Reconstruction of Deforming Surfaces from Moving Silhouettes

by

Peter Sand

Submitted to the Department of
Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2003

Author ....
Department of
Electrical Engineering and Computer Science
May 9, 2003

Certified by ....
Jovan Popović
Assistant Professor
Thesis Supervisor

Accepted by ....
A. C. Smith
Chairman, Department Committee on Graduate Students

Reconstruction of Deforming Surfaces from Moving Silhouettes

by

Peter Sand

Submitted to the Department of
Electrical Engineering and Computer Science
on May 9, 2003, in partial fulfillment of the
requirements for the degree of
Master of Science in Computer Science and Engineering

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.

Thesis Supervisor: Jovan Popović
Title: Assistant Professor

Acknowledgments

Thanks to Jovan Popović for helping out in numerous ways, including giving feedback on the endless revisions of the text. Thanks to Leonard McMillan, Frédo Durand, Seth Teller, William Freeman for useful comments and suggestions. Thanks to Barb Cutler and Adnan Sulejmanpašić for enduring the motion-capture suits. Thanks to Farid Jahanmir, Tom Buehler, Daniel Vlasic, Eric Chan, and Eugene Hsu for technical help. 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.

Contents

List of Figures

List of Tables

Chapter 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 [22].

We demonstrate an automatic method for reconstructing skin geometry from observations of a moving person. We build a model of the subject's skin deformations using 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 complete models for each section of the body.

The deformation of each body part is represented using prototype shapes that are parameterized according the skeleton pose. After reconstructing multiple shapes of each body part, we combine these shapes to fill an exponentially larger space of complete body poses. The junctions between the parts remain consistent because each body part takes the correct shape for the local skeleton pose.

We obtain a high level of reconstruction accuracy through the use of silhouettes. In an environment that allows good segmentation, silhouettes never underpredict the true geometric volume. Because this error is one-sided, it can be removed with relative ease. Within our algorithm, all possible sources of two-sided error (such as mistakes in calibration and synchronization) are carefully minimized, while the one-sided error is accepted and subsequently removed using a simple reconstruction technique. This allows our approach to simplify the traditionally difficult issues of visibility and occlusion.

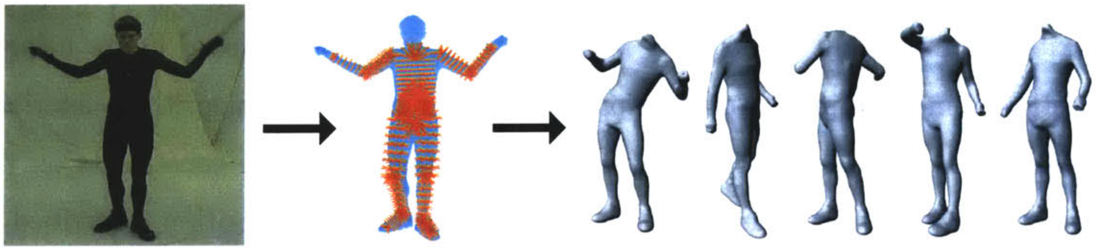

Figure 1-1: We extract silhouettes from video sequences to build a deformable skin model that can be animated with new motion.

1.1 Dependence on Tracking Systems

Our approach is dependent on having high-quality positions of the skeleton of the subject. Fortunately, such information can be obtained from commercial hardware and software systems that track the motion of the skeleton in a process known as motion capture. Motion-capture systems can use optical, magnetic, or mechanical sensors. Our implementation uses an optical system that tracks reflective markers placed on the body surface.

Throughout our work, we assume that we know the positions of the markers placed on the subject's skin and the 3D locations of the subject's major bones. Obtaining bone positions from skin marker positions is an open research problem, but effective solutions have been implemented in commercial motion-capture software. Our approach relies heavily on the accuracy of bone localization; we need the bone locations to be close biologically correct.

The reliance on a motion-capture device is a limitation of our approach, but it allows a complex problem to be solved by using a pre-built system (literally a black box) to handle a difficult sub-problem. Our work is the first that has effectively used a motion-capture system to acquire human skin geometry.

1.2 Summary of Contributions

This thesis describes a set of algorithms for combining silhouette observations into a complete 3D model that can be animated with new motions. In Chapter 3, we describe a simple skin model that represents a complex articulated figure using a collection of intersecting deformable primitives. Our acquisition algorithm, described in Chapter 4, uses the silhouettes from video footage to obtain constraints on the geometry of the deformable primitives. The reconstruction algorithm, described in Chapter 5, parameterizes these deformations with the motion of the skeleton, allowing the skin model to predict skin geometries for a new skeleton pose.

While each of these components is similar to methods that have been previously developed, the manner in which we combine the techniques is new. By selecting the right combination of components—a needle-based skin representation, silhouette-based constraints, and specialized reconstruction—we are able to find a simple and effective solution. Our choice of how to gather and represent our data allows the difficult problem of 3D reconstruction to be reduced to a much simpler problem of filtering and estimation. The system produces results of a quality that has not been obtained before for models of a moving person.

Chapter 2

Related Work

2.1 Reconstruction of General Moving Objects

The most general 3D reconstruction systems 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 [16] and voxel coloring [26]. 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 [14, 29].

Some of these methods make frame-to-frame comparisons of the geometry [29, 26], 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 object can be moved while it's digitized [20]. 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 [4, 3, 25]. The algorithms use texture or surface intensity to estimate motion in the image. These methods overcome the uncertainty that traditionally troubles image-based motion estimation by assuming that the observed object is a morphable model (a linear combination of basis shapes), which constrains possible optical flow measurements. A matrix of image observations is then 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 define 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.

All of these methods are intended to reconstruct models of generic subjects. In contrast, our approach is aimed specifically at reconstruction of skeleton-driven bodies, which allows the approach to be used in conjunction with skeleton-based animation. Furthermore, by assuming the existence of a skeleton, we are able to create human models with greater spatial and temporal resolution than any of the methods for capturing general moving objects.

