Face Model Reconstruction via Camera Pose Estimation Under Orthographic Projection

Tzer-Yih Wu [1,2], Chu-Song Chen [1], Kuo-Young Cheng [1,2]

[1] Institute of Information Science, Academia Sinica, Taipei, Taiwan
[2] Department of Computer Science and Information Engineering, National Taiwan University

Email: song@iis.sinica.edu.tw

Abstract

In this paper, we developed a systematic approach for the reconstruction of a 3D mesh model of a human face from multiple images in a semi-automatic way. Our approach can be divided into three steps: (1) estimation of the camera poses from multiple images under the assumption that the camera projection model is orthographic, (2) computation of the 3D coordinates of a set of control points with multi-view stereo, and (3) reconstruction of a mesh model of human face by deforming a generic model to be fit with the control points. Experiments results show that our method is effective for reconstruction of human face models.

Keywords: *Face-model Reconstruction, Orthographic Projection, Camera Pose Estimation, Stereo Vision, Mesh Deformation.*

1 Introduction

Reconstruction of human face models is an important problem in computer graphics and virtual reality due to many of its promising applications. In the past, approaches for face model reconstruction can be divided into two classes. The first class is the *bottom-up* approach. In this class of approaches, the connectivity is recovered along with geometry. Among them, many researchers concentrate on stereo methods. Lengagne et al. proposed a method combining stereo vision and differential constraints [10][11]. Sara builds a system that four cameras are available for stereo reconstruction [13]. Chen et al. added smoothness constraints in estimating disparity surface [3]. In addition, shading technique and is available too [12]. The second one is the *top-down* approach that the connectivity information is treated as prior knowledge. In [5][6], video stream is used to reconstruct head and natural expressions. Ho, et al. developed an approach for reconstruction of the face model from a single image [8]. The animation results of Guenter et al. [7] are impressive for animation. Delingette and Montagnot proposed a method for deforming face model by a cloud of 3D points [4]. Modeling has also been done from picture data by detecting features, modifying a given generic model and then mapping texture on it [1][9].

In this paper, we developed a semi-automatic approach for face model reconstruction from multiple images of a human head. To reconstruct the 3D structure from multiple images, a fundamental problem is to estimate the camera pose for each image. In this paper, based on the method of Tomasi and Kanade [14], we develop a modified approach for camera pose estimation under the assumption that the camera model is orthographic. Our approach is more computationally stable because that (i) we propose a linear method to give a good initial estimation for nonlinear optimization, and (ii) we propose a new missing-feature recovering process for handling the partial occlusion problem. Once the camera pose of each image has been estimated, our approach reconstructs the human face model in a two-phase procedure. In phase 1, a set of 3D coordinates of the selected control points are obtained via multi-view stereo. In phase 2, a face model is reconstructed by deforming a generic one so as to fit the set of control points obtained in phase 1.

2 Camera Pose Estimation under Orthographic Projection

Estimation of the camera parameters from multiple images and reconstruction of the 3D structures of the objects contained in the scene (referred to as *structure from motion*) is a central issue in 3D computer vision. If the camera has not been calibrated in advance, its parameters have to be estimated directly from the image correspondences among a sequence. In this paper, the camera model used for 3D reconstruction is assumed to be orthographic. Orthographic projection is a good approximation of the camera model when the depth variation of the object contained in the scene image is small compared to the average distance from the object to the camera. In our work, some (usually five) photographs of a human head are taken with a zoom-in camera in the positions which distances to the human head are approximately the same. Hence, it is reasonable to adopt such a camera model for 3D reconstruction in our application. The camera calibration problem can also be simplified by using an orthographic camera model because no intrinsic camera parameters (but only the camera poses) have to be estimated.

2.1 Review of The Factorization Method

We first review the method proposed by Tomasi and Kanade [14] developed for camera pose estimation under orthographic projection. Given P feature points being tracked over F frames in an image sequence, $\{(u_{fp},v_{fp}) \mid f = 1, .., F, p = 1, .., P\}$. The centering coordinates of the feature points in each image frame is defined as $\{(u_{fp}-a_f, v_{fp}-b_f) \mid f = 1, ..,F, p = 1, ..,P\}$, where

$$a_f = \frac{1}{P}\sum_{p=1}^{P} u_{fp} \text{ , and } b_f = \frac{1}{P}\sum_{p=1}^{P} v_{fp} \text{ .}$$

Consider the $2F \times P$ matrix $\widetilde{W}$, the measurement matrix, defined as follows:

$$\widetilde{W} = \begin{bmatrix} \widetilde{U} \\ \widetilde{V} \end{bmatrix},$$

where $\widetilde{U}$ and and $\widetilde{V}$ are the $F \times P$ matrices whose entries of the fth row and pth column are $u_{fp}-a_f$ and $v_{fp}-b_f$, $f = 1, ..,F, p = 1, ..,P$, respectively.

