# Continuous Capture of Skin Deformation

Peter Sand* Leonard McMillan† Jovan Popović

Laboratory for Computer Science
Massachusetts Institute of Technology
†University of North Carolina, Chapel Hill

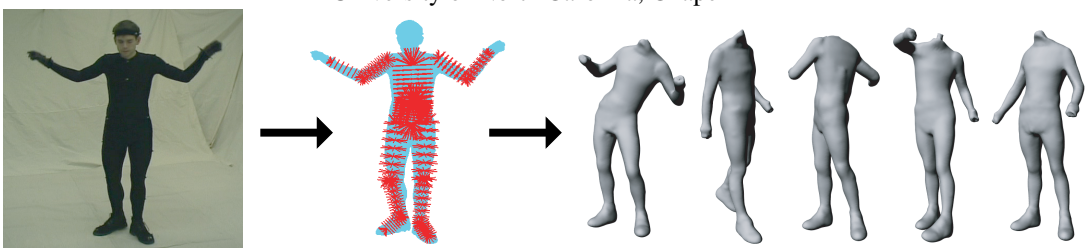

Figure 1: We extract silhouettes from video sequences to build a deformable skin model that can be animated with new motion.

## Abstract

We describe a method for the acquisition of deformable human geometry from silhouettes. Our technique uses a commercial tracking system to determine the motion of the skeleton, then estimates geometry for each bone using constraints provided by the silhouettes from one or more cameras. These silhouettes do not give a complete characterization of the geometry for a particular point in time, but when the subject moves, many observations of the same local geometries allow the construction of a complete model. Our reconstruction algorithm provides a simple mechanism for solving the problems of view aggregation, occlusion handling, hole filling, noise removal, and deformation modeling. The resulting model is parameterized to synthesize geometry for new poses of the skeleton. We demonstrate this capability by rendering the geometry for motion sequences that were not included in the original datasets.

**CR Categories:** I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation; I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Shape

**Keywords:** motion capture, skin modeling, human animation

## 1 Introduction

A digital replica of a moving human body has applications in video games, teleconferencing, automated news shows, and filmmaking. For example, the physical appearance of a celebrity actor could be recorded and later animated with acrobatic motions controlled by an animator or performed by a stunt double in a motion-capture suit. In current filmmaking, this application requires extensive manual labor to position and adjust skin around each bone and muscle. In some cases, months are spent matching a virtual character to an existing actor [Stokdyk et al. 2002].

Our goal is to build a skin model that replicates the skin deformations of a particular person. The technique described in this paper builds this model automatically from video of the subject and motion data that describes how the subject's skeleton moves throughout the video recording. To build the model from this data, we exploit the idea that video of a moving person provides many observations of the same surface. A single set of silhouettes (even from several viewpoints) provides a highly incomplete characterization of the geometry. By having the subject move through many different poses, local configurations of the body parts are repeated, allowing the construction of a complete model.

Our main contribution is a method of gathering silhouette observations such that a simple reconstruction algorithm can create a complete deformable model, parameterized in a way that is useful for animation. We do not contribute new techniques in the areas of skin representation and skin interpolation, but in ways of quickly acquiring skin data. By using the right combination of prior tools, we substantially simplify the problem of generating a 3D model from moving silhouettes.

Our skin model, described in Section 3, represents a complex articulated figure using a collection of elongated deformable primitives. Our acquisition algorithm, described in Section 4, uses the silhouettes to provide constraints on the possible body geometry. The reconstruction algorithm, described in Section 5, uses these constraints to find a model of the skin deformations, parameterized with the motion of the skeleton. This parameterization allows animation of the skin with new motion data.

## 2 Related Work

The most general 3D reconstruction systems attempt to build a model of the scene at each successive time frame, allowing the acquisition of moving objects. These systems use vision methods such as binocular stereo [Nebel et al. 2001] and voxel coloring [Vedula et al. 2002]. For certain kinds of scenes, the geometry can be reasonably represented using a *visual hull*: the space carved about by silhouettes from a set of viewpoints [Matusik et al. 2000; Würmlin et al. 2002].

Some of these methods make frame-to-frame comparisons of the geometry [Würmlin et al. 2002; Vedula et al. 2002], but they do not accumulate observations to improve the geometry. The strength of gathering information from temporally distinct views is illustrated in recent work in real-time model acquisition, in which a rigid ob-

*77 Massachusetts Avenue, Cambridge, MA 02139

ject can be moved while it's digitized [Rusinkiewicz et al. 2002]. Real-time feedback and freedom of movement allow the operator to fill in holes and build a complete model. While this technique allows accurate and complete models to be generated from multiple observations of an object, it is limited to rigid objects.

Factorization techniques, in contrast, can build models of deforming objects. Surface deformations are represented as a linear combination of prototype shapes, found via matrix factorization [Bregler et al. 2000; Brand 2001; Torresani and Bregler 2002]. A matrix of image observations is factored into a matrix of pose vectors, which defines the object's motion, a matrix of geometry vectors, which defines the basis shapes, and a vector of basis weights, which defines the deformation of the object. While these factorization methods are quite powerful, they have not been applied to capture deformations of an entire human body.

To overcome the difficulties of general reconstruction, a model of an object class can be fit to observations of a particular object. For example, numerous methods reconstruct and reanimate the human face [Guenter et al. 1998; Cootes et al. 1998; Blanz and Vetter 1999]. These techniques are successful at modeling a range of human faces, but would be difficult to extend to capturing an entire human body, due to large-scale occlusions and deformations. Nonetheless, they would be an excellent complement to our current system, which cannot capture facial expressions.

Several systems reconstruct human bodies by fitting prior model to observations of a moving person. For tracking applications, simple models consisting of ellipsoids can be fit using silhouettes [Mikić et al. n. d.]. Plänkers and Fua [2001] use an elaborate anatomical model, in which the skin surface is defined as the level set of Gaussians rigidly attached to a skeleton. The dimensions of these Gaussians are optimized according to observations from the silhouettes and stereo depth estimates. Kakadiaris and Metaxas [1993] use a pre-defined protocol of motions to extract 2D contours of body parts. These 2D contours can deform for different poses and be interpolated to obtain an approximation of the 3D geometry.

