Facial Expression Recognition based on Multi-view Observations with Application to Social Robotics

Bogdan Raducanu[1], Alireza Bosaghzadeh [2] and Fadi Dornaika[2,3]

[1] Computer Vision Center, 08193 Bellaterra, Barcelona, Spain
bogdan@cvc.uab.es
[2] University of the Basque Country UPV/EHU, San Sebastian, Spain
alireza.bossaghzadeh@gmail.com
[3] IKERBASQUE, Basque Foundation for Science, Bilbao, Spain
fadi.dornaika@ehu.es

Abstract. Human-robot interaction is a hot topic nowadays in the social robotics community. One crucial aspect is represented by the affective communication which comes encoded through the facial expressions. In this paper, we propose a novel approach for facial expression recognition, which exploits an efficient and adaptive graph-based label propagation (semi-supervised mode) in a multi-observation framework. The facial features are extracted using an appearance-based 3D face tracker, view- and texture independent. Our method has been extensively tested on the CMU dataset, and has been conveniently compared with other methods for graph construction. With the proposed approach, we developed an application for an AIBO robot, in which it mirrors the recognized facial expression.

1 Introduction

In the field of Human-Computer Interaction (HCI), computers will be enabled with perceptual capabilities in order to facilitate the communication protocols between people and machines. In other words, computers will be endowed with natural ways of communication people use in their everyday life. Among them, facial expression represents a powerful mean people use to express their emotions and other aspects related with their social or psychological status.

In the past, a lot of effort was dedicated to recognize facial expression in still images (static recognition). For this purpose, many techniques have been applied: neural networks [1], Gabor wavelets [2] and Active Appearance Models (AAM) [3]. A very important limitation to this strategy is the fact that still images usually capture the apex of the expression, i.e., the instant at which the indicators of emotion are most marked. More recently, attention has been shifted particularly towards modelling dynamical facial expressions [4]. Recent research has shown that it is not just the particular facial expression, but also the associated dynamics that are important when attempting to decipher its

meaning. The dynamics of facial expression can be defined as the intensity of the Action Units coupled with the timing of their formation. This is a very relevant observation, since for most of the communication act, people rather use subtle facial expressions than showing deliberately exaggerated poses in order to convey their message. In [5], the authors found that subtle expressions that were not identifiable in individual images suddenly became apparent when viewed in a video sequence.

More recently, attention has been shifted particularly towards dynamic modelling of facial expressions. Dynamical approaches can use shape deformations, texture dynamics [6] or a combination of them [7]. Dynamical classifiers try to capture the temporal pattern in the sequence of feature vectors related to each frame such as the Hidden Markov Models (HMMs) and Dynamic Bayesian Networks [8]. In [7], the authors propose a dynamic recognition based on the differential Active Appearance Model parameters. A sequence of input frames is fitted using the classical AAM then a specific frame is selected as reference frame. Then the corresponding sequence of differential AAM parameters is recognized by computing the directed Hausdorff distance and the K Nearest Neighbor classifier. In [9], a two-stage approach is used. Initially, a linear classification bank was applied and its output was fused to produce a characteristic signature for each universal facial expression. The signatures thus computed from the training data set were used to train discrete Hidden Markov Models to learn the underlying model for each facial expression. In [10], the authors propose a Bayesian approach to modelling temporal transitions of facial expressions represented in a manifold. In [4], the authors propose a dynamic classifier that is based on building spatio-temporal model for each universal expression derived from Fourier transform. The recognition of unseen expressions uses Hausdorff distance in order to compute dissimilarity values for classification. Local Binary Patterns (LBP) have been used for facial expression recognition in [11, 12].

Modelling the variability of facial expressions is a very challenging task. Facial expressions form a class of *objects* with a well-defined structure which suffers elastic deformations due to changes in face appearance. Ideally, an optimal representation would be able to cope with all these complex transformations. Furthermore, the majority of view-based methods often require very tedious learning stages. In order to overcome the above limitations, multi-observation based recognition can offer an alternative. In this case, the observations can be either a temporal sequence of face images (video sequence) or just a subset of images. Obviously, recognizing a facial expression by using more than a single snapshot can improve the performance of recognition systems since the test images contain more information that may include more variations that help reducing ambiguities that affect single image based recognition systems. Most of video-based recognition methods use complicated training schemes in order to classify the multiple observations (e.g., [13]). In the context of semi-supervised learning, graph-based label propagation can be seen as a powerful tool that solves the multi-observation recognition problem. In [14], the authors proposed a graph-based label propagation method that can infer the labels of unknown

observations by optimizing a penalty function based on label consistency. In [15], the authors extended the work of [14] by including the constraint that multiple observations have the same label. However, in both works the graph was constructed in an ad-hoc way, that is, it uses a K-NN graph [16].

