

# Bilinear Active Appearance Models

Jose Gonzalez-Mora<sup>†</sup> Fernando De la Torre<sup>‡</sup> Rajesh Murthi<sup>‡</sup> Nicolas Guil<sup>†</sup> Emilio L. Zapata<sup>†</sup>  
<sup>†</sup>, Department of Computer Architecture, University of Malaga, Malaga, Spain 29071.

jgmora@ac.uma.es nico@ac.uma.es ezapata@ac.uma.es

<sup>‡</sup>, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

ftorre@cs.cmu.edu rajeshm@andrew.cmu.edu

## Abstract

Appearance Models have been applied to model the space of human faces over the last two decades. In particular, Active Appearance Models (AAMs) have been successfully used for face tracking, synthesis and recognition, and they are one of the state-of-the-art approaches due to its efficiency and representational power. Although widely employed, AAMs suffer from a few drawbacks, such as the inability to isolate pose, identity and expression changes. This paper proposes Bilinear Active Appearance Models (BAAMs), an extension of AAMs, that effectively decouple changes due to pose and expression/identity. We derive a gradient-descent algorithm to efficiently fit BAAMs to new images. Experimental results show how BAAMs improve generalization and convergence with respect to the linear model. In addition, we illustrate decoupling benefits of BAAMs in face recognition across pose. We show how the pose normalization provided by BAAMs increase the recognition performance of commercial systems.

## 1. Introduction

Model-based methods revolutionized the computer vision field in the late 80's. They introduced a way to model an object's appearance/shape or image properties from training data, preventing difficulties that can arise in the computational or mathematical definition of the object. Since the early work of Sirovich and Kirby [34] parameterizing the human face using Principal Component Analysis (PCA) and the successful eigenfaces of Turk and Pentland [38], much computer vision research has used *eigen-whatever* or Component Analysis (CA) techniques to construct linear models of shape, appearance or optical flow [5, 10, 13, 28, 29]. The modeling power of subspace methods is specially useful when modeling visual data, because with the increase in features (pixels) there is a need for dimensionality reduction that preserves relevant attributes for

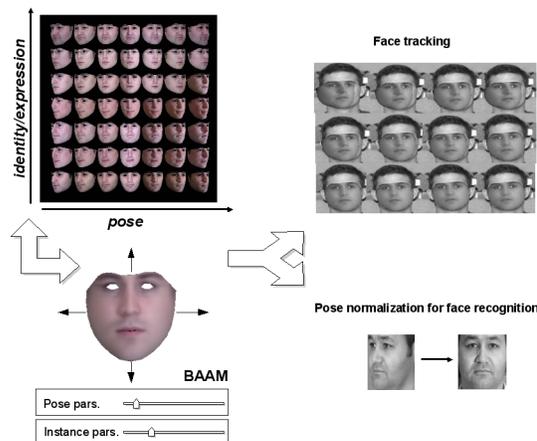


Figure 1. Synthetic images generated by BAAMs showing its factorization properties. Once the model is learned, it can be used for effective tracking and pose normalization in face recognition.

the tasks involved (e.g. tracking, classification, synthesis).

In particular, Active Appearance Models (AAMs) [10, 27, 4, 11, 16] have proven to be a good statistical tool to build a model of shape and appearance variation of human faces (also other objects). AAMs have been extensively applied to detection, tracking and synthesis of faces over the last decade. One of the problems of standard AAMs is its inability to decouple factors such as pose, illumination and expression, since PCA is used to jointly model all these sources of variability. Recent work in Multi-linear models (e.g. [35, 39, 40]) provide the basis to incorporate multi-factorial models that can decouple the image changes due to expression, identity or pose. In particular, pose changes are usually hard to represent with 2D linear models because of the resulting non-linear point transformations (e.g. points disappear with occlusion). This paper describes Bilinear Active Appearance Models (BAAMs), an extension of AAMs that effectively decouples pose changes from expression/identity variation in both shape and appearance. We propose to use bilinear models [35] to factorize pose

from expression/identity changes within the AAMs framework. Given a new image, an efficient gradient descent algorithm is developed to search for the optimal bilinear parameters. We also show how BAAMs can be used as a pose normalization step for face recognition algorithms, and we compare their performance with commercial systems, reporting conclusive results. Fig. 1 shows the main idea of the paper.

The rest of the paper is organized as follows. Section 2 reviews previous work on AAMs. Section 3 proposes BAMMs and derives the gradient-descent fitting algorithm. Section 4 summarizes the experiments for face tracking and face recognition across pose changes.