Assume that the world coordinate system is selected according to that its origin is the centroid of the P feature points in the 3D space and that its X and Y axes are parallel to the row direction and the column direction of the first frame of the image sequence, respectively. Denote that the 3D coordinates of the P feature points to be $S_1, S_2, .., S_p$, and let i_n and j_n ($n = 1, .., F$) be the vector parallel to the row and the column directions of the n-th frame with respect to the world coordinate system, respectively. Then, $\widetilde{W}$ can be decomposed as $\widetilde{W} = RS$ where R is a 2F×3 matrix and S is a 3×P matrix as shown below:

$$R_{2F\times 3} = \begin{bmatrix} i_1^T \\ \cdot \\ \cdot \\ \cdot \\ i_F^T \\ j_1^T \\ \cdot \\ \cdot \\ \cdot \\ j_F^T \end{bmatrix} \qquad \text{and} \qquad S = [S_1, S_2, .., S_p].$$

In particular, R has to satisfy the following two constraints.

(1) Constraints of the first frame:

$$i_1 = [1\ 0\ 0]^T \text{ and } j_1 = [0\ 1\ 0]^T, \tag{2.1}$$

(2) The metric constraints:

$$\|i_f\| = \|j_f\| = 1 \quad \text{and} \quad i_f^T j_f = 0 \quad \text{for} \quad f = 1...F \text{ .} \tag{2.2}$$

In fact, the constraint (2.1) can be ignored in the beginning. It is because that if we have found a solution (R, S) satisfying both $\widetilde{W} = RS$ and the constraint (2.2) but $i_1 \neq [1\ 0\ 0]^T$ or $j_1 \neq [0\ 1\ 0]^T$, then i_1, j_1 and $i_1 \times j_1$ (where "×" is the cross product) still forms the three axes of a 3D Euclidean coordinate system which can be transformed via a unique rotation, namely R_o, of that of the world coordinate system defined above. This rotation, R_o, can then rotate i_1 to $[1\ 0\ 0]^T$ and j_1 to $[0\ 1\ 0]^T$.

Hence, we can concentrate on the problem of finding the solution (R, S) satisfying both $\widetilde{W} = RS$ and the constraint (2.2). Notice that the rank of $\widetilde{W}$ is at most three because the ranks of both R and S are three. However, in practice, the rank of $\widetilde{W}$ is usually more than three due to image noise. By applying the singular value decomposition to $\widetilde{W}$ and only keeping the largest three singular values, we can obtain a good approximation of $\widetilde{W}$ by decomposing $\widetilde{W}$ into a $2F \times 3$ matrix $\hat{R}$ and a $3 \times P$ matrix $\hat{S}$, i.e., $\widetilde{W} \cong \hat{R}\ \hat{S}$. Notice that $\hat{R}$ and $\hat{S}$ are of the same sizes of R and S, respectively. However, $\hat{R}$ and $\hat{S}$ may not satisfy the constraint (2.2). In [14], it is clarified that R (or S) is a linear transformation of $\hat{R}$ (or $\hat{S}$) as shown in the following:

$$R = \hat{R}Q \qquad \text{and} \qquad S = Q^{-1}\hat{S} \tag{2.3}$$

Here, Q is a 3×3 matrix transforming the matrix $\hat{R}$ (or $\hat{S}$) to the true one R (or S). Tomasi and Kanade [14] suggest to changing the problem of factorization of $\widetilde{W}$ into R and S to the problem of finding Q in (2.3). By applying the constraint equation of (2.2) and (2.3), they formulate $3F$ equations in terms of the nine unknowns contained in Q and thus induce an over-constrained equation system when F>3 as shown in the following.

$$\begin{aligned} \hat{i}_f^T QQ^T \hat{i}_f &= 1 \\ \hat{j}_f^T QQ^T \hat{j}_f &= 1 \qquad \text{for all } f = 1...F \text{ ,} \\ \hat{i}_f^T QQ^T \hat{j}_f &= 0 \end{aligned} \tag{2.4}$$

where $\hat{i}_f$ ($f = 1, .., F$) are the first F rows of $\hat{R}$, and $\hat{j}_f$ ($f = 1, .., F$) are the last F rows of $\hat{R}$. Notice that this equation system yields a highly *nonlinear* data-fitting problem because it consists of the quadratic terms of the entries of Q.

2.2. Linear Solution to the Factorization Method

A major problem occurred in the method introduced above is that it requires to solve a highly nonlinear equation system which is difficult to be solved in a stable way. In this section, we develop a method which can get the solution of the data-fitting problem with the same constraints shown in (2.2) and (2.3) by simply solving a *linear* equation system.

Define a 3×3 symmetric matrix Φ to be

$$\Phi = QQ^T \qquad (2.5)$$

Because Φ is symmetric, only six variables are required to be estimated. Rewrite (2.4) with the entries of Φ yields the following equation system:

$$\Phi_{11}\hat{i}_{f1}^{\,2} + \Phi_{22}\hat{i}_{f2}^{\,2} + \Phi_{33}\hat{i}_{f3}^{\,2} + 2\Phi_{12}\hat{i}_{f1}\hat{i}_{f2} + 2\Phi_{13}\hat{i}_{f1}\hat{i}_{f3} + 2\Phi_{23}\hat{i}_{f2}\hat{i}_{f3} = 1$$

