Detailed Human Shape and Pose from Images

by
Alexandru O. Bălan
B. S., Lafayette College, 2003
B. A., Lafayette College, 2003
Sc. M., Brown University, 2005

Submitted in partial fulfillment of the requirements
for the Degree of Doctor of Philosophy in the
Department of Computer Science at Brown University

Providence, Rhode Island
May 2010

This dissertation by Alexandru O. Bălan is accepted in its present form by the Department of Computer Science as satisfying the dissertation requirement for the degree of Doctor of Philosophy.

Date _____ _____

Michael J. Black, Director

Recommended to the Graduate Council

Date _____ _____

Gabriel Taubin, Reader
Division of Engineering

Date _____ _____

Demetri Terzopoulos, Reader
Computer Science Department, UCLA

Approved by the Graduate Council

Date _____ _____

Sheila Bonde
Dean of the Graduate School

Curriculum Vitae

Alexandru O. Bălan was born on July 17$^{\text{th}}$ 1980 in Bucharest, Romania, where he spent the first 19 years of his life. Son of two computer scientists and brother of another, his choice of a career path in the same direction was somewhat predictable. In 1999 Alexandru Bălan graduated from *Tudor Vianu* computer science high-school in Bucharest and then attended Lafayette College in Easton, PA. Alexandru Bălan was valedictorian of his 2003 graduating class; he received a B.S. degree in computer science and a joint B.A. degree in mathematics and economics. He went on for his Ph.D. studies at Brown University where he enjoyed studying Computer Vision under the direct supervision of Michael J. Black. Before completing the degree in early 2010, he got a taste of doing research in an industry setting too while interning at Intel Research for three summers in a row. His most recent research interests include: geometric shape modeling, shape estimation and registration, 3D structure from images, multi-view vision, 3D photography, and model-based articulated object tracking.

Education

- *Ph.D. in Computer Science*, Brown University, Providence, RI, USA, May 2010.
- *M.S. in Computer Science*, Brown University, Providence, RI, USA, May 2005.
- *B.S. in Computer Science*, Lafayette College, Easton, PA, USA, May 2003.
- *B.A. in Mathematics & Economics*, Lafayette College, Easton, PA, USA, May 2003.

Honors

- *Rosh Fellowship*, Brown University, fall 2006.
- *Paris Kanellakis Fellowship*, Brown University, 2003-2004 academic year.
- *Valedictorian (highest GPA from 655 students)*, Lafayette College, May 2003.
- *Microsoft / UPE Scholarship Award*, Lafayette College, fall 2002.

- *Inducted in Phi Beta Kappa Liberal Arts and Sciences Honor Society*, Lafayette College, spring 2002.
- *President of Lafayette Chapter of Upsilon Pi Epsilon Computing Honor Society*, academic year 2002-2003, inducted in spring 2002.
- *Vice-president of Lafayette Chapter of Association for Computing Machinery*, academic year 2002-2003.
- *Vice-president of Lafayette Chapter of Pi Mu Epsilon Mathematics Honor Society*, academic year 2002-2003, inducted in spring 2001.
- *Inducted in Omicron Delta Upsilon Economics Honor Society*, Lafayette College, spring 2002.

Academic Experience

- Research Assistant, Department of Computer Science, Brown University, 2004-2010.
- Teaching Assistant, Topics in Computer Vision, Brown University, Spring 2009.
- Teaching Assistant, Introduction to Computer Vision, Brown University, Fall 2004.
- Undergraduate Research Assistant, Department of Mathematics, Lafayette College, 2000-2002.

Professional Experience

- Intern at Intel Research, Santa Clara, CA, summers 2005, 2006, 2007.

Peer-reviewed Journal Articles

- Leonid Sigal, Alexandru O. Bălan and Michael J. Black. HumanEva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision*, volume 87, number 1–2, pages 4–27, March 2010. doi:10.1007/s11263-009-0273-6.
- Alexandru O. Bălan and Lorenzo Traldi. Preprocessing MinPaths for sum of disjoint products. *IEEE Transactions on Reliability*, volume 52, number 3, pages 289–295, September 2003. doi:10.1109/TR.2003.816403.

Peer-reviewed Conference Articles

- Peng Guan, Alexander W. Weiss, Alexandru O. Bălan and Michael J. Black. Estimating human shape and pose from a single image. In *IEEE International Conference on Computer Vision*, September 2009.

