FACIAL EXPRESSION RECOGNITION IN IMAGE SEQUENCES USING A NOVEL CLASSIFIER

Irene Kotsia[†]*, Nikolaos Nikolaidis*[†] *and Ioannis Pitas*[†]

[†]Aristotle University of Thessaloniki
Department of Informatics
Box 451, 54124 Thessaloniki, Greece

ABSTRACT

In this paper, a novel class of Support Vector Machines (SVM) is introduced to deal with facial expression recognition. The system developed performs facial expression recognition in facial videos. The grid tracking and deformation system used, based on deformable models, tracks the grid in consecutive video frames over time, as the facial expression evolves, till the frame corresponding to the greatest facial expression intensity. The geometrical displacement of Candide nodes, defined as the difference of the node coordinates on the first and the greatest facial expression intensity frame, is used as an input to a bank of novel SVM classifiers, that are used to recognize the six basic facial expressions. The results on the Cohn-Kanade database show a recognition accuracy of 98.2%.

Index Terms— Facial expression recognition, Facial Action Coding System, Support Vector Machines, Candide grid.

1. INTRODUCTION

Facial expression recognition has attracted a great interest during the past two decades, due to its importance for human centered interfaces. A set of six basic facial expressions (anger, disgust, fear, happiness, sadness and surprise) were defined as the basic ones [1]. A set of muscle movements (known as Action Units) that produce each facial expression when combined following specific rules [2], was created by psychologists, thus forming the so called $Facial\quad Action\quad Coding\quad System\quad (FACS)$ [3]. A survey on automatic facial expression recognition can be found in [2].

In the current paper, a method for recognizing facial expressions using a novel class of SVM, is proposed. The geometrical displacement of the Candide grid points through time, defined as the difference of each point's coordinates between the first and the last frame of the image sequence, are used as an input to a novel multi-class SVM system, where each classification class represents one of the 6 basic facial expressions. The experiments were performed using the Cohn-Kanade database and the results show that the above mentioned system can achieve an accuracy of 98.2% when recognizing the six basic facial expressions.

This work was supported by the "SIMILAR" European Network of Excellence on Multimodal Interfaces of the IST Programme of the European Union (www.similar.cc).

2. SYSTEM DESCRIPTION

The diagram of the system used for the experiments is shown in Figure 1.

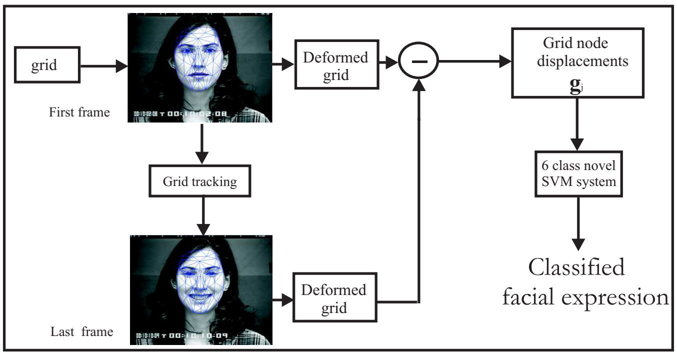

Fig. 1. System architecture for facial expression recognition in facial videos.

2.1. Geometrical displacement information extraction

The geometrical information extraction is performed by a grid adaptation system, based on deformable models. The Candide grid is randomly initialized on the first frame of the image sequence, being in its neutral state. The points around the eyes, eyebrows and mouth are the ones with the greatest importance, since they are the ones responsible for the formation of movement according to FACS. The software automatically adjusts the grid to the face and then tracks it through the image sequence, following the facial expression evolving through time [4]. At the end, the grid adaptation software produces the deformed Candide grid that corresponds to the facial expression appearing at the image sequence.

Let $\mathcal{U}$ be the database that contains the geometrical displacement information separated into the 6 different classes, $\mathcal{U}_k(k \in \{1,\dots,6\})$, each one representing one of six basic facial expressions . The geometrical information used is the displacement of one point $\mathbf{d}_j^i$, defined as the difference between the last and the first frame's coordinates:

$$\mathbf{d}_j^i = [\Delta x_j^i \quad \Delta y_j^i]^T, \quad i \in \{1,\dots,K\} \quad \text{and} \quad j \in \{1,\dots,N\} \tag{1}$$

where i is the number of points taken under consideration, here K, equal to 104 (Candide grid's nodes) and j is the the number of image sequences to be examined, here equal to N.

