Simultaneous tracking of rigid head motion and non-rigid facial animation by analyzing local features statistically

Yisong Chen, Franck Davoine
HEUDIASYC Mixed Research Unit, CNRS,
Compiegne University of Technology, Compiegne, France
ychen@hds.utc.fr,franck.davoine@hds.utc.fr

Abstract

A quick and reliable model-based head motion tracking scheme is presented. In this approach, rigid head motion and non-rigid facial animation are robustly tracked simultaneously by statistically analyzing the local regions of several representative facial features. The features are defined and operated based on a mesh model that helps maintain a global constraint on the local features and avoid the time-consuming appearance computation. A statistical model is computed from a moderate training set that is obtained by synthesizing different poses from a given standard initial image. During tracking, feature-based local distributions are obtained directly from the video frames and the troublesome feature detection or model rendering process is avoided. The observed distribution is compared with the pre-computed statistical model and the tracking is achieved by minimizing an error function based on the maximum likelihood principle. Experimental results show that this tracking strategy is robust to a wide range of head motion, facial animation and partial occlusion. The tracking can be conducted in nearly real-time and is easy to recover from failures.

1 Introduction

Model based 3D head tracking is a continuing research problem in computer vision. Much work has been conducted with the emphasis gradually moving from 2D matching approaches to full 3D model based systems [3, 16]. Different head models in conjunction with advanced techniques have been tested and reported [2, 5, 10, 20].

In recent years, the demand for tracking simultaneously both the global motion and the local motion of a face has received increasing attention [7, 8]. This task poses challenging problems because of the variability of facial appearance within a video sequence. In particular, the tracking problem is formulated as one of estimating the parameters of a deformable model in a high-dimensional state space which best fits the input video sequence [15, 18]. To date, active appearance model based approach seems a promising one and has been adopted by most recent work [6, 11, 21]. Roughly speaking, in appearance model based approaches a textured 3D head model is formed using the initial

BMVC 2006 doi:10.5244/C.20.63

image. During tracking the textured model is fitted to each consecutive frame to find the optimal parameters. The complex light conditions in real-life videos prove a great difficulty for appearance based tracking and often cause the loss of tracking. One possible solution is to construct a model from a large database of the object of different poses and expressions under various illuminations. The target is compared to the model in measuring the error of tracking. However, this approach demands a huge-size training pool and a time-consuming preparation procedure which is generally not tractable in the case of a high-dimensional parameter space [22]. Another approach is to continuously update the initial model during tracking to adapt it to environment changes. This approach suffers from error accumulation and the tracking is likely to drift away from the target and is not easy to recover in a relatively long sequences [7].

With the help of the local features the above problem can be to a large extent alleviated. Previous work has proved that within local regions in space and time, local features are able to provide a concise description of changes of non-rigid motion [4, 12, 17]. In contrast to the global appearance, a good feature can characterize local details reliably and thus is less fragile to pose and lighting changes. Carefully designed local feature-based approaches can effectively treat both global and local motions and are tolerant to illumination changes as well as partial occlusions. However, an obvious drawback of local feature based approach is that the feature detection algorithm is always a big time-consumer, which prevents a quick tracking.

In this paper we suggest a new tracking algorithm that integrates the advantages of local features and global appearances. In this approach, the parameter space is established from a mesh-based face model that encodes conveniently both rigid and non-rigid head motions. The error function in model fitting is defined based on several representative local features that can be easily located and operated with the help of a mesh model. The benefit is twofold: first, the employment of local features helps prevent the object drift away during tracking and make it easy to recover from tracking failures; second, the troublesome feature detection or model rendering process is avoided and hence the tracking can be realized in a quick and reliable manner.