Allen and colleagues [2002] acquire multiple poses of the human body using a 3D laser scanner to obtain a high level of detail and accuracy. Each of the reconstructed poses is related to a skeletal configuration through the use of dots placed on the skin. New poses are then synthesized by interpolating nearby key poses. This method has successfully created animations of the upper body, but it requires a substantial amount of time and effort in order to acquire hundreds of 3D range scans. In contrast, our system acquires the deformation automatically as the subject moves freely through various poses, building a complete model using only a few minutes of motion. However, because our models are built from video, rather than laser scanning, we do not obtain the same level of detail.

Like many of these acquisitions systems, our work uses interpolation to combine models of different poses. These interpolation techniques (such as [Lewis et al. 2000; Sloan et al. 2001; Wang and Phillips 2002]) vary in the interpolation mechanisms, the particular quantities being interpolated, and the way in which the skeleton drives the interpolation. Several of these papers give theoretical results on the relative strengths and limitations of different representations of geometry and deformation—a subject not addressed in this paper. Instead, we focus on how to position and reconstruct prototype shapes in a fast and automatic manner.

# 3 Skin Model

Our skin model simplifies the complex process of acquiring geometry of a moving human body. We represent the skin surface using points along needles that are rigidly attached to a skeleton. This model describes complex areas near joins by combining nearby samples. Deformation is parameterized with a configuration space for each bone.

## 3.1 Deformable Primitives

We represent the geometry of an articulated human figure using a collection of elongated deformable primitives. Each deformable primitive consists of a rigid axis, which usually corresponds to a bone in the skeleton, and a set of needles, which are rigidly attached to the axis. Each needle originates at a point along the axis and extends outward in a fixed direction with respect to the axis.

Our deformable primitive is equivalent to a discrete sampling of a pose-varying generalized cylinder [Nevatia and Binford 1977]. Smooth surfaces can be reconstructed from the point samples by fitting an arbitrary function to the needle endpoints. Our implementation triangulates the needles to create a piece-wise linear surface model. Triangulation is simplified by positioning the needles in rings around the axis, as shown in Figure 2. We can vary the sampling density by changing the number of needles in the radial and axial directions. Although we use regular sampling for rendering purposes, our acquisition and estimation algorithms do not require any particular needle arrangement. Indeed, irregular sampling density may provide a more economical representation of the human form (e.g. using additional samples near joints).

As an alternative to our needle model, a surface could be represented by oriented particles that model deformation by moving in three dimensions [Szeliski and Tonnesen 1992]. This would complicate our acquisition and estimation algorithms because the position of each particle would be a function of three parameters instead of one. By using a scalar value for each needle, we can infer how a particular observation changes with the motion of the skeleton.

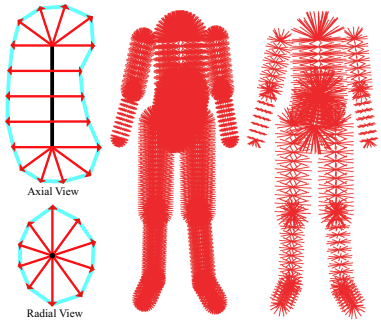


Figure 2: Deformable primitives describe the human body with variable-length needles (red) attached to a fixed axis (black). The left skeleton uses needle counts given in Table 1. The skeleton on the right uses one quarter as many needles (half as many radially and half as many axially). In both cases, the needles are shown at half the maximum length indicated in the table.

## 3.2 Representation of Junctions

Junctions between limbs are traditionally difficult to model: the combination of linked bone structures, muscles, and tendons create complex surface behaviors. We represent a junction between two deformable primitives by taking the union of the their volumes, as illustrated in Figure 3. These interpenetrating objects work together to describe the deformation of the skin near a joint. We do not use explicit constraints to ensure continuity between the surfaces from different skin models. The continuity arises naturally because each deformable primitive deforms appropriately.

Although this representation is well-suited to our acquisition process, it is more expensive to render. Because each primitive renders as a separate mesh, rendering the entire body requires merging all the meshes. Furthermore, the nodes on the surface do not move like the real skin, which complicates texturing. Possible solutions to these problems are discussed in Section 7.

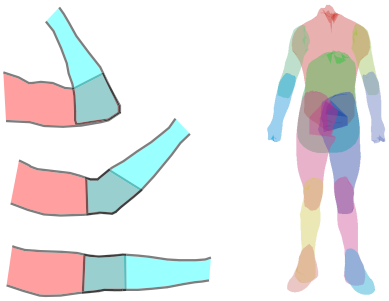

Figure 3: We represent an elbow using overlapping deformable primitives for the upper arm and forearm. Both primitives deform as the elbow bends, maintaining continuity in the junction. The image on the right shows how the segments overlap in a complete body.

### 3.3 Parameterization of Skin Deformation

The length of each needle can depend on parameters that influence skin deformation. For example, we may wish that the geometry of the upper arm varies as a function of the angle of the elbow and as a function of the angle of the shoulder. We could also make the geometry vary as a function of muscle force (for muscular people) and the direction of gravity (for heavy people).

The results in this paper demonstrate deformations caused by the motion of a skeleton. Each deformable primitive has a limited configuration space that is a subset of the configuration of the entire body. For example, the deformation of the left arm does not depend on the configuration of the right knee. We make this assumption to cope with the combinatorial complexity of the human pose space. By decoupling remote parts of the body, we can capture a wide range of deformations in a short amount of time.

To avoid the issues of joint-angle representation, we use marker coordinates to determine the configuration space. For example, the configuration of the right thigh depends on markers attached to the hip and the right calf, where the positions are expressed with respect to the local coordinate frame of the thigh bone. Table 1 summarizes the configuration parameters for each deformable primitive.

## 4 Acquisition of Skin Observations

Our system extracts surface observations by combining information from two separate sources: a commercial motion-capture system and a set of standard video cameras. The motion-capture system tracks reflective markers, which are used to compute the motion of each bone. Because the motion-capture cameras in our system use infrared strobes and filters, they are not suitable for silhouette extraction. Instead, the silhouettes are extracted from one or more video cameras placed around the motion-capture workspace. Our system does not require any special camera arrangement; we position the cameras such that the subject is within the view throughout the motion, as shown in Figure 4.

Our system first calibrates and synchronizes the video and the motion data. It then combines these two data sources to measure the intersection of needles and silhouettes. The reconstruction algorithm described in Section 5 subsequently processes the resulting measurements to parameterize the motion of the skin surface.

