = 1,\ldots,N$ to the samples $\mathbf{x}'_i \in \Re^d$, $i = 1,\ldots,N$ such that $\mathbf{x}'_i = \left(\mathbf{F}\mathbf{L}^{-1/2}\right)^{\mathrm{T}} \mathbf{x}_i$, where $\mathbf{F}$ is a square matrix consisting of the eigenvectors of $\mathbf{S}_W$ and $\mathbf{L}$ is a diagonal matrix containing the corresponding eigenvalues of $\mathbf{S}_W$. For notation simplicity we consider the samples $\mathbf{x}_i \in \Re^d$, $i = 1,\ldots,N$, to be pre-whitened and, thus, have as within scatter the identity matrix, i.e., $\mathbf{S}_W = \mathbf{I}$. Of course, the between scatter given is (2) is calculated over the pre-whitened samples. Let us also denote by $P_l$ the prior probability of class $\ell$. For simplicity, it is assumed that each class $\ell$ contains $N_l$ number of samples and its prior is estimated by $P_l = \frac{N_l}{N}$. Then, the between scatter matrix is given by:

$$\mathbf{S}_B = N\sum_{l=1}^{C} P_l(\boldsymbol{\mu}_l - \boldsymbol{\mu})(\boldsymbol{\mu}_l - \boldsymbol{\mu})^{\mathrm{T}} = N\sum_{l=1}^{C} P_l\left(\boldsymbol{\mu}_l\boldsymbol{\mu}_l{}^{\mathrm{T}} + \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} - 2\boldsymbol{\mu}\boldsymbol{\mu}_l{}^{\mathrm{T}}\right) \tag{4}$$

Replacing the global mean by the class means, $\boldsymbol{\mu} = \sum_{k=1}^{C} P_k\boldsymbol{\mu}_k$, and considering that $\sum_{l=1}^{C} P_l\boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} = \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}}$ we get:

$$\begin{aligned}
\mathbf{S}_B &= N\sum_{l=1}^{C} P_l\left(\boldsymbol{\mu}_l\boldsymbol{\mu}_l{}^{\mathrm{T}} - 2\sum_{k=1}^{C} P_k\boldsymbol{\mu}_k\boldsymbol{\mu}_l{}^{\mathrm{T}}\right) + N\sum_{k=1}^{C} P_k\boldsymbol{\mu}_k\left(\sum_{l=1}^{C} P_l\boldsymbol{\mu}_l\right)^{\mathrm{T}} \\
&= N\left(\sum_{l=1}^{C} P_l\boldsymbol{\mu}_l\boldsymbol{\mu}_l{}^{\mathrm{T}} - 2\sum_{k=1}^{C}\sum_{l=1}^{C} P_kP_l\boldsymbol{\mu}_k\boldsymbol{\mu}_l{}^{\mathrm{T}} + \sum_{k=1}^{C}\sum_{l=1}^{C} P_kP_l\boldsymbol{\mu}_k\boldsymbol{\mu}_l{}^{\mathrm{T}}\right) \\
&= \frac{N}{2}\left(2\sum_{l=1}^{C} P_l\boldsymbol{\mu}_l\boldsymbol{\mu}_l{}^{\mathrm{T}} - 2\sum_{k=1}^{C}\sum_{l=1}^{C} P_kP_l\boldsymbol{\mu}_k\boldsymbol{\mu}_l{}^{\mathrm{T}}\right)
\end{aligned} \tag{5}$$

Finally, by algebraic manipulations and interchanging of indices in (5) it is straightforward to show that:

$$\begin{aligned}
\mathbf{S}_B &= \frac{N}{2}\left(\sum_{l=1}^{C} P_l\boldsymbol{\mu}_l\boldsymbol{\mu}_l{}^{\mathrm{T}} - 2\sum_{k=1}^{C}\sum_{l=1}^{C} P_kP_l\boldsymbol{\mu}_k\boldsymbol{\mu}_l{}^{\mathrm{T}} + \sum_{k=1}^{C} P_k\boldsymbol{\mu}_k\boldsymbol{\mu}_k{}^{\mathrm{T}}\right) \\
&= \frac{N}{2}\sum_{l=1}^{C}\sum_{k=1}^{C} P_lP_k(\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)(\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)^{\mathrm{T}}
\end{aligned} \tag{6}$$

It can be seen in (6) that the between-class scatter matrix can be represented by an accumulation of pair-wise scatters of class-mean differences. Each pair-wise scatter, or the term inside the double summation, is the between-class scatter matrix of classes $\ell$ and $k$, as it would be modeled in a two-class classification problem using FLD. This is shown next:

$$\begin{aligned}
\mathbf{S}_B &= N\sum_{l=1}^{2} P_l(\boldsymbol{\mu}_l - \boldsymbol{\mu})(\boldsymbol{\mu}_l - \boldsymbol{\mu})^{\mathrm{T}} = \frac{N}{2}\sum_{l=1}^{2}\sum_{k=1}^{2} P_lP_k(\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)(\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)^{\mathrm{T}} \\
&= NP_1P_2(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\mathrm{T}}.
\end{aligned} \tag{7}$$

The double summation in (6) results in calculating the between-class scatter matrix of a pair of classes twice. Moreover, the double summation also involves calculating the between-class scatter matrix for the case where $l = k$, where the outcome would be zero. Therefore, we can remove the terms in (6) that are redundant when calculating $\mathbf{S}_B$ and re-state it as:

$$\mathbf{S}_B = N\sum_{l=1}^{C-1}\sum_{k=l+1}^{C} P_l P_k (\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)(\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)^{\mathrm{T}}. \tag{8}$$

Clearly, it is seen that $\mathbf{S}_B$ is calculated by accumulating results from $\frac{C(C-1)}{2}$ pair-wise scatters of class-mean differences, or, equivalently, by accumulating the between-class scatter matrices that result from $\frac{C(C-1)}{2}$ two-class classification problems.

Next, the $\mathbf{S}_B$ expression of (8) is used to illustrate the cases for which $C$-class, or multi-class, LDA produces a large classification error due to the accumulation operation, unlike two-class LDA which circumvents this problem. Since the samples are pre-whitened and $\mathbf{S}_W = \mathbf{I}$, the denominator in (1) becomes $trace(\mathbf{W}^{\mathrm{T}}\mathbf{S}_W\mathbf{W}) = trace(\mathbf{W}^{\mathrm{T}}\mathbf{W}) = trace(\mathbf{I}) = d$, where we have used the fact that the projection vectors in $\mathbf{W}$ are orthonormal [30]. That is, the denominator of (1) does not contribute to the optimization problem after whitening and thus we should only focus our analysis on the numerator. Let us substitute (8) into (1). Then, the Fisher criterion essentially reduces to maximizing:

$$\begin{aligned} J(\mathbf{W}) &= \frac{1}{d} trace(\mathbf{W}^{\mathrm{T}}\mathbf{S}_B\mathbf{W}) = \frac{N}{d} trace(\mathbf{W}^{\mathrm{T}}\sum_{l=1}^{C-1}\sum_{k=l+1}^{C} P_l P_k (\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)(\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)^{\mathrm{T}}\mathbf{W}) \\ &= \frac{N}{d} trace(\sum_{l=1}^{C-1}\sum_{k=l+1}^{C} P_l P_k \mathbf{W}^{\mathrm{T}}(\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)(\boldsymbol{\mu}_l - \boldsymbol{\mu}_k)^{\mathrm{T}}\mathbf{W}) \end{aligned} \tag{9}$$

It can be seen that the Fisher criterion seeks to find a projection that maximizes the weighted mean squared distance between all pairs of classes $l$ and $i$. Moreover, if we consider classes with equal priors, the distance between each pair of classes is set to be the difference between their means. Therefore, LDA is based on a single prototype per class, the class mean, which is often insufficient for the multi-class case. It is well-known that for homosedastic classes the multi-class LDA is optimal in terms of Bayes error when we can project the samples keeping $C-1$ dimensions [31]. However, when we project to one dimension, even for homoscedastic classes, multi-class LDA is not optimal in terms of Bayes error [30]. In this case, two-class LDA is optimal and thus one-against-one classification is optimal for each two-class problem defined. The number of two-class classifiers that

need to be defined in this case is $\frac{C(C-1)}{2}$. To illustrate more clearly that multi-class LDA is not optimal when projecting to one dimension we discuss the form of equation (9). Specifically, (9) indicates that the accumulation (double summation) operation of multi-class LDA is likely to produce classification errors for cases where the training set contains at least one class, whose mean varies significantly from the means of the remaining classes. These errors are expected to exacerbate when the total number of classes is relatively small, as is the case with FEC, where the number of classes is $C = 7$, since (9) seeks to produce the maximum mean difference between the means of the classes. As a result, when producing a multi-class separation, (9) will heavily take into account the presence of the class that is well-separated from the remaining classes. In this case, multi-class LDA will produce a projection onto which large overlaps between the remaining classes are produced, rendering the classification process problematic.

Fig. 1 illustrates the aforementioned problem by presenting classes A, B, and C in the two-dimensional space. The difference between the means of the classes A,C and B,C is much larger than the difference between the means of the classes A,B. We want to observe the effect of this class topology on the numerator of Fisher's criterion, thus we again assume that the within-class scatter matrix is the identity matrix after whitening. Fig. 1a shows the hyper-plane that corresponds to the discriminant weight vector $\mathbf{w}_{A,B,C}$ that is produced by multi-class LDA. When classes A and B are projected onto this hyper-plane, there is significant overlap and, as a result, classification errors are produced when attempting to classify between these two classes. On the contrary, Fig. 1b shows the 3 hyper-planes that correspond to the discriminant weight vectors $\mathbf{w}_{A,B}$, $\mathbf{w}_{A,C}$, and $\mathbf{w}_{B,C}$. These weight vectors can be produced by 3 two-class LDA processes where the pairs of classes under consideration are A-B, A-C, and B-C, respectively. It is clear that each two-class solution provides a clean separation between the two classes. Assuming that a proper classification scheme can be developed to integrate the classification results from these 3 pair-wise comparisons, the two-class LDA will produce a better solution than multi-class LDA.

A second case where multi-class LDA is not expected to produce good classification results is if a non-linear separation is required in order to separate all the classes from one another. Fig. 2 illustrates such a case, with Fig. 2a showing the discriminant solution – the projection coordinates illustrated as dotted line segments – that a multi-class classifier should utilize for a valid separation between all 5 classes. Clearly, this discriminant solution is non-linear, thus, multi-class LDA cannot handle this particular classification problem. On the other hand, Fig. 2b shows the discriminant solution that can be produced from a pair-wise classification process. Each hyper-

plane – a dotted line – can be produced by applying LDA on each pair of classes, for all pair-wise class combinations. Once again, it is shown that each two-class discriminant solution produces a clean separation between the two classes. As a result, assuming that a proper classification scheme can be developed to integrate the pair-wise classification results, two-class LDA will produce a better solution than multi-class LDA. That is, two-class LDA, in one-against-one classifiers, defines nonlinear separating surfaces (piecewise linear) whereas multiclass LDA defines linear separating hyperplanes.

