A Model Based Method for Automatic Facial Expression Recognition

Hans van Kuilenburg[1], Marco Wiering[2], and Marten den Uyl[3]

[1] VicarVision, Amsterdam, The Netherlands
van.kuilenburg@wanadoo.nl
http://home.wanadoo.nl/van.kuilenburg/
[2] Utrecht University, Utrecht, The Netherlands
marco@cs.uu.nl
[3] VicarVision, Amsterdam, The Netherlands
denuyl@vicarvision.nl

Abstract. Automatic facial expression recognition is a research topic with interesting applications in the field of human-computer interaction, psychology and product marketing. The classification accuracy for an automatic system which uses static images as input is however largely limited by the image quality, lighting conditions and the orientation of the depicted face. These problems can be partially overcome by using a holistic model based approach called the Active Appearance Model. A system will be described that can classify expressions from one of the emotional categories joy, anger, sadness, surprise, fear and disgust with remarkable accuracy. It is also able to detect smaller, local facial features based on minimal muscular movements described by the Facial Action Coding System (FACS). Finally, we show how the system can be used for expression analysis and synthesis.

1 Introduction

Facial expressions can contain a great deal of information and the desire to automatically extract this information has been continuously increasing. Several applications for automatic facial expression recognition can be found in the field of human-computer interaction. In every day human-to-human interaction, information is exchanged in a highly multi-modal way in which speech only plays a modest role. An effective automatic expression recognition system could take human-computer interaction to the next level.

Automatic expression analysis can be of particular relevance for a number of expression monitoring applications where it would be undesirable or even infeasible to manually annotate the available data. E.g., the reaction of people in test-panels could be automatically monitored and forensic investigation could benefit from a method to automatically detect signs of extreme emotions, fear or aggression as an early warning system.

Decades of research have already led to the development of systems that achieve a reasonable expression classification performance. A detailed account

J. Gama et al. (Eds.): ECML 2005, LNAI 3720, pp. 194–205, 2005.

of all the advances on the field of automatic expression analysis can be found in [17] or [10]. Unfortunately, most of the developed systems have severe limitations on the settings of their use, making them unsuitable for real-life applications.

The limitations in automatic expression classification performance are to a large extend the result of the high variability that can be found in images containing a face. If we do not want to be limited to a specific setting and if we do not want to require active participation of the individuals depicted on the images, we will see an extremely large variety in lighting conditions, resolution, pose and orientation. In order to be able to analyze all these images correctly, an approach seems to be desirable that can compactly detect and describe these sources of variation and thus separate them from the actual information we are looking for.

The Active Appearance Model (AAM) first described by Cootes and Taylor [4] enables us to (fully) automatically create a model of a face depicted in an image. The created models are realistic looking faces, closely resembling the original. Previous research projects have indicated that the AAM provides a good generalization to varying lighting / pose conditions as it is able to compactly represent these sources of variations.

Many leading researchers in the field of expression classification have chosen very different, local methods for classification. Local methods have the advantage of potentiality achieving a very high resolution in a small area of the face. However, as they lack global facial information, it will be very hard for a local method to separate changes caused by differences in lighting or pose from changes caused by expressions. Consequently, the local method will have rather poor generalization properties. We do not want to limit ourself to situations where we have high-resolution video material available either, but instead want a single facial image to be sufficient. We have therefore chosen to use the holistic, model based Active Appearance Model as our core technique. To make this system fully automatic, a deformable template face framing method, very similar to the one described in [20] is used preliminary to the AAM modeling phase.

The next section will describe the AAM implementation that was used for this project (based on previous work by [16]). In section 3 we will show how appearance models can be used to classify facial expressions based on two different categorization systems. Section 4 describes how we can further analyze or synthesize facial expressions. Finally, we will come to a conclusion in section 5.

