Accurate 3D Face Reconstruction from Weakly Calibrated Wide Baseline Images with Profile Contours

Yuping Lin, Gérard Medioni, and Jongmoo Choi
Computer Science Department, University of Southern California
3737 Watt Way, PHE 101, Los Angeles, CA, 90089
{yupingli, medioni, jongmooc}@usc.edu

Abstract

We propose a method to generate a highly accurate 3D face model from a set of wide-baseline images in a weakly calibrated setup. Our approach is purely data driven, and produces faithful 3D models without any pre-defined models, unlike other statistical model-based approaches. Our results do not rely upon a critical initialization step nor parameters for optimization steps. We process 5 images (including profile views), infer the accurate poses of cameras in all views, and then infer a dense 3D face model.

The quality of 3D face models depends on the accuracy of estimated head-camera motion. First, we propose to use an iterative bundle adjustment approach to remove outliers in corresponding points. Contours in the profile views are matched to provide reliable correspondences that link two opposite side of views together. For dense reconstruction, we propose to use a face-specific cylindrical representation which allows us to solve a global optimization problem for N-view dense aggregation. Profile contours are used once again to provide constraints in the optimization step. Experimental results using synthetic and real images show that our method provides accurate and stable reconstruction results on wide-baseline images. We compare our method with state of the art methods, and show that it provides significantly better results in terms of both accuracy and efficiency.

1. Introduction

We address a very difficult problem, the inference of an accurate 3D face model from a set of images separated by a wide-baseline in a weakly calibrated setup. Unlike other statistical model-based methods, such as Vetter's [8], our approach is purely data-driven, thus produces faithful models. This gives two important advantages. First, our method provides stable reconstruction results while other methods require a critical initialization step. Second, our method generates a more accurate shape of a person. The use of a generic face model allows the reduction of the computational complexity or the inference of uncertain face shape, but restricts the use of the obtained 3D surface for some critical applications such as face biometrics and medical surgery. Our method should handle asymmetry of face, and local geometry (*e.g.* scar), which are important factors for realistic 3D face modeling.

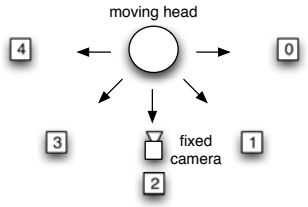

Figure 1. The setup for capturing the five facial images of a subject. The five boxes are the targets to face at when taking pictures. The circle represents the head of the subject.

We process 5 images acquired from a fixed camera and a turning head facing 5 different targets which are roughly 45° apart, as shown in Figure 1. The head motion is mostly rotational, but may include translation. This results in 5 images that include a left and a right profile view, as shown in Figure 2. The main contribution of this paper is the inference of an accurate 3D face model from this weakly calibrated setup. We start by estimating camera poses in all views, building a sparse reconstruction, then inferring a dense 3D face model. We propose a set of key algorithmic components in order to make high quality 3D face models.

The quality of 3D face models depends first on the accuracy of estimated head-camera motion. It is very difficult to establish correspondences between views in wide-baseline stereo images due to perspective distortions and lighting changes. We propose a new method to estimate the relative head-camera motion. First, we use an iterative bundle adjustment method to handle outliers in corresponding points. Even one single outlier might yield catastrophic errors in the motion estimation result. Second, silhouette matching links

978-1-4244-6985-7/10/$26.00 ©2010 IEEE

J_0 I_0 I_1 I_2 I_3 I_4 J_4

Figure 2. The five facial images and two profile images. Image I_i is the image when the subject faces target i in Figure 1.

two profile views that have no point-wise correspondences, and provides additional constraints for calibration.

For dense reconstruction, we employ a face-specific cylindrical representation, which allows us to solve a global optimization problem for N-view dense aggregation. In [12], individual pairwise reconstructions are integrated into a single 3D point cloud, and then an additional post-processing (e.g. tensor voting) is applied to enforce surface self-consistency. We aggregate all information to get a single 3D point cloud using a well-known optimization technique. With this representation, dynamic programming can be applied directly for all views. Profile contours are then used once again to produce accurate shape along the center (vertical) face region.

Experimental results using synthetic and real images show that our method provides accurate and stable reconstructions results with wide-baseline images in which conventional data driven techniques, such as Boujou [3], Arc3D [2], and Photosynth [6], fail to reconstruct correct camera poses in the first place. Even by providing accurate calibration from our setup, the state of the art PMVS [14] software package failed to build the model.

The remainder of the paper is organized as follows. Section 2 summarizes the related work. Our approach is presented in Section 3, and Section 4 shows the experimental results and Section 5 concludes the paper.