*2.2 One-against-all vs. two-class LDA*

The one-against-all method converts a $C$-class problem into $C$ classification problems [29]. Each classification problem must discriminate between one of the $C$ classes, e.g., $\Upsilon_i$, and the union of the remaining $C-1$ classes, i.e. $\overline{\Upsilon_i} = \bigcup_{j \neq i} \Upsilon_j$. Accordingly, the between class scatter-matrix of any $i-$th one-against-all classification problem, out of the total $C$, can be defined as:

$$\mathbf{S}_B^i = P_i(1-P_i)\left(\boldsymbol{\mu}_i - \overline{\boldsymbol{\mu}_i}\right)\left(\boldsymbol{\mu}_i - \overline{\boldsymbol{\mu}_i}\right)^{\mathrm{T}}, \tag{10}$$

where $\boldsymbol{\mu}_i = \frac{1}{N_i}\sum_{\mathbf{x}_j \in \Upsilon_i} \mathbf{x}_j$, $\overline{\boldsymbol{\mu}_i} = \frac{1}{N - N_i}\sum_{\mathbf{x}_j \notin \Upsilon_i} \mathbf{x}_j$, and $N_i$ is the number of feature vectors in class $\Upsilon_i$. Assuming that $\mathbf{S}_W$ is an identity matrix, the Fisher criterion reduces to maximizing:

$$J(\mathbf{W}) = \frac{1}{d} trace(\mathbf{W}^{\mathrm{T}}\mathbf{S}_B^i\mathbf{W}) = \frac{1}{d} trace(P_i(1-P_i)\mathbf{W}^{\mathrm{T}}\left(\boldsymbol{\mu}_i - \overline{\boldsymbol{\mu}_i}\right)\left(\boldsymbol{\mu}_i - \overline{\boldsymbol{\mu}_i}\right)^{\mathrm{T}}\mathbf{W}). \tag{11}$$

The fact that the maximum difference between the mean of one class from the mean of the union of the remaining classes is sought, can make the process susceptible to classification errors. Specifically, the same problems that were previously shown to affect multi-class LDA can also affect one-against-all LDA. Fig. 1c shows an example of the classification errors that can be produced when the mean of a class, in this case class C, is located at a relatively large distance from the means of the remaining classes. When the task is to classify between class A and the group of classes B and C, or between class B and the group of classes A and C, the projections to the corresponding hyper-planes $\mathbf{w}_{\mathrm{A,BC}}$ and $\mathbf{w}_{\mathrm{B,AC}}$ produce a significant overlap between classes A and B. Thus, it becomes clear that using the union of the remaining classes, which in this case is simply their mean, to form one new class can be problematic for classification purposes.

As with multi-class LDA, Fig. 2c shows that one-against all LDA can also face insurmountable problems that are related to non-linearity, though, in this case, the problem doesn't seem to exacerbate as much. It is shown that

a non-linear separation is required in order to separate any one class, in this case class C, from the remaining classes. Fig. 2c shows the discriminant solution that a one-against-all classifier should utilize to do so. Since at least 2 hyper-planes are required, it is obvious that this discriminant solution is non-linear, therefore, one-against-all LDA may have difficulty to solve this particular classification problem. Moreover, there are cases when the classification problem becomes more difficult, e.g., when the task is to produce a separation between class A and the remaining classes. In that case, the mean of the remaining classes roughly maps onto the space of A. As mentioned before, Figs. 1b and 2b show that two-class LDA produces a valid separation between the classes.

*2.3 Computational and time complexity*

At this point, the computational complexity of the three LDA-based classification approaches is considered. Once again, it is assumed that each class $\ell$ contains the same number of $d-$dimensional feature vectors (samples), $N_l = L = \frac{N}{C}$. For an LDA process, the computational complexity associated with evaluating the covariance matrixes is $N \cdot d^2$ and the computational complexity associated with the eigenvector decomposition process is $N \cdot d \cdot \min(N,d)$, thus, the total complexity is $N \cdot d \cdot \left(d + \min(N,d)\right)$. The computational complexity for multi-class LDA is $N \cdot d \cdot \left(d + \min(N,d)\right)$, since one LDA process is implemented using all $N$ data. The computational complexity for one-against-all LDA is $C \cdot N \cdot d \cdot \left(d + \min(N,d)\right)$, since $C$ LDA processes are implemented, each using all $N$ data and each with complexity $N \cdot d \cdot \left(d + \min(N,d)\right)$. For two-class LDA, the computational complexity is $(C-1) \cdot N \cdot d \cdot \left(d + \min\left(2\frac{N}{C}, d\right)\right)$, since $\frac{C(C-1)}{2}$ LDA processes are implemented, each using $2\frac{N}{C}$ data and each with complexity $2\frac{N}{C} \cdot d \cdot \left(d + \min\left(2\frac{N}{C}, d\right)\right)$. From the above, two conclusions are drawn: a) two-class LDA as well as multi-class LDA are always more efficient than one-against-all LDA, and b) as $C$ increases, multi-class LDA becomes more efficient than two-class LDA. The two approaches are equally efficient only for the $C = 2$ case.

With the development of the distributed computing field, parallel processing is often employed to train large-scale algorithms in order to reduce training time. Using parallel processing, the training time complexity for one-against-all LDA improves by a factor of $C$ and becomes $N \cdot d \cdot \left(d + \min(N,d)\right)$. Two-class LDA improves by a

factor of $\frac{C(C-1)}{2}$ and becomes $\frac{2}{C} \cdot N \cdot d \cdot \left( d + \min\left( 2\frac{N}{C}, d \right) \right)$. This is because the multiple LDA processes can be computed in parallel fashion. Since one LDA process is implemented for multi-class LDA, there is no reduction in time complexity (under this basic optimization consideration of parallel processing). From the above, two conclusions are drawn regarding the time complexity during the training phase, under the parallel processing assumption: a) one-against-all LDA becomes equally efficient as multi-class LDA, and b) as $C$ increases, two-class LDA becomes more efficient than either multi-class LDA, or one-against-all LDA. All three approaches are equally efficient only for the $C = 2$ case. In general, when parallel computing is employed, the training cost for two-class LDA is the smallest due to the fact that the number of samples trained at a time is also the smallest.

In this section, it was shown that, under certain conditions, two-class LDA is expected to perform better than either multi-class or one-against-all LDA. Moreover, in [32], it was shown that, over a large number of randomly generated problems, the two-class solution produces the best results. It is noted, however, that our analysis used the assumption that $\mathbf{S}_W$ is an identity matrix that can be produced using whitening for homoscedastic classes, so it cannot be claimed that two-class LDA will always produce the best solution. Usually, the more $\mathbf{S}_W$ resembles an identity matrix, the more restricted the data distribution becomes, so the more statistically significant the class mean becomes, in terms of providing the ability to model the class data accurately. For example, when $\mathbf{S}_W$ is an identity matrix, a class is represented as a circle in the 2-D space, with the data being centered around the class-mean. In general, the more statistically significant the means of the classes are, the more the conclusions of this section are expected to hold. So, another conclusion is that the relative merits of a classification approach depend on the problem at hand [32].

Two-class LDA can effectively handle cases where one or more classes have highly dissimilar characteristics, e.g. related to the class mean, to the remaining classes. Moreover, it can overcome the non-linearity problems for the cases that were presented. Naturally, cases may exist where two specific classes cannot be linearly separated, resulting to classification errors by all methods, including two-class LDA. Nevertheless, to the extent that the results of this section can serve as a guide, the performance of two-class LDA is expected to be better than that of its two counterparts. Stemming from the observations of the two aforementioned cases, we choose to implement $\frac{C(C-1)}{2}$ two-class classification processes, where each discriminant hyper-plane attempts to separate between

two facial expression classes. In section 4, we develop a classification scheme that can efficiently and effectively integrate the results from all two-class classifiers in order to produce the final classification decision.

## 3. CREATING THE SALIENT FEATURE VECTORS AND THE SALIENT DISCRIMINANT HYPER-PLANES

In this section, the 2-D Gabor-based feature extraction process is presented. Moreover, the feature selection technique that utilizes a Fisher-based class separability measure to construct the SFVs is introduced. Lastly, the process that produces the SDHs by applying discriminant analysis on the SFVs is presented.

### *3.1 Gabor-based Feature Extraction*

Initially, a feature set that contains $M$ features is extracted from each training facial image. These $M$ features correspond to the image being convolved with $M$ two-dimensional Gabor filters of different scales and orientations. The 2-D Gabor filter is produced by modulating a complex exponential by a Gaussian envelope, and can allow the direction of oscillation to any angle in the 2-D cartesian plane. Thus, a filter is produced with local support that is used to determine the image's oscillatory component in a particular direction at a particular frequency. This is particularly useful for FEC since different facial expressions (e.g. happiness vs. disgust, or neutral) produce these components at different directions and/or frequencies. A complex-valued 2-D Gabor function can be defined as [33]:

$$\Psi(k,x)=\frac{k^2}{\sigma^2}\exp\left(-\frac{k^2x^2}{2\sigma^2}\right)\left[\exp(jkx)-\exp(-\frac{\sigma^2}{2})\right] \tag{12}$$

To produce $M=M_s.M_o$ different Gabor functions, let us assume that $M_s$ different scales and $M_o$ different orientations are investigated. The different scales can be obtained by setting $k_i=\pi/2^i$, where $i=1,\ldots,M_s$. The different angular orientations can be obtained by selecting $M_o$ angles between 0 and 180 degrees, i.e. starting from $0°$ with a step-angle of $\frac{180°}{M_o}$.

### *3.2 Class Separability Measure for Feature Selection*

Next, a combination of the $N$ most useful features, out of the $M$ total, is selected when the task at hand is to discriminate between a specific pair of facial expression classes. Based on the consideration that different facial expressions produce more oscillatory components at different directions and frequencies, it is expected that, for a given pair of classes, the features extracted by Gabor functions that are related to the different scales and orientations that are more associated with each of these two classes, but not with both, can be used to produce a

larger class separation, than the rest of the features can. In total, there exist $\frac{C(C-1)}{2}=21$ distinct pair-wise class combinations for the $C=7$ facial expression classes: $\langle E1,E2\rangle$, $\langle E1,E3\rangle$, $\langle E1,E4\rangle$, $\langle E1,E5\rangle$, $\langle E1,E6\rangle$, $\langle E1,E7\rangle$, $\langle E2,E3\rangle$, $\langle E2,E4\rangle$, $\langle E2,E5\rangle$, $\langle E2,E6\rangle$, $\langle E2,E7\rangle$, $\langle E3,E4\rangle$, $\langle E3,E5\rangle$, $\langle E3,E6\rangle$, $\langle E3,E7\rangle$, $\langle E4,E5\rangle$, $\langle E4,E6\rangle$, $\langle E4,E7\rangle$, $\langle E5,E6\rangle$, $\langle E5,E7\rangle$, and $\langle E6,E7\rangle$. Thus, the feature selection process that is presented next creates 21 sets of salient features. At this point, the $M$ 2-D features that were extracted from each facial image are converted into $M$ 1-D vectors via row-concatenation. Accordingly, $M$ feature vectors, $\mathbf{f}^i$, where $i=1,\ldots,M$, are produced.

