

Improved Face Model Fitting on Video Sequences[†]

Xiaoming Liu Frederick W. Wheeler Peter H. Tu

Visualization and Computer Vision Lab
General Electric Global Research Center
Niskayuna, NY 12309, USA
{liux,wheeler,tu}@research.ge.com

Abstract

Active Appearance Models (AAMs) represent the shape and appearance of an object via two low-dimensional subspaces, one for shape and one for appearance. AAMs for facial images are currently receiving considerable attention from the computer vision community. However, most existing work focuses on fitting AAMs to a single image. For many applications, effectively fitting an AAM to video sequences is of critical importance and challenging, especially considering the varying quality of real-world video content. This paper proposes a hybrid model to address this problem. Both a generic AAM and a subject-specific model are employed simultaneously in the proposed fitting scheme. Experimental results from outdoor surveillance video sequences demonstrate the improved image registration across video frames and faster fitting convergence.

1 Introduction

Model-based image registration/alignment is a fundamental topic in computer vision. *Active Appearance Models (AAMs)* have been one of the most popular models for image registration [4]. Face alignment using an AAM is receiving considerable attention from the computer vision community because it enables various capabilities such as facial feature detection, pose rectification, and gaze estimation. However, most existing work focuses on fitting the AAM to a single facial image. With the abundance of surveillance cameras and greater need for face recognition from video, methods to effectively fit an AAM to facial images in videos are of increasing importance. This paper addresses this problem and proposes a novel algorithm for it.

There are two basic components in face alignment using an AAM: *face modeling* and *model fitting*. Given a set of facial images, *face modeling* is the procedure of training the AAM, which is essentially two distinct linear subspaces modeling facial shape and

[†]This project was supported by awards #2005-IJ-CX-K060 and #2006-IJ-CX-K045 awarded by the National Institute of Justice, Office of Justice Programs, US Department of Justice. The opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the Department of Justice.

appearance respectively. *Model fitting* refers to estimating the parameters of the resulting AAM on faces in an image or video frames by minimizing the distance measured between the image and the AAM.

In the context of fitting an AAM to video sequences, conventional methods directly fit the AAM to each frame by using the fitting results, i.e., the shape and appearance parameters, of the previous frame as the initialization of the current frame. However, as shown in the previous work [6], fitting to faces of an unseen subject can be hard due to the mismatch between the appearance of the facial images used for training the AAM and that of the video sequences, especially when the video sequences are captured in the outdoor environment. Also, the conventional method only registers each frame with respect to the AAM, without enforcing the frame-to-frame registration across video sequences, which is necessary for many practical applications, such as multi-frame super-resolution [13].

To address this problem, we propose a novel approach to continuously fit the AAM to video sequences. The proposed algorithm is an extension of the state-of-the-art image alignment algorithm – the Simultaneous Inverse Compositional (SIC) method [1], which minimizes the distance of the warped image observation and the generic AAM model during the fitting. We call our proposed approach as “*SIC for Video (SICOV)*” algorithm, which not only minimizes the above distance measure, but also the distance between the warped image and a model obtained from the warped images of previous video frames. Experimental results show that the SICOV algorithm improves both the fitting accuracy across frames and the fitting speed.

Many approaches have been proposed for modeling faces with AAMs [4, 1]. Baker and Matthews [1] proposed the Inverse Compositional (IC) method and SIC method that greatly improves the fitting speed and performance. However, little work has been done in fitting AAMs to facial video sequences in particular. Koterba *et al.* [7] proposed to use a 3D face model as a constraint in fitting multiple video frames. Matthews *et al.* [11] also updated the generic AAM using the warped image observation, such that a subject-specific model can be obtained during the fitting process. Comparing to their approach, we will show that treating the previous frame information as an additional constraint can improve the fitting speed, not to mention saving the extra time needed to update the bulky eigenspace of the appearance model in an AAM. Bosch *et al.* [2] proposed an Active Appearance Motion Model that captures the motion pattern in video sequences by taking the concatenation of the landmarks from multiple frames as training samples. This approach takes advantage of the periodic motion pattern in medical image sequences. In contrast, our approach does not make assumption on the object’s motion.