### 4.1 Calibration

Camera calibration relates the motion data (the location of markers and bones in a single 3D coordinate system) to the image coordinates of each camera. We perform calibration using a simple device, shown in Figure 5, which allows us to match an image point to an identical point in the motion data. The calibration process starts with synchronization of video and motion data. We move the calibration device up and down in a plane roughly parallel to the image plane of a particular camera and correlate the vertical image coordinate with the vertical world coordinate. Once synchronized, we resample the motion-capture data to obtain a sequence of matching image $p_i \in \Re^2$ and world $w_i \in \Re^3$ points. The mapping between these points depends on camera position, camera orientation, focal length, aspect ratio, and radial distortion. Our system estimates these parameters by minimizing Euclidean error in image space:

$$\min_{q,f,a,c,r} \sum_i ||p_i - D_{c,r}(P_{q,f,a} w_i)||$$

The matrix $P_{q,f,a}$ describes a perspective projection (parameterized by camera pose $q$, focal length $f$, and aspect ratio $a$) and the function $D_{c,r}()$ describes first-order radial distortion (with center of distortion $c$ and a distortion coefficient $r$). For simplicity we simultaneously optimize the parameters using the downhill simplex method [Nelder and Mead 1965]. The method quickly converges to a solution that obtains a sub-pixel RMS error over several hundred $(w_i, p_i)$ input points.

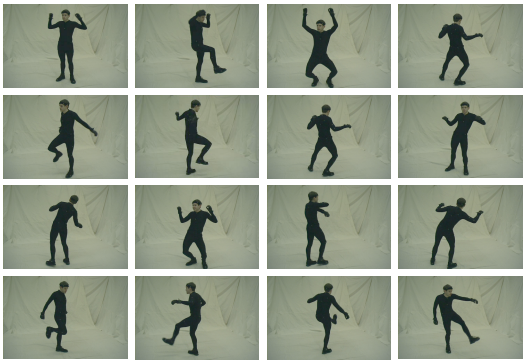

Figure 4: The input video includes images of the subject in a wide variety of poses. As discussed in Section 6.6, the quality of the final model depends on the range of motion in the training sequences.

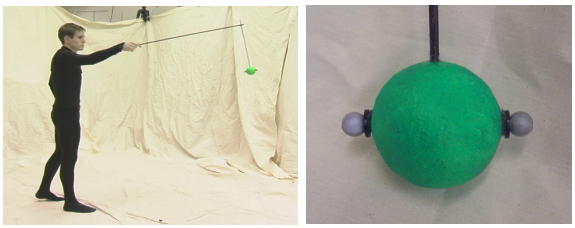

Figure 5: Our calibration device consists of a green sphere with two motion-capture markers. We find the center of the sphere in image coordinates by detecting green pixels. We find the center of the sphere in world coordinates by taking the midpoint of the two marker positions. This gives a single correspondence that be varies through time to obtain a number of spatial correspondences for calibration.

### 4.2 Silhouette Extraction

Our system uses standard background subtraction to obtain silhouettes from video data. For each pixel, background subtraction finds the difference between the current frame and an empty background

| Bone Name | Configuration Depends On | Dim. of Config. Space | Radial Needles | Axial Needles | Maximum Needle Length |
|---|---|---|---|---|---|
| Torso | Upper Arms, Hips | 9 | 30 | 30 | 30cm |
| Hips | Torso, Thighs | 9 | 30 | 30 | 30cm |
| Right Upper Arm | Torso, Right Forearm | 6 | 20 | 20 | 15cm |
| Left Upper Arm | Torso, Left Forearm | 6 | 20 | 20 | 15cm |
| Right Forearm | Right Upper Arm | 3 | 20 | 20 | 10cm |
| Left Forearm | Left Upper Arm | 3 | 20 | 20 | 10cm |
| Right Thigh | Hips, Right Calf | 6 | 20 | 30 | 20cm |
| Left Thigh | Hips, Left Calf | 6 | 20 | 30 | 20cm |
| Right Calf | Right Thigh, Right Foot | 6 | 20 | 30 | 15cm |
| Left Calf | Left Thigh, Left Foot | 6 | 20 | 30 | 15cm |
| Right Foot | Right Calf | 3 | 20 | 20 | 15cm |
| Left Foot | Left Calf | 3 | 20 | 20 | 15cm |

Table 1: Each deformable primitive is described with a configuration space (Section 3.3), needle counts (Section 3.1), and a maximum needle length (Section 4.3).

frame and labels pixels with a high difference as part of the foreground. Our system uses a large subtraction threshold to overcome shadows and video compression artifacts. The threshold near the head is smaller to account for the closeness of skin color to the background (where the head position is determined directly from the motion capture data). These thresholds are sufficiently robust that the same values can be used across multiple cameras and across multiple sequences.

We use the silhouettes and camera calibration to synchronize the video data and motion data for a human subject. Our system uses a simplex optimizer (the same one used for camera calibration) to minimize an objective function that measures the image-space distance from projected arm and leg markers to the silhouettes over a number of video frames.

### 4.3 Accumulation of Needle Observations

After calibrating and synchronizing the video and motion data, the system projects each needle into each video frame to compute the needle length from its intersection with the silhouette. Starting at the origin of the needle, we traverse the image outward until the needle projection leaves the silhouette, as illustrated in Figure 6. If the traversal extends beyond a prescribed maximum length, the measurement is discarded. Thus the system discards observations for needles that are nearly perpendicular to the image plane or that extend into distant parts of the body. Our maximum length values (specified in Table 1) are relatively large; the same values can be used for a wide variety of people.

For each needle length observation, we also record the bone's current position in configuration space, as described in Section 3.3. By annotating each observation with the conditions under which the observation was made (a location in configuration space), we can estimate skin deformation, as described in the next section.

## 5 Skin Reconstruction

The acquisition process accumulates observations of needle lengths. Subsequent reconstruction will refer only to these observations, not the original video and motion data. Because the needle observations do not give a complete description of the geometry at any time instant, reconstruction integrates observations over time to obtain a complete model. Skin reconstruction determines which observations are valid measurements of the true needle length and which are invalid due to occlusion.

As shown in Figure 6, multiple types of invalid observations occur. In each case, the measurements overestimate the true geometry. Thus, by taking the minimum of these observations, we find the least upper bound on the true geometry. Equivalently, we seek the maximal geometry that is consistent with the observations.