- Alexandru O. Bălan and Michael J. Black. The naked truth: Estimating body shape under clothing. In *European Conference on Computer Vision*, volume 5303, pages 15–29, October 2008.
- Leonid Sigal, Alexandru O. Bălan and Michael J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Advances in Neural Information Processing Systems, (NIPS 2007)*, volume 20, pages 1337–1344, MIT Press, 2008.
- Alexandru O. Bălan, Michael J. Black, Leonid Sigal and Horst W. Haussecker. Shining a light on human pose: On shadows, shading and the estimation of pose and shape. In *IEEE International Conference on Computer Vision*, October 2007.
- Alexandru O. Bălan, Leonid Sigal, Michael J. Black, James E. Davis and Horst W. Haussecker. Detailed human shape and pose from images. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007.
- Alexandru O. Bălan and Michael J. Black. An adaptive appearance model approach for model-based articulated object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 758–765, June 2006.

Peer-reviewed Workshop Articles

- Alexandru O. Bălan, Leonid Sigal, and Michael J. Black A quantitative evaluation of video-based 3D person tracking. In *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, pages 349-356, October 2005.

Undergraduate Thesis

- Alexandru O. Bălan. An enhanced approach to network reliability using boolean algebra. Honors Thesis, Lafayette College, Easton, PA, May 2003.

Patents

- Michael J. Black, Alexandru O. Bălan, Alexander W. Weiss, Leonid Sigal, Matthew M. Loper and Timothy S. St. Clair. Method and apparatus for estimating body shape. Patent application filed August 14, 2009.

Professional Activities

- Reviewer for MVA (2010), IJCV (2008-2010), PAMI (2009), ICCV (2005, 2007), ECCV (2008), CVPR (2005-2009), NESCAI (2007), IEEE Workshop on Motion (2005).

Acknowledgments

This thesis is the product of many interactions and ideas that were born following collaborations with very creative people to whom I owe sincere appreciation and respect.

First of all, I would like to thank my advisor, Prof. Michael Black, for offering me his guidance throughout my years at Brown. His exemplary ethics, enthusiasm and drive for perfectionism inspired me to mature into a better scholar and to keep being demanding of myself. He taught me the correct research approach and tirelessly instilled confidence in the potential of our ideas. Michael is truly a visionary who knew how to make me see the forest for the trees. He introduced me to the computer vision community and exposed me and my research to top-tier conferences, advising me along the way how to best showcase my work to the community. Michael has been much more than a brilliant advisor: he has been a mentor and a friend who kindly advised and supported me in so many walks of life. He has always showed a fair point of view when I had to make important career decisions. For me he remains not only a professional role model, but also an example of selflessness, modesty and morality. For all of these and for his remarkable patience during this journey, I cannot thank him enough.

I am grateful to my thesis committee: Gabriel Taubin (Brown University) and Demetri Terzopoulos (UCLA). This dissertation has benefited from their critical questions and suggestions, particularly during the thesis proposal. Thank you for pointing my writing into the right direction.

Inspiration does not emerge in a vacuum. Many of my research ideas came to life through external research collaborations. This thesis would not have been possible had it not been for Dragomir Anguelov who invented the SCAPE model, a key ingredient of this thesis. I am also extremely thankful to him for helping me get started on this project by making some of the scan data available to me. I am grateful as well for the three amazing summers I spent as an intern at Intel Research in Santa Clara, CA under Horst Haussecker's supervision. Horst has played a major role in expanding my research interests toward shape modeling and its application to human body estimation from images. Besides Horst, I would like to show my appreciation to all the members of the Applications Research Laboratory at Intel for providing me with a welcoming research environment and insightful discussions: Ara Nefian, Scott Ettinger, Jean-Yves Bouguet, Adam Seeger, Oscar Nestares. I would also like to point out that Scott has been instrumental in rebuilding the SCAPE model during the mesh alignment process described in Section 3.4.1. During my internships, James Davis and his student Steven Scher from UC Santa Cruz also provided me with important background information

that helped me re-create the SCAPE model from scratch and for that I am very thankful.

I would certainly like to thank the members of the SCAPE team at Brown. Not only did they bring valuable contributions to the research on body shape estimation, they also created a very enjoyable working setting. Lots of credits go to Matthew Loper particularly for providing the off-screen rendering functionality, Loretta Reiss for helping with processing the ECCV data, as well as Alexander Weiss, Oren Freifeld, David Hirshberg and Peng Guan for many insightful discussions. Many thanks to Leonid Sigal, with whom I wrote several noteworthy conferences and journal papers. Leon was also a great conversation buddy in the department and during our conferences and internships. I would also like to thank Deqing Sun, Silvia Zuffi and Carleton Coffrin for offering me their constructive input during my thesis proposal and subsequent practice presentations.

The Brown faculty has constantly kept the bar high for academic conversations while at the same time showing an informal, friendly attitude. Chad Jenkins, Gabriel Taubin, Philip Klein, Thomas Dean, Sorin Istrail, John Hughes, thank you for your openness and the productive conversations, for your research suggestions and for offering me insights into your fields of expertise.