In this paper, we propose a graph construction method that is based on efficient and adaptive coding scheme. We use the obtained graph in order to infer the label of multi-observation (semi-supervised mode). Our approach will be tested on the public CMU facial expression database [17]. The paper is structured as follows. Section 2 describes the extraction of temporal signatures associated with universal expressions. Section 3 introduces our new approach for graph construction and multi-observation recognition based on label propagation. In section 4 we report the experimental results. In section 5, we present an application scenario for our method, in which an AIBO robot is mirroring the facial expression perceived. Finally, in section 6 we draw our conclusions.

2 Modelling facial expressions from videos

The objective of this work is to recognize facial expressions in continuous videos using data-driven machine learning algorithms. Therefore, encoding the displayed universal expressions is a crucial step. Extracting facial dynamics associated with facial muscle deformations from video sequences is a challenging task. This task is made more difficult if the subject's head moves in 3D space. The recognition of facial expressions with significant head motion is required by many applications such as human computer interaction and computer graphics animation [18–20] as well as training of social robots [21, 22].

2.1 Modelling faces

In our work, we use a common 3D deformable face model—the *Candide* model [23] (See Figure 1). Despite the simplicity of this 3D wireframe model, it can be used to extract a subset of 3D facial dynamics in real time using one single camera. The 3D shape of this wireframe model is directly recorded in coordinate form. It is given by the coordinates of the 3D vertices $\mathbf{P}_i, i = 1, \ldots, n$ where n is the number of vertices. Thus, the shape up to a global scale can be fully described by the $3n$-vector $\mathbf{g}$; the concatenation of the 3D coordinates of all vertices $\mathbf{P}_i$. The vector $\mathbf{g}$ is written as:

$$\mathbf{g} = \mathbf{g}_s + \mathbf{A}\,\boldsymbol{\tau_a} \tag{1}$$

where $\mathbf{g}_s$ is the static shape of the model, $\boldsymbol{\tau_a}$ the animation control vector, and the columns of $\mathbf{A}$ are the Animation Units. In this study, we use six modes for the facial Animation Units (AUs) matrix $\mathbf{A}$. We have chosen the following AUs: lower lip depressor, lip stretcher, lip corner depressor, upper lip raiser, eyebrow lowerer, outer eyebrow raiser (see Figure 1**.(a)**). These AUs are enough to cover most common facial animations. Moreover, they are essential for conveying emotions.

In equation (1), the 3D shape is expressed in a local coordinate system. However, one should relate the 3D coordinates to the image coordinate system. To this end, we adopt the weak perspective projection model. We neglect the perspective effects since the depth variation of the face can be considered as small compared to its absolute depth. Thus, the state of the 3D wireframe model is given by the 3D face pose parameters (three rotations and three translations) and the internal face animation control vector $\boldsymbol{\tau_a}$. This is given by the 12-dimensional vector $\mathbf{b}$:

$$\mathbf{b} = [\theta_x, \theta_y, \theta_z, t_x, t_y, t_z, \boldsymbol{\tau_a}^T]^T \tag{2}$$

Note that if only the aspect ratio of the camera is known, then the component t_z is replaced by a scale factor having the same mapping role between 3D and 2D. In this case, the state vector is given by (s denotes the scale factor):

$$\mathbf{b} = [\theta_x, \theta_y, \theta_z, t_x, t_y, s, \boldsymbol{\tau_a}^T]^T \tag{3}$$

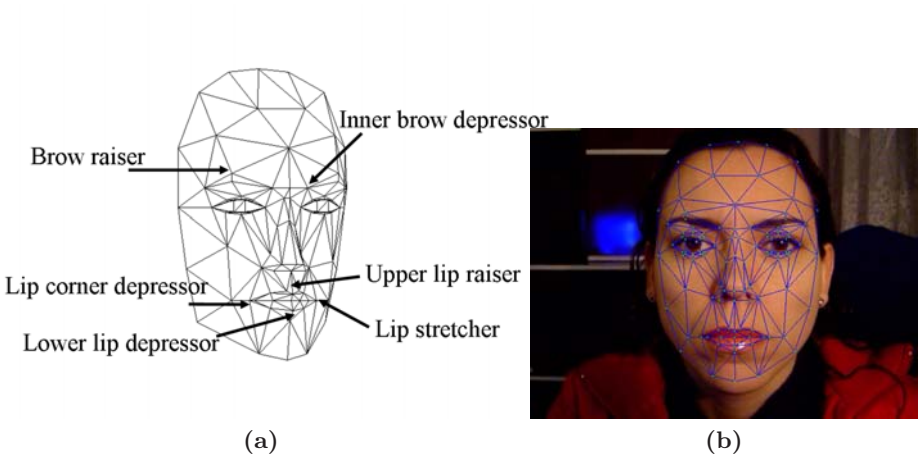

Fig. 1. (a) *Candide* model. **(b)** *Candide* model adapted to an input facial image.

2.2 Simultaneous face and facial action tracking

In order to recover the facial expression one has to compute the facial actions encoded by the vector $\boldsymbol{\tau_a}$ which encapsulates the facial deformation. Since our recognition scheme is view-independent these facial actions together with the 3D head pose should be simultaneously estimated. In other words, the objective is to compute the state vector $\mathbf{b}$ for every video frame.

