

FACIAL EXPRESSION RECOGNITION USING ACTIVE APPEARANCE MODELS

Pedro Martins, Joana Sampaio, Jorge Batista *

*ISR-Institute of Systems and Robotics, Dep. of Electrical Engineering and Computers
FCTUC-University of Coimbra, Coimbra, Portugal*

Keywords: Active appearance models (AAM), Linear discriminant analysis (LDA), Facial expression recognition.

Abstract: A framework for automatic facial expression recognition combining Active Appearance Model (AAM) and Linear Discriminant Analysis (LDA) is proposed. Seven different expressions of several subjects, representing the neutral face and the facial emotions of happiness, sadness, surprise, anger, fear and disgust were analysed. The proposed solution starts by describing the human face by an AAM model, projecting the appearance results to a Fisherspace using LDA to emphasize the different expression categories. Finally the performed classification is based on malahanobis distance.

1 INTRODUCTION

Facial expressions recognition plays an important role in human communication since has more influence than simpler audio information. Psychology studies (T. Dalglish, 1999) describe that there are six basic emotions universally recognized: joy, sadness, surprise, fear, anger and disgust. Notice that, these expressions are also compatible with MPEG-4 norm.

Clearly, in order to extract facial information from one image we need to solve first the problem of finding the face on it. Traditionally, there are two different ways to approach this problem. Anthropometric feature extraction (Batista, 2007) based on detection of facial characteristics such as eyes, eyebrows, mouth and nose, or using appearance based methods. The last one, is preferable since it is able to extract relevant face information without background interference and describes facial characteristics in a reduced model. Our work on facial expressions recognition belongs to the appearance based approaches.

Appearance-based face recognition involves image preprocessing and the use of statistical redundancy reduction for compact coding. To synthesize a complete image face, both shape and texture are modelled. The AAM represents both shape and texture variations observed in a training image set and the correlations between them. Supervised dimension reduc-

tion use the knowledge of class structure and the use of multi-linear models allow low-dimensional representations which account for variations in geometry, orientation and illumination.

In statistical supervised learning, Linear Discriminant Analysis (Peter N. Belhumeur and Kriegman, 1997) is a classical solution that finds the basis vectors maximizing the interclass distances while minimizing the intraclass distances. Linear discriminant analysis is performed in order to extract the most discriminant features which maximizes class separability. The discriminating power of the appearance parameters (AAM) is analysed when projected in Fisherspace and the degree of similarity is measured in that subspace using Mahalanobis distances. A series of experiments were conducted on a set of unknown images showing the faces of different subjects with facial expressions ranging neutral to intensely expressive.

This paper is organised as follows: section 2 gives a brief introduction to the standard Active Appearance Models (AAM) theory, section 3 describes the facial recognition methodology used, section 4 shows experimental results and section 5 discusses the results.

*This work was funded by FCT Project POSI/EEA-SRI/61150/2004

2 ACTIVE APPEARANCE MODELS

AAM is a statistical based segmentation method, where the variability of shape and texture is captured from the dataset. Building such a model allows the generation of new instances with photorealistic quality. In the search phase the model is adjusted to the target image by minimizing the texture residual. For further details refer to (T.F.Cootes and C.J.Taylor, 2001).

2.1 Shape Model

The shape is defined as the quality of a configuration of points which is invariant under Euclidian Similarity transformations (T.F.Cootes and C.J.Taylor, 2004). This landmark points are selected to match borders, vertices, profile points, corners or other features that describe the shape.

The representation used for a single n -point shape is a $2n$ vector given by:

$$\mathbf{x} = (x_1, y_1, x_2, y_2, \dots, x_{n-1}, y_{n-1}, x_n, y_n)^T \quad (1)$$

With N shape annotations, follows a statistical analysis where the shapes are previously aligned to a common mean shape using a Generalised Procrustes Analysis (GPA) removing location, scale and rotation effects. Optionally, we could project the shape distribution into the tangent plane, but omitting this projection leads to very small changes (Stegmann and Gomez, 2002).

