Facial Expression Modeling using Component AAM Models – Gaming Applications

Ioana Bacivarov
College of Engineering & Informatics,
National University of Ireland Galway
Galway, Ireland
ioanabacivarov@yahoo.com

Peter M. Corcoran, *Senior Member, IEEE*
College of Engineering & Informatics,
National University of Ireland Galway
Galway, Ireland
peter.corcoran@nuigalway.ie

Abstract—**In this paper we provide preliminary results on a new modeling approach for improved determination of facial expressions from a low-resolution video stream. An initial proof-of-concept using an extension of an Active Appearance Model (AAM) to measure facial parameters and a set of classifiers to determine facial states are described. We also discuss the potential for applying this technique to determine the emotional state of the players of a computer game and suggest how this information can be integrated into the workflow of a game. In addition we describe a number of use cases where the game environment can be adapted based on feedback from the players.**

Keywords- interactive gaming, affective imaging, expression classification and recognition, AAM, SVM

I. Introduction (Heading 1)

Computer gaming has grown from its humble origins to become a global industry rivaling the movie industry in terms of scale and economic impact. The technology of gaming continues to improve and evolve at a very rapid pace both in terms of control interface and the graphical display of the gaming world. Today's user interfaces feature more sophisticated techniques for players to interact and play co-operatively with one another. It is possible to have real-time video and audio links between the real players so they can co-ordinate their group gameplay. .

However the emphasis remains on the player being drawn into the artificial game world of the computer. There is still little scope for the gaming environment to reach back to the the players, sensing and empathizing with their moods and feelings. Given the sophistication of modern AI game engines we feel this is a missed opportunity and that gaming engines will soon need to evolve to develop and provide methods to empathize with individual game players.

In this paper we present one practical approach to providing a direct connection back to the game player. We describe an approach to modeling and classifying the facial expressions of a game player using a low-resolution webcam and state-of-art face detection and face modeling techniques. Our system in unoptimized form, is capable of running in real-time on a standard desktop PC. With further refinement it could be implemented on a dedicated embedded device, or converted to a dedicated hardware subsystem. However, in this paper we are primarily concerned with presenting a working proof-of-concept.

This paper is organized as follows. An overview of our system is given in section II; section III describes how facial features can be extracted using AAM models; the section IV and V respectively describe improved eye and lips models; section VI describe the full component model for a face region and section VII explains how we optimized the relevant features which are extracted using this model. In Section VIII we explain our approach to train classifiers and how these are used to determine facial expressions. In section IX a range of representative results are presented showing the discriminative abilities of our system between different facial expressions. Finally in section X we discuss some applications of our system in computer gaming, draw some conclusions and present proposals for future work.

II. System Architecture

A system that performs automatic face recognition or expression recognition typically is comprised of three main subsystems, as shown in *Figure 1*: (i) a face detection module, (ii) a feature extraction module, and (iii) a classification module which determines a similarity between the set of extracted features and a library of reference feature sets. Other filters or data preprocessing modules can be used between these main modules to improve the detection, feature extraction or classification results..

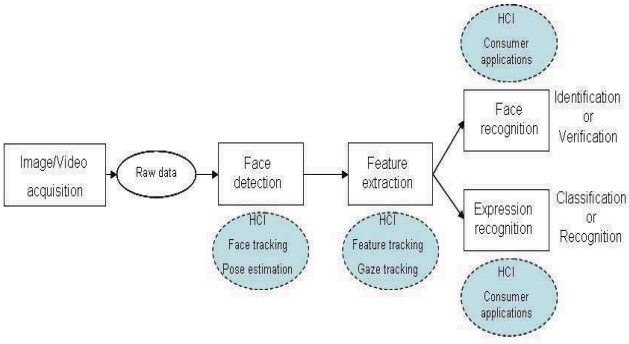


Figure 1: Generic Facial Analysis/Classification System

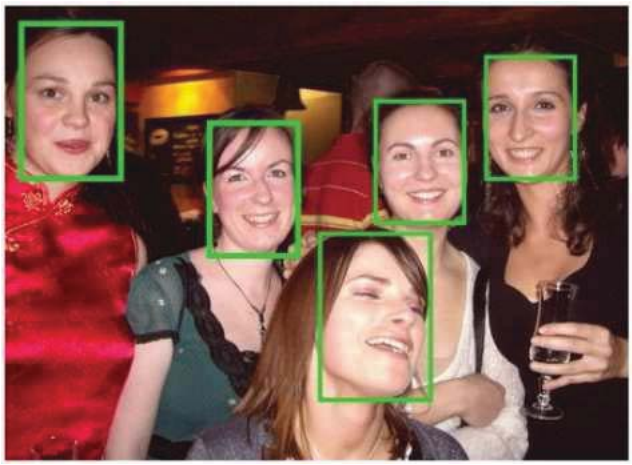
In the first module, it is decided whether the input picture or the input video sequence contains one or more faces. If so, then facial features are extracted from the detected faces, in our case by applying an advanced face model which encodes the facial features by a set of parameters. As a third step, the facial features – determined as a set of output parameters from the

978-1-4244-4460-1/09/$25.00 ©2009 IEEE

model - are classified in order to perform facial recognition or expression classification..

A. Face Detection Module

In our system, we employ a face detector module as initialization for the AAM search. ***Face detection*** can be defined as the ability to detect and localize faces within an image or a scene. In the last few years, many different face detection techniques have been proposed in literature. A comprehensive survey of the face detection methods is presented in [1]. State-of-the-art face detection methods provide real-time solutions that report high detection rates.



In this field the most significant advances in the last decade are due to the work of Paul Viola and David Jones who proposed a face detector based on rectangular *Haar classifiers* and the *integral image* representation of an input image [2]. This detector is the fastest face detector reported in the literature so far. It is able to perform for semi-frontal faces in real-time and is highly accurate. The Viola-Jones face detector is presently the gold standard against which other face detection techniques are benchmarked. In our system, we employ the Viola-Jones face detector, as initialization for the AAM search. An example of face detection using the Viola and Jones algorithm is shown in *Figure 2*. This algorithm has been implemented in OpenCV [3], [4] which is a free computer vision library used widely by the computer vision research community. Figure2: Examples of detected faces using the Viola-Jones algorithm.

B. Feature Extraction Module

In our work, we extend the AAM approach for facial feature extraction. Although AAM is a powerful tool for image interpretation, the conventional algorithm is unreliable when dealing with generalized facial expressions. It is a global face model whereas the key features which determine facial expressions are local features. It is these local features which are responsible for most of the relevant facial variation.

Component-based AAM [5] offers a practical solution to these drawbacks. It combines a global face model with a series of sub-models. These sub-models are typically component parts of the object to be modeled. This approach benefits from both the generality of a global AAM model and the local optimizations provided by its sub-models. We adjust the model through training to be robust to small to medium pose variations and to directional illumination changes. We then demonstrate the benefits of our system, with respect to expression classification and recognition, taking into account the real-time requirements of such a system.

C. Expression Classification Module

In our work, two classifiers are compared: Nearest Neighbor (NN) and Support Vector Machine (SVM). The choice of the two classifiers is based on their positive results obtained in the literature. In [6] it is stated and it is proved by experiments for gender classification and face recognition that SVM is typically among the top two classifiers, and the other top ranking classifier is one of the Euclidean-NN rule or the cosine-NN rule [7]. As we designed them, in this work these classifiers use the relevant AAM parameters to choose between facial expressions.

III. The Extraction of Facial Features using AAM

A. Statistical models of appearance (AAM)

AAM was proposed by Cootes et al. [8] in 1998 as a deformable model, capable of interpreting and synthesizing new images of the object of interest. The desired shape to be modeled – in our case a facial region - is annotated by a number of landmark points. A shape vector is given by the concatenated coordinates of all landmark points and may be formally written as, $s=(x1,x2,\ldots,xL,y1,y2,\ldots,yL)^T$, where L is the number of landmark points.

The shape model is obtained by applying Principal Component Analysis (PCA) on the set of aligned shapes:

$$s=\bar{s}+\varphi_s b_s \quad , \quad \bar{s}=\frac{1}{N_s}\sum_{i=1}^{N_s} s_i \qquad (1)$$

where $\bar{s}$ is the mean shape vector, and N_S is the number of shape observations; φ_s is the matrix having the eigenvectors as its columns; b_s defines the set of parameters of the shape model.

The texture, defined as the pixel values across the object of interest, is also statistically modeled. Face patches are first warped into the mean shape based on a triangulation algorithm. Then a texture vector $t=(t1,t2,\ldots,tp)^T$ is built for each training image by sampling the values across the shape normalized patches. The texture model is also derived by means of PCA on the texture vectors:

$$t=\bar{t}+\varphi_t b_t \quad , \quad \bar{t}=\frac{1}{N}\sum_{i=1}^{N_t} t_i \qquad (2)$$

where $\bar{t}$ is the mean texture vector, with N_t as the number of texture observations; φ_t is the matrix of eigenvectors, and b_t the texture parameters.

The sets of shape and texture parameters $c=\begin{pmatrix} W_s b_s \\ b_t \end{pmatrix}$ are used to describe the overall appearance variability of the modeled object, where W_s is a vector of weights used to compensate the differences in units between shape and texture parameters.

This work supported by the Innovation Partnership Program of Enterprise-Ireland and FotoNation Ireland Ltd

After a statistical model of appearance is created, an AAM algorithm is employed to fit the statistical model to a new image. This determines the best match of the model to the image allowing us to find the parameters of the model which generate a synthetic image as close as possible to the target image.