In brief, our framework is composed of a training step and a tracking step. In the training step, we create a moderate training set of images by rendering the different views of a texture mapped 3D model. Afterwards, we select scores of representative feature points from the face model. For each feature, the corresponding local regions in all training images are extracted to form a high-dimensional vector space. We build the statistical model by estimating the corresponding probability distribution parameters under the assumption of multivariate Gaussian distribution. In the tracking step, the error function is formulated as the joint probability distribution observed around all local features and the optimization is achieved by maximum likelihood estimate. This approach is effective compared to most previous analysis-by-synthesis schemes in that under this optimization strategy we do not have to render the model repeatedly and the tracking can be executed much faster. The downhill simplex method is adopted to perform optimization because its optimization strategy automatically maintains multiple hypotheses, which is a most desirable feature in advanced tracking algorithms to ensure robust tracking.

The rest of the paper is organized as follows. Section 2 introduces the face model and the problem formulation. Section 3 demonstrates the training process. Section 4 describes the statistical tracking strategy. Section 5 gives the experimental results as well as some discussions. Finally, Section 6 concludes the paper.

2 Problem formulation

We adopt the *Candide*-3 model as the basis model in our study [1]. This 3D deformable wireframe model was first developed for the purpose of model-based image coding facial animation and has exhibited many desirable properties in tracking applications. The 3D shape of this wireframe model is directly recorded in coordinate form. It is given by the coordinates of the 3D vertices $\mathbf{P}_i, i = 1 \cdots n$, where n is the number of vertices. Thus, different face shapes and animations up to a global scale can be fully described by a $3n$-vector $\mathbf{g}$ using the following equation.

$$\mathbf{g} = \mathbf{g}_s + \mathbf{A}\boldsymbol{\tau_a} \qquad (1)$$

where the $3n$-dimensional column vector $\mathbf{g}_s$ is the standard shape of the model, the $3n \times k$ matrix $\mathbf{A}$ is the animation unit (AU) matrix containing the control coefficients for k animation units under consideration, and the k-dimensional column vector $\boldsymbol{\tau_a}$ contains the corresponding facial animation parameters. Here we consider six facial animation units: lower lip depressor, lip stretcher, lip corner depressor, upper lip raiser, eyebrow lower and outer eyebrow raiser. Even if the proper dimensionality of facial deformation might be more than 6, these units are enough to cover most common facial animations involving mouth and eye movements. The 6-vector $\boldsymbol{\tau_a}$ constitutes the non-rigid facial animation parameters in our study. This model is favored by incorporating knowledge about facial deformation, motion and appearance. In addition, to formulate the global rigid motion of the head, we adopt an enhanced orthographic model. That is, the orthographic projection is supposed and an additional scale factor is introduced to compensate the slight motion in the direction of optical axis. This model decouples the focal length from the other parameters and is a good weak perspective approximation suitable for many applications [9]. Another benefit of the model is that it greatly simplifies the training process and makes a tractable training size possible, as we will show in Section 3.

In summary, the state space in our tracking framework is given by a 12-dimensional vector $\mathbf{b}$:

$$\mathbf{b} = \left[\theta_x \; \theta_y \; \theta_z \; t_x \; t_y \; s_z \; \boldsymbol{\tau_a}^T\right]^T = \left[\mathbf{h}^T \; \boldsymbol{\tau_a}^T\right]^T \qquad (2)$$

where the vector $\mathbf{h}$ represents the six degree of freedom under enhanced orthography associated with the rigid head motion. The AU-based wireframe structure of the *Candide*-3 model provides great conveniences in operating and analyzing several key points that can greatly help the feature based tracking scheme. This is further demonstrated in the next two sections.

