Tracking human joint motion for turntable-based static model reconstruction

Neil Birkbeck Dana Cobzas Martin Jagersand
University of Alberta
Edmonton, Alberta, CA
{birkbeck,dana,jag}@cs.ualberta.ca

Abstract

We propose a method that makes standard turntable-based vision acquisition a practical method for recovering models of human geometry. A human subject typically exhibits some unintended joint motion while rotating on a turntable. Ignoring such motion causes shape-from-silhouette to excessively carve the model, resulting in loss of geometry (especially on limbs). We utilize silhouette cues with an initial automatically recovered skinned-model to recover this joint motion, or wobbling. The recovered joint motion gives the calibration of each rigid body of the subject, allowing for temporal fusion of image cues (e.g., silhouettes and texture) used to refine the geometry. Our method gives improved results on real data sets when considering silhouette overlap in novel views. The recovered geometry is useful in vision tasks such as multi-view image-based tracking of humans, where the recent trend of using a priori laser-scanned geometry could be replaced with a more cost effective vision-based geometry.

1. Introduction

For some time now the benefits of turntable-based vision acquisition systems for low cost 3D modeling have been recognized and exploited [5, 14]. Turntables boast the ability to quickly acquire an image stream about an object that can quickly be calibrated and easily be foreground segmented for use in both silhouette and stereo reconstruction. In this work we argue that turntable acquisition is still feasible for human scale geometry, something that has only been exploited in few works [8, 10] and, in the case of some, it was only used for the recovery of appearance [26].

There is no doubt that convenient vision-based acquisition of static human geometry is useful, with example applications ranging from gaming to anthropometric studies. There exist full body laser range finders built exclusively for the task of recovering dense static human geometry, but this hardware is expensive (e.g., Cyberware ™'s Whole Body 3D Scanner $200K+). By comparison, a two camera system, such as ours, costs on the order of hundreds or a few thousand dollars. In terms of applications in vision, a recent trend has seen many of the multi-view human tracking and deformation recovery methods being formulated around an initial laser scanned geometry [13, 12, 3]. In fact many methods in this category go on to recover deformations over time from vision, but have skipped the application of vision in the first step by relying on the scanned geometry [12].

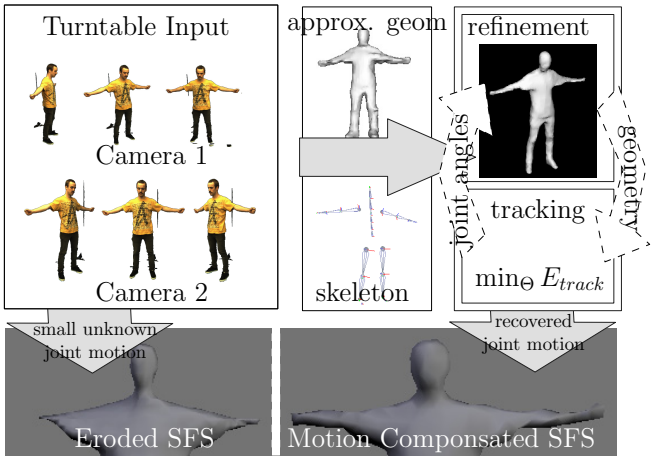


Figure 1. Overview of our solution. SFS with no motion compensation illustrates eroded body. Interleaving motion estimation with SFS gives more accurate results. See website for videos [6].

One vision-based solution commonly used to capture human geometry consists of a large set of fixed, pre-calibrated cameras that observe a moving person [32, 21, 30]. Geometry for each time frame is reconstructed either using the visual hull [31] or multi-view stereo [21] and then related to each other either using differential constrains like scene flow [32], through feature point correspondences [30], or registered with marker-based motion capture data in the coordinate system of the joint [22]. We take a different approach and propose a method that acquires human geometry using a traditional turntable approach that requires only two cameras and reconstructs a model of the rotating human unified in time. Full geometry at each time frame cannot

2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops
978-1-4244-4441-0/09/$25.00 ©2009 IEEE

be recovered due to the low number of cameras (2) in our setup.

One limitation in simply extending rigid-object turntable-based approaches to a human geometry is the fact that a rotating human is not rigid and will undoubtedly move over time while rotating. Such motion causes methods like shape-from-silhouette to excessively carve the object (Fig. 1) and causes misalignment of any recovered appearance. Since the human is a kinematic chain containing a hierarchy of coordinate systems, this problem of registration can not be solved by a simple application of single rigid body calibration. As silhouettes do not require a subject to wear textured clothing, we propose to solve the joint motion calibration problem through an interleaved tracking and model recovery step. Our contributions are two-fold:

- using as few as two cameras, the small kinematic human motion relative to a rotating turntable is tracked by utilizing silhouette consistency while enforcing kinematic constraints.
- recovered joint angles for a kinematic structure are used to re-compute a unified shape-from-silhouette model that is the union of the visual hull for each of the kinematic links.