B. Relevant AAM Parameters for illustrating emotions

AAM extracts two types of features: g*eometric features* describe shapes, deformations and locations of facial components, and poses variations; *appearance features* describe skin texture changes, e.g., furrows and bulges, blushing, expression wrinkles, illumination variations. .

Geometric features are more affected when expressing emotions. As examples, when surprised, eyes and mouth open widely, the latter resulting in an elongated chin; when sad, we often blink; when angry, eyebrows tend to be drawn together. Both types of features are important for fitting the AAM model to unseen pictures. The right choice of parameters is important for optimal determination of expression. A more detailed discussion is given in section VII.

C. Relevant facial features to indicate expressivity

Facial expressions are defined by the dynamics of the individual facial features. Psychophysical experiments [9] indicate that the eyes and the mouth are the most relevant facial features in terms of facial expressions. Experiments show that, in some cases, one individual facial region can entirely represent an expression. In other cases, the interaction of more than one facial area is needed to clarify the expression.

A thorough description of the eye area is provided using an AAM model, as described in our previous work [10]. A detailed AAM lip model, including a hue filtering is proposed by us for the lips area in [11]. Each of these independent models is used now for expression recognition.

Our work shows that on average, the scores obtained for the eyes shape represent 70% of the information contained by the entire face shape and the mouth is also an important emotion feature carrier for certain emotions, e.g., surprise, with around 60% independent contribution. Further, when combined and processed together emotion decoding accuracies increase. Results of expression recognition for eyes, lips, and faces are summarized in Tables II-III of Section VI-B.

D. Component-based AAM representation of expressivity

Component-based AAM [5] is an approach that benefits from both the generality of a global AAM model and the local optimizations provided by its sub-models. In addition to our global model, we use component models of the mouth and two eye-models.

In summary, the component-based algorithm is as follows. Two sub-models are built, one for the eye region and one for the lips. The eye sub-model is then derived in a left and, respectively, a right eye model. At each iteration, the sub-models are inferred from the global model. Their optimums are detected by an AAM fitting procedure. Then the fitted sub-models are projected back into the global model.

IV. Improved Component Eye Models

A. Extension of the AAM eye model

The initial eye model developed in chapter 5 is based on the standard formulation of AAM [8]. The model offers a detailed analysis of the eye region in terms of degree of eyelid opening, position of the iris, and shape and texture of the eye. The model is designed to be robust to small pose variation. Blink and gaze actions were also modeled in [10], [12].

However, when we tested the model on general conditions not included in the training set, it failed in 60% of cases, cf. section 5.1.4 of [12]. Its challenges include unseen head pose, occlusions of one or more components of the eye model, or difference in expression between the two eyes such as "winking" where one eye is open, the other closed. This is explained by the limitations of a standard AAM. Although AAMs are powerful tools for image interpretation, they present several limitations when used as global appearance models. This formulation, despite all its advantages has the limitation of constraining the model to the variations learned during the training phase. As we created a global model for the two eyes together, we subsequently realized that this constraint does not allow the two eyes to deform independently.

A visual explanation is provided by *Figure.3* which presents two types of challenges: in-plane-head-rotation and independent actions of the eyes, i.e., winking. In the training set we included pictures with the two eyes open or closed. However, as we modeled both eyes together, the model cannot adapt to situations such as one eye open, while the other is closed.

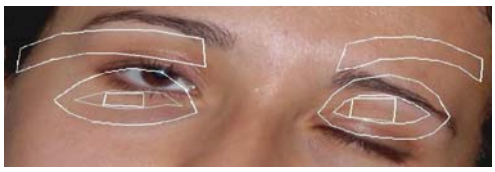


Figure 3:. Example of poor fitting for winking in-plane-rotated eyes.

The ***component-based AAM***, as described in [13] and [14], offers a solution to this particular drawback of AAM, i.e., constraining the model components to global variations. These authors propose a face model that combines a global model with a series of sub-models. This approach benefits from both the generality of a global AAM and the local optimizations provided by its sub-models. We now adapt this approach to independently model the eye regions within a face. In the following sections, we propose two versions of the component-based AAM adapted for the eye region. We test both versions and then determine the best of these approaches for our application of facial expression recognition. This will then be used in our improved component model described in section VI.

B. Component-based AAM formulations

In a first stage, we adapted the component-based AAM for the eye-region using two different approaches. For the first approach the open-eye and closed-eye states are modeled separately; for the second approach we model each eye separately, retaining the mixed (overlapping points) open/closed information for each eye region. Using these two

distinct approaches we also hope to learn more about how AAM models behave under different training constraints.

1) Separation of open and closed eye states

In our first approach, we separately modeled the two-eye appearance for open or closed eyes. We started from the hypothesis that mixing open and closed eye shapes or textures can introduce errors into the model. The procedure is as follows. We firstly create a global model using both eyes, open and then with both eyes closed. The model is refined with two sub-models modeling only some particular features of both eyes, meticulously annotated as shown in *Figure 4*. A first sub-model represents components of open eyes, i.e. inner eyelid and iris. A second sub-model represents components of closed eyes, i.e., inner eyelid and outer eyelid.

In the case of closed eyes, the inner eyelid is not composed by overlapped points, but represented as a straight line. The eyebrows are included only in the global model for better eye location. They are considered superfluous for local modeling stages, as the eyebrows are mostly necessary for accurate eye location. In these stages, the eye location is believed accurate, coming from the initial global stage.

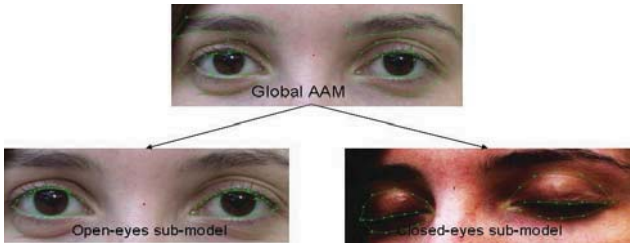


Figure 4: Annotation for the global AAM and for the two sub-models: for open and closed eyes.

The fitting process is represented in *Figure 5*. We firstly match the global AAM which gives us a rough eye modeling. Then, a blink detector is applied, determining if the eyes are open or closed; the blink detector is described in [10] and [12]. Next, the corresponding sub-model is applied, the closed-eyes sub-model if the detector indicats a blink, otherwise the open-eyes model. This local sub-model provides a more accurate match to the eye-region.

Only one of the open or closed sub-models is used in this fitting process, saving computational time. Another advantage is the accuracy of the shape annotation, as closed-eye shape is no longer obtained from open-eye shape. In consequence, fewer errors are introduced in the appearance model. The blinking information is still extracted thanks to the global model, but the accuracy of the shape is refined by the relevant sub-model.

2) Independent modeling of the left and the right eyes

The idea of our second approach started from our idea to model each eye independently. A two-eye global model is necessary for an accurate location and initial modeling. Moreover we would like to be able to better optimize the matching of the two eyes independently.

A global model is created using both eyes open and closed. Then a separate sub-model is created describing a single open or closed eye and the different variations in between the two states. The two models, i.e., one global model and one sub-model, are trained and are independently generated using the same procedures, as described in section III. The interested reader is also referred to [12] for a more detailed description.

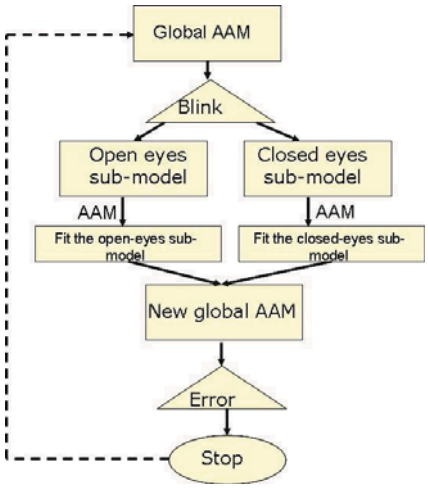


Figure 5: The fitting algorithm for the open/ closed eyes sub-models.

One valuable aspect of the eye-model is the symmetry between the two eyes. This characteristic permits, starting from one eye model, e.g. left eye, to auto-generate an equivalent model for the other eye, e.g., right eye, respectively, simply by mirroring the data. An additional advantage is that we have less memory space requirements, as only one set of eye-model characteristics needs to be stored. This can be very important when considering the implementation of such techniques in low-cost consumer electronic devices such as gaming peripherals.

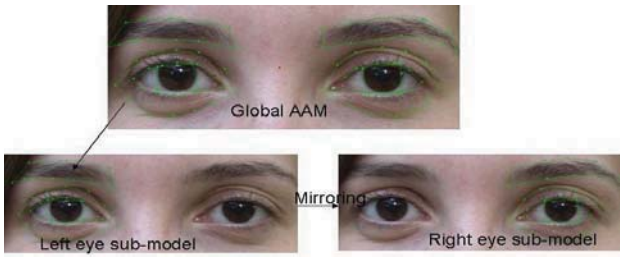


Figure 6: Examples of annotation for the global model, for the local sub-model and its mirroring in order to obtain the right eye.

The advantage of this modeling approach is that it permits that the two eyes find their optimal position and shape independently. There are situations, especially when dealing with large pose variations, plane rotations, occlusions, or strong facial expressions, when the 2-D projection of the eyes loses the property of global symmetry. An example of such a situation was presented in *Figure 1* above where we exemplified poor fitting of the global AAM.