In that way, for every image sequence to be examined, a feature vector $\mathbf{g}_j$ that belongs to one of the six facial expression classes $\mathcal{U}_k$ is constructed:

$$\mathbf{g}_j = [\mathbf{d}_j^1 \quad \mathbf{d}_j^2 \quad \dots \quad \mathbf{d}_j^K]^T. \tag{2}$$

where the vector $\mathbf{g}_j$ has $F = 62 \cdot 2 = 124$ dimensions.

2.2. Geometrical displacement information classification

The feature vector $\mathbf{g}_j \in \Re^F$ is used as an input to a multi-class SVM that classifies each set of the grid's geometrical displacements to one of the six basic facial expressions. More specifically, as an input for the SVM system, the feature vector $\mathbf{g}_j$ is used, labelled properly with the true corresponding facial expression. The output of the SVM system is a label that classifies the grid under examination to one of the six basic facial expressions. In the following, the two-class and multi-class SVM-based classification problem used for facial expression recognition are described.

2.3. Classical Two Class SVM

In order to train a two-class SVM network, the following minimization problem has to be solved [5]:

$$\min_{\mathbf{w}_k, b_k, \boldsymbol{\xi}^k} \quad \frac{1}{2}\mathbf{w}_k^T\mathbf{w}_k + C_k \sum_{j=1}^{N} \xi_j^k \tag{3}$$

subject to the separability constraints:

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

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

After solving the optimization problem (3) subject to the separability constraints (4) ([6]), the decision function is:

$$f_k(\mathbf{g}) = \text{sign}(\mathbf{w}_k^T \phi(\mathbf{g}) + b_k) \tag{4}$$

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

$$h(\mathbf{g}_i, \mathbf{g}_j) = \phi(\mathbf{g}_i)^T \phi(\mathbf{g}_j). \tag{5}$$

2.4. A Modified Two Class SVM

The modified classifier proposed here, is based on a modified two class SVM formulation proposed in [8].

2.4.1. The Linear Case

In order to form the optimization problem of the SVM proposed in [8], the within class scatter matrix of the training set should be defined:

$$\mathbf{S}_w^k = \sum_{\mathbf{g}_i \in \mathcal{U}_k^1} (\mathbf{g}_i - \boldsymbol{\mu}_k^1)(\mathbf{g}_i - \boldsymbol{\mu}_k^1)^T + \sum_{\mathbf{g}_i \in \mathcal{U}_k^2} (\mathbf{g}_i - \boldsymbol{\mu}_k^2)(\mathbf{g}_i - \boldsymbol{\mu}_k^2)^T \tag{6}$$

where $\boldsymbol{\mu}_k^1$ and $\boldsymbol{\mu}_k^2$ are the mean vectors of the classes $\mathcal{U}_k^1$ and $\mathcal{U}_k^2$, respectively. The optimization problem of the modified SVM is [8]:

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

subject to the separability constraints (4) (in the linear case considered here $\phi(\mathbf{g}) = \mathbf{g}$). The solution of the optimization problem (7) subject to the constraints (4) is given by the saddle point of the Lagrangian:

$$L(\mathbf{w}_k, b_k, \boldsymbol{\alpha}^k, \boldsymbol{\beta}^k, \boldsymbol{\xi}^k) = \mathbf{w}_k^T \mathbf{S}_w^k \mathbf{w}_k + C_k \sum_{i=1}^{N} \xi_i^k - \\ - \sum_{i=1}^{N} a_i^k [y_i^k (\mathbf{w}_k^T \mathbf{g}_i - b_k) - 1 + \xi_i^k] - \sum_{i=1}^{N} \beta_i^k \xi_i^k \tag{8}$$

where $\boldsymbol{\alpha}^k = [\alpha_1^k,\dots,\alpha_N^k]$ and $\boldsymbol{\beta}^k = [\beta_1^k,\dots,\beta_N^k]$ are the vectors of Lagrangian multipliers for the constraints (4).