Let us assume that we need to classify between a specific pair $E_x$ and $E_y$. In order to select the subset of $N$ most useful feature vectors, where $N<M$, a class separability measure that is based on the maximum value of Fisher's criterion is employed. It is noted that, for our purposes, this is a suitable measure since the discriminant hyper-planes that we later produce to train each two-class classifier stem from Fisher's criterion. When examining the $i-\text{th}$ feature vector, this separability measure, which is associated with both the between-class and the within-class distance, is defined as:

$$J_{E_{x,y}}^{\max}(i)=J(\mathbf{w}_{0,E_{x,y},i})=\frac{\left(\mu_{0,E_x,i}-\mu_{0,E_y,i}\right)^2}{\sigma_{0,E_x,i}^2+\sigma_{0,E_y,i}^2}, \tag{13}$$

where $\mu_{0,E_x,i}$ and $\mu_{0,E_y,i}$ denote the sample mean and $\sigma_{0,E_x,i}^2$ and $\sigma_{0,E_y,i}^2$ the sample variance of the training feature vectors of classes $E_x$ and $E_y$, respectively, when projected to the subspace defined by $\mathbf{w}_{0,E_{x,y},i}$. The discriminant vector $\mathbf{w}_{0,E_{x,y},i}$ is given by

$$\mathbf{w}_{0,E_{x,y},i}=\mathbf{S}_{W,E_{x,y},i}^{-1}\left(\mathbf{m}_{E_x}^i-\mathbf{m}_{E_y}^i\right), \tag{14}$$

where $\mathbf{m}_{E_x}^i$ and $\mathbf{m}_{E_y}^i$ denote the sample mean of the feature vectors of classes $E_x$ and $E_y$, respectively, for the $i-\text{th}$ feature. Moreover, $\mathbf{S}_{W,E_{x,y},i}$ is the within-class scatter matrix for the $i-\text{th}$ feature, and is defined as

$$\mathbf{S}_{W,E_{x,y},i}=\sum_{j:E_j=E_x}\left(\mathbf{f}_j^i-\mathbf{m}_{E_x}^i\right)\left(\mathbf{f}_j^i-\mathbf{m}_{E_x}^i\right)^{\mathrm{T}}+\sum_{j:E_j=E_y}\left(\mathbf{f}_j^i-\mathbf{m}_{E_y}^i\right)\left(\mathbf{f}_j^i-\mathbf{m}_{E_y}^i\right)^{\mathrm{T}}, \tag{15}$$

where $j$ indicates the class (either $E_x$ or $E_y$) to which the $i-\text{th}$ feature vector, $\mathbf{f}_j^i$, belongs to. So, each summation term adds up all the $i-\text{th}$ feature vectors that belong to a specific class.

Using (13), we now have a class separability measure that indicates how useful each of the $M$ features is. One option on how to select the $N$ best features, out of the total $M$, could be by identifying the ones that produce the $N$ largest values for this separability measure:

$$J_{E_{x,y}}^{max_N} = \max_{N}\left(J_{E_{x,y}}^{max_1}(1,\ldots,M)\right) = J_{E_{x,y}}^{max_1}(\mathbf{i}),\ \ \mathbf{i} \underset{N}{\subset} 1,\ldots,M\ . \tag{16}$$

However, we also need to consider that the $N$ (per class) best feature vectors, which were selected in order to discriminate between classes $E_x$ and $E_y$, will subsequently be processed by a two-class LDA process in order to produce the discriminant hyper-plane. To do so, the $N$ selected feature vectors are concatenated to produce one large column vector, the SFV, as such:

$$\mathbf{f}_j^{SFV:E_{x,y}} = \left[\mathbf{f}_j^{\mathbf{i}(1)^{\mathrm{T}}},\ldots,\mathbf{f}_j^{\mathbf{i}(N)^{\mathrm{T}}}\right]^{\mathrm{T}},\ \text{where}\ \mathbf{i} \underset{N}{\subset} 1,\ldots,M,\ \text{and}\ \mathbf{f}_j \in \mathcal{E}_x,\ \text{or,}\ \mathbf{f}_j \in \mathcal{E}_y. \tag{17}$$

As a result, notions such as linear dependency between the feature vectors should be taken into account when selecting the $N$ best features. For example, if two feature vectors are linearly dependent, or close to being linearly dependent, then the selection of both these vectors, rather than only one of them, would not provide any additional benefit to the discriminant ability of the hyper-plane being produced to train the two-class classifier. For this reason, an iterative feature selection process that is again based on the class separability measure (13) is developed in order to define the group of feature vectors that should compose each pair of SFVs, for all two-class problems. Specifically, the separability measure is not applied independently to each feature but, rather, to groups of features, in order to identify the feature combination that produces the largest class separation.

### *3.3 Creating SFVs to produce SDHs*

A feature selection methodology is now applied in order to construct the $\frac{C(C-1)}{2}$ SDHs, where each hyper-plane is associated with two realizations (one per class) of $N$ selected features that compose the two SFVs. The process that is followed to construct the SFVs is based on the well-known 'forward-selection' process that has been extensively used in statistics [34]. Using forward selection, we add one of the $M$ Gabor features to the SFV one at a time. At each step, each Gabor feature that is not already in the SFV is tested for inclusion in the SFV. The most significant of the remaining Gabor features is then added to the SFV. The class separability measure of (13) is utilized to determine which one of the remaining Gabor features is the most significant. The process iterates until $N$ Gabor features are selected. Specifically, the first feature vector to be selected is the one that

produces the maximum $J_{E_{x,y}}^{\max}$ value, out of all the original $M$ feature vectors, when attempting to discriminate between the two facial expression classes $E_x$ and $E_y$. Subsequently, each feature vector to be selected next is identified by creating groups of features in vector form, i.e. $\mathbf{f}_{group}$, where each group contains the feature vectors that were previously selected and a new candidate feature vector. A candidate feature vector, for which (19) and (20) are used to determine if it is selected next or not, is simply a feature vector that has not yet been selected as being one of the N vectors that compose the SFV. In general, if this is the $i-\text{th}$ feature vector to be selected, then $M-i+1$ distinct groups of features are created:

$$\mathbf{f}_{group_i}^{j} = \left[\mathbf{f}_{selected}^{1,\ldots,i-1\,\mathrm{T}}, \mathbf{f}_{candidate}^{j\,\mathrm{T}}\right]^{\mathrm{T}}, \quad j = 1,\ldots,M-i+1. \tag{18}$$

These groups contain the same set of feature vectors that were previously selected and one of the $M-i+1$ candidate feature vectors. For example, to select the second feature vector, $M-1$ different feature groups are created. All groups, which, in this case, are simply pairs of feature vectors, contain the first previously selected feature vector as well as one of the $M-1$ candidate feature vectors.

Next, for each group of features in (18), the corresponding FLD hyper-plane that is used to discriminate between the facial expression classes $E_x$ and $E_y$ is produced:

$$\mathbf{w}_{0,E_{x,y},group_i^j} = \mathbf{S}_{W,E_{x,y},group_i^j}^{-1}\left(\mathbf{m}_{E_x,group_i^j} - \mathbf{m}_{E_y,group_i^j}\right), \tag{19}$$

where $group_i^j$ indicates that this expression only uses the group of features that are currently under consideration, which include the $i-1$ feature vectors that were previously selected and the $j$-th candidate feature vector that is being considered to be selected as the $i-\text{th}$ feature vector. Subsequently, the value of the corresponding separability measure for this group of features is calculated by:

$$J_{E_{x,y},group_i^j} = J(\mathbf{w}_{0,E_{x,y},group_i^j}) = \frac{\left(\mu_{0,E_x,group_i^j} - \mu_{0,E_y,group_i^j}\right)^2}{\sigma_{0,E_x,group_i^j}^2 + \sigma_{0,E_y,group_i^j}^2}. \tag{20}$$

The selected feature, out of all the candidate features, is set to be the one whose corresponding group produces the maximum value of this separability measure, i.e. $J_{E_{x,y},group_i}^{\max}$.

Each selected feature, which was extracted from images that correspond to classes $E_x$ and $E_y$, belongs to the group that produces a $J_{E_{x,y},group_i}^{\max}$ value, where $i = 1,\ldots,N,$ when used by the expressions in (19) and (20). In order to select all $N$ feature vectors, this process is iterated $N$ times. After each such iteration, we operate on a

consistently reduced set of candidate feature vectors, since the feature vectors that have already been selected are no longer candidates for the remaining feature selection process. Once this process is completed, the $N$ selected feature vectors are concatenated to form the SFV of each class, as (17) indicates. The two SFVs that correspond to classes $E_x$ and $E_y$, are also related to a specific SDH, via (19). By using this feature selection process, the two-class LDA algorithm can potentially evade problems relating to non-linear class separability. This is because multiple combinations of groups of features are examined and the group that produces the largest class separation $J^{\max}_{E_{x,y},group_i}$ is selected. Since the separability value is based on Fisher's criterion, it is expected that a combination of features that would form a non-linear separation between the classes would produce a small $J_{E_{x,y},group_i^j}$ value, thus, this combination of features would be rejected. For the same reason, each SFV that is produced should not contain features that are, or are close to being, linearly dependent. It is noted that more accurate results can be achieved if exhaustive search, rather than forward selection, is utilized in order to consider all possible subsets of features. Forward selection is a sub-optimal process, however, we chose to utilize it due to the fact that it requires a significantly smaller computational cost than exhaustive search does. It is also worth noting that our feature selection method is different than a method described in [35], where forward selection is applied and Wilk's $\Lambda$ is utilized as the class separability criterion. Wilk's $\Lambda$ is inversely related to the eigenvalue, $\lambda$, which indicates the separation that can be produced in a subspace defined by an eigenvector, $\mathbf{w}_0$. Our method measures the actual separation that is produced by Fisher's linear discriminant, i.e. after the data is projected to the aforementioned subspace, thus, more accurate results are attained. In fact, the accuracy of the method in [35] can match the accuracy of our method only if each class has a Gaussian density with a common covariance, which is a rare occurrence in real-life applications, so that $\mathbf{w}_0$ shall define the best transforming axes, and the feature space with the largest possible linear separability can be obtained [36]. Next, the novel RCS classification scheme that also utilizes the class separability measure of (13) is presented.