I also had the opportunity to meet extraordinary people before coming to Brown. I hold a particular appreciation for Barry McCarthy, my "host father" at Lafayette College, who welcomed me into the American culture and into the warmth of his family at a time when the U.S. was a world too new for me; for Lorenzo Traldi, for the special friendship and the fun we had during the several summers we spent researching network reliability through Boolean algebra; for Chun Wai Liew for initiating me in AI and graphics and for supporting me in my grad school application process.

The Brown CS department is an enjoyable academic environment largely thanks to its students. There are colleagues whom I would like to thank for their friendship and generosity: Victor Naroditskiy, Radu Jianu, Deqing Sun, Stefan Roth, Payman Yadollahpour. You either offered me a hand when needed or you were a great company during the breaks away from the keyboard. I would also like to thank my office mates for the relaxing chats we had during any given day: Eric Rachlin and Fabio Vandin (Fabio, I hope you'll forgive me for my outrageous Italian accent).

Last but in no way least, I am humbly grateful to my roots for making me what I am today. My mother Adina and father Theodor nurtured my passion for math and computers since childhood and later fully supported my decision to apply for colleges in the U.S. My brother Catalin kept the competition and playfulness alive - thank you brother. These acknowledgments would be incomplete if I did not thank the one I am fortunate to call my wife: Adina. Thank you for loving and trusting me, for being my moral support, my friend and for soon becoming the mother of our already dearly loved son.

To my family.

Table of Contents

⋆ Parts of this dissertation are joint work with other authors and have previously appeared in [Bălan *et al.* (2007a,b); Bălan and Black (2008)].

List of Tables

List of Illustrations

Abstract of "Detailed Human Shape and Pose from Images" by Alexandru O. Bălan, Ph.D., Brown University, May 2010.

Automating the process of measuring human shape characteristics and estimating body postures from images is central to many practical applications. While the problem is difficult in general, it can be made tractable by employing simplifying assumptions and relying on domain specific knowledge, or by engineering the environment appropriately.

In this thesis we demonstrate that using a data-driven model of the human body supports the recovery of both human shape and articulated pose from images, and has many benefits over previous body models. Specifically, we represent the body using SCAPE, a low-dimensional, but detailed, parametric model of body shape and pose deformations. We show that the parameters of the SCAPE model can be estimated directly from image data in a variety of imaging conditions and present a series of techniques enabled by this model.

We first consider the case of multiple calibrated and synchronized camera views and assume the subject wears tight-fitting clothing. We define a cost function between image silhouettes and a hypothesized mesh and formulate the problem as an optimization over the body shape and pose parameters. Second, we relax the tight-fitting clothing assumption and develop a robust method that accounts for the fact that observed silhouettes of clothed people provide only weak constraints on the true shape. Our approach is to accumulate many weak silhouette constraints while observing the subject in various poses and combine them with strong constraints from regions detected as skin and with a prior expectation of typical shapes to infer the most likely shape under clothing. Third, we consider scenes with strong lighting and show that a point light source and the shadow of the body cast on the ground provide an additional view equivalent to a silhouette from an actual camera. This approach effectively reduces the number of cameras needed for successful recovery of the body model by taking advantage of the lighting information in the scene. Results on a novel database of thousands of images of clothed and "naked" subjects, as well as sequences from the HumanEva dataset, suggest these methods may be accurate enough for biometric shape analysis in video.

Chapter 1

Introduction

1.1 Thesis Statement

Realistic models of the intrinsic human shape can be estimated directly from images by relying on data-driven deformable shape models.

1.2 Introduction

For an artificial system to perform the high-level tasks of understanding and interacting with the physical world, it needs, among other things, to be able to perceive, represent and reason about its environment. Computer vision uses *visual perception* to observe the world, analogous to the visual system in humans that allows individuals to assimilate information from the environment based on the visible light reaching the eye. The main goal of computer vision is to extract low-level features from images obtained using digital image sensors and infer meaningful properties about objects in the scene that support the high-level tasks like content understanding. To this end, the computer needs to be able to represent and reason about the entities in the scene and their properties. In particular, vision-based capture and analysis of humans and their actions is an active research area and accurate recovery of appropriate shape and pose representations enables many potential applications in surveillance, robotics, entertainment and health-care industries.