Applying a Principal Components Analysis (PCA), we can model the statistical variation with:

$$\mathbf{x} = \bar{\mathbf{x}} + \Phi_s \mathbf{b}_s \quad (2)$$

where new shapes \mathbf{x} , are synthesised by deforming the mean shape, $\bar{\mathbf{x}}$, using a weighted linear combination of eigenvectors of the covariance matrix, Φ_s . \mathbf{b}_s is a vector of shape parameters which represents the weights. Φ_s holds the t_s most important eigenvectors that explain a user defined variance.

2.2 Texture Model

For m pixels sampled, the texture is represented by the vector:

$$\mathbf{g} = [g_1, g_2, \dots, g_{m-1}, g_m]^T \quad (3)$$

Building a statistical texture model, requires warping each training image so that the control points match those of the mean shape. In order to prevent

holes, the texture mapping is performed using the reverse map with bilinear interpolation correction.

The texture mapping is performed, using a piecewise affine warp, i.e. partitioning the convex hull of the mean shape by a set of triangles using the Delaunay triangulation. Each pixel inside a triangle is mapped into the correspondent triangle in the mean shape using barycentric coordinates, see figure 1.

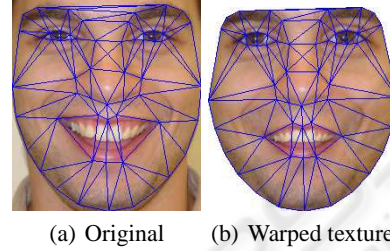


Figure 1: Texture mapping example.

This procedure removes differences in texture due shape changes, establishing a common texture reference frame.

To reduce the influence of global lighting variation a scaling, α and offset, β is applied

$$\mathbf{g}_{norm} = (\mathbf{g}_i - \beta \cdot \mathbf{1}) / \alpha \quad (4)$$

After the normalization we get $\mathbf{g}_{norm}^T \cdot \mathbf{1} = 0$ and $|\mathbf{g}_{norm}| = 1$.

A texture model can be obtained by applying a PCA on the normalized textures:

$$\mathbf{g} = \bar{\mathbf{g}} + \Phi_g \mathbf{b}_g \quad (5)$$

where \mathbf{g} is the synthesized texture, $\bar{\mathbf{g}}$ is the mean texture, Φ_g contains the t_g highest covariance texture eigenvectors and \mathbf{b}_g is a vector of texture parameters.

Another possible solution to reduce the effects of differences in illumination is to perform a histogram equalization independently in each of the three color channels (G. Finlayson and Tian, 2005).

Similarly to shape analysis, PCA is conducted in texture data to reduce dimensionality and data redundancy. Since the number of dimensions is greater than the number of samples ($m \gg N$) it is used a low-memory PCA.

2.3 Combined Model

The shape and texture from any training example is described by the parameters \mathbf{b}_s and \mathbf{b}_g . To remove correlations between shape and texture model parameters a third PCA is performed to the following data:

$$\mathbf{b} = \begin{pmatrix} \mathbf{W}_s \mathbf{b}_s \\ \mathbf{b}_g \end{pmatrix} = \begin{pmatrix} \mathbf{W}_s \Phi_s^T (\mathbf{x} - \bar{\mathbf{x}}) \\ \Phi_g^T (\mathbf{g} - \bar{\mathbf{g}}) \end{pmatrix} \quad (6)$$

where \mathbf{W}_s is a diagonal matrix of weights that measures the unit difference between shape and texture parameters. A simple estimate for \mathbf{W}_s is to weight uniformly with ratio, r , of the total variance in texture and shape, i.e. $r = \frac{\sum_i \lambda_{gi}}{\sum_i \lambda_{si}}$. Then $\mathbf{W}_s = r\mathbf{I}$ (Stegmann, 2000).

As result, using again a PCA, Φ_c holds the t_c highest eigen vectors, and we obtain the combined model.

$$\mathbf{b} = \Phi_c \mathbf{c} \quad (7)$$

Due the linear nature for the model, is possible to express shape, \mathbf{x} , and texture, \mathbf{g} , using the combined model by:

$$\mathbf{x} = \bar{\mathbf{x}} + \Phi_s \mathbf{W}_s^{-1} \Phi_{c,s} \mathbf{c} \quad (8)$$

$$\mathbf{g} = \bar{\mathbf{g}} + \Phi_g \Phi_{c,g} \mathbf{c} \quad (9)$$

where

$$\Phi_c = \begin{pmatrix} \Phi_{cs} \\ \Phi_{cg} \end{pmatrix} \quad (10)$$

\mathbf{c} is a vector of appearance controlling both shape and texture. One AAM instance is built by generating the texture in the normalized frame using eq. 9 and warping-it to the control points given by eq. 8. See figure 2.