2. Related Work

In the context of recovering dense static geometric models of humans from vision-based methods, many of the general multi-view stereo methods for static scene reconstruction are relevant(e.g., [23]). For humans specifically, some attention has been directed to convenient acquisition of a deformable human model with limited hardware assumptions (e.g, using as few as two or three images to quickly instantiate a deformable human template model [25]). Similarly, we are also concerned with convenient capture of human geometry under limited hardware assumptions; therefore, we focus on methods for recovering the joint motions of a rotating human so as to utilize all silhouette observations in the geometry reconstruction.

Classical feature-based correspondences or feature tracks, such as those used in standard structure from motion (SFM), offer one route to recover these joint motions. Articulated structure from motion factorization techniques decompose such feature tracks into rigid parts and the corresponding points, but are often based on restricted camera models [29, 34]. On the other hand, given that feature tracks are segmented into corresponding parts, the more recent applications of SFM that refine Euclidean camera parameters based on dense matches could also be used to recover the rigid deformation of individual joints [15]. However, these feature-based methods may still be prone to failure in regions where few features are available, such as the arms which tend to be one of the more problematic regions.

As the geometry of these problematic regions is well classified by silhouettes, it is useful to consider the use of silhouettes for the purpose of calibration. Calibrating the relative position of cameras in a multi-view environment using dynamic silhouettes has been done [24, 7], but in our case we assume the relative poses of cameras are known. Alternatively, similar cues such as epipolar tangents, frontier points, or silhouette consistency have also been used to calibrate the position of cameras viewing a scene under restricted turntable motion [18, 16]. Again, it is not the turntable motion given a rigid geometry we are trying to recover, as we assume that the turntable motion is known; we are instead trying to recover the arbitrary, possibly small, motion of each joint relative to the turntable.

One of the most relevant methods for combining silhouettes over time utilizes both silhouette and image appearance cues. The shape-from-silhouette over time work of Cheung *et al*. [9] recovers the motion of a rigidly moving object observed by multiple image sequences by the use of colored surface points (e.g., frontier points with color values) and a silhouette constraint. The recovered transformation ensures surface points extracted at time t agree with the silhouettes extracted at time $t + 1$ (and vice-versa) and that the observed color values at these points is consistent. This method is also used to fuse images for recovery of human geometry under turntable motion and perform multi-view tracking [9, 10]. Unfortunately, the method relies on the colored surface points which cannot be extracted at each time frame with only two cameras.

Some integration of silhouettes between time steps is accomplished by the spatio-temporal SFS method of Aganj *et al*., but the approach seems to be more useful for interpolating between SFS geometries at independent time steps [1]. The vast assortment of multi-view human tracking methods attempt to solve the problem of recovering the motion of the kinematic links [2, 17, 20, 27]. These approaches rely on a known geometry and often combine multiple cues, such as stereo, flow and silhouettes, in order to recover the joint angles. A practical use of the silhouette is to minimize the exclusive-or between input silhouette and model silhouette [27]; this cost function is closely related to silhouette-based camera calibration [7, 18]. Our approach leverages the same strengths as multi-view tracking, but uses them with the intent of recovering a better static geometry.

Many of the multi-view human tracking methods also try to refine geometries over time [19], deform temporal geometries between time-steps [30], or ensure that the silhouette of the tracked model is consistent with input silhouettes (e.g., [33]). These dynamic geometries are often reconstructed each frame (e.g., often 6-8 or more views are available), meaning they rely mostly on the inter-camera

Figure 2. Capture setup illustrating the typical position of L and R cameras.

correspondence between numerous fixed cameras for reconstructing geometry. In our case, where we have only two widely separated views, it is not practical to reconstruct an independent geometry per frame. Instead, we exploit the intra-camera relationship for geometry reconstruction by recovering and compensating for the restricted human motion that occurs on the turntable.