2 The Active Appearance Model

To train the AAM [4], we require the presence of a (manually) annotated set X of facial images. The shape of a face is defined by a *shape vector* S containing the coordinates of M *landmark points* in a face image I.

$$S = ((x_1, y_1), (x_2, y_2), ..., (x_M, y_M))^T$$

Landmark points are points in the 2D plane of a face image at easily distinguishable reference points, points which can be identified reliably in any face image we might want to analyze. Considering the invariability of shapes under Euclidian transformations, we can remove the effect of misplacement, size and rotation by aligning each shape vector in the set of all shape vectors X^s to the mean shape vector $\bar{s}$, which can be implemented as an iterative procedure.

We then apply Principle Component Analysis (PCA) [13], which transforms the shapes to a new low dimensional shape subspace in R^D where $D < 2M$. An element S from the original set of shapes can now be approximated by some b^s of length D where:

$$S \approx \Phi^s \cdot b^s + \bar{s} \tag{1}$$

Where Φ^s is the covariance matrix consisting of the D principal orthogonal modes of variation in X^s:

$$\Phi^s = (e_1^s | e_2^s | ... | e_D^s)$$

The eigenvalues λ_j^s of the covariance matrix define the variance of the data in the direction of the corresponding eigenvector e_j^s. Thus, when generating new shapes, we can bound the elements in b^s as shown below, to allow variation within 99% of a normally distributed function:

$$-3\sqrt{\lambda_j^s} \leq b_j^s \leq 3\sqrt{\lambda_j^s}$$

A *texture vector* of a face image is defined as the vector of intensity values of N pixels that lie within the outer bounds of the corresponding shape vector:

$$T = [g_1, g_2, ..., g_N]^T$$

Delauny triangulation [19] is performed on the texture maps to transform them to a reference shape (the mean shape can be used for this). This results in so called *shape-free patches* of pixel intensities, which should then be *photometrically aligned* to remove the effect of general lightning differences. This can again be done using an iterative approach.

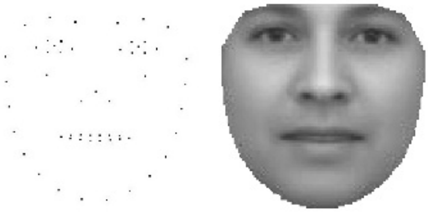


Fig. 1. The mean face shape and the mean face texture aligned to the mean shape

PCA is then applied on the texture vectors, after which a texture vector T from the original data set can be represented by a vector b^t.

$$T \approx \Phi^t \cdot b^t + \bar{t} \tag{2}$$

The elements in b^t are again bounded by:

$$-3\sqrt{\lambda_k^t} \leq b_k^t \leq 3\sqrt{\lambda_k^t}$$

Where λ_k^t represents the eigenvalue of the corresponding eigenvector across the data set X^t.

The appearance model combines the two vectors b^s and b^t into a single parameter vector b^a. First, the shape and texture vector are concatenated. Because these two are of a different nature and thus of a different relevance, one of the terms will be weighted:

$$b^{st} = \begin{pmatrix} w^s b^s \\ b^t \end{pmatrix}$$

Estimating the correct value for w^s can be done by systematically displacing the elements of the shape vector over the examples in the training set and calculating the corresponding difference in pixel intensity. As an alternative, we can set w^s as the ratio of the total pixel intensity variation to the total shape variation. PCA is then applied one last time to remove possible correlation between shape and texture parameters and create an even more compact representation:

$$b^{st} = \Phi^a b^a \qquad (3)$$

We will refer to b^a as the *appearance vector* from now on as it compactly describes both the shape and the texture of an object.

The online task of the Active Appearance Model is to find a model instance which optimally models the face in a previously unseen image (a model-fit). Given a face image I, the AAM attempt to find the optimal model parameters b^a and the optimal pose parameters $u = [t^x, t^y, s, \Theta]^T$ where t^x and t^y are translations in the x and y directions, s is the scaling factor and Θ is the rotation.

The difference vector $\delta t = t^{image} - t^{model}$ defines the difference in pixel intensity between input image and the model instance. By minimizing $E = ||\delta t||^2$ we thus minimize the difference in pixel intensities. The method used by the AAM for doing this assumes that the optimal parameter update can be estimated from δt. Moreover, this relationship is assumed to be nearly linear.

