Automatic 3D Face Modeling from Video

Le Xin[1], Qiang Wang[2], Jianhua Tao[1], Xiaoou Tang[2], Tieniu Tan[1], and Harry Shum[2]
[1]National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing 100080, P.R. China
[2]Microsoft Research Asia, Beijing 100080, P.R. China
[1]{xinle, jhtao, tnt}@nlpr.ia.ac.cn, [2]{qiangwa, xitang, hshum}@microsoft.com

Abstract

In this paper, we develop an efficient technique for fully automatic recovery of accurate 3D face shape from videos captured by a low cost camera. The method is designed to work with a short video containing a face rotating from frontal view to profile view. The whole approach consists of three components. First, automatic initialization is performed in the first frame with approximately frontal face. Then, to handle the case of low quality image captured by low cost camera, the 2D feature matching, head poses and underlying 3D face shape are estimated and refined iteratively in an efficient way based on image sequence segmentation. Finally, to take advantage of the sparse structure of the proposed algorithm, sparse bundle adjustment technique is further employed to speed up the computation. We demonstrate the accuracy and robustness of the algorithm using a set of experiments.

1. Introduction

Accurate face modeling has extensive applications in areas such as human computer interaction (HCI), multimedia, and faces recognition [1, 3]. In recent years, a number of approaches have been proposed for 3D face modeling from images [2, 5, 8, 9, 12]. In [8], large angle multiple views are used for accurately recovering shape information. But the system with a manually intensive procedure is far from flexible since the user needs to manually specify point matching across multiple images and 2D-3D feature correspondences.

Blanz and Vetter [2] proposed another impressive approach based on the morphable 3D face model. They can get face model reconstruction from a single image, which demonstrated the advantage of using these linear classes of model. Because of the sensitivity of the texture descriptor to illumination change [3, 5], the quality of shape reconstruction will degrade in uncontrolled illumination. So the texture descriptor was replaced by pair-wise point matching to increase robustness to illumination change in [5].

Model-based bundle adjustment technique proposed by Shan et al. [9] and the newly presented approach by Dimitrijevic et al. [5] inspired our work. In model-based bundle adjustment, prior model knowledge is included into the traditional bundle adjustment. For face modeling, the 3D shape can be reliably recovered [9, 12]. But the approach involves the use of 3D features points whose projection is known and relies on a sparse face mesh structure which is not a sufficient representation of the real face geometry. Dimitrijevic et al. [5] presented a similar bundle adjustment procedure based on the sophisticated PCA based model learned from real 3D face data [2]. Given pair-wise feature correspondences as input, the approach is robust to uncontrolled lighting condition. Especially, they showed that the precision of the reconstructed face model can be predicted as a function of the number and quality of the correspondences.

However, the computational complexity of the algorithm in [5] grows cubically with the number of the frames being processed, which makes it infeasible to process a relatively long sequence. In addition, this approach depends on the quality of point matching between adjacent frames which we know is unreliable in low quality video.

In this paper, we develop an efficient technique for automatic recovery of accurate 3D face models from videos captured by a low cost camera. Based on the segmentation of the whole sequence into segments, 2D point matching, head poses, and 3D face model are iteratively estimated and refined in each segment. In this way, a moderate number of accurate feature matching across images in one segment can be established even under large illumination change. So a final post-processing can be efficiently done over all segments with the appropriate estimation of all the factors. Sparse bundle adjustment technique is further employed to speed up the computation. There are two key contributions of this paper based on the proposed overall technique: 1) a two-layer iterative optimization algorithm for the estima-

Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05)
1550-5499/05 $20.00 © 2005 **IEEE**

tion and refinement of 2D point matching, head poses, and 3D model; 2) a system which provides feasible, efficient, and automatic processing of input video frames to get an accurate face model, even when noise perturbed real video are used as an input.