## 2. Previous Work

Generative appearance models such as Active Appearance Models (AAMs) [8, 4, 36, 6, 27, 14] have been extensively used in computer vision applications to model the class-variability and motion of objects in images (e.g. human faces). AAMs build a shape/appearance model from a labeled training set by means of Principal Component Analysis (PCA). Once the model is learned, AAMs use gradient-descent approaches to fit the model to a new image. One of the major limitations of AAMs is the fact that the PCA model is unable to decouple variations due to pose, expression and identity. There are several drawbacks of using a non-factorized model. Firstly, it is hard to interpret the parameters. Secondly, a non-decoupled PCA model of pose, expression and identity can introduce non-valid spaces and generate non-realistic shapes/appearance configurations.

One of the major and most complicated sources of face variability in video sequences is pose change. Several attempts have been done over the last decade to extend AAMs to deal with pose change. Lanitis et al [1] illustrated how a simple linear model is able to model changes in viewpoint assuming that the landmarks remain visible. Cootes et al. [9] extend AAMs to track the face from profile to profile by continuously switching several discrete view-dependent models (linear basis). However, it remains unclear how to impose consistency over the identity parameters in successive views; a possible solution is to find out the identity subspace at each pose basis in order to extrapolate the parameters to the new basis [12]. Using non-linear techniques such as Kernel PCA, Romdhani et al [33] show how to track the head from profile to profile. However, finding the pre-image is an iterative optimization problem prone to local minima. Moreover, they do not model the dependencies between appearance and shape, since the KPCA is just applied to the shape. On the other hand, there have been several solutions to deal with pose change that contemplate the use of 3D models. Blanz and Vetter [7] built a 3D Morphable model of face shape and texture. Using a general optimization scheme this technique performs good results in a broad

range of poses, and yields better reconstructions. However, the method usually requires 7 points to manually initialize, it needs depth information that is hard to acquire, and the fitting process is also slower.

On the other hand, bilinear models have shown good results in many factorized problems in vision, such as modeling illumination/identity or pose/identity for recognition or synthesis tasks. In their seminal work, [23] use bilinear models for classification across pose, considering the pose factor a discrete value and making use of separable mixture models in the low dimensional space. Several authors have applied this model to face, human motion and speech. Abboud and Davoine [2, 3] use an asymmetric bilinear model to separate identity/expression parameters using AAMs [10] for face recognition and synthesis. However, in this approach one of the factors is considered to be known in advance, using the same algorithm than [23] for model fitting when no previous information is given. The same authors [2] successfully applied bilinear problems in the context of face recognition, improving LDA results. In the area of human motion, [22] make use of bilinear models to transform an input motion into a new style while preserving its original content. Other extensions of bilinear models explore the probabilistic formulation; Grimes et al. [17] are able to model non-gaussian distributions over the parameter space using particle filters. Also, several extensions of multilinear models have been applied for face synthesis and recognition [39, 40].

To model more than two factors, Vasilescu and Terzopoulos [39, 40] have applied tensor factorization (e.g. [37]) methods to recognize and synthesize human motion. Similar in spirit, [21] propose a more efficient higher-order singular value decomposition algorithm to learn a factorized model of expression and use it for recognizing seven of the universal facial expressions. Lee and Elgammal [25] extended this model by adding a non-linear factor to take into account view-point changes. In [24], the same authors propose a gait recognition algorithm based on bilinear decomposition of gait data into time-invariant gait-style and time-dependent gait-content factors. Multilinear face models have also been successfully applied by Vlasic et. al [41] to face analysis. However, they use 3D scanned face data and pose variations are not explicitly modeled. Another difference is that their fitting process is based on optical flow, contrary to AAM approaches that make use of full appearance/shape observations.

Unlike previous work, we integrate the bilinear model in the core of the AAMs and develop a gradient descent algorithm for efficient fitting. We compare and characterize their performance in terms of convergence w.r.t linear models. We also show how it can be used as a normalization step for face recognition; experimental results are provided comparing the achieved recognition rate with a commercial

system.

### 3. Bilinear Active Appearance Models

The main aim of BAMMs is to provide a simple and effective 2D tracking algorithm able to decouple pose from identity/expression changes. In this section, we illustrate the training and fitting process for this factorized model.