The linear decision function is:

$$f_k(\mathbf{g}) = \text{sign}(\mathbf{w}_k^T \mathbf{g} + b_k) = \text{sign}(\frac{1}{2} \sum_{j=1}^{N} y_i^k a_i^k \mathbf{g}_j^T {\mathbf{S}_w^k}^{-1} \mathbf{g} + b_k). \tag{9}$$

2.4.2. The Non-Linear Case

The nonlinear multi-class decision surfaces can be created in the same manner as the one proposed in [8]. By applying the nonlinear function ϕ to the vectors ${\mathbf{S}_w^k}^{-\frac{1}{2}} \mathbf{g}_i$, it is derived that $h(\mathbf{S}_w^{k\,-\frac{1}{2}}\mathbf{g}_i, \mathbf{S}_w^{k\,-\frac{1}{2}}\mathbf{g}_j) = \phi(\mathbf{S}_w^{k\,-\frac{1}{2}}\mathbf{g}_i)^T\phi(\mathbf{S}_w^{k\,-\frac{1}{2}}\mathbf{g}_j)$ [8]. Then, kernel functions are applied in (10) as:

$$W(\boldsymbol{\alpha}_k) = \sum_{i}^{N}\alpha_i^k - \frac{1}{4}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i^k\alpha_j^k y_i^k y_j^k h(\mathbf{S}_w^{k\,-\frac{1}{2}}\mathbf{g}_i, \mathbf{S}_w^{k\,-\frac{1}{2}}\mathbf{g}_j). \tag{10}$$

The corresponding non-linear decision function is given by:

$$f_k(\mathbf{g}) = \frac{1}{2}\sum_{j=1}^{N} y_i^k\alpha_i^k h(\mathbf{S}_w^{k\,-\frac{1}{2}}\mathbf{g}_j, \mathbf{S}_w^{k\,-\frac{1}{2}}\mathbf{g}) + b_k. \tag{11}$$

2.5. Classical Multi-class SVM

The multi-class SVM problem solves only one optimization problem [9]. It constructs 6 facial expressions rules, where the $k-$th function $\mathbf{w}_k^T\phi(\mathbf{g}_j) + b_k$ separates training vectors of the class k from the rest of the vectors, by minimizing the objective function:

$$\min_{\mathbf{w},\mathbf{b},\boldsymbol{\xi}} \quad \frac{1}{2}\sum_{k=1}^{6}\mathbf{w}_k^T\mathbf{w}_k + C\sum_{j=1}^{N}\sum_{k\neq l_j}\xi_j^k \tag{12}$$

subject to the constraints:

$$\mathbf{w}_{l_j}^T\phi(\mathbf{g}_j) + b_{l_j} \geq \mathbf{w}_k^T\phi(\mathbf{g}_j) + b_k + 2 - \xi_j^k \tag{13}$$
$$\xi_j^k \geq 0, \quad j = 1,\ldots,N, \quad k \in \{1,\ldots,6\}\backslash l_j$$

where ϕ is the function that maps the deformation vectors to a higher dimensional space, where the data are supposed to be linearly or near linearly separable. C is the term that penalizes the training errors. The vector $\mathbf{b} = [b_1\ldots b_6]^T$ is the bias vector and $\boldsymbol{\xi} = [\xi_1^1,\ldots,\xi_i^k,\ldots,\xi_N^6]^T$ is the slack variable vector. Then, the decision function is:

$$h(\mathbf{g}) = \operatorname*{argmax}_{k=1,\ldots,6}(\mathbf{w}_k^T\phi(\mathbf{g}) + b_k). \tag{14}$$

2.6. A Modified Class of Multi-class SVM

2.6.1. The Linear Case

Let that the within class scatter matrix of the grid deformation feature vectors $\mathbf{g}_i$ is defined as:

$$\mathbf{S}_w = \sum_{k=1}^{M}\sum_{\mathbf{g}_i\in\mathcal{U}_k}(\mathbf{g}_i - \boldsymbol{\mu}_k)(\mathbf{g}_i - \boldsymbol{\mu}_k)^T \tag{15}$$

where M is the number of facial expression classes (here equal to six), $\boldsymbol{\mu}_k$ is the geometrical displacement vector for the class k. In this Section, the within class scatter matrix $\mathbf{S}_w$ is assumed to be invertible, which holds in the under examination case, since classically for the deformation, the feature vector dimension is smaller than the available training examples.