2. System overview

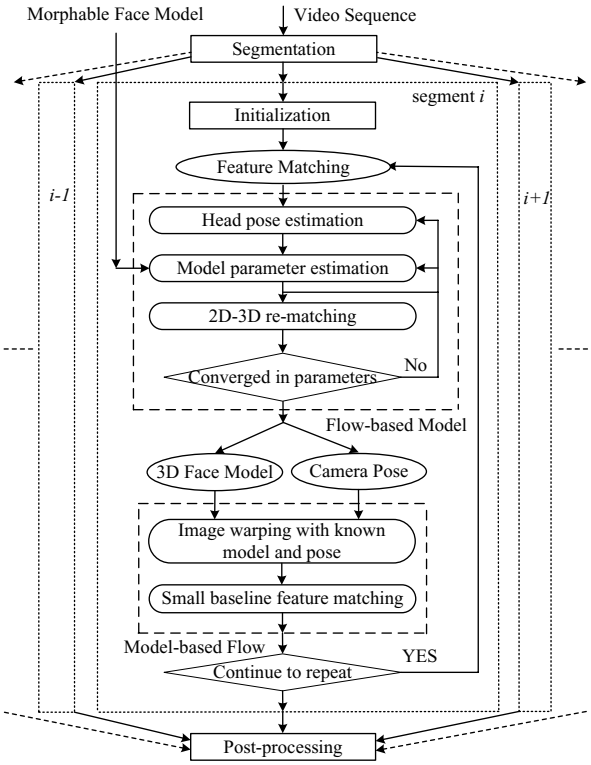


Figure 1. *The block diagram of the proposed algorithm.*

Fig. 1 shows the block diagram of the proposed algorithm. In the system, we use the 3D morphable face model which was proposed in [2]. The input is a video sequence containing a face rotating from frontal view to profile view before a fixed camera. The main concern about face modeling from video here is how to match the generic 3D morphable face model to all frames accurately in an automatic and efficient way. Due to the use of a lot of shape information in the frames from frontal view and profile view, the 3D face shape will be precisely reconstructed. Here, we take the coordinate system of the 3D morphable model in the 3D world to be fixed, which is convenient in our situation that the camera is fixed and the head moves.

After dividing the whole sequence into segments, the computation is processed in all segments recursively as shown in Fig. 1. The computation in each segment consists of two iterative estimation processes which are started after the initialization in that segment. In the first layer iteration, we first estimate and refine the motion parameters of each input face image and the 3D face shape in the flow-based model module (see Sec. 3) given the rough feature matching results. The optimization in this module integrates batch processing and model-based bundle adjustment, which results in robust 3D information recovery. Then, the feature matching will be refined in the model-based flow module (see Sec. 4) with known underlying 3D face model and camera poses for each frame. By pre-warping image pairs into small baseline image pairs and searching under geometrical constraints, a better feature matching result can be obtained. In this way, 2D point matching, head poses, and 3D face model are iteratively estimated and refined. As a result, consistent correspondences across images in one segment can be obtained accurately. In the second layer iteration, the 3D face geometry and the motion parameters (head poses) for each frame are estimated efficiently in the flow-based model module. Thus, when expanding to the whole sequence, the feature matching can be efficiently estimated across all frames with the adaptive refinement of face shape. The refined feature correspondences for each segment are combined together in the last post-processing step to further refine the 3D face model by using the same two layers iterative estimation and refinement procedure as the one used in each segment.

2.1. Video sequence segmentation

Because of self-occlusion and feature detection failure, a surface point can only be observed and detected in a subset of images. Thus, we should divide the whole sequence into several segments. Note that there is one overlapped frame between successive segments. The number of frames in one segment depends on the speed of object movement and illumination changes in order to get enough feature correspondences to recover the 3D model and camera poses robustly. With more than two frames tracked at one time in each segment, the recovered 3D information using batch processing in the second layer iteration will be more reliable comparing with that obtained from consecutive pairs. Furthermore, accurate and reliable correspondences across images will be established under our two layer iteration. Consequently, the pose parameter computed from the previous segment in the first frame of the current segment (the last frame of the previous segment) is reliable to continue feature tracking, which will be more efficient than the recursive processing of the sequence with images added at both ends iteratively as in [5].