2.2 Model-Based Object Reconstruction

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 [10, 7, 2]. 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 the purpose of motion tracking, these models can be quite simple, using a set of ellipses and cylinders to represent the human form. To improve tracking quality, the dimensions of these model components can be fit to observations of a particular person. Mikić and colleagues [15] use silhouettes to set the dimensions of ellipses used to represent a tracked subject.

Plänkers and Fua [19] use an elaborate anatomical model, in which the skin surface is implicitly described by the level-set surface of various Gaussians rigidly attached to a skeleton. Observations from silhouettes and stereo reconstruction are used to optimize the dimensions of the Gaussians in order to match the anatomical model to an observed human. The demonstrated results do not model any deformation and do not show substantial changes from the original anatomical model, which is assumed to be given.

Kakadiaris and Metaxas [11] develop a technique that combines orthogonal 2D contours to estimate a deforming 3D body shape. The subject moves through a pre-defined protocol of motions aimed at given a set of three mutually orthogonal cameras the necessary views to construct a complete body model. Silhouettes provide 2D contours that are combined across multiple frames in time; 3D geometry and deformation are generated by interpolating the 2D contours. This method is similar to ours in that it combines silhouette data from multiple poses, but the resulting model is not shown in its entirety and not demonstrated through animation.

2.3 Interpolation of Static Poses

Allen and colleagues [1] 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 [13, 21, 28]) 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 thesis. Instead, we focus on how to position and reconstruct prototype shapes in a fast and automatic manner.

Chapter 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 [18]. 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 3-1. 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 [23]. 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.

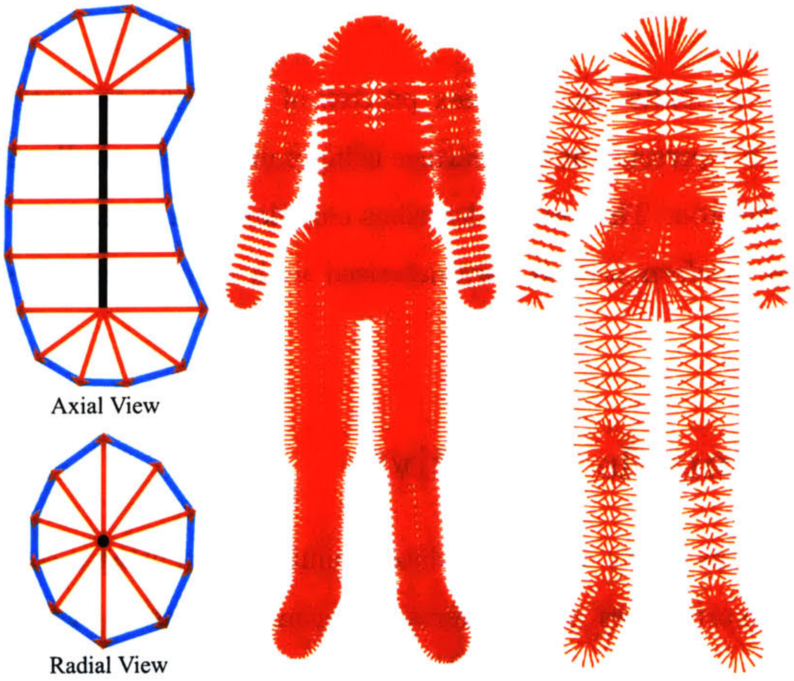

Figure 3-1: Deformable primitives describe the human body with variable-length needles (red) attached to a fixed axis (black). The left skeleton uses needle counts given in Table 3.1. The skeleton on the right uses one quarter as many needles (half as many radially and half as many axially). In both 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 3-2. 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 Chapter 7.

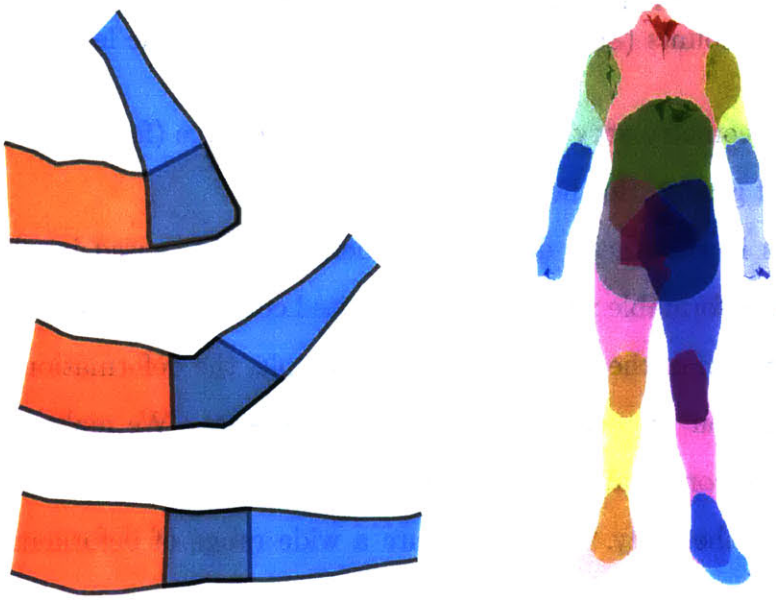

Figure 3-2: We represent an elbow using overlapping deformable primitives for the upper arm and forearm. Both primitives deform as the elbow bends, maintaining continuity in the junction. The image on the right shows how the segments overlap in a complete body.

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

Name of		**Dim. of**			**Maximum**
Deformable	**Configuration**	**Config.**	**Radial**	**Axial**	**Needle**
Primitive	**Depends On**	**Space**	**Needles**	**Needles**	**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 3.1: 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).

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 thesis 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 3.1 summarizes the configuration parameters for each deformable primitive.

Chapter 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 4-1. In a multi-camera system, we allow some cameras to be placed closer to the subject in order to obtain more detail for certain parts of the model.