For this purpose, we use the tracker based on Online Appearance Models [24]. This appearance-based tracker aims at computing the 3D head pose and the facial actions, i.e. the vector $\mathbf{b}$, by minimizing a distance between the incoming

warped frame and the current *shape-free* appearance of the face. This optimization is carried out using a gradient descent method. The statistics of the *shape-free* appearance as well as the gradient matrix are updated every frame. This scheme leads to a fast and robust tracking algorithm.

2.3 Representing dynamic universal expressions by features

In order to learn the spatio-temporal structures of the facial actions associated with facial expressions, we have used a simple supervised learning scheme that consists in two stages. In the first stage, training video sequences depicting different universal facial expressions are tracked using the appearance-based face tracker. The retrieved facial actions $\boldsymbol{\tau}_{\mathbf{a}}$ are represented by time series. In other words, an example (expression going from neutral to apex) is encoded by a sequence of facial actions $\boldsymbol{\tau}_{\mathbf{a}(1)}, \ldots, \boldsymbol{\tau}_{\mathbf{a}(T)}$. One can note that this temporal sequence (trajectory) can be considered as a compact representation of the spatio-temporal facial structure that one expects to observe whenever the face undergoes a given universal expression. In the second stage, since we are using example based classifiers all examples should have the same dimension. To this end, all facial action sequences are aligned in the time domain using the Dynamic Time Warping (DTW) technique [25]. Dynamic Time Warping is a well-known technique to find an optimal alignment between two given (time-dependent) sequences under certain restrictions. Thus, a given example (universal expression) is represented by a feature vector obtained by concatenating the vectors $\boldsymbol{\tau}_{\mathbf{a}}(t)$ belonging to the aligned temporal sequence.

More precisely, video sequences have been picked up from the CMU database [17]. These sequences depict five frontal view universal expressions (surprise, sadness, joy, disgust and anger). Each expression is performed by 7 different subjects, starting from the neutral one. Altogether we select 35 video sequences composed of around 15 to 20 frames each, that is, the average duration of each sequence is about half a second. The training video sequences have an interesting property: all performed expressions go from the neutral expression to a high magnitude expression by going through a moderate magnitude around the middle of the sequence.

In the final stage of the learning all training trajectories are aligned in the time domain using the Dynamic Time Warping technique by fixing a nominal duration for a facial expression. In our experiments, this nominal duration is set to 18 frames. This choice was guided by many observations that show that a complete expression can be displayed in 15-20 frames assuming that the video rate is 30 fps.

Finally, a training video sequence associated with a universal expression is represented by a time series corresponding to the second half of the aligned trajectory (only nine frames are used). Thus, the feature vector $\mathbf{y}$ can be any of the following vectors:

$$\left(\boldsymbol{\tau}_{\mathbf{a}(10)}^{T}, \boldsymbol{\tau}_{\mathbf{a}(11)}^{T}, \boldsymbol{\tau}_{\mathbf{a}(12)}^{T}, \boldsymbol{\tau}_{\mathbf{a}(13)}^{T}, \boldsymbol{\tau}_{\mathbf{a}(14)}^{T}, \boldsymbol{\tau}_{\mathbf{a}(15)}^{T}, \boldsymbol{\tau}_{\mathbf{a}(16)}^{T}, \boldsymbol{\tau}_{\mathbf{a}(17)}^{T}, \boldsymbol{\tau}_{\mathbf{a}(18)}^{T}\right)^{T}$$

The dimension of this vector is 6. Figure 2 shows nine frames associated with the joy expression.

We decided to remove in our analysis the first half trajectory (from initial, neutral state to half-apex) since we found them irrelevant for the purposes of the current study. Therefore, a feature vector associated with a given universal expression is encoding a signature of one realization of this expression that goes from a moderate magnitude to the apex.

Fig. 2. Constructing the feature vector (54 components) from nine frames associated with joy expression dynamics.

3 Multi-observation recognition based on label propagation

3.1 Overview

Graph-based methods have been been effective in a wide variety of domains like machine learning, computer vision and signal processing. Theoretical justification for graph-based methods is an area of active research [26–29]. The graph on which learning is performed is a central object for any graph-based method.

Graph-based methods operate on a graph where a node corresponds to a data instance and a pair of nodes are connected by a weighted edge.

On the other hand, semi-supervised learning (SSL) algorithms can be very appealing since they learn from limited amounts of labeled data combined with widely available unlabeled data. Of the current SSL methods, graph based approaches have emerged as methods of choice for general semi-supervised tasks in terms of accuracy and computational efficiency. Most of the graph based SSL algorithms concentrate primarily on the label inference part, i.e. assigning labels to nodes once the graph has already been constructed, with very little emphasis on construction of the graph itself. Only recently, the issue of graph construction has begun to receive attention [30, 31]. In fact, the way to establish high-quality graphs is still an open problem. At present, the most popular graph construction manner is based on the K nearest neighbor and ϵ-ball neighborhood criteria. Once a neighborhood graph is constructed, the edge weights are assigned by Gaussian Kernels or coefficients provided by local reconstruction algorithms [32, 33].