1.3 Problem Statement

In this thesis we address the problem of extracting geometric information about the human body from images. Specifically, we are interested in recovering two fundamental properties that are related to physiological and behavioral characteristics of humans, namely *body shape* and *pose*. In a dynamic setting, these are also referred to as motion and shape deformations. Earlier engineering solutions focused on estimating one but not the other, (e. g., marker-based motion capture system for pose

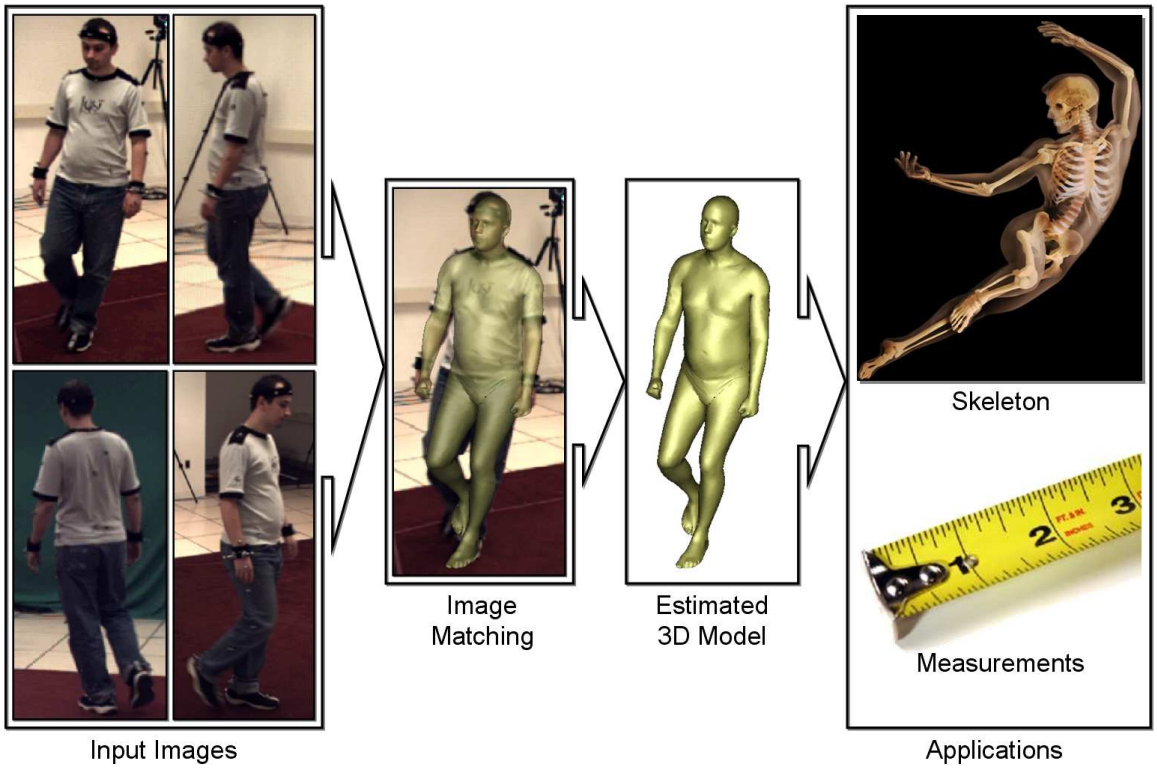


Figure 1.1: **3D Body Model Estimation from Images.** Given image observations of a subject from multiple camera views, a deformable body model is matched to various image features. We recover a detailed three dimensional representation of the body in the form of a triangular mesh which can be used in many ways, including inferring the kinematic structure of the skeleton underneath the skin or extracting biometric measurements from the virtual body. (Top-right illustration is part of the exhibition "The Human Body Revealed" by *Anatomical Travelogue of New York* on display at the National Museum of Health and Medicine in Washington, D.C.).

estimation or 3D scanners for capturing rigid shapes), required specialized hardware and worked in carefully controlled environments. In contrast we propose a computer vision solution that uses images for extracting shape and pose measurements and allows the human subject to move freely in front of the cameras. As illustrated in Figure 1.1, our approach uses multiple camera views to recover a detailed 3D representation of the person. To make this possible, we exploit a statistical model of human form which is capable of adapting to the shape of different people in various poses. This graphics model is learned from a database of human shapes and can be controlled by a relatively small number of parameters. In this thesis we formulate several solutions that estimate these parameters by matching the body model to image features in several imaging conditions. This line of research produces a three dimensional representation of the human body that embodies both shape as well as kinematic information at specific time instances. This representation can be used directly for animations, or indirectly for extracting important body attributes such as shape measurements or the location of the joints in the skeleton.

In this chapter we give a broad overview of the problem of capturing and analyzing the human

form. We motivate the problem and identify the difficulties involved, describe our approach and conclude with a brief presentation of the major contributions of the work, and the structure of the thesis.

1.4 Motivation

To understand what we mean by pose and shape and what are suitable representations for them, we first consider some of the applications.