3. Model, Tracking & Refinement

We assume that the motion of the human rotating on the turntable is governed completely by the joint angles of her kinematic skeleton. The problem is then to recover both the geometry, G, and these joint angles, $\mathbf{\Theta}$, such that the geometry deformed by the joint angles is consistent with the two input image streams.[1]

As input we have two image streams $I_{L,t}, I_{R,t}$ and silhouette images $S_{L,t}, S_{R,t}$ at time $t \in \{1, \ldots, T\}$. The projection matrices $\mathbf{P}_{L,t} = [\mathbf{K}_L|0]\mathbf{E}_t$ and $\mathbf{P}_{R,t} = [\mathbf{K}_R|0]\mathbf{E}_{R,L}\mathbf{E}_t$ are also available. The relative pose between the cameras, $\mathbf{E}_{R,L}$, is fixed, and the motion of the cameras relative to the turntable is characterized only by the known transformation $\mathbf{E}_t$ (recovered using a pattern placed on the turntable - see Section 4).

Based on the observation that multi-view silhouette-driven human tracking is often successful with an approximate geometric model, we propose to solve this problem by interleaving tracking and refinement. The entire procedure is summarized below:

1. *Initialize* a geometry, G, and align a kinematic structure.
   - Obtain a geometry, G, from SFS where silhouettes are grown to ensure the model has all appendages. Align and attach a kinematic structure to G.
2. *Iterate* tracking the joint angles and refining the model:
   - *Joint Tracking:* recover smoothly varying joint angles, $\mathbf{\Theta}$, that deform the geometry, G, to satisfy image cues, and keep the feet stationary.
   - *Refinement:* use $\mathbf{\Theta}$ to register the image observations in the coordinates of each joint, and then compute a piece of the SFS geometry, G_b for each joint, b. Take the union of the piece geometries, $G \leftarrow \bigcup_b G_b$, and attach to kinematic structure.

[1] Our approach easily generalizes to more images.

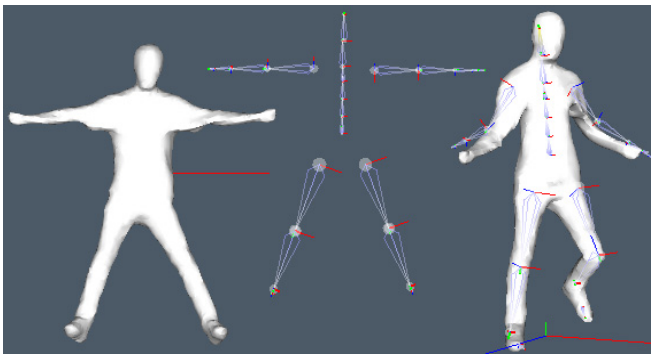


Figure 3. A *rest* geometry (e.g., $\mathbf{v}_k(\mathbf{0})$), the corresponding unposed skeleton, and a geometry posed by the skeleton.

We first define our geometric model and specify how it is attached to a posed kinematic structure. This association of a skeleton with a geometry occurs in both the initialization when the pose is manually specified and after each refinement step when a new model has been computed.

3.1. Model

Our model follows the standard graphics model for human skeletons and consists of two parts: a mesh geometry and a kinematic skeleton (Fig. 3). The geometry, a triangulated mesh, is used to *skin* the skeleton; the motion of the geometry is determined solely by the kinematic model–an assumption we will use during the tracking.

3.1.1 Kinematic Model

The kinematic hierarchy is represented as a tree of transformations. Each node (or bone), b, is positioned in the coordinate system of its parent node, $P(b)$ with a Euclidean transformation $\mathbf{T}_b$ and has a set of rotational freedoms, $\mathbf{R}_b(\boldsymbol{\theta}_b)$. The transformation from a joint to world coordinates is then

$$\mathbf{M}_b([\boldsymbol{\theta}_b, \boldsymbol{\theta}_{anc}]) = \mathbf{M}_{P(b)}(\boldsymbol{\theta}_{anc})\mathbf{T}_b\mathbf{R}_b(\boldsymbol{\theta}_b) \qquad (1)$$

where the parent transformation is influenced by a set of ancestor joint angles, $\boldsymbol{\theta}_{anc}$. The root is an exception to this structure as it has no parent and its freedoms are a full Euclidean transformation. For notational convenience we will

Bone	Parent	Freedoms
Root	nil	R_x,R_y,R_z,Tx,Ty,Tz
Back	Root	$R_x \in [-20, 45]$ $R_y, R_z \in [-30, 30]$
Thorax	Back	$R_x \in [-20, 45]$ $R_y, R_z \in [-30, 30]$
Clavicle	Thorax	$R_y \in [-10, 20]$ $R_z \in [-20, 0]$
Humerus	Clavicle	$R_x \in [-60, 90]$ $R_z \in [-90, 90]$
Radius	Humerus	$R_x \in [0.01, 170]$
Femur	Root	$R_x \in [-160, 20] R_z \in [-70, 60]$
Tibia	Femur	$R_x \in [0.01, 170]$
Foot	Tibia	–

Table 1. A breakdown of the bone names, their freedoms, and their parents for a total of 34 freedoms.

treat $\mathbf{M}_b$ as a function of all joint angles, $\boldsymbol{\theta}$, although freedoms of children have no affect on the parent transformation. Each joint (other than the root) is affected by at most 3 parameters.