2. Literature Review

Most proposed approaches to solve this problem start from a generic model, which is then deformed to fit the data. In [8], a deformable 3-D face model was introduced, which is fit to an image using a nonlinear optimization technique and a suitable distance measure. In [7], the authors create a 3D head model from only a frontal view, and a profile view by adjusting a generic model. Tang [25] creates 3D face model by the location of 32 feature points on the face in multiple images, and relates them to the nodes in a generic face model. This method builds a reasonable 3D model, but is not dense enough to represent the face shape accurately. Fua [13] uses a model-driven, bundle adjustment method to get the head model from a raw video sequence. It takes advantage of stereo data, silhouette edges and 2D feature points. It has shown good results, but it is not clear whether the silhouette edges and 2D feature points help. Combining the three into a uniform representation is not a trivial problem. In [22], the authors propose a model-based bundle adjustment algorithm that takes a set of image tracks as input, without a prior 2D to 3D association. The use of a generic face model allows the reduction of the complexity of the bundle adjustment algorithm, but restricts the use of the obtained 3D surface for identification purposes. The 3D features of the face are mapped onto the generic model, and consequently ignore the very specific features that allow the identification of a person from his/her 3D characteristics.

On the other hand, the modeling problem can fit in multi-view stereo (MVS) framework that has no assumption on the reconstructed shape. Both internal and external camera parameters, however, must be provided to locate the point projections in images. A thorough survey of early MVS works has been presented in [21]. Several methods have achieved sub-millimeter accuracy on the benchmark objects. In recent developments, Furukawa [15] presents a patch-based approach that ranks top in high resolution imagery benchmarking [24]. Starting from a sparse set of matched keypoints, their corresponding surface(patch) normals are estimated, which are then used in iterative expansion and filtering phases that gradually completes the entire scene. Despite the outstanding performance in terms of accuracy, experimental results show it fails to perform dense reconstruction on our image sets. In addition, the processing time for high resolution images takes hours, which is too computational demanding for some applications.

Note that most MVS approaches do not scale well with images that have wide baselines. Fua [26] presents an algorithm that can efficiently produce SIFT like local descriptors, DAISY, for dense matching purposes. The sophistication of its descriptors allows wide-baseline matching to be performed on low resolution images. Strecha [23] addresses the self-occlusion problem in wide baseline images. He presents a probabilistic approach in which the depth with respect to each image is optimized using EM algorithms.

We reiterate that the aforementioned methods all assume known external and internal camera calibration, which is difficult in practice. Although structure from motion (SfM) techniques [16, 20] can perform camera calibration directly from images, the accuracy relies on the quality of correspondences, which may be poor in wide baseline images.

We present our work that starts from a set of 5 weakly

calibrated wide baseline images that still produces qualitatively competitive results, as described in the next section.

3. Approach

There are two main modules in our approach, sparse reconstruction and dense reconstruction. Sparse reconstruction aims at recovering the unknown camera poses, which is a challenging problem given wide baseline images. For dense reconstruction, the goal is to reconstruct the dense 3D surface of the face, where we use contour lines in the profile images as a strong constraint.

3.1. Sparse Reconstruction

To define a set of cameras in the same space, we need to establish point correspondences in at least 3 views. Finding such correspondences in our imagery is a very challenging task since any 2 adjacent views are 45° apart. Lighting changes and perspective distortions make the matching process error prone. In classical SfM method with known camera intrinsic parameters, 3 view correspondences inliers can be naturally extracted by employing RANSAC on top of PnP [18] algorithms. However, such approach fails most of the time due to both the scarcity of absolute inliers and the small ratio of inliers to outliers (we typically get less than 10 inliers out of a total of 30 in the frontal 3 views, as shown in the experiment section). We thus present a new algorithm to extract 3 view correspondence inliers.

Our approach is illustrated in Figure 3. We choose to use state of the art SURF features [9] as interest points, and match them between image pairs using nearest neighbor matching with a certain threshold of nearest neighbor distance ratio [19]. The next step is to estimate epipolar geometry from the matches. Ideally, given camera intrinsic matrices, we are able to estimate the essential matrix, from which the relative camera poses can be derived. However, in practice, we find that the focal length information recorded is not accurate enough, and the estimated essential matrix does not yield reasonable camera poses. We thus instead estimate the fundamental matrix using a plane+parallax method [10]. Note that with the estimated $\mathbf{F}$ matrix, we perform another pass of guided feature matching to improve the quality of matches, from which a more accurate epipolar geometry can be estimated, and a more accurate set of $\mathbf{F}$ matrix inliers can be produced. We denote the $\mathbf{F}$ matrix inliers between image I_m and I_n by $\pi_{m,n}$.