A prediction matrix is used to update the model parameters b^a and u in an iterative way until no significant change occurs anymore. Usually, separate prediction matrices are used for b^a and u, so we have:

$$\delta b^a = R^{b^a}\delta t \quad \text{and} \qquad \delta u = R^u \delta t$$

The prediction matrices R^{b^a} and R^u are learned from the training data by linear regression using examples which are systematically displaced over one of the model or pose parameters.

After obtaining the prediction matrices, parameters are updated in an iterative way ($b^a \leftarrow b^a + \alpha\delta b^a$ and $u \leftarrow u + \alpha\delta u$) where α is a stepsize parameter, until no significant change in error occurs anymore. Under this iterative approach, the linearity assumption seems to hold well enough when the initial model placement does not deviate too much from the actual position of the face in the image.

3 Expressions and Classification

In order to create a system that can automatically derive meaningful expression information from a face, it might prove important to have a clear and formalized method of describing the expression on a face, for which several methods have been proposed. We describe two commonly used classification systems for facial expressions and then present our results.

3.1 Facial Action Coding System (FACS)

The Facial Action Coding System (FACS) presented by Ekman & Friesen in 1978 [7] is by far the most used system. It codes the various possible facial movements based on an analysis of the facial anatomy. FACS contains a list of approximately (depending on the specific revision) 46 minimal facial movements called ‘Action Units’ and their underlying muscular basis. Over the years, this system has become a standard for coding facial expressions. The original system has undergone a major revision in 2002 [9] and extended to include gradations of activation for the Action Units (AUs).

Using FACS, practically all facial cues can be accurately described in terms of Action Units, which appear to be the smallest possible changing units in a face. This makes it a very powerful system to accurately annotate facial expressions.

The only major downside to this approach is the fact that no or little meaning can be attached to the activation of one of the Action Units. E.g., to know that the Levator palpebrae superioris is contracted is information that might only be relevant for a very select number of applications. Instead, a categorization based on the meaning expressions convey may be more useful for most applications.

3.2 Emotional Expressions

Many expressions carry an emotional content with them. The exact relationship between expressions and emotions has been studied extensively, which has resulted in several theories. For a detailed discussion see [11, 8, 18, 6].

Already in 1970, P. Ekman reported the existence of 6 universal facial expressions related to the emotional states: anger, disgust, fear, joy, sadness and surprise [5]. A constant debate on whether these expressions are really universal, or vary by culture, has been going on ever since.

Obviously it is not the case that any expression can be classified into one of Ekman’s 6 emotional expression categories [12]. Facial movements can be of varying intensity and there are blends of emotional expressions and variations within a category. Also, there are facial movements which are meant only for conversational purposes or are considered idiosyncratic. However, if we do want to make a categorization of expressions based on emotions, Ekman’s universal emotional expressions might be an obvious choice, for the system is already widely used and categories that are made represent clear concepts, making them intuitively easy to deal with.

3.3 Automatic Emotion Expression Classification

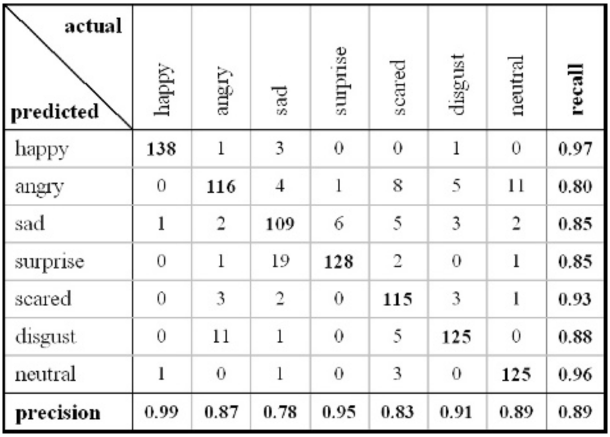
In the previous section, we described how the AAM can be used to derive a realistic model of a depicted face. There are several ways to extract a compact representation of the facial features using this model. After a model fit has been created, the accurate position of a face is clear and so are the locations of all the key points in the face (the landmark points). This introduces two promising options. First, accurate image slices can be made of selected regions of the face to be used directly by a classifier or after applying a 'smart' compression. However, an easier option seems to exist. The face model which has been constructed is represented entirely by a very compact vector (the appearance vector). This appearance vector could be perfectly suitable for use as input for a classification method, if it contains all relevant information needed to distinguish between the different expression classes. Previous experiments [21, 14] have shown that this latter option gives far better results.