We extract a default kinematic structure (e.g., the $\mathbf{T}_b$) complete with joint angle limits from a subject in the CMU motion capture database [11] (see Table 1 for a listing of the degrees of freedom and kinematic structure). The lengths of the kinematic links are optimized to align the structure to the human subject. The registration is done by locating approximate joint positions in the initial geometry (detected through assumptions on body size) and optimizing the kinematic parameters and scales such that these joint position constraints are met using inverse kinematics.

3.1.2 Kinematic & Geometric Surface Coupling

The geometric model is attached to the skeletal model in a default or rest pose using linear blend skinning (LBS). In LBS a vertex deforms through a linear combination of a set of joints it has been associated with

$$\mathbf{v}_k(\boldsymbol{\theta}) = \sum_{b \in B(k)} w_{k,b} \mathbf{H}_b(\boldsymbol{\theta}) \hat{\mathbf{v}}_k \tag{2}$$

where $\hat{\mathbf{v}}_k$ is the vertex in rest position, $B(k)$ is the set of links to which vertex k is attached, and $w_{k,b}$ is the skinning weight association of vertex k with bone b. The transformation matrix $\mathbf{H}_b(\boldsymbol{\theta}) = \mathbf{M}_b(\boldsymbol{\theta})\hat{\mathbf{M}}_b^{-1}$, where $\hat{\mathbf{M}}_b = \mathbf{M}_b(\mathbf{0})$ is the rest transformation matrix for bone b and $\mathbf{M}_b(\boldsymbol{\theta})$ is the animated pose of bone b. Given a posed kinematic skeleton (e.g., as a result of tracking or manual initialization in the first frame) we extract the vertex skinning weights automatically using the heat diffusion process of Baran and Popovic [4].

In our case the geometry is given in the context of a posed kinematic structure (e.g., the vertices $\mathbf{v}_k(\boldsymbol{\theta}_{pose})$ are already deformed with joint parameters $\boldsymbol{\theta}_{pose}$). The weights are assigned to the geometry in this posed frame, so the rest geometry must be obtained through the inverse of the transformation in Eq. 2, i.e., $(\sum_{b \in B(k)} w_{k,b} \mathbf{T}_b(\boldsymbol{\theta}_{pose}))^{-1}$.

3.2. Joint Tracking

Given a skinned geometric mesh parametrized only with joint angles, e.g., $G(\boldsymbol{\theta}_t)$, we treat the recovery of all the joint angles $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, ..., \boldsymbol{\theta}_T\}$ as the optimization of a cost function that is a linear combination of several terms:

$$\min_{\boldsymbol{\Theta}} E = E_{sil} + \alpha_{kin} E_{kin} + \alpha_{smooth} E_{smooth} \tag{3}$$

The silhouette term, E_{sil}, is based on an energy used in motion tracking [28] and measures agreement of the model with the input silhouettes. It is computed as a sum of XOR's over all input images:

$$E_{sil} = \sum_{t=1}^{T} \sum_{i \in \{L,R\}} \sum_{\mathbf{x}} \frac{S_{i,t}(\mathbf{x}) \otimes P_{i,t}(G(\boldsymbol{\theta}_t), \mathbf{x})}{width(I_{i,t}) * height(I_{i,t})} \tag{4}$$

where the shorthand $P_{i,t}(G(\boldsymbol{\theta}_t))$ denotes the projected silhouette of the geometry by $P_{i,t}$.

The smoothness term prefers no joint motion from one frame to the next:

$$E_{smooth} = \sum_{t=2}^{T} \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}\|^2 \tag{5}$$

Finally, due to the assumption of our input being a human rotating on a platform, the kinematic term, E_{kin}, enforces the constraint that the feet stay on the ground. As our kinematic skeleton is naturally rooted at the tailbone, we use the E_{kin} term to limit deviations of the feet position, $\mathbf{X}_{foot}$, in frames $t > 1$ from their position at time $t = 1$:

$$E_{kin} = \sum_{t=2}^{T} \sum_{foot} \|(\mathbf{X}_{foot}(\boldsymbol{\theta}_t) - \mathbf{X}_{foot}(\boldsymbol{\theta}_1))\|^2 \tag{6}$$

Due to the discrete nature of the silhouette XOR term, we use Powell's method to optimize the cost function [28]. The parameters are initialized with the pose from the first frame and all the parameters are optimized simultaneously.