## 4. Classification via Reliable Classifier Selection

For the FEC problem, the facial expression of a test image $r$ is to be classified to one of the $C$ facial expression classes. During the training phase of the SFRCS algorithm, the original $C$-class classification problem is decomposed to $\frac{C(C-1)}{2}$ two-class classification problems, one for each pair of $E_{x,y}$ classes, where

$x = 1,\ldots,C-1$ and $y = x+1,\ldots,C$, as the summation operators in (9) indicate. For each $E_{x,y}$ pair of classes, the two SFVs are created and FLD is applied on them, as is shown in (14), thus producing a SDH denoted as $\mathbf{w}_{0,E_{x,y}}$, since it is independent of $r$. This discriminant process is repeated for all pair-wise combinations of facial expression classes and, overall, $\frac{C(C-1)}{2}$ SDHs are created. During the test phase of SFRCS, 2-D Gabor features are extracted from the test facial image $r$. Then, in order to classify $r$ to either $E_x$ or $E_y$, the corresponding SFV is formed – by choosing the same subset of features that was selected to train the corresponding classifier – and denoted as $\mathbf{f}_r^{SFV:E_{x,y}}$, as is defined in (17). For a two-class classifier to produce a decision on whether a test image $r$ should be assigned to class $E_x$ or $E_y$, the test image and the two class means are projected onto $\mathbf{w}_{0,E_{x,y}}$ as such:

$$\tilde{\mathbf{f}}_r^{SFV:E_{x,y}} = \mathbf{w}_{0,E_{x,y}}^{\mathrm{T}} \mathbf{f}_r^{SFV:E_{x,y}}, \tag{21}$$

$$\tilde{\mathbf{m}}_{E_x} = \mathbf{w}_{0,E_{x,y}}^{\mathrm{T}} \mathbf{m}_{E_x}, \tag{22}$$

$$\tilde{\mathbf{m}}_{E_y} = \mathbf{w}_{0,E_{x,y}}^{\mathrm{T}} \mathbf{m}_{E_y}. \tag{23}$$

Then, the following classification decision is made, where $\|\bullet\|$ represents the $L_2$ norm between the vectors:

$$\begin{aligned} &\textit{if} \quad \left\|\tilde{\mathbf{f}}_r^{SFV:E_{x,y}} - \tilde{\mathbf{m}}_{E_x}\right\| < \left\|\tilde{\mathbf{f}}_r^{SFV:E_{x,y}} - \tilde{\mathbf{m}}_{E_y}\right\| \quad \textit{then} \quad r \in E_x \\ &\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad \textit{else} \quad r \in E_y. \end{aligned} \tag{24}$$

Now that the training and test phases of the two-class classification process have been presented, the RCS classification scheme, which is tailored to the SFRCS FEC algorithm, is developed to integrate the classification results generated by all two-class problems and produce the final classification decision. RCS is presented next.

### *4.1 Interim Classification Scheme (ICS)*

Initially, the computationally efficient ICS is developed, which possesses certain useful properties, as is later explained. For our purposes, the ICS provides a good reference to appropriately integrate results produced by a set of two-class classifiers, in order to produce the final FEC decision. In fact, we later show that the ICS and the proposed RCS classification scheme share the same properties, enabling us to deduce useful conclusions via direct experimental comparisons. Essentially, the ICS can be converted to the RCS scheme by incorporating the reliable classifier selection process that is described in sub-section 4.2. The ICS topology is developed in accordance with the Acyclic Digraph (AD) principle, which defines directed graphs that contain no directed cycles [37, 38]. The ICS accommodates multiple classification routes and is illustrated in Fig. 3, where each

circle represents a two-class classification problem, or classifier, and the arrows indicate the direction that can be followed, i.e. the route that connects one classifier to the next. The final classification decision is indicated inside the square boxes. Fig. 3 illustrates that a classification path, which originates at the top-left of this topology and terminates at one of the $C$ square boxes, only uses $C-1$ routes and, equivalently, utilizes $C-1$ two-class classifiers sequentially in order to produce the final classification decision. For these $C-1$ classifiers, all $C$ facial expression classes are considered as possible outcomes. Therefore, before the final decision is made, it is always guaranteed that a classifier will consider the true expression class, i.e., the facial expression class that matches the expression of the test face, as one of the two possible classification outcomes, even if none of the previous classifiers on that path did so. Moreover, the ICS is optimized, in the sense that if a class is rejected at any point during the classification process, i.e. at any route, none of the remaining paths that can be followed considers the rejected class as a possible outcome. Next, we develop the improved RCS classification scheme, for which these same useful properties hold, where the classification decision is produced via a multi-step class rejection process. An example illustrates that, for each step, the classifier topology of Fig. 3 reduces in a way that the aforementioned useful properties are maintained.

### *4.2 Reliable Classifier Selection (RCS) Classification Scheme*

The RCS classification scheme that is proposed as part of the SFRCS methodology consists of $C-1$ steps and makes use of the class separability measure $J_{E_{x,y}}^{\max}$ of (13). Specifically, RCS uses the resulting $J_{E_{x,y}}^{\max}$ values that were calculated during the training phase and are associated with the SDHs that were produced. This measure is used to identify, at each step, the two-class classification problem for which the separation between the two classes is the largest, pointing to the most reliable classifier out of all existing ones. Initially, there are $\frac{C(C-1)}{2}=21$ classifiers. Then, the classification decision for this specific two-class problem is produced and the facial expression class that represents the worse match to the expression of the test face is identified. This is equivalent to saying that, based on the available training data, this class has the least probability of being the true facial expression class out of all the existing classes. This hypothesis renders all the two-class problems, for which this particular class is considered as a possible outcome, useless, thus, they are removed from the classification topology. This removal is illustrated later on by an example in Fig. 4.

The simultaneous removal of multiple two-class problems, at each classification step, results in producing a fast and computationally efficient classification solution, much-like ICS does. Moreover, a more accurate

classification decision can be produced, than the ones produced by typical schemes that solve all two-class problems, such as voting [33], or even ICS. This is because RCS discards all but $C-1$ two-class problems, the ones associated with the most reliable classifier during each of the $C-1$ classification steps. The (15 out of the total 21) two-class classification problems that are discarded is where a misclassification is most likely occur, since they are associated with smaller $J^{\max}_{E_{x,y}}$ values. Moreover, all $C$ facial expression classes are considered as possible outcomes in the $C-1$ two-class classification problems that are retained. Thus, for RCS, much-like ICS, the true class is always a candidate when determining the best match to the expression class of the test face. More importantly, when the two-class classification problem that contains the true class is selected, it is the one that produces the largest separation between the true class and a false class, out of the existing/retained false classes. This leads to a higher probability for the true class to be selected over the false class, since a proper discriminant space, which is also related to the proper SFVs, is utilized to produce the classification decision.

Next, a step-by-step description of the RCS classification scheme shows how the FEC decision is produced. As aforementioned, initially, there exist 21 two-class classification problems for which we have information regarding their associated class separation values $J^{\max}_{E_{\mathbf{x}(l),\mathbf{y}(l)}}$, where $l=1,\ldots,21$ and $E_{\mathbf{x}(l)}$ and $E_{\mathbf{y}(l)}$ represent the competing facial expression classes for all 21 two-class problems. The RCS classification scheme attempts to solve only 6 of these two-class classification problems. Initially, the class separation values are sorted, from the largest to the smallest, and the sorting index vector $\mathbf{j}_{sorting\_index}$ is generated as such:

$$\mathbf{j}_{sorting\_index} = \arg \underset{l}{sort}(J^{\max}_{E_{\mathbf{x}(l),\mathbf{y}(l)}}),\ \ l=1,\ldots,21, \tag{25}$$

where $\mathbf{x}=[1,1,1,1,1,1,2,2,2,2,2,3,3,3,3,4,4,4,5,5,6]$ and $\mathbf{y}=[2,3,4,5,6,7,3,4,5,6,7,4,5,6,7,5,6,7,6,7,7]$ indicate the 21 pair-wise class combinations. Then, RCS is realized by implementing Steps 1-6 that are presented next.

**Classification Step 1:** By making use of $\mathbf{j}_{sorting\_index}$, the two-class classifier that is associated with the largest class separation value is identified. The two classes that are associated with this classification problem are $E_{\mathbf{x}(\mathbf{j}_{sorting\_index}(1))}$ and $E_{\mathbf{y}(\mathbf{j}_{sorting\_index}(1))}$. We apply (24) to solve this problem and identify the class that represents the worst match to the expression of the test face. Then, we discard this class, as well as all two-class problems that consider this class as a possible outcome. Each class is involved in 6 of the 21 two-class classification problems, as the **x** and **y** vectors, as well as Fig. 3, indicate. Thus, 15 classification problems are now retained. Then, the

associated class separation values of the 6 two-class classification problems that have been discarded are also discarded from the $\mathbf{j}_{sorting\_index}$ vector, thus producing $\mathbf{j}'_{sorting\_index}$.

**Classification Step 2:** Next, the two-class classifier that is associated with the largest class separation value, out of the 15 remaining ones, is found using $\mathbf{j}'_{sorting\_index}$ (the desired index is the first element of this vector). Once again, (24) is applied and the class that produces the worse match is discarded, as well as the 5 two-class classification problems that consider this class as a possible outcome, leaving only 10. As before, the 5 associated entries in the $\mathbf{j}'_{sorting\_index}$ vector are also discarded and $\mathbf{j}''_{sorting\_index}$ is produced.

**Classification Step 3:** The same process is repeated using $\mathbf{j}''_{sorting\_index}$ and 4 additional two-class classification problems are discarded, leaving only 6. As before, the $\mathbf{j}'''_{sorting\_index}$ vector is produced.

**Classification Step 4:** The same process is repeated using $\mathbf{j}'''_{sorting\_index}$ and 3 additional two-class classification problems are discarded, leaving only 3. As before, the $\mathbf{j}''''_{sorting\_index}$ vector is produced.

**Classification Step 5:** This process is repeated one last time where, again, the largest class separation value is found using $\mathbf{j}''''_{sorting\_index}$ and 2 additional two-class classification problems are discarded, leaving only one last two-class classification problem.

**Classification Step 6:** The final classification decision is made by identifying the facial expression (in this final two-class classification problem) that produces the best match to the expression of the test face.

The graphical example in Fig. 4 is used to illustrate how the ICS two-class classifier topology of Fig. 3 reduces after each classification step of RCS to produce a computationally efficient solution. Let us assume that for Classification Steps 1 through 5, the following classes, as well as their associated two-class problems, get sequentially discarded: E2, E4, E6, E1, E7. Figs. 4-a through 4-e show the discarding process. Fig. 4-e also illustrates that a selection between classes E3 and E5 produces the final decision in Classification Step 6.