Our goal is to come to a classification of all the 6 universal emotional expressions, plus the neutral expression. In principle, we can achieve this by creating and training 7 independent classifiers. We used neural networks trained with backpropagation since these have been proven their abilities for pattern recognition tasks [1]. The final expression judgement of a face image could then be based on the network with the highest output. Experiments have shown, however, that the networks resulting from this procedure miss what you might call 'mutual responsiveness'. When observing a series of images, where one emotional expression is shown with increased intensity, one would expect the output of one network to increase and the outputs of the other networks to automatically decrease or level out to zero. However, this behavior does not appear in all situations and not seldom are there several or no networks at all with a high output, even though this situation does not appear in the training data.

We can create more favorable behavior by training one classification network with 7 outputs for the different emotional categories. This also boosts overall performance significantly. We used a 3-layer feed-forward neural network, with 94 input neurons (=the length of the appearance vector), 15 hidden neurons and 7 output neurons (=the number of expression categories). We used the backpropagation algorithm to train the network and leave-one-out cross-validation to determine the true test performance. The optimal number of training epochs (which was around 1500) was estimated by iteratively searching around the optimum found using a small stop-set. The training material consisted of 1512 appearance vectors that were automatically extracted and had an accurate AAM fit.

Table 1 shows the results in the form of a confusion matrix when we force the network to make a choice (by picking the highest output value) on the 'Karolinska Directed Emotional Faces' set [15] containing 980 high quality facial images showing one of the universal emotional expressions or a neutral expression. 89% of all faces presented to the classifier is classified correctly, which is a very promising result as it is among the highest reported results on emotional expression classification from static images.

Table 1. Performance of the 7-fold classifier on the Karolinska data set using leave-one-out cross-validation



actual / predicted	happy	angry	sad	surprise	scared	disgust	neutral	**recall**
happy	**138**	1	3	0	0	1	0	**0.97**
angry	0	**116**	4	1	8	5	11	**0.80**
sad	1	2	**109**	6	5	3	2	**0.85**
surprise	0	1	19	**128**	2	0	1	**0.85**
scared	0	3	2	0	**115**	3	1	**0.93**
disgust	0	11	1	0	5	**125**	0	**0.88**
neutral	1	0	1	0	3	0	**125**	**0.96**
precision	**0.99**	**0.87**	**0.78**	**0.95**	**0.83**	**0.91**	**0.89**	**0.89**

3.4 FACS Classification

Although we have mainly focussed on the automatic classification of facial expressions in one of Ekman's 6 universal emotional expression categories, we have as a side-study, trained the system to give the FACS scoring of a face (using the 2002 revision including gradations). If this classifier performs well, this would suggest that even local features can be modeled correctly by the AAM, without requiring a training set specifically selected for this.

Action Units (AUs) described by the FACS do not necessarily have to be independent. In practice, there are many constraints on the co-occurrence of AUs. This is reason to take a similar approach as we did for the emotional classifier concerning the choice between building separate classifiers and building one large classifier for all AUs at once. If there are constraints, these can be modeled in the large network and outputs could be better adjusted to one another.

For some AUs, far too little training data was available to perform a meaningful training. Only the 15 AUs present most frequently in the training set (AUs 1, 2, 4, 5, 6, 7, 9, 12, 15, 17, 20, 23, 24, 25 and 27) were therefore selected for training. This limits the functionality of the system, but retraining the classifier with more annotated faces will always remain an option for real applications. Again, we used a 3-layer feedforward neural network, with 94 input neurons, 20 hidden neurons and 15 output neurons (=the number of selected AUs) and use backpropagation with leave-one-out cross-validation. The training material consisted of 858 appearance vectors of images from the Cohn-Kanade AU-Coded Facial Expression Database [3] with an accurate AAM fit.