3 Training process

For the training process, first we need to select several representative facial features which are robust to scale, pose and lighting changes during tracking. Then we need to obtain a training set that covers most rigid and non-rigid motions. Finally we need to study the distribution of these features to create a statistical model that best describes the rigid and non-rigid face motion under consideration. Related work has validated that a good feature generally has a strong local luminance contrast in its neighboring region. By contrast, feature points in a relatively flat region are relatively fragile to image noise and

Feature No.	Description
1	Outer corner of the left eye
2	Inner corner of the left eye
3	Outer corner of the right eye
4	Inner corner of the right eye
5	Left corner of outer lip contour
6	Right corner of outer lip contour
7	Middle point of outer upper lip contour
8	Middle point of outer lower lip contour
9	Inner corner of the left eyebrow
10	Inner corner of the right eyebrow
11	Outer corner of the left eyebrow
12	Outer corner of the right eyebrow
13	Uppermost point of left outer lip counter
14	Uppermost point of right outer lip counter
15	Left outer lower lip counter
16	Right outer lower lip counter
17	Center of upper outer left eye lid
18	Center of upper outer right eye lid
19	Center of lower outer left eye lid
20	Center of lower outer right eye lid

Table 1: The 20 features adopted in our approach

might fail to work in case of illumination variations. Therefore, priorities should be given to the corner and edge points around which strong illumination gradient can be detected. Moreover, to ensure to cover all rigid and non-rigid motions under consideration, we should select as many features as possible. Considering the above rules, we select 20 features, which are listed in Table 1 and illustrated in Figure 1. From Table 1, we can see that these features are mostly around the major non-rigid motion units (eyes, eyebrows and lips). This makes them capable of tracking facial animation effectively.

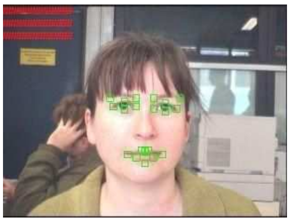


Figure 1: The 20 feature regions in a frontal view

To avoid the problem of time-consuming data collection process, we create our training set by rendering different views with the help of graphics tools from a single frontal view face image. Particularly, we superimpose and align the *Candide* model to the image and then create a virtual 3D face by triangle-wise texture mapping. Then we warp the virtual model by changing the 12-vector of the parameter space to acquire the views of different poses and animations. Note that for a 12-dimensional state space it is not possible to do a dense sampling for each parameter because this will cause a tremendous training set. Therefore, we have to be careful to create those ”most representative” views to maintain a modest training set. Fortunately, the fact that only local features are considered allows us to effectively simplify the problem without significant loss of performance. Firstly, under a normalized enhanced orthographic projection ideally the two translation parameters t_x and t_y will not affect the local region of the features. Secondly, although the different expressions will cause noticeable non-rigid motions in the form of distance changes between different facial features, the influence of the six facial animation parameters is indeed not significant to the local distribution around any of these features. Based on these facts, we only sample three rotation parameters in preparing the training set. In particular, for each of the three rotation parameter, we uniformly take 7 samples around the standard frontal view. Therefore the size of the training set is only $7^3 = 343$. This strategy successfully avoids the problem of dimensionality explosion. Four example images which are synthesized from the initial image in figure 1 are illustrated in figure 2.

The next step is to define a local region around each feature point for future analysis. Here we adopt a similar approach as in [12]. A 9×9 square block is extracted after scale normalization and a 81-variate vector is formed for each feature. This size is verified by our experiments a good trade-off that is big enough to contain necessary grayscale information around a feature and small enough to remove irrelative noise. Finally, for each of the twenty features, with the training set of 343 samples, the 1×81 mean and the 81×81 covariance matrix are computed to build a multiple local feature-based statistical model. This completes the work of training data preparing and analyzing.