$$\Phi_{11}\hat{j}_{f1}^{\,2} + \Phi_{22}\hat{j}_{f2}^{\,2} + \Phi_{33}\hat{j}_{f3}^{\,2} + 2\Phi_{12}\hat{j}_{f1}\hat{j}_{f2} + 2\Phi_{13}\hat{j}_{f1}\hat{j}_{f3} + 2\Phi_{23}\hat{j}_{f2}\hat{j}_{f3} = 1$$

$$\begin{aligned}&\Phi_{11}\hat{i}_{f1}\hat{j}_{f1} + \Phi_{22}\hat{i}_{f2}\hat{j}_{f2} + \Phi_{33}\hat{i}_{f3}\hat{j}_{f3} + \Phi_{12}(\hat{i}_{f1}\hat{j}_{f2} + \hat{i}_{f2}\hat{j}_{f1}) + \\ &\Phi_{13}(\hat{i}_{f1}\hat{j}_{f3} + \hat{i}_{f3}\hat{j}_{f1}) + \Phi_{23}(\hat{i}_{f2}\hat{j}_{f3} + \hat{i}_{f3}\hat{j}_{f2}) = 0\end{aligned} \qquad (2.6)$$

where

$$\begin{aligned}\hat{i}_f &= \begin{bmatrix}\hat{i}_{f1} & \hat{i}_{f2} & \hat{i}_{f3}\end{bmatrix} \\ \hat{j}_f &= \begin{bmatrix}\hat{j}_{f1} & \hat{j}_{f2} & \hat{j}_{f3}\end{bmatrix}\end{aligned} \quad \text{for} \quad f = 1...F.$$

Notice that (2.6) is a linear over-constrained system with 3F equations. Hence, the least-squared error solution of Φ can be effectively obtained by calculating the pseudo-inverse of the coefficient matrix with respect to this linear system.

Once Φ is obtained, we have to decompose it into the form QQ^T. In principle, there are infinite many solutions of Q satisfying that $\Phi = QQ^T$ because the number of free variables of Q is nine but that of Φ is only six. It can also be clarified with the fact that if we replace Q by $Q' = QR_o$ in (2.3) with an arbitrary rotation matrix R_o, it still satisfies that $RS = \hat{R}\ \hat{S}$ because $R_o^T R_o$=I. Hence, we can first find an arbitrary Q satisfying $\Phi = QQ^T$, and it is equivalent to ignoring the constraint (2.1) in the beginning because different solutions of Q only induce different selections of the three valid axes of the world coordinate system as introduced in Section 2.1. To achieve this purpose, we use the eigenvalue decomposition of Φ:

$$\Phi = QQ^T = PDP^T = (PD^{1/2})(D^{1/2}P^T) \qquad (2.7)$$

where D is diagonal matrix consists of the eigenvalues of Φ, and P the matrix consists of the eigenvectors of Φ.

Another problem of the method described in Section 2.1 is that the solution obtained may not *exactly* satisfy the metric constraints shown in (2.3) if we treat the constrains to be solved as a data-fitting problem and solve it in an over-constrained manner. To get the solution exactly satisfying the metric constraints in (2.3), we should formulate a constrained optimization problem instead of a data-fitting problem. In our work, the solution obtained via the linear method described above is used as an initial estimate of the following constrained optimization problem with the error function as shown in (2.8).

$$\underset{R,S}{MIN} \ \left\|\tilde{W} - RS\right\|^2 \qquad (2.8)$$

subject to the constraints in (2.1) and (2.2).

In our work, the MATCOM library is used for solving this constrained nonlinear optimization problem. To obtain a good initial estimation, R and S computed through the above linear procedure are transformed by R ¬RR_0 and S ¬ $R_0^T S$, where R_0 is the least-squared-error rotation matrix which transforms i_1, j_1 and $i_1 \times j_1$ to $[1\ 0\ 0]^T$, $[0\ 1\ 0]^T$ and $[0\ 0\ 1]^T$, respectively [1].

2.3. Handling Partial Occlusion

In practice, feature points may appear and disappear in an image sequence due to occlusions. When occlusions occur, the factorization method introduced above cannot be applied directly because the measurement matrix $\tilde{W}$ contains empty entries. The empty entries are recoverable if either the same feature is seen often enough in other frames or the frame in question includes enough other features. A characteristic of the empty entries occurred in the measurement matrix $\tilde{W}$ is that they are pair-wise. That is, if an empty entry occurred in the U-field, then it should also occurred in the corresponding position of the V-field, as shown in Figure 1(a) and 1(b).

[1] The Arun et. al method [2] is used for computing the least-squared-error rotation matrix R_0.

Therefore, without lost of generality, our discussion on recovering the empty entries is based on the U-field in the following.

In fact, through a detailed analysis we can find that the method introduced in Sections 2.1 and 2.2 requires at least three views of four points or four views of three points to determine the structure and motion under orthographic projection. Hence, if $U' \subseteq \tilde{U}$ is a 4×4 sub-matrix containing one empty entry, then it can be recovered by the following procedure [14]: (1) computing the 3D coordinate of the occluded feature with the other three visible frames, (2) computing the camera pose of the frame containing the occluded feature with the other three visible points, and (3) projecting the 3D coordinate onto the image plane of the frame containing the occluded feature. In fact, the above procedure can also be extended to recover the empty entry if $U' \subseteq \tilde{U}$ is a *K ´K* (*K ³4*) sub-matrix containing one empty entry [14].