2.2. Initialization

To make the algorithm fully automatic, we need to automatically get initial pose for the first frame of each seg-

ment. For the first segment, the first frame contains an approximately frontal face. We automate the initialization by detecting the face region using a face detector [11] and extracting the salient face features using the face alignment algorithm [13]. The semantic 2D-3D feature correspondences are then established and we use the POSIT algorithm [4] to get an approximate initial face pose. For other segments, the pose in the first frame is already known from the estimation result of previous segment. Note that the face detector and alignment is never used again in the following process.

In each segment, the good features are selected in the first frame and then the KLT algorithm [10] is applied to give an initial feature matching result.

3. Flow-based model

In this section, we will show that we can recover the 3D face geometry and the motion parameters for each frame under the perspective projection camera model in the flow-based model module efficiently.

3.1. Problem formulation

First, a 3D morphable face model is constructed using the USF Human ID 3-D database, which includes 100 laser-scanned heads [2]. Each face model in the database has approximately 70,000 vertices. In our work, the number of the vertices is reduced to about 9,000 for better performance, which is still a fine approximation to the ground truth 3D face surface. The triangulated mesh structure of a 3D face is represented by a shape vector, $S = (V_1^T, ..., V_N^T)$, where $V_i (i = 1, ..., N)$ are the vertices of the mesh, thus S is obtained by concatenating the X, Y and Z coordinates of all its vertices. Then a new face shape S can be expressed as:

$$S = \bar{S} + \sum_{k=1}^{r} \alpha_k S_k, \tag{1}$$

where $\bar{S}$ represents an average face model, S_k are orthogonal vectors, and α_k are scalar per orthogonal-vector weights that indicate the contributions of the shape deformation to each shape. So a face is a surface defined by a set of r parameters, denoted by $b = \{\alpha_k | k = 1, ..., r\}$, called model parameters. In our work, the number of model parameters r is 50, so a small number of observation data are enough to compute the model parameters of the eigenvectors, and then the 3D face shape is created using those parameters.

As shown in the system overview in Sec. 2, features are selected and tracked in n frames at one time in each segment. Here we will set $n = 3$ without lose of generality. Thus, we have corresponding feature sets $p_{j,0}, p_{j,1}, p_{j,2}$ for the three frames, where in $p_{j,i}$, j is an index over 3D points, and i is an index over frames. We can compute a 3D point S_j in the face surface by back-projecting $p_{j,0}$ based on the initial pose estimates M_0 in the first frame. The 3D point S_j is on the l-th planar facet in the triangulated mesh structure of 3D morphable face model. With the correct camera poses M_1, M_2 in the last two frames, the $p_{j,1}, p_{j,2}$ can be predicted based on the 3D point. Since the face shape is a triangular mesh, any point on a triangle is a linear combination of the three triangle vertexes, which is functions of model parameters, and any point on a triangle is also a function of model parameters. So the optimization function in one tracking segment should be formulated as:

$$F_1 = min \sum_j \Psi(p_{j,0}, p_{j,1}, p_{j,2}, M_0, M_1, M_2, b)^2, \tag{2}$$

where Ψ is the essential optimization function for one feature matching result in this segment, which can be implemented using the re-projection constraint. Here we assume that the index of the planar facet is not changed when the 3D point S_j is refined with the model parameters b until Sec. 3.2.3.