3.3. Refinement

Tracking gives an updated estimate of coordinate transforms of each link at each time of the image sequence. These transformations are used to integrate all the silhouette observations and create a new, refined geometry. The geometry is first computed as a number of partially overlapping pieces (one for each bone, b) that are subsequently merged (e.g., Fig. 4). Each bone is assumed to be a rigid body, and the resulting joint to world transforms (from $\boldsymbol{\Theta}$) are concatenated with the world to camera transforms to get all observations in the coordinate frame of a bone. The camera matrices relative to the bone are then

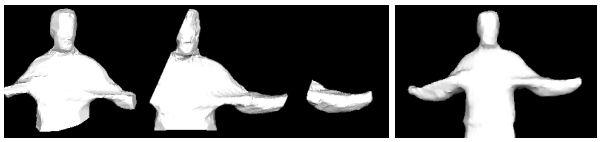


Figure 4. Overlapping pieces of geometry computed from SFS (corresponding to the head, shoulder, and elbow) are merged into a single manifold geometry.

$$\mathbf{P}^b_{i,t} = \mathbf{P}_{i,t}\mathbf{M}_b(\boldsymbol{\theta}_t) \tag{7}$$

The piece of geometry associated with the bone, $G_b = SFS(\mathbf{P}^b_{i,t}, S_{i,t})$, is computed from all observations (i.e., all time frames and both cameras) using SFS with the marching intersections (MI) data structure. The SFS computation for bone b is bounded by the axis-aligned bounding box (AABB) of the set of vertices from the previous geometry whose skinning weights to link b are above a threshold:

$$bounds^b = AABB(\{\hat{\mathbf{M}}_b^{-1}\hat{\mathbf{v}}_k : w_{k,b} \geq \tau\})$$

The threshold is low ($\tau = 0.05$) to allow for errors in the previous, approximate geometry. This piece of the mesh, G_b, is then transformed to the turntable coordinate system at time $t = 1$ using $\mathbf{M}_b(\boldsymbol{\theta}_1)$.

We now have pieces of geometries for each part that are originally disconnected and represented in the turntable coordinate system. A final manifold geometry is then obtained as the volumetric union of the pieces:

$$G \leftarrow \bigcup_b G_b$$

Subsequently, this posed geometry is attached to the skeleton by finding new skinning weights, $\{w_{k,b}\}$, and the corresponding rest vertices, $\hat{\mathbf{v}}_k$ (as in Section 3.1). This refined mesh is then used in the next iteration of the tracking (starting with the estimated motion parameters from the previous iteration).

4. Experiments

We have implemented our tracking and refinement algorithm in C++ using OpenGL to render the model for computing the XOR score. In our tracking implementation and experiments, we perform a few (3-4) iterations of the tracking/refinement and use the the following weights to combine the terms in Eq. 3: $\alpha_{kin} \approx 0.4$ and $\alpha_{smooth} \approx 0.25$. An iteration of the tracking takes roughly 16 minutes; the refinement and merging takes about 2 minutes.[2]

To evaluate the accuracy of our approach we have used synthetically generated sequences with a ground truth reference model. We also demonstrate the strengths of our refinement with several real-world sequences. Our approach gives improved geometries on these sequences compared to simply ignoring the human motion and using shape-from-silhouette.

[2]These timings are for an unoptimized implementation running on a single core of an Intel®Quad Core™2.4GHz machine.