Table 2 shows the performance of the FACS classifier after training, where a classification is considered correct if it does not deviate more than one point on the five-point scale of intensity by which the training data is annotated. The AUs are detected with an average accuracy of 86%, but it should be mentioned

Table 2. FACS classifier performance on 15 Action Units

Action Unit:	01	02	04	05	06	07	09	12	15	17	20	23	24	25	27	Average
Accuracy:	.86	.88	.81	.86	.81	.89	.93	.83	.89	.86	.84	.83	.83	.90	.89	**.86**

that this still means that most classified faces will have one or more AUs scored incorrectly. If we are only interested in the activation of one or two AUs, these results are promising, but if we are looking for an accurate automatic FACS scoring device, significant improvements are still needed.

4 Expression Analysis and Synthesis

The previous experiments have shown that the appearance vector contains expression information that can be used to classify a face model. Alternatively, it is also possible to extract and isolate the information that is related to expressions, which enables us to visualize the distinguishing features for a certain expression and also allows expression synthesis.

4.1 Visualization of Features Relevant for Emotional Expression Classification

Blanz and Vetter (1999) have shown that the information concerning expressions can be extracted from appearance vectors in a straightforward way. Consider two images of the same individual with similar lighting and pose, one image showing some expression and the other showing a neutral face. We can calculate the difference between the two corresponding appearance vectors, which would give us information about the expression shown for this person. By averaging over a set of image-pairs, we can derive 'prototypical vectors' for a certain expression.

Besides some concerns about the reliability of this approach, since features might be averaged out, another downside is the fact that although the derived 'prototypical vectors' can give some cues to what expressions look like, where they are formed and what influence they have on shape and texture of a face, they can not be used directly to analyze those features which are important to *distinguish* one expression from another, even though this would be very useful information to have for anyone working in the field of expression classification.

There is an alternative possibility however. Consider a feedforward neural network, which is calculated as [1]:

$$y_k = g(\sum_{j=0}^{M_2} w_{kj} f(\sum_{i=0}^{M_1} w_{ji} x_i)) \qquad (4)$$

Where y_k is the output of the k-th output neuron; g and f are the activation functions of the output layer and hidden layer respectively, w_{kj} is the weight between the k-th output neuron and the j-th hidden neuron, w_{ji} is the weight between the j-th hidden neuron and the i-th input neuron and x_i is the activation of the i-th input neuron.

We train a network using face images showing a certain expression as positive examples and using images of all other expressions as negative examples. This network turns out to have an optimal (or nearly optimal) performance when it has only one hidden neuron. In this case formula 4 can be greatly simplified to:

$$y_1 = g(w_1 f(\sum_{i=0}^{M_1} w_{1i} x_i)) \tag{5}$$

Since all our input neurons are connected to all hidden neurons, we can also write: $y_1 = g(w_1 f(w^T x))$ where x is the input vector and w is a vector containing the weights between input neurons and hidden neuron.

By iteratively propagating an error over the network we can find an instance of x which gives the output $y_1 = 1$. This is not necessary however, as we can already see in the formula above that w determines exactly the relative magnitude of the influence the input neurons will have on the output of the network, since functions g and h are monotonously increasing and w_1 only determines the sign of this influence. w thus denotes the relevance of elements in the appearance vector for the classifier output.

In order to visualize what w represents, we can create a new model instance where we take w directly as the appearance vector ($b^a = w$) with the bias value left out of w. As explained in section 2, we can extract the uncompressed texture and shape vectors from this new appearance vector using formulas 1 and 2, but for visualization purposes we do not add the mean shape and texture. These vectors are indicators of the relevance of either the positioning of landmark points or the pixel intensity in a shape-free patch. The relevance of pixel intensities can be visualized straightforwardly as shown in figure 2. All elements have been converted to absolute values, and a global scaling and offset operator has been used to create black pixels for the most significant indicators and white pixels for what is considered not significant at all by the classifier.