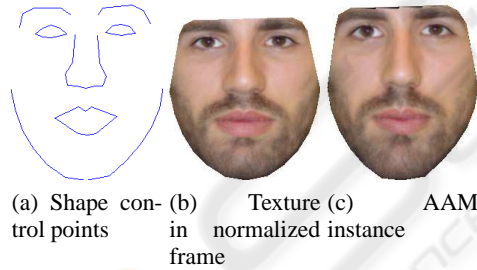


Figure 2: Building a AAM instance.

2.4 Model Training

An AAM search can be treated as an optimization problem, where the texture difference between a model instance and a target image is minimized, $|\mathbf{I}_{image} - \mathbf{I}_{model}|^2$ updating the appearance parameters \mathbf{c} and pose.

Apparently, this could be a hard optimization problem, but we can learn how to solve this class of problems, learning how the model behaves due parameters change, i.e. learning offline the relation between the texture residual and the correspondent parameters error.

Additionally, are considered the similarity parameters for represent the 2D pose. To maintain linearity and keep the identity transformation at zero, these pose parameters are redefined to: $\mathbf{t} = (s_x, s_y, t_x, t_y)^T$ where $s_x = (s \cos(\theta) - 1)$, $s_y = s \sin(\theta)$ represents a combined scale, s , and rotation, θ . The remaining parameters t_x and t_y are the usual translations.

Now the complete model parameters, \mathbf{p} , (a $t_p = t_c + 4$ vector) are given by:

$$\mathbf{p} = (\mathbf{c}^T | \mathbf{t}^T)^T \quad (11)$$

The initial AAM formulation uses the multivariate linear regression approach over the set of training texture residuals, $\delta \mathbf{g}$, and the correspondent model perturbations, $\delta \mathbf{p}$. The goal is to get the optimal prediction matrix, in the least square sense, satisfying the linear relation:

$$\delta \mathbf{p} = \mathbf{R} \delta \mathbf{g} \quad (12)$$

Solving eq. 12 involves perform a set s experiences, building the residuals matrices (\mathbf{P} holds by columns model parameters perturbations and \mathbf{G} holds correspondent texture residuals):

$$\mathbf{P} = \mathbf{R} \mathbf{G} \quad (13)$$

In AAM its safe to say that $m \gg t_s > t_p$, so one possible solution of eq. 13 can be obtained by Principal Component Regression (PCR) projecting the large matrix \mathbf{G} into a k -dimensional subspace, where $k \geq t_p$ which captures the major part of the variation.

Later (T.F.Cootes and C.J.Taylor, 2001) it was suggested a better method, computing the gradient matrix $\frac{\partial \mathbf{r}}{\partial \mathbf{p}}$.

The texture residual vector is defined as:

$$\mathbf{r}(\mathbf{p}) = \mathbf{g}_{image}(\mathbf{p}) - \mathbf{g}_{model}(\mathbf{p}) \quad (14)$$

whre the goal is to find the optimal update at model parameters to minimize $|\mathbf{r}(\mathbf{p})|$. A first order Taylor expansion leads to

$$\mathbf{r}(\mathbf{p} + \delta \mathbf{p}) \approx \mathbf{r}(\mathbf{p}) + \frac{\partial \mathbf{r}(\mathbf{p})}{\partial \mathbf{p}} \delta \mathbf{p} \quad (15)$$

minimizing, in the least square sense, eq. 15 gives

$$\delta \mathbf{p} = - \left(\frac{\partial \mathbf{r}}{\partial \mathbf{p}} \frac{\partial \mathbf{r}}{\partial \mathbf{p}} \right)^{-1} \frac{\partial \mathbf{r}}{\partial \mathbf{p}} \mathbf{r}(\mathbf{p}) \quad (16)$$

and

$$\mathbf{R} = \left(\frac{\partial \mathbf{r}}{\partial \mathbf{p}} \right)^\dagger \quad (17)$$

$\delta\mathbf{p}$ in eq. 16 gives the parameters probable update to fit the model. Regard that, since the sampling is always performed at the reference frame, the prediction matrix, \mathbf{R} , is considered fixed and it can be only estimated once.