3.2 Proposed graph construction

Although graph-based methods have been successfully applied to several domains in machine learning, very little attention has been given to the graph construction part, with majority of the research focus devoted to the post-graph construction learning algorithms.

We devise a new efficient strategy for graph construction that simultaneously estimates the graph adjacency and its associated weight matrix.

In our proposed method, we construct the graph of a dataset by directly using the coding of any training samples with respect to the rest of the set. We were inspired by recent advances in collaborative coding, namely the Weighted Regularized Least Square (WRLS) minimization method proposed in [34]. In this work, the authors proposed a linear coding scheme in order to classify samples according to the collaborative reconstruction error. Their proposed criterion is based on the sum of three parts: (i) L_2 norm of the reconstruction error, (ii) a regularization term set to the L_2 norm of the coefficients vector, (iii) a weighted sum of the squared coefficients. Since the weights are set to the distances between the test sample and the training samples, a kind of sparsity is included in the global criterion.

Let $\mathbf{y}$ denote a given training or test sample and $\mathbf{X}$ denote the training set. In our work we use the following coding scheme in order to automatically generate the data graph:

$$\min_{\mathbf{a}} \left(\|\mathbf{y} - \mathbf{X}\,\mathbf{a}\|^2 + \sigma \sum_{i=1}^{N} p_i\, a_i^2 \right) \tag{4}$$

where σ is a regularization parameter having small positive value, $\mathbf{a} \in \mathbb{R}^N$ is the coefficient vector and the p_i's are positive weights.

The above optimization problem has the following closed form solution:

$$\mathbf{a}^\star = (\mathbf{X}^T\,\mathbf{X} + \sigma\,\mathbf{P})^{-1}\mathbf{X}^T\,\mathbf{y} \tag{5}$$

where $\mathbf{P}$ is $N \times N$ diagonal matrix whose diagonal elements P_{ii} are set to p_i. In our work, we use the following formula for the weights:

$$P_{ii} = p_i = 1 - \exp(-\|\mathbf{y} - \mathbf{x}_i\|^2)$$

If the test sample $\mathbf{y}$ is far from the sample $\mathbf{x}_i$ then the weight of the unknown coefficient a_i tends to 1 so that the program in (4) attempts to get a small a_i. On the other hand, if the test sample is very close to the sample $\mathbf{x}_i$, the constraint on a_i is released.

The detailed procedure for the WRLS graph construction is listed in Algorithm 1. Note that the constructed WRLS graph is a directed graph, i.e., the weight matrix $\mathbf{W}$ is asymmetric.

Data: A given training sample set $\mathbf{X}$
Result: A weight matrix $\mathbf{W}$

Set the diagonal elements of $\mathbf{W}$ to zero ;
for $i = 1, \ldots, N$ **do**
  Pick the sample $\mathbf{x}_i$ from $\mathbf{X}$;
  $\mathbf{X}' = \mathbf{X}$ - $\{\mathbf{x}_i\}$;
  Compute the $(N-1) \times (N-1)$ diagonal matrix $\mathbf{P}$;
  Calculate the $N-1$ vector $\mathbf{a}$ as $\mathbf{a} = (\mathbf{X}'^T \mathbf{X}' + \sigma \mathbf{P})^{-1} \mathbf{X}'^T \mathbf{x}_i$;
  for $j = 1, \ldots, N$ **do**
    if $i < j$ **then**
      Set $W_{ij} = |a_j|$
    else
      Set $W_{ij} = |a_{j-1}|$
    end
  end
end

Algorithm 1: WRLS graph construction.

3.3 Multi-observation recognition based on label propagation

Label propagation is very often linked to the case of semi-supervised learning where the goal is to infer the unknown labels from the known ones using a given criterion [35]. Let C denotes the total number of classes. Let $\mathbf{X}_u$ (a $D \times r$ matrix) denote the r unknown observations (a set of facial actions belonging to the same class/expression). Let $\mathbf{X}_l$ (a $D \times N$ matrix) denote the N known observations (i.e., the set of all training facial action samples). The union of both data sets provides the data matrix $\mathbf{X} = (\mathbf{X}_l, \mathbf{X}_u)$. The corresponding label matrix is denoted by $\mathbf{Y} = (\mathbf{Y}_l, \mathbf{Y}_u)$ (a $C \times (N+r)$ matrix). Each column vector $\mathbf{y}_i$ of $\mathbf{y}$ is a vector characterizing the probabilities of the sample $\mathbf{x}_i$ belonging to

different classes, namely,

$$y_i(c) = p(c|\mathbf{x}_i); c = 1, 2, \ldots, C$$

where $p(c|\mathbf{x}_i)$ is the posterior probability of the class c for the given sample $\mathbf{x}_i$. For a labeled sample $\mathbf{x}_i$, $y_i(c) = 1$ if $\mathbf{x}_i$ belongs to the c^{th} class; $\mathbf{y}_i(c) = 0$, otherwise.

