Model-Assisted 3D Face Reconstruction from Video

Douglas Fidaleo and Gérard Medioni

Institute for Robotics and Intelligent Systems
University of Southern California
dfidaleo@gmail.com, medioni@usc.edu

Abstract. This paper describes a model-assisted system for reconstruction of 3D faces from a single consumer quality camera using a structure from motion approach. Typical multi-view stereo approaches use the motion of a sparse set of features to compute camera pose followed by a dense matching step to compute the final object structure. Accurate pose estimation depends upon precise identification and matching of feature points between images, but due to lack of texture on large areas of the face, matching is prone to errors.

To deal with outliers in both the sparse and dense matching stages, previous work either relies on a strong prior model for face geometry or imposes restrictions on the camera motion. Strong prior models result in a serious compromise in final reconstruction quality and typically bear a signature resemblance to a generic or mean face. Model-based techniques, while giving the appearance of face detail, in fact carry this detail over from the model prior. Face features such as beards, moles, and other characteristic geometry are lost. Motion restrictions such as allowing only pure rotation are nearly impossible to satisfy by the end user, especially with a handheld camera.

We significantly improve the robustness and flexibility of existing monocular face reconstruction techniques by introducing a deformable generic face model only at the pose estimation, face segmentation, and preprocessing stages. To preserve data fidelity in the final reconstruction, this generic model is discarded completely and dense matching outliers are removed using tensor voting: a purely data-driven technique. Results are shown from a complete end to end system.

1 Introduction

3D face models are important for a wide array of applications including surveillance, computer gaming, military simulation, virtual teleconferencing/chat, and surgical simulation. Most existing techniques for face reconstruction require special hardware or multiple cameras to create faces which prevents wide-spread adoption. This is unfortunate, as the rapidly increasing quality of consumer digital cameras has reduced the need for such specialized equipment for high quality reconstruction of faces.

S.K. Zhou et al. (Eds.): AMFG 2007, LNCS 4778, pp. 124–138, 2007.

The goal of this work is to reconstruct high accuracy 3D geometry of human faces from 2D video sequences acquired from a single, consumer quality, monocular, video camera. Performing reconstruction from a single camera enables leveraging of existing ubiquitous surveillance and web camera infrastructure for security and entertainment purposes.

Structure from motion has been used in single camera architectural and terrain modeling with great success [1]. These domains tend to have many distinguishable features that simplify the pose estimation and dense reconstruction phases. Reconstruction of faces from images is challenging as the face has very little texture outside of the eye, eyebrow, and mouth region. The image may be corrupted by noise, shadows, or other environmental aberrations which, combined with the lack of texture, makes it difficult to identify precise feature points required for accurate pose estimation. Inaccuracies in pose estimation translate into geometric distortions and noise.

Existing *model-based* work on single camera face reconstruction utilizes very strong prior knowledge of the structure and appearance of faces to constrain the reconstruction [2][3][4][5][6]. However, by imposing these constraints, these methods do not capture the subtle details present in the original face that are critical for recognition. *Data-driven* approaches can capture more subtle details from faces by disregarding strong face model constraints and relying strictly on the observed data [7][1]. However, rejecting a regularizing model can be dangerous due to more prevalent outliers on faces with limited texture.

It turns out that the use of a generic model for face reconstruction is not always bad, as long as it is used at the appropriate points in the reconstruction process.

We make two significant contributions to face reconstruction literature. First, it was observed in [8][9] that introduction of prior knowledge of the face can significantly improve the stability and accuracy of face pose estimation. We use this knowledge by incorporating the results of a model-based face tracker into the pose estimation stage. To prevent bias in the final face geometry, and to allow for reconstruction of face features such as beards that are not in the model prior, this model is discarded after pose estimation. Second, to deal with outliers in the dense reconstruction phase, we use tensor voting to perform model-free outlier rejection. Both of these contributions result in significant improvement of face reconstruction generality and robustness over existing methods.

This paper describes the entire end-to-end reconstruction system. It should be noted that, while this paper focuses on faces, the technique could be easily extended to reconstruction of other textured surfaces for which a rough initial estimate of the object's structure is available.

2 Previous Work

Several active and passive techniques exist for creating 3D faces. The majority of past work has required the use of specialized hardware, e.g. laser range finders, stereo camera rigs, or structured light projectors. These methods can achieve outstanding accuracy but the hardware requirement prevents use with existing imaging infrastructure such as consumer digital cameras and surveillance cameras. In this work we are only interested in reconstruction methods from a single camera.