2.4.1 Perturbation Scheme

Table 1 shows the model perturbation scheme used in the s experiences to compute \mathbf{R} . The percentage values are referred to the reference shape.

Table 1: Perturbation scheme.

Parameter \mathbf{p}_i	Perturbation $\delta\mathbf{p}_i$
\mathbf{c}_i	$\pm 0.25\sigma_i, \pm 0.5\sigma_i$
Scale	90%, 110%
θ	$\pm 5^\circ, \pm 10^\circ$
t_x, t_y	$\pm 5\%, \pm 10\%$

2.5 Iterative Model Refinement

For a given estimate \mathbf{p}_0 , we used (P.Viola and Jones, 2004) method, the model can be fitted by

- Sample image at $\mathbf{x} \rightarrow \mathbf{g}_{image}$
- Build an AAM instance $\mathbf{AAM}(\mathbf{p}) \rightarrow \mathbf{g}_{model}$
- Compute residual $\delta\mathbf{g} = \mathbf{g}_{image} - \mathbf{g}_{model}$
- Evaluate Error $E_0 = |\delta\mathbf{g}|^2$
- Predict model displacements $\delta\mathbf{p} = \mathbf{R}\delta\mathbf{g}$
- Set $k = 1$
- Establish $\mathbf{p}_1 = \mathbf{p}_0 - k\delta\mathbf{p}$
- If $|\delta\mathbf{g}_1|^2 < E_0$ accept \mathbf{p}_1
- Else try $k = 1.5, k = 0.5, k = 0.25, k = 0.125$

this procedure is repeated until no improvement is made to error $|\delta\mathbf{g}|$. Figure 3 shows a successful AAM search. Note that, as better the initial estimate is, minor the risk of being trap in a local minimum.

3 FACIAL EXPRESSION RECOGNITION

3.1 Linear Discriminant Analysis

The facial expression recognition procedure is performed by firstly describing a set of faces using the AAM model (Bouchra Abboud, 2004). Afterwards, each vector of appearance \mathbf{c} is projected into

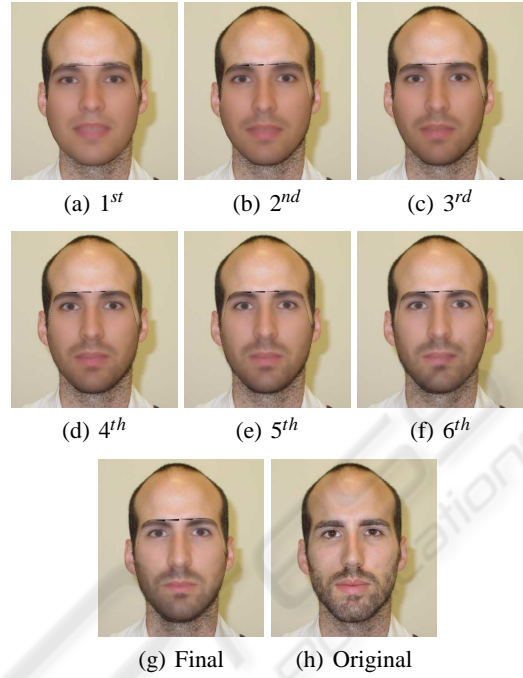


Figure 3: Iterative model refinement.

Fisherspace, applying a Linear Discriminant Analysis (LDA) (Peter N. Belhumeur and Kriegman, 1997). This supervised learning technique consists in optimizing the separability of the dataset observations according to the expression class they belong to. This is done by maximizing the between-class variance while minimizing the within-class variance. These variances are expressed by the two scatter matrixes shown in eq. 18 and eq. 19.

$$S_b = \sum_{j=1}^{n_c} (\mu_j - \mu)(\mu_j - \mu)^T \quad (18)$$

$$S_w = \sum_{j=1}^{n_c} \sum_{i=1}^{N_j} (\mathbf{X}_i^j - \mu_j)(\mathbf{X}_i^j - \mu_j)^T \quad (19)$$

In these expressions, n_c is the number of classes, \mathbf{X}_i^j represents the i^{th} sample of class j , and μ_j and μ are the mean of class j and the mean of all classes, respectively. This linear transformation of data will allow subsequent classification of new images representing one of the expression categories j .