This paper is organized as follows. Section 2 introduces the conventional methods for training the AAM and model fitting. Sections 3 and 4 present the proposed SICOV algorithm and its detailed derivation. Section 5 provides experimental results, and conclusions are given in Section 6.

2 Active Appearance Models and Model Fitting

The shape model and appearance model part of an AAM are trained with a representative set of facial images. The distribution of facial landmarks are modeled as a Gaussian distribution, which is regarded as the shape model. The procedure for training a shape model is as follows. Given a face database, each facial image is manually labeled with

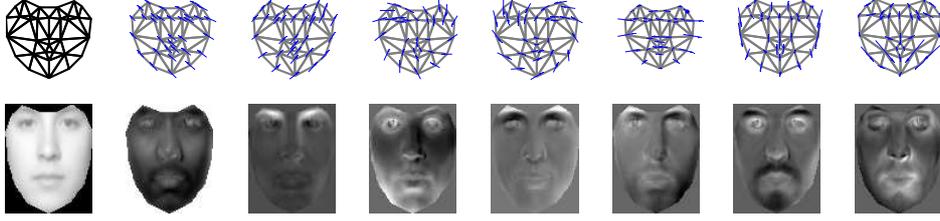


Figure 1: The mean and first 7 basis vectors of the shape model (top) and the appearance model (bottom) trained from the ND1 database. The shape basis vectors are shown as arrows at the corresponding mean shape landmark locations.

a set of 2D landmarks, $[x_i, y_i]$ $i = 1, 2, \dots, v$. The collection of landmarks of one image is treated as one observation from the random process defined by the shape model, $\mathbf{s} = [x_1, y_1, x_2, y_2, \dots, x_v, y_v]^T$. Eigen-analysis is applied to the observation set and the resulting linear shape model represents a shape as,

$$\mathbf{s}(\mathbf{P}) = \mathbf{s}_0 + \sum_{i=1}^n p_i \mathbf{s}_i, \quad (1)$$

where \mathbf{s}_0 is the mean shape, \mathbf{s}_i is the i^{th} shape basis, and $\mathbf{p} = [p_1, p_2, \dots, p_n]$ are the shape parameters. By design, the first four shape basis vectors represent global rotation and translation. Together with other basis vectors, a mapping function from the model coordinate system to the coordinates in the image observation is defined as $\mathbf{W}(\mathbf{x}; \mathbf{p})$, where \mathbf{x} is a pixel coordinate defined by the mean shape \mathbf{s}_0 .

After the shape model is trained, each facial image is warped into the mean shape using a piecewise affine transformation. These shape-normalized appearances from all training images are fed into an eigen-analysis and the resulting model represents an appearance as,

$$A(\mathbf{x}; \lambda) = T(\mathbf{x}) + \sum_{i=1}^m \lambda_i A_i(\mathbf{x}), \quad (2)$$

where T is the mean appearance, A_i is the i^{th} appearance basis, and $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_m]$ are the appearance parameters. Figure 1 shows an AAM trained using 534 images of 200 subjects from the ND1 3D face database [3].

An AAM can synthesize facial images with arbitrary shape and appearance within the range expressed by the training population. Thus, the AAM can be used to *explain* a facial image by finding the optimal shape and appearance parameters such that the synthesized image is as similar to the image observation as possible. This leads to the cost function used for model fitting [5],

$$J(\mathbf{p}, \lambda) = \sum_{\mathbf{x} \in \mathbf{s}_0} [I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - A(\mathbf{x}; \lambda)]^2, \quad (3)$$

which is the mean-square-error (MSE) between the image warped from the observation $I(\mathbf{W}(\mathbf{x}; \mathbf{p}))$ and the synthesized appearance model instance $A(\mathbf{x}; \lambda)$.