Two categories of single camera reconstruction methods exist: data-driven and model-based. Data driven methods are largely based on textbook structure from motion techniques [10]. An example of such an approach is the body of work by Pollefeys et. al. for reconstruction of architectural scenes from a single moving camera. In [1] camera pose is estimated using self calibration on a sparse set of points matched between local image pairs. Following camera estimation and image rectification, dense feature matching is performed to compute a dense disparity map which may be triangulated and textured using image intensity information. Very nice results are achieved with architectural objects having large amounts of surface texture. The direct application of this approach to faces is challenging as faces have large areas of relatively uniform texture and are prone to highlights. This can result in large amounts of noise due to uncertainty in the matching process.

Pesenti et. al. use a similar framework for reconstruction applied to faces [7]. Epipolar geometry estimation and self calibration in Pollefeys' work is replaced with bundle adjustment. The initial bundle configuration is derived from a simplified motion model; assuming the head undergoes pure rotation in a given triplet of frames.

A common approach to deal with geometry and pose uncertainty is to introduce very strong prior models of the face. Fua uses bundle adjustment and a strong prior to model heads [5]. The prior model is tightly integrated with both pose estimation and modeling and hence the final reconstruction, while visually appealing, cannot deviate far from the original model. Morphable models [3] assume the face is a linear combination of a set of basis face shapes and appearance. The reconstruction process involves a minimization of the image reconstruction error over basis vector weights as well as camera and lighting parameters. Similarly, work by Shan et.al. parameterizes the face by a generic model and a set of face "metrics" or deformation parameters [4]. DeCarlo et.al. use optical flow and a component-wise deformable decomposition of the face to regularize feature motion and model the face [2].

By design, these approaches are limited by the constructed parameter space. Though modeling can be performed from single images real faces have variations not within the space defined by the basis shapes, resulting in reconstructions that bear a signature resemblance to a generic (or mean) face. Optimization is performed using an image based error functional and can be very slow to converge. To constrain the search space fewer basis shapes are used, consequently the resulting reconstructions do not capture the subtle details present in the original face. Recently in work by Ilic et. al., the silhouettes from multiple views have been integrated with implicit surfaces to bias a Morphable Model solution towards a more faithful reconstruction [11].

The approach most similar to ours uses a multi-view stereo approach for monocular reconstruction of faces [7]. A set of individual face scans is derived from each pair of images in a video sequence of a user rotating his head. These scans are merged into a single point cloud which is processed for outliers and triangulated to form a final dense mesh. The quality of the final mesh depends on the proper alignment of the individual scans, and hence, the proper estimation of the camera poses in the images.

Pose estimation is dependent on precise feature matching between images. Errors in feature matching propagate to tracking and ultimately to pose and reconstruction errors. Pesenti et.al. determine feature candidates on each face image with an interest operator [12] and match points with maximum correlation. However, most faces have very little texture in the cheek and forehead regions. Hence, unconstrained correlation produces egregious outliers which results in extremely poor pose estimation and reconstruction. To remove outliers and initialize the bundle adjuster, Pesenti uses RANSAC and a pure rigid rotation motion model. While this removes gross outliers, foreshortening effects and lack of texture will still produce erroneous matches that are consistent with the motion model, resulting in poor pose estimation and reconstruction. Furthermore, head motion is almost never purely rotational. Neglecting translation in the motion model imposes serious constraints on the kinds of input sequences that can be reliably reconstructed.

Undoubtedly, there are cases where these errors are small and the resulting reconstruction is reasonable (as shown in the paper), however it is suspected due to the fact that there was no continuation of that work in either journal or conference publication, that these are isolated cases and in general the method fails.

Our work relaxes the constraints imposed in [7] by borrowing principals from the model-based literature. A strong face model is used to dramatically enhance pose accuracy without biasing the geometry solution. Our approach also enables the use of still images with wider baselines for reconstruction. Outliers in dense matching are handled with tensor voting, a model free surface extraction technique, which only becomes possible after accurate pose estimation.