In *Figure 6* we describe the fitting algorithm adapted for this component-based version. Initially the global AAM is fitted, roughly locating the two eyes. The points of the global model which correspond to the sub-models are firstly extracted to form the two shape vectors, providing a good initialization for each sub-model. At each iteration, the sub-model is then inferred from the global model, its optimum is detected by an AAM fitting procedure, and the fitted sub-model is projected back into the global model providing a refined initialization for

that model. Another projection of the global shape vector onto its principal components space is required. This step is necessary in order to constrain the two independently obtained eyes such that they remain within the limits specified by the global model. In the last step of the fitting process, the fitting error for this refined global model is compared with the original global model fitting error and a decision is taken to use the global model with the least error. This process can be repeated until a convergence criteria for the global and local errors is achieved for each of the models within the component AAM framework. However a single step process will generally achieve sufficient convergence if a Viola-Jones face detector has been used to provide the first initialization for the global model. A detailed process flowchart is provided in *Figure 8* below and the interested reader can find a more detailed description in [12].

In *Figure 7* we present some comparative examples of fitting: the first column shows the effects of fitting the standard, holistic AAM eye model; the second column shows the effect of fitting the independent-eyes sub-models without applying constraints from the global model; finally the third column shows the results using the independent-eyes models combined with constraints from the holistic model.

Figure 7: Fitting the standard AAM model; b. fitting the left/right eye sub-models without refitting the global model; c. fitting the left/right eye sub-models and refitting the global model.

C. Comparison of proposed component-AAM approaches

We proposed two different versions of adapting the component-based AAM for the eye region. The first version locally models both eyes, separating the open and closed eye situations. The second version independently models each eye, but it simultaneously includes open and closed eyes. We tested both versions, with a view to using the more effective model in a range of applications. Both versions were trained using the same training set as for the standard eye model developed in [10] and [12]. A dataset of 70 pictures were carefully selected to provide a high degree of variability in terms of subject individuality, pose, illumination, and facial features. Samples of the training set can be seen in *Figure 7* and thumbnails of the entire dataset are available as an appendix in [12].

In order to compare these two modeling approaches, we used the same test set as the set used for testing the standard AAM eye model. A detailed description of testing the component-based AAM compared with the standard AAM formulation is given in chapter 7 of [12].

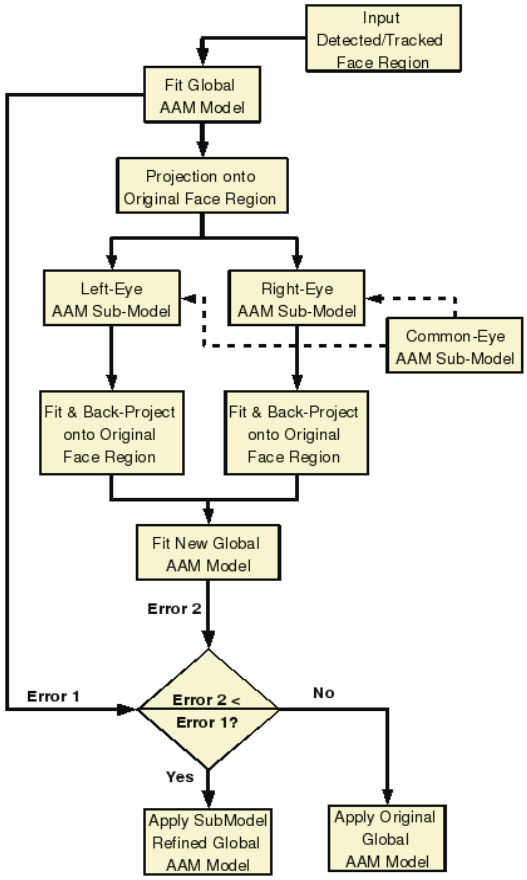


Figure 8: Fitting algorithm for component based AAM eye model.

D. Conclusions & findings for the component eye-model

For now, we conclude that the version of the component-based AAM which uses independent left and right eye-models proved more useful and effective across a range of applications. In this version, we use as local sub-models single-eye models, adapted for open and closed eyes. This approach proved capable of accurately determining the eye parameters, particularly in cases of facial expression, where the two eyes deform independently. It is especially robust to pose variations and to in-plane rotations.

The first version that we proposed, besides providing a poorer fitting in most cases, has the additional drawback that it is critically dependent on the accuracy of blink detection. This drawback is explained in *Figure 9*. In the first picture the blink detector fails by indicating open eyes. This causes a catastrophic failure of the algorithm, as the open eye sub-model is chosen for fitting. Not only does it attempt to match the closed eyes with the open-eye model but the global alignment of the eye-region is completely lost.

In the second case where independent eye-models are used in the component-based AAM, each eye is fitted independently. Thus even if the global model fails by matching open eyes, the sub-models still correctly match the eye image

with closed eyes because each local model contains both open and closed-eye training data. This situation is represented in the second picture of *Figure 9*.

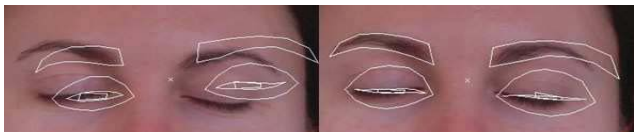


Figure 9: Comparison of the two proposed component-based versions: the two-eyes sub-model vs. the single-eye sub-model.

The second version that we proposed, the component-based eye AAM independently modeling the left and right eye, is the one employed in our work on expression recognition described in the remainder of this paper.

E. A direct quantitative comparison of Modeling techiques

After visually inspecting the results, a quantitative evaluation of the proposed model is performed on representative examples of the three test sets. The quantitative evaluation of a model performance is realised in terms of boundary errors, calculated as Point-to-Point (Pt-Pt) shape error, calculated as the Euclidean distance between the ground-truth shape vector and the converged shape vector inside the image frame:

$$Pt-Pt=\frac{1}{n}\sum_{i=1}^{n}((x_i-x_i^g)^2+(y_i-y_i^g)^2)^{1/2} \quad , \qquad (3)$$

where the index g marks the ground truth data, obtained in our case by hand annotation.

Another type of error can be calculated, namely the Point-to-Curve shape error. It is calculated as the Euclidian norm of the vector of distances from each landmark point of the exact shape to the closest point on the associated border of the optimized model shape in the image frame :

$$Pt-Crv=\frac{1}{n}\sum_{i=1}^{n}min((x_i-r_x^g(t))^2+(y_i-r_y^g(t))^2)^{1/2} \quad , \qquad (4)$$

The mean and standard deviation of Pt-Pt and Pt-Crv are used to evaluate the boundary errors over a whole set of images. In *Figure 10* we present the histogram of Pt-Pt shape errors calculated with respect to manual ground-truth annotations. The ground-truth annotations represent hand-annotated shapes. In these tests we compare the standard AAM formulation, the component-based method, when omitting its last stage, i.e., the global fitting, and the component-based AAM with global fitting. The initialization provided from the face detection step is used as benchmark.

From the figure it can be observed that the boundary errors for the tested fitting algorithms are concentrated within lower values as compared to the initial point generated from the detection algorithm, showing the methods improvement for eye location. Furthermore, it can be noticed that the shape boundary errors are concentrated within lowest values, indicating that the full component-based AAM performs the best in terms of fitting accuracy, thus resulting in a clear improvement over the initial position, as well as over the other fitting algorithms. More results for the component-based AAM can be found in chapter 5 of [12]. The advantages of using a component-based initialization and fitting is mirrored in the higher accuracies obtained for eye tracking, blink, or gaze detection.

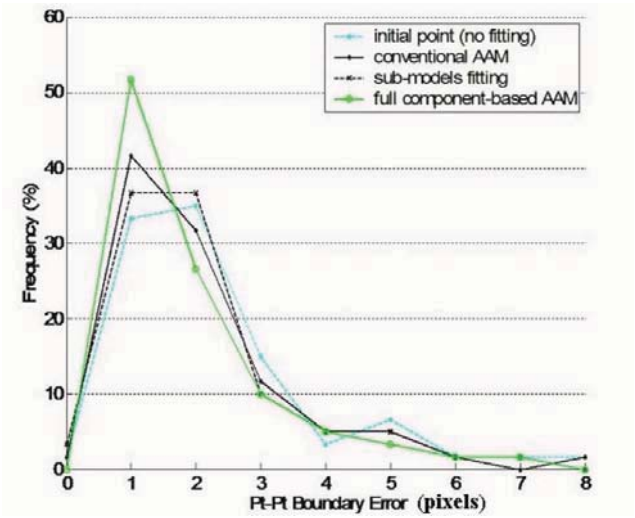


Figure 10: The histogram of the boundary error for the three algorithms: conventional AAM, sub-model fitting and the component-based AAM.

V. IMPROVED LIPS MODEL

In [11] and [12] we developed a lips model, based on the standard AAM formulation. It was however observed that the model failed our tests on unseen images in a proportion of 80%. We concluded that the main cause for this failure is a poor initialization of the model. This is mainly due to the weak contrast between the color of the lips and the surrounding skin. We note that there is significant overlap in color ranges between the lips and skin regions of the face [15]. In this section, we propose an improved version of our AAM lips model. We improve the standard AAM formulation by applying a pre-processing step that offers a more accurate initialization of the lip region. The overall approach, embracing the initialization and the AAM modeling, is described in the remainder of this section. We have tested the performance of our lip model, by developing two consumer applications: a lip tracker and a smile detector. More thorough experimental testing of this model is documented in chapter 7 of [12].