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 Chapter 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

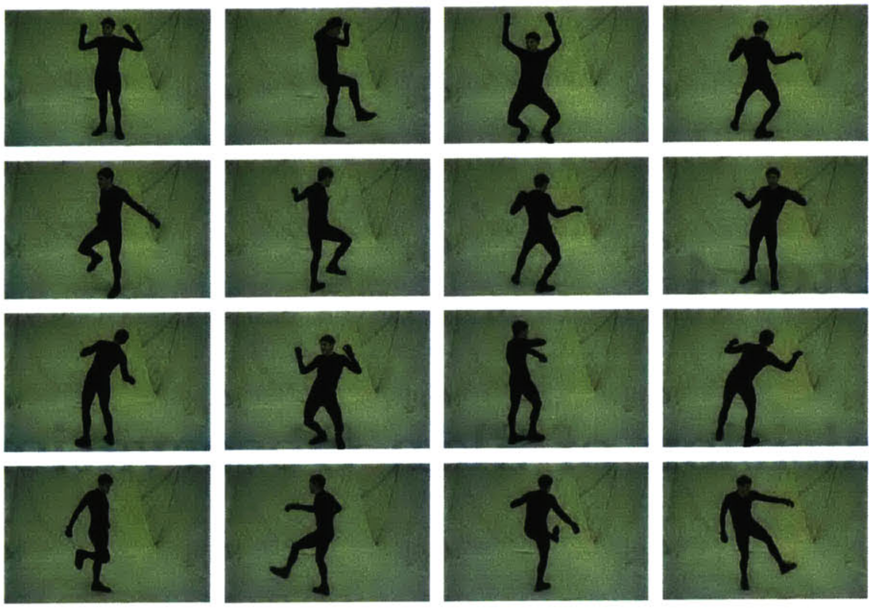

Figure 4-1: The input video includes images of the subject in a wide variety of poses. As discussed in Section 6.5.3, the quality of the final model depends on the range of motion in the input sequence.

calibration using a simple device, shown in Figure 4-2, 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.

After synchronization, the calibration device can be moved freely in three dimensions. We resample the resulting motion-capture sequence 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 [17]. The method quickly converges to a solution that obtains a sub-pixel RMS error over several hundred (w_i, p_i) input points.

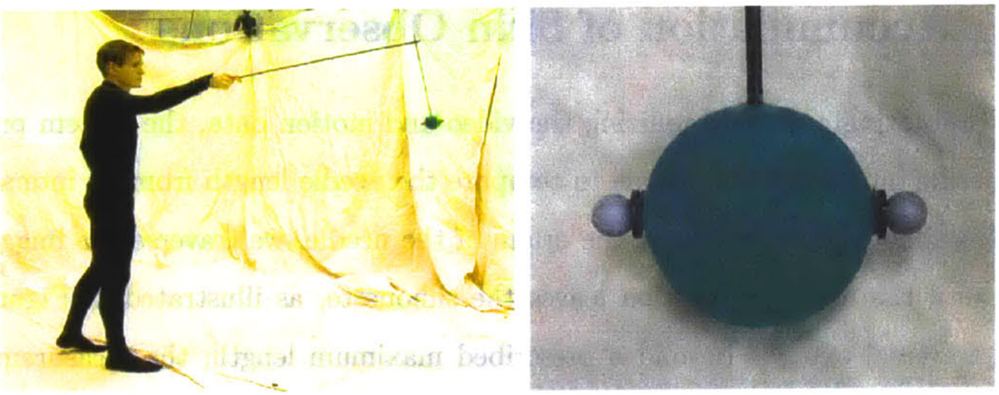

Figure 4-2: Our calibration device consists of a green sphere with two motion-capture markers. We find the center of the sphere in image coordinates by detecting green pixels. We find the center of the sphere in world coordinates by taking the midpoint of the two marker positions. This gives a single 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 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. Because we are synchronizing observations of a human, we use a different method than used for synchronizing the calibration data. Our system uses 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 Skin 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 4-3. 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 3.1) are relatively large; the same values can be used for a wide variety of people.

Based on the world-space orientation of the needle, we convert the image-space measurement into a world-space length. For example, we may compute that in frame 117, viewed from camera 2, needle 17 of bone 3 intersects the silhouette at a distance of 10cm from the bone axis.

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.

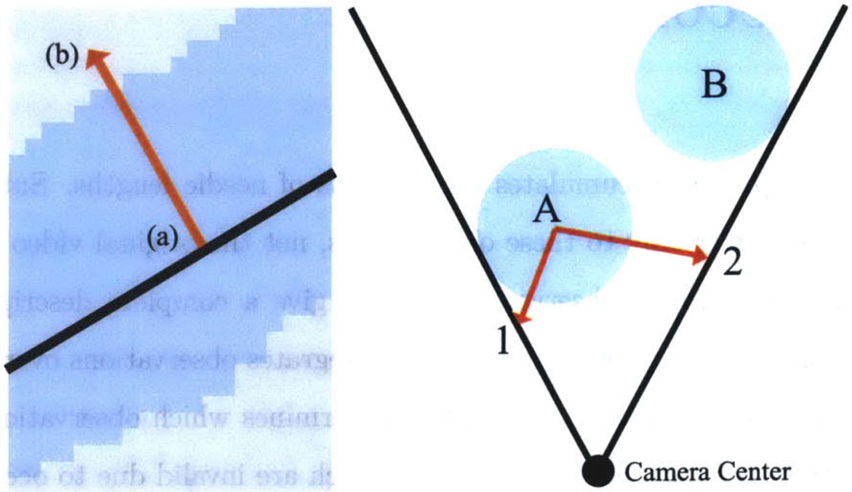

Figure 4-3: **Left:** To obtain a needle length observation, we project the needle into the image plane. We traverse the image along the needle (from (a) towards (b)), to find the image space distance from the bone to the edge of the silhouette (in blue). This length is converted to a world space distance 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.

Chapter 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 4-3, 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.

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 ob-

served 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.

Surface smoothness is one criterion we choose not to include in our list of design goals. Although we seek temporal smoothness, we do not wish to impose artificial spatial smoothness. Needle-to-needle smoothness constraints could possibly improve some parts of the model, but they would remove detail from other parts of the model. Our experiments suggest that spatial smoothness constraints are unnecessary for our system.