Definition 1: [Valid Sub-matrix] A sub-matrix is *valid* for recovering an empty entry u_{ij} if its dimension is larger than or equal to *4 ´4* and it only contains one single empty entry, i.e., u_{ij}.

The empty entries in a *2F ´P* measurement matrix $\tilde{W}$ can be recovered in turn. However, the selection of the sub-matrix can affect the final solution. For example, when we want to fill u_{44} in the *U*-field as shown in the left side of Figure 1(b), both the sub-matrices shown in Figure 1(c) are valid. Hence, to select a better valid sub-matrix is an important issue. Consider an empty entry in *U* to be recovered, we formulate the procedure of finding a valid sub-matrix as a *row/column deletion* process as described below.

Property 1: [Definite Column Deletion] Assume that the empty entry to be recovered is u_{ij}. Consider all of the other empty entries in the *i*-th row of *U*, namely $u_{i,e(1)}$, $u_{i,e(2)}$, ..., $u_{i,e(n)}$. Then, the columns *e(1), e(2), .., e(n)* can not be used to recover the empty entry u_{ij}.

pf: The entry u_{ij} should be the only empty one contained in a valid sub-matrix which can be used to recover u_{ij}. If the sub-matrix contains any rows, *e(k), k=1, 2, ..n*, then it also contains another empty entry $u_{i,e(k)}$, which causes a contradiction.

Hence, we can delete the columns *e(1), e(2), .., e(n)* in the process of finding a valid sub-matrix. Similarly, the following property also holds.

Property 2: [Definite Row Deletion] Assume that the empty entry to be recovered is u_{ij}. Consider all of the other empty entries in the *j*-th column of *U*, namely $u_{e(1),j}$, $u_{e(2),j}$, ..., $u_{e(n),j}$. Then, the rows *e(1), e(2), .., e(n)* can not be used to recover the empty entry u_{ij}, and thus can also be deleted in the sub-matrix selection process.

For example, if we want to recover the empty entry, u_{33}, of the matrix *U* shown in Figure 2(a), then the seventh row and the fifth column can be deleted at first according to Property 1 and Property 2 because they are not possible to be contained in any valid sub-matrices, as shown in Figure 2(b). In particular, we also hope that the sub-matrix selected is *maximal* as defined bellow:

Definition 2: [Maximal Sub-matrix] A sub-matrix is *maximal* if it is valid and it is not the proper sub-matrix of any other valid sub-matrices.

Basically, a maximal sub-matrix can exploit as more as feature points or image frames for estimation of the values of the empty entries, and thus can estimate them more accurately.

If all of the empty entries contained in *U* have been removed by considering Property 1 and Property 2, then the remaining sub-matrix is a maximal one and can be used to recover u_{ij}. On the other hand, consider an empty entry, $u_{i1,j1}$, which remains after the definite row and column deletion processes described in Property 1 and Property 2. Then, either the *i1*-throw or the *j1*-th column can be deleted. For example, consider one of the remaining empty entries, u_{57}, in Figure 2(c). Either the fifth row or the seventh column can be deleted. If the fifth row is turn out to be deleted, then there are also two choices of either deleting the ninth row or deleting the sixth column for another remaining empty entry, u_{96}, and so on. The above process is iterated until that there is no empty entry remained. Finally, one of the possible maximal sub-matrix can be obtained as shown in Figure 2(f).

In the above process, the deletion of the row and the column with respect to a remaining empty entry can be implemented randomly. In this paper, to find a better sub-matrix for recovering the empty entries, we proposed a method called *randomized median* via the principle of robust estimation. In each time of selection of rows or columns to be deleted, we apply a non-deterministic process for the determination of the deletion of columns or rows. Once a K×K (K≥4) sub-matrix containing only one empty entry is selected, we recover the empty entry and then apply the least square technique introduced in Section 2.2 to solve its over-constrained solution. The above two steps of selecting a sub-matrix and estimating the value of the empty entry are iterated for several times and the result is recorded in each round. Finally, the *median* of the values recorded for each empty entry is taken to be the estimated value of it.

Although the selection of a sub-matrix in each iteration is a non-deterministic process in our approach, we do not select them pure randomly like the standard LMedS method does. On the other hand, we incorporate some priori knowledge for the selection of a possibly better sub-matrix. First, we only consider the sub-matrix with enough feature points and images frames by forbidding the deletions allowing the remaining sub-matrix to be smaller than 4×4. Second, we prefer to select a sub-matrix as square as possible because a thin sub-matrix may either contain too few features or too near frames which may cause the computation to be un-stable. Hence, when the remaining rows are more than the remaining columns during the process of selection of rows or columns to be deleted, we give a higher probability to delete rows, and vice versa.