Traditionally this minimization problem is solved by iterative gradient-descent methods which estimate $\Delta \mathbf{p}$, $\Delta \lambda$ and add them to \mathbf{p} , λ . Baker and Matthews [1] proposed the

compositional method to generate the new shape parameter based on $\Delta \mathbf{p}$ in their IC and SIC method. The key idea of IC and SIC is that the role of the appearance template and the input image is switched when computing $\Delta \mathbf{p}$. This enables the time-consuming steps of parameter estimation to be pre-computed and performed outside of the iteration loop. We will borrow this key idea in deriving the solution of our SICOV algorithm.

3 The SICOV algorithm

Given a generic AAM and a video frame I_t at time t , SICOV uses the following cost function to perform the face model fitting:

$$J_t(\mathbf{p}, \lambda) = \sum_{\mathbf{x} \in \mathbf{s}_0} [T(\mathbf{x}) + \sum_{i=1}^m \lambda_i A_i(\mathbf{x}) - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p}))]^2 + k \sum_{\mathbf{x} \in \mathbf{s}_0} [M_t(\mathbf{x}) - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p}))]^2, \quad (4)$$

which is composed of two terms weighted by a constant k . The first one is the same as Eq. (3), i.e., the MSE between the warped image and the synthesized appearance model instance. The second one is the MSE between the current warped image $I_t(\mathbf{W}(\mathbf{x}; \mathbf{p}))$ and the appearance information of the current subject from previous frames, $M_t(\mathbf{x})$.

There are different options in defining $M_t(\mathbf{x})$. Firstly, it can be the warped image of the video frame at time $t - 1$:

$$M_t(\mathbf{x}) = I_{t-1}(\mathbf{W}(\mathbf{x}; \mathbf{p}_{t-1})). \quad (5)$$

Secondly, the warped images of L previous video frames averaged by a decaying factor can also represent $M_t(\mathbf{x})$:

$$M_t(\mathbf{x}) = \frac{1-r}{r(1-r^L)} \sum_{l=1}^L r^l I_{t-l}(\mathbf{W}(\mathbf{x}; \mathbf{p}_{t-l})), \quad (6)$$

where r is a decaying factor between 0 and 1. In practice, when fitting the video frame at time t , both definitions of $M_t(\mathbf{x})$ are known and can be computed efficiently from the previous fitting results. Of course, other definitions of $M_t(\mathbf{x})$ are also possible, for example, the average of L previous warped images without decaying, and a dynamic eigenspace model of the previous warped images [9]. In the latter case, an efficient eigenspace updating method can be used to sequentially add the most recent warped image into the model [8], and additional appearance parameters of this eigenspace model should be incorporated into the the second term of Eq. (4).

These two terms in Eq. (4) can be treated as the distance between the current image observation and the generic face model and the subject-specific model respectively, which is obtained in an on-line fashion from image observation at the previous time instances. Thus in the fitting of each frame, both distance measures are served as constraints to guide the fitting process.

There are clear benefits from using these two models during the face model fitting. First of all, in practical applications there is always mismatch between the imaging environment of the images used for training face models and the images to be fit, as well as the presence of the specific appearance information of the subject being fit that is not modeled by the generic face models. Thus the distance-to-subject-specific-model is employed to bridge such a gap. Secondly, if we only use the subject-specific model, the alignment error would propagate over time. The generic model is well suited for preventing the error propagation and correcting the drifting.