In practice, we use the re-projection error in the second image $\sum_j d(\widetilde{p}_{j,i}, H\widetilde{p}_{j,j+1})^2$ as our unit cost function which may be minimized in order to estimate parameters for over-determined solutions. Here

$$H = A_{i+1}(R_{i,i+1} - t_{i,i+1} \cdot \bar{n}^T / d) A_i^{-1}, \tag{3}$$

is the general expression for the homography [6] induced by the plane for two views defined by their relative motion. $R_{i,i+1}, t_{i,i+1}, A_i$ and A_{i+1} are the camera intrinsic parameters; $\pi = (\bar{n}^T, d)^T$ is the parameter the plane has; $\widetilde{p}$ is the homogenous coordinates of p; $\propto$ denotes equality up to a scale.

Because our simple way to perform the re-projection is equivalent to assuming that the points in the first frame are noise-free, the point matching of two image pairs in one segment are allocated as $p_{j,0} \leftrightarrow p_{j,1}$ and $p_{j,0} \leftrightarrow p_{j,2}$ instead of as correspondences in the consecutive pairs. The location of $p_{j,0}$, determined by the process of good feature selection, is more robust than that of $p_{j,1}$ and $p_{j,2}$. So the optimization function in one tracking segment should be formulated as

$$F_1 = \sum_j d(\widetilde{p}_{j,0}, H_{j,1}\widetilde{p}_{j,1})^2 + d(\widetilde{p}_{j,0}, H_{j,2}\widetilde{p}_{j,2})^2. \tag{4}$$

In the prior shape model, not all possible values of model coefficients are acceptable. Based on the PCA dimensionality reduction algorithm, it is necessary to impose constraint making parameters subject to bounds: $|\alpha_k| < 3\sigma_k$, where σ_k is the k-th eigenvalue. In practice, we add the regularization term σ' instead of using some constrained optimization techniques. The final function to be minimized in the IRLS (iterative reweighed least square) way is

$$F_1 + \sigma'^2 \sum_{k=1}^{r} \frac{\alpha_k^2}{\sigma_k^2}, \tag{5}$$

where σ' is adaptively determined for better performance (see Sec. 3.2.2).

In this way, the optimization function can be used in each segment of the whole sequence recursively with the feature matching refined in the following model-based flow module.

3.2. Iterative parameters estimation

Here, the 3D face geometry and the six free degree motions for each frame are estimated efficiently by the iterative processing.

3.2.1. Camera poses estimation. It is well known that the camera pose can be estimated reliably before an accurate face model has been obtained. Moreover, the face geometry will not have much discrepancy in the proximate segment because of the recursive processing of all segments. Thus, the optimal value of each camera pose in each segment can be computed as:

$$(\hat{M}_0, \hat{M}_1, \hat{M}_2) = \arg \min_{M_0, M_1, M_2} F_1. \tag{6}$$

As demonstrated in [5], the accuracy of the 3D shape estimation from homography constraints will increase with the number of correspondences. But the computation requirement of the nonlinear estimation problem increases quickly at the same time. At last, in our post-processing stage, the overall processing time will be long for a not very long sequences.

The major computation cost of our implementation under Levenberg-Marquardt optimization framework comes from the computation relative to Jacobi matrix (J and $J^T J$). Fortunately, the computation of J relative to different pose parameters is independent, so J is a matrix having high sparse structure [7], such as:

$$\begin{bmatrix} J_{0,1}^0 & J_{0,1}^1 & 0 & 0 & 0 \\ J_{0,2}^0 & 0 & J_{0,2}^2 & 0 & 0 \\ 0 & 0 & J_{2,3}^2 & J_{2,3}^3 & 0 \\ 0 & 0 & J_{2,4}^2 & 0 & J_{2,4}^4 \end{bmatrix} \tag{7}$$

. In (7), we show the sparse structure of J when tracking 5 frames in two segments and 3 frames in one segment. This will be done over all two segments for further face model refinement in the post-processing. For one segment, the sparse structure is similar. The simple using block indicator under matrix multiplication for early exiting will speed up the computation greatly. With this speeding up process, our two layer iterative process can be carried through efficiently.