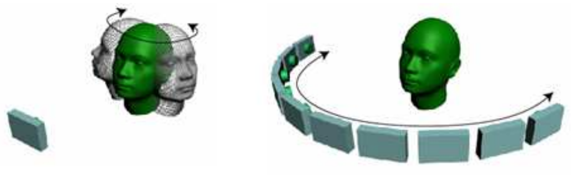

Fig. 1. The problem of reconstructing a single rigid object (static or mobile) with a single camera (static or mobile) can be equivalently modeled as reconstruction from multiple static cameras with the same relative external coordinate system

3 Reconstruction Overview

Given a sequence of images of the face $I = \{I_0, I_1 ... I_k\}$ in different poses our goal is to transform this sequence into an accurate textured 3D model M. Though not required, for simplicity we will assume that the subject's face is frontal in frame I_0, and that indices k are ordered in time.

It is assumed that the subject remains expressionless through the duration of the sequence, providing a rigid surface for reconstruction. Given these parameters and constraints, the "moving head - static camera" problem can be modeled equivalently as an array of multiple static cameras placed around a static object, each acquiring a different view of the face as shown in figure 1.

Figure 2 shows a diagram of the reconstruction process described in detail in this document. Novel contributions are highlighted. We begin with an input video sequence of a subject. It is presumed that multiple views of the subject's head are present in the sequence: as much coverage as desired for the final reconstruction. We utilize a 3D face tracking algorithm to derive an initial head pose estimate and mask for the face. Optimal views are selected from the set of images and passed to a sparse feature tracking module. Sparse feature tracking produces a set of feature correspondences for each successive image pair. Global optimization is performed over the entire sequence of feature points and cameras to refine the tracking camera estimate and compute the 3D structure at the sparse feature locations.

The optimized camera positions are used to rectify pairs of images, constraining the search space for corresponding feature points to a horizontal scanline in the paired image. Dense feature matching is performed across pairs and correspondences are reconstructed by triangulation using the optimized camera poses resulting in a dense 3D point cloud. Point clouds corresponding to individual pairs are merged into a single cloud and outliers are removed.

A connected surface is fit to the final cleaned point cloud and the face texture is acquired from a frontal image. The final result is a clean 3D mesh that is ready to be used for animation, recognition, or rendering.

4 Model-Based Pose Estimation

In the interest of remaining purely data-driven, Pesenti et.al. assume the head in the video sequence undergoes pure rotation and apply a simple motion constraint at the sparse feature matching stage. These constraints are necessary to achieve a decent pose estimate to initialize the bundle adjustment algorithm. Unfortunately, these restrictions are much too prohibitive for a robust system.

We make the observation that we can reap the benefits of a geometric model at the pose estimation stage without biasing the shape at the final reconstruction stage. Therefore, in our system, the initial pose estimate is obtained from the 3D head tracker developed by Vacchetti et.al. [9]. This tracker relies heavily on an approximate generic model of the subject's face.

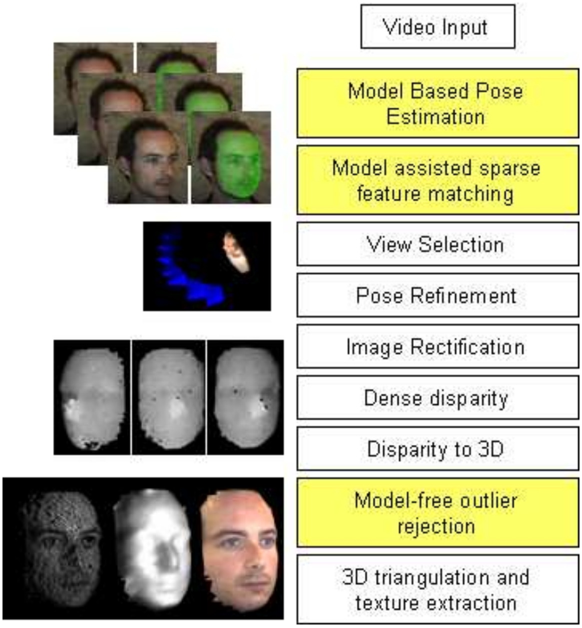
Video Input
Model Based Pose Estimation
Model assisted sparse feature matching
View Selection
Pose Refinement
Image Rectification
Dense disparity
Disparity to 3D
Model-free outlier rejection
3D triangulation and texture extraction

Fig. 2. Overview of the reconstruction system presented in this paper. Highlighted modules are novel contributions.

4.1 Tracking

This section presents a brief overview of the tracking approach, but the reader is referred to the original paper [9] for details.