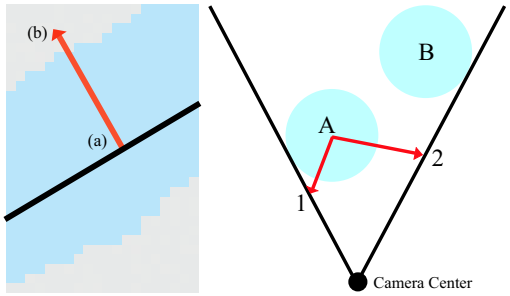


Figure 6: **Left:** To obtain a needle length observation, we project the needle into the image plane. We traverse the image along the needle (from (a) towards (b)), to find the image space distance from the bone to the edge of the silhouette (in blue). This length is converted to a world space distance and later used to estimate deformation. **Right:** The black lines indicate the silhouette observed for the pair of objects A and B. The length of needle 1 is overestimated because the background is occluded by object A while the length of needle 2 is overestimated because the background is occluded by object B. In general, the silhouette provides an upper bound on the geometry.

Because the silhouettes provide an upper bound on the geometry, the needle data effectively has a one-sided error. This contrasts the two-sided errors that occur with other reconstruction methods (e.g. stereo and factorization). This is a key element of our approach: a one-sided error can be removed more easily than a two-sided error.

The reconstruction algorithms uses the following design goals to compute the maximal consistent geometry:

**occlusion handling.** Invalidate measurements that are incorrect because of visibility.

**time aggregation.** Combine multiple observations to complete partially observed shapes.

**hole filling.** Borrow an observation from a nearby configuration if there are no valid observations for a given configuration.

**noise filtering.** Remove outliers caused by errors in silhouette extraction and motion capture.

**deformation modeling.** Obtain geometry estimates that vary smoothly with configuration.

## 5.1 Deformation Model

The skin deforms with the motion of the skeleton. We model this relationship with a set of functions $l_{ij}(x)$ that each map a joint configuration $x$ to an appropriate needle length, where the index $i$ ranges over all deformable primitives in the body and the index $j$ ranges over all needles in that primitive. The configuration point $x \in C_i$ describes the configuration of a deformable primitive as discussed in Section 3.3. We represent lengths $l_{ij}(x)$ using a normalized radial basis function (NRBF) [Broomhead and Lowe 1988], which interpolate prototype shapes via distance-weighted averaging:

$$l_{ij}(x) = \frac{\sum_k v_{ijk} K(x, p_{ik})}{\sum_k K(x, p_{ik})},$$

where index $k$ ranges over all prototypes. Each prototype has a location $p_{ik}$ in the configuration space $C_i$ and a shape $v_{ijk}$, which gives the length of the $j$th needle in the $k$th prototype of primitive $i$. The weighting function $K(x_1, x_2)$ is an arbitrary distance kernel. We choose a Gaussian kernel because it is well-behaved over a range of dimensionalities.

This formulation obtains better extrapolation than non-normalized radial basis functions (which go to zero as they move further from the basis locations). The NRBF extrapolates by replicating the nearest values outside the realm of observed data. In the context of skin modeling, we prefer this kind of extrapolation because it avoids generating extreme geometry for extreme configurations. Allen and colleagues [2002] use nearest-neighbor interpolation for the same reason.

Although NRBF interpolation is simple and effective, more sophisticated techniques have been developed for interpolating skin prototypes [Lewis et al. 2000; Sloan et al. 2001; Wang and Phillips 2002]. The use of these other techniques could provide better results (at the cost of increased conceptual complexity).

We use the term *prototype* because it is a conceptually useful way to think about our model. Many other methods represent deformation via the interpolation of pre-defined prototypes [Lewis et al. 2000; Sloan et al. 2001; Blanz and Vetter 1999; Allen et al. 2002]. In our work, however, the prototypes are not pre-defined. Their locations are randomly scattered in the configuration space and their shapes are inferred from the data.

## 5.2 Prototype Locations

Before we estimate the prototype shapes ($v_{ijk}$) we neeed to determine the prototype locations ($p_{ik}$). We want the prototypes to be well scattered across the space of training poses so that we can model the complete range of observed deformations.

For each deformable primitive, we greedily select prototype locations from among the set of observed points in the configuration space. We choose the first prototype location $p_{i0}$ at random from the known configurations. We then select $p_{i1}$ to be the furthest (in Euclidean distance) from $p_{i0}$ and proceed by selecting each additional prototype $p_{ik}$ to be furthest from the previously selected prototypes ($p_{il}$ for $l < k$). An exhaustive search, which is linear in the number of datapoints and quadratic in the number of prototypes, can be used to find each prototype location. The results are illustrated in Figure 7. Unlike clustering the observed configurations or sampling from the observed configurations, this results in prototypes being placed even where the data density is low.

## 5.3 Prototype Shapes

Once each prototype has been assigned to a particular location in configuration space, we can determine the shape of the prototype by finding lengths for each needle in the prototype. Due to occlusion, the length observations may include many incorrect values, so we must select multiple observations to form a reliable estimate of the correct length. Because the geometry varies with pose, we want to select these observations from nearby points in the configuration space. For each needle of each prototype, we select the $n$ nearest observations. To remove dependence on the dataset size, we choose $n$ to be equal to the number of observations multiplied by a fixed fraction $F_{near}$. By selecting the points according to this fraction instead of a fixed distance, we consider a narrow range of data where the observations are dense and a wide range of data where the observations are sparse. This satisfies the hole-filling goal by borrowing observations from other poses when there are no observations for a given pose.

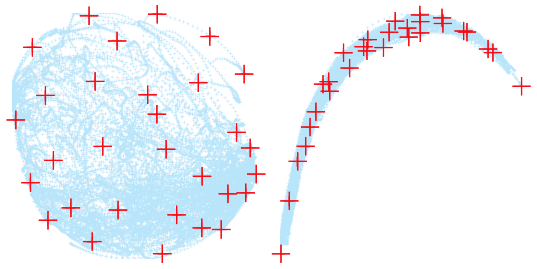

Figure 7: Prototype locations in configuration space: the small dots represent observed poses of the forearm (left) and lower leg (right). The configuration space consists of 3D marker coordinates in the bone's local coordinate system (projected into 2D for these plots). The red marks show projected locations of prototypes, which are randomly scattered across the observed configurations.

To estimate the prototype shape based on these nearby observations, we compute a robust minimum by taking the $F_{min}$ percentile observation after sorting by needle length. This achieves the goal of finding the maximal consistent geometry while allowing a small number of outliers.