1.4.1 Applications for Shape and Motion Capture

The entertainment industry uses computer graphics for generating captivating virtual reality content in the form of animated movies, special effects and video games. As such, either or both shape and motion capture technologies are instrumental for achieving realistic animations of virtual human characters (e. g., *The Polar Express* (2004), *Avatar* (2009)). Shape capture can also be used to create avatars inside video games that can be animated using motion capture. Animating a virtual character involves modeling the shape and pose of the character by first designing the surface geometry and shading, then rigging the character with a skeleton for pose editing and finally specifying the motion trajectories for the bones. Motion capture is useful because it is transferable between different characters. Attractive special effects can be achieved when the body shape is reconstructed from a few camera views and rendered from novel views, similar to the *bullet-time effect* in the movie *The Matrix* (1999).

Figure 1.2: **Motion Capture for Animation.** Marker-based motion capture used for animating human-like characters in the movie *The Polar Express* (2004).

1.4.2 Applications for Shape Capture

From an anatomical point of view, people are similar to each other, but at the same time the shape of the human body is very diverse and distinctive. There are many factors that contribute to the body shape variations, including among others, gender (a human observer can easily tell women apart from men based on shape), age (children grow up to be adults and start shrinking as they become old), race (Scandinavians are typically much taller than Asians), and lifestyle (fitness, nutrition). Anthropometry deals with measuring human shape attributes (height, weight, etc.) with the purpose of understanding these types of human physical variation, playing an important role in

industrial design (clothing, vehicles, furniture) and ergonomics.

Modern anthropometric approaches streamline the shape acquisition process by capturing three-dimensional body scans from which measurements can be automatically extracted. This is important for large scale projects. For instance, in an effort to reduce health care costs, Japan is using the waist circumference as an indicator for the *metabolic syndrome* and started in 2008 a massive campaign (Figure 1.3) to acquire measurements from more than 56 million of its population [Onishi (2008)]. It is an open question as to whether more detailed measurements of the body could provide better predictions of health risks.

Figure 1.3: **Metabo.** A Japanese poster that reads "Can you still wear pants that you wore at the age of 20?" promotes awareness of the *metabolic syndrome*, a term that is related to being overweight [Onishi (2008)].

Having access to a dense shape representation can be useful for other medical applications. During a weight-loss program, it is helpful to be able to visualize and monitor changes in body shape over time as well as ensure that the body mass index (BMI) stays within healthy limits. Indicators like BMI can be automatically derived from the captured 3D shape.

Shape capture from video footage is also useful for extracting biometric traits to uniquely recognize humans, which is important for identity access management, forensic analysis and visual surveillance, as well as video search and retrieval applications.

In addition to the aforementioned factors that contribute to the body shape variations, the perceived shape of an individual also changes over time during movement. Capturing the human body in different poses is important in computer graphics for modeling shape deformations due to pose in a data-driven way and facilitating character editing.

1.4.3 Applications for Motion Capture

There is a wide variety of methods for capturing and analyzing the motion of the human body. Capturing motion amounts to capturing pose over time. Motion is a very important cue for activity and gesture recognition such as hand waiving or head nodding or pointing and therefore very useful for human-computer interaction. In robotics, being able to observe and anticipate dynamic human actions is critical for real-time interaction as well as autonomous navigation and obstacle avoidance. Certain types of behavior captured by traits such as gait or typing rhythm can sometimes be used as biometric characteristics for identifying particular individuals. Automated visual surveillance systems detect suspicious human activity in a scene by tracking people's pose and discriminating between normal activity patterns and anomalous activity. Closely related are health-care systems that provide monitoring of elderly people by tracking the patterns of daily activities and signaling

emergency situations. In biomechanics, scientists are interested in capturing human motion for modeling mechanical properties of the bones and soft tissue and their interaction at the joints, which also has medical applications for injury detection, while in sports the goal is to understand athletic performance through modeling and simulation of motion.

1.4.4 Shape and Pose Representation

All these applications have different requirements, which brings the next question: what constitutes a solution to our problem and how do we represent it?

Different levels of abstraction are possible. Shapes can be characterized sparsely in terms of a small set of shape measurements (height, weight), which is sufficient for anthropometric and biometric applications, while more dense representations are needed for character animation. In general, it is useful to think of articulated objects as being a collection of parts with almost rigid shape and connected by joints. Most computer graphics applications see the human body as consisting of an articulated skeleton structure with rigid bones, surrounded by soft tissue (muscle and fat) and covered by the outer skin. In this view, the apparent shape of the human body can simply be described as the topology of the skin surface and is typically represented as meshes or dense point clouds. In contrast, computer vision applications have traditionally modeled the human shape more abstractly using much simpler parametric geometric primitives such as generalized cylinders or superquadrics. As the body moves, the apparent shape changes due to pose variations. It is therefore useful to also envision an intrinsic pose-independent representation of the body shape that is specific to each individual and which remains the same during movement. We address this aspect later on in the thesis.