In the initial frame I_0 a 3D face model is manually aligned to the face. This establishes a reference keyframe consisting of a set of 2d feature locations detected on the face with a Harris corner detector and their 3D positions estimated by back-projecting onto the model. The keyframe accuracy is dependent on both the model alignment in the keyframe image, as well as the geometric structure of the tracking mesh.

As the subject rotates her head, there may be several newly detected feature points not present in any keyframe that are useful to determine inter-frame motion. These points are matched to patches in the previous frame and combined with keyframe points for pose estimation.

The current head pose estimate for I_t serves as the starting point for a local bundle adjustment. Classical bundle adjustment is typically a time consuming process, even when a reasonable estimate of camera and 3D parameters is provided. However, by constraining the 3D points to lie on the surface of the tracking model, the method is modified to run in real-time without substantial sacrifice in accuracy. When an accurate 3D model of the tracked object is used, reported accuracy approaches that of commercial batch processing bundle adjustment packages requiring several minutes per frame.

4.2 Model-Assisted Sparse Feature Matching

We use the result of the model-based tracker to improve the number and accuracy of sparse feature points used in the bundle adjustment stage. Indeed, the accuracy of sparse feature matching is critical as errors in feature matching will propagate into the bundle solution. In general, this problem is difficult, as local image patches may resemble more than one patch in another image, especially in large textureless areas. Our approach utilizes salient feature points and model constraints to converge on a set of accurate feature matches.

Extraction of the feature matches is divided into 2 parts: *feature matching* and *feature chaining*.

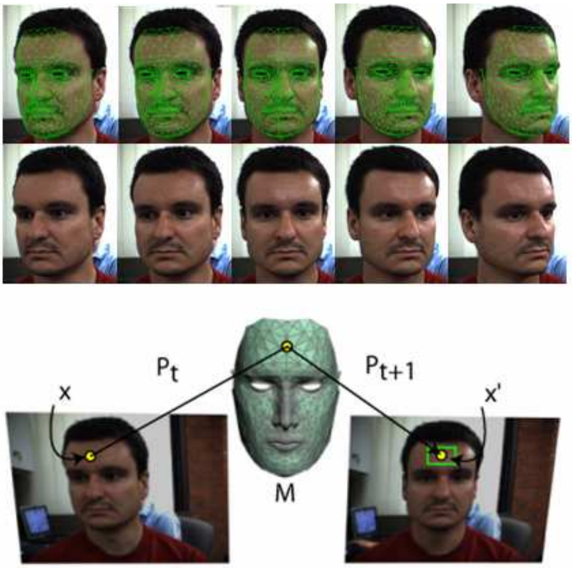

Fig. 3. (top) Model-based face tracking. (bottom) Model assisted feature matching.

Feature matching is performed on consecutive pairs of images I_t and I_{t+1}. Feature points are first detected in each image using a Harris corner detector. A sample window is extracted at the feature location x. The predicted match location of x in the corresponding frame is computed using the transfer function

$$x' = T(x, P_t, P_{t+1}, X) \tag{1}$$

This function is illustrated in Figure 3. T uses P_t to back-project x to the the 3D generic model M. The corresponding 3D point X is then projected to the I_{t+1} using P_{t+1}. This new 2D location x' is the predicted feature location. Note that if we were to use this location as the feature match, the resulting pose estimate would be identical to the tracking estimate, when we know to be imprecise.

Instead, a search area is defined around the predicted location x'. In this area, we find the best correspondence candidate c via normalized cross correlation.

If the match score is below a threshold τ, the match is rejected. In practice we set $\tau = .9$ and the search area to be $21x21$ to ensure only high quality correspondences. Features are chained across image pairs.

4.3 View Selection

The bundle adjustment process is sensitive to the angular baseline between images as well as the number of matched feature points in each frame. The baseline between virtual camera locations is directly related to the uncertainty of the reconstruction estimate [10]. A narrow baseline increases measurement uncertainty but ensures visibility of points in multiple images. A wide baseline decreases measurement uncertainty but increases the chances that a point is occluded.

In practice, we select an angular baseline between 8 and 15 degrees and select frames with at least 6 points matched across a minimum of 3 frames (determined experimentally).

4.4 Bundle Adjustment and Bundle Surface Extraction

The tracker provides a rough estimate of the head pose. However, to accommodate the real-time constraint, only current, previous, and key frames are used in the optimization. This narrow view range causes misalignment between pairs of images at wider baselines and hence distortion in the reconstruction. We therefore introduce an offline pose estimation stage that considers the full set of views.