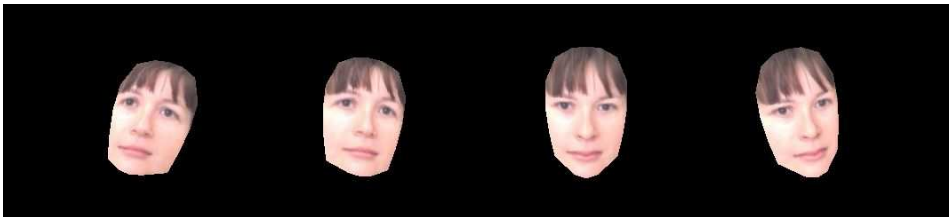


Figure 2: Four example training images

4 Tracking process

The tracking system starts from a real-life face image with a standard frontal view, from which the training data are prepared as demonstrated in Section 3. The initial parameters are computed from the manually aligned candidate model. For the tracking of each successive frame, the tracking framework tries to maximize the similarity between a rendered model image and the target video. This can be deemed as an explicit nonlinear minimization of the error metric over the model parameters [13]. The key to successful tracking is to define an error function which minimizes the distance between the statistical model and the observed distribution. Several different error functions have been suggested in related work, such as pixel difference, mutual information, and correlation ratio [13, 19]. To date all these algorithms are given in the context of a global appearance model and the model has to be virtually rendered for each comparison. This repetitive rendering substantially limits the tracking speed. Here we try to improve the performance by defining a new local feature based error function making full use of the benefits brought about by a feasible model and the pre-computed statistical models. With this strategy we successfully avoid repetitive model rendering and greatly accelerate the tracking process. We reasonably assume that for each of the 20 feature, the corresponding feature vector obeys multivariate Gaussian distribution. We additionally assume that these 20 distributions are independent of each other. With the above assumptions and the preparing work introduced in Section 3, we can easily convert the tracking problem to a maximum likelihood estimate problem and establish a non-linear optimization procedure with the joint probability distribution function over the 20 features as the similarity function to be maximized. More precisely, in evaluating the similarity between an assumption and the target image, the 2D projections of the twenty feature points in the image coordinate can be computed without trouble by mapping the *Candide* model to the video image using the temporary guess of the parameters. Here we do not need to render the textured face model because we are only interested in the 2D projections of the 20 features. Then the 20 blocks around the corresponding feature points are extracted from the target image to form twenty 81-vectors. Treating these 20 vectors as the samples from the associated multivariate Gaussian distributions, we can now calculate the joint probability using the following equation

$$p(\mathbf{X}_t) = \prod_{i=1}^{20} N(\mathbf{x}_i; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \propto \prod_{i=1}^{20} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_i)\right) \qquad (3)$$

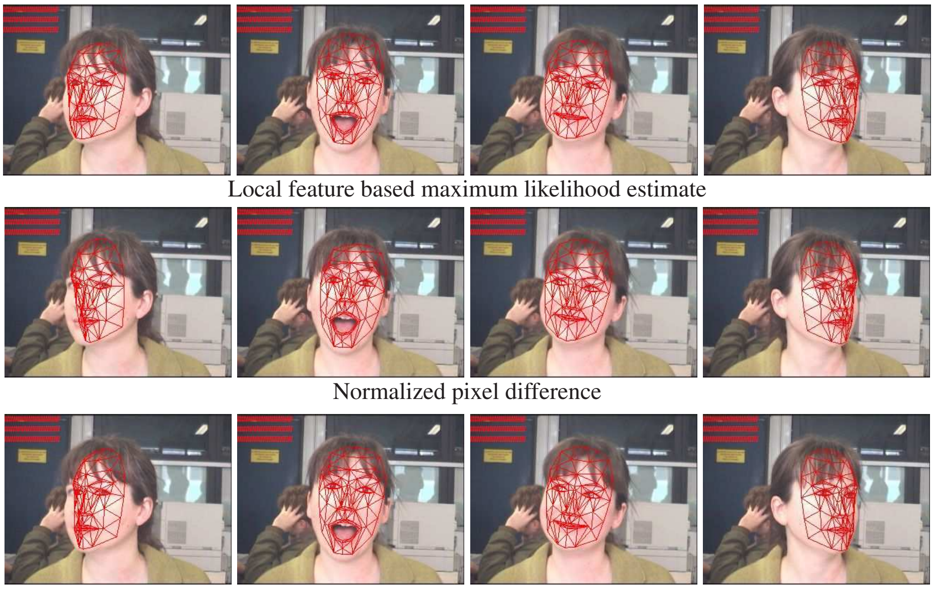
where $N(\mathbf{x}_i; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ indicates the function value at the vector $\mathbf{x}_i$ for the i^{th} multivariate Gaussian distribution with the mean $\boldsymbol{\mu}_i$ and the covariance matrix $\boldsymbol{\Sigma}_i$, which are pre-computed in the training step presented in Section 3. For each single term, the probability is obtained by computing the Mahalanobis distance between the observed value and the center of the distribution. Then this joint probability is considered as the similarity function that has to be maximized. In practice the actual error function computation is formulated by taking the negative log function of the right term of equation (3), which is to be minimized by a nonlinear optimization procedure.