We then use a bundle adjustment based approach to estimate the 5 camera poses. Here we use $\mathfrak{B}$ to denote the bundle adjustment process, and use subscripts $_R$, $_T$, and $_S$ to represent three types of free parameters, which are camera rotation, camera translation, and structure respectively (*e.g.*, $\mathfrak{B}_{R|S}$ represents the camera rotational component and the structure are free parameters, while the camera positions are fixed). To initialize the camera poses, we exploit the knowledge that any two adjacent cameras are roughly 45° apart, and have roughly the same distance to the subject. Along with the estimated $\mathbf{F}$ matrices, we are able to initialize the cameras at the locations that are very close to their true ones. For the correspondences, we extract sets of 3-view correspondences in 3 groups of images (I_0, I_1, I_2), (I_1, I_2, I_3), and (I_2, I_3, I_4) respectively. Let p_i^j denotes the j^{th} feature in I_i, a 3-view correspondence is a triple $(p_{i-1}^a, p_i^b, p_{i+1}^c)$ where $(p_{i-1}^a, p_i^b) \in \pi_{i-1,i}$ and $(p_i^b, p_{i+1}^c) \in \pi_{i,i+1}$. We use $\Omega_i, i = [1, 2, 3]$ to denote the set of 3-view correspondences and the 3D points they triangulate to interchangeably. To relate I_0 and I_4, contour lines in J_0 and J_4 are matched using iterative closest point (ICP) algorithm, and a number of correspondences, denoted as $\bar{\Omega}$, are extracted. We then use two passes of bundle adjustment, $\mathfrak{B}_{R|S}$ followed by $\mathfrak{B}_{R|T|S}$, on the correspondences ($\Omega_1, \Omega_2, \Omega_3$ and $\bar{\Omega}$) to produce the final camera calibration result. The objective in $\mathfrak{B}_{R|S}$ is to produce a suboptimal set of 3D points that lead to a faster convergence in $\mathfrak{B}_{R|T|S}$. We also fit a vertical cylinder to the reconstructed sparse structure, and estimate its center axis and radius, which are used in dense reconstruction as later discussed in Section 3.2.

input : $\mathbf{F}$ matrix inliers $\pi_{i-1,i}$ and $\pi_{i,i+1}$. Initial camera poses P_{i-1}, P_i, P_{i+1}
output: 3-view correspondence inliers Ω_i

```
Construct initial Ω_i from π_{i-1,i} and π_{i,i+1};
while true do
    B_{R|S}(Ω_i, P_{i-1}, P_i, P_{i+1})
    Reproject points in Ω_i and compute individual reprojection error;
    if max repojection error > T then
        remove 5% of points in Ω_i with the largest reprojection error;
    else
        break;
    end
end
return Ω_i;
```

Algorithm 1: Iterative bundle adjustment

As mentioned earlier, the challenge here is to obtain accurate sets of Ω_i. We propose to use an iterative bundle adjustment approach as shown in Alg.1 that produces Ω_1, Ω_2, and Ω_3 individually. The idea is to iteratively remove a small portion of points in Ω_i that have the largest reprojection error, and then perform bundle adjustment again on the remaining points to recalibrate the cameras. We use $\pi_{i-1,i}$ and $\pi_{i,i+1}$ to construct an initial set of Ω_i. Since outliers in Ω_i greatly influence the optimization result, we fix the camera positions and adjust their rotation components

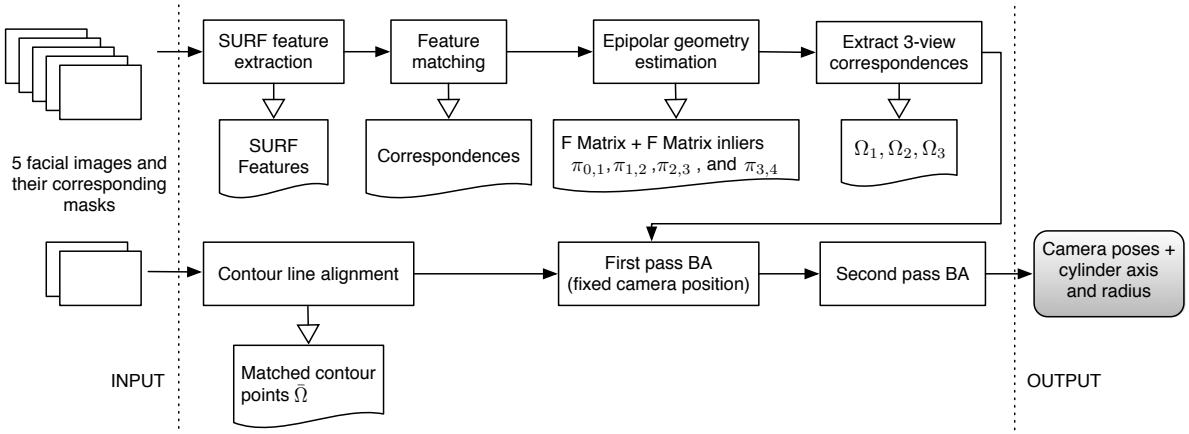

Figure 3. Flow chart of the sparse reconstruction module.