The problem of label propagation is to infer the label matrix $\mathbf{Y}_u$ given the whole data $\mathbf{X} = (\mathbf{X}_l, \mathbf{X}_u)$ and the known label matrix $\mathbf{Y}_l$. This can be achieved by minimizing the following criterion:

$$\min e(\mathbf{Y}) = \sum_{i,j} \|\mathbf{y}_i - \mathbf{y}_j\|^2 W_{ij} \tag{6}$$

An explanation of this objective is as follows. When the samples $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar, namely, the graph weight W_{ij} is large, the distance between $\mathbf{y}_i$ and $\mathbf{y}_j$ should be small in order to minimize the objective, namely the class information of the sample $\mathbf{x}_i$ and $\mathbf{x}_j$ should be similar.

The objective can be further rewritten as

$$\begin{aligned} \min e(\mathbf{Y}) &= \sum_{i,j} \|\mathbf{y}_i - \mathbf{y}_j\|^2 W_{ij} && (7)\\ &= trace(\mathbf{Y}\,\mathbf{D}_{row}\,\mathbf{Y}^T + \mathbf{Y}\,\mathbf{D}_{col}\,\mathbf{Y}^T - \mathbf{Y}\,\mathbf{W}\,\mathbf{Y}^T - \mathbf{Y}\,\mathbf{W}^T\,\mathbf{Y}^T) && (8)\\ &= trace(\mathbf{Y}\,\mathbf{L}\,\mathbf{Y}^T) && (9) \end{aligned}$$

where $\mathbf{D}_{row}$ is a diagonal matrix whose diagonal elements are the row sums of the corresponding rows of $\mathbf{W}$, and $\mathbf{D}_{col}$ is a diagonal matrix whose diagonal elements are the column sums of the corresponding columns of $\mathbf{W}$. $\mathbf{D}_{row} - \mathbf{W}$ and $\mathbf{D}_{col} - \mathbf{W}^T$ are the row and column Graph Laplacian matrices respectively. It is obvious that the matrix $\mathbf{L} = \mathbf{D}_{row} + \mathbf{D}_{col} - (\mathbf{W} + \mathbf{W}^T)$ is symmetric.

Since the r observations have the same unknown label, the unknown label matrix $\mathbf{Y}_u$ will have C configurations $(\mathbf{Y}_u(1), \ldots, \mathbf{Y}_u(C)$ where $\mathbf{Y}_u(c)$ has only the c^{th} row equal to ones and the the rest of the rows are zeros. Therefore, the whole label matrix $\mathbf{Y} = (\mathbf{Y}_l, \mathbf{Y}_u)$ can be written as $\mathbf{Y} = (\mathbf{Y}_l, \mathbf{Y}_u(c))$ where $\mathbf{Y}_l$ is constant. To infer the label of the unknown observations $\mathbf{X}_u$, the following formula can be used:

$$c^\star = \arg\min e(\mathbf{Y}_c) \tag{10}$$

where $\mathbf{Y}(c) = (\mathbf{Y}_l, \mathbf{Y}_u(c))$. Thus, the optimal label is inferred using C evaluations of the term $e(\mathbf{Y}_c)$. The procedure for the multi-observation recognition based on the WRLS graph is illustrated in Algorithm 2.

4 Experimental results

4.1 Data preparation

Our approach has been tested on a subset of the the Cohn-Kanade CMU facial expression dataset, consisting of 250 sequences belonging to 50 persons which are

Data: A set of multiple observations $\mathbf{X}_u$, a training set $\mathbf{X}_l$ and their labels $\mathbf{Y}_l$
Result: The label of the unknown observations c*

Compute the WRLS graph over the data $\mathbf{X} = (\mathbf{X}_l, \mathbf{X}_u)$ (Algo 1) ;
Estimate the label $c^\star$ using Eq. (10)

Algorithm 2: Multi-observation recognition via WRLS graph based label propagation.

displaying 5 expressions: surprise, sadness, joy, anger and disgust. The expression in each frame was coded using the normalized intensities of 6 facial features: inner brow depressor, brow raiser, lower lip depressor, upper lip raiser, lip stretcher and lip corner depressor. For assessing the recognition, we adopted 10 random train/test splits with various sizes. We have successively set the amount of known labels to 20%, 30%, 40% and 50%, respectively of the total data. The test data has been grouped in chunks of size 3, 5, 7 and 9, respectively. The chunk contains frames belonging to the same short video sequence. This last size, 9, represents the whole sequence of the dynamic facial expression, raging from half-apex to apex. By setting the chunk size equal to 1, label propagation corresponds to the particular case of static facial expression.