Given the 2D feature chains identified in section 4.2 we perform a global bundle adjustment to refine the camera poses using the tracking estimate as a starting point for the optimization [13].

The 3D point cloud resulting from bundle adjustment is a coarse estimate of the structure of the subject's face. We create an interpolated version of this surface to filter the dense face reconstruction using scattered data interpolation via radial basis functions [14]. We refer to the resulting interpolated model as the *bundle surface.*

The refined camera poses are used to rectify the images to assist in the dense feature matching stage. We perform rectification using standard computer vision techniques [10].

5 Model-Assisted Face Segmentation

Using the model based face tracker, we do not require the face to be in the foreground as in [7]. The region under the tracking model provides a good approximation of the face area and is used to define a binary reconstruction mask where white pixels are to be reconstructed and black omitted. The initial mask is defined by the entire area under the tracking model as shown.

5.1 Skin Mask

Due to alignment problems, the reconstruction mask derived from the tracking model may not cover the entire face region. Too little coverage may be sufficient for recognition applications, however for animation a complete head model is generally required. If the mask is too large, non-face pixels may interfere with the reconstruction process.

We refine the initial segmentation by segmenting the skin region belonging to the face; learning the skin color distribution and classifying skin pixels.

We make the basic assumption that skin color is largely constant over the face with most variations occuring due to lighting. The skin color sample is defined by the area under the tracking mask. Separation of color from illumination is difficult in the standard RGB space. We therefore convert the image window to HSV space, and discard all but the hue component. The majority of the image under the face rectangle is skin and therefore the strongest mode of a histogram plot of the hue data will belong to the skin color.

A Gaussian model is fit to the data. Each pixel in the image is then classified as skin/non-skin with respect to this model using a fixed threshold. The binary face mask is augmented with new skin pixels.

This process generally performs quite well, but may fail if the image contains multiple overlapping faces or areas with color distributions resembling skin. In this case, the initial reconstruction mask may be used.

5.2 Highlight Removal

Specular highlights on the face cause problems in the reconstruction. The center of the highlights saturate and hence obscure any texture in the area. As the face rotates, the highlights shift, and cause errors in matching along the highlight border. Saturated areas can be detected by analyzing the variance of a window of pixels. If the variance is below a threshold, this window is removed from the reconstruction mask M. To deal with highlight boundaries which may not saturate, the highlight mask is dilated.

5.3 View Angle Filtering

Due to foreshortening effects on face texture, surfaces nearly orthogonal to the camera view ray will have greater matching uncertainty. We use the normal map derived from the tracking model to compute this angle and reject points whose approximated normal is greater than 50 degrees from the view ray. Each mask is combined to produce the final reconstruction mask as shown in Figure 4.

6 Model-Free Dense Reconstruction

6.1 Dense Feature Matching

Dense feature matching determines a dense set of corresponding points in each pair of rectified images. The matching is restricted to the area covered by the

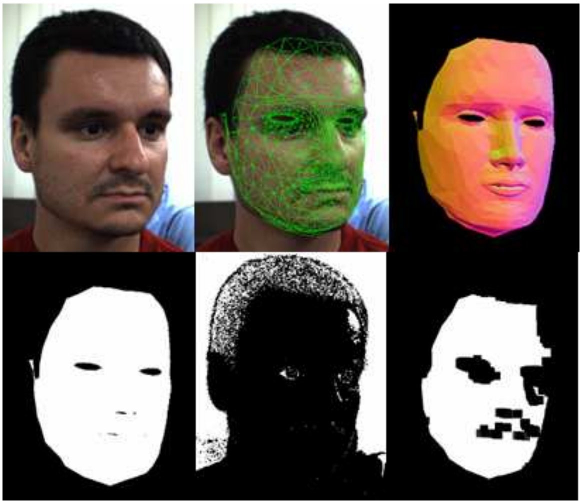

Fig. 4. (top) Input image, aligned tracking mask, normal map. (bottom) Initial reconstruction mask, highlight mask, and final combined reconstruction mask.

reconstruction mask from Section 5. For each pixel in one image, a template is extracted using a fixed window size. This template is matched along the corresponding epipolar line in the paired image. A minimum correlation threshold and restricted disparity range suitable for faces is used to reduce the number of spurious matches. Multiple candidate matches can be retained, but locations with flat correlation plots (no obvious peak) are rejected, as this is an indicator of a textureless, or possibly occluded region.