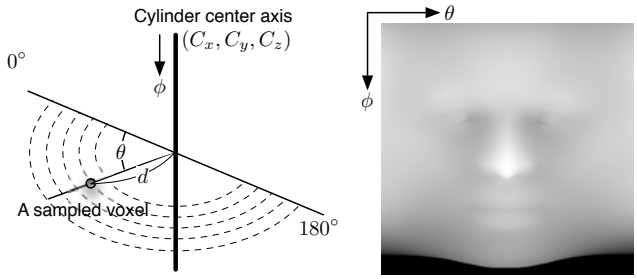

Figure 4. Cylindrical voxel sampling for dense 3D reconstruction.

only, *i.e.*, use $\mathfrak{B}_{R|S}$. We also include $\pi_{i-1,i}$ and $\pi_{i,i+1}$ in the bundle adjustment so that the optimization will bias the points that conform to the estimated epipolar geometry. We then use the adjusted 3D points and cameras to compute the reprojection error for each point in Ω_i. If the largest reprojection error is over a threshold, we remove 5% in Ω_i that have the largest reprojection error, and perform another iteration again. Compared with RANSAC based approaches, ours can still work when the number of inliers is small. Note that the correspondences in $\bar{\Omega}$ extracted by aligning the two profile contours are only accurate when the subject's facing direction is *exactly orthogonal* to the camera. As the face pans away, the correspondences are less accurate, which in turn degrades the accuracy of the camera poses. We give a full quantitative analysis of this matter in our experiments.

3.2. Dense Reconstruction

Once the cameras are calibrated, we can perform dense 3D reconstruction of the face from the five images, which is a multi-view stereo problem. It has been studied for years and a number of approaches have been developed. Unlike reconstruction of a scattered scene, reconstruction of faces is relatively easy due to the smoothness of the surface. However, the demand for accuracy is also higher since the reconstructed shape is critical for applications.

We use a voxel-based approach. However, instead of sampling voxels in a Cartesian coordinate system, we sample voxels in a cylindrical coordinate system. The advantage of using a cylindrical coordinate system is two fold. First, it yields a compact representation of a 3D surface. The geometry of a facial surface can be represented using an image $\mathcal{D}$, where the value at $\mathcal{D}(\theta, \phi)$ is the horizontal distance to the cylinder center axis. The 3D point correspond to pixel $\mathcal{D}(\theta, \phi) = d$ is therefore

$$X_{\theta,\phi,d} = \big(d\cos(\theta) + C_x, k\phi + C_y, d\sin(\theta) + C_z\big), \quad (1)$$

as shown in Figure 4 (left). (C_x, C_y, C_z) is the bottom end point of the cylinder center axis, and k is the distance between adjacent vertical samples. Second, $\mathcal{D}$ is a smooth field. This is a desirable characteristic for global optimization methods that enforce smoothness constraint. Figure 4 (right) shows an example of $\mathcal{D}$ that is generated from a 3D face model.

To locate the surface voxel of pixel (θ, ϕ), we compute a matching score $\mathcal{M}$ at each discrete level d along the corresponding ray. Our matching score at point $X_{\theta,\phi,d}$ is defined as

$$\mathcal{M}(\theta, \phi, d) = \max\big(ncc(\mathbf{p}_i, \mathbf{p}_{i-1}), ncc(\mathbf{p}_i, \mathbf{p}_{i+1})\big), \quad (2)$$

where i is the index of the image that is most fronto parallel to $X_{\theta,\phi,d}$. $\mathbf{p}_i$ is the projection of $X_{\theta,\phi,d}$ in image I_i, and $ncc(\mathbf{a}, \mathbf{b})$ is the normalized cross correlation of two image patches centering at pixel $\mathbf{a}$ and $\mathbf{b}$ respectively. Ideally, the matching score is maximum when the voxel is closest to the surface. However, in textureless or shadow regions, such a local method produces poor results.

Our reconstruction results are further optimized by exploiting the smoothness property of a facial surface. We use the two-pass dynamic programming (DP) method presented in [17] that optimizes $\mathcal{D}$ in both horizontal and vertical directions. In a typical dynamic programming scheme, our

optimization can be performed along a horizontal scanline ϕ_i that maximizes the following function

$$E_h(\mathcal{D}(\theta,\phi_i)) = \sum_\theta \mathcal{M}(\theta,\phi_i,\mathcal{D}(\theta,\phi_i)) - \sum_\theta \rho(\mathcal{D}(\theta,\phi_i) - \mathcal{D}(\theta',\phi_i)) \quad (3)$$

where ρ is an increasing function that penalizes discontinuity between two adjacent points, and θ' is the previous point of θ along the scanline. Most papers use Potts model [11] for the function ρ that accounts for discontinuities. Since $\mathcal{D}$ is smooth, we use the following equation instead