Some of the features we can distinguish are according to what we might have expected beforehand, just to name a few, we can see the changes in the mouth corners when a person smiles, the drawn together eyebrows for an angry face, eyes being slightly closed and opened wide for the sad and surprised expression respectively, the ‘lip puckerer’ in the sad expression and the widened nose in the disgusted expression. Another indicator that the method is working successfully is that the irises are considered irrelevant for all expressions since their appearance remains fairly constant for all expressions. Other features that are considered very significant by the classifier are less obvious to explain and might be interesting material for experts to look at and further analyze.

The vector containing the relevance of shape information does not consist of coordinates, but rather of directed velocity vectors starting at the landmark points we have defined. This can be visualized by drawing the velocity vectors using the mean shape as a reference frame [21].

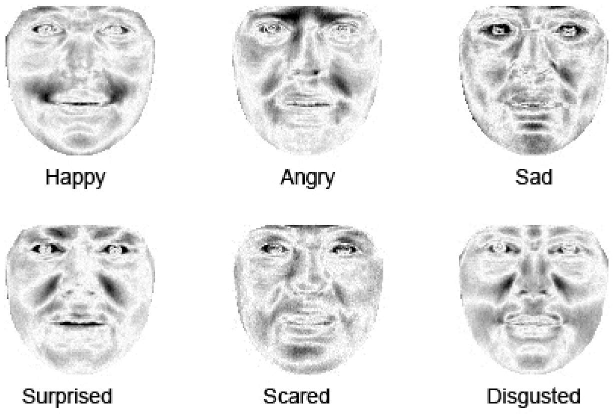


Fig. 2. Relevance of texture information for the 6 emotional expression classifier

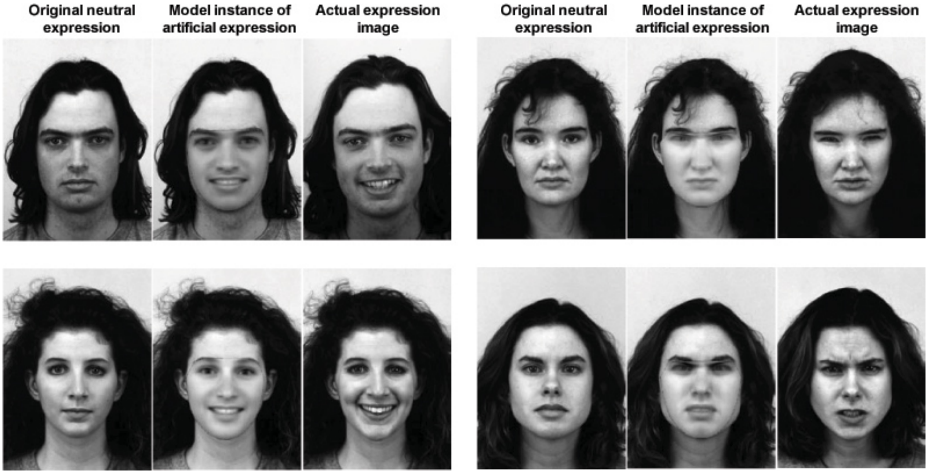

Original neutral expression
Model instance of artificial expression
Actual expression image
Original neutral expression
Model instance of artificial expression
Actual expression image

Fig. 3. Artificially created expressions; original images from [15]

4.2 Expression Synthesis Through Network Analysis

In the previous experiment, we have modeled the information considered relevant by an emotional expression classifier. If we on the other hand purely want to synthesize an expression, the discriminating features between one emotional expression and all the other different expression categories are of little use. Therefore, we trained new neural networks using only neutral images and images of one emotional expression category at a time. Thus, the discriminating features the classifier is supposed to model are those features which discriminate between a neutral face and a face showing some expression. Extracting the weight vector from these networks and adding a multiple of them to an appearance vector might proof to be a successful way for expression synthesis. Since the elements of the appearance vector are orthogonal, this is a valid operation.

Figure 3 gives some examples of neutral faces which have been changed into faces displaying a certain expression using the method above. As a reference, a real picture of the person displaying this expression has been added.

The artificially created expressions look natural and convincing and only little identity information appears to be lost. As we only have one fixed difference vector for each expression, one might expect that the synthesized expressions contain no personal traits. However, the two series on the right in figure 3 show that variations in expressions can occur for different initial model instances.