2.4. Estimation of Camera Poses with Human Face Images

Typically, five images of a human head are taken for face model reconstruction in our work. Occlusions may occur in the images taken from different view angles. To estimate the camera poses of the images, a set of feature correspondences have to be given. In the present version of our implementation, the correspondences of feature points are provided by human in a semi-automatic way. In principle, the more accurate are the feature correspondences, the more accurate are the pose parameters estimated. Hence, in the camera-pose stage, we hope the users to select some salient and explicit feature points in the human face, such as eyes, nose, lips, and ears. Typically, nine features in a face image are selected in our work as shown in Figure 3(b). Then, the correspondences of the visible feature points in other images are also selected as shown in Figure 3(a) and 3(c).

Figure 4 shows the results of recovering the occluded feature points by using the *randomized median* procedure as introduced in Section 2.3. In Figure 4, the square dots are the feature points selected by the user and the diamond dots are the recovered feature points which are occluded in the input images. One can see that the missing (or occluded) points can be well complemented using our method. When all occluding points are recovered, the 3D coordinates of all feature points can be computed. To verify the results, we show the corresponding epi-polar geometry. Given two images whose poses are given under the assumption that the camera model is orthographic, a plane is called the epi-polar plane if it is perpendicular to both images. The epi-polar line pairs can then be obtained by finding the intersection lines between each epi-polar plane and both the image planes. We call the intersection line between an epi-polar plane and the left image plane the *left epi-polar line*, and that of the same epi-polar plane and the right image plane the *corresponding right epi-polar line*, respectively. An important characteristic is that the correspondences of the image points contained in the left epi-polar line should be lie on its corresponding right epi-polar line. The more accurate correspondences of the pairs of epi-polar lines, the more accurate is the camera pose estimated. Figure 5 shows an example of the pairs of epi-polar lines computed with the camera poses estimated using our approach. One can see that the corresponding epi-polar lines are well matched. In particular, the epi-polar line is also served as a good hint to the users in our work for selecting more correspondences of features as described in Section 3.

3. Face Model Reconstruction

In our work, the face model of a particular person is obtained by deforming a generic face model as shown in Figure 6. Assume that a set of 3D points on the face of a particular person is reconstructed. The generic model can then be deformed in such a way that it fits these 3D points as well as possible. In the previous section, the camera poses have been estimated under the assumption that the camera model is orthographic and nine salient and explicit feature points have also been computed. However, it is usually not sufficient enough to reconstruction a face model using only nine feature points through the deformation of a generic model. Although we can allow the users to select more points in the camera-pose estimation stage, it may not give too much help because the estimation of the camera poses requires to using salient and explicit feature points for accurate estimation. However, such feature points are very likely to be within the neighborhood regions of the nine points shown in Figure 3(b), and hence carries out not too much more information for camera pose estimation. In fact, the 3D positions of the salient and explicit regions in a human have been estimated in a considerably accurate manner in the camera-pose estimation stage, the remaining critical regions to be reconstructed are mainly the intensity-smooth part in a human face. In the phase 1 of our work, more control points (many of them are in the critical regions) are chosen and reconstructed using multi-view stereo. In the phase 2 or our work, the generic model is then deformed to fit the 3D positions of those control points.

- **Phase 1: Multi-view Stereo for Reconstruction of Coarse Model**

Because the camera poses have been estimated, other 3D points can be obtained via simple triangulation (or referred to as *multi-view stereo*) as shown in Figure 7 if the correspondences of the points among images are given. In principle, at least two images are required in this phase, and typically 3-5 images are used in our work. To reconstruct more 3D points on the face model, the users are required to give the correspondences of points for other 35 points as shown in Figure 9(a), and hence totally 44 points on a human face will be reconstructed in our work. As stated before, most of the other 35 points lie in the intensity-smooth region. In principle, to select their corresponding positions in images is a more difficult problem to human perception. To help the user's selection via his (or her) perception, the epi-polar line pairs introduced in Section 2.4 are used to give good hints as shown in Figure 9. The 44 reconstructed points can be connected to form a coarse model, as shown in Figure 10.

- **Phase 2: Deformation Process**

The deformation process used in this paper is the method of control triangulation proposed by Fua [5]. Figure 9 shows the deformation results by deforming the generic model shown in Figure 6 to fit with the control points (i.e., the vertices of the coarse model) shown in Figure 10.

4 Experimental Results

Figures 12-17 show some human face model reconstructed using the approach introduced in this paper. The upper rows shows the five images of a particular person, and his face model reconstructed, respectively. The lower rows show some texture-mapped human face models observed from different viewing directions. Our main purpose is to reconstruct the face region of the human. The hair and the neck regions are reconstructed roughly and shown simply for better visual effects. It can be observed that our approach can reconstruct the human face model accurately to a considerable extent.

5 Summary

In this paper, we developed a systematic method for reconstruction of the human face model from multiple images in a semi-automatic way. First, the camera poses with respect to the images are computed under the assumption that the camera model is orthographic. Then, more 3D control points are reconstructed using multi-view stereo. Finally, a mesh model of the human face is obtained by deforming an initial generic model to fitting the control points. The contribution of this paper includes (1) we propose a linear method to give a good initial estimation for nonlinear optimization, so as to allow the computation process of camera pose estimation to be more stable. (2) We propose a new hole-filling process for handling the partial occlusion problem so as to recover the missing feature points in a more robust way. In the current implementation of our approach, feature points and correspondences are selected manually. The computation process can be more automatic by tracking the features in the images. Some future work includes the combination of the multi-view stereo and the deformation phases to be a single energy minimization process and to use the contour information in images for refinement of the 3D model.