#### 3.1. Learning the bilinear model

The standard bilinear models can be formulated as:

$$y_k^{(p)(i)} = \sum_{j=1}^J \sum_{l=1}^L w_{kjl} a_j^{(p)} b_l^{(i)} \quad (1)$$

$y_k^{(p)(i)}$  is the  $k_{th}$  component (e.g pixel) of the observed vector for the pose  $p$  and the individual  $i$ .  $w_{kjl}$  is the  $(k, j, l)$  component of the mixing matrix, that models the coupling between pose and identity/expression.  $a_j^{(p)}$  and  $b_l^{(i)}$  are the pose and instance parameters respectively. In many applications, one of the combined factors  $\mathbf{b}$  is considered to represent an intrinsic property of the signal and it is called *content*, while the other,  $\mathbf{a}$ , modulates in some way the actual representation of the signal, and it is named *style*. In our case, the content factor determines the considered person's appearance/shape characteristics (including identity and expression), while the style factor represents the pose from which the face image was captured. In the rest of the paper, we will use a vectorial notation<sup>1</sup> that will simplify the problem formulation.

Texture and shape observations are expected to be highly correlated across pose. We make use of this bilinear formulation to model both shape  $\mathbf{s}$  and texture  $\mathbf{t}$  vector instances in a coupled manner using the same parameters,  $\mathbf{a}$  and  $\mathbf{b}$ , to synthesize both of them (see figure 2).

$$t_k^{(p)(i)} = \hat{t}_k + \mathbf{a}^{(p)T} \mathbf{W}_k \mathbf{b}^{(i)} \quad (2a)$$

$$\mathbf{s}_k^{(p)(i)} = \begin{bmatrix} s_{xk} \\ s_{yk} \end{bmatrix} = \mathbf{H} \left( \begin{bmatrix} \hat{s}_{xk} \\ \hat{s}_{yk} \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{a}^{(p)T} \mathbf{R}_1 \mathbf{Y}_{xk} \mathbf{R}_2 \mathbf{b}^{(i)} \\ \mathbf{a}^{(p)T} \mathbf{R}_1 \mathbf{Y}_{yk} \mathbf{R}_2 \mathbf{b}^{(i)} \\ 0 \end{bmatrix} \right) \quad (2b)$$

subject to  $rank(\mathbf{R}_1) = c$  and  $rank(\mathbf{R}_2) = c$

$\mathbf{W}$  and  $\mathbf{Y}$  are the mixing matrices for appearance and shape respectively. Training shape and appearances are previously normalized, subtracting the mean shape  $\hat{\mathbf{s}}$  and the

<sup>1</sup>Bold capital letters denote a matrix  $\mathbf{D}$ , bold lower-case letters a column vector  $\mathbf{d}$ .  $\mathbf{d}_j$  represents the  $j$  column of the matrix  $\mathbf{D}$ .  $v_k$  names the  $k$  component of the vector  $\mathbf{v}$ . All non-bold letters will represent variables of scalar nature.  $\|A\|_F = tr(\mathbf{A}^T \mathbf{A}) = tr(\mathbf{A} \mathbf{A}^T)$  designates the Frobenious norm of a matrix.

mean appearance  $\hat{\mathbf{t}}$ . For the shape samples a rigid pose transformation  $\mathbf{H}$  is also included:

$$\mathbf{H} = \begin{pmatrix} 1 + h_1 & -h_2 & h_3 \\ h_2 & 1 + h_1 & h_4 \\ 0 & 0 & 1 \end{pmatrix}$$

Similarly to AAM approaches, training shapes are aligned in an initial step using Procrustes analysis, removing relative displacements, scales and in-plane rotations. When synthesizing a face instance, it is important to notice that parameter  $\mathbf{a}$  is responsible to model those shape/appearance pose variations corresponding to non-rigid 2D transformations due to out-of-plane rotations, still present in training shapes after Procrustes alignment. After that, pose matrix  $\mathbf{H}$  rigidly displaces the face onto the desired position/scale/rotation in the image plane (similarity transform).

Mixing matrices and coefficients are learned in a supervised manner using a criss-cross regression algorithm [18][15]. As the number of required parameters to represent shape and appearance within a given accuracy can greatly differ, two reduced-rank matrices  $\mathbf{R}_1$  and  $\mathbf{R}_2$  are also learned during the model construction process, coupling the two spaces.

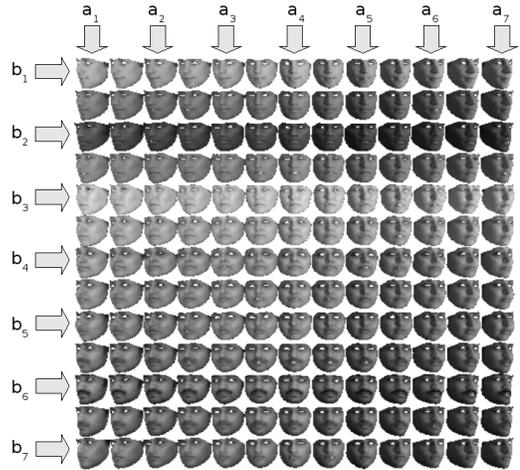


Figure 2. Synthesizing faces using BAAMs. Marked rows and columns correspond to the labeled training data. The remaining faces are synthesized by linearly interpolating the pose ( $a$ ) and instance parameters ( $b$ ).