A. initializaiton of the lip region by chrominance analysis

The lips model requires a strong initialization in order to achieve an accurate fitting to unseen images. Consequently we propose a pre-processing method that can provide such a robust initialization. The most valuable information related to lips is their red color, although red varies with respect to individuals, make-up, illumination etc. Therefore, by filtering the red lip color from the face region, we should be better able to identify the global shape of the lip region. This approach is based on the work of Pantic et al [16]-[18], and is adapted for our AAM models. Firstly, the input image is transformed into the HSV colour space, as hue representation is less affected by variations in illumination [16], [17]. This colour space brings invariance to shadows, shading, and highlights and it permits using only the hue component for segmentation. Then the object of interest is filtered into the red domain, by applying the following hue filter [19]:

$$f(h) = \begin{cases} \dfrac{1 - (h - h_0)^2}{w^2} & |h - h_0| \le w \\ 0 & |h - h_0| > w \end{cases} \quad (5)$$

where h is the shifted hue value of each pixel so that $h_0 = 1/3$ for red color. Note that h_0 controls the positioning of the filter in the *hue* color plane and w controls the color range of the filter around its h_0 value. As the color for the lip region varies with respect to person identity, light conditions, make-up etc., the challenge is to find optimal parameters for the filter. Although an optimal solution would be an adaptive hue filter as in [19], the simplified solution adopted in our case is to find the optimal parameters for predefined conditions, e.g., for a specific database.

A statistical analysis was performed on our training set to study the variation of lip colour between individuals and the differences caused by varying illumination on lip pixels for the each picture was investigated. We noted that standard deviation does not vary much from picture to picture, as the pictures belong to the same database, with controlled acquisition conditions. The filter coefficients are chosen after performing an average on the mean and on the standard deviation for all pictures. The overall mean is 0.01 and it corresponds to the positioning of the filter h_0. The overall standard deviation approximating the filter width w is 0.007.

In our use of these techniques, after determining the parameters of the filter and performing the actual filtering operation, each image is binarized, as shown in *Figure 11*, using a predefined threshold. The value of this threshold was set to 0.5, determined after a trial-and-error testing. Morphological operations such as closing, followed by opening, can be used in order to fill in gaps and to eliminate pixels that do not belong to the lip region. After the lip region is determined, its center of gravity (COG) is calculated. This point is then used as the initialization point for the AAM fitting algorithm.

Figure 11: Lip region pre-processing: a. original image, b. after the hue filter, c. after the binarisation.

B. The final formulation of the AAM lip model

Our lips modeling is composed of two main steps: an initialization step and a modeling step. Firstly, before the lip feature can be extracted and analysed, a face must be detected in an image and its features must be traced. The face is inferred from the Viola-Jones face detector applied on the input image. Then, our region of interest (ROI), i.e. the lip region, is deducted from the rectangle describing the surroundings of the face. Thus the ROI is reduced to the lower third on the y axis, while 3/5 of the face box is retained on the x axis, as shown in *Figure 12*. A hue filter is then used to provide an initial location of the lip region within this ROI.

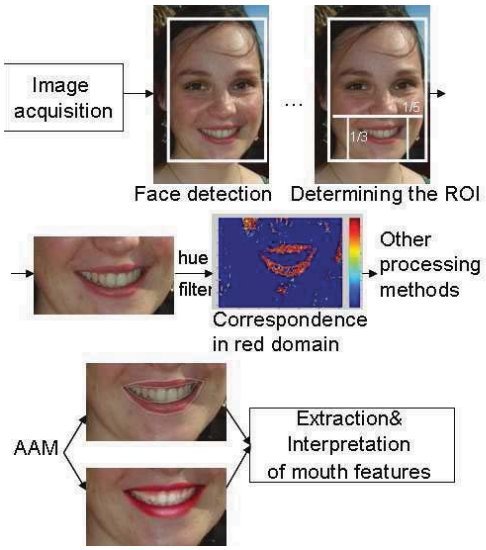


Figure 12: Lip modeling system overview

In a second step, AAM is applied in order to perform a refined detection and to determine detailed lips features. The starting point for the algorithm is the COG of the hue filtered ROI. The AAM adjusts the parameters so that a synthetic example is generated, which matches the image as closely as possible, as seen in *Figure 12*. Optimal texture and shape parameters are determined using the standard AAM lip model. In consequence information regarding lip features, such as its curvature, degree of opening, or its texture, can be readily determined.

VI. The Full Component-AAM Face Model

The low expression classification/recognition rates obtained in the previous experiments, when a conventional holistic AAM face model is used, serve to confirm the limitations of this approach. This poor performance of the conventional AAM when used for expression analysis can be understood because this model is based on a global approach and is thus more sensitive to changes in configuration than to changes in local features. Such a representation cannot be sufficiently flexible to permit adaptation to wider ranges of facial variability, such as the deformations which are present in the majority of distinctive facial expressions.

From a practical perspective we note that it is not possible to include all possible variations of shape and texture in the training set. The overall degrees of freedom inherent in the model are restricted by the number of model parameters. Or to put this another way, a model which did incorporate practically all possible facial variations in its training set would have a correspondingly large set of model parameters making it unwieldy and impractical, particularly for applications in consumer electronics or for implementations in low-cost gaming peripherals. Thus any practical model we construct must necessarily restrict its potential variations. To achieve real improvements in our classification it would seem that we need more than a single holistic AAM model.

In this section, we propose to adapt the component-based AAM [5] for facial expression analysis. This approach benefits from both the generality of a global AAM and the

optimizations provided by local sub-models. It adds degrees of freedom to the global model, by accounting for individual variations of facial features. The principles of component-AAM were previously explained in detail in section IV, for the eye features of a face. For a more comprehensive treatment the interested reader is referred to [12].

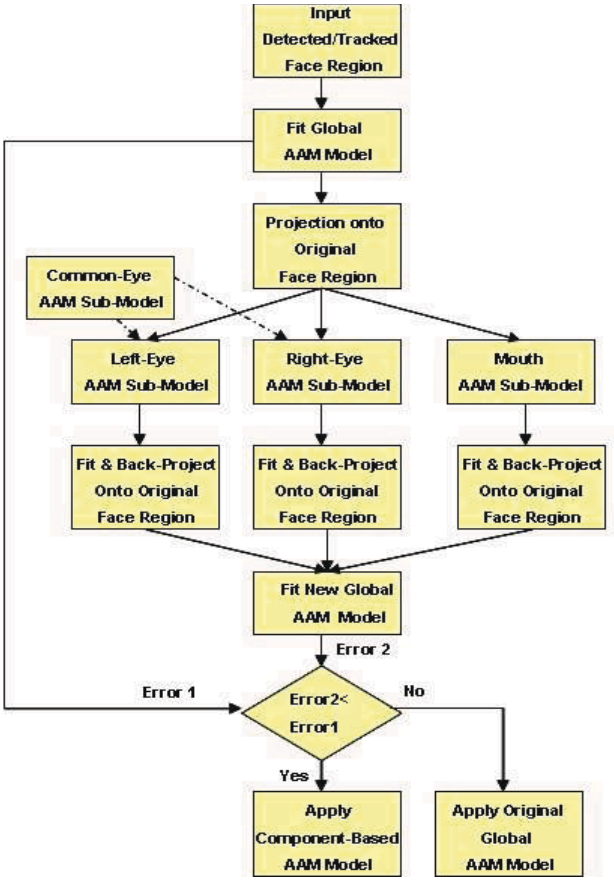


Figure 13: Full component-AAM Face Model incorporating both improved Eyes and Lips models in a single component-AAM framework.

Now *Figure 13*, explains its further adaptation for the entire face region, using two sub-models: both an eye model and a lips model. These sub-models are based on the models just described in sections IV and V above. As for the eye region model, at each iteration, the relevant sub-models are inferred from the global model. Optimal fittings are determined through an AAM fitting procedure, based solely on the shape parameters. Then the fitted sub-models are projected back into the global model.

In *Figure 14* below we describe one practical example of the benefits of a component-based representation, on an image containing both pose and expression variations. This shows how the sub-models improve fitting of the global model to the entire face region, which in turn improves the alignment of each sub-model with the local features which are some important to accurate expression recognition. The first is the result of a conventional AAM; the second picture represents the fitting of the AAM sub-models, while the third picture depicts the component-based result.

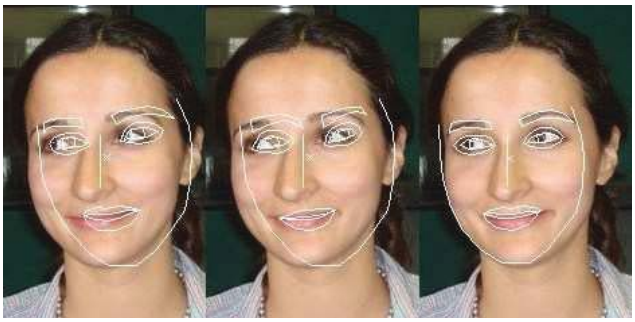


Figure 14: Example of shape fitting on an unseen picture with expression and pose variations.

Now, in order to quantitatively evaluate the overall performances of a component-based AAM, we must next measure its accuracy in terms of expression classification/ recognition rates. But before we can do this we first consider which AAM features are most relevant for expression analysis. The results, presented in the next section, provide some useful practical details for researchers working in this field.

VII. Relevant Feature Extraction for Facial Expressions

Feature selection consists of keeping the most relevant features for classification and discarding irrelevant or redundant features. The quality of the extracted features plays a key role in their classification. Two types of features can be extracted when applying an AAM: geometric features and features depicting the texture. If required, *appearance* parameters can be obtained by applying PCA on the concatenated geometric and texture parameters. Both types of feature are important when fitting an AAM model to unseen images. A question we now to ask is which of these parameters are more significant for determining and recognizing facial expressions?