The complete reconstruction algorithm is illustrated in Figure 8 and summarized as follows:

```
for each bone i do
  C_i ← get_config_space_observations(i)
  for each prototype k do
    p_ik ← find_prototype_location(k, C_i)
  end for
  for each needle j do
    S_ij ← get_needle_observations(i, j)
    for each prototype k do
      R ← nearest_neighbors(S_ij, p_ik, F_near)
      v_ijk ← robust_minimum(R, F_min)
    end for
  end for
end for
```

The nearest_neighbors($S$, $p$, $f$) function finds the fraction $f$ of points in $S$ that are closest to the point $p$.

## 5.4 Animation

The prototype locations and shapes provide a representation that is sufficient to synthesize new geometry. When animating the model for a new motion sequence, we are given a pose for each frame of the animation. The given pose determines a point in the configuration space of each deformable primitive. We then interpolate the prototype shapes (using the NRBF equation from Section 5.1) to obtain a complete geometry.

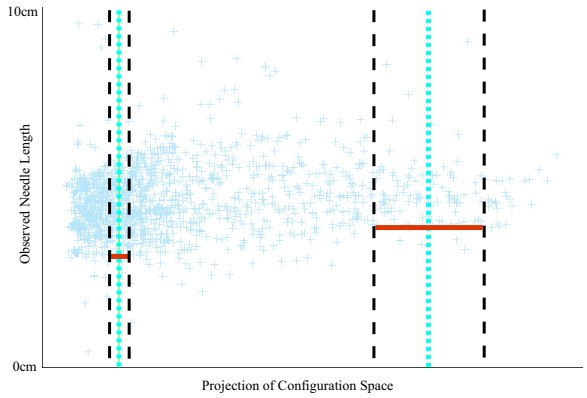


Figure 8: A plot of observed lengths for a single needle in a deformable primitive. To estimate the length of a needle at a given prototype location (blue dotted line), we consider a set of nearby observations (between black dashed lines). The neighborhood is selected as the closest fraction $F_{near}$ of observations, resulting in a narrow neighborhood where the data is dense (left) and a wide neighborhood where the data is sparse (right). Once the neighborhood is selected, we find a low percentile length value (red line) to be the length of the needle in this prototype shape.

To animate our model using motion from a different person, we need to retarget the motion to the original skeleton. This retargeting is a well-studied problem that can be performed by commercial software (for example, Kaydara's FilmBox [Kaydara 2001]). Our models can also be animated using standard key-framing techniques by mapping the motion onto the original subject's skeleton.

### 5.5 Computational Efficiency

Our system is intended for off-line reconstruction of geometry, but it is reasonably efficient. The data acquisition phase is linear in the number of frames: the background subtraction and traversal of the needles in image space is performed separately for each frame and can be done in real time. The prototype reconstruction phase is a batch process that is super-linear in the number of frames, but nonetheless can be performed quickly (we process observations from 30 minutes of video in less than 30 minutes).

## 6 Results and Analysis

Using the methods described in this paper, we have successfully reconstructed deformable models from video sequences. These models can be animated with new motion, as shown in Figure 9.

### 6.1 Experimental Setup

Our default model configuration is given in Table 1. The number of prototypes per deformable primitive and other reconstruction parameters are set as described in Section 6.3. Unless otherwise specified, all models were trained using 8 minutes of motion recorded with 3 video cameras (for a total of about 24 minutes of video). The video cameras record 720 by 480 images at 30 frames per second. The cameras were placed on one side of the workspace to allow easy segmentation using a cloth backdrop.

The motion capture system uses 10 Vicon MCAM cameras with mega-pixel resolution to track 41 reflective markers at a rate of 120 frames per second. The Vicon iQ software [Vicon 2003] extracts the position of each bone from these marker trajectories.

### 6.2 Model Validation

We quantify the accuracy of our reconstruction by comparing the observed and reconstructed silhouettes. This silhouette-based error measure is biased towards parts of the body that tend to appear on the silhouette and ignores concave parts of the surface that never appear on the silhouette from any viewpoint (such as the navel). Nonetheless, silhouette matching provides an automated way to perform various experiments about the trade-offs of our design decisions.

We measure the silhouette matching error by comparing the segmented video images to a projection of the reconstructed geometry. To reduce the effect of unmodeled geometry (such as the head), we consider only pixels near the projected silhouette boundary. We define the *silhouette error* of our algorithm on a particular dataset to be the fraction of pixels for which the predicted and observed silhouette do not match, as shown in Figure 10. We normalize the error value by dividing by the number of frames. This notion of silhouette error is effectively equivalent to the silhouette mapping error used by [Gu et al. 1999].

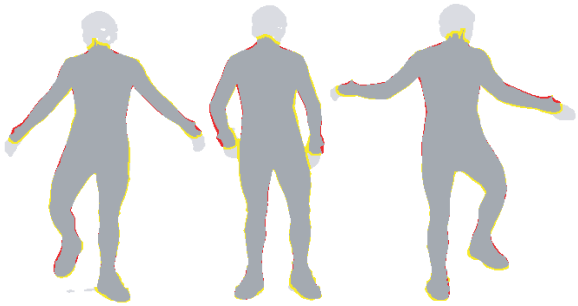

Figure 10: Pixels are colored according to differences between the estimated geometry and video silhouette: red denotes overprediction while yellow denotes underprediction. Regions that are more than a few pixels from the estimate geometry are ignored (i.e. the head and fingers).

### 6.3 Selection of Reconstruction Parameters

Using the silhouette error, we can improve the model by automatically selecting optimal reconstruction parameters. For a given arrangement of needles, the prototype estimation algorithm has four free parameters: the fraction of nearby points $F_{near}$, the percentile of the minimum point $F_{min}$, the kernel width $W$ (part of $K(x_1,x_2)$), and the number of prototypes per bone $N$. Although we were able to set these parameters manually with good results, we now describe an automatic parameter selection that produces better results.

The parameter selection algorithm varies the parameters and computes the silhouette error for each set of values. We perform repeated optimizations of each individual parameter to account for the dependence between the parameters. In each case, the other parameters were held near their optimal values ($F_{near} = 0.022, F_{min} = 0.10, W = 7$), as shown in Figure 11. Setting the number of prototypes is more difficult because the error continues to decrease as more prototypes are added; we selected $N = 100$ based on the silhouette error plot. Because this is part of the training process, we optimize these parameters using the same dataset that we use for the geometry capture.