We additionally add a simple but effective mechanism to handle occlusion. Namely, during tracking when the optimal value computed by equation (3) is smaller than a pre-assigned threshold, we judge that an occlusion occurs and the parameters are directly copied from those of the previous frame. With this strategy we expect that the target will not drift too far away when occlusions occur and we have a good chance to relocate the target and the tracking is able to recovered and continued when the occlusion ends.

The major advantage of this tracking strategy is that the 2D projections of all the features can be easily computed and the local distribution can be directly extracted from the video frame under tracking and compared with the pre-computed model. Therefore, the frequent model rendering work that bothers most previous work is totally avoided and the tracking can be achieved in a much faster manner. Note that in this context the multi-dimensional function we wish to optimize is not easily differentiated analytically. So a derivative-free optimizer such as the downhill simplex method is preferred [14]. To be exact, for each tracking step thirteen initial points are randomly selected in the 12-dimensional parameter space around the current state to form a simplex which deforms and contracts during optimization and finally traps the optimum. This optimization strategy automatically endows this approach the desirable property of multiple hypotheses tracking as in particle filtering and makes it an ideal solution in our study.

5 Experimental results, comments and future work

In the first experiment we compare several tracking strategies based on different error functions, namely, the normalized pixel difference based error function, the mutual information based error function, and the maximum likelihood estimate based error function described in this paper [13]. The result is illustrated in Figure 3. The figure shows that all these three approaches are capable of tracking facial expressions in the case of slight rigid motion. Nevertheless, under big head rotation the performance is different. The pixel difference based approach is most fragile to big rotations. The flat part of the cheek is disturbed by the illumination changes and prevents the tracker from correct tracking. The mutual information based approach improves the performance a little but the error is still visible. By contrast, our approach presents the best performance of robustness. All the local features as well as the global pose are faithfully tracked throughout the whole video. This in turn ensures that the strategy of global model-constrained local feature based tracking is a promising approach.



Local feature based maximum likelihood estimate

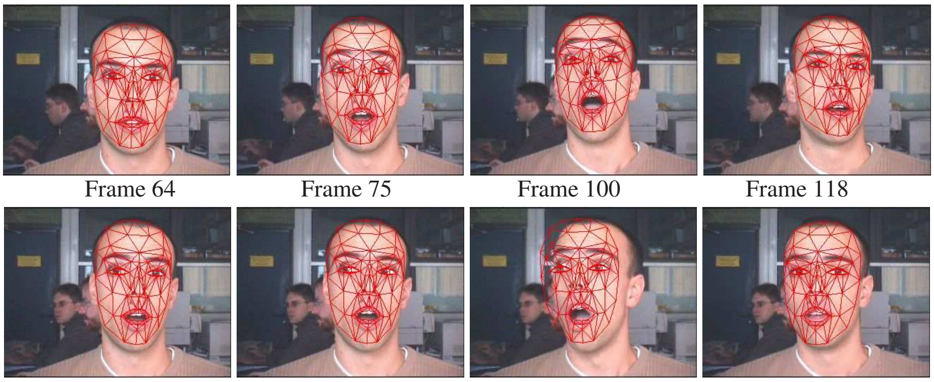
Normalized pixel difference

Mutual information

Figure 3: Tracking results comparison of three different strategies.