A. Features for FER – literature review and first thoughts

In chapter 4 of [12], the published research on this topic is summarized in some detail. It was concluded that opinions are divided in the literature. Researchers agree that shape features have a large role to play in facial expression recognition [20], [21], but some insist that both shape and texture features are required [22] in order to obtain satisfactory results.

Let us begin here with a qualitative analysis. On one hand, the skin texture of the facial region exhibits only slight changes during facial expressions. These minor changes arise due to local lighting effects, blushing, or wrinkling of the skin. Such changes can be considered relatively invariant when compared with the more obvious changes of shape observed for different facial expressions. Such geometrical features are very directly related to expressions. As examples, when surprised, we widely open the eyes and eventually the mouth, resulting in an elongated chin; when sad, we often blink; when angry, the eyebrows are usually drawn together. Thus, we concur with the hypothesis that shape parameters are the most significant features in facial expression decoding. We next describe some experiments conducted to verify our initial hypothesis [12].

B. *Quantitative determination of AAM parameters for FER*

While we have just argued that the shape parameters of an AAM face model contain the greatest "information density" for determining facial expression, there is still likely to be a significant amount of redundant information within this parameter set. Contra-wise, there may also be significant useful information contained in a small subset of the texture parameters. Ideally we would like to be able to further refine our use of AAM model parameters to achieve a more optimal set of parameters which are more closely tuned to the requirements of facial expression analysis.

Our approach to achieve this was to investigate the particular shape parameters that best represent facial expressions. We used both our own customized database and images from the FEEDTUM database. A set of 42 images were used to train the improved component model, 21 from each database. This training set was picked to include images of many different individuals, with varying facial poses, and facial expressions. Corresponding recognition tests were performed on a set of 50 unseen images from our database and 100 from FEEDTUM. A Euclidean NN based classifier was used for measuring each category of facial expression and, after discarding the lowest order shape parameter which is dominated by the facial pose information, the 17 lowest order shape parameters of the AAM model were used as features.

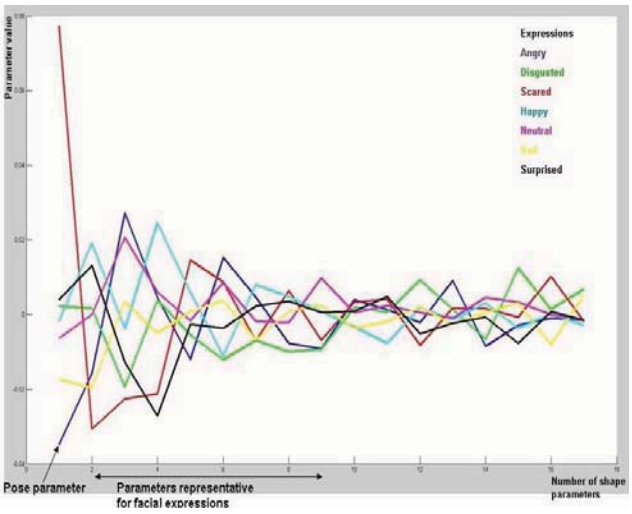


Figure 15: Mean over the shape parameters over each of the seven universal emotions.

The results are presented in *Figure 15* where we have plotted the expression recognition rates versus the number of model parameters employed. This shows the mean over the AAM shape parameters for each of the six universal facial expressions and the neutral one of the training set.

A maximum of 17 shape parameters was used in total. Then, the number of parameters is reduced, eliminating in turn the parameter with the smallest variation across the test set of images. As the number of parameters is reduced we are left with the parameters which exhibit the widest variation. These results are presented in *Figure 16* where FER accuracy is plotted against the number of shape parameters used in that particular test.

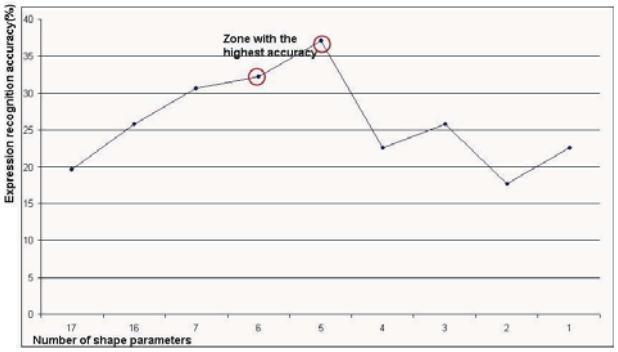


Figure 16: FER accuracy versus the number of shape parameters.

It can be noticed that optimal results are obtained when the five to six most variable parameters are used. As we increase the number of parameters beyond the 6th parameter the model accuracy deteriorates. We conclude that an educated choice of parameters positively affects the system performances. In our case, it is shown empirically that using the first 30-40% of model parameters provides higher expression recognition rates.

C. *A note on the lowest-order shape parameter*

In this experiment we noted that after performing PCA on shape parameters, the information on out-of-plane head pose was encoded in the first shape parameter. This is explained by the fact that the variations caused by pose cause significantly more geometric distortion than the variation caused by differences between individuals or between facial expressions. Consequently, pose variation is uncorrelated to a large extent with other sources and manifests itself in the first-order PCA parameter.

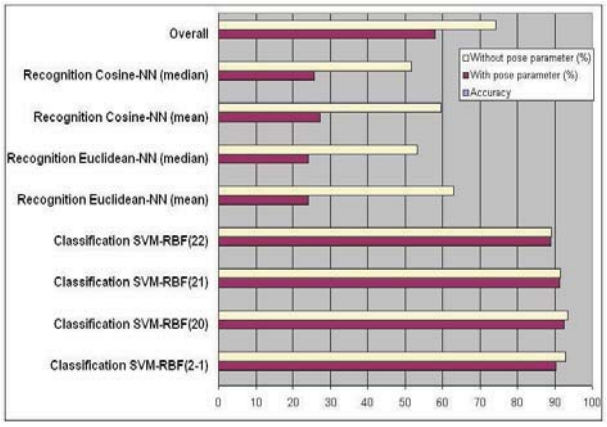


Figure 17: Different FER classification schemes comparing recognition rates with and without the lowest order AAM shape parameter.

The pose parameters are very important for the AAM fitting stage and the subsequent extraction of facial features. The pose variation information contributes to the shape and texture of facial features. However when analyzing facial expressions, the lowest order pose parameter should generally be eliminated, as it will not contain the required facial expression information. This is illustrated in *Figure 17* where we compare a range of FER classification methods each with and without the first-order AAM pose parameter. We note that SVM based methods are more robust than NN methods to pose. Even where the

effect of this pose parameter are negligible it still adds an additional computational burden which is redundant as even for SVM the recognition rates are slightly lower with this parameter included.

Note that this would not be the case if the model was trained using only frontal images – but then the model would not be able to generalize to non-frontal faces.

D. *Conclusions on the significance of AAM features*

Based on the results of these tests we determined that shape parameters did indeed prove to be overall the most valuable features for facial expression decoding accuracy. Other tests confirmed that while shape results are comparable with the results obtained when applying a combined shape and texture model, i.e., when using the AAM appearance features the number of shape parameters is significantly less. In turn the computational requirements both for feature extraction and to subsequently perform classification are also reduced. Thus shape parameters on their own have a demonstrable advantage over approaches based on texture-only and both concatenated and merged shape and texture. These finding were also confirmed by the authors of [21].

It should be remarked that the accuracy rates of classification were not specifically addressed in this series of experiments. Improvements aiming to increase the accuracy of expression classifiers will be presented in a later section.

VIII. Expression Classification and Recognition

The last step in a facial expression recognition (FER) system is expression classification and recognition. In our work two classifiers were compared: SVM [30] and NN. This choice is based on their positive performances in the literature. As we designed them, the classifiers use as input relevant AAM parameters and they present at output the choice between two facial expressions. When dealing with poses, the pose parameters are discarded as described previously.

A. *Defining a fixed set of classes for facial expression*

Facial Action Coding System (FACS), originally developed by Ekman and Friesen in 1976 [22], is the most widely used coding system in the behavioural sciences. The system was originally developed by analysing video footage of a range of individuals and associating facial appearance changes with contractions of the underlying muscles. The outcome was an encoding of 44 distinct action units (AUs), i.e., anatomically related to contraction of specific facial muscles, each of which is intrinsically related to a small set of localised muscular activations. Using FACS, one can manually code nearly any anatomically possible facial expression, decomposing it into the specific AUs and their temporal segments that produced the expression. All resulting expressions can be described using the 44 AUs described by Ekman or a combination of the 44 AUs. In 2002, a new version of FACS is published, with large contributions by Joseph Hager [23].

Ekman and Friesen [24] have also postulated six primary emotions which they consider to be universal across human ethnicities and cultures. These six universal emotions, commonly referred as basic emotions are: happiness, anger, surprise, disgust, fear, and sadness illustrated in *Figure 18*. The leading study of Ekman and Friesen [24] formed the origin of facial expression analysis, when the authors proclaimed that the six basic prototypical facial expressions are recognised universally. Most researchers argue that these expressions categories are not sufficient to describe all facial expressions in detail. However, most of the existing facial expression analysers still use Ekman and Friesen's theory.

We also include the neutral expression, or a face *without expression* as a seventh category.

Figure 18: On each row are represented the six universal facial expressions and the neutral state, as expressed by different subjects (in order, anger, happiness, neutral, surprise, fear, sadness, and disgust).