## 5. EXPERIMENTAL RESULTS

In this section, the performance of the proposed SFRCS method is evaluated and compared against contemporary state-of-the-art FEC methods. The JAFFE [33] facial expression database, which contains images captured at disjoint temporal instances, has been extensively used when evaluating the classification performance of spatial facial expression algorithms. Hence, our method, as well as the spatial FEC methods of [14, 15, 16, 17, and 18]

that it is compared against, is evaluated on the JAFFE database. The JAFFE database contains 213 images of 7 standard [11] facial expressions (happiness, sadness, anger, fear, surprise, disgust, and neutral) posed by 10 Japanese female models. The 10 expressers posed 3 or 4 examples of each of the seven facial expressions. Additionally, the proposed FEC methodology is evaluated on the 2009 version of the MMI Facial Expression Database [39], from which 235 images were extracted, showing the 7 standard facial expressions posed by both male and female models in the frontal face position. On certain occasions, expressers posed just 1 example of an expression. It is noted that, when implementing a LDA process, neither the JAFFE nor the MMI databases avail a sufficient number of training examples, i.e. larger than the dimensionality of the samples. So, in our experiments, the solution in [40] was utilized, which combines local discriminant hyper-planes, whose dimensionality is set to the number of available training samples per class minus 1, to overcome the problem with $\mathbf{S}_W$ being singular.

A simple preprocessing step is applied to the JAFFE and MMI images before performing FEC. Each face is manually cropped by taking as reference the hairline, the left and right cheek and the chin of each face. Next, the average ratio between the vertical and horizontal dimensions of all the cropped images was calculated to be 1.28 and used to resize/down-sample the cropped images to $50 \times 39$ pixels using bicubic interpolation. Lastly, histogram equalization is applied in order to limit differences in illumination between the images, thus completing the preprocessing step. Note that facial expression changes produce variations throughout relatively large areas of the face, e.g. for 'surprise' the eye-brows move up and for 'happy' the lips stretch out. Therefore, it is not critical to maintain high-resolution images, since it is expected that observing low-frequency features should be sufficient for FEC. It is noted that more sophisticated preprocessing methods can be used, such as producing fiducial grids, with nodes manually being placed on facial landmarks of the face [29, 22] to properly align the facial features between all faces. To experimentally evaluate the proposed SFRCS method, we set $M_o = 6$ and $M_s = 4$, which results to obtaining $M = 24$ 2-D Gabor features that correspond to 6 different orientations and 4 different scales. Furthermore, for the salient feature selection process, we set $N = 4$, so each SFV is composed by a selection of 4 Gabor features, out of the 24 total that are created. The selection of $N$=4 was made using a process where $N$ was varied from 1-to-24 (concurrently for all 21 expression pairs) and the overall (averaged over all 21 pairs) class separability measure was found to reach its maximum value for $N$=4. Fig. 5a shows an expresser posing the 7 basic facial expressions, whereas Figs. 5b and 5c show the magnitude of the 24 gabor features that were extracted from the two images in Fig. 5a that express 'happiness' and 'surprise'. Each row corresponds to one of the four scales and each column to one of the six orientations of the Gabor functions.

In our experiments, when ‘happiness’ was to be distinguished from ‘surprise’, the following 4 features were selected, in this order, as the salient set of features – based on the depiction in Figs. 5b and 5c: ($2^{nd}$ row , $4^{th}$ column) , ($3^{nd}$ row , $3^{th}$ column) , ($1^{nd}$ row , $5^{th}$ column) , and ($3^{nd}$ row , $2^{th}$ column).

Now that all the parameters of SFRCS have been set, a choice needs to be made as to which testing protocols should be used to evaluate its classification performance. To do so, we implement experiments that correspond to two commonly used scenarios. In the first scenario, it is assumed that the FEC system is familiar with the subjects whose images are to be classified. That is, the set used to train the system also includes ‘previous’ facial expression examples from these subjects. Essentially, this translates to using the common ‘leave-one-sample-out’ (L-O-sample-O) evaluation strategy [14, 15, 16, 17, and 41]. During each run of the L-O-sample-O strategy, one specific image is selected as the test data, whereas the remaining images are used to train the classification system. This strategy makes maximal use of the available data for training. This process is repeated 213 times on the JAFFE and 235 on the MMI database, so that all the images in the database will represent the test set once. Then, the 213, or 235, classification results are averaged and the final FEC rate is produced. The FEC rate of the SFRCS algorithm when evaluated under the L-O-sample-O strategy is calculated to be 96.71% on the JAFFE and 93.61% on the MMI database. Table 1 shows the corresponding mean confusion matrix that analyzes the confusion between the 7 expressions when applying SFRCS on the JAFFE database. This confusion matrix illustrates that, overall, the expressions are detected with high accuracy and that the largest confusion occurs when classifying images that correspond to the ‘anger’ and ‘disgust’ expressions. Specifically, ‘disgust’ is confused with ‘fear’ 6.90% of the time, whereas ‘anger’ is confused with ‘disgust’ 6.67% of the time.

The second scenario that is employed in order to evaluate the classification performance of SFRCS, assumes that the FEC system is not familiar with the subjects whose images are to be classified. This translates to using the also common ‘leave-one-subject-out’ (L-O-subject-O) evaluation strategy [15, 16, 18, and 41] which is more indicative of the ability of the system to generalize its performance and recognize a new person’s expressions. During each run of the L-O-subject-O strategy, the training set does not contain any examples (images) that correspond to the identity of the test subject. Therefore, for the JAFFE database, the test set is composed by images from only one subject, out of the 10 total, whereas the remaining images compose the training set. It is noted that the leave-one-subject-out evaluation process could not be applied to the MMI database since quite a few of the expressers posed certain expressions only once. The FEC rate of the SFRCS algorithm when evaluated on the JAFFE database under the L-O-subject-O strategy is calculated to be 85.92%. Table 2 shows the

corresponding mean confusion matrix that analyzes the confusion between the 7 expressions when applying SFRCS. This confusion matrix shows that, overall, expressions are detected with less accuracy under the L-O-subject-O than under the L-O-sample-O strategy. This is an expected observation since, now, the classification system is not trained with examples of expressions from the test person. Once again, we observe that the 'disgust' and, most notably, the 'anger' expression present the largest difficulties to the classification system. Specifically, 'anger' is misclassified as 'sadness' 26.67% of the time, and 'disgust' as 'anger' 17.24% of the time. Tables 1 and 2 show that significant drops in classification performance are observed when moving from the leave-one-sample-out to the leave-one-subject-out evaluation strategy. Specifically, the rate for 'anger' drops by 46.66%, for 'disgust' by 24.13%, and for 'fear' by 12.50%. This is a strong indication that the three expressions, in this order, are expressed differently from one subject to the next. For example, one person may express disgust a lot differently than a second person would. Therefore, 'disgust' is expressed less consistently over different subjects than, e.g., 'happiness' is. Conversely, people seem to have – to some degree – more personalized, or less consistent, ways to express anger, disgust, or fear, than to express happiness or sadness.

Next, we experimentally investigate the benefits of applying the proposed RCS classification scheme that is presented in sub-section 4.2. To do so, we develop a variant of SFRCS, where the RCS classification scheme is replaced by the ICS of sub-section 4.1. Onwards, we refer to this variant as Salient Feature Selection with ICS (SFS-ICS). As discussed in section 4, the ICS and RCS classification schemes share the same properties, rendering a comparison between the two particularly useful in isolating the benefits of selecting the most reliable classifiers. Tables 3 and 4 present the SFRCS and SFS-ICS FEC rates. Table 3 indicates that the difference between the performance of the classification schemes used by SFRCS and SFS-ICS is more significant for the L-O-subject-O, rather than for the L-O-sample-O, experiments. It is anticipated that this is because most of the classifiers are well-trained for the L-O-sample-O experiments due to the maximal use of the training data. Hence, both classification schemes produce a FEC rate that exceeds 96%. Nevertheless, this is not the case for the L-O-subject-O experiments, since none of the images of the test face are included in the training set, thus, it is more crucial to identify the most reliable classifiers when producing the final classification decision. As a result, the RCS classification scheme of SFRCS presents a significantly better classification performance than SFS-ICS.

The performance of SFRCS is compared against both baseline and state-of-the-art contemporary methods that were also evaluated on the JAFFE database under the L-O-sample-O and/or L-O-subject-O evaluation strategies. The baseline algorithms that we implemented provide good classification solutions for overcoming the "small

sample size" (SSS) problem, where the facial image sample dimensionality is larger than the number of available training samples per class [42, 43]. As a result, the lack of sufficient training samples causes improper estimation of a linear separation hyper-plane between the classes. This is clearly the case for FEC on most facial expression databases, including JAFFE and MMI. Moreover, the two-class discriminant analysis that SFRCS uses can exacerbate the SSS problem, since each classifier uses examples that correspond to only two facial expressions. As a result, we code the Principal Component Analysis (PCA) followed by LDA (PCA+LDA) [44], the Regularized LDA (RLDA) [45], the Subclass Discriminant Analysis (SDA) [46], and the Weighted Piecewise LDA (WPLDA) [47] discriminant methods that were shown to handle the SSS problem effectively. The PCA+LDA method represents the traditional solution to the SSS problem. It applies LDA in a lower-dimensional PCA subspace, so as to discard the null space (i.e., the subspace defined by the eigenvectors that correspond to zero eigenvalues) of the within-class scatter matrix of the training data set. The RLDA method employs a regularized Fisher's separability criterion. The purpose of regularization is to reduce the high variance related to the eigenvalue estimates of the within-class scatter matrix, at the expense of potentially increased bias. The SDA method applies sub-class discriminant analysis by approximating the underlying distribution of each class as a mixture of Gaussians. Rather than working with complex nonlinear methods that require a large number of training samples to work properly, the classification problem is now solved by dividing each class into a set of subclasses. The WPLDA method overcomes the SSS problem by producing lower-dimensionality piecewise discriminant hyper-planes that are then weighted properly in order to produce the overall classification decision. For each of the methods in [44, 45, 46, and 47], we produce the set of $\frac{C(C-1)}{2}=21$ two-class discriminant features. Subsequently, FEC results are obtained by applying the RCS classification scheme of sub-section 4.2, which SFRCS also uses. Thus, comparable FEC results are produced that indicate how effective the SFRCS feature extraction and salient feature selection process is. Tables 3 and 4 show the corresponding classification rates. The FEC performance of SFRCS is significantly better than that of [44, 45, 46, or 47], thus, it is concluded that better quality features are obtained by SFRCS.

In addition, the feature selection and classifier training process of SFRCS is compared against the AdaBoost algorithm. For the comparison to make sense, the AdaBoost methodology is applied under the pair-wise classification framework using the adapted version of [48]. For each of the 21 pairs of facial expressions, AdaBoost is run for $N$ rounds and $N$ hypotheses are constructed, where each one uses a single feature. So, $N$=4 features are selected along with $N$ weak classifiers. The final hypothesis is a linear combination of the $N$

hypotheses, where the weights are inversely proportional to the training errors. Tables 3 and 4 show the classification results by selecting the *N* features, for all 21 classifiers, using AdaBoost, and utilizing ICS to produce the final decision. The process is referred to as AdaBoost-ICS and shows good performance, though not quite as good as the performance of SFS-ICS, where the class separability measure of (13) was utilized to select the *N* features, or SFRCS, which, in addition, benefits from the *C*-1 classifier selection process.