3.2.2. Model parameter estimation. Given the estimated camera poses, the model parameter can be estimated more reliably. For better smoothness in the surface recovery in the estimation of the face shape geometry, the regularization term is also required. The optimal value of model coefficients can be computed as:

$$\hat{b} = \arg \min_b (F_1 + \sigma'^2 \sum_{k=1}^{r} \frac{\alpha_k^2}{\sigma_k^2}), \tag{8}$$

where the regularization term σ' is determined adaptively as follows:

$$\sigma_k^2 = \frac{F_1/m}{\sum_{k=1}^{r} \frac{\alpha_k^2}{\sigma_k^2}/r}, \tag{9}$$

where m is the total number of the feature matching in this segment.

We can see clearly that the regularization term is used to normalize the dimension size of the two terms when minimizing the objective function. In practice, the adaptive regularization term is applied after the objective function is decreased in the first several optimization iterations. And for keeping up the power of the regularization term in smoothing face surface, we set the minimal value for it in our experiment.

3.2.3. 2D-3D re-matching. With the change of 3D face geometry, the relation about the 2D-3D correspondences, the index of the corresponding planar facet, which is assumed known in Sec. 3.1, should be changed accordingly. In this way, the convergence of the approach can be guaranteed. In practice, this stage is processed alternatively with the first two stages for stable performance of convergence.

4. Model-based flow

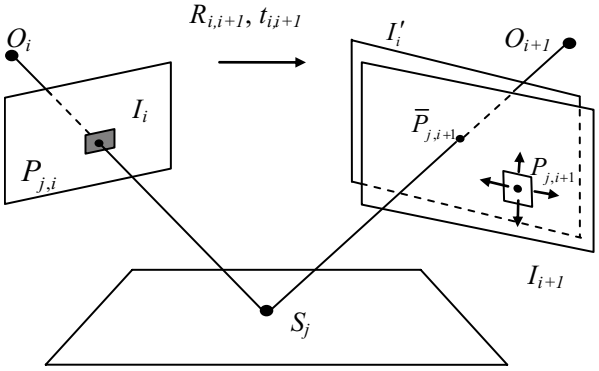


Figure 2. *Model-based feature matching.*

Since the image features are constrained by the geometry of the underlying 3D face shape, so we can use such constraints to handle the perspective distortion on the correction window when there is large degree of rotation in referenced images. The matching between relatively wider

baseline images reduces the total number of features and thus improves the algorithm's efficiency.

The geometrically constrained feature matching process for refining feature correspondence in image I_i and I_{i+1} is shown in Fig. 2. After the optimization in flow-based model step, we get a rough 3D face model and head poses for each frame. With the dense mesh structure of our reconstructed face model, the image I_i is first pre-warped to I_i'. Thus small baseline feature matching can be processed in the new image pair I_i' and I_{i+1}. The location of the feature image window $\bar{P}_{j,j+1}$ in the frame I_{i+1} can be predicted based on transfer relation between $P_{j,i}$ and $P_{j,i+1}$ under the same back-projected 3D points S_j in the refined 3D model. Then a block matching search is performed in its neighborhood in I_{i+1}, denoted as confidence region, based on the new narrow baseline pair I_i' and I_{i+1}. Since the block matching can be done approximately over integer image coordinates, no image interpolation is required, and the resulted operation is extremely efficient.

5. Experiments

To test the performance of our proposed algorithm, real video sequences and noise perturbed real video sequences were utilized. For real video sequences, we show the accurate reconstruction of our algorithms. Perturbed real video testing reflects the robustness of the algorithm to high noise level.

5.1. Real videos captured using USB camera

Fig. 3 and Fig. 4 show the experimental results on the real video sequences. The sequence used in Fig. 3 is the shared sequence in [12], and that used in Fig. 4 is the sequence captured in uncontrolled lighting condition in normal office environment using a low cost USB camera, when head movement is not restricted for the comfort of the subject. In all these real situation examples, the sequences are obtained when a person turns his head in front of a static camera. The typical sequence contains 22 to 23 images of resolution 640*480. The first example shown in Fig. 3 has 22 frames, and the second shown in Fig. 4 has 23 frames.