4 Derivation of SICOV algorithm

Using an approach similar to the IC and SIC algorithms [1], the proposed SICOV algorithm iteratively minimizes:

$$\sum_{\mathbf{x}} \left[T(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p})) + \sum_{i=1}^m (\lambda_i + \Delta \lambda_i) A_i(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p})) - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p})) \right]^2 + k \sum_{\mathbf{x}} [M_t(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p})) - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p}))]^2 \quad (7)$$

with respect to $\Delta \mathbf{p}$ and $\Delta \lambda = (\Delta \lambda_1, \dots, \Delta \lambda_m)^T$ simultaneously, and then updates the warp $\mathbf{W}(\mathbf{x}; \mathbf{p}) \leftarrow \mathbf{W}(\mathbf{x}; \mathbf{p}) \circ \mathbf{W}(\mathbf{x}; \Delta \mathbf{p})^{-1}$ and the appearance parameter $\lambda \leftarrow \lambda + \Delta \lambda$.

In order to solve for $\Delta \mathbf{p}$ and $\Delta \lambda$, the non-linear expression in Eq. (7) is linearized by performing a first order Taylor series expansion on $T(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p}))$, $A_i(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p}))$, and $M_t(\mathbf{W}(\mathbf{x}; \Delta \mathbf{p}))$, and assuming that $\mathbf{W}(\mathbf{x}; \mathbf{0})$ is the identity warp. This gives:

$$\sum_{\mathbf{x}} \left[T(\mathbf{x}) + \nabla T \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} + \sum_{i=1}^m (\lambda_i + \Delta \lambda_i) (A_i(\mathbf{x}) + \nabla A_i \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p}) - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p})) \right]^2 + k \sum_{\mathbf{x}} \left[M_t(\mathbf{x}) + \nabla M_t \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p})) \right]^2. \quad (8)$$

The first term in the above equation can be simplified as follows by neglecting the second order terms:

$$\sum_{\mathbf{x}} \left[T(\mathbf{x}) + \sum_{i=1}^m \lambda_i A_i(\mathbf{x}) - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p})) + (\nabla T + \sum_{i=1}^m \lambda_i \nabla A_i) \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} + \sum_{i=1}^m A_i(\mathbf{x}) \Delta \lambda_i \right]^2. \quad (9)$$

To simplify the notation, firstly we denote $\mathbf{q} = (\mathbf{p}^T \lambda^T)^T$ and similarly $\Delta \mathbf{q} = (\Delta \mathbf{p}^T \Delta \lambda^T)^T$. Thus \mathbf{q} is a $n + m$ dimensional vector including both the shape parameters \mathbf{p} and the appearance parameters λ . Secondly, we denote $n + m$ dimensional steepest-decent images:

$$\mathbf{SD}(\mathbf{x}) = \left[(\nabla T + \sum_{i=1}^m \lambda_i \nabla A_i + k \nabla M_t) \frac{\partial \mathbf{W}}{\partial p_1}, \dots, (\nabla T + \sum_{i=1}^m \lambda_i \nabla A_i + k \nabla M_t) \frac{\partial \mathbf{W}}{\partial p_n}, A_1(\mathbf{x}), \dots, A_m(\mathbf{x}) \right]. \quad (10)$$

Thirdly, we denote the error image:

$$E(\mathbf{x}) = T(\mathbf{x}) + \sum_{i=1}^m \lambda_i A_i(\mathbf{x}) - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p})) + k(M_t(\mathbf{x}) - I_t(\mathbf{W}(\mathbf{x}; \mathbf{p}))). \quad (11)$$

Equation (8) is simplified to:

$$\sum_{\mathbf{x}} [E(\mathbf{x}) + \mathbf{SD}(\mathbf{x}) \Delta \mathbf{q}]^2. \quad (12)$$

The partial derivative of Eq. (12) with respect to $\Delta \mathbf{q}$ is:

$$2 \sum_{\mathbf{x}} \mathbf{SD}^T(\mathbf{x}) [E(\mathbf{x}) + \mathbf{SD}(\mathbf{x}) \Delta \mathbf{q}]. \quad (13)$$

Pre-compute:

- (3) Evaluate the gradients ∇T , ∇M_i , and ∇A_i for $i = 1, 2, \dots, m$
- (4) Evaluate the Jacobian $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ at $(\mathbf{x}; \mathbf{0})$