The second part of the experimental process is concerned with observing how the overall SFRCS FEC methodology competes against state-of-the-art FEC methods. Specifically, the [14, 15, 16, 17, and 18] methods that were summarized in section 1 are considered. All these algorithms attempt to solve the FEC problem by applying the L-O-sample-O and/or the L-O-subject-O evaluation strategies on the JAFFE database. The algorithms presented in [14] and their (24 total) variants are evaluated using the L-O-sample-O strategy. The classification rates that are achieved when classifying the 7 facial expressions range from 63.86% to 90.34%. The method that is proposed in [15] uses both the L-O-sample-O and L-O-subject-O strategies to evaluate its performance. Several variants of the proposed algorithm are considered, with the maximum L-O-sample-O performance being 85.79% and the maximum L-O-subject-O performance being 74.34%. Moreover, if the semantic information that the proposed algorithm uses is replaced by class label information, the L-O-sample-O performance increases to an impressive 98.36%, whereas the L-O-subject-O performance increases to 77.05%. The method in [16] also uses both the L-O-sample-O and L-O-subject-O strategies to evaluate its performance. Its L-O-sample-O rate is roughly 92.9% and its L-O-subject-O rate 86.75%. The L-O-sample-O strategy is used to evaluate the performance of [17]. Various classification rates are reported, corresponding to varying the dimensionality of the input features, i.e. the number of principal components, and the number of output features that are retained and fed to the nearest neighbor classifier. The maximum FEC rate that is achieved is 94.97%. The FEC performance of [18] is evaluated using the L-O-subject-O strategy. Results are reported for optimally reduced dimensions where a classification rate of 83.18% is achieved. The maximum FEC results that are obtained by the aforementioned methods are reported in Table 3. This table again shows that SFRCS competes well with the state-of-the-art solutions, with only [15] surpassing its performance under the L-O-sample-O evaluation strategy, but not for L-O-subject-O, and only [16] surpassing it under the L-O-subject-O strategy, but not for L-O-sample-O. Therefore, the SFRCS algorithm is proven to be a good learner by example that also possesses good generalization abilities. In fact, when both of these classification attributes are sought, SFRCS is the algorithm of choice, out of all competing FEC algorithms that are referenced in Table 3. Furthermore, SFRCS

is compared against multi-class LDA followed by a nearest neighbor classifier, which we refer to as C-classLDA-NN. C-classLDA-NN does not show as strong a performance as SFRCS does. In fact, its performance reduces dramatically under the leave-one-subject-out strategy, as is illustrated in Table 3. The results in Tables 3 and 4 justify our decision to utilize two-class LDA in order to solve the FEC problem, based on the analysis presented in section 2.

Table 4 shows FEC results when the MMI database was used to evaluate performance, with SFRCS showing the best performance. In general, lower FEC rates were achieved on the MMI than on the JAFFE database. This is probably due to the fact that, sometimes, the training set did not include examples of the true expression that were matching the identity of the test face. Moreover, it is likely that additional complexity is introduced when both genders are represented in the dataset. Next, we selected an equal amount of examples showing male-only and female-only expressers and run independent experiments in order to examine whether gender plays a role in FEC tasks. The results, shown in Table 4, indicate that gender may affect performance, since higher rates are achieved for male expressers. We can only speculate as to why this may be so. Perhaps male expressers may be more consistent in the way they express certain emotions, or perhaps the perceived quality of a posed expression is affected by gender. For example, in [49], empirical studies showed that when human subjects rated a series of facial expressions regarding the emotion expressed, male expressers were perceived to produce higher degrees of happiness, whereas female expressers were perceived to produce higher degree of anger.

## 6. CONCLUSION

In this paper, a novel FEC methodology is presented and evaluated. The SFRCS algorithm produces high-quality features and implements a classification scheme, where results from the most reliable classifiers are integrated in order to produce the classification decision. Moreover, it was shown why a FEC approach that solves a set of two-class problems is expected to produce better results than the ones produced by the multi-class or one-against-all classification approaches. The SFRCS methodology was tested on the JAFFE and MMI databases under the common L-O-sample-O and L-O-subject-O evaluation strategies. Results indicate that SFRCS provides a good solution to the FEC problem by producing classification rates of 96.71% and 85.92% on the JAFFE and 93.61% on the MMI database, and that it compares well with state-of-the-art methods. It is anticipated that the performance of other FEC methods can be enhanced by utilizing processes that stem from this framework in order to produce high-quality features and/or implement fast and accurate classification schemes.

ACKNOWLEDGEMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 211471 (i3DPost). Portions of the research in this paper use the MMI-Facial Expression Database collected by M. Pantic and her group (www.mmifacedb.com).

REFERENCES

[1] B. Fasel and J. Luettin, "Automatic facial expression analysis: a survey", *Pattern Recognition, Elsevier*, vol. 36, no. 1, pp. 259-275, Jan. 2003.

[2] M. Pantic and J. M. Rothkrantz, "Automatic Analysis of Facial Expressions: The State of the Art", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp.1424-1445, Dec. 2000.

[3] W. Fellenz, J. Taylor, N. Tsapatsoulis, and S. Kollias, "Comparing template-based, feature-based and supervised classification of facial expression from static images", in *Proc. IMACS/IEEE Int. Multi-conf. on Circuits, Systems, Communications, and Computers*, vol. 1, pp. 5331-5336, July 1999.

[4] I. S. Pandzic and R. Forchheimer, Eds., *MPEG-4 Facial Animation*. New York: Wiley, 2002.

[5] M. Pantic, A. Pentland, A. Nijholt, and T.S. Huang, "Human computing and machine understanding of human behavior: A survey", in *Artificial Intelligence for Human Computing*, T.S. Huang, A. Nijholt, M. Pantic & A. Pentland, Eds. Springer, Lecture Notes in Artificial Intelligence, vol. 4451, pp. 47-71, 2007.

[6] M. Pantic and I. Patras, "Dynamics of facial expression: Recognition of facial actions and their temporal segments from face profile image sequences", *IEEE Trans. on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 36, no.2, pp. 433-449, April 2006.

[7] J. Russell and J. Fernandez-Dols, *The Psychology of Facial Expression*. New York: Cambridge Univ. Press, 1997.

[8] A. Mehrabian, "Communication without words", *Psychology Today*, vol. 2, no. 4, pp. 53-56, Sep. 1968.

[9] D. Keltner and P. Ekman, "Facial expression of emotion", in *Handbook of Emotions*, M. Lewis and J. M. Haviland-Jones, Eds. New York: Guildford, pp. 236-249, 2000.

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

[11] P. Ekman and W. Friesen, *Facial Action Coding System: A technique for the measurement of facial movement*. Palo Alto, CA: Consulting Psychologists Press, 1978.

[12] A. J. Calder, R. Duncan, A. W. Young, I. Nimmo-Smith, J. Keane, and D. I. Perrett, "Caricaturing facial expressions", *Cognition*, vol. 76, no. 2, pp. 105-146, Aug. 2000.

[13] A. J. Calder, A. M. Burton, P. Miller, A. W. Young, and S. Akamatsu, "A principal component analysis of facial expressions", *Vision Research*, vol. 41, no. 9, pp. 1179-1208, April 2001.

[14] I. Buciu, C. Kotropoulos, and I. Pitas, "ICA and Gabor representation for facial expression recognition", in *Proc. IEEE Int. Conf. on Image Processing*, vol. 2, no. 3, pp. 855-858, Barcelona, Spain, Sep. 14-17, 2003.

[15] W. Zheng, X. Zhou, C. Zou, and L. Zhao, "Facial expression recognition using kernel canonical correlation analysis (KCCA)", *IEEE Trans. on Neural Networks*, vol. 17, no. 1, pp. 233-238, Jan. 2006.

[16] D. Liang, J. Yang, Z. Zheng and Y. Chang, "A facial expression recognition system based on supervised locally linear embedding", *Elsevier, Pattern Recognition Letters*, vol. 26, no. 15, pp. 2374-2389, Nov. 2005.

[17] N. Kwak, "Feature extraction for classification problems and its application to face recognition", *Elsevier, Pattern Recognition*, vol. 41, no. 5, pp. 1718-1734, May 2008.

[18] H. Wang, S. Chen, Z. Hu, and W. Zheng, "Locality-preserved maximum information projection", *IEEE Trans. on Neural Networks*, vol. 19, no. 4, pp. 571-585, April 2008.

[19] M.S. Bartlett, G. Littlewort, I. Fasel, and J.R. Movellan, "Real time face detection and expression recognition: Development and application to human-computer interaction", in *Proc. CVPR Workshop on Computer Vision and Pattern Recognition for Human-Computer Interaction*, 2003.

[20] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. 13th Int. Conf. on Machine Learning*, pp. 148–146. Morgan Kaufmann, 1996.

[21] G. Zhao and P. Matti, "Dynamic texture recognition using local binary patterns with an application to facial expressions", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 915–928, June 2007.

[22] I. Kotsia and I. Pitas, "Facial expression recognition in image sequences using geometric deformation features and Support Vector Machines", *IEEE Trans. on Image Processing*, vol. 16, no. 1, pp. 172–187, Jan. 2007.

[23] P.S. Aleksic, A.K. Katsaggelos, "Automatic facial expression recognition using facial animation parameters and multistream HMMs", *IEEE Trans. on Information Forensics and Security*, vol. 1, no. 1, pp. 3-11, March 2006.

[24] Y. Zhang and Q. Ji, "Active and dynamic information fusion for facial expression understanding from image sequences", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 699-714, May 2005.

[25] I. Cohen, N. Sebe, A. Garg, L. S. Chen, and T. S. Huang, "Facial expression recognition from video sequences: temporal and static modeling", *Computer Vision and Image Understanding, Special Issue on Face Recognition*, vol. 91, issues 1-2, pp. 160-187, July-August 2003.

[26] V. Bruce, "Fleeting images of shade: Identifying people caught on video", *The Psychologist*, vol. 11, no. 7, July 1998.

[27] D. Fidaleo and M. M. Trivedi, "Manifold Analysis of Facial Gestures for Face Recognition", *in Proc. ACM SIGMM: Workshop on Multimedia Biometrics Methods and Applications*, pp. 65-69, Berkeley, California, Nov. 8, 2003.

[28] R.A. Fisher, "The use of multiple measurements in taxonomic problems", in *Annals of Eugenics*, vol. 7, no. 2, pp. 179-188, 1936.

[29] A. Beygelzimer, J. Langford, and B. Zadrozny, "Weighted one-against-all", in *Proc. AAAI Conf. on 20th Artificial Intelligence and the 7th Innovative Applications of Artificial Intelligence*, pp. 720-725, Pittsburgh, July 9-13, 2005.

[30] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Academic Press, San Diego, CA, 1990.

[31] O.C. Hamsici and A.M. Martinez, "Bayes optimality in linear discriminant analysis", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 647-657, April, 2008.

[32] J. H. Friedman. Another approach to polychotomous classification. Technical report, Stanford University, Oct. 1996.

[33] M. J. Lyons, J. Budynek, and S. Akamatsu, "Automatic classification of single facial images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1357-1362, Dec. 1999.