In the experiment shown in Fig. 3, four images are processed in one segment at one time. In each segment, 500 feature points are selected in the first frame and tracked until the final frame of that segment is reached. Then the corresponding features across all frames are used for the shape model parameters and head motion parameters recovery. The known values of parameters computed in previous segment are used for setting the initialization values of the non-linear optimization problem (Eq.(5)) in current segment. In experiments shown in Fig. 4, the specification is changed to selecting and tracking 300 feature points in three images of one segment. In all these experiments, our iteration incorporating flow-based model and model-based flow for the refinement of point matching, head poses and 3D face model is started after all frames are processed, only using the flow-based model step for better efficiency without sacrificing accuracy.

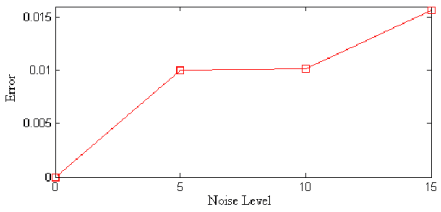
In Fig. 3 and Fig. 4, left column contains three captured images having different degree of head rotation, approximately 0°, 45°, and 90°, respectively. The other images show the experimental results: the projection of the recovered face models on the face images based on the estimated camera pose parameter for these images only after flow-based model tracking step (second column), after three time flow-based model and model-based flow iteration (third column), the shaded views of the reconstructed face model in the same pose (fourth column) and the textured views of the reconstructed face model in the same pose (right column).

We can see from the projection of the recovered model overlaid on the images, especially the occluding contour, that the accuracy of the models progressively increases with the using of multiple images and the iterative process. And the mouth region in profile view illustrate this more clearly. For notability, we intentionally indicate the projection of the 37 salient points in each overlaid image.

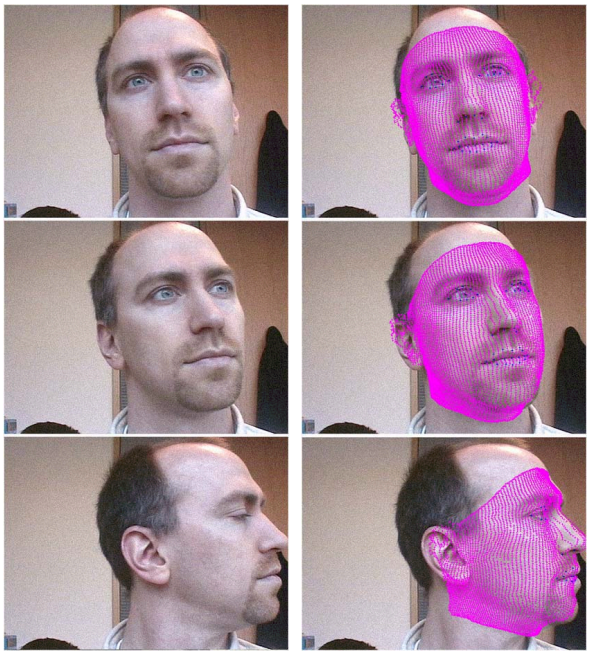
In a non-optimized implementation, the total running time of three time iteration of flow-based model and model-based flow step and the estimation of rough feature correspondences across all image pairs are efficiently estimated by the flow-based model step is about 8 minutes on a 1.3GHz CPU and 256M Memory Pentium 4 machine for the sequence in Fig. 3. For the sequence in Fig. 4, the total running time is about 6 minutes on the same machine.