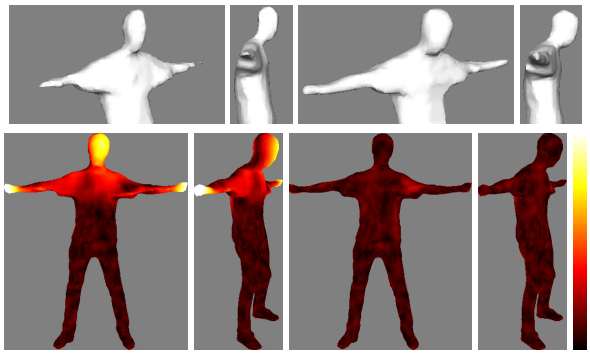
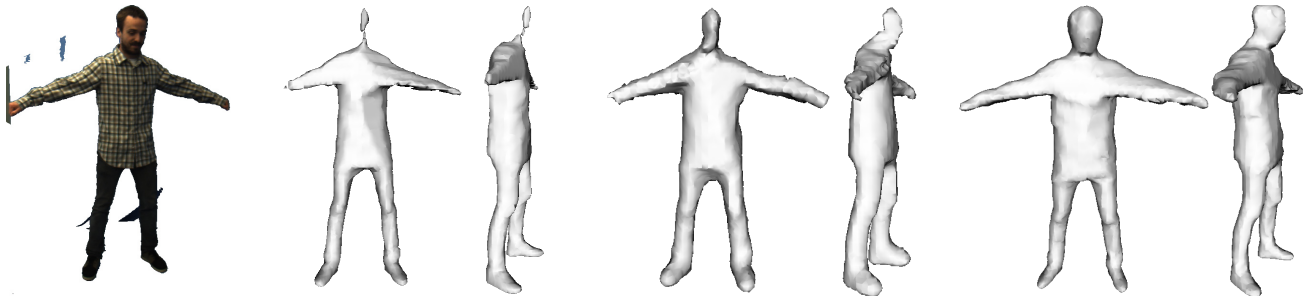


Figure 5. Synthetic data set results. Top: recovered model without (left) and with motion compensation (right). Bottom: Ground-truth synthetic model colored with the distance to the recovered model without (left) and after refinement (right). Colormap: black=0cm, white=10cm.

First, we demonstrate the accuracy of the method using a synthetic image sequence generated under similar conditions as our real turntable setup. Thirty images of a skinned figure rotating on a platform and having small motions in the back, neck, and arms were rendered from two viewpoints (a top-down and side-view). Figure 5 shows views of the recovered model without registration and with our registration. The unregistered model is missing portions of the arm, and the head is carved due to motion in the back. Coloring the ground-truth model with the distance to the recovered model illustrates how our method improves the reconstruction in these regions (bottom right of Fig. 5, notice small error over the surface).

In our real experiments we have captured several data sets of human subjects rotating on a turntable. All of the data sets contain three video streams; two of the streams were used for reconstruction and the third was used for comparison. The image sequences consist of 30 images (with the exception of the *Green Sweater* data set sequence which contains 22 images). The images are 800x600 color images captured from Point Grey grasshopper cameras. The relative positions of the cameras were calibrated in advance, and a calibration pattern on the turntable was used to recover the relative pose of the turntable with respect to the first camera through the sequence.

In each case we bootstrapped our algorithm with a geometry that was obtained from all of the images in the data sets using SFS (without correcting for any human motion); the silhouette boundaries were extended (by roughly 5-6 pixels) to ensure that the extremities were present in the initial geometry. The *Plaid* data set contains significant motion causing the head and arms to be almost entirely eroded if the human motion is ignored (Fig. 6). From this example,



Input image | Degraded SFS reconstruction | Initial geometry | Our reconstruction

Figure 6. An input image, the severely degraded SFS reconstruction, the initial geometry, and our reconstruction for the *Plaid* data set .

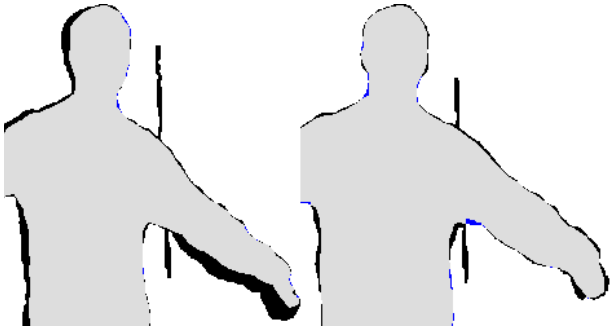


Figure 7. The *Red Sweater* data set had little motion, but SFS geometry (left) disagrees with the input silhouette (black indicates regions of input not covered by geometry). Motion recovered silhouette matches better (right).

	Yel-Shirt	Red-Swtr	Green	Girl	Plaid
SFS	0.86	0.84	0.81	0.85	0.71
Ref	0.91	0.89	0.88	0.92	0.86

Table 2. Average Jaccard score in the extra view between silhouette and the uncompensated SFS model and the motion compensated *ref*ined model.

we see that our method is capable of recovering an accurate geometry, even when the initial estimate is poor.