#### 3.2. Fitting the model to a new image

Fitting a known model to a new image can be formulated as the following optimization problem:

$$f(\lambda, \mathbf{a}, \mathbf{b}) = \min_{\lambda, \mathbf{a}, \mathbf{b}} \|\mathbf{i}(\mathbf{w}(\mathbf{x}, \lambda, \mathbf{a}, \mathbf{b})) - \mathbf{t}(\mathbf{x}, \mathbf{a}, \mathbf{b})\|_2^2$$

, where:

- $\mathbf{i}(\mathbf{x})$  is the input image.
- $\mathbf{t}(\mathbf{x}, \mathbf{a}, \mathbf{b})$  is the reference texture model that models the appearance of the tracked face, depending on the bilinear model parameters  $\mathbf{a}$  and  $\mathbf{b}$ .
- $\mathbf{s} = \mathbf{w}(\mathbf{x}, \lambda, \mathbf{a}, \mathbf{b})$  represents a warping function converting the pixel coordinates from the reference frame into the input image. It is defined as a piecewise linear function that depends on the current position of the tracked face in the input image, defined by the pose parameters  $\lambda$ , and model parameters  $\mathbf{a}$  and  $\mathbf{b}$ .
- $\lambda = (h_1 \ h_2 \ h_3 \ h_4)^T$  is the pose parameter vector (more compact notation for rigid transformation matrix  $\mathbf{H}$ ).

Similar in spirit to previous work [8, 32], we linearize resulting cost function and optimize it using a Gauss-Newton method.

$$f(\lambda, \mathbf{a}, \mathbf{b}) = \|\mathbf{i}(\mathbf{w}(\lambda + \delta\lambda, \mathbf{a} + \delta\mathbf{a}, \mathbf{b} + \delta\mathbf{b})) - \mathbf{t}(\mathbf{x}, \mathbf{a} + \delta\mathbf{a}, \mathbf{b} + \delta\mathbf{b})\|_2^2 \quad (3)$$

A first order approximation around current parameter estimation  $(\lambda^0, \mathbf{a}^0, \mathbf{b}^0)^T$  is done in order to get a linear solution, resulting in:

$$f(\lambda, \mathbf{a}, \mathbf{b}) = \|\mathbf{e}(\lambda^0, \mathbf{a}^0, \mathbf{b}^0) + \mathbf{i}'(\mathbf{w}(\lambda^0, \mathbf{a}^0, \mathbf{b}^0)) \begin{pmatrix} \delta\lambda \\ \delta\mathbf{a} \\ \delta\mathbf{b} \end{pmatrix} - \mathbf{t}'(\mathbf{a}^0, \mathbf{b}^0) \begin{pmatrix} \delta\mathbf{a} \\ \delta\mathbf{b} \end{pmatrix}\|_2^2 \quad (4)$$

, where:

$$\mathbf{e}(\lambda^0, \mathbf{a}^0, \mathbf{b}^0) = \mathbf{i}(\lambda^0, \mathbf{a}^0, \mathbf{b}^0) - \mathbf{t}(\mathbf{a}^0, \mathbf{b}^0)$$

The least squares optimization leads to the following optimal parameters updates:

$$\begin{pmatrix} \delta\lambda \\ \delta\mathbf{a} \\ \delta\mathbf{b} \end{pmatrix} = -(\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \mathbf{e}$$

, with  $\mathbf{J}$  the Jacobian matrices obtained from the gradient matrices of  $\mathbf{i}$  and the derivatives of the bilinear model in respect to pose and bilinear model parameters.

$$\mathbf{J} = \begin{pmatrix} \frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \lambda, \mathbf{a}, \mathbf{b}))}{\partial \lambda} \\ \frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \lambda, \mathbf{a}, \mathbf{b}))}{\partial \mathbf{a}} - \frac{\partial \mathbf{t}(\mathbf{x}, \mathbf{a}, \mathbf{b})}{\partial \mathbf{a}} \\ \frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \lambda, \mathbf{a}, \mathbf{b}))}{\partial \mathbf{b}} - \frac{\partial \mathbf{t}(\mathbf{x}, \mathbf{a}, \mathbf{b})}{\partial \mathbf{b}} \end{pmatrix}$$

See section Appendix for more details about the derivatives computation.