### 6.4 Visualization

To visualize the results, we use radial basis functions (RBFs) to extract a continuous mesh from our needle endpoints. We gener-

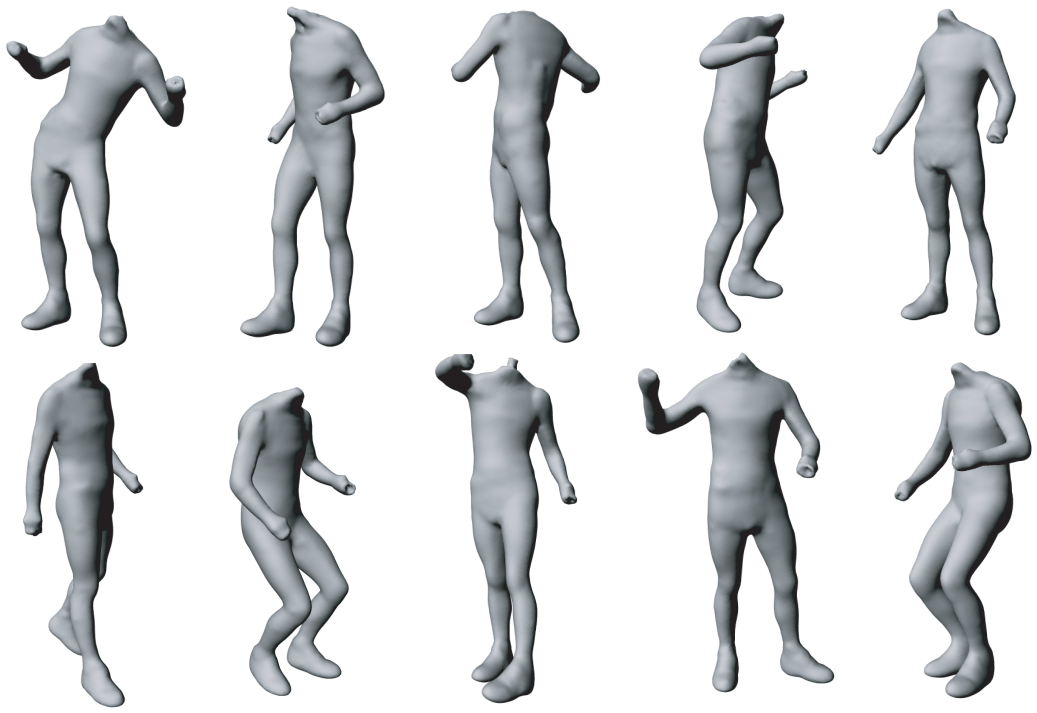

Figure 9: These meshes were synthesized for a motion sequence that was not in the training set.

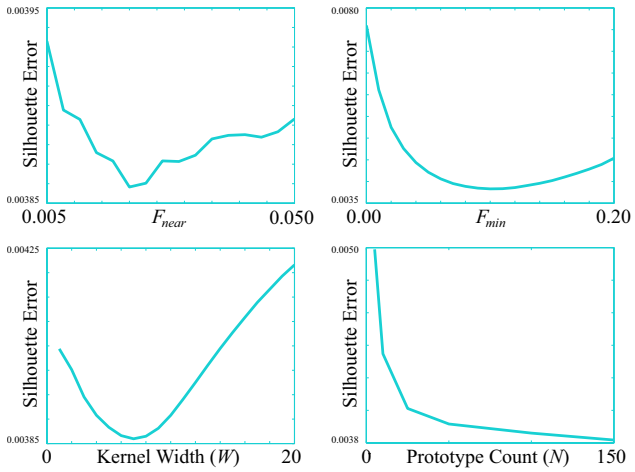


Figure 11: We use the silhouette error to automatically determine values of the estimation parameters $F_{near}$, $F_{min}$, and the kernel width $W$. The fourth plot demonstrates that the error drops as we increase the number of prototypes per bone.

ate points that are both on and above the surface, then label exterior points with the distance to the surface. This data (a total of about 15,000 points) is given to a software package (FastRBF version 1.4 [Carr et al. 2001]) that fits a radial basis function to the point/distance data and extracts an isosurface mesh.

This entire process can be scripted to render long motion sequences, but it is much too slow for real-time rendering on current hardware. Building the RBF and extracting a high-quality isosurface mesh takes about 20 seconds per frame. Section 7 discusses faster alternatives.

### 6.5 Qualitative Results

By inspection of the rendered geometry, the reconstructed models capture as much detail as a human observer can see in the source videos. Examining the surfaces, one can discern the location of geometric features such as protruding hip bones and the belt of the motion-capture suit. The primary flaws seem to occur in regions of high deformation (e.g. a twisting torso) or where the surface was rarely on the silhouette (e.g. at the junction of the legs).

### 6.6 Sources of Error

The number of needles can be increased arbitrarily without concern for overfitting. This increases the spatial resolution of the surface at the cost of longer computation. Even with a high needle density, certain geometries cannot be accurately represented. For example, when using a perpendicular needle arrangement, the model cannot represent deep folds in the skin such as those that occur under drooping breasts and stomachs. Not only are these kinds of surfaces hard for the model to represent, but they are difficult for our algorithm to acquire because they rarely (if ever) appear on the silhouette. In practice, however, these parts of the body would typically be covered with clothing placed on top of the acquired model.

The number of prototypes can also be increased arbitrarily (again at a computational cost). Overfitting is possible, but this is determined by the fraction ($F_{near}$) of nearby points contributing to each prototype. Adding prototypes without adjusting this fraction does not cause overfitting so long as the fraction is sufficiently high that valid observations are selected for each prototype.

In practice, the generality of the deformation model is not fully exploited because of flaws in the data acquisition and reconstruction processes. Camera resolution introduces an error on the order of a pixel for each observation. However, because we typically have multiple observations of each surface patch, we can in principle combine these observations in a way that allows sub-pixel accuracy.

This super-resolution effect is lost due to other sources of error, such as the accuracy of the motion-capture system. Modern motion-capture systems are able to track markers with high precision, but the markers do not provide a perfect estimate of bone position because they are placed on the deforming skin. Inconsistent bone estimation appears to be a substantial source of error in our reconstructions.