$$\rho(t) = \alpha \times t^4 \quad (4)$$

In a two pass DP scheme, the first pass is performed along horizontal scanlines, which computes a bias term for second pass DP defined as

$$E_{\bar{h}} = E_{h_1} + E_{h_2} \quad (5)$$

E_{h_1} and E_{h_2} are E_h computed in the forward (increasing θ) and backward (decreasing θ) directions respectively. Then the second pass dynamic programming performs optimization along a vertical scanline θ_j to maximize the following function

$$E_v(\mathcal{D}(\theta_j,\phi)) = \sum_\phi \mathcal{M}(\theta_j,\phi,\mathcal{D}(\theta_j,\phi)) + \sum_\phi E_{\bar{h}}(\theta_j,\phi,\mathcal{D}(\theta_j,\phi)) - \sum_\phi \rho(\mathcal{D}(\theta_j,\phi) - \mathcal{D}(\theta_j,\phi')) \quad (6)$$

Figure 5 (a) shows an example after optimization. Except for the collapsed nose tip, other regions are well reconstructed. This is due to strong highlights in this region that cause the optimization to over smooth it. We therefore introduce the contour lines in the left and right profile images to provide a boundary constraint for the reconstruction. Let Q denotes the family of planes that pass through the cylinder center axis. We make a reasonable assumption that all the points along the center contour belong to one of the planes. The best fitting plane $Q_{\hat{\theta}}$ can be determined by

$$\hat{\theta} = \arg\min_\theta \sum_i E_{proj}(X_\theta^i, \bar{\Omega}^i), \quad (7)$$

where X_θ is the set of 3D points on plane Q_θ which correspond to the matches in $\bar{\Omega}$. $E_{proj}(\cdot)$ is the total reprojection error of a 3D point X_θ^i with respect to $\bar{\Omega}^i$.

Once $\hat{\theta}$ is determined, we apply the silhouette constraint by setting the matching score of the voxels along the contour to a very large number, and run the dynamic programming algorithm again. Since those voxels are set to a large number, the optimal path definitely goes through them, thus reconstructing the correct shape. Figure 5 (b) shows the result on the same person after applying the silhouette constraint. We can see that the subject's profile is then very well reconstructed.

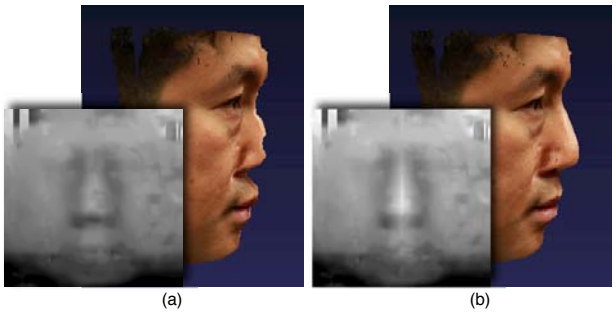

Figure 5. Reconstruction results: (a) without using and (b) using profile contours.

Note that the number of discrete depth levels in cylindrical coordinates is relatively small, which leads to less computation complexity. Most importantly, the consumed memory is low enough for our dense reconstruction (both ncc computation and 2 pass dynamic programming) to run on a GPU platform in a few seconds. The computation speed is shown in our experiments.

4. Experimental Results

We evaluate our method using two datasets. The first dataset contains synthetic, but realistic, face models and known cameras, which allows us to quantitatively define the performance with respect to the ground truth. The second dataset includes a set of real facial images. First, to show our problem is challenging, we tried both our datasets on several commercial software packages, including Boujou [3], Arc3D [2], and Photosynth [6]. All of them fail in to reconstruct reasonable camera poses. We also test our synthetic data sets on state of the art PMVS [14] providing ground truth camera poses. However, it fails to establish an initial sparse set of patches for further reconstruction in its feature matching stage.

4.1. Synthetic Face Models

The dataset consists of 10 synthetic 3D faces, including different genders and ethnic groups, which are generated by the FaceGen [4] software. Each face model has a mesh and a texture map. To obtain photo realistic skin texture maps, all texture maps in the synthetic faces are replaced with real facial images from the real dataset. We then use a 3D rendering software package [1] to simulate lighting condition similar to the one we have when taking real pictures, and render the views from different angles.