## 4. Experimental results

In this section we show the benefits of modeling faces with BAAMs in two applications: face tracking and face recognition.

### 4.1. Tracking across pose with BAAMs

In the first experiment, we test the ability of BAAMs to represent and track faces across pose changes and compare them with standard AAMs [8]. All the models are trained using the CMU Multi-PIE face database [19], where there are changes in pose, illumination and expression. We select the subset of -45 to 45 angles to avoid severe occlusion problems. In this section, 40 individuals are used, labelled across 7 poses between -45 and 45 degrees. We divide them into two groups: a training set of 30 individuals and a testing set of 10 individuals.

The capability of BAAMs to decouple pose and identity parameters has several advantages. When using person-independent models, reduced-rank representations are usually preferred in order to avoid overfitting problems in the tracking process. Reducing the basis energy has the major drawback of increasing the model reconstruction error. Thus, a tradeoff between the number of model parameters and the reconstruction error must be found.

Figure 3 shows the first three principal components of pose coefficients after training a reduced-rank bilinear model. As can be observed, the pose parameters distribution is non-linear, similar to the camera configuration. The full-rank solution would give the same pose coefficient for all the samples having the same pose. The reduced-rank approach yields non-zero reconstruction error and spreads the coefficients, but it can be seen how BAAMs still cluster the coefficients keeping together the instances with the same pose. This is a desirable property in order to reduce overfitting while maintaining the decoupled structure of the modeled space.

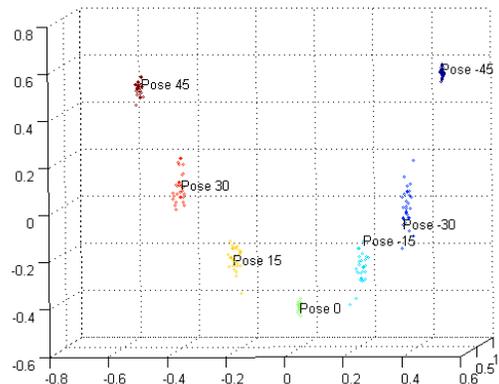


Figure 3. Distribution of pose coefficients for 40 individuals

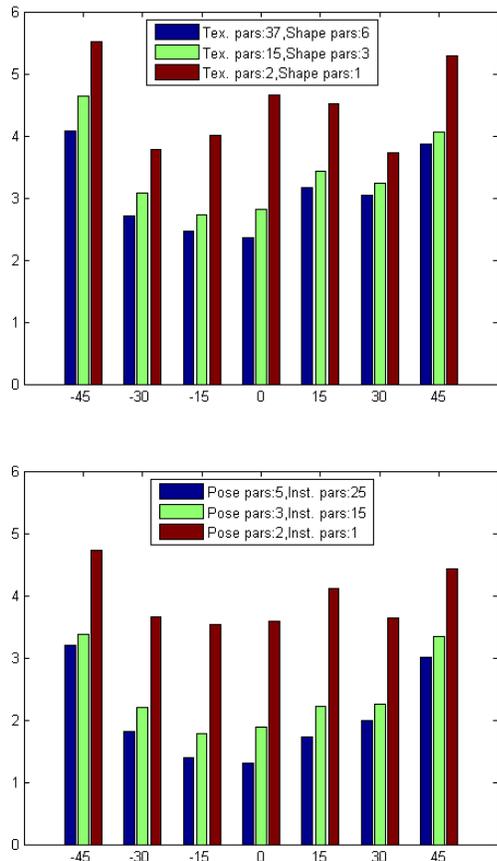


Figure 4. Mean geometrical reconstruction error using different numbers of parameters. Top: error for the reconstructed shapes at different poses for the AAM models. Bottom: reconstruction error for BAAM models

Figure 4 shows some reconstruction errors for reduced rank models in different poses. Observe that, in general, we can achieve lower reconstruction errors using BAAMs with the same (or fewer) number of coefficients than AAMs. As a result, BAAMs reduce the model basis dimensions to avoid overfitting, introducing a smaller reconstruction error than traditional linear approaches. This offers better generalization properties for person-independent trackers.