More importantly, the pixel difference and the mutual information based approach require frequent model rendering and model-observation comparison, which takes a lot of time. By contrast, our approach avoids these operations because only observations around several features are needed. Therefore, our approach is much faster than the other two. The tracking speed is around 10 frames per second for our approach but only about 0.6 frames per second for the other two.

Figure 4 gives another example for a more difficult video with abundant face animations to illustrate the robustness of the approach. In most time of the tracking the rapid changes of the expressions are faithfully tracked. Occasionally either a couple of local features or the global pose might slightly drift away due to some difficult poses and exag-

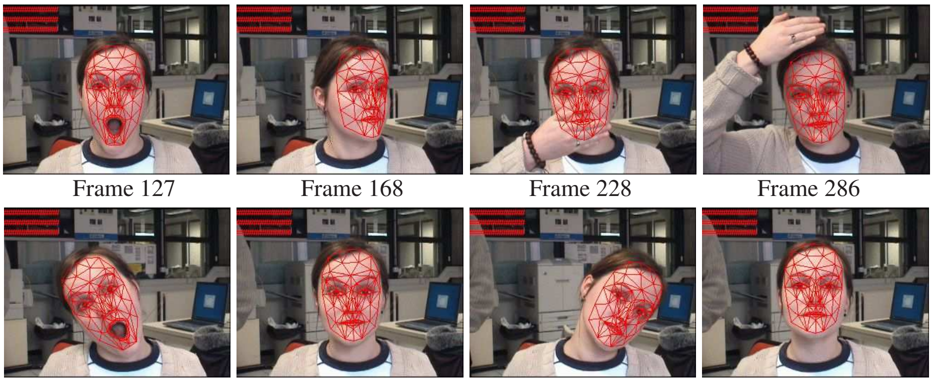


Frame 64 Frame 75 Frame 100 Frame 118

Frame 270 Frame 277 Frame 290 Frame 330

Figure 4: The two lost-reestablished examples. The tracking starts to drift away around frames 75 and 277. The error reaches maximum around frames 100 (eyebrows slightly lost) and 290 (3D pose lost). The tracking is reestablished around frames 118 and 330.

gerate expressions, but the tracking is able to be automatically reestablished after a short while when most of the features can be reliably observed and analyzed again. The capability of reestablishing tracking is due to the downhill simplex algorithm adopted in our tracking framework. The optimization strategy uses multiple samples to approximate the optimal point. This helps to maintain multiple hypotheses, hence makes our algorithm robust to a certain degree of noises and is easy to recover from temporary failures.



Frame 127 Frame 168 Frame 228 Frame 286

Frame 484 Frame 531 Frame 608 Frame 653

Figure 5: Tracking result for a challenging test video.

Figure 5 gives some representative results for a very challenging video that contains dramatic rigid and non-rigid motion (around frames 127, 168, 484, 531, 608, 653) as well as a long-period occlusion (between frame 228 and frame 286). The result is quite encouraging. Throughout the video our algorithm tracks both rigid and non-rigid motions with high accuracy and can always easily recover from temporary failures caused by occlusions or dramatic motions. For example, when an occlusion starts around frame 228, it is detected as an unacceptable small probability and the occlusion treating mechanism starts. When the occlusion comes to an end around frame 286, the target is quickly realigned with the model and the tracking successfully continues.

Our C++ based program is conducted on a PC workstation equipped with a 3.6G P4 processor and 1G RAM. For a 320 × 240 video sequence, our algorithm tracks with a speed of around 10 fps. This nearly real-time speed can satisfy the requirements of many applications.