References

[1] T. Akimoto, Y. Suenaga, R. S. Wallace, "Automatic Creation of 3D Facial Models", *IEEE Computer Graphics and Applications*, 1993.

[2] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-square Fitting of Two 3-D Point Set," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 9, pp. 698-700, 1987.

[3] Q. Chen, G. Medioni, "A Volumetric Stereo Matching Method: Application to Image-Based Modeling", *Proceedings of International Conference on Computer Vision and Pattern Recognition, CVPR' 99*, 1999.

[4] H. Delingette, J. Montagnat, "General Deformable Model Approach for Model-based Reconstruction", *Proceedings of IEEE International Workshop on Model-Based 3D Image Analysis*, pp. 55-64, 1998.

[5] P. Fua, "From Multiple Stereo Views to Multiple 3D Surfaces", *International Journal of Computer Vision*, Vol. 24, No.1, pp. 19-35, 1997.

[6] P. Fua, "Face Models from Uncalibrated Video Sequences", *Workshop on Modeling and Motion Capture Techniques for Virtual Environments*, Geneva, Switzerland, pp. 215-228, Nov. 1998.

[7] B. Guenter, C. Grimm, D. Wood, "Making Faces", *Proceedings of SIGGRAPH' 98*, pp.55-66, 1998.

[8] S. Y. Ho, H. L. Huang, and C. F. Lee, "An Optimization Algorithm for the Reconstruction of a 3-D Facial Model from a Monocular Facial Image," *Proceedings of IPPR Conference on Computer Vision, Graphics, and Image Processing*, Taipei, pp. 144-151, 1998.

[9] W. S. Lee, Nadia Magnenat-Thalmann, "Generating a Population of Animated Faces from Pictures", *Proceedings of IEEE International Workshop on Modeling People*, Corfu, Greece, pp. 62-69, 1999.

[10] R. Lengagne, J. Tarel, O. Monga, "From 2D Images to 3D Face Geometry", *Proceedings of the 2nd International Conference on Automated Face and Gesture Recognition*, 1996.

[11] R. Lengane, P. Fua, O. Monga, "3D Face Modeling from Stereo and Differential Constraints", *Proceedings of IEEE and ATR Workshop on Computer Vision for Virtual Reality-based Human Communications*, Bombay, India, 1998.

[12] A. Maki, M. Watanabe, C. Wiles, "Geotensity Combining Motion and Lighting for 3D Surface Reconstruction", *Proceedings of International Conference on Computer Vision*, Bombay, India, 1998.

[13] R. Sara, R. Baijcsy, "Fish-Scale Representation Fuzzy Manifolds", *Proceedings of International Conference on Computer Vision*, Bombay, India, 1998.

[14] C. Tomasi and T. Kanade, "Shape and Motion from Images Streams under Orthography: A Factorization Method", *International Journal of Computer Vision*, Vol. 9, pp. 137-154, 1992.

$$\tilde{W} = \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} & \times & \times & \times \\ u_{21} & u_{22} & u_{23} & u_{24} & \times & \times & \times \\ u_{31} & u_{32} & u_{33} & u_{34} & \times & \times & \times \\ u_{41} & u_{42} & u_{43} & \times & u_{45} & u_{46} & u_{47} \\ \times & \times & \times & u_{54} & u_{55} & u_{56} & u_{57} \\ \times & \times & \times & u_{64} & u_{65} & u_{66} & u_{67} \\ \times & \times & \times & u_{74} & u_{75} & u_{76} & u_{77} \\ v_{11} & v_{12} & v_{13} & v_{14} & \times & \times & \times \\ v_{21} & v_{22} & v_{23} & v_{24} & \times & \times & \times \\ v_{31} & v_{32} & v_{33} & v_{34} & \times & \times & \times \\ v_{41} & v_{42} & v_{43} & \times & v_{45} & v_{46} & v_{47} \\ \times & \times & \times & v_{54} & v_{55} & v_{56} & v_{57} \\ \times & \times & \times & v_{64} & v_{65} & v_{66} & v_{67} \\ \times & \times & \times & v_{74} & v_{75} & v_{76} & v_{77} \end{bmatrix}$$

(a)

$$\tilde{U} = \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} & \times & \times & \times \\ u_{21} & u_{22} & u_{23} & u_{24} & \times & \times & \times \\ u_{31} & u_{32} & u_{33} & u_{34} & \times & \times & \times \\ u_{41} & u_{42} & u_{43} & \times & u_{45} & u_{46} & u_{47} \\ \times & \times & \times & u_{54} & u_{55} & u_{56} & u_{57} \\ \times & \times & \times & u_{64} & u_{65} & u_{66} & u_{67} \\ \times & \times & \times & u_{74} & u_{75} & u_{76} & u_{77} \end{bmatrix} \quad \tilde{V} = \begin{bmatrix} v_{11} & v_{12} & v_{13} & v_{14} & \times & \times & \times \\ v_{21} & v_{22} & v_{23} & v_{24} & \times & \times & \times \\ v_{31} & v_{32} & v_{33} & v_{34} & \times & \times & \times \\ v_{41} & v_{42} & v_{43} & \times & v_{45} & v_{46} & v_{47} \\ \times & \times & \times & v_{54} & v_{55} & v_{56} & v_{57} \\ \times & \times & \times & v_{64} & v_{65} & v_{66} & v_{67} \\ \times & \times & \times & v_{74} & v_{75} & v_{76} & v_{77} \end{bmatrix}$$