3.2 LDA Evaluation Metric

Similarly to a Principal Component Analysis, a LDA transformation also involves performing an eigenvector decomposition which reflects the importance of the data variance on the transformed subspace. The

resulting data model retains the features that maximize class separability while holding a percentage of the data variance.

In order to evaluate the quality of the data discrimination, we developed a metric based on applying a k-means clustering algorithm on the result of the LDA. Let us consider the ideal case where all the observations were completely separated after LDA. In this case, we could apply a clustering algorithm on the data and obtain n_c groups containing all faces with the same expression. Table 2 shows what would be the ideal clustering result for a dataset of n_s subjects per expression. Note that since there are seven expressions the k-means algorithm would be applied to get seven groups.

Table 2: k-means clustering result for ideal LDA.

	1	2	3	4	5	6	7
Neut	n_s	0	0	0	0	0	0
Happ	0	n_s	0	0	0	0	0
Sad	0	0	n_s	0	0	0	0
Surp	0	0	0	n_s	0	0	0
Ang	0	0	0	0	n_s	0	0
Fear	0	0	0	0	0	n_s	0
Disg	0	0	0	0	0	0	n_s

The metric assigns a discrimination quality value to the clustering result. Its value is calculated by summing the difference between the higher and the lower values of each row of this table (i.e. the sum of the degree of concentration of each class). The higher the metric output, the better is the discrimination. By applying this metric to several LDA runs, with different number of features retained, we can estimate how many modes of variation should be hold to optimize the discrimination.

3.3 Classification

Once defined the axes that maximize the data classes separability, it is possible to classify either training images or unseen faces. The classification for an unknown face image consists in two steps. The first is projecting its appearance vector, \mathbf{c} , on the hyperspace that resulted from the training process. The second is to estimate to each group does this projected point belongs to. In our case this is done by using an adaptation of the nearest-neighbourhood algorithm, which takes in consideration not only the distance of the point to the centre of each group, but also its dispersion. The distance to each group is measured using malahanobis distance (Krzanowski, 1988), eq. 20, since it gives a scaled measure of an instance from a particular class.

$$D = (\mathbf{c} - \bar{\mathbf{c}}_i) \Sigma^{-1} (\mathbf{c} - \bar{\mathbf{c}}_i) \quad (20)$$

\mathbf{c}_i is the vector of extracted appearance parameters, $\bar{\mathbf{c}}_i$ is the centroid of the class multivariate distribution and Σ is the within-class covariance matrix.

4 EXPERIMENTAL RESULTS

For the purpose of this work, an expression database was built. It consists in 21 individuals showing 7 different facial expressions each. These expressions are: neutral expression, happiness, sadness, surprise, anger, fear and disgust. The data set is therefore formed by a total of 147 colour images (640×480). Figure 4 shows AAM model instances of a given subject for each one of the 7 facial expressions used.

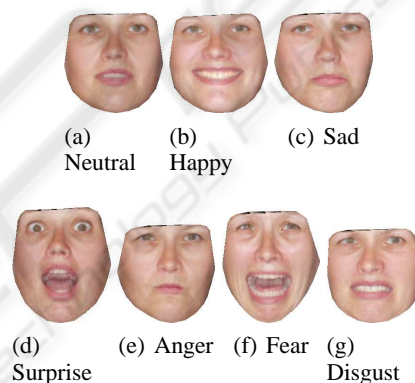


Figure 4: AAM instances of facial expressions used.

The AAM shape model was built using 58 annotated landmarks ($N = 58$). The texture model was generated sampling around 47000 pixels using also colour information ($g = 141000$). Table 3 shows the different values for retained variance used in each of the three PCA that are required to create the combined model, as well as the combined variation modes for each one of this values.

Table 3: Retained variance and correspondent number of modes, t_c .

Variance(%)	t_c
95.0	17
97.0	29
98.0	42
99.0	70
99.5	97
99.9	133

To evaluate the classification results, a leave-one-out cross-validation method was adopted. In this

method each one of the dataset observations is used once as the testing sample, while all the remaining data is used for training.