Another possible source of reconstruction error is silhouette extraction. If too many pixels are mislabeled as background when they are really foreground, the robust minimum could fail, resulting in holes in the geometry. Fortunately, we can easily avoid this by reducing the background subtraction threshold. This will result in labeling some background pixels as part of the foreground, but such errors are not a problem because the algorithm assumes that the silhouette provides only an upper bound on the geometry.

The quality of the silhouettes can also be effected by motion blur. Because the video cameras use a relatively large exposure window (e.g. $\frac{1}{60}$ of a second), the motion of the subject introduces up to a couple pixels of blur. We were unable to use shorter exposure times due to interference with the fluorescent lighting. An ideal capture environment would use bright incandescent lights (allowing very short exposure windows) and a chroma-key background (allowing better foreground extraction).

The final and most complicated source of error is the range of input motion. Ideally we would make a valid (non-occluded) observation of each needle at each prototype location. When this is not the case, we need to increase $F_{near}$ to borrow values from other parts of the configuration space. Since we take a minimum (albeit a robust minimum) of the borrowed values, we will underestimate the geometry in regions of deformation.

To minimize this problem, we direct the subject to move through a wide range of poses. In our experiments, we found that a few minutes of video from a single camera was sufficient to build a decent model. However, because we can easily gather additional data, we also considered larger datasets consisting of multiple video cameras and up to 8 minutes of video footage per camera. By adding cameras, we effectively reduce the amount of performance time needed to obtain a given level of quality. In Figure 12 we illustrate the influence of the amount of data on the quality of the results.

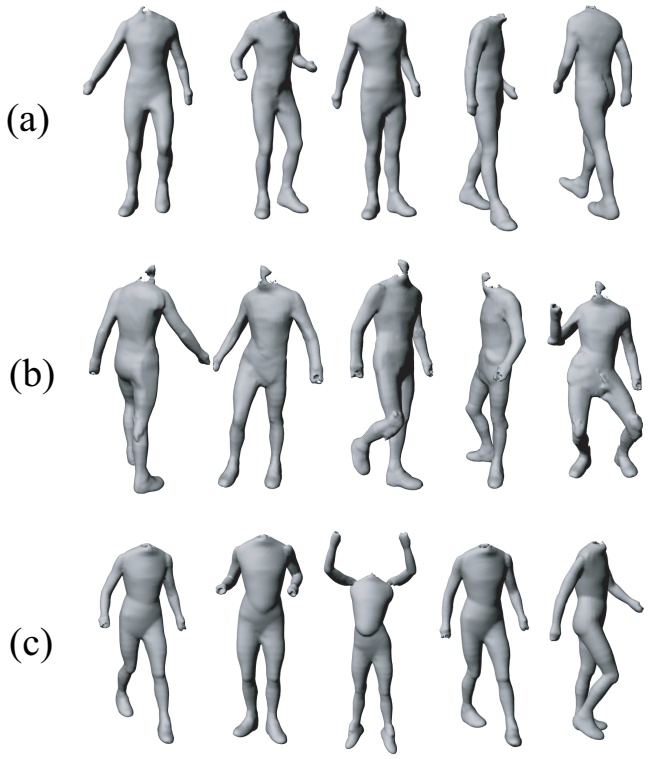


Figure 12: **Part (a):** With 3 minutes of motion observed with a single camera, we can obtain a good model, but its range of motion is limited. **Part (b):** With only 30 seconds of motion observed from a single camera, the model has a number of unpleasant artifacts. **Part (c):** When we train a model without any deformation (by setting $F_{near} = 1$), the joints are poorly represented, illustrating that deformation is essential to an accurate human skin model.

## 7 Conclusion

We have presented a new method for digitizing skin geometry using motion capture and video cameras. We attach needles to a skeleton obtained from motion capture, then determine where these needles intersect silhouettes to obtain constraints on the geometry. These observed constraints are accumulated and filtered—simultaneously solving the problems of occlusion, hole-filling, deformation, and noise-removal. Using a few minutes of video footage, we can create a human model that can be animated with new motions. The quality of our reconstruction is primarily limited by the amount of detail captured in the silhouette, the accuracy of skeleton estimation from motion-capture markers, and the range of motion in the training set.

Our primary future goal is to increase our reconstruction accuracy. We would like to consider other estimation algorithms, such as ones that use advanced visibility reasoning, make probabilistic models of noise and occlusion, or perform iterative refinement of the surfaces found by our reconstruction algorithm. We would also like to investigate the use of additional configuration space parameters, such as the direction of gravity and estimated muscle force. Additionally, we hope to use better methods for estimating skeletons from the motion-capture data and improve input fidelity by using mega-pixel FireWire cameras, better lighting, and a chroma-key background. To validate these improvements, we intend to compare our reconstruction results with a synthetic model by rendering silhouettes to train our model.

We also intend to investigate faster ways to obtain a continuous surface mesh from the interpenetrating deformable primitives. One option would be to reorient the needles (as a function of pose) such that they do not overlap and permit a single continuous triangulation over the entire body. Alternately, we could fit a mesh to our existing geometry and iteratively re-fit the mesh as the underlying skeleton moves. In either case, our existing acquisition and estimation algorithm could still be used.

For many animation purposes, such as creating large crowds of extras, animators would like to create new geometries without capturing additional people. Given data for a variety of people, we could create a basis of human geometries, including variations such as male vs. female (see Figure 13), thin vs. fat, muscular vs. smooth. By interpolating prototype shapes, we would automatically obtain not only new geometry but also new deformations.

Recent work in markerless motion capture [Mikić et al. n. d.; Theobalt et al. 2002] suggests that we may be able to use our method without requiring special motion-capture equipment. By eliminating the need for a marker-covered suit, we could capture meaningful skin texture by projecting video images onto the model and examining how the texture appearance changes as a function of pose. Because we have estimates of geometry, this method could even account for variations in reflectance and lighting. Furthermore, by capturing body texture, we could make use of the factorization methods described in Section 2, allowing reconstruction of concave regions such as the eyes. This would be a substantial step toward complete and automatic acquisition of human subjects.