The final geometry and the SFS geometry without motion compensation for the remaining data sets are presented in Fig. 9. In all cases the original SFS is again eroded, with parts of the arms missing and the bodies shaved too far in general. The motion compensation successfully recovers these parts of the geometry. Of these data sets the *Red Sweater* data set contained little motion, but the improvement is still evident when inspecting the silhouette agreement between the recovered models and the input images (Fig. 7).

As we do not have access to ground truth geometry for the real data, we use the extra video streams that were not used during reconstruction to perform a quantitative comparison. The average Jaccard score[3] between input

[3]The Jaccard score between two sets A and B is $J = \frac{|A \bigcap B|}{|A \bigcup B|}$

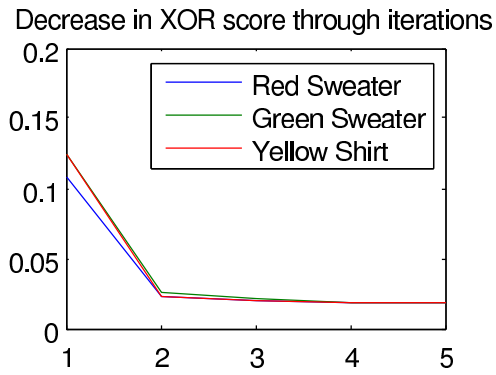


Figure 8. Decreasing cost after interleaved tracking/refinement.

silhouette and rendered silhouette of our motion compensated model is always higher (better) than that of the non-compensated model in this video stream (Table 2). This data suggests that the motion compensated model is a better match to the true visual hull.

Refinement of the geometry only affects the E_{sil} term of the energy because it does not change joint angles. One may question the validity of using only SFS in the refinement, as SFS does not directly minimize the XOR score, meaning that on successive tracking/refinement iterations the energy could in fact go up. Although this is possible, in practice we have found that the E_{sil} term often does go down, and only in the latter iterations does the cost sometimes increase slightly. Figure 8 shows the XOR score for several data sets after subsequent refinement/tracking iterations.

5. Conclusion

We have presented an iterative method that uses as few as two camera streams (with a wide baseline) to recover small human motion using silhouette cues. The recovered motion allows the registration of silhouettes to improve the geometry using SFS.

One limitation of our model is that it does not incorporate

Figure 9. Sample input image, reconstruction without motion compensation (e.g., SFS), the refined model, and a textured model (single average texture).

a texture consistency term, meaning the recovered motion may not respect texture motion cues. We tried adding a texture preserving term in the joint angle refinement (Eq. 3) but the initial geometry is too approximate. We are currently investigating the use of optic-flow to ensure texture consistency without requiring an accurate model during the motion recovery. Incorporating feature-based constraints when available (e.g., as in typical structure and motion) should also help ensure consistent texture motion.

Another limitation is that our method needs to be bootstrapped with an initial geometry. We currently based this geometry on an enveloping SFS geometry that is obtained by growing the silhouette boundaries. Any appendage missing in the initial geometry will likely remain missing throughout the refinement. As such, we would like to explore the sensitivity of our solution to this initial geometry.

We would also like to explore using our recovered model in the context of tracking. Another possible future direction is to see if refining the model in this manner can be done in an on-line manner with general motion.

References

[1] E. Aganj, J.-P. Pons, F. Segonne, and R. Keriven. Spatio-temporal shape from silhouette using four-dimensional delaunay meshing. In *ICCV*, pages 1–8, Oct. 2007.

[2] A. O. Balan, L. Sigal, and M. J. Black. A quantitative evaluation of video-based 3d person tracking. In *ICCCN '05: Proc. of the 14th Int. Conf. on Computer Communications and Networks*, pages 349–356, 2005.

[3] L. Ballan and G. M. Cortelazzo. Marker-less motion capture of skinned models in a four camera set-up using optical flow and silhouettes. In *3DPVT*, June 2008.

[4] I. Baran and J. Popović. Automatic rigging and animation of 3d characters. *ACM Trans. Graph.*, 26(3):72, 2007.

[5] A. Baumberg, A. Lyons, and R. Taylor. 3D S.O.M. - a commercial software solution to 3d scanning. In *Proc. of Vision, Video, and Graphics (VVG'03)*, pages 41–48, July 2003.

[6] N. Birkbeck, M. Jagersand, and D. Cobzas. Project web page and videos. http://www.cs.ualberta.ca/˜vis/ibmr/tracking-turntable.

[7] E. Boyer. On using silhouettes for camera calibration. In *ACCV*, pages 1–10, 2006.