The result of the matching process is a disparity volume where each (x,y,d) value maps a pixel (x,y) in one rectified image to a pixel (x+d,y) in the paired image.

6.2 3D from Disparity

The known camera poses allow us to convert each disparity value to a true 3D point by triangulation. Each disparity pixel is transformed to the original image space using the inverse of the rectifying transform. The 3D location of each match is given by the intersection of the rays passing through the camera optical centers and the corresponding feature matches in the image planes. In practice, due to imperfect feature matching and camera estimates, these lines will not intersect exactly. We therefore compute the 3D point that minimizes the orthogonal distance to the two rays [10].

6.3 Outlier Rejection

Tensor Voting. Errors in feature matching result in considerable reconstruction noise. If the noise is uncorrelated within and between views, it will appear

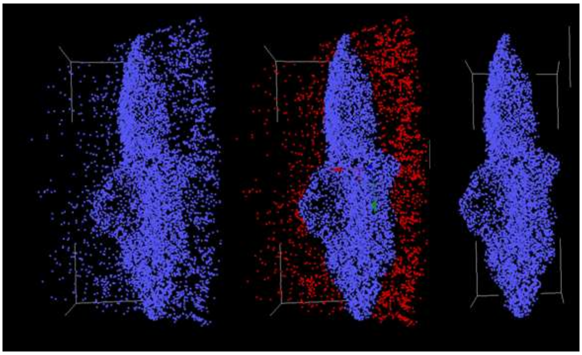

Fig. 5. Tensor Voting Outlier rejection. (left) original points. (middle) Outliers shown in red. (right) 3D points after tensor voting outlier rejection.

as sparse, high frequency variations in the 3D structure. Correct matches will, however, be correlated between views due to the smoothness and continuity of face structure. We use tensor voting to uncover this correlation structure and reject outliers.

Tensor voting is a technique used to uncover the intrinsic dimensionality of data in arbitrary dimensional spaces. Data points share information with their neighbors encoding their saliency in a given dimension. In 3D tensor voting each 3D point can be encoded as a ball tensor (no orientation preference) or stick tensor (if normal information at the point is available). In the voting process, orientation and structural saliency information (dimensionality) are shared with neighboring points. Neighbors with similar structure reinforce each other with the amount of structural reinforcement defined by the initial structural saliency.

When the individual point clouds are integrated, points on face surface are correlated and will reinforce each other during tensor voting. Incorrect matches due to sensor noise, lack of texture, or other artifacts will result in uncorrelated noise in the 3d structure. These points will have very low surface saliency and are removed by simple thresholding. This approach is similar to that of Mordohai et. al. for denoising of disparity maps [15].

In practice, a good initial estimate of point normals is preferred to blindly encoding points as ball tensors. We therefore use the bundle surface to approximate the point normals. Fixing the normal as the first eigenvector in a 3x3 eigensystem, the remaining basis vectors are computed using singular value decomposition (SVD). Initial surface saliency (defined by the difference in magnitude of the first two eigenvectors) is set uniformly for all points initially.

As the 3D from the bundle adjustment is a very accurate sparse estimate of the face structure, these points are added to the tensor voting point set with boosted surface saliency.

After two passes of tensor voting, points with low surface saliency are removed, leaving a dense cloud of points distributed across the surface of the face as shown in Figure 5.