In order to evaluate model fitting performance, we measure the final geometric error by running instances of the person-independent AAM and BAAM trackers using different input images. In the figure 5 we compare the final mean geometrical errors achieved at each pose using these techniques, selecting the model with basis dimensions that performs better in each case. All the tracking experiments are initialized using the mean parameters obtained for the training data set (zero vectors for the AAM models) and an approximate pose aligning the model with the new input face bounding box. We can see as BAAMs tracking pro-

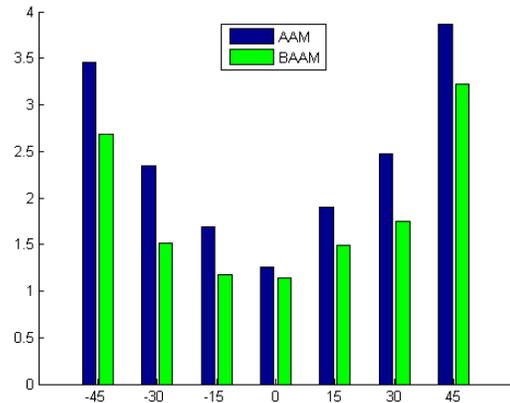


Figure 5. Final geometrical error comparison between linear and bilinear models

vides lower mean geometrical error in the recovered landmark positions.

Separate spaces give us the opportunity to perform more advanced optimization techniques than traditional linear model fitting algorithms (i.e. using several starting pose values so the tracking instance with best convergence error can be chosen). Finally, they give us the capability to extrapolate instances of known individuals using a novel pose. As shown in the next subsection, it will be very useful to perform face recognition using software designed to handle frontal views.

## 4.2. Face Recognition

Face recognition (see [42] for a review) is an important feature for many applications. However, for face recognition systems to be applicable in real world videos, they have to be robust to pose, illumination and expression changes of the face. Large variations in pose, illumination or expression typically occur in unconstrained video where subjects are non-cooperative. Usually they (1) tend not to look directly at the camera, leading to a non-frontal pose; (2) may be talking, leading to a non-neutral expression; and (3) may appear anywhere within a large area, resulting in low resolution and uncontrolled illumination.

Unfortunately, these conditions have been identified consistently in performance evaluations as a source of variations that cause poor recognition accuracies. In particular, uncontrolled face-pose scenarios correspond to situations that cause current commercial face recognition systems to fail, as in the Facial Recognition Vendor Test 2002, for example [31][20]. This specially is a major inconvenience in those applications using only frontal faces as gallery images and probe images captured in any random pose (e.g. border control, web page).

In this section, we show how a bilinear model can be

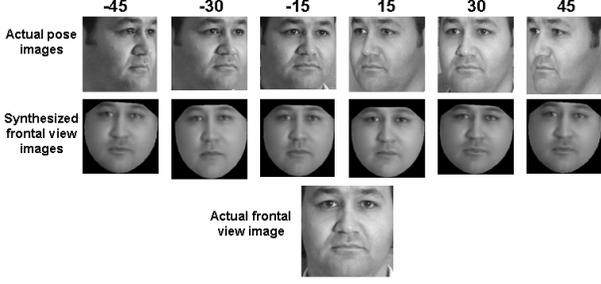


Figure 6. Reconstruction of frontal view images from pose images for a subject

used to synthesize frontal views (shape and appearance) from an arbitrary pose [26, 30] to perform face recognition across pose, and some preliminary results, using the CMU Multi-PIE database, are presented. Firstly, the corresponding instance parameters  $\mathbf{b}$  for an input face image are computed. Secondly, the frontal face is generated by keeping the instance parameters and rendering the image with the preset frontal parameter  $\mathbf{a}_0$ . See equation 2 for the synthesis expressions. We are only interested in evaluating the BAAMs capability to extrapolate a frontal view of a new subject. Thus, in order to isolate the errors involved in the tracking process, we perform a projection of the ground truth data for the new individual on the BAAMs.

To gauge how well our proposed pose normalization scheme improves facial recognition, we compared the recognition performance of commercial system FaceIt with and without pose normalization. We chose to use the FaceIt system because it performed best in the Face Recognition Vendor Test 2002 [31]. The Multi-PIE database has 337 subjects in 15 poses and 18 illuminations. Out of the available 337 subjects, 250 subjects were chosen. A bilinear model for pose changes is trained for the 200 subjects in 7 poses as described in the previous sections. The remaining 50 subjects, with one image per subject in all the 7 poses, constitute the test set. In figure 6 there are some examples of a synthesized frontal view image from all seven poses.

The performance of FaceIt is shown in table 1. Firstly we perform face identification without pose normalization. Then, we repeat the experiments having the same gallery images and new probe images obtained by synthesizing frontal view images from  $15^\circ$ ,  $30^\circ$ ,  $45^\circ$ ,  $-15^\circ$ ,  $-30^\circ$  and  $-45^\circ$  pose images. The results show that pose normalization improves face identification by over 80%.