The modified constraint optimization problem is:

$$\min_{\mathbf{w}_k,\mathbf{b},\boldsymbol{\xi}} \quad \sum_{k=1}^{6}\mathbf{w}_k^T\mathbf{S}_w\mathbf{w}_k + C\sum_{j=1}^{N}\sum_{k\neq l_j}\xi_j^k \tag{16}$$

subject to the separability constraints in (13) (in the linear case considered here $\phi(\mathbf{g}) = \mathbf{g}$). The solution of the above constraint optimization problem can be given by finding the saddle point of the Lagrangian:

$$L(\mathbf{w},\mathbf{b},\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\beta}) = \sum_{k=1}^{6}\mathbf{w}_k^T\mathbf{S}_w\mathbf{w}_k + +C\sum_{i=1}^{N}\sum_{k=1}^{6}\xi_i^k -$$
$$-\sum_{i=1}^{N}\sum_{k=1}^{6}\alpha_i^k[(\mathbf{w}_{l_i} - \mathbf{w}_k)^T\mathbf{g}_i + b_{l_i} - b_k - 2 + \xi_i^k]$$
$$-\sum_{i=1}^{N}\sum_{k=1}^{6}\beta_i^k\xi_i^k \tag{17}$$

where $\boldsymbol{\alpha} = [\alpha_1^1,\ldots,\alpha_i^k,\ldots,\alpha_N^6]$ and $\boldsymbol{\beta} = [\beta_1^1,\ldots,\beta_i^k,\ldots,\beta_N^6]$ are the Lagrangian multipliers for the constraints (13) with :

$$\alpha_i^{l_i} = 0,\ \xi_i^{l_i} = 2,\ \beta_i^{l_i} = 0,\ i = 1,\ldots,N \tag{18}$$

and constraints:

$$\alpha_i^k \geq 0,\ \beta_i^k \geq 0,\ i = 1,\ldots,l,\ k \in \{1,\ldots,6\}\backslash l_i. \tag{19}$$

After a series of manipulations, the search of the saddle point of the Lagrangian (17) is reformulated to the maximization of the Wolf dual problem:

$$W(\boldsymbol{\alpha}) = 2\sum_{i=1}^{N}\sum_{k=1}^{6}\alpha_i^k + \frac{1}{4}\sum_{i,j,k}(-\frac{1}{2}c_j^{l_j}A_iA_j + \alpha_i^k\alpha_i^{l_i} - \frac{1}{2}\alpha_i^k\alpha_j^k)\mathbf{g}_i\mathbf{S}_w^{-1}\mathbf{g}_j \tag{20}$$

which is a quadratic function in terms of $\boldsymbol{\alpha}$ with the linear constraints:

$$\sum_{i=1}^{N}a_i^k = \sum_{i=1}^{N}c_i^kA_i,\ k = 1,\ldots,6. \tag{21}$$

The above optimization problem can be solved using optimization packages like [9]. The corresponding decision hyperplane is:

$$f(\mathbf{g}) = \operatorname{argmax}_{k=1,\ldots,6}(\mathbf{w}_k^T\mathbf{g} + b_k) =$$
$$= \operatorname{argmax}_{k=1,\ldots,6}[\frac{1}{2}\sum_{i=1}^{N}(c_i^kA_i - \alpha_i^k)\mathbf{g}_i^T\mathbf{S}_w^{-1}\mathbf{g} + b_k]. \tag{22}$$