The following 3 experiments are based on the synthesized images. For each face, we produce a 180×150 ground

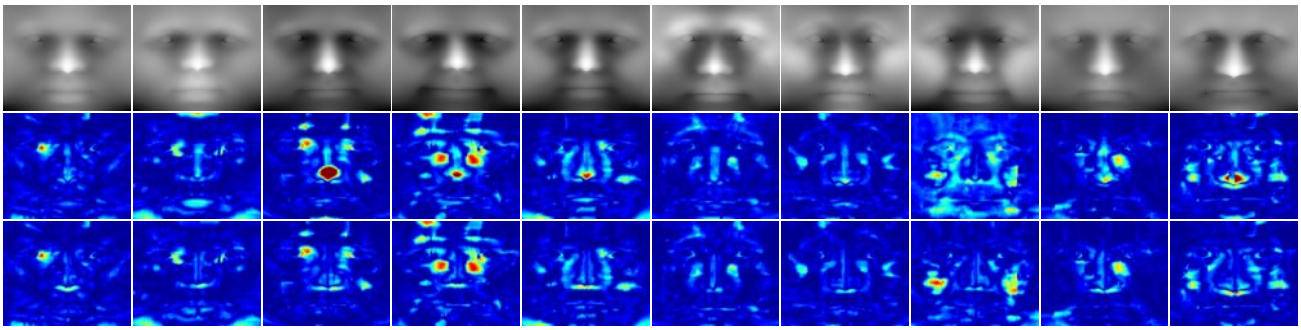

Figure 6. Reconstruction error comparison: with profile vs. without profile. Top row: ground truth depth maps. Middle row: error maps of the reconstruction without using profile information. Bottom row: error maps of the reconstruction using profile information.

truth depth map from the model that covers the central area of the face, as shown in Figure 6 (top row).

4.1.1 Accuracy of reconstruction using silhouette

We compare the enhancement by using additional silhouette information in the dense reconstruction step. Figure 6 middle row and bottom row are the error maps of each individual that use and do not use silhouettes respectively. We can see that by using silhouette information, results are more accurate particularly for the nose region.

To see the accuracy of the reconstruction, Figure 7 shows the mean error distribution of our 10 data sets. First, using profile contours provides better results, which is consistent with the results shown in Figure 6. Second, by using profile contours, 82.5% of the points show less than 2mm error. Compared with modern MVS approaches, our method can still produces comparable results even without prior knowledge of the camera poses.

4.1.2 Accuracy in imperfect profile views

As mentioned earlier in Section 3.1, the correspondences obtained by matching points on the two profile contours are only accurate when the subject's facing direction is exactly orthogonal to the camera. In this experiment, we analyze the accuracy degradation with respect to the angle of deviation. We perform the experiments on 3 sets of images that have $5°$, $10°$, and $15°$ of deviation respectively, and measure the error in terms of reconstructed structure and the reconstructed camera poses. Given the true camera translation and rotation, the relative error of the estimated translation and rotation are computed as $E_T = \|t_{true} - t\|/\|t\|$ and $E_R = \|r_{true} - r\|/\|r\|$ respectively. The results by averaging the 10 data sets are shown in Figure 7 (b) and (c).

Results show that the accuracy in terms of reconstructed structure and reconstructed camera poses both degrade slowly within $10°$ of deviation. This verifies the robustness of our algorithm since the profile views are hardly perfect.

4.2. Real Face Models

In real face experiments, we collect 10 image sets with a Canon Rebel T1i camera and a EF-S 18-55mm f/3.5-5.6 IS lens. The resolution of each image is 2400×1600, and the distance between two eyes is roughly 200 pixels.

4.2.1 Computation complexity

Sparse reconstruction, including feature extraction, feature matching, 3-view correspondence extraction, and the final 2 pass bundle adjustments takes less than 5 minutes in average. For dense reconstruction, we use 192 depth levels, while the size of θ and ϕ are both 540. This amounts to 56 millions voxels, for which we run both per voxel NCC computation and dynamic programming optimization on a NVIDIA GTX 295 graphic card (240 stream processors, processors clock 1242 MHz), and complete the task in an average of 25 seconds. This is much faster than PMVS, which usually takes hours on similar inputs.

4.2.2 Quality of reconstruction results

To show the difficulty in getting 3-view correspondences, Figure 8 shows the number of correspondence inliers vs. total number of correspondences in our data sets. The numbers from 1 to 10 correspond to the 10 subjects, while the three pair of numbers in each column are the number of 3-view correspondence inliers (left) and the number of 3-view correspondences (right) in Ω_1, Ω_2, and Ω_3 respectively. We can see the number of correspondence inliers in Ω_2 is much smaller than that in Ω_1 and Ω_3. In the most extreme case, no correspondence inliers can be extracted from Ω_2 for subject #4. However, we are still able to reconstruct visually reasonable facial shape of subject #4 as shown in Figure 9 (top row). Also note that RANSAC based camera pose estimation approach can never work given the limited number of correspondence inliers in Ω_2. This demonstrates the robustness of our proposed camera calibration approach.