4.2 Recognition accuracy

The numerical results of the experimental protocol discussed before are presented in the tables below. Table 1 contains the results of our label propagation method. These results are obtained for two values of the regularization parameter σ: $\sigma = 100$ and $\sigma = 1000$ (see eq. 4). The recognition rate is assessed function of the multi-observation chunk size. The best results are marked with bold characters. We could appreciate that the longer the test sequence, a higher recognition rate is achieved. We can also observe that by increasing the number of tested frames, the recognition of the whole chunk is not increasing significantly. This is due to the fact that the multiple observations used in our case are having different intensity levels. So whenever low or moderate intensity frames are included in the test, those tend to confuse the global decision of the label propagation since at half apex intensities it is hard to correctly discriminate facial expression. This phenomenon is illustrated by the last column of the Table where all frames are included in the test. Ideally, a trade-off is reached by using the few last frames.

Furthermore, in order to show the superiority of our method we have compared it with two other methods for graph construction: the one based on the K-nearest neighbors (K-NN), which is used by the graph Laplacian [16] (see Table 2) and the one based on the Locally Linear Embedding (LLE) [33] (see Table 3). LLE graph can be obtained by applying a two-stage procedure: (i) adjacency matrix computation followed by (ii) a linear reconstruction of samples from their neighbors. The adjacency matrix can be computed using K-NN or ϵ-Neighborhood method. The non-zero entries of the weight matrix $\mathbf{W}$ are esti-

mated by reconstructing the sample from its neighboring points and minimizing the ℓ_2 reconstruction error defined as:

$$\sum_{i=1}^{N} \|\mathbf{x}_i - \sum_j W_{ij}\,\mathbf{x}_j\|^2 \quad s.t. \sum_{j=1}^{N} W_{ij} = 1. \tag{11}$$

where $W_{ij} = 0$ if $\mathbf{x}_i$ and $\mathbf{x}_j$ are not neighbors.

Due to space limitation, we have shown only the best results which were obtained. For instance in the case of Table 2 these were obtained for a value of K-NN=7 and K-NN=11. In the case of Table 3 these were obtained for K-NN=5.

Mode	Number of frames			
	r=3	r=5	r=7	r=9
$\sigma = 100$				
$20\% - 80\%$	**76.40%**	76.00%	**76.40%**	74.40%
$30\% - 70\%$	84.00%	**84.50%**	84.00%	83.50%
$40\% - 60\%$	**91.33%**	88.00%	90.00%	86.00%
$50\% - 50\%$	92.00%	**93.00%**	**93.00%**	89.00%
$\sigma = 1000$				
$20\% - 80\%$	78.80%	78.40%	**79.20%**	77.20%
$30\% - 70\%$	83.00%	83.50%	83.50%	**85.00%**
$40\% - 60\%$	**90.00%**	**90.00%**	**90.00%**	88.66%
$50\% - 50\%$	93.00%	**94.00%**	**94.00%**	92.00%

Table 1. Average recognition rate based on the WRLS graph construction algorithm.

Mode	Number of frames			
	r=3	r=5	r=7	r=9
$K - NN = 7$				
$20\% - 80\%$	74.40%	**78.40%**	70.00%	67.66%
$30\% - 70\%$	79.00%	**84.00%**	69.50%	69.00%
$40\% - 60\%$	82.66%	**85.33%**	74.66%	70.66%
$50\% - 50\%$	83.00%	**86.00%**	76.00%	71.00%
$K - NN = 11$				
$20\% - 80\%$	74.40%	**78.40%**	77.20%	72.40%
$30\% - 70\%$	79.00%	**84.00%**	81.50%	81.00%
$40\% - 60\%$	82.66%	85.33%	**86.00%**	82.00%
$50\% - 50\%$	83.00%	**87.00%**	**87.00%**	84.00%

Table 2. Average recognition rate based on the K-NN graph construction algorithm.

Mode	Number of frames			
	r=3	r=5	r=7	r=9
20% − 80%	**78.80%**	77.66%	65.66%	63.20%
30% − 70%	**77.00%**	76.00%	61.50%	60.00%
40% − 60%	**82.66%**	76.00%	66.66%	66.00%
50% − 50%	**83.00%**	77.00%	70.00%	67.00%

Table 3. Average recognition rate based on the LLE graph construction algorithm for K-NN=5.

From these results we can have the following observations: (i) the performance of label propagation with the proposed graph construction outperforms that obtained by the competing methods. K-NN graph method is better than LLE method. This can be explained by the fact that LLE method works with a classic coding scheme on low dimension vectors; (ii) we can observe that, for our proposed graph construction method, a high value for the balance parameter is preferable. This means that a weighted regularization is indeed required in order to get informative graphs for this kind of data.

5 Application to social robotics

In this subsection, we describe a human-robot interaction application based on our proposed approach. The application refers to mimicking the facial expressions of a person perceived by a robot's camera.

Without any loss of generality, we used an AIBO robot for our application. The input to the system is a video stream capturing the user's face. AIBO's human-like communication system is implemented through a series of *instincts* and *senses*: affection, movement, touch, hearing, sight and balance. AIBO is able to show its emotions through an array of LEDs situated in the frontal part of the head. These are depicted in figure 3, and are shown in correspondence with the six universal expressions. Notice that the blue lights that appear, in certain images, on each part of the head, are blinking LEDs whose meaning is to inform that the robot is remotely controlled[1]. This is a built-in feature and can not be turned off.