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. Each function depends on a configuration point $x \in C_i$ that describes the configuration of a deformable primitive as discussed in Section 3.3.

We represent each length function $l_{ij}(x)$ using a normalized radial basis function (NRBF) [5], 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 jth needle in the kth 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 [1] 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 [13, 21, 28]. 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 predefined prototypes [13, 21, 2, 1]. In our work, however, the prototypes are not predefined. 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 5-1. Unlike clustering the observed configurations or sampling from the observed configurations, this results in prototypes being placed even where the data density is low.

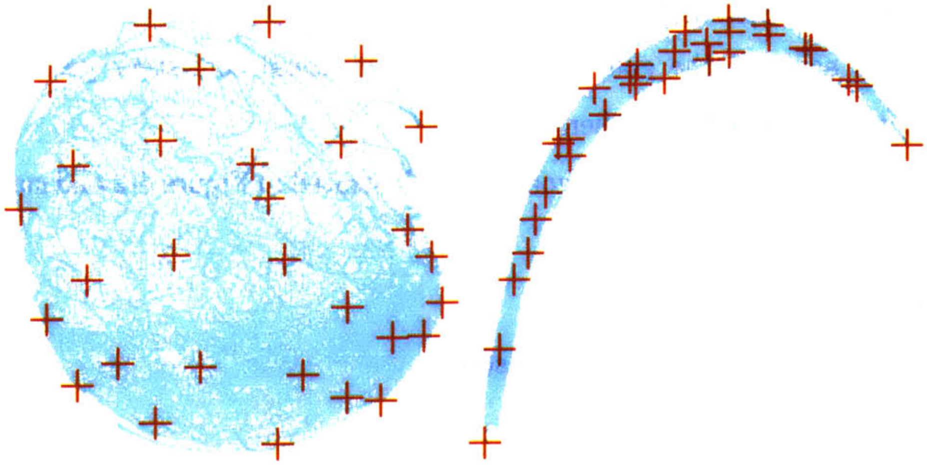

Figure 5-1: Prototype locations in configuration space: the small dots represent observed poses of the forearm (left) and lower leg (right). The configuration space consists of 3D marker coordinates in the bone's local coordinate system (projected into 2D for these plots). The red marks show projected locations of prototypes, which are randomly scattered across the observed configurations.

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.

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 5-2 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

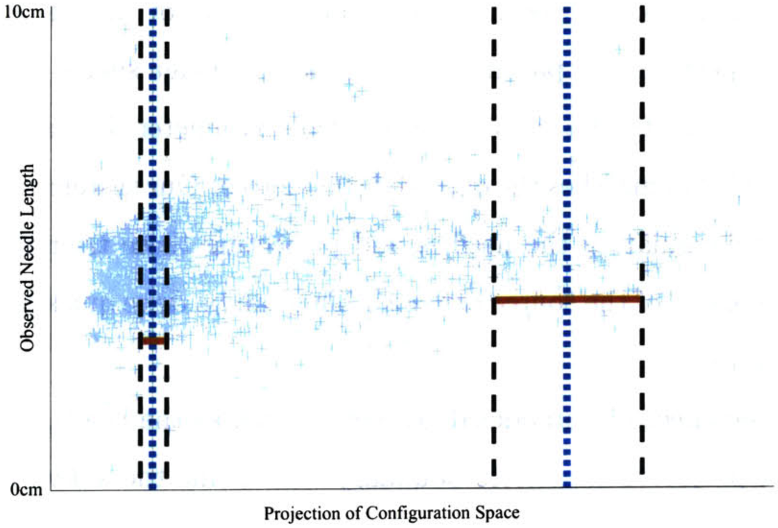

Figure 5-2: A plot of observed lengths for a single needle in a deformable primitive. To estimate the length of a needle at a given prototype location (blue dotted line), we consider a set of nearby observations (between black dashed lines). The neighborhood is selected as the closest fraction F_{near} 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.

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.

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 [12]). Our models can also be animated using standard key-framing techniques by mapping the motion onto the original subject's skeleton.

5.5 Selection of Reconstruction Parameters

For a given placement of needles, our 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. We seek a way to select these parameters using a training dataset or validation dataset of the same form. Because these datasets do not include any 3D surface information, we perform our parameter evaluation in image space.

We determine quality of a set of reconstruction parameters by measuring how well the reconstructed skin matches the observed silhouettes. Our system renders synthetic silhouettes by triangulating the needles and projecting these triangles according to the parameters of one of the source cameras. This can be done quickly using standard graphics hardware.

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 5-3. 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 [9].

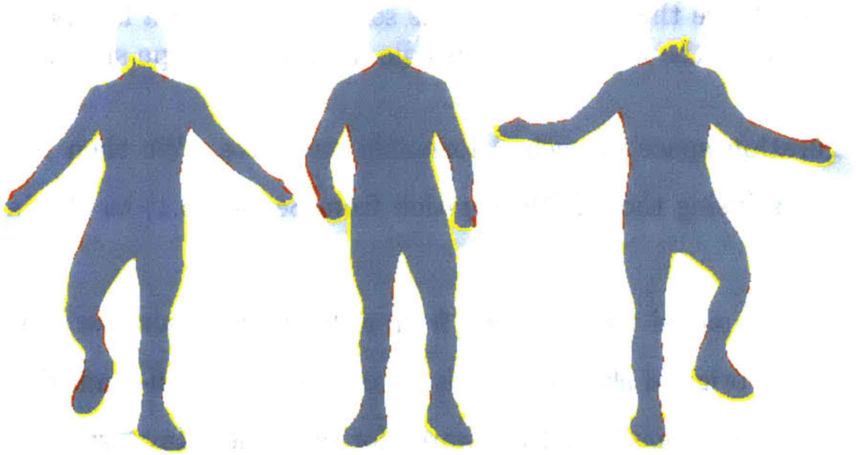

Figure 5-3: Pixels are colored according to differences between the estimated geometry and video silhouette: red denotes overprediction while yellow denotes underprediction. Regions that are more than a few pixels from the estimate geometry are ignored (i.e. the head and fingers). The *silhouette error* for a given frame is defined to be the fraction of pixels overpredicted or underpredicted. Potential causes of the error are discussed in Section 6.5.