4.1 First Experience - Optimizing AAM and LDA variation Modes

In a first experiment we studied the influence of the variation modes retained either during AAM construction or during LDA on the classification performance. On these experiment we used all the 147 images of the database. First we performed an evaluation of the discrimination quality by varying the number of discrimination features retained on several face models with different PCA variation modes retained. As discrimination quality metric output depends on the result of a k-means clustering, these tests were repeated for 250 times. A histogram of the best LDA features retained in each trial was created for every AAM model used. Figure 5 shows the result when 99.0% variation modes are retained on our dataset.

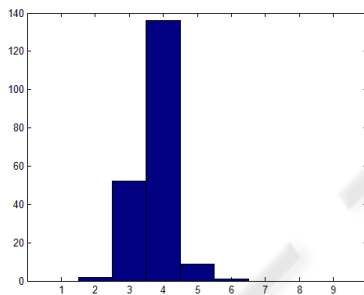


Figure 5: Histogram for best LDA modes on 250 trials.

As it can be seen, three or four LDA modes maximize the separation of classes. This was also observed for 95.0% and 97.0% AAM variation modes.

Using these two values as LDA modes retained, we performed leave-one-out cross-validation classification for face models with 95.0%, 97.0%, 98.0, 99.5% and 99.9% AAM modes. The global classification results are shown on figure 6.

It is clear from this graph observation that the classification performance varies with the percentage of variation retained on the AAM construction process. The best global classifications results were obtained for 99.0% of variance retained.

4.2 Second Experiment - 7 Expressions

On the second experiment we analyse the classification performance for each expression. The same dataset with 21 subjects in 7 different expressions is

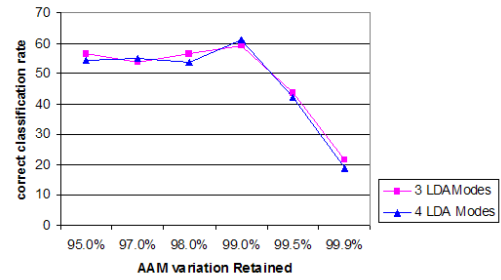


Figure 6: Variation of global classification results with different number of AAM modes retained.

used and a confusion matrix is created that represent the classification results for each expression. The results of the classification for all datasets are resumed on tables 4, 5 and 6 for different values of retained variance.

Table 4: Confusion Matrix 97.0% Overall recognition rate = 55%.

	Neut	Happ	Sad	Surp	Ang	Fear	Disg
Neut	19.0	0	52.38	0	9.52	19.05	0
Happ	0	90.48	0	0	0	4.77	4.77
Sad	9.52	0	61.90	4.77	9.52	14.29	0
Surp	0	0	0	80.95	0	19.05	0
Ang	0	0	14.29	0	33.33	9.52	42.86
Fear	0	4.77	9.52	19.05	14.29	52.38	0
Disg	0	14.29	0	0	38.10	0	47.62

Table 5: Confusion Matrix 98.0% Overall recognition rate = 56.5%.

	Neut	Happ	Sad	Surp	Ang	Fear	Disg
Neut	33.33	0	61.90	0	0	4.76	0
Happ	0	80.95	0	0	0	9.52	9.52
Sad	0	4.76	66.67	0	19.05	9.52	0
Surp	0	0	0	71.43	0	28.57	0
Ang	0	4.76	4.76	0	42.86	14.29	33.33
Fear	0	4.76	9.52	19.05	9.52	52.38	4.76
Disg	0	9.52	0	0	38.10	4.76	47.62

We can see from this results that there is an effect of confusion between neutral and sad expressions, and also between anger and disgust. This suggests that there is an appearance correlation between these two pairs of expressions. In fact, these results are not surprising. Several neuroscience (Killgore and Yurgelun-Todd, 2003) studies have showed that in case of human emotion recognition, to percieve sad and anger expressions, an specific cognitive process is required.

4.3 Third Experiment - 5 Expressions

Experiment three tries to eliminate the confusion effect described on the previous experiment. Faces expressing sadness and anger were excluded both from training and testing. The results of the classification for the same LDA and AAM conditions are expressed on tables 7, 8 and 9.