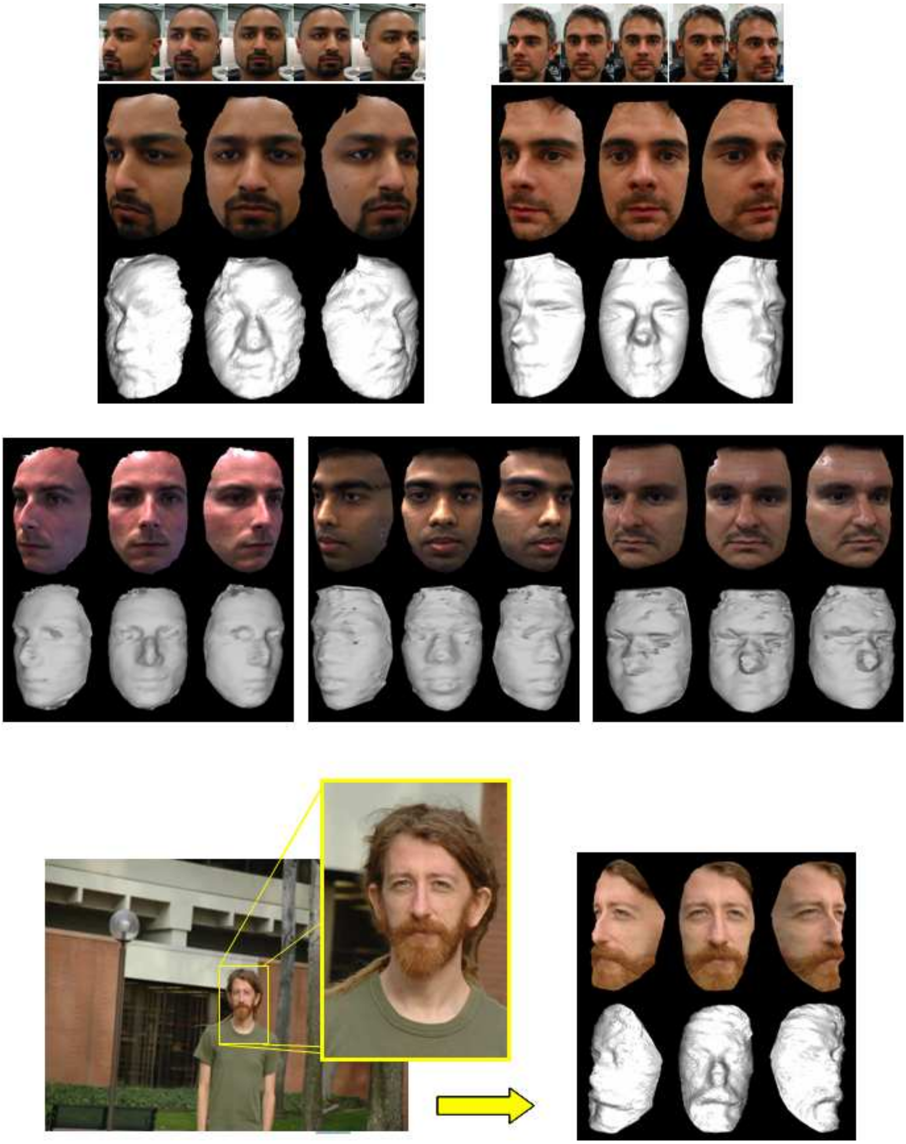

Fig. 6. Reconstruction results. (top) Reconstruction from 5 digital still images shown above models. (middle) Reconstruction from video sequence. (bottom) Reconstruction outdoors at a distance of 15 feet.

Tensor voting is computationally intensive in software. The computation time grows nearly linearly with point set size, assuming a local voting field. This implies that each reconstruction pair adds a constant amount of computational effort. With an average cost of 30 sec. per pair (reconstructing 50k points) this cost can become prohibitive for more interactive applications.

If constant computation time is desired, the final point cloud may be quantized first by voxelizing the set then reducing each voxel to a single point. Tensor voting may then be performed on the quantized set. The speed and accuracy tradeoff may be adjusted by altering the quantization resolution. In practice we break the face volume into 10^6 voxels and take the mean value for each voxel. This results in a set of $<$ 20k 3D points which can be processed by the tensor voting module in less than 15 seconds.

Voxel filtering. Tensor voting removes uncorrelated outliers, however there are many cases where correlated artifacts can arise such as on the boundary of the face or shifting highlights. The bundle surface provides a strong constraint on the allowable 3D computed in the dense reconstruction; the computed structure should not deviate far from the bundle derived structure. This structure is used to filter the data by voxelizing the interpolated bundle structure and rejecting data at a predefined distance from the bundle voxels. It should be noted that this is not the same as using a generic model constraint, as the bundle structure can be considered optimal based on the observed data.

7 Meshing

It is difficult to visualize the reconstruction results from the point clouds alone, therefore the 3D point cloud is converted to a textured 3D mesh using a standard graphics technique. The point cloud is projected to the surface of a cylinder whose axis is aligned with the axis passing through the center of the head with the cylinder axis defined by the tracking model.. The points are triangulated in the 2D cylindrical space. Texture is acquired by projecting the resulting triangles to the image using the optimized camera pose. High frequency noise in the mesh is eliminated using a volume preserving diffusion technique by Desbrun et.al. [16].