When reconstructing geometry, matching the silhouette is necessary but not sufficient for matching 3D reality. Our 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, the silhouette error provides an automated way to perform various experiments about the trade-offs of our design decisions.

The parameter selection algorithm varies the reconstruction 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. Figure 5-4 shows plots of each parameter vs. silhouette error. In each plot, the other parameters were held near their optimal values ($F_{near} = 0.022, F_{min} = 0.10, W = 7$). 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.

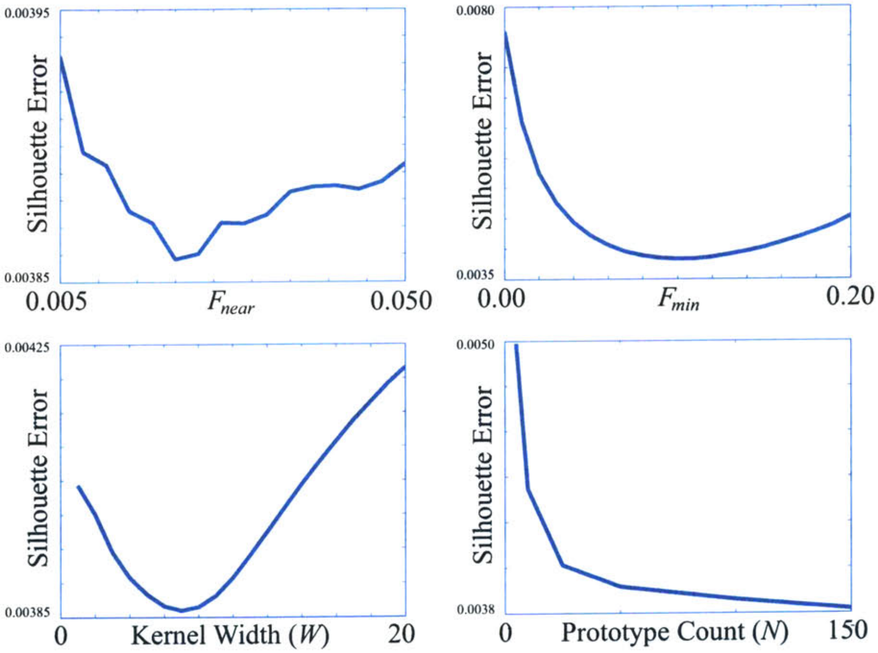

Figure 5-4: We use the silhouette error to automatically determine values of the estimation parameters F_{near}, F_{min}, and the kernel width W. The fourth plot demonstrates that the error drops as we increase the number of prototypes per bone.

Chapter 6

Results

Using the methods described in this thesis, we have successfully reconstructed deformable models from video sequences. These models can be animated with new motion, as shown in Figure 6-1.

6.1 Experimental Setup

Our default model configuration is given in Table 3.1. The number of prototypes per deformable primitive and other reconstruction parameters are determined as described in Section 5.5. Unless otherwise specified, all models were trained using 8 minutes of motion recorded with 3 cameras (for a total of about 24 minutes of video).

The cameras were placed on one side of the workspace to allow easy segmentation using a cloth backdrop. Each camera uses the MiniDV tape format with a resolution of 720 by 480 pixels, recorded at 30 frames per second. In some cases, the cameras recorded interlaced video, in which case we used software de-interlacing (effectively reducing the vertical resolution to 240 pixels). Progressive-scan (non-interlaced) recording was used when available.

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 [27] extracts the position of each bone from these marker trajectories.

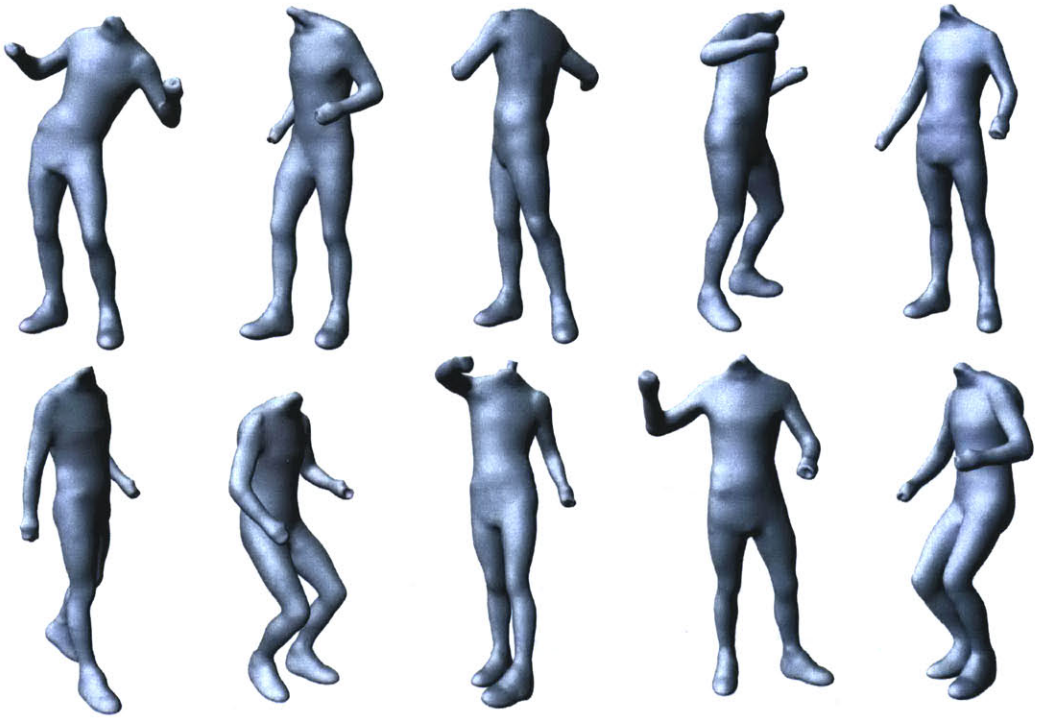

Figure 6-1: These meshes were synthesized for a motion sequence that was not in the training set.