[34] P.A. Devijver and J. Kittler, *Pattern Recognition: a statistical approach*. Prentice-Hall International, Inc., London, 1982.

[35] S. Sharma and S. Sharma, *Applied Multivariate Techniques*, Wiley, New York, 1995.

[36] Y. Yu and F. Song, "Feature extraction based on a linear separability criterion", *Int. Journal of Innovative Computing, Information, and Control*, vol. 4, no. 4, pp. 857-865, April 2008.

[37] F. Hanary and E. M. Palmer, *Graphical Enumeration*. New York, NY: Academic Press, 1973.

[38] F. Hanary, *Graph Theory*. Reading, MA: Addison-Wesley, 1994.

[39] M.Pantic, M.F. Valstar, R. Rademaker and L. Maat, "*Web-based database for facial expression analysis"*, in Proc. IEEE Int. Conf. on Multimedia and Expo (ICME'05), Amsterdam, The Netherlands, July 2005.

[40] M. Kyperountas, A. Tefas, and I. Pitas, "Face verification using locally linear discriminant models", in *Proc. of Int. Conf. on Image Processing*, vol. 4, pp. 469-472, San Antonio, Sep. 16-19, 2007.

[41] S. Klanke and H. Ritter, "A Leave-K-Out cross-validation scheme for unsupervised kernel regression", in *Proc. Int. Conf. on Artificial Neural Networks*, vol. 2, pp. 427-436, Athens, Greece, Sep. 10-14, 2006.

[42] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos, "Boosting linear discriminant analysis for face recognition", in *Proc. IEEE Int. Conf. on Image Processing*, vol. 1, pp. I - 657-60, Barcelona, Spain, Sep. 14-17, 2003.

[43] M. Kyperountas, A. Tefas, and I. Pitas, "Dynamic training using multistage clustering for face recognition", *Pattern Recognition*, Elsevier, vol. 41, no. 3, pp. 894-905, March 2008.

[44] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, Jul. 1997.

[45] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, "Regularization studies of linear discriminant analysis in small sample size scenarios with application to face recognition", *Pattern Recognition Letters*, vol. 26, no. 2, pp. 181-191, Jan. 2005.

[46] M. Zhu and A.M. Martinez, "Subclass discriminant analysis", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1274-1286, Aug. 2006.

[47] M. Kyperountas, A. Tefas, and I. Pitas, "Weighted piecewise LDA for solving the small sample size problem in face verification", *IEEE Trans. on Neural Networks*, vol. 18, no. 2, pp. 506-519, March 2007.

[48] K. Tieu and P. Viola, "Boosting image retrieval", in *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, vol. 1, pp. 228–235, 2000.

[49] U. Hess, R. B. Adams Jr., and R. E. Kleck, "Facial appearance, gender, and emotion expression", *Emotion* , vol. 4, no. 4, pp. 378-388, 2004.

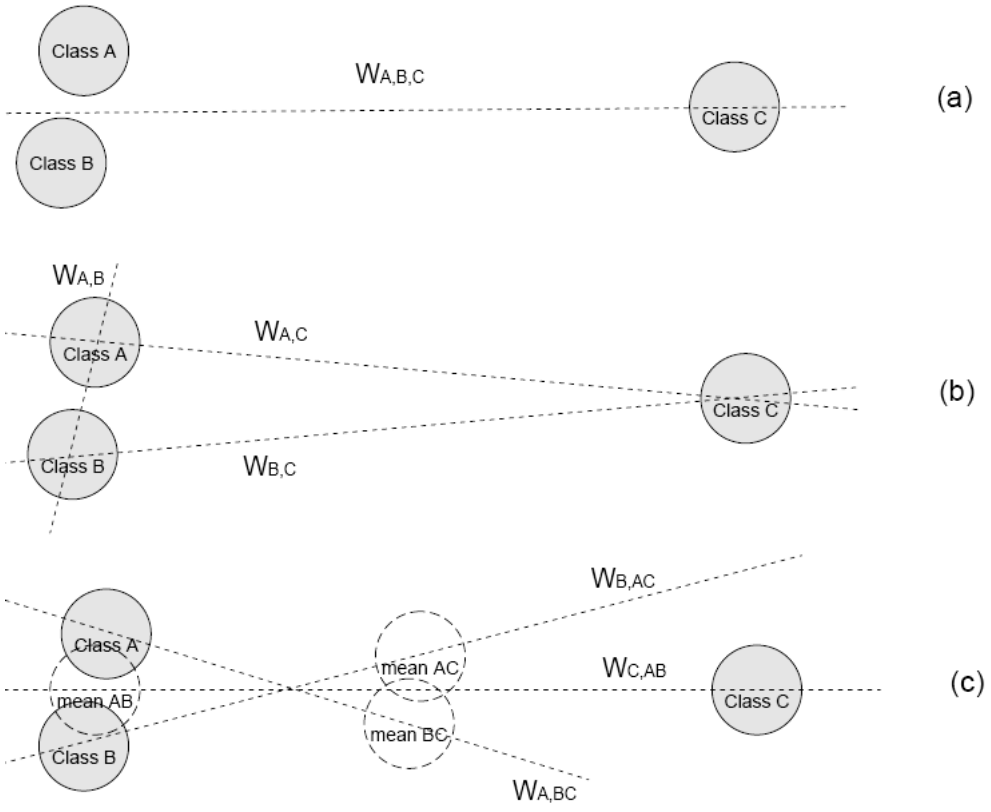



**Fig. 1:** (a) Discriminant hyper-plane ($\mathbf{w}_{A,B,C}$) produced by a multi-class LDA process attempting to discriminate between the three two-dimensional classes A, B, and C. (b) Discriminant hyper-planes ($\mathbf{w}_{A,B}, \mathbf{w}_{A,C}, \mathbf{w}_{B,C}$) produced by the three two-class LDA processes attempting to discriminate between the two-dimensional classes A-B, A-C, and B-C, respectively. (c) Discriminant hyper-planes ($\mathbf{w}_{A,BC}, \mathbf{w}_{B,AC}, \mathbf{w}_{C,AC}$) produced by the three one-against-all LDA processes attempting to discriminate between the two-dimensional classes A-BC, B-AC, and C-AC, respectively. The means of the class pairs BC, AC, and AC are illustrated as dotted circles.

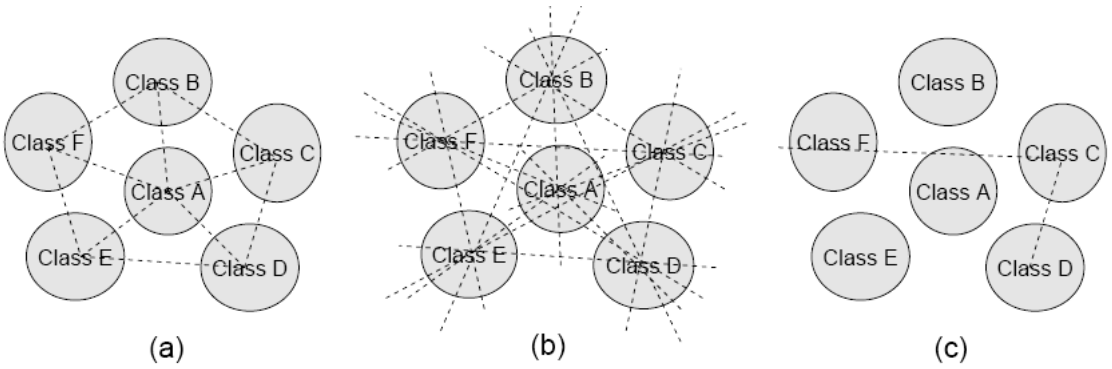



**Fig. 2:** Classification solutions, under different approaches, that need to be produced in order to discriminate between the two-dimensional classes A, B, C, D, E, and F. The dotted lines show the projection coordinates that are required by each approach. (a) Classification via a multi-class classifier where a separation between all classes is produced. (b) Classification via a set of two-class classifiers. One linear projection is required in order to produce a separation between any two classes. (c) Classification via one-against-all classifiers that need to produce a separation between one class, in this case class C, and the remaining classes.

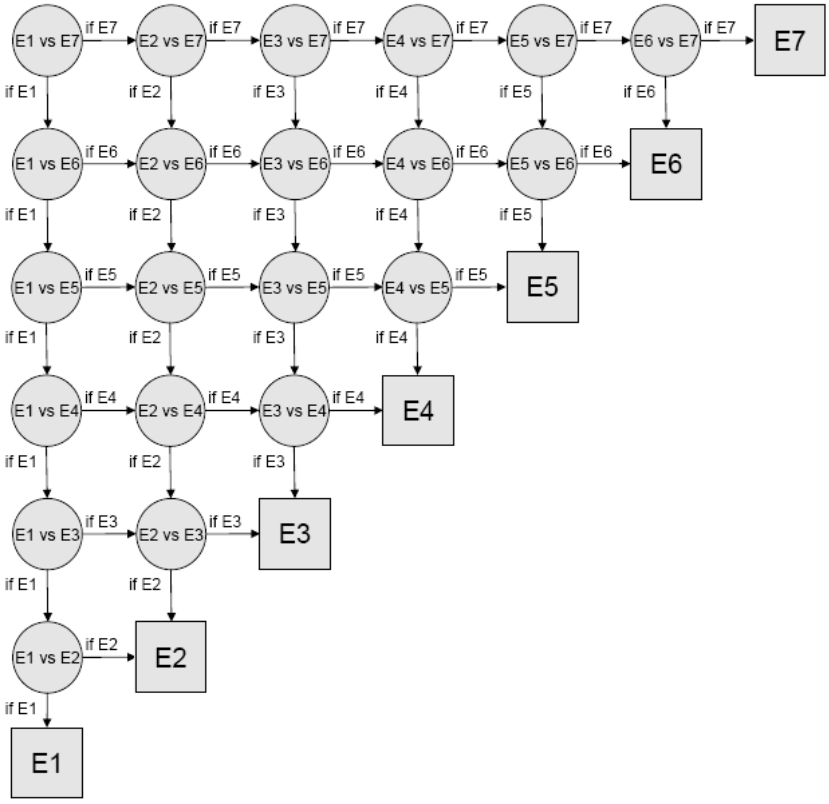



**Fig. 3:** The ICS organization implemented using two-class classifiers. This topology is optimized in the sense that if a class is rejected at any point during the classification process, none of the remaining paths that can be followed considers that rejected class as a possible outcome.