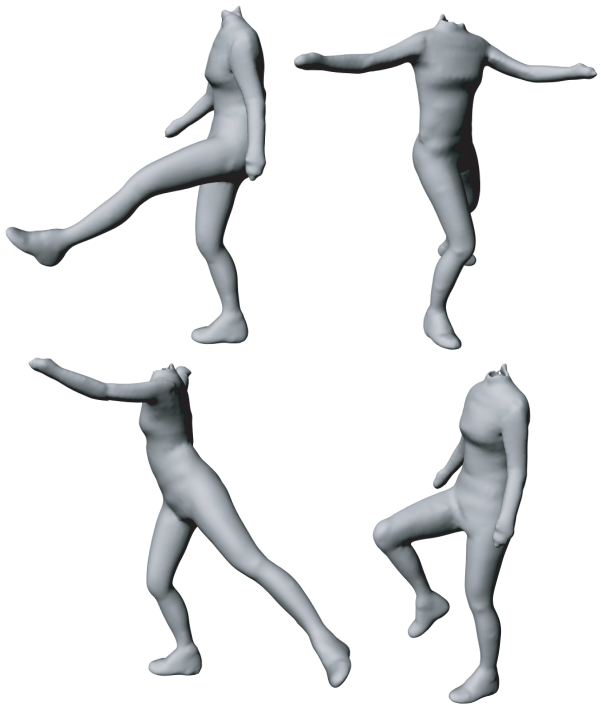

Figure 13: This model was generated from a female subject using 5 minutes of motion and silhouettes from three viewpoints. In the future we would like to capture a wide variety of people and interpolate their geometries to synthesize new deformable models.

## Acknowledgements

Special thanks to Frédo Durand, Seth Teller, William Freeman, Farid Jahanmir, Barb Cutler, Tom Buehler, Adnan Sulejmanpašić, Daniel Vlasic, Eric Chan, and Eugene Hsu. We are grateful to Vicon Motion Systems for their permission to use the beta release of the Vicon iQ software. This research was sponsored in part by the NTT Corporation.

## References

ALLEN, B., CURLESS, B., AND POPOVIĆ, Z. 2002. Articulated body deformation from range scan data. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 612–619.

BLANZ, V., AND VETTER, T. 1999. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 187–194.

BRAND, M. 2001. Morphable 3D models from video. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, II:456–463.

BREGLER, C., HERTZMANN, A., AND BIERMANN, H. 2000. Recovering non-rigid 3D shape from image streams. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, II:690–696.

BROOMHEAD, D., AND LOWE, D. 1988. Multivariable functional interpolation and adaptive networks. *Complex Systems 2*, 3, 321–355.

CARR, J. C., BEATSON, R. K., CHERRIE, J. B., MITCHELL, T. J., FRIGHT, W. R., MCCALLUM, B. C., AND EVANS, T. R. 2001. Reconstruction and representation of 3d objects with radial basis functions. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 67–76.

COOTES, T. F., EDWARDS, G. J., AND TAYLOR, C. J. 1998. Active appearance models. In *Proceedings of the Fifth European Conference on Computer Vision (ECCV)*, 484–498.

GU, X., GORTLER, S. J., HOPPE, H., MCMILLAN, L., BROWN, B. J., AND STONE, A. D. 1999. Silhouette mapping. Tech. Rep. TR-1-99, Harvard.

GUENTER, B., GRIMM, C., WOOD, D., MALVAR, H., AND PIGHIN, F. 1998. Making faces. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 55–66.

KAKADIARIS, I. A., AND METAXAS, D. 1993. 3d human body model acquisition from multiple views. In *Proceedings of the 5th IEEE International Conference on Computer Vision (ICCV)*, 618–623.

KAYDARA. 2001. *FiLMBOX Reference Guide*. Kaydara Inc., Montréal, Québec.

LEWIS, J. P., CORDNER, M., AND FONG, N. 2000. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 165–172.

MATUSIK, W., BUEHLER, C., RASKAR, R., GORTLER, S. J., AND MCMILLAN, L. 2000. Image-based visual hulls. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 369–374.

MIKIĆ, I., TRIVEDI, M., HUNTER, E., AND COSMAN, P. Human body model acquisition and tracking using voxel data. *International Journal of Computer Vision*. In Press.

NEBEL, J.-C., RODRIGUEZ-MIGUEL, F. J., AND COCKSHOTT, W. P. 2001. Stroboscopic stereo rangefinder. In *Proceedings of the Third International Conference on 3D Imaging and Modeling*, 59–64.

NELDER, J. A., AND MEAD, R. 1965. A simplex method for function minimization. *Computer Journal 7*, 4, 308–313.

NEVATIA, R., AND BINFORD, T. O. 1977. Description and recognition of curved objects. *Artificial Intelligence 8*, 1, 77–98.

PLÄNKERS, R., AND FUA, P. 2001. Articulated soft objects for video-based body modeling. In *Proceedings of the 8th IEEE International Conference on Computer Vision (ICCV)*, I:394–401.

RUSINKIEWICZ, S., HALL-HOLT, O., AND LEVOY, M. 2002. Real-time 3d model acquisition. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 438–446.

SLOAN, P.-P. J., CHARLES F. ROSE, I., AND COHEN, M. F. 2001. Shape by example. In *Proceedings of the 2001 symposium on Interactive 3D Graphics*, 135–143.

STOKDYK, S., HAHN, K., NOFZ, P., AND ANDERSON, G., 2002. Spiderman: Behind the mask. Special Session of SIGGRAPH 2002.

SZELISKI, R., AND TONNESEN, D. 1992. Surface modeling with oriented particle systems. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, 185–194.

THEOBALT, C., MAGNOR, M., SCHUELER, P., AND SEIDEL, H.-P. 2002. Combining 2d feature tracking and volume reconstruction for online video-based human motion capture. In *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, 96–103.

TORRESANI, L., AND BREGLER, C. 2002. Space-time tracking. In *Proceedings of the 7th European Conference on Computer Vision (ECCV)*, 801–812.

VEDULA, S., BAKER, S., AND KANADE, T. 2002. Spatio-temporal view interpolation. In *Proceedings of the 13th ACM Eurographics Workshop on Rendering*, 65–76.

VICON. 2003. *Vicon iQ Reference Manual*. Vicon Motion Systems Inc., Lake Forest, CA.

WANG, X. C., AND PHILLIPS, C. 2002. Multi-weight enveloping: least-squares approximation techniques for skin animation. In *Proceedings of the ACM SIGGRAPH symposium on Computer animation*, 129–138.

WÜRMLIN, S., LAMBORAY, E., STAADT, O. G., AND GROSS., M. H. 2002. 3d video recorder. In *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, 325–334.