Iterate:

- (1) Warp I with $\mathbf{W}(\mathbf{x}; \mathbf{p})$ to compute $I(\mathbf{W}(\mathbf{x}; \mathbf{p}))$
- (2) Compute the error image $E(\mathbf{x})$ using Eq. (11)
- (5) Compute the steepest decent image $\mathbf{SD}(\mathbf{x})$ using Eq. (10)
- (6) Compute the Hessian matrix \mathbf{H} using Eq. (15) and invert the matrix
- (7) Compute $\sum_{\mathbf{x}} \mathbf{SD}^T(\mathbf{x})E(\mathbf{x})$
- (8) Compute $\Delta \mathbf{q}$ using Eq. (14)
- (9) Update $\mathbf{W}(\mathbf{x}; \mathbf{p}) \leftarrow \mathbf{W}(\mathbf{x}; \mathbf{p}) \circ \mathbf{W}(\mathbf{x}; \Delta \mathbf{p})^{-1}$ and $\lambda \leftarrow \lambda + \Delta \lambda$

until $\|\Delta \mathbf{p}\| \leq \varepsilon$

Figure 2: Summary of the SICOV algorithm.

The closed form solution of Eq. (7) is obtained by setting Eq. (13) to equal zero:

$$\Delta \mathbf{q} = -\mathbf{H}^{-1} \sum_{\mathbf{x}} \mathbf{SD}^T(\mathbf{x})E(\mathbf{x}), \quad (14)$$

where \mathbf{H}^{-1} is the inverse of the Hessian matrix:

$$\mathbf{H} = \sum_{\mathbf{x}} \mathbf{SD}^T(\mathbf{x})\mathbf{SD}(\mathbf{x}). \quad (15)$$

The algorithm is summarized in Figure 2. The computation cost of the SICOV algorithm is summarized in Table 1. It can be seen that although the additional constraint results in slight more computation in Step (2) and Step (5), the computation cost per iteration of SICOV is almost the same as that of the SIC algorithm [1].

Pre-computation	Step 3	$O(mN)$	$O((n+m)N)$
	Step 4	$O(nN)$	
Per Iteration	Step 1	$O(nN)$	$O((n+m)^2N + (n+m)^3)$
	Step 2	$O(mN)$	
	Step 5	$O((n+m)N)$	
	Step 6	$O((n+m)^2N + (n+m)^3)$	
	Step 7	$O((n+m)N)$	
	Step 8	$O((n+m)^2)$	
	Step 9	$O(n^2 + m)$	

Table 1: The computation cost of the SICOV algorithm. The right column indicates the total cost for the pre-computation and each iteration.

5 Experiments

To evaluate our algorithm, we collect a set of 400 images from two public available databases, the ND1 database [3], which contains 953 facial images with mostly frontal



Figure 3: Examples of the face dataset: ND1 database (left) and FERET database (right).

views from 273 subjects, and the FERET database [12], which contains a large number of subjects with various poses and expressions. Figure 3 shows sample images from these two databases. In our experiment, we use a 200-image subset from the ND1 database and a 200-image subset from the FERET database. Each one of the 400 images comes from different subjects. This 400-image set is used to train a generic AAM. Iterative model enhancement [10] is used in the training stage and results in a more compact model than the conventional approach. The resulting AAM has 10 shape bases, 52 appearance bases, and the width of the mean shape is 62 pixels.