(b)

$$U_1 = \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ u_{21} & u_{22} & u_{23} & u_{24} \\ u_{31} & u_{32} & u_{33} & u_{34} \\ u_{41} & u_{42} & u_{43} & \times \end{bmatrix} \quad U_2 = \begin{bmatrix} u_{55} & u_{56} & u_{57} & u_{54} \\ u_{65} & u_{66} & u_{67} & u_{64} \\ u_{75} & u_{76} & u_{77} & u_{74} \\ u_{45} & u_{46} & u_{47} & \times \end{bmatrix}$$

(c)

Figure 1: (a) An example of the measurement matrix containing empty entries. (b) Its U-field and V-field. The empty entries are the same in the U-field as that in the V-field. (c) Possible sub-matrices which can be used to recover the empty entries.

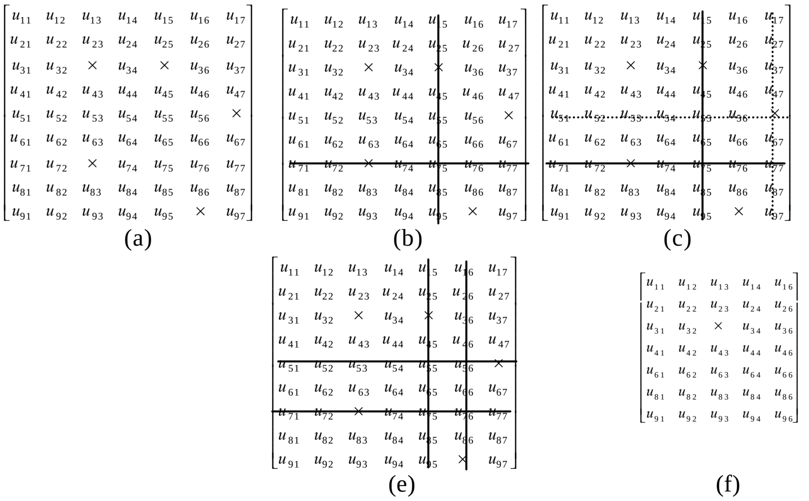


Figure 2: An example of the row/column deletion process. (a) A measurement matrix with some empty entries and assume that the empty entry *u₃₃* is the target to be recovered. (b) The empty entries which are definitely deleted according to Property 1 and Property 2. (c) The dashed lines show the two possible deletions considering the remaining empty entry *u₅₇*. (d) If the fifth row is turn out to be deleted, then there are also two choices of deletions for another remaining empty entry *u₉₆*. (e) If the sixth column is turn out to be deleted. (f) The remaining maximal sub-matrix which can be used to recover *u₃₃*.

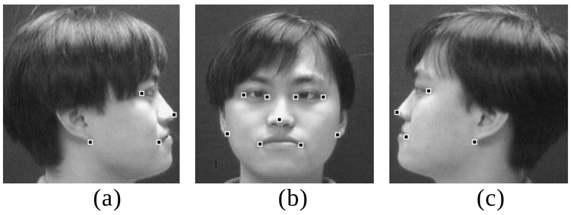


Figure 3. To estimate the camera poses, nine feature points are selected in the middle view as show in (b). The corresponding points of the nine features are then selected by the user in other views as shown in (a) and (c). Due to occlusion, some features in the middle may not have corresponding points in other views.

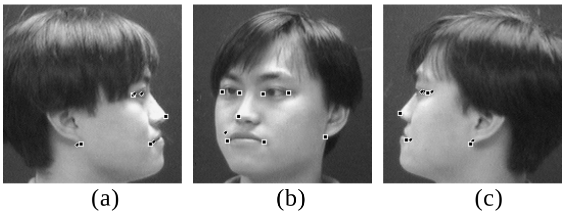


Figure 4. Input points and the recovered occluding points. Square dots are the input points, and the diamond dots are the recovered occluding points.

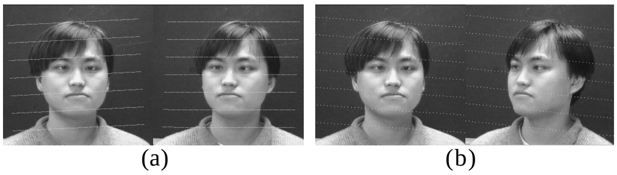


Figure 5 (left). Pairs of epi-polar lines in two image pairs. The corresponding point of a point in the left (or right) image have to be lie on the corresponding epi-polar line in the right (or left) image.

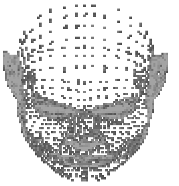


Figure 6. The generic face model used in our work.