Table 6: Confusion Matrix 99.0% Overall recognition rate = 61.2%.

	Neut	Happ	Sad	Surp	Ang	Fear	Disg
Neut	52.38	0	42.86	0	4.76	0	0
Happ	0	90.48	4.76	0	0	4.76	0
Sad	4.76	4.76	76.19	0	4.76	4.76	4.76
Surp	0	0	0	76.19	0	23.81	0
Ang	4.76	0	9.52	0	33.33	23.81	28.57
Fear	0	9.52	4.76	14.29	4.76	66.67	0
Disg	0	23.81	4.76	0	33.33	4.76	33.33

Table 7: Confusion Matrix 97.0% Overall recognition rate = 74.3%.

	Neut	Happ	Surp	Fear	Disg
Neut	57.14	0	0	42.86	0
Happ	0	90.48	0	9.52	0
Surp	0	0	76.19	23.81	0
Fear	9.52	4.76	23.81	61.90	0
Disg	0	9.52	0	4.76	85.71

Table 8: Confusion Matrix 98.0% Overall recognition rate = 76.2%.

	Neut	Happ	Surp	Fear	Disg
Neut	90.48	0	0	9.52	0
Happ	0	85.71	0	4.76	9.52
Surp	0	0	71.43	28.57	0
Fear	4.76	4.76	19.05	66.67	4.76
Disg	0	19.05	0	14.29	66.67

Table 9: Confusion Matrix 99.0% Overall recognition rate = 63.8%.

	Neut	Happ	Surp	Fear	Disg
Neut	80.95	9.52	4.76	4.76	0
Happ	9.52	66.67	0	14.29	9.52
Surp	4.76	0	66.67	28.57	0
Fear	4.76	9.52	28.57	52.38	4.76
Disg	9.52	23.81	4.76	9.52	52.38

5 CONCLUSIONS

We used standart AAM model to discribe face entities in a compact way. With LDA we are able to separate the several emotion expression classes and perform classification using mahalanobis distance.

On the AAM model building process, holding more information on the appearance vectors not always results on a better discrimination result. In our experiments, this value rounds 99.0% of variance retained.

The number of LDA eigenvectors used is another parameter of great importance. K-means is used in order to give us a good estimation for this value.

As expected, the larger the number of expressions used, the worse is the overall successful classification rate. The reason for this is that there are correlations between the two pairs of expressions neutral, sad and anger, disgust, comprowed by psico-physics studies.

We achieved, with all the seven expressions, an

overall successful recognition rate of 61.1%. Removing correlated expressions, this recognition rate increases to a maximum of 76.2%.

REFERENCES

- Batista, J. P. (2007). Locating facial features using an anthropometric face model for determining the gaze of faces in image sequences. *ICIAR2007 - Image Analysis and Recognition*.
- Bouchra Abboud, Frank Davoine, M. D. (2004). Facial expression recognition and synthesis based on an appearance model. *Signal Processing Image Communication*.
- G. Finlayson, S. Hordley, G. S. and Tian, G. Y. (2005). Illuminant and device invariant color using histogram equalisation. *Pattern Recognition*.
- Killgore, W. D. and Yurgelun-Todd, D. A. (2003). Activation of the amygdala and anterior cingulate during nonconscious processing of sad versus happy faces. *NeuralImage*.
- Krzanowski, W. J. (1988). Principles of multivariate analysis. *Oxford University Press*.
- Peter N. Belhumeur, J. P. H. and Kriegman, D. J. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- P.Viola and Jones, M. (2004). Rapid object detection using a boosted cascade of simple features. *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*.
- Stegmann, M. B. (2000). Active appearance models theory, extensions & cases. Master's thesis, IMM Technical University of Denmark.
- Stegmann, M. B. and Gomez, D. D. (2002). A brief introduction to statistical shape analysis. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark.
- T. Dalgleish, M. P. (1999). Handbook of cognition and emotion. *John Wiley & Sons Ltd*.
- T.F.Cootes and C.J.Taylor (2004). Statistical models of appearance for computer vision. Technical report, Imaging Science and Biomedical Engineering - University of Manchester.
- T.F.Cootes, G. E. and C.J.Taylor (2001). Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.