2.6.2. The Non-Linear Case

The nonlinear multi-class decision surfaces can be created in the same manner as the two class non-linear decision surfaces that have been proposed in [8]. The fact that the term $\mathbf{g}_i^T\mathbf{S}_w^{-1}\mathbf{g}_j$ can be written in terms of dot products as $(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{g}_i)^T(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{g}_j)$, is exploited. Then, kernels can be applied in (20) as:

$$W(\boldsymbol{\alpha}) = 2\sum_{i=1}^{N}\sum_{k=1}^{6}\alpha_i^k + \frac{1}{4}\sum_{i,j,k}(-\frac{1}{2}c_j^{l_j}A_iA_j + \alpha_i^k\alpha_i^{l_i} - \frac{1}{2}\alpha_i^k\alpha_j^k)h(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{g}_i, \mathbf{S}_w^{-\frac{1}{2}}\mathbf{g}_j), \tag{23}$$

the corresponding decision surface is:

$$f(\mathbf{g}) = \underset{k=1,\dots,6}{\operatorname{argmax}} \frac{1}{2}[\sum_{i=1}^{N}(c_i^k A_i - \alpha_i^n)h(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{g}_i, \mathbf{S}_w^{-\frac{1}{2}}\mathbf{g}) + b_k]. \quad (24)$$

3. EXPERIMENTAL RESULTS

The Cohn-Kanade database [3] was used for the experiments regarding facial expression recognition in 6 basic facial expressions classes. These combinations of FAUs this database is annotated with, were translated into facial expressions according to [2], in order to define the corresponding ground truth for the facial expressions. All the subjects were taken under consideration to form a database of over 400 image sequences.

The following procedure was followed for the experiments. All image sequences contained in the database were divided into 6 classes, each one corresponding to one of the 6 basic facial expressions to be recognized. Neutral state is not considered as a class, as the system tries to recognize the fully expressed facial expression starting from the neutral state. Five sets containing 20% of the data for each class, chosen randomly, were created. One set containing 20% of the samples for each class is used for the test set, while the remaining sets form the training set. The average classification accuracy is the mean value of the percentages of the correctly classified facial expressions.

When the classical six class SVM was applied to the Candide grid, the facial expression recognition accuracy achieved was equal to 91.4%. The equivalent facial expression recognition accuracy when the modified six class SVM was used, was equal to 98.2%.

The confusion matrix calculated from the experiments, shown in Table 1, presents the results obtained while applying the modified six class SVM to the Candide grid (the abbreviations an, di, fe, ha, sa and su represent anger, disgust, fear, happiness, sadness and surprise, respectively). As can be seen, the most ambiguous facial expression is anger, since it was the only one being misclassified as another facial expression (sadness and disgust by 14% and 5%, respectively).

Table 1. Confusion matrix for facial expression recognition using the modified SVM to the Candide grid

$lab_{ac} \backslash lab_{cl}$	an	di	fe	ha	sa	su
an	122	0	0	0	0	0
di	7	135	0	0	0	0
fe	0	0	345	0	0	0
ha	0	0	0	375	0	0
sa	21	0	0	0	270	0
su	0	0	0	0	0	270

4. CONCLUSION

Facial expression recognition using SVM has been investigated in this paper. The grid adaptation system, based on deformable models, tracks the grid, as the facial expression progresses through the time, thus producing a deformed grid that corresponds to the highest intensity of the facial expression under examination. The nodes geometrical displacement information, defined as the difference between the first and the last frame, is used as an input to a novel six class (one for each facial expression) SVM. The output of the SVM is the facial expression recognized from the image sequence. The above mentioned system achieves a recognition rate of 98.2%.

5. REFERENCES

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

[2] M. Pantic and L. J. M. Rothkrantz, “Expert system for automatic analysis of facial expressions,” *Image and Vision Computing*, vol. 18, no. 11, pp. 881–905, August 2000.

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

[4] I. Kotsia, and I. Pitas, “Facial Expression Recognition in Image Sequences using Geometric Deformation Features and Support Vector Machines,” *IEEE Transactions on Image Processing*, 2006.

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

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

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

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

[9] J. Weston and C. Watkins, “Multi-class Support Vector Machines,” Tech. Rep.Techical report CSD-TR-98-04, 2004.