5.2. Noise Perturbed real videos

Because feature matching is included in our overall framework, we use the noise perturbed real videos to demonstrate the robustness of our algorithm for low quality image data. Here, the video sequence shared from [12] is added with noise with the standard deviation 5%, 10% and 15% of the range of gray value. Fig. 5 shows the experimental results of perturbed video with different noise level. The same parameter settings of Fig. 3 is used. The first row shows the comparison between the reconstructed models in noise-added sequences and the noise-free case in Fig. 3. The horizontal axis is the standard deviation of the added noise. The vertical axis is the difference between the reconstructed model from noise-added video and reconstructed model from noise-free video, which is normalized by the 3D size of the reference model. We plot the average value of difference of all model points in Fig.5(a). Other figures in Fig.5 show the projection of the final reconstruction results from perturbed video with noise level of 15% range of gray value. It shows that our approach is robust to high noise level in the low quality video data.



(a)



(b) (c)

Figure 5. *Experimental results from noisy sequence. (a) the normalized difference between the reconstruction model from noise-added video and noise-free video with increasing noise level. (b) Three images of the same face in Fig. 3 perturbed with noise level of 15% range of gray value in different head rotation. (c) The projection of the recovered model overlaid on the images.*

6. Conclusions

In this paper we have developed an efficient technique for automatic and accurate recovery of the 3D face shape from videos captured by a low cost camera. Through segmentation of the whole sequence into segments, 2D feature matching, head poses, and 3D face model are iteratively estimated and refined in an efficient way. The method produces reliable and high quality feature matching under large illumination change. Thus, a final post-processing can be efficiently done over all segments. The whole process is efficient, automatic, and the result achieved is accurate even for low quality video.

References

[1] C. Beumier and M. Acheroy. Automatic face authentication from 3D surface. In *Proc. British Machine Vision Conference*, pp.449-458, 1998.

[2] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proc. SIGGRAPH*, pp.187-194, 1999.

[3] V. Blanz and T. Vetter. Face recognition based on fitting a 3D morphable model. *IEEE PAMI*, 25(3):1505-1518, 2003.

[4] D. DeMenthon and L. S. Davis. Model-based Object Pose in 25 Lines of Code. In *Proc. ECCV*, pages 335-343, 1992.

[5] M. Dimitrijevic, S. Ilic, and P. Fua. Accurate Face Models from Uncalibrated and Ill-Lit Video Sequence. In *Proc. IEEE CVPR*, pages 188-202, 2004.

[6] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2000.

[7] M. Lourakis and A. Argyros. The Design and Implementation of a Generic Sparse Bundle Adjustment Software Package Based on the Levenberg-Marquardt Algorithms. *ICS/FORTH Technical Report TR-340*, 2004. Available from http://www.ics.forth.gr/ lourakis/sba.

[8] F. Pighin, R. Szeliski, and D. Salesin. Modeling and animation realistic faces from images. *IJCV*, 50(2):143-169, 2002.

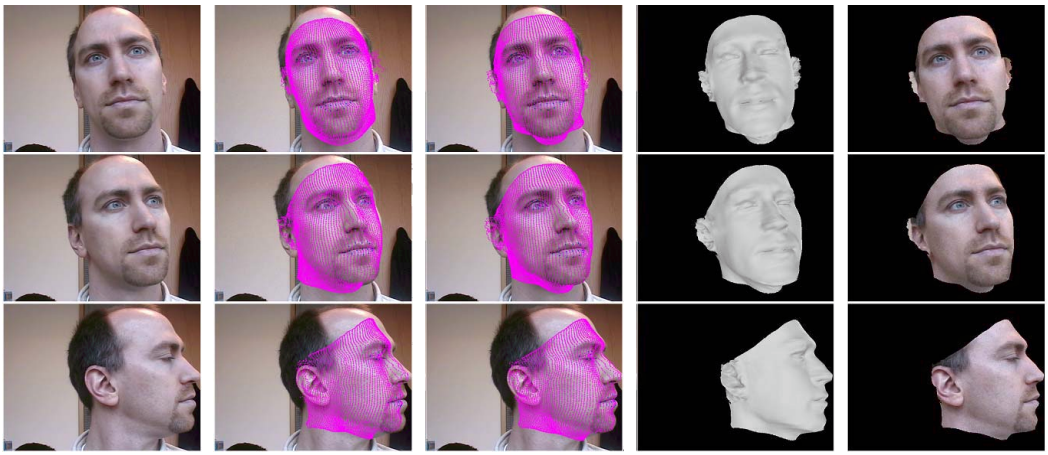
[9] Y. Shan, Z. Liu, and Z. Zhang. Model-based Bundle Adjustment with Application to Face Modeling. In *Proc. ICCV*, pages 644-651, 2001.