As far as posture is concerned, it characterizes the skeletal configuration at any given time instance and is defined in terms of the position of the bones and joints. A fine distinction can be made here about using a global coordinate system for all body parts versus a relative representation between consecutive parts. Applications that involve activity recognition typically represent motion in terms of the evolution of the relative joint angles between adjacent parts over time. Sometimes even recovering 2D joint angles in the image space is sufficient for recognizing some activities. In contrast, being able to recover 3D joint positions can prove valuable for a robotic system to physically interact with humans.

1.4.5 Why Vision-based?

Our goal is to recover dense body shape and pose information directly from images. This is in sharp contrast with the state of the art motion capture system that uses markers near each joint to identify the motion by the positions or angles between the markers. Marker-based motion capture can have sub-pixel accuracy, works in real-time and is often taken as the gold-standard. However, it only works in controlled environments and requires tedious placement of markers on the subject's body. Video-based motion capture approaches often provide the only non-invasive solution,

motivating the approach we take in this thesis.

1.5 Challenges/Difficulties

In its most general form, the problem of estimating pose or shape is severely under-constrained due to many factors such as: the image formation process, occlusions, changes in appearance, and the complexity of the human body structure itself. We briefly review each below.

Image capture. From a geometric point of view, an image is a projection of the 3D world onto a 2D image plane. As such, explicit depth information is lost during the imaging process. From a technological point of view, various settings for the digital camera used for capture also limit and degrade image quality (e. g., image resolution, motion blur, lens distortions, image noise, etc.). All these imaging factors cause ambiguities in matching image features to 3D surface points.

Occlusions. In a single image, at least half of the body is not visible due to self occlusion, where the side facing the camera occludes the side away from the camera. Given the highly articulated nature of the human body, body parts tend to be occluded by other parts. Clothing as well as other objects present in the scene also pose problems by obscuring the body shape.

Appearance changes. As people move in the scene with respect to the camera, the surface appearance changes as well due to motion, clothing, viewpoint or lighting.

High-dimensional search space. The complexity of the human kinematic structure and the large variability in body shape between individuals imply there are many parameters that need to be estimated. When defining the problem as an optimization of an objective function over the model parameters, the search space becomes very high dimensional and needs to be explored efficiently to get as close as possible to the global optimum.

These difficulties can be overcome by employing simplifying assumptions and domain specific knowledge, or by engineering the environment appropriately. In certain scenarios, we can control the lighting and camera placement, we can use multiple cameras to reduce depth ambiguities, and/or we can require the subject to wear tight-fitting clothing.

1.6 Previous Approaches

Given the wide array of potential applications for markerless motion capture for animation, gait analysis, action recognition, surveillance and human-computer interaction, this problem has received broad attention from the vision community. On the other hand, recovering the shape has not really been addressed until recently. We argue that there are many motivations for recovering body shape information simultaneously with pose. For some graphics applications, having direct access to the shape model for a particular subject removes an additional step of mapping kinematic motions to

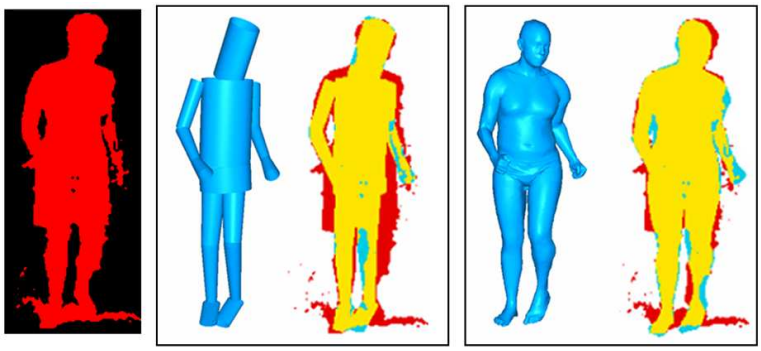


Figure 1.4: **Use of a Body Model to Explain Image Evidence.** Two body models are being matched to the image silhouette shown in red. The parameters of the model are optimized to maximize the overlap area shown in yellow between the projected body model and the image silhouette. Because of the shape mis-match, an ambiguity remains in placing the cylinders of the left model inside the image silhouette. The body model on the right explains the image evidence much better given its shape.

3D models. By recovering a shape model that more closely matches the image observations, it can even make pose estimation more robust (Chapter 4).