## 5. Conclusions

In this paper, we have proposed BAAMs to effectively track and decouple pose changes from identity/expression. We have implemented an efficient gradient-descent algorithm and empirically show better convergence and generalization properties of BAAMs versus traditional AAMs. In

Pose	With pose normalization	W/o pose normalization
$15^\circ$	86%	36%
$30^\circ$	84%	28%
$45^\circ$	80%	0%
$-15^\circ$	86%	38%
$-30^\circ$	80%	34%
$-45^\circ$	75%	0%

Table 1. Recognition rates with and without pose normalization

addition, we have shown how BAAMs can be used as a pre-processing step for pose normalization in face recognition applications.

Although BAAMs have shown promising results both in tracking and recognition, there are a number of issues that need to be addressed. There is a need to take into account more factors such as illumination or separate identity from expression. This could be done in a straightforward manner with multilinear models. For pose changes, it will be convenient to have larger pose variation range and incorporate the other two rotational angles. This will require a more extensive database.

## 6. APPENDIX

Derivatives in respect to model style parameters:

$$\frac{\partial \mathbf{t}(\mathbf{x}, \mathbf{a}, \mathbf{b})}{\partial a_i} = \sum_{l=1}^L \mathbf{w}_{il} b_l \quad (5)$$

$$\frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \lambda, \mathbf{a}, \mathbf{b}))}{\partial a_i} = (\nabla_x \mathbf{i} \nabla_y \mathbf{i}) \begin{pmatrix} (1 + h_1) \sum_{l=1}^L y_{xil} b_l - h_2 \sum_{l=1}^L y_{yil} b_l \\ h_2 \sum_{l=1}^L y_{xil} b_l + (1 + h_1) \sum_{l=1}^L y_{yil} b_l \end{pmatrix} \quad (6)$$

Derivatives in respect to model content parameters:

$$\frac{\partial \mathbf{t}(\mathbf{x}, \mathbf{a}, \mathbf{b})}{\partial b_i} = \sum_{j=1}^J w_{ji} a_j \quad (7)$$

$$\frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \lambda, \mathbf{a}, \mathbf{b}))}{\partial b_i} = (\nabla_x \mathbf{i} \nabla_y \mathbf{i}) \begin{pmatrix} (1 + h_1) \sum_{j=1}^J y_{xji} a_j - h_2 \sum_{j=1}^J y_{yji} a_j \\ h_2 \sum_{j=1}^J y_{xji} a_j + (1 + h_1) \sum_{j=1}^J y_{yji} a_j \end{pmatrix} \quad (8)$$

Derivatives in respect to pose parameters:

$$\frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \mathbf{p}, \mathbf{a}, \mathbf{b}))}{\partial h_1} = (\nabla_x \mathbf{i} \nabla_y \mathbf{i}) \begin{pmatrix} x \\ y \end{pmatrix} \quad (9)$$

$$\frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \mathbf{p}, \mathbf{a}, \mathbf{b}))}{\partial h_2} = (\nabla_x \mathbf{i} \nabla_y \mathbf{i}) \begin{pmatrix} -y \\ x \end{pmatrix} \quad (10)$$

$$\frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \mathbf{p}, \mathbf{a}, \mathbf{b}))}{\partial h_3} = \nabla_x \mathbf{i} \quad (11)$$

$$\frac{\partial \mathbf{i}(\mathbf{w}(\mathbf{x}, \mathbf{p}, \mathbf{a}, \mathbf{b}))}{\partial h_4} = \nabla_y \mathbf{i} \quad (12)$$