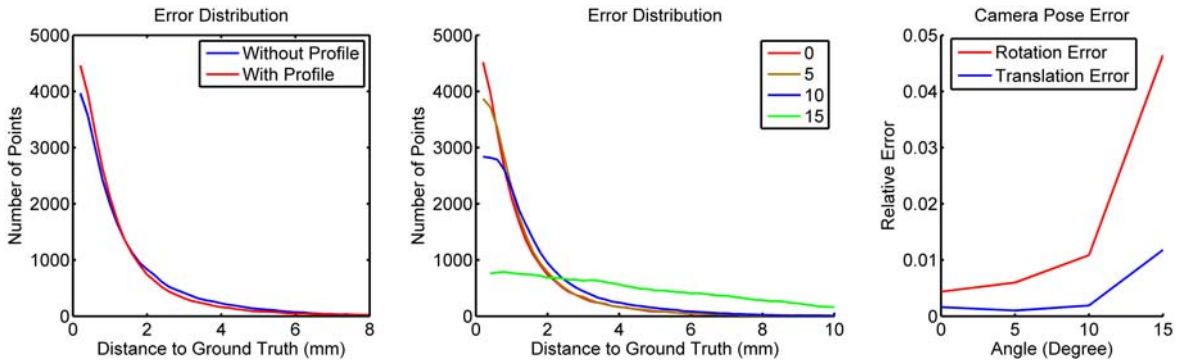

Figure 7. (a) Mean error distributions of the 10 synthetic data sets: with profile vs. without profile, (b) Mean error distributions of the 10 synthetic datasets with respect to different amount of deviation angle in the profile views, and (c) Mean camera pose errors in terms of rotation and translation with respect to different amount of deviation angle in the profile views.

	1	2	3	4	5	6	7	8	9	10
Ω_1	11/38	34/62	28/41	12/33	48/151	41/84	72/100	40/58	41/75	30/66
Ω_2	5/29	3/26	2/16	0/13	6/29	2/22	17/68	8/23	2/19	3/17
Ω_3	3/21	30/53	44/59	49/78	63/84	76/106	59/105	45/69	28/50	29/39

Figure 8. Number of 3-view correspondence inliers (left) vs. total number of 3-view correspondences (right) in each group of images for each individual.

Although there is no ground truth data to compare with, our results are visually accurate. Figure 9 shows a few results from real image sets. The left and right columns correspond to the reconstruction results without and with using contours respectively.

5. Summary

We presented a method to generate a highly accurate 3-D model from wide-baseline images acquired in a weakly calibrated setup. To perform accurate calibration, we propose an iterative bundle adjustment approach that produce very stable results given noisy correspondences with very few number of inliers. Contours in the profile views are matched to provide additional calibration constraints.

In dense reconstruction, we use a face-specific cylindrical representation to solve a global optimization problem for N-view dense aggregation, and again use profile contours to provide constraints for the optimization. Experimental results, using carefully designed datasets and real images, confirm the accuracy and stability of our method on the challenging inputs. Our method is also shown to outperform other state of the art methods in terms of robustness and efficiency.

References

[1] 3ds max. http://usa.autodesk.com. 5

[2] Arc 3d. http://www.arc3d.be/. 2, 5

[3] boujou. http://www.2d3.com. 2, 5

[4] Facegen. http://www.facegen.com. 5

[5] Meshlab. http://meshlab.sourceforge.net. 8

[6] Photosynth. http://photosynth.net/. 2, 5

[7] T. Akimoto, Y. Suenaga, and R. S. Wallace. Automatic creation of 3d facial models. *IEEE Comput. Graph. Appl.*, 13(5):16–22, 1993. 2

[8] B. Amberg, A. Blake, A. Fitzgibbon, S. Romdhani, and T. Vetter. Reconstructing high quality face-surfaces using model based stereo. *ICCV 2007*, 0:1–8. 1, 2

[9] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. *Computer Vision and Image Understanding*, 110(3):346–359, 2008. 3

[10] B. Boufama and R. Mohr. Epipole and fundamental matrix estimation using virtual parallax. In *ICCV 1995*, page 1030. 3

[11] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11):1222–1239, 2001. 5

[12] D. Fidaleo and G. Medioni. Model-assisted 3d face reconstruction from video. In *AMFG 2007*, pages 124–138. 2

[13] P. Fua. Using model-driven bundle-adjustment to model heads from raw video sequences. In *ICCV 1999*, pages 46–53. 2

[14] Y. Furukawa and J. Ponce. Patch-based multi-view stereo software. http://grail.cs.washington.edu/software/pmvs. 2, 5

[15] Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. *PAMI*, 2009. 2

[16] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 2

[17] J. C. Kim, K. M. Lee, B. T. Choi, and S. U. Lee. A dense stereo matching using two-pass dynamic programming with generalized ground control points. In *CVPR 2005*, pages 1075–1082. 4