5 Conclusions

By using the Active Appearance Model and directly using the appearance vectors as classifier input, we have managed to achieve very promising classification performance. Since we are using a model based method, lighting and orientation differences have little, if any, effect on the classifier's performance. Background variation is no significant problem for the system and the system requires only static images of reasonable quality; laboratory conditions are not required.

An emotional expression classifier was trained which has an accuracy of 89%. The emotional expressions that were investigated must thus have been represented quite accurately in the appearance vectors. Using a similar approach, a classifier has been trained to detect very local facial movements which can be coded using the Facial Action Coding System. This classifier has been trained on 15 different facial movements (Action Units) and classifies each Action Unit with an average performance of 86%.

Using trained classification networks, it is possible to visualize exactly what the classifiers consider relevant/discriminating information for a certain expression. This provides us with accurate information concerning the areas of the face which provide information that is important for a good classifier performance. Further analysis of these results is needed in order to come to a more detailed conclusion.

Again using information obtained from trained classifiers, a difference vector can be extracted which characterizes a certain emotional expression. By adding this difference vector to the appearance vector of a face model, we have shown how expressions can be generated. This method seems to work rather well, as only little personal information appears to be lost, while the generated expressions are clearly identifiable and convincing to a human observer.

References

1. C. M. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.
2. V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194. ACM Press/Addison-Wesley Publishing Co., 1999.

3. J.F. Cohn and T. Kanade. Cohn-Kanade AU-Coded Facial Expression Database. Pittsburgh University, 1999.
4. T. Cootes and C. Taylor. Statistical models of appearance for computer vision. Technical report, University of Manchester, Wolfson Image Analysis Unit, Imaging Science and Biomedical Engineering, 2000.
5. P. Ekman. Universal facial expressions of emotion. *California Mental Health Research Digest*, 8:151–158, 1970.
6. P. Ekman and R.J. Davidson. *The Nature of Emotion - Fundamental Questions.* Oxford University Press, New York, 1994.
7. P. Ekman and W. Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement.* Consulting Psychologists Press, Palo Alto, CA, 1978.
8. P. Ekman, W.V. Friesen, and P. Ellsworth. *Emotion in the Human Face.* Pergamon Press, 1972.
9. P. Ekman, W.V. Friesen, and J.C. Hager. *The Facial Action Coding System.* Weidenfeld & Nicolson, London, 2002.
10. B. Fasel and J. Luettin. Automatic facial expression analysis: A survey. *Pattern Recognition*, 36(1):259–275, 2003.
11. N. Frijda. *The Emotions.* Cambridge University Press & Editions de la Maison des Sciences de l'Homme, Cambridge, Paris, 1986.
12. J. Hager and P. Ekman. The essential behavioral science of the face and gesture that computer scientists need to know. In *Proceedings of the International Workshop on Automatic Face and Gesture Recognition*, pages 7–11, 1995.
13. J.E. Jackson. *A User's Guide to Principal Components.* John Wiley and Sons, Inc., 1991.
14. E. Lebert. Facial expression classification. Experiment report, Sentient Machine Research, Amsterdam, the Netherlands, 1997.
15. D. Lundqvist, A. Flykt, and A. Öhman. The Karolinska Directed Emotional Faces - KDEF. CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet, 1998.
16. M. Nieber. Global structure of the ActiveModelLib. Software architecture description, Vicar Vision BV, Amsterdam, the Netherlands, 2003.
17. M. Pantic. Automatic analysis of facial expressions: The state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1424–1445, 2000.
18. J.A. Russell and J.M. Fernandez-Dols, editors. *The Psychology of Facial Expression.* Cambridge University Press, 1997.
19. J.R. Shewchuk. Triangle: engineering a 2D quality mesh generator and Delaunay triangulator. *Applied Computational Geometry, FCRC96 Workshop*, pages 203–222, 1996.
20. K.K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, 1998.
21. H. Van Kuilenburg. Expressions exposed: Model based methods for expression analysis. Master's thesis, Department of Philosophy, Utrecht University, The Netherlands, 2005.