[8] K. M. Cheung, S. Baker, J. K. Hodgins, and T. Kanade. Markerless human motion transfer. In *3DPVT*, pages 373 – 378, September 2004.

[9] K. M. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette across time part i: Theory and algorithms. *IJCV*, 62(3):221 – 247, May 2005.

[10] K. M. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette across time: Part ii: Applications to human modeling and markerless motion tracking. *IJCV*, 63(3):225 – 245, August 2005.

[11] CMU graphics lab motion capture database. http://mocap.cs.cmu.edu/.

[12] E. de Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun. Performance capture from sparse multi-view video. In *SIGGRAPH*, pages 1–10, New York, NY, USA, 2008. ACM.

[13] E. de Aguiar, C. Theobalt, C. Stoll, and H.-P. Seidel. Marker-less 3d feature tracking for mesh-based human motion capture. In *ICCV Workshop on Human Motion*, pages 1–15. Springer, 2007.

[14] C. H. Esteban and F. Schmitt. Silhouette and stereo fusion for 3d object modeling. In *3DIM*, pages 46–53, October 2003.

[15] Y. Furukawa and J. Ponce. Accurate camera calibration from multi-view stereo and bundle adjustment. *IJCV*, 84(3):257–268, 2009.

[16] Y. Furukawa, A. Sethi, J. Ponce, and D. Kriegman. Structure and motion from images of smooth textureless objects. In *ECCV*, pages 287–298, 2004.

[17] D. M. Gavrila and L. S. Davis. 3-d model-based tracking of humans in action: a multi-view approach. In *CVPR '96*, page 73, 1996.

[18] C. Hernandez, F. Schmitt, and R. Cipolla. Silhouette coherence for camera calibration under circular motion. *PAMI*, 29(2):343–349, February 2007.

[19] A. Hilton and J. Starck. Multiple view reconstruction of people. In *3DPVT*, pages 357–364, 2004.

[20] R. Kehl, M. Bray, and L. V. Gool. Full body tracking from multiple views using stochastic sampling. In *CVPR '05*, volume 2, pages 129–136, 2005.

[21] J.-P. Pons, R. Keriven, and O. Faugeras. Modelling dynamic scenes by registering multi-view image sequences. In *CVPR '05*, pages 822–827, 2005.

[22] P. Sand, L. McMillan, and J. Popović. Continuous capture of skin deformation. *ACM Trans. Graph.*, 22(3):578–586, 2003.

[23] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR '06*, pages 519–528, 2006.

[24] S. N. Sinha, M. Pollefeys, and L. McMillan. Camera network calibration from dynamic silhouettes. In *CVPR*, pages 195–202, 2004.

[25] J. Starck, A. Hilton, and J. Illingworth. Human shape estimation in a multi-camera studio. In *British Machine Vision Conference (BMVC)*, pages 573–582, 2001.

[26] C. Theobalt, N. Ahmed, H. Lensch, M. Magnor, and H.-P. Seidel. Seeing people in different light-joint shape, motion, and reflectance capture. *IEEE Transactions on Visualization and Computer Graphics*, 13(4):663–674, 2007.

[27] C. Theobalt, J. Carranza, M. A. Magnor, and H.-P. Seidel. Combining 3d flow fields with silhouette-based human motion capture for immersive video. *Graphical Models*, 66(6):333–351, 2004.

[28] C. Theobalt, E. de Aguiar, M. Magnor, and H.-P. Seidel. Reconstructing human shape, motion and appearance from multi-view video. In *Three-Dimensional Television: Capture, Transmission, and Display*, pages 29-58. Springer, Heidelberg, Germany, November 2007.

[29] P. Tresadern and I. Reid. Articulated structure from motion by factorization. In *CVPR*, pages 1110–1115, 2005.

[30] K. Varanasi, A. Zaharescu, E. Boyer, and R. P. Horaud. Temporal surface tracking using mesh evolution. In *ECCV*, pages 30–43, October 2008.

[31] S. Vedula, S. Baker, and T. Kanade. Image-based spatio-temporal modeling and view interpolation of dynamic events. *ACM Transactions on Graphics*, 24(2):240 – 261, April 2005.

[32] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. *PAMI*, 27(3):475 – 480, March 2005.

[33] D. Vlasic, I. Baran, W. Matusik, and J. Popović. Articulated mesh animation from multi-view silhouettes. *ACM Trans. Graph.*, 27(3):1–9, 2008.

[34] J. Yan and M. Pollefeys. Articulated motion segmentation using ransac with priors. In *ICCV Workshop on Dynamical Vision*, pages 75–85, 2005.