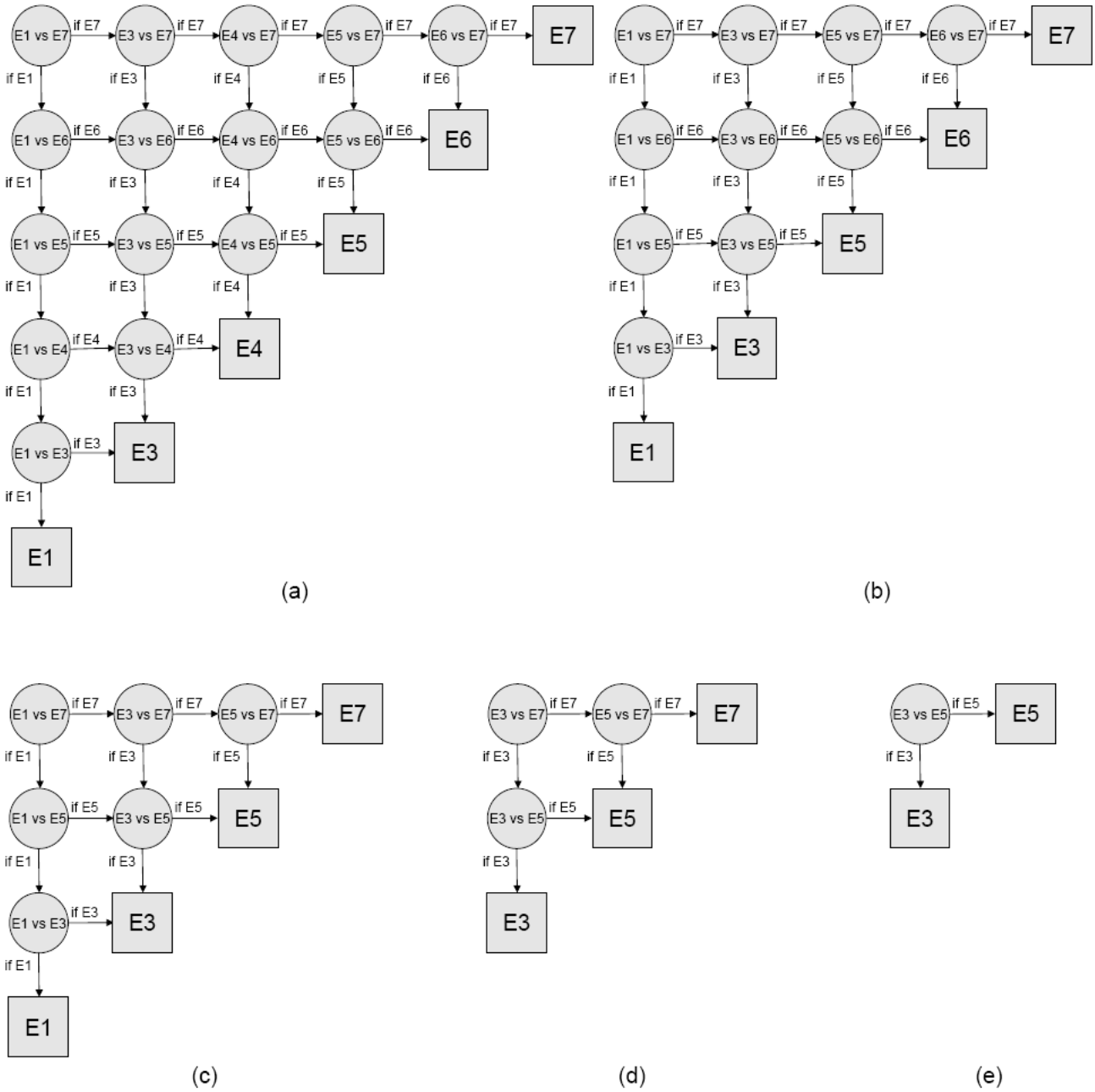
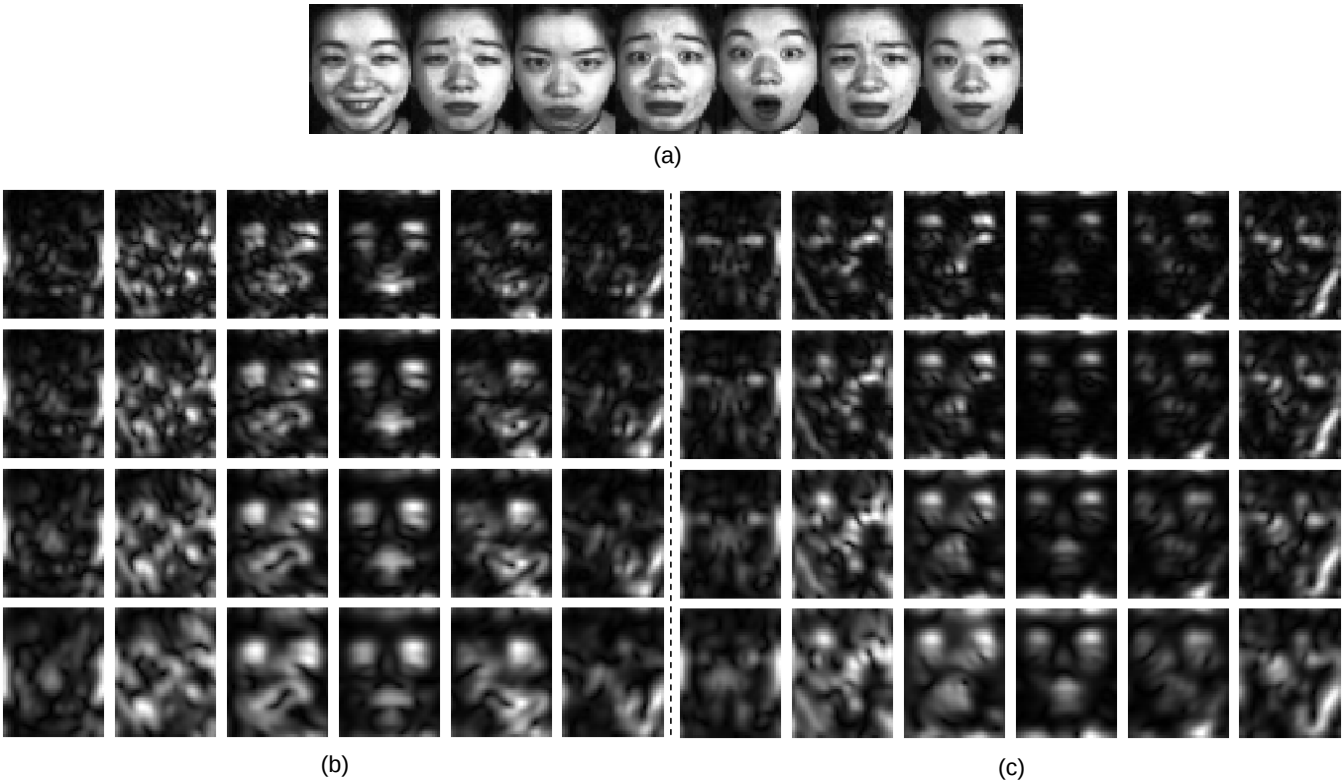



**Fig. 4:** Example of the RCS classification scheme illustrating how the ICS classifier topology dynamically reduces after each step, by utilizing the class separability measure: (a) Classification Step 1, where E2 gets discarded, (b) Classification Step 2, where E4 gets discarded, (c) Classification Step 3, where E6 gets discarded, (d) Classification Step 4, where E1 gets discarded, (e) Classification Step 5, where E7 gets discarded, and Classification Step 6, where the final decision is made between classes E3 and E5.



(a) (b) (c)

**Fig. 5:** (a) Portrayal of happiness, sadness, anger, fear, surprise, disgust, and neutral expression. (b) Magnitude of the 24 (4 x 6) Gabor features that were extracted from the face image in (a) that expresses 'happiness'. (c) Magnitude of the 24 (4 x 6) Gabor features that were extracted from the face image in (a) that expresses 'surprise'.

**Table 1:** Mean confusion matrix for FEC using SFRCS under the leave-one-sample-out evaluation strategy.

| Expression | Happiness | Sadness | Anger | Fear | Surprise | Disgust | Neutral |
|---|---|---|---|---|---|---|---|
| Happiness | 96.97% | 0% | 0% | 0% | 0% | 0% | 3.03% |
| Sadness | 0% | 100% | 0% | 0% | 0% | 0% | 0% |
| Anger | 0% | 0% | 93.33% | 0% | 0% | 6.67% | 0% |
| Fear | 0% | 0% | 0% | 96.88% | 3.13% | 0% | 0% |
| Surprise | 0% | 0% | 0% | 0% | 96.55% | 0% | 3.45% |
| Disgust | 0% | 0% | 0% | 6.90% | 0% | 93.10% | 0% |
| Neutral | 0% | 0% | 0% | 0% | 0% | 0% | 100% |

**Table 2:** Mean confusion matrix for FEC using SFRCS under the leave-one-subject-out evaluation strategy.

| Expression | Happiness | Sadness | Anger | Fear | Surprise | Disgust | Neutral |
|---|---|---|---|---|---|---|---|
| Happiness | 100% | 0% | 0% | 0% | 0% | 0% | 0% |
| Sadness | 0% | 100% | 0% | 0% | 0% | 0% | 0% |
| Anger | 0% | 26.67% | 46.67% | 3.33% | 0% | 13.33% | 10% |
| Fear | 0% | 15.63% | 0% | 84.38% | 0% | 0% | 0% |
| Surprise | 0% | 0% | 0% | 0% | 100% | 0% | 0% |
| Disgust | 0% | 6.90% | 17.24% | 6.90% | 0% | 68.97% | 0% |
| Neutral | 0% | 0% | 0% | 0% | 0% | 0% | 100% |

**Table 3:** Classification performance of various FEC methods on the JAFFE database.

| Method | Leave-one-sample-out | Leave-one-subject-out |
|---|---|---|
| | FEC rate | FEC rate |
| PCA+LDA [44] | 64.32% | 46.48% |
| RLDA [45] | 77.96% | 54.31% |
| SDA [46] | 72.25% | 49.47% |
| WPLDA [47] | 85.92% | 58.53% |
| GWs+SVMs [14] | 90.34% | - |
| Class-label KCCA [15] | **98.36**% | 77.05% |
| SLLE [16] | 92.90% | **86.75**% |
| ICA-FX [17] | 94.97% | - |
| LPMIP [18] | - | 83.18% |
| AdaBoost-ICS | 94.84% | 81.41% |
| C-classLDA-NN | 89.67% | 74.73% |
| SFS-ICS | 96.02% | 82.28% |
| SFRCS | 96.71% | 85.92% |

**Table 4:** Classification performance of various FEC methods on the MMI database using the leave-one-sample-out evaluation strategy.

| Method | Full Set<br>FEC rate | Male Subset<br>FEC rate | Female Subset<br>FEC rate |
|---|---|---|---|
| PCA+LDA [44] | 60.43% | 62.79% | 56.98% |
| RLDA [45] | 69.36% | 74.42% | 70.93% |
| SDA [46] | 63.83% | 68.60% | 63.95% |
| WPLDA [47] | 81.70% | 82.56% | 79.07% |
| AdaBoost-ICS | 90.21% | 90.70% | 89.53% |
| C-classLDA-NN | 84.68% | 88.37% | 87.21% |
| SFS-ICS | 91.48% | 91.86% | 90.70% |
| SFRCS | **93.61%** | **94.19%** | **93.02%** |