B. *Expression classification*

The choice of the two classifiers is based on their positive results obtained in the literature. In [8] it is stated and it is proved by experiments for gender classification and face recognition that SVM is typically among the top two classifiers, and the other top ranking classifier is one of the Euclidean-NN rule or the cosine-NN rule.

1) *Nearest Neighbour (NN)-based classifier*

We next present some experimental results with variations on the NN technique to determine its practical utility. In particular we investigate several different types of similarity metric and templates for NN, including the Euclidean, cosine, and Mahalanobis distances. Also, two types of templates can

be employed, based on calculating the mean or the median over the input AAM parameters.

For these experiments each picture in the test set is classified by comparing its parameters with a template corresponding to each expression. The class yielding the minimum distance is selected. The templates for each class are obtained by a mean, or a median, over the shape parameter vector for each expression. The classifiers are of type expression 1/expression 2, i.e., neutral/non-neutral, sad/non-sad, angry/non-angry, disgusted/non-disgusted, fear/non-fear, surprised/non-surprised, and happy/non-happy, and of type expression/non-expression. Considering that we have six expressions and the neutral one, 28 classifiers are obtained, seven for the former type and 21 for the latter.

For our first set of experiments the AAM training and test sets coincide, i.e., there are no unseen images. This is so that we do not bias our results with poor AAM fittings. The Tables NN1-NN4 summarise the classification rates, as for example, in Table 8.2: the system 80% correctly recognises angry from non-angry faces and 75% angry from disgusted faces.

TABLE NN1. EXPRESSION CLASSIFICATION ACCURACIES (%) FOR THE MMI DATABASE (TRAINING AND TEST SETS OVERLAP) WHEN USING A NN WITH A *MEAN* TEMPLATE RULE- AVERAGE OF EXPRESSION CLASSIFICATION 83.67 %.

	A	D	F	H	N	Sa	Su
A	80	75	85	90	75	80	87.5
D		87.5	85	95	92.5	95	95
F			57.5	85	67.5	72.5	82.5
H				87.5	85	95	95
N					80	70	85
Sa						85	87.5
Su							85

TABLE NN2. EXPRESSION CLASSIFICATION ACCURACIES (%) FOR THE MMI DATABASE (TRAINING AND TEST SETS OVERLAP) WHEN USING A NN WITH A *MEDIAN* TEMPLATE RULE - AVERAGE OF EXPRESSION CLASSIFICATION 83.41%.

	A	D	F	H	N	Sa	Su
A	82.5	90	72.5	77.5	77.5	90	85
D		85	90	82.5	90	97.5	92.5
F			67.5	75	75	85	82.5
H				77.5	75	87.5	90
N					90	72.5	85
Sa						87.5	88
Su							85

As can be noticed from these experimental results, a template obtained by averaging the AAM shape parameters outperforms, slightly, a template based on the median approach.

TABLE NN3. EXPRESSION CLASSIFICATION ACCURACIES (%) FOR THE FEEDTUM DATABASE (TRAINING AND TEST SETS OVERLAP) WHEN USING A NN WITH A WITH A *MEAN* TEMPLATE RULE - AVERAGE OF EXPRESSION CLASSIFICATION 93.58 %.

	A	D	F	H	N	Sa	Su
A	100	96.7	93.4	100	96.7	96.7	100
D		80	90	100	70	90	100
F			83.4	96.7	80	93.4	100
H				90	90	100	100
N					100	93.4	100
Sa						80	100
Su							100

TABLE NN4. EXPRESSION CLASSIFICATION ACCURACIES (%) FOR THE FEEDTUM DATABASE (TRAINING AND TEST SETS OVERLAP) WHEN USING A NN WITH A *MEDIAN* TEMPLATE RULE - AVERAGE OF EXPRESSION CLASSIFICATION 89.79%.

	A	D	F	H	N	Sa	Su
A	100	63.4	93.4	100	96.7	90	100
D		73.4	86.7	96.7	70	93.4	100
F			73.4	96.6	66.7	90	93.4
H				90	90	96.7	100
N					83.4	86.7	100
Sa						83.4	100
Su							100

Generalization of the AAM to unseen subjects is tested in a second series of experiments with *leave-one-out* tests, in which images of the tested subject are excluded from training. As might be expected we observe a decrease in the classification accuracies. Tables NN5 and NN6 summarize the corresponding results of these tests. It can be noticed also that the classifier performs better on the FEEDTUM database; the MMI presents more variations in illumination, a fact that affects the AAM fitting and, in consequence, the precisions of facial feature extraction, particularly for unseen images.

TABLE NN5. EXPRESSION CLASSIFICATION ACCURACIES (%) FOR THE MMI DATABASE (TRAINING AND TEST SETS DO NOT OVERLAP) WHEN USING A NN WITH A MEAN TEMPLATE RULE- AVERAGE OF EXPRESSION CLASSIFICATION 62.99 %.

	A	D	F	H	N	Sa	Su
A	83.33	50	63.4	80	50	63.4	76.7
D		60	70	66.7	56.7	53.4	73.4
F			56.7	83.4	56.7	80	63.4
H				70	73.4	53.4	80
N					60	56.7	83.4
Sa						60	86.7
Su							63.4

TABLE NN6 EXPRESSION CLASSIFICATION ACCURACIES (%) FOR THE FEEDTUM DATABASE (TRAINING AND TEST SETS DO NOT OVERLAP) WHEN USING A NN WITH A MEAN TEMPLATE RULE - AVERAGE OF EXPRESSION CLASSIFICATION 66.94%.

	A	D	F	H	N	Sa	Su
A	83.33	50	63.4	80	50	63.4	76.7
D		60	70	66.7	56.7	53.4	73.4
F			56.7	83.4	56.7	80	63.4
H				70	73.4	53.4	80
N					60	56.7	83.4
Sa						60	86.7
Su							63.4

We conclude that a template obtained by averaging the AAM shape parameters outperforms, slightly, a template based on the median approach. Thus in the experiments that follow, a template obtained by averaging the shape parameters is used for NN. From these experiments on NN-based expression classification we determined that Euclidean and cosine distances perform almost equally well, but there is a slight advantage for the Euclidean distance from the point of view of a more consistent level of performance. Thus we recommend use of the Euclidean distance as being representative of the optimal NN technique.

2) Support Vector Machines (SVM) Classifier

We next present our experiments on SVM classifiers to determine the optimal settings for SVM when applied to expression classification. Here we search for the best kernel function, and also for the optimal settings for each function. Again the interested reader is referred to [12] for background details on this work. Two potential kernel functions are investigated: the residual basis function (RBF) and the polynomial function. Best grade for a polynomial kernel and the optimal δ values for an RBF kernel are searched.

In the first part of the experiment, seven SVM classifiers of type expression 1/expression 2 are built to distinguish the six universal expressions and the neutral one. The polynomial kernel, with order from 1 to 6 and the RBF kernel with δ from 2^{-2} to 2^{6} results are investigated. The results for the MMI database are detailed in Table SVM1. The optimal kernel function proved to be RBF, with δ fixed on small values. It is our preferred choice based on the accuracy and the consistency of its results. In a second series of experiments we sought to confirm the previous conclusions, this time using 28 classifiers, as in our earlier NN trials. The results, presented in tables SVM2 and SVM3, confirm that RBF with δ fixed on small values achieves the highest classification rates.

We conclude that the RBF kernel function with δ fixed on small values gave the best results. Our findings are confirmed also in the literature, where the RBF kernel is the most common function to be used in expression recognition with SVM. The choice is also motivated by theory. RBF has fewer adjustable parameters than any other commonly used kernel and it thus has less numerical complexity.

TABLE SVM1 EXPRESSION CLASSIFICATION USING SVM FOR THE MMI DATABASES, WHEN USING DIFFERENT KERNELS.

Classifier	Polynomial-grade					
	1	2	3	4	5	6
H/non-H	88.7	78.7	82.7	65.4	59.4	51.4
D/non-D	56	46.7	46	46	41.4	40.7
Su/non-Su	74.7	52.7	51.4	50	46	44.7
A/non-A	51.4	47.4	47.4	46	53.4	53.4
Sa/non-Sa	74	58	54	47.4	45.4	43.4
F/non-F	78.7	72	71.4	64.7	60.7	54
N/non-N	62.7	56	59.4	54.7	50.7	45.5
Average	69.5	58.8	58.9	53.5	51	47.6

Classifier	RBF						
	2^{-3}	2^{-2}	2^{-1}	2^{0}	2^{1}	2^{2}	2^{3}
H/non-H	78.7	82	82.7	82.7	90	92.7	92.7
D/non-D	63.4	58	60.7	55.4	70.7	79.4	62.7
Su/non-Su	40.7	46.7	61.4	71.4	72	71.4	68.7
A/non-A	71.4	71.4	40	72.7	66	70	48
Sa/non-Sa	66	64	61.4	62.7	69.4	68.7	70.7
F/non-F	69.4	69.4	70	70.7	72.7	72	72
N/non-N	38.7	42.7	58.7	62.7	61.4	79.4	59.4
Average	61.2	62.1	62.2	68.4	71.8	76.3	67.8

TABLE SVM2. ACCURACY (%) OF EXPRESSION CLASSIFICATION ON FEEDTUM FOR THE AAM SHAPE PARAMETERS WHEN APPLYING A SVM CLASSIFIER FOR RBF 2^2, AVERAGE OF EXPRESSION CLASSIFICATION 69.43%.