6.2 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).

6.3 Visualization

To visualize the results, we use radial basis functions (RBFs) to extract a continuous mesh from our needle endpoints. We generate 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 [6]) 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 iso-surface mesh takes about 20 seconds per frame on current desktop hardware. Chapter 7 discusses faster alternatives.

6.4 Qualitative Discussion

By inspection of the rendered geometry, the reconstructed models for the training poses 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).

When the poses in the test motion sequence are substantially different than the poses in the training sequence, we observe significant artifacts. This and other limitations of our approach are discussed in more detail in the following chapter.

Ideally we would compare our 3D reconstruction results to those obtained via another method. Unfortunately, no other method is able to acquire a 3D model of a moving person with a suitable level of resolution. Because other methods [26, 16, 11] appear to have lower accuracy, a comparison with them would not establish the relative or absolute correctness of our models.

A comparison with a static 3D scan would be useful, but such a comparison would be entirely dependent on the positioning of bones within the scanned body. We would effectively be measuring how well we could match the pose of our model to the pose of the scanned person, not measuring the actual accuracy of our model.

Alternatively, we could compare the generated results to a known synthetic model by rendering silhouettes of this model in various poses. However, this comparison would be dependent on a number of technical details which are not within the scope of this work.

6.5 Sources of Error

The quality of our reconstructed geometry is influenced by many factors, such as the camera resolution and the range of input motion. Some of these sources of error can be avoided by our algorithm or minor variations of of our algorithm. However, a few of these error sources are due to fundamental limitations of our approach.

6.5.1 Constraints of the Needle Model

The number of needles can be increased arbitrarily without concern for overfitting. Thus, at the cost of increased computation, our model does not limit the spatial resolution of surfaces.

Even with an arbitrarily 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 at all) 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.

6.5.2 Input Resolution and Noise

In practice, the expressiveness of the model is not fully exploited because of flaws in the data acquisition and estimation processes. Camera resolution is one limitation that results in an error on the order of a pixel for each sample. 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).

6.5.3 Range of Motion

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.

Clearly we cannot expect the subject to move through all possible human poses. Fortunately, we can cope with an exponential number of body poses by observing a small number of poses for each body part. However, in some cases, we may not have good observations for a desired pose of a particular body part. This may result in dramatically incorrect prototype shapes that can be exposed during the animation of a new motion sequence.

To minimize this extrapolation problem, we direct the subject to move through a wide range of poses. This is not an ideal solution, but it is certainly feasible for filmmaking and video-game applications, where the desired kinds of motion are known in advance. The subject does not need to act out the desired motion, but at least move through a suitable range of configurations for each joint.

In our experiments, we found that a few minutes of video from a single camera was sufficient to build a decent model. 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 6-2 we illustrate the influence of the amount of data on the quality of the results.