Much of the recent work on human pose estimation and tracking exploits Bayesian methods which require generative models of image structure. Most of these models, however, are quite crude and, for example, model the human body as an articulated tree of simple geometric primitives [Gavrila and Davis (1996); Pentland and Horowitz (1991); Sminchisescu and Triggs (2003); Terzopoulos and Metaxas (1991)]. For instance, a model based on cylindrical body parts proposed by Marr and Nishihara in 1978 [Marr and Nishihara (1978)] is still used today in human tracking applications [Deutscher and Reid (2005)]. In a generative framework, methods make use of the model to adjust the joint angles between the rigid parts to make them align with the image features. Arguably these generative models are a poor representation of human shape because they do not explain the image evidence very well, leaving room for extra ambiguities. Figure 1.4 illustrates one such ambiguity when trying to match different body types to image silhouettes. This ambiguity can be reduced if the shape of body model better conforms to the shape of the observed person. Moreover, because these models are typically designed with only posture estimation in mind, they are too simplistic to be suitable for shape estimation.

Based on the hypothesis that using more realistic shape models can lead to better tracking, more recent approaches [Mündermann *et al.* (2007); Rosenhahn *et al.* (2006)] have used body models obtained from 3D laser scans. They assume the body shape has been estimated *a priori* and consider only articulated deformations, effectively chopping the laser scan into body parts and using the rigid segments instead of cylinders to perform pose estimation. They keep the shape of the parts fixed during tracking and ignore non-rigid deformations close to the joints.

In the graphics community however, the emphasis is put on photo-realistic display of human performances. As such, they require dense faithful reconstruction of surface geometry and motion

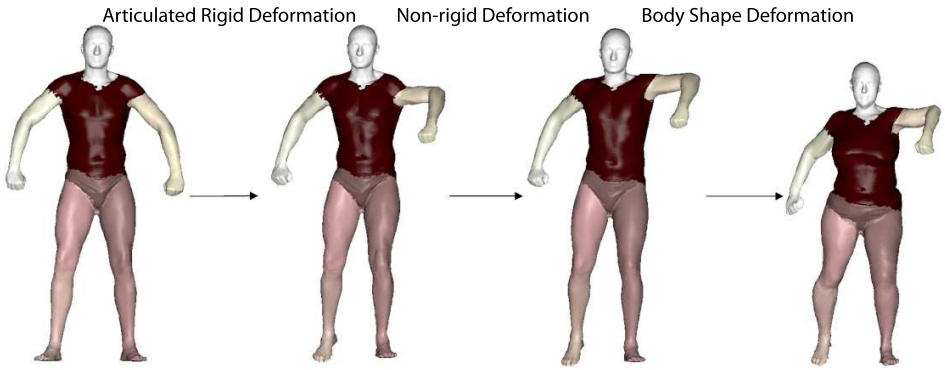


Figure 1.5: **SCAPE Deformation Process.** The SCAPE model is a deformable body model that admits a low dimensional parameterization of shape and pose. Starting from a reference mesh shown on the left, we can change the pose by specifying new rigid orientations for each of the 15 body parts. After the model automatically corrects for non-rigid deformations, we control the shape by adjusting a few shape parameters.

[de Aguiar *et al.* (2008); Vlasic *et al.* (2008)]. One way to make the problem practical is to use a detailed template model, typically obtained by laser scanning the subject prior to the capture. User assistance is often considered acceptable in applications targeted to movie production and animation. Other techniques are able to densely estimate shape of arbitrary objects without employing any knowledge about the object being scanned. Approaches include structured light, laser scanning, volumetric reconstructions from silhouettes, and stereo matching. Being free of a model allows for generalizations to arbitrary shapes (e. g. garments) to be captured; on the other hand, this non-parametric representation lacks robustness to noise in the image observations and makes it unsuitable for making any inferences about the articulated pose without an associated kinematic skeleton.

1.7 Proposed Approach

As an alternative, we propose the use of a deformable graphics model of human shape that is learned from a database of detailed 3D range scans of multiple people. Specifically we use the SCAPE (Shape Completion and Animation of PEople) model [Anguelov *et al.* (2005b)] which represents both articulated and non-rigid deformations of the human body. This model is also capable of adapting to the shape of previously unseen subjects. SCAPE can be thought of as having two components. The pose deformation model captures how the body shape of a person varies as a function of their pose. For example, this can model the bulging of a bicep or calf muscle as the elbow or knee joint varies. The second component is a shape deformation model which captures the variability in body shape across people using a low-dimensional linear representation. These two models are learned from examples and consequently capture a rich and natural range of body shapes, and provide a more detailed 3D triangulated mesh model of the human body than previous models used in video-based pose estimation. We illustrate the SCAPE deformation model in Figure 1.5.