	A	D	F	H	N	Sa	Su
A	70	63	70	66.7	70	70	73.4
D		79.4	63.4	70	73.4	53.4	70
F			72	66.7	56.7	80	53.4
H				92.7	73.4	60	66.7
N					79.4	53.4	80
Sa						68.7	76.7
Su							71.4

TABLE SVM3. ACCURACY (%) OF EXPRESSION CLASSIFICATION ON MMI FOR THE AAM SHAPE PARAMETERS WHEN APPLYING A SVM CLASSIFIER FOR RBF 2^2, AVERAGE OF EXPRESSION CLASSIFICATION 62.59%.

	A	D	F	H	N	Sa	Su
A	62.5	55	65	52.5	52.5	55	65
D		65	65	52.5	60	62.5	77.5
F			55	62.5	60	52.5	77.5
H				57.5	57.5	65	75
N					52.5	57.5	77.5
Sa						57.5	75
Su							80

C. Expression Recognition

Now, after we trained a set of classifiers to discriminate between two facial expressions, we would like to be able to associate a human face with one of the six universal expressions or the neutral one. This process is known as facial expression recognition [25]. We will again compare NN and SVM based techniques, but this time adapted for a multi-class decision framework.

We begin by presenting the results from a series of experiments to investigate the performances of NN in expression recognition. Then we describe techniques to adapt the binary SVM to a multi-class problem, such as expression recognition. Results are presented for some of these approaches, and we make an empirical determination of the most suitable approach.

1) Nearest Neighbour

The first set of experiments was performed on FEEDTUM, MMI and our own database using the Euclidean-NN and the optimal number of 7 shape parameters. Between 20 and 30 pictures were used for training and 150 test images were drawn each from FEEDTUM and MMI with an additional 50 from our own database. Table AAM1 summarises expression recognition rates when a conventional AAM facial representation is used. It can be noticed that the highest recognition rates are obtained on the MMI database. This is explained by the fact that MMI has a better image resolution and less variation in illumination that the other datasets we used. And as we would expect, the poorest results were obtained on our own database which contains the strongest pose and illumination variations.

These results serve mainly to confirm the relatively poor performance of conventional AAM.

TABLE AAM1-SUMMARY OF THE SYSTEM ACCURACIES (%) FOR RECOGNISING EMOTIONS WITH EUCLIDEAN-NN.

Database	Recognition rate (%)
FEEDTUM	35.71
MMI	40.88
Our database	22.66
Overall	33.08

2) Multi-class SVM

By their nature SVMs are binary classifiers. However, there exist strategies by which SVMs can be adapted to multi-class tasks, such as One-Against-One (1A1), One-Against-All (1AA), Directed Acyclic Graph (DAG), and SVMs in cascade structures. In the following experiments we only exemplify the 1A1 and the cascade structures. The choice is based on their simplicity and their good results presented in the literature [26], [27]. The following experiments use the same test inputs as the NN tests presented in table AAM1.

For the first part of this experiment, we employed the 1A1 approach. Altogether 21 classifiers are applied for each picture in our test-bench. A general score is calculated for each picture. The "recognized" facial expression is considered to be the one which obtains the highest score. As an example, to calculate the score for a "happy face" we apply: happy/fear; happy/sad; happy/surprised; happy/neutral; happy/angry; and happy/disgusted. Every time that a happy face is identified a counter is incremented that represents its score. The results are summarised in Table AAM2.

TABLE AAM2-EXPRESSION RECOGNITION ACCURACIES FOR 1AA-SVM ON THE FEEDTUM AND MMI DATABASES.

Emotion	FEEDTUM (%)	MMI (%)
Surprise	71	76.1
Fear	66.3	62.5
Happiness	65.4	62.5
Anger	64.8	56.8
Neutral	62.8	60.4
Sadness	61.5	59.7
Disgust	57.6	64
Overall	64.2	63.15

In the second part of the experiment, another alternative to extend the binary SVM to a multi-class SVM is investigated. A cascaded structure consisting of the most effective six classifiers of the seven which classifying the six universal expressions and the neutral expression are used. The workflow and the corresponding results are summarized in *Figure 17*. The figure shows the recognition rate after each stage of the cascade, e.g. our system correctly recognizes a surprised face with a probability of 84.5% or an angry face with a probability of 70%.

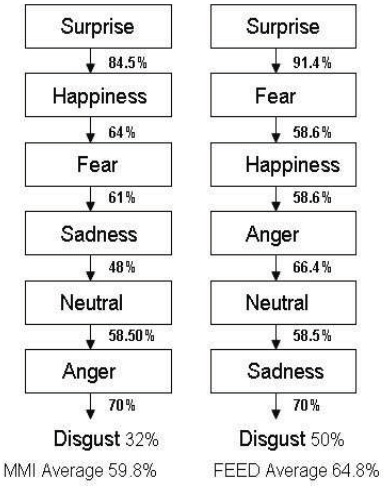


Figure 18: Performances of expression recognition of SVM classifiers in a cascade structure, for MMI and FEEDTUM databases

D. Conclusions on classifiers performances

In this section we analyzed approaches and corresponding results for expression classification and recognition in still images. Two classifiers, namely SVM and NN, were compared. After performing a series of five experiments we conclude the following:

- Overall, happiness proved to be the most recognisable expression, followed by surprise. These observations can be explained by the fact these particular expressions affect the

shape of the facial features more than other expressions. Note the open mouth and raised mouth corners - these expressions are followed by anger, sadness, disgust, neutral state, and fear.

- The results prove system behavior is consistent across subjects of both genders and several races and ages.
- SVM is more effective than NN as a classifier both w.r.t. higher classification/recognition rates and better consistency of the results
- Best results for SVM were obtained when using RBF kernel function with δ fixed on small values.

Best results for NN were obtained when using the Euclidean distance and a template obtained from averaging the shape parameters as a metric for classification. These settings are also to be used in our next series of experiments.

IX. Final Results and Applications Using Our Model

A. FER compared across different modeling strategies

Towards the end of the previous section we presented results for both NN and SVM FER using a conventional AAM model for feature extraction. As expected the results of both approached were quite disappointing. In this section we present a summary of detailed comparisons across different modeling strategies for both NN and SVM techniques. As before, the interested reader is referred to [12] for additional details.

TABLE NN-Final - System accuracy (%) for classifying facial expressions using eye-area, mouth area, face modeled with a conventional or component-based AAM, with Euclidean-NN

Classifier	FEEDTUM			
	Eyes	Lips	Face	Comp
Surprised/ Disg. (%)	44.37	55.87	85.83	83.33
Surprised/ Happy (%)	28.75	69.37	88.33	83.33
Happy/Sad (%)	41.87	46.25	68.33	82.22
Surprised/ Sad (%)	33.12	51.66	93.33	83.33
Overall (%)	37.02	55.78	83.95	83.05

Classifier	MMI			
	Eyes	Lips	Face	Comp
Surprised/ Disg. (%)	73.12	76.66	85.62	78.33
Surprised/ Happy (%)	68.75	75	86.25	79.16
Happy/Sad (%)	72.5	50	73.12	77.5
Surprised/ Sad (%)	69.37	52.5	77	73.33
Overall (%)	70.93	63.54	80.49	77.08

The expression classification and recognition methods are tested on the specialized databases FEEDTUM [26], and [27] and MMI [18], while their accuracy against pose variations is tested on pictures collected especially for this experiment. The results obtained from our tests, suggest that the system is robust in dealing with subjects of both genders. Also, it is independent of race and age.

Here the contributions of each of the AAM sub-models to expression analysis are evaluated. The results are compared with the results of a holistic AAM face model and a component-based AAM facial representation. *Table NN-Final* summarizes the classification results for a mouth model, two-eye model, a conventional AAM face model, and a component-based face AAM when using NN. In the corresponding *Table SVM-Final* we show corresponding results for the SVM classification scheme.

B. Comparison and discussion of FER results

The classification rate average when using a component-based AAM is of 73.02%, while for a classical AAM is of 69.43%. The results confirm the improvement brought by a sub-models representation although these are not quite as impressive as we would have felt from the initial improvements noted during our studies on the feature extraction and the classifiers for individual facial expressions. From these results it would appear that the SVM is quite good at compensating for the deficiencies of the conventional AAM and has reduced the benefits we would have expected from the use of our improved component-AAM approach.

TABLE SVM-Final - System accuracies (%) for svm classification of emotions a conventional(1) and component-based(2) AAM.

	A		D		F		H	
	1	2	1	2	1	2	1	2
A	70	75	63	75	70	75	66.7	68
D			79.4	73.4	63.4	70	70	75
F					72	70	66.7	73.4
H							92.7	93.4
N								
Sa								
Su								

	N		Sa		Su	
	1	2	1	2	1	2
A	70	75	70	79.4	73.4	68.7
D	73.4	75	53.4	68	70	71.4
F	56.7	68.7	80	83.4	53.4	63
H	73.4	75	60	75	66.7	66.7
N	79.4	68.7	53.4	63	80	83.4
Sa			68.7	63.4	76.7	75
Su					71.4	73.4

X. Application to Computer Gaming Workflows

Our system demonstrates the real-time capability to determine a persons emotions. Such a system suggests a range of applications relevant to computer gaming environments, both single and multi-player.

A. Adaptation of game difficulty based on user emotions

Most computer games allow a user to select between a number of difficult levels. Typically this selection is made only once, at the beginning of a game and cannot be changed without starting a new instance of the game.