In addition to the LEDs' configuration, the robot response contains some small head and body motion. From its concept design, AIBO's affective states are triggered by the Emotion Generator engine. This occurs as a response to its internal state representation, captured through multi-modal interaction (vision, audio and touch). For instance, it can display the 'happiness' feeling when it detects a face (through the vision system) or it hears a voice. But it does

[1] The application described in this paper, was built using the Remote Framework (RFW) programming environment (based on C++ libraries), which works on a client-server architecture over a wireless connection between a PC and the AIBO

Fig. 3. The figure illustrates the LEDs configuration for each universal expression.

not possess a built-in system for vision-based automatic facial-expression recognition. For this reason, the application we created for AIBO could be seen as an extension of its pre-defined behaviors. This application is a very simple one, in which the robot is just imitating the expression of a human subject. In other words, we wanted to see its reaction according to the emotional state displayed by a person. Usually, the response of the robot occurs slightly after the apex of the human expression. The results of this application were recorded in a 2 minutes video which can be downloaded from the following address: http://www.cvc.uab.es/∼bogdan/AIBO-emotions.avi. In order to be able to display simultaneously in the video the correspondence between person's and robot's expressions, we put them side by side. In this case only, we analyzed offline the content of the video and commands with the facial expression code were sent to the robot. Figure 4 illustrates nine recognized facial expressions from a 1600 frame-long video sequence.

6 Conclusions

In this paper, we proposed a novel approach for facial expression recognition, which exploits an efficient and adaptive graph-based label propagation (semi-supervised mode) in a multi-observation framework. The facial features are extracted using an appearance-based 3D face tracker, view- and texture independent. The presented approach has been tested on the CMU facial expression dataset. Furthermore, we developed a real application, where an AIBO robot is mirroring the facial expression perceived. Future work will be devoted to apply the same framework to other related applications, such as gesture and activity recognition.

ACKNOWLEDGMENTS

This work is supported in part by the Spanish Government under the project TIN2010-18856.

References

1. Tian, Y., Kanade, T., Cohn, J.: Recognizing action units for facial expression analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence **23** (2001) 97–115
2. Bartlett, M., Littlewort, G., Lainscsek, C., Fasel, I., Movellan, J.: Machine learning methods for fully automatic recognition of facial expressions and facial actions. In: Proc. of IEEE. Int'. Conf. on SMC. Volume I., The Hague, The Netherlands (2004) 592–597
3. Sung, J., Lee, S., Kim, D.: A real-time facial expression recognition using the staam. In: Proc. of Int'l. Conf. on Pattern Recognition. Volume I., Hong Kong, PR China (2006) 275–278
4. Xiang, T., Leung, M., Cho, S.: Expression recognition using fuzzy spatio-temporal modeling. Pattern Recognition **41** (2008) 204–216
5. Ambadar, Z., Schooler, J., Cohn, J.: Deciphering the enigmatic face: the importance of facial dynamics to interpreting subtle facial expressions. Psychological Science **16** (2005) 403–410
6. Yang, P., Liu, Q., Cui, X., Metaxas, D.: Facial expression recognition using encoded dynamic features. In: Computer Vision and Pattern Recognition. (2008)
7. Cheon, Y., Kim, D.: Natural facial expression recognition using differential-aam and manifold learning. Pattern Recognition **42** (2009) 1340–1350
8. Zhang, Y., Ji, Q.: Active and dynamic information fusion for facial expression understanding from image sequences. **27** (2005) 699–714
9. Yeasin, M., Bullot, B., Sharma, R.: Recognition of facial expressions and measurement of levels of interest from video. IEEE Trans. on Multimedia **8** (2006) 500–508
10. Shan, C., Gong, S., McOwan, P.: Dynamic facial expression recognition using a bayesian temporal manifold model. In: Proc. of British Machine Vision Conference. Volume I., Edinburgh, UK (2006) 297–306
11. Shan, C., Gong, S., McOwan, P.: Facial expression recognition based on local binary patterns: A comprehensive study. Image and Vision Computing **27** (2009) 803–816
12. Zhao, G., Pietikainen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans. on Patt. Anal. and Machine Intell. **29** (2007) 915–928
13. Liu, X., Chen, T.: Video-based face recognition using adaptive hidden markov models. In: Proc. of IEEE Conf. on CVPR. (2003) 340–345
14. Zhou, D., Bousquet, O., Lal, T., Weston, J., Scholkopf, B.: Learning with local and global consistency. In: Proc. of NIPS. (2003)
15. Kokiopoulou, E., Frossard, P.: Graph-based classification of multiple observation sets. Pattern Recognition **43** (2010) 3988–3997
16. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation **15** (2003) 1373–1396
17. Kanade, T., Cohn, J., Tian, Y.: Comprehensive database for facial expression analysis. In: Proc. IEEE Intl. Conf. on Automatic Face and Gesture Recognition, Grenoble, France (2000) 46–53
18. Cañamero, L., Gaussier, P.: Emotion understanding: robots as tools and models. In Nadel, J.e.a., ed.: Emotional Development: Recent Research Advances. Oxford University Press, New York (2005) 235–258