8 Results

Figure 6 shows results from the reconstruction system. We demonstrate the robustness and flexibility of the system in different environments using different camera hardware. All images used for reconstruction are 640x480 pixels. Models in the center row were taken in a standard office environment at a distance of roughly 3 feet using a PointGrey Dragonfly video camera. Our system is not limited to video sequences. The model at the top left was created using a set of 6 still images from a Canon Powershot S300 digital camera in a similar environment. We are also not limited to indoor sequences with short focal lengths. The model at the bottom was created outdoors at a distance of 15 feet. Note the ability to capture the prominent facial hair on the subject. Model based methods completely fail in such cases as these personalized variations are difficult to embed in the model space.

9 Conclusion

We have demonstrated a robust system for reconstruction of faces from a single camera. We make a fundamental observation that introduction of prior face knowledge at the pose estimation stage can significantly improve reconstruction results without biasing the geometry solution towards this prior. Performing outlier rejection using tensor voting, a purely data driven method, preserves the subtle details present in the subject's face. The use of the model based face tracker relaxes the constraints on the subject's motion and enables the use of both video sequences with small angular baseline between images, as well as a set of still images with significantly larger baselines.

References

1. Pollefeys, M., Gool, L.V., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J., Koch, R.: Visual modeling with a hand-held camera. International Journal of Computer Vision 59, 207–232 (2004)
2. DeCarlo, D., Metaxas, D.: The integration of optical flow and deformable models with applications to human face shape and motion estimation (1996)
3. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3D faces. In: Rockwood, A. (ed.) Siggraph 1999, Computer Graphics Proceedings, pp. 187–194. Addison Wesley Longman, Los Angeles (1999)
4. Shan, Y., Liu, Z., Zhang, Z.: Model-based bundle adjustment with application to face modeling. In: International Conference on Computer Vision, Vancouver, Canada (2001)
5. Fua, P.: Using model-driven bundle-adjustment to model heads from raw video sequences. In: Proceedings of the 7th International Conference on Computer Vision, Corfu, Greece, p. 4653 (1999)
6. Romdhani, S., Vetter, T.: Efficient, robust and accurate fitting of a 3d morphable model. In: ICCV 2003. Proceedings of the Ninth IEEE International Conference on Computer Vision, p. 59. IEEE Computer Society Press, Washington, DC, USA (2003)
7. Pesenti, B., Medioni, G.: Generation of a 3d face model from one camera. In: Proceedings of the 16th International Conference on Pattern Recognition, Quebec City, Quebec, Canada, pp. 667–671 (2002)
8. Fidaleo, D., Medioni, G., Fua, P., Lepetit, V.: An investigation of model bias in 3d face tracking. In: IEEE Analysis and Modeling of Faces and Gestures, pp. 125–139. IEEE Computer Society Press, Los Alamitos (2005)
9. Vacchetti, L., Lepetit, V., Fua, P.: Stable real-time 3d tracking using online and offline information. IEEE Trans. Pattern Anal. Mach. Intell. 26, 1385–1391 (2004)
10. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge, UK (2000)
11. Ilic, S., Fua, P.: Implicit meshes for surface reconstruction. IEEE Transactions on Pattern Analysis and Machine Intelligence 28, 328–333 (2006)

12. Shapiro, L., Haralick, R.: Image matching - an interest operator. In: Computer and Robot Vision Volume II, pp. 341–343. Prentice-Hall, Englewood Cliffs (1992)
13. Lourakis, M., Argyros, A.: The design and implementation of a generic sparse bundle adjustment software package based on the levenberg-marquardt algorithm. Technical Report 340, Institute of Computer Science - FORTH, Heraklion, Crete, Greece (2004)
14. Powell, M.J.D.: Radial basis functions for multivariable interpolation: a review, 143–167 (1987)
15. Mordohai, P., Medioni, G.: Stereo using monocular cues within the tensor voting framework. IEEE Transactions on Pattern Analysis and Machine Intelligence 28, 968–982 (2006)
16. Desbrun, M., Meyer, M., Schröder, P., Barr, A.H.: Implicit fairing of irregular meshes using diffusion and curvature flow. In: ACM SIGGRAPH Proceedings, vol. 33, pp. 317–324. ACM Press, New York (1999)