The model has many advantages over previous deformable body models used in computer vision. In particular, since it is learned from a database of human shapes it captures the correlations between the sizes and shapes of different body parts. It also captures a wide range of human forms and shape deformations due to pose. Modeling how the shape varies with pose reduces problems of other approaches associated with modeling the body shape at the joints between parts. In contrast with earlier methods that only estimate pose, learning a low dimensional shape model decoupled from pose means that the number of total parameters increases by as few as six coefficients. These advantages come at the expense of increased computational complexity and being able to only represent naked bodies.

We claim such a deformable model of human shape can be estimated directly from images in a variety of imaging conditions. The model is capable of adapting to the shape of previously unseen subjects in various poses. This eliminates the need for building subject-specific body models *a priori*. In addition to recovering shape, such a detailed body model can play a central role in improving the reliability of pose inference by being able to more closely match image observations. Even when the body is occluded by normal clothing, constraints from multiple camera views and poses, and from bare skin regions in the images can be combined to infer the most likely shape model that lies under the clothes. Furthermore, such a detailed shape model allows us to exploit additional image cues such as shadows to more robustly estimate shape and pose. Lastly, the recovered shape model can be used for gender classification as well as for extracting anthropometric measurements.

1.8 Contributions

In this thesis we have developed a collection of techniques to recover a deformable body model in a variety of imaging conditions.

1. We have developed a method to recover the shape and pose of a person using multiple calibrated and synchronized camera views. The method relies on silhouette matching and assumes the subject is wearing tight-fitting clothing (Chapter 4).

2. We have developed a method that is robust to strong lighting present in the scene. Rather than causing problems, we find that we can take advantage of cast shadows to more robustly estimate the pose and shape of a person. We rely on the concept of a "shadow camera" illustrated in Figure 1.6 that consists of one or more point light sources and a ground surface on which shadows are cast. This effectively allows us to reduce the number of real cameras and enables monocular pose and shape estimation (Chapter 5).

3. We have developed a method to infer the most likely shape of a person wearing clothing. Unlike the graphics techniques that specifically target the capture of the garment surface, our goal is to actually infer the intrinsic human body shape. The method relies on multiple calibrated and synchronized camera views and defines an image matching function that is robust to clothing. It integrates body shape constancy constraints across pose with a generalization of visual hulls

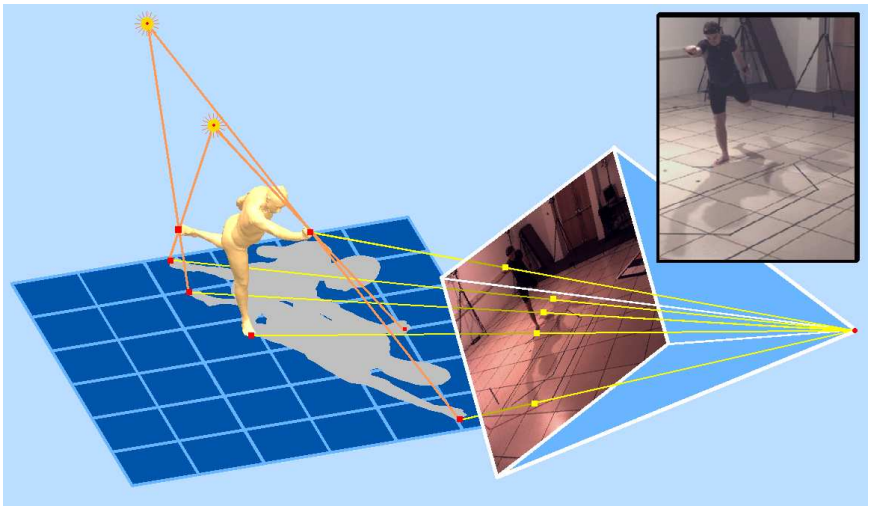


Figure 1.6: **The Shadow Camera.** Shadows cast on the ground may reveal structure not directly observed on an object such as the right arm or the left leg of the person in the image. The lights together with the ground plane act like another camera view providing an additional silhouette of the object.

to account for clothing (Figure 1.7) as well as tight constraints in regions detected as bare skin (Chapter 6).

1.9 Thesis Outline

Chapter 1. Introduction. Thesis statement, motivation, challenges and contributions.

Chapter 2. State of the Art. We provide a brief overview of the state of the art of the field of human shape and motion capture, concentrating on the type of body models used in the computer vision and computer graphics literature.

Chapter 3. SCAPE: A Deformable Body Model of Shape and Pose. Central to this thesis is a recently proposed deformable body model called SCAPE that is more suitable for analyzing human activity in images than traditional models. The SCAPE model is reviewed and our specific implementation is described in detail.

Chapter 4. A Framework for Model Fitting to Images. This chapter describes our basic approach to estimating the pose and shape of a person from multiple images