19. Pantic, M.: Affective computing. In Pagani, M.e.a., ed.: Encyclopedia of Multimedia Technology and Networking. Volume I. Idea Group Publishing (2005) 8–14
20. Picard, R., Vyzas, E., Healy, J.: Toward machine emotional intelligence: analysis of affective physiological state. IEEE Trans. on Patt. Anal. and Machine Intell. **23** (2001) 1175–1191
21. Breazeal, C.: Robot in society: friend or appliance? In: Proc. of Wksp on Emotion-Based Agent Architectures. (1999) N/A
22. Breazeal, C.: Sociable machines: Expressive social exchange between humans and robots. PhD thesis, MIT, Cambridge, US (2000)
23. Ahlberg, J.: Model-based coding: extraction, coding and evaluation of face model parameters. Ph.D. Thesis, Dept. of Elec. Eng., Linköping Univ., Sweden (2002)
24. Dornaika, F., Davoine, F.: On appearance based face and facial action tracking. IEEE Trans. on Circuits and Systems for Video Technology **16** (2006) 1107–1124
25. Müller, M.: Information Retrieval for Music and Motion. Springer Berlin Heidelberg (2007)
26. Niyogi, P.: Manifold regularization and semi-supervised learning: Some theoretical analyses. Technical report, Computer Science Department, University of Chicago (2008)
27. von Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing **17** (2007) 395–416
28. Singh, A., Nowak, R., Zhu, X.: Unlabeled data: Now it helps, now it doesnt. In: Advances in Neural Information Processing Systems. (2009)
29. Wang, Z., Song, Y., Zhang, C.: Knowledge transfer on hybrid graph. In: IJCAI. (2009)
30. Jebara, T., Wang, J., Chang, S.: Graph construction and b-matching for semi-supervised learning. In: ICML. (2009)
31. Daitch, S., Kelner, J., Spielman, D.: Fitting a graph to vector data. In: International Conference on Machine Learning. (2009)
32. Wang, F., Zhang, C.: Label propagation through linear neighborhoods. IEEE Transactions on Knowledge and Data Engineering **20** (2008) 55–67
33. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. Science **290** (2000) 2323–2326
34. Waqas, J., Yi, Z., Zhang, L.: Collaborative neighbor representation based classification using l2-minimization approach. Pattern Recognition Letters **34** (2013) 201–208
35. Zhou, S., Chellappa, R., Mogghaddam, B.: Visual tracking and recognition using appearance-adaptive models in particle filters. IEEE Transactions on Image Processing **13** (2004) 1473–1490

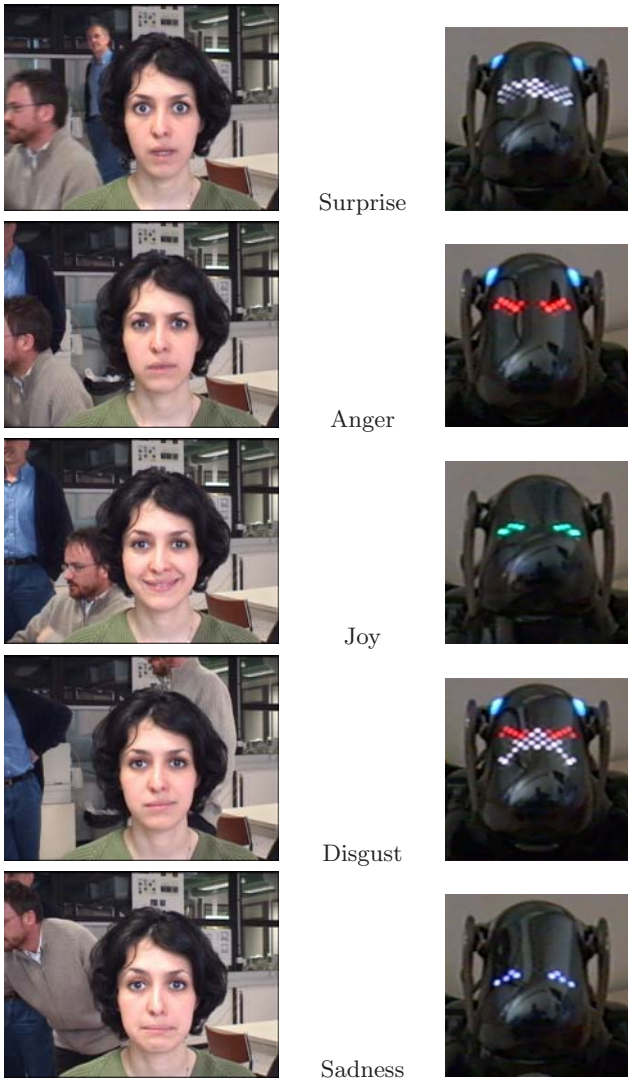

Fig. 4. Person's facial expressions are shown in correspondence with the robot's response.