There are several possibilities for future work. Firstly, the program is still not able to be run in real time for videos of high frame rate of more than 20 frames per second. We are considering some optimization which can speed up the search while maintain the desirable property of multiple hypotheses. Secondly, more animation units or some global constraints can be added to handle richer facial expressions and head motions. However, how to make an efficient search in higher dimensional space remains an open question. Finally, more powerful models and error functions may be established with some alternative training and tracking strategies to handle more difficult cases.

6 Conclusion

In this paper, we have proposed a local feature based stochastic framework for tracking simultaneously rigid and non-rigid human face motion. In this approach we use local features to overcome the drawbacks of existing active appearance model based approach. In implementing this we make full use of a wireframe model to control facial animation and at the same time avoid the trouble of feature detection and model rendering. A statistical model is computed from a carefully prepared training set to help achieve a MLE based similarity function which successfully characterizes the local details as well as the global motions. The downhill simplex method is employed to perform optimization to handle the derivative-free case and maintain multiple hypotheses during tracking, which greatly improves the robustness and the reliability.

References

[1] J. Ahlberg and R. Forchheimer. Face tracking for model-based coding and face animation. *International Journal on Imaging Systems and Technology*, 2003.

[2] S. Basu, I.A. Essa, and A.P. Pentland. Motion regularization for model-based head tracking. In *ICPR*, 1996.

[3] S. Birchfield. Elliptical head tracking using intersity gradients and color histograms. In *ICCV*, 1998.

[4] M. J. Black and Y. Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In *ICCV*, 1995.

[5] M. Cascia, S. Sclaroff, and V. Athitsos. Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models. *IEEE PAMI*, 22:322–336, 2000.

[6] T. Cootes, G. Wheeler, K. Walker, and C. Taylor. Coupled-view active appearance models. In *BMVC*, 2000.

[7] D. DeCarlo and D. N. Metaxas. Optical flow constraints on deformable models with applications to face tracking. *IJCV*, 38(2):99–127, 2000.

[8] F. Dornaika and F. Davoine. Simultaneous facial action tracking and expression recognition using a particle filter. In *ICCV*, pages 1733–1738, 2005.

[9] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2004.

[10] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *ECCV*, pages 343–356, 1996.

[11] A. Jepson, D. Fleet, and T. El-Maraghi. Robust online appearance models for visual tracking. *IEEE PAMI*, 25(10):1296–1311, 2003.

[12] V. Lepetit, J. Pilet, and P. Fua. Point matching as a classification problem for fast and robust object pose estimation. In *CVPR*, pages 244–250, 2004.

[13] J. Paterson and A. Fitzgibbon. 3d head tracking using non-linear optimization. In *BMVC*, pages 609–618, 2003.

[14] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1998.

[15] S. Romdhani, V. Blanz, and T. Vetter. Face identification by fitting a 3d morphable model using linear shape and texture error function. In *ECCV*, 2002.

[16] I. Schodl and A. Haro. Head tracking using a textured polygonal model. In *Workshop on Perceptual User Interfaces*, 1998.

[17] T. Tuytelaars and L. VanGool. Wide baseline stereo matching based on local, affinely invariant regions. In *BMVC*, pages 412–422, 2000.

[18] T. Vetter and V. Blanz. Estimating colour 3d face models from a single image: An example based approach. In *ECCV*, pages 499–513, 1998.

[19] P. Viola and W. Wells. Alignment by maximization of mutual information. In *ICCV*, pages 16–23, 1995.

[20] Y. Zhang and C. Kambhamettu. Robust 3d head tracking under partial occlusion. *Pattern Recognition*, 35(7):1545–1557, 2002.

[21] S. Zhou, R. Chellappa, and B. Mogghaddam. Visual trackingand recognition using appearance-adaptive models in particle filters. *IEEE Trans. on Image Processing*, 13(11):1473–1490, 2004.

[22] Z. Zivkovic and F. van der Heijden. A stabilized adaptive appearance changes model for 3d head tracking. In *ICCV workshop on recognition, analysis, and tracking of faces and gestures in real-time systems*, 2001.