A number of outdoor test surveillance video sequences, whose subjects are not included in the training dataset, are captured at 30 frames per second (FPS). For comparison purpose, we have implemented both the SIC and SICOV algorithms in MatlabTM. By manually placing the mean shape on the first video frame, SICOV and SIC algorithms are used to fit the above generic AAM to these test videos respectively. The only parameter for the SICOV algorithm, k , is set to $k = 1$ throughout the experiments. Ideally k should be set according on the *correctness* of the individual model $M_i(\mathbf{x})$. We use Eq. (5) as the definition of $M_i(\mathbf{x})$. The first video sequence contains 980 frames. The proposed SICOV algorithm successfully fits the face over the whole video sequence while the SIC algorithm loses the fitting starting from frame 780 due to large pose change. In the case where there is no manual label for each frame of the test video sequences, visual inspection of the fitting results is one way of evaluating the performance. Figure 4 shows the comparison between two methods on 6 frames in this video. A visually more accurately fitted mesh is observed when using the SICOV algorithm.

Other than visual inspection, an alternative way to evaluate the fitting performance is to quantitatively compute the registration consistency across frames, which is represented by the MSE of the warped image observations between consecutive frames. As shown in Figure 5, SICOV provides on average lower MSE for the entire sequence, especially when SIC has high MSE at certain frames due to the changing facial appearance. Hence this shows superior frame-to-frame registration using the SICOV algorithm. On one hand, this is a favorable property for many applications that requires accurate registration across time, such as super resolution from video sequences. On the other hand, this is also an expected result since the frame-to-frame registration measure is part of the SICOV's objective function.

Our proposed method can improve not only the fitting robustness and accuracy, but also the fitting speed. Figure 6 shows the number of iterations for fitting each frame using the SIC and SICOV algorithm. The lower curve of SICOV indicates that SICOV can converge much faster than SIC. This improvement is expected because the additional



Figure 4: Comparison of the fitted mesh using the SICOV algorithm (dashed line) and the conventional SIC algorithm (solid line) on 6 frames (frame 1, 40, 87, 287, 734 and 767).

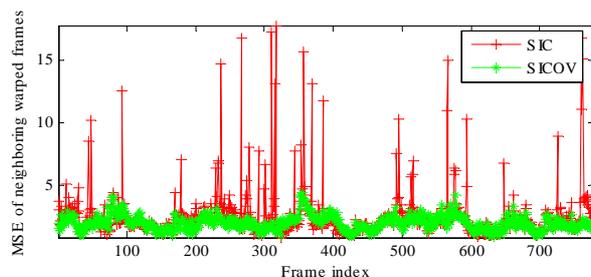


Figure 5: The MSE of neighboring warped frames of a video sequence. Constant lower MSE indicates the improved frame-to-frame registration using the SICOV algorithm.

constraint in SICOV helps the minimization procedure. Given the fact that the computation cost per iteration in the fitting is almost the same as SIC, the average time for fitting one frame using SICOV is much lower because less iterations are needed for fitting to converge. Based on a MatlabTM implementation running on a conventional 2.13 GHz PentiumTM4 computer, on average SICOV takes 0.1254 sec. to fit one frame compared to 0.2526 sec. by SIC. We have also implemented the SICOV using C++ and resulting facial fitting system can run 25+ FPS on a conventional PC for unseen subjects.

Figure 7 shows the fitting results on another 970-frame-long video sequence, where a Pan-Tilt-Zoom (PTZ) camera is pointing at three subjects and continuously zooming out. This is to mimic the scenario where in surveillance applications the subjects can have various distance to a camera and the face image can be of low resolution. How to effectively fit a face model onto this type of challenging real-world video sequence receives relatively little attention in the vision community. The proposed SICOV algorithm successfully fits the entire video sequence, even when zooming happens and large scale change appears in consecutive frames. Note that the smallest face size in this video sequence only has the face width of 15 pixels. However, when applying the conventional SIC algorithm, the fitting diverges starting at frame 34 when the first zooming happens.