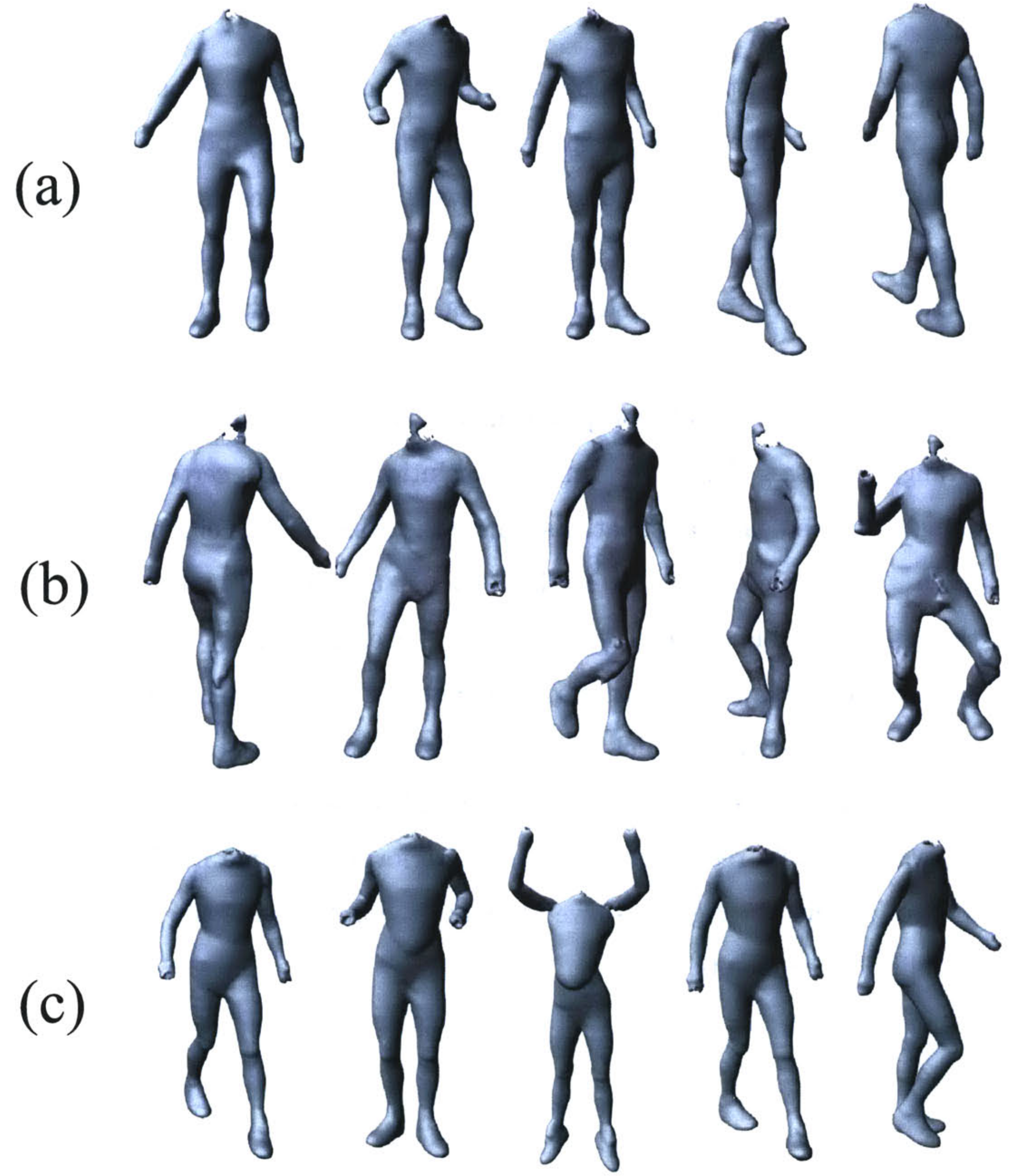

Figure 6-2: **Part (a):** With 3 minutes of motion observed with a single camera, we can obtain a good model, but its range of motion is limited. **Part (b):** With only 30 seconds of motion observed from a single camera, the model has a number of unpleasant artifacts. **Part (c):** When we train a model without any deformation (by setting $F_{near} = 1$), the joints are poorly represented, illustrating that deformation is essential to an accurate human skin model.

Chapter 7

Conclusion

We have presented a new method for digitizing skin geometry using a motion tracking system and video cameras. Our method attaches needles to a skeleton obtained from motion capture, then determines where these needles intersect the silhouettes to obtain estimates of geometry. These estimates are accumulated and filtered—simultaneously solving the problems of occlusion, hole-filling, deformation modeling, 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.

Despite these limitations, this work provides progress in the direction of automatically acquiring human geometry. This has a wide range of uses, such as 3D video conferencing, documenting important speeches, analyzing sporting events, identifying people, fitting custom clothing, performing physical evaluation of athletes and medical patients, and creating synthetic actors for video games and films.

7.1 Short-term Extensions

This work leaves open a variety of improvements that can be made via variations on our technique. For example, the input fidelity could be improved by using mega-pixel FireWire cameras, better lighting, a chroma-key background, and improved estimation of skeleton positions from the motion-capture data.

In addition to improving the input quality, we could consider a more sophisticated reconstruction algorithm. Advanced visibility reasoning could remove excess errant observations, reducing the robustness demands on the reconstruction process. Separately or in conjunction with this, probabilistic models of noise and occlusion could be used to develop a method that finds a maximum likelihood deformation function.

Another aspect of our algorithm that deserves improvement is the representation of the reconstructed geometry. For real-time rendering applications, we would need much 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, permitting 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. (Not only would this speed rendering, but it could improve skin texturing, so long as the surface mesh moves across the underlying geometry like skin moves across the underlying muscles and bones.) In either case, our acquisition and reconstruction algorithm could be used without modification.

7.2 Long-term Extensions

Once these core technical issues have been addressed, further work remains in making this a wide-spread technology. One such improvement would be generalizing the models across multiple people. This could reduce the difficulty of extrapolation: if we see how one body deforms in a particular pose, we can by analogy determine how a different body deforms. In addition to extrapolating to new poses, we could generate body geometries that are different from ones that have been previously captured. Given data for a variety of people, we could build a basis of human geometries, including variations such as male vs. female (see Figure 7-1), thin vs. fat, muscular vs. smooth. By interpolating the prototype shapes, we would automatically obtain not only new geometry but also new deformations. This would allow animators to synthesize a wide range of fully-deformable figures.

Recent work in markerless motion capture [8, 24] suggests that we may be able to use our method without requiring special motion-capture equipment. Using a basis of human shapes, we could quickly fit a model to an observed person and use this approximate model for tracking (as in [15, 19]). Throughout tracking, the model could undergo further specialization to the geometry of the subject. Because our geometry has the potential to be substantially more accurate than many of the models previously used for human tracking, there is hope that the tracking quality could be improved.

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.

We hope that research in this direction will continue, making this work a step toward the long-term goal of acquiring, modeling, and animating 3D human geometry.