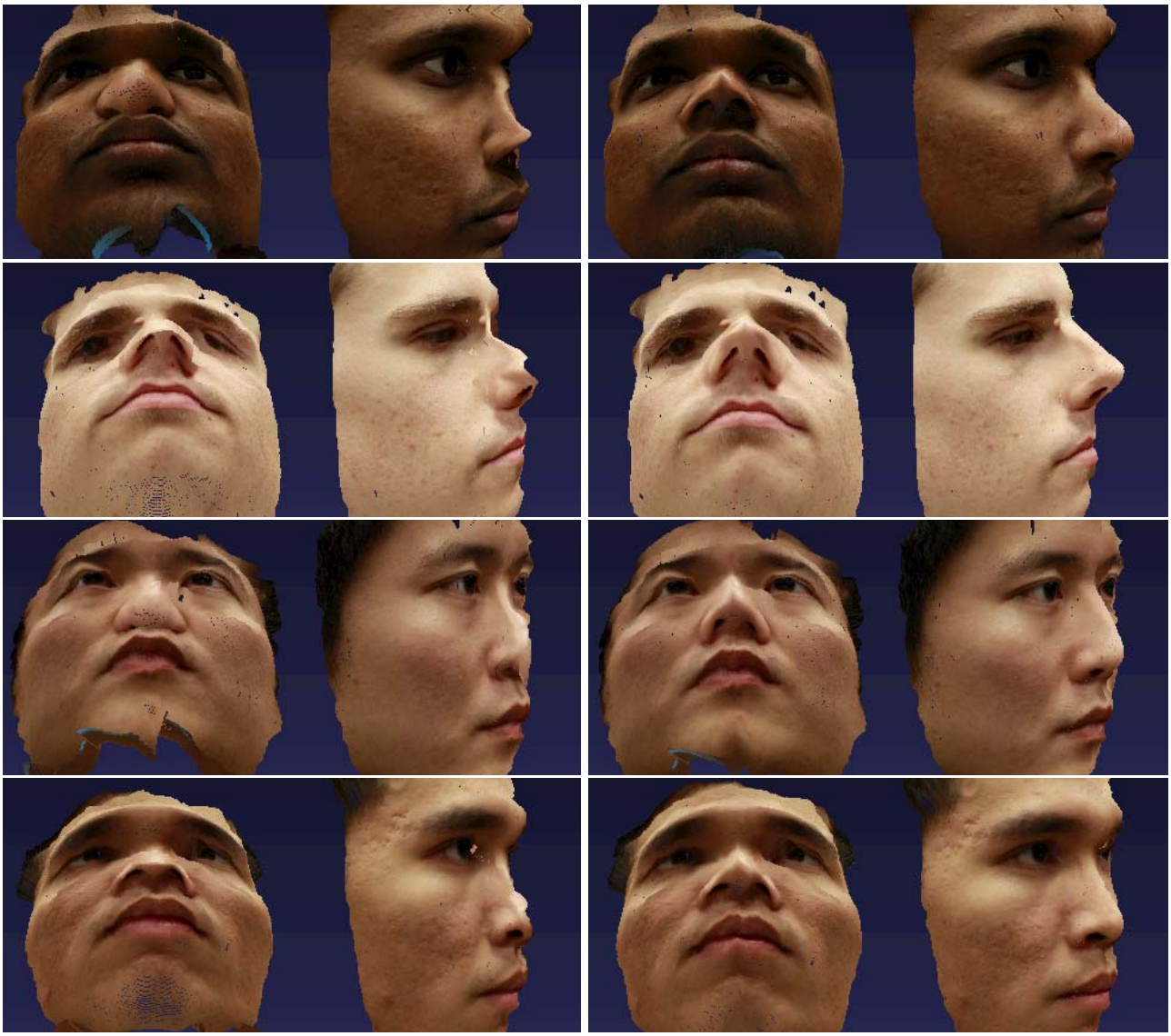

Figure 9. Reconstruction of real images (291600 points rendered in MeshLab [5]): without profile contours (left) vs. with profile contours (right).

[18] V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnp: An accurate o(n) solution to the pnp problem. *Int. J. Comput. Vision*, 81(2):155–166, 2009. 3

[19] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 27(10):1615–1630, Oct. 2005. 3

[20] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modeling with a hand-held camera. *Int. J. Comput. Vision*, 59(3):207–232, 2004. 2

[21] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR 2006*, pages 519–528. 2

[22] Y. Shan, Z. Liu, and Z. Zhang. Model-based bundle adjustment with application to face modeling. In *ICCV 2001*, pages II: 644–651. 2

[23] C. Strecha, R. Fransens, and L. V. Gool. Wide-baseline stereo from multiple views: A probabilistic account. In *CVPR 2004*, volume 1, pages 552–559. 2

[24] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *CVPR 2008*, pages 1–8. 2

[25] L. Tang and T. Huang. Automatic construction of 3d human face models based on 2d images. In *ICIP 1996*, pages III: 467–470. 2

[26] E. Tola, V. Lepetit, and P. Fua. Daisy: An efficient dense descriptor applied to wide baseline stereo. *PAMI*, 2009. 2