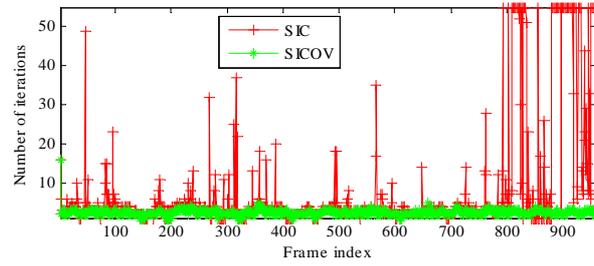


Figure 6: The number of iterations in fitting each frame of a video sequence. Constant lower number of iterations is observed from the proposed SICOV algorithm.



Figure 7: Fitting results with zoom in facial area using SICOV. Reliable fitting is observed in dealing with zooming and low resolution, even for the facial area of 15 pixels wide (lower right).

6 Conclusions

This paper studies methods to effectively fit an AAM to facial video sequences by using a hybrid model. Both a generic AAM and a subject-specific model are employed simultaneously in the proposed fitting scheme. Borrowing the idea of the SIC algorithm, we also introduce the efficient implementation of the proposed algorithm. Experimental results from outdoor surveillance video sequences demonstrate the improved fitting robustness, accuracy and speed. Future directions of this work can be experimenting with other definitions of the subject-specific model, such as Eq. (6), and as well as investigating the option of dynamically determining the weighting factor k based on the observed video frame.

References

- [1] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework. *Int. J. Computer Vision*, 56(3):221–255, March 2004.
- [2] Johan G. Bosch, Steven C. Mitchell, Boudewijn P. F. Lelieveldt, Francisca Nijland, Otto Kamp, Milan Sonka, and Johan H. C. Reiber. Automatic segmentation of echocardiographic sequences by active appearance motion models. *IEEE Trans. Medical Imaging*, 21(11):1374–1383, 2002.
- [3] K. Chang, K. Bowyer, and P. Flynn. Face recognition using 2D and 3D facial data. In *Proc. ACM Workshop on Multimodal User Authentication*, pages 25–32, December 2003.
- [4] T. Cootes, D. Cooper, C. Tylor, and J. Graham. A trainable method of parametric shape description. In *Proc. 2nd British Machine Vision Conference, Glasgow, UK*, pages 54–61. Springer, September 1991.
- [5] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(6):681–685, June 2001.
- [6] R. Gross, I. Matthews, and S. Baker. Generic vs. person specific active appearance models. *Image and Vision Computing*, 23(11):1080–1093, November 2005.
- [7] Seth C. Koterba, Simon Baker, Iain Matthews, Changbo Hu, Jing Xiao, Jeffrey Cohn, and Takeo Kanade. Multi-view AAM fitting and camera calibration. In *Proc. 10th Int. Conf. on Computer Vision, Beijing, China*, volume 1, pages 511–518, October 2005.
- [8] A. Levey and M. Lindenbaum. Sequential Karhunen-Loeve basis extraction and its application to images. *IEEE Trans. Image Processing*, 9(8):1371–1374, 2000.
- [9] Xiaoming Liu, Tsuhan Chen, and Susan M. Thornton. Eigenspace updating for non-stationary process and its application to face recognition. *Pattern Recognition*, 36(9):1945–1959, 2003.
- [10] Xiaoming Liu, Peter Tu, and Frederick Wheeler. Face model fitting on low resolution images. In *Proc. 17th British Machine Vision Conference, Edinburgh, UK*, volume 3, pages 1079–1088, 2006.
- [11] Iain Matthews, Takahiro Ishikawa, and Simon Baker. The template update problem. In *Proc. 14th British Machine Vision Conference, Norwich, UK*, September 2003.
- [12] P. J. Phillips, H. Moon, P. J. Rauss, and S. Rizvi. The FERET evaluation methodology for face recognition algorithms. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(10):1090–1104, October 2000.
- [13] Frederick W. Wheeler, Xiaoming Liu, Peter H. Tu, and Ralph Hocht. Multi-frame image restoration for face recognition. In *Proc. IEEE Signal Processing Society Workshop on Signal Processing Applications for Public Security and Forensics (SAFE 2007), Washington, DC*, 2007.