## References

- [1] T. C. A. Lanitis, C.J. Taylor. An automatic face identification system using flexible appearance models. pages 65–74, 1994.
- [2] B. Abboud and F. Davoine. Appearance factorization based facial expression recognition and synthesis. In *ICPR*, volume 4, pages 163–166, 2004.
- [3] B. Abboud and F. Davoine. Bilinear factorization for facial expression analysis and synthesis. In *Vision Image and Signal Processing*, pages 327–333, 2005.
- [4] S. Baker, I. Matthews, and J. Schneider. Automatic construction of active appearance models as an image coding problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1380 – 1384, October 2004.
- [5] M. J. Black, D. J. Fleet, and Y. Yacoob. Robustly estimating changes in image appearance. *Computer Vision and Image Understanding*, 78(1):8–31, 2000.
- [6] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, 1999.
- [7] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Siggraph 1999, Computer Graphics Proceedings*, pages 187–194, 1999.
- [8] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. In *Proceedings of the European Conference on Computer Vision*, 1998.
- [9] T. Cootes, K. Walker, and C. Taylor. View-based active appearance models. In *IEEE International Conference on Automatic Face and Gesture Recognition*, 2000.
- [10] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *European Conference Computer Vision*, pages 484–498, 1998.
- [11] T. F. Cootes, G. V. Wheeler, K. N. Walker, and C. J. Taylor. Coupled-view active appearance models. In *British Machine Vision Conference*, pages 52 – 61, 2000.
- [12] N. Costen, T. Cootes, G. Edwards, and C. Taylor. Automatic extraction of the face identity subspace. In *X British Machine Vision Conference*, volume 2, pages 513–522, 1999.
- [13] F. de la Torre, S. Gong, and S. McKenna. View-based adaptive affine alignment. In *European Conference on Computer Vision*, pages 828–842, 1998.
- [14] F. de la Torre, J. Vitrià, P. Radeva, and J. Melenchón. Eigenfiltering for flexible eigentracking. In *International Conference on Pattern Recognition*, pages 1118–1121, 2000.
- [15] K. R. Gabriel and S. Zamir. Lower rank approximation of matrices by least squares with any choice of weights. *Technometrics*, Vol. 21, pp., 21:489–498, 1979.
- [16] S. Gong, S. Mckenna, and A. Psarrou. *Dynamic Vision: From Images to Face Recognition*. Imperial College Press, 2000.
- [17] D. Grimes, A. Shon, and R. Rao. Probabilistic bilinear models for appearance-based vision. In *IEEE International Conference on Computer Vision*, 2003.
- [18] R. Gross, F. D. la Torre, and S. Baker. An unifying view of bilinear models. In *In preparation*.
- [19] R. gross, I. matthews, J. cohn, and S. Baker. Guide to cmu multi-pie face database. 2002.
- [20] R. Gross, J. Shi, and J. Cohn. Quo vadis face recognition? In *Proceedings of the Third Workshop on Empirical Evaluation Methods in Computer Vision*, 2001.
- [21] W. Hongcheng and N. Ahuja. Facial expression decomposition. In *IEEE International Conference on Computer Vision*, pages 958–965, 2003.
- [22] E. Hsu, K. Pulli, and J. Popovic. Style translation for human motion. In *International Conference on Computer Graphics and Interactive Techniques*, 2005.
- [23] W. F. J.B. Tenenbaum. Separating style and content. In *Adv in Neural Info. Proc. Systems*, volume 9, 1997.
- [24] C. Lee and A. Elgammal. Gait style and gait content: bilinear models for gait recognition using gait re-sampling. In *Automatic Face and Gesture Recognition.*, 2004.
- [25] C. Lee and A. Elgammal. Facial expression analysis using nonlinear decomposable generative models. In *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, pages 17–31, 2005.
- [26] S. Lucey and T. Chen. Learning patch dependencies for improved pose mismatched face verification. In *CVPR*, 2006.
- [27] I. Matthews and S. Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135–164, Nov. 2004.
- [28] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *Pattern Analysis and Machine Intelligence*, 19(7):137–143, July 1997.
- [29] S. K. Nayar and T. Poggio. *Early Visual Learning*. Oxford University Press, 1996.
- [30] J. Ni and H. Schneiderman. Face view synthesis across large angles. *Technical Report*.
- [31] P. J. Phillips, P. Grother, J. M. Ross, D. Blackburn, E. Tabassi, and M. Bone. Face recognition vendor test 2002: evaluation report, 2003.
- [32] I. M. S. Baker. Lucas-kande 20 years on: A unifying framework. *International Journal of Computer Vision*, 56:221–255, 2004.
- [33] A. P. S. Romdhani, S. Gong. A multiview non-linear active shape model using kernel pca. In *X British Machine Vision Conference*, pages 483–492, 1999.
- [34] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am. A*, 4(3):519–524, March 1987.
- [35] J. B. Tenenbaum and W. T. Freeman. Separating style and context with bilinear models. *Neural Computation*, 12(6):1247–1283, 2000.
- [36] N. F. Troje and T. Vetter. Representations of human faces. Technical Report 41, Max-Planck-Institut fr biologische kybernetik.

- [37] L. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.
- [38] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal Cognitive Neuroscience*, 3(1):71–86, 1991.
- [39] M. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *Proc. European Conf. on Computer Vision*, 2002.
- [40] M. Vasilescu and D. Terzopoulos. Multilinear subspace analysis of image ensembles. In *Computer Vision and Pattern Recognition*, 2003.
- [41] D. Vlasic, M. Brand, H. Pfister, and J. Popovi. Face transfer with multilinear models. In *SIGGRAPH*, 2005.
- [42] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM computing surveys*, 35(4):399–458, 2003.