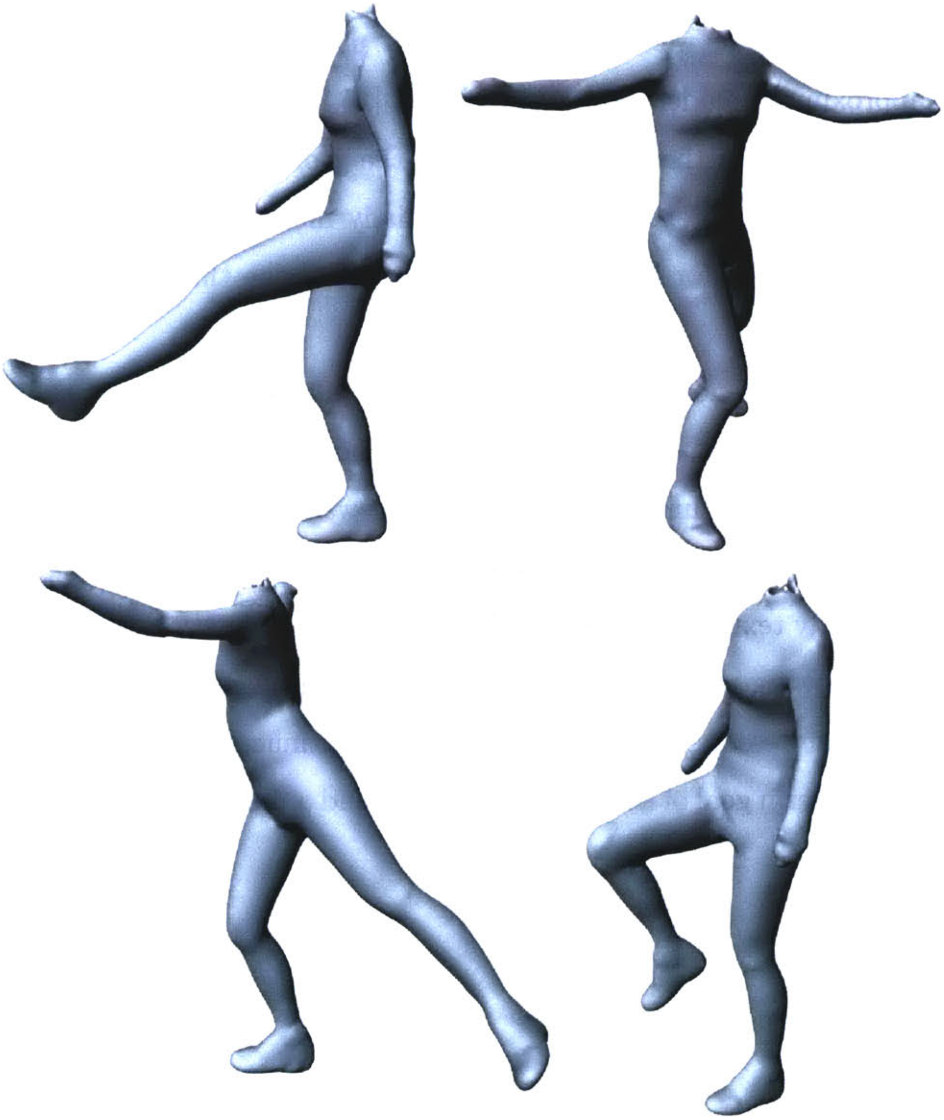

Figure 7-1: This model was generated from a female subject using 5 minutes of motion and silhouettes from three viewpoints. In the future we would like to capture a wide variety of people and interpolate their geometries to synthesize new deformable models.

Bibliography

[1] Brett Allen, Brian Curless, and Zoran Popovic. Articulated body deformation from range scan data. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 612–619, 2002.

[2] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 187–194, 1999.

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

[4] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3D shape from image streams. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages II:690–696, 2000.

[5] D. Broomhead and D. Lowe. Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2(3):321–355, 1988.

[6] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3d objects with radial basis functions. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 67–76, 2001.

[7] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *Proceedings of the Fifth European Conference on Computer Vision (ECCV)*, pages 484–498, 1998.

[8] A. J. Davison, J. Deutscher, and I. D. Reid. Markerless motion capture of complex full-body movement for character animation. In *Proceedings of the Eurographics Workshop on Animation and Simulation*, 2001.

[9] Xianfeng Gu, Steven J. Gortler, Hugues Hoppe, Leonard McMillan, Benedict J. Brown, and Abraham D. Stone. Silhouette mapping. Technical Report TR-1-99, Harvard, 1999.

[10] Brian Guenter, Cindy Grimm, Daniel Wood, Henrique Malvar, and Fredric Pighin. Making faces. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 55–66, 1998.

[11] I. A. Kakadiaris and D. Metaxas. 3d human body model acquisition from multiple views. In *Proceedings of the 5th IEEE International Conference on Computer Vision (ICCV)*, pages 618–623, 1993.

[12] Kaydara. *FiLMBOX Reference Guide.* Kaydara Inc., Montréal, Québec, 2001.

[13] J. P. Lewis, Matt Cordner, and Nickson Fong. 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)*, pages 165–172, 2000.

[14] Wojciech Matusik, Chris Buehler, Ramesh Raskar, Steven J. Gortler, and Leonard McMillan. Image-based visual hulls. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 369–374, 2000.

[15] I. Mikić, M. Trivedi, E. Hunter, and P. Cosman. Human body model acquisition and tracking using voxel data. *International Journal of Computer Vision.* In Press.

[16] Jean-Christophe Nebel, Francisco J. Rodriguez-Miguel, and W. Paul Cockshott. Stroboscopic stereo rangefinder. In *Proceedings of the Third International Conference on 3D Imaging and Modeling*, pages 59–64, 2001.

[17] J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7(4):308–313, 1965.

[18] R. Nevatia and T. O. Binford. Description and recognition of curved objects. *Artificial Intelligence*, 8(1):77–98, 1977.

[19] R. Plänkers and P. Fua. Articulated soft objects for video-based body modeling. In *Proceedings of the 8th IEEE International Conference on Computer Vision (ICCV)*, pages I:394–401, Vancouver, Canada, July 2001.

[20] Szymon Rusinkiewicz, Olaf Hall-Holt, and Marc Levoy. Real-time 3d model acquisition. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 438–446, 2002.

[21] Peter-Pike J. Sloan, III Charles F. Rose, and Michael F. Cohen. Shape by example. In *Proceedings of the 2001 symposium on Interactive 3D Graphics*, pages 135–143, 2001.

[22] Scott Stokdyk, Ken Hahn, Peter Nofz, and Greg Anderson. Spider-man: Behind the mask. Special Session of SIGGRAPH 2002, 2002.

[23] Richard Szeliski and David Tonnesen. Surface modeling with oriented particle systems. In *Proceedings of the 19th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 185–194, 1992.

[24] Christian Theobalt, Marcus Magnor, Pascal Schueler, and Hans-Peter Seidel. 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*, pages 96–103, 2002.

[25] L. Torresani and Chris Bregler. Space-time tracking. In *Proceedings of the 7th European Conference on Computer Vision (ECCV)*, pages 801–812, 2002.

[26] Sundar Vedula, Simon Baker, and Takeo Kanade. Spatio-temporal view interpolation. In *Proceedings of the 13th ACM Eurographics Workshop on Rendering*, pages 65–76, June 2002.

[27] Vicon. *Vicon iQ Reference Manual.* Vicon Motion Systems Inc., Lake Forest, CA, 2003.

[28] Xiaohuan Corina Wang and Cary Phillips. Multi-weight enveloping: least-squares approximation techniques for skin animation. In *Proceedings of the ACM SIGGRAPH symposium on Computer animation*, pages 129–138, 2002.

[29] S. Würmlin, E. Lamboray, O. G. Staadt, and M. H. Gross. 3d video recorder. In *Proceedings of the 10th Pacific Conference on Computer Graphics and Applications*, pages 325–334, 2002.