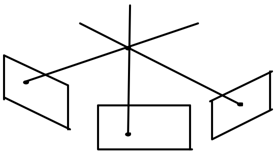


Figure 7. Stereo reconstruction with multi-views.

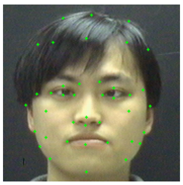
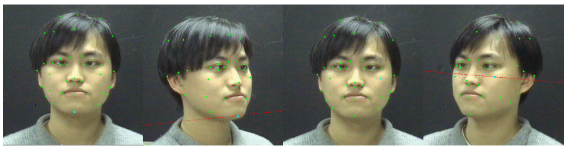
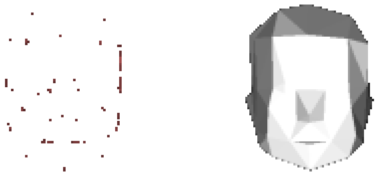


Figure 8. Pick other 35 points in the front view.



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

Figure 9. Computer-aided selection for picking the correspondences of the other 35 control points. (a) Pick the jaw in the front view. (b) The corresponding epi-polar line is served as a good hint for selection. (c) Pick the cheekbone in the front view. (d) Its correspondence epipolar line is served as a good hint for selection.



(a) (b)

Figure 10. Reconstruction of the 44 control points and connecting them to form a coarse model. (a) Wire-frame result of the generated coarse model. (b) Shading result of the generated coarse model.

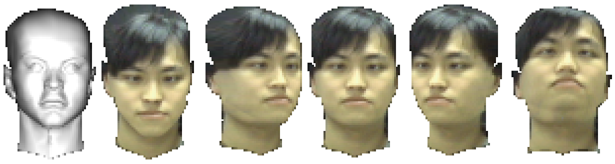


Figure 11. The left-upper shows the reconstructed face model of Mr. Wu rendered with a standard shading technique. The other images are some texture-mapped results of the reconstructed model observed from different viewing directions.

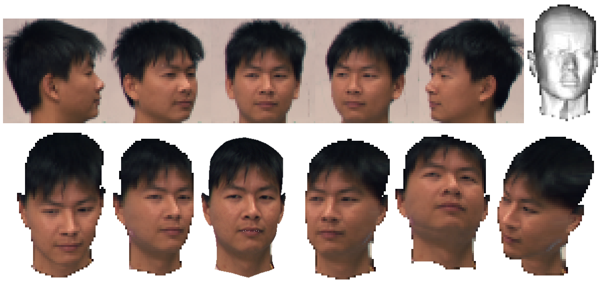


Figure 12. The upper row shows the five images of Mr. Tsai and his reconstructed face model. Some texture-mapped results of the reconstructed model observed from different viewing directions are shown in the lower row.

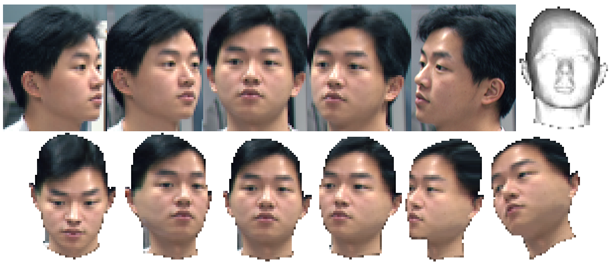


Figure 13. The upper row shows the five images of Mr. Kao and his reconstructed face model. Some texture-mapped results of the reconstructed model observed from different viewing directions are shown in the lower row.

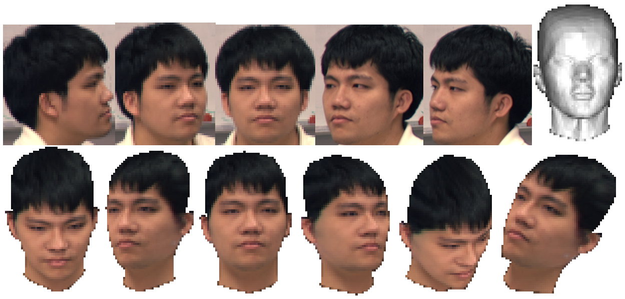


Figure 14. The upper row shows the five images of Mr. Chen and the reconstructed face model of him. Some texture-mapped results of the reconstructed model observed from different viewing directions are shown in the lower row.

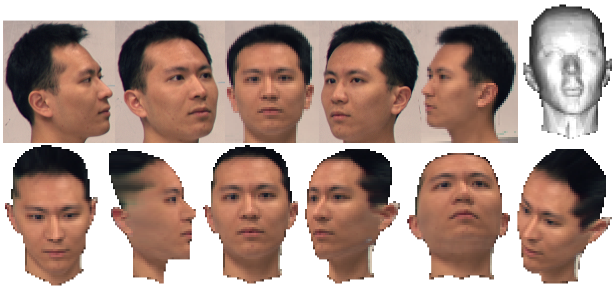


Figure 15. The upper row shows the five images of Mr. Chen and the reconstructed face model of him. Some texture-mapped results of the reconstructed model observed from different viewing directions are shown in the lower row.