[10] J. Shi and C. Tomasi. Good Features to Track. In *Proc. IEEE CVPR*, pages 593 - 600, 1994.

[11] R. Xiao, L. Zhu, and H.J. Zhang. Boosting Chain Learning for Object Detection. In *Proc. ICCV*, pages 709-715, 2003.

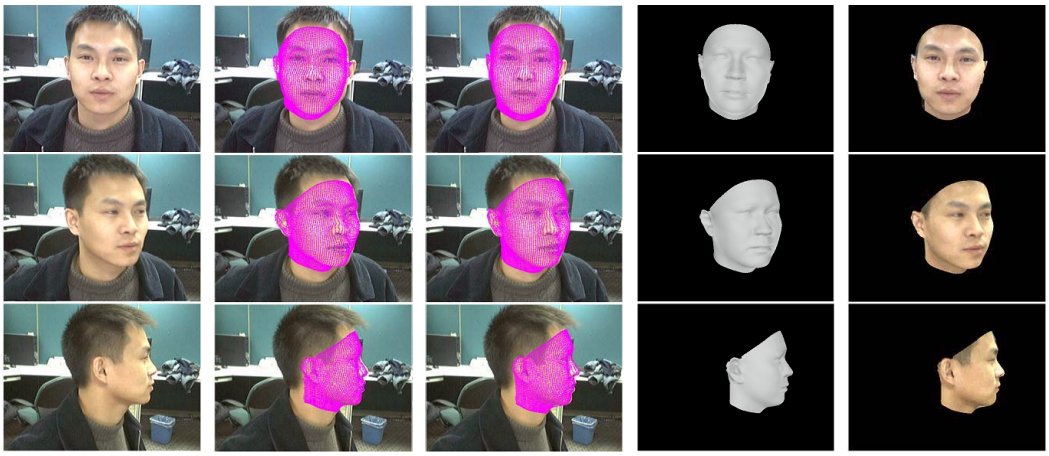
[12] Z. Zhang, Z. Liu, D. Adler, M.F. Cohen, E. Hanson, and Y. Shan. Robust and Rapid Generation of Animated Faces from Video Images: A Model-Based Modeling Approach. *Technical Report MSR-TR-01-101*, Microsoft Research, 2001.

[13] Y. Zhou, G. Lie, and H.J. Zhang. Bayesian tangent shape model: Estimating shape and pose parameters via Bayesian inference. In *Proc. IEEE CVPR*, pages 16-22, 2003.



(a) (b) (c) (d) (e)

Figure 3. *Experimental results of sequence 1 shared from [12]. From the left column to the right column: (a) Three captured images having different head rotation from frontal view to profile view. (b) The projection of the recovered face models on the face images based on the estimated camera pose parameters for these images after the flow-based model step. (c) The projection of the face models after three time iteration of flow-based model and model-based flow step. (d)The shaded views of the reconstructed face model in the same pose. (e) The textured views of the reconstructed face model in the same pose.*



(a) (b) (c) (d) (e)

Figure 4. *Experimental results of sequence 2 captured from a USB camera. From the left column to the right column:(a) Three captured images having different head rotation from frontal view to profile view. (b) The projection of the recovered face models on the face images based on the estimated camera pose parameters for these images after the flow-based model step. (c) The projection of the face models after three time iteration of flow-based model and model-based flow step. (d) The shaded views of the model in the same pose. (e) The textured views of the face model in the same pose.*