We envisage a new game workflow where the emotions presented by a user via their facial expressions are evaluated in an ongoing basis and where certain criteria are met the difficult level of a game will be adjusted upwards or downwards accordingly. In particular we can use the angry, disgusted and sad face expressions as negative indicators suggesting that the current difficult level is making the player unhappy. If a user continues in such negative states then after a couple of minutes it would be important to ease the difficulty level or to provide some hints to achieving the next gameplay goal. Contra-wise if the player is showing too much neutral face then it is likely that the game has become too easy and they are bored. A happy face can be considered as an indicator of the correct level of difficulty as the user is still enjoying the game and presumably succeeding in realizing their objectives.

B. Adapting Game Workflow from User Responses

In addition to the basic determination of difficulty level we have also demonstrated a graded classification of several of these emotions in a real-time embodiment [12]. When combined this enables not only the classification of these expressions but a measurement of the transition from a mild to a more intense expression. Such a metric provides interesting possibilities for adapting the more detailed workflow and storyline of a game.

Here we take the use of facial expressions to the next level suggesting that the actual workflow and storyline of the gaming environment can be adapted according to the emotional state of the game player at critical waypoints or challenges along the game path. In conventional gaming such alterations can only be achieved based on the actual outcomes of a challenge in the game environment. Our techniques offer a means for the game designer to achieve a richer and more detailed interaction with the players state of mind during and immediately after critical junctures in the gaming storyline. immediately after critical junctures in the gaming storyline.

C. Real-time personalized avatars

In a recent computer game, *Little Big Planet,* the game avatars associated with each player can be endowed with rudimentary facial expressions; pushing the up-arrow on the gamepad will generate a smiling face; a second press makes the expression even happier and a few more button-presses and your avatar will have a very silly grin throughout the game! Our concept is more challenging – we propose the dynamic detection of user facial expression which is mirrored by their in-game avatar in real-time.

Such a concept is not unknown in the literature. For example, Y. Fu et al have presented a novel framework of multimodal human-machine or human-human interaction via real-time humanoid avatar communication for real-world mobile applications [28]. Their application is based on a face detector and a face tracker. The face of the user is detected and the movement of the head is tracked detecting the different angles, sending these movements to the 3D avatar. This avatar is used for low-bit rate virtual communication. The drawback of this approach is that the shape of the avatar needs to be specified by the user an forward-backward movement of the user is not detected so the avatar appears as a fixed-distance portrait in the display.

In a companion paper we describe an enhanced face model derived from active appearance model (AAM) techniques which employs a differential spatial subspace to provide an enhanced real-time depth map. Employing techniques from advanced AAM face model generation [31] and the information available from an enhanced depth map we can generate a real-time 3D face model. The next step, based on the 3D face model is to generate a 3D avatar that can mimic the face of a user in real time. We are currently exploring various approaches to implement such a system using our real-time stereoscopic imaging system.

References

[1] Yang, M.-H., D.J. Kriegman, and N. Ahuja, Detecting Faces in Images: A Survey. IEEE Transactions on pattern analysis and machine intelligence, 2002. 24(1): p. 34-59.

[2] P. A. Viola, M. J. Jones, “Robust real-time face detection”, International Journal of Computer Vision, vol. 57, no. 2, pp. 137–154, 2004.

[3] http://www.mathworks.com/matlabcentral/fileexchange/19912

[4] G. Bradski, A. Kaehler, and V. Pisarevski, "Learning-based computer vision with intel's open source computer vision library," Intel Technology Journal, vol. 9, no. 2, pp. 119-130, May 2005.

[5] Zhang and F.S. Cohen Component-based Active Appearance Models for face Modeling, in International Conference of Advances in Biometrics, ICB, Hong Kong, China, January 5-7 2006.

[6] Vishnubhotla, S., Support Vector Classification. 2005.

[7] Dasarathy, B.V., Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. 1991

[8] T. F. Cootes, G. J. Edwards, and C. J. Taylor, “Active appearance models”, Lecture Notes in Computer Science, vol. 1407, pp. 484–, 1998.

[9] M. Nusseck, D. W. Cunningham, C. Wallraven, H. H. Bülthoff, The contribution of different facial regions to the recognition of conversational expressions, Journal of Vision, 8(8):1, pp. 1–23, 2008.

[10] I. Bacivarov, M. Ionita, P. Corcoran, Statistical Models of Appearance for Eye Tracking and Eye-Blink Detection and Measurement. IEEE Transactions on Consumer Electronics, August 2008.

[11] I. Bacivarov, M.C. Ionita, and P. Corcoran, A Combined Approach to Feature Extraction for Mouth Characterization and Tracking, in ISSC, Galway, Ireland, 2008.

[12] Ioana Barcivarov, “Advances in the modeling of Facial Subregions and Facial Expression using Active Appearance Modeling Techniques”, PhD Thesis, National University of Ireland Galway, June 2009.

[13] M.D. Cordea, E.M. Petriu, T.E. Whalen, “A 3D-anthropometric-muscle-based active appearance model, in IEEE Symposium on Virtual Environments”, Human-Computer Interfaces and Measurement Systems, (VECIMS), pp. 88-93, 2004.

[14] H. Choi, S. Oh, “Real-time Recognition of Facial Expression using Active Appearance Model with Second Order Minimization and Neural Network”, International IEEE Conference on Systems, Man and Cybernetics, SMC 06, vol. 2, pp.1559 – 1564.

[15] N. Eveno, A. Caplier, P.Y. Coulon, New color transformation for lips segmentation, Proceedings of IEEE Fourth Workshop on Multimedia Signal Processing, October 2001, Cannes, France, pp. 3-8.

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

[17] M. Pantic, M. Tomc, LJ.M. Rothkrantz, "A Hybrid approach to mouth features detection", in Proceeding of the 2001 Systems, Man and Cybernetics Conference, 2001, pp. 1188-1193.

[18] M. Pantic, M. F. Valstar, R. Rademaker, L. Maat, Web-based database for facial expression analysis, IEEE International Conference on Multimedia and Expo (ICME'05), http://www.mmifacedb.com, 2005.

[19] S. Chindaro, F. Deravi, "Directional Properties of Colour Co-occurrence Features for Lip Location and Segmentation", Proceedings of the 3rd International Conference on Audio and Video-Based Biometric Person Authentication, pp. 84 - 89, 2001.

[20] Lucey, S., A.B. Ashraf, and J. Cohn, "Investigating Spontaneous Facial Action Recognition through AAM Representations of the Face", Face Recognition Book,edited by K. Kurihara, ProLiteratur Verlag, Mammendorf, Germany, 2007.

[21] Zalewski, L. and S. Gong. "2D statistical models of facial expressions for realistic 3D avatar animation", in Computer Vision and Pattern Recognition, CVPR. 20-25 June 2005

[22] Kotsia, I., et al. "Texture and Shape Information Fusion for Facial Action Unit Recognition", in First International Conference on Advances in Computer-Human Interaction (ACHI). 2008.

[23] Hager, J., P. Ekman, and W. Friesen, "Facial action coding system", Salt Lake City, UT: A Human Face, 2002.

[24] Ekman, P. and W. Friesen, "Facial Action Coding System: A Technique for the Measurement of Facial Movemen",. Consulting Psychologists Press, Palo Alto, 1976.

[25] Y.Tian, T. Kanade, J. F. Cohn, Facial Expression Analysis, Book Chapter in Handbook of face recognition, S.Z. Li & A.K. Jain, ed., Springer, October, 2003.

[26] Wallhoff, F, "The Facial Expressions and Emotions Database Homepage (FEEDTUM)", www.mmk.ei.tum.de/ ˜ waf/fgnet/feedtum.html. Sept. 2005.

[27] Wallhoff, F., et al. Efficient Recognition of authentic dynamic facial expressions on the FEEDTUM database. in IEEE International Conference on Multimedia and Expo. 9-12 July 2006.

[28] Hao Tang; Yun Fu; Jilin Tu; Hasegawa-Johnson, M.; Huang, T.S., "Humanoid Audio–Visual Avatar With Emotive Text-to-Speech Synthesis," Multimedia, IEEE Transactions on , vol.10, no.6, pp.969-981, Oct. 2008

[29] M. C. Ionita, "Advances in the design of statistical face modelling techniques for face recognition", Ph.D. Thesis, National University of Ireland Galway, December 2008.

[30] Gualtieri, J.A. and R.F. Cromp. Support vector machines for hyperspectral remote sensing classification. in 27th AIPR Workshop: Advances in Computer Assisted Recognition. 1998. Washington, DC: SPIE

[31] M.C. Ionita and P. Corcoran, "Enhanced Real-Time Face Models from Stereo Imaging for Gaming Applications", at International IEEE Consumer Electronics Society's Games Innovations Conference 2009 (ICE-GIC 09), London, UK.

[32] I. Andorko and P. Corcoran, "FPGA Based Stereo Imaging System with Applications in Computer Gaming", at International IEEE Consumer Electronics Society's Games Innovations Conference 2009 (ICE-GIC 09), London, UK. .

Ioana Bacivarov received the M. Eng. Degree in Signal Processing from the National Polytechnic Institute of Grenoble, France and from the University "Politehnica" of Bucharest, Romania in a double diploma agreement in 2006, 2007 respectively. She is currently pursuing a Ph.D. degree in Image Processing & Computer Vision at NUI, Galway. Her research interests include signal processing and pattern recognition with applications in face recognition.

Peter Corcoran received the BAI (Electronic Engineering) and BA (Math's) degrees from Trinity College Dublin in 1984. He continued his studies at TCD and was awarded a Ph.D. for research work in the theory of Dielectric Liquids. He is currently Vice-Dean of research in the Collge of Engineering & Informatics, National University of Ireland Galway. His research interests include embedded systems, home networking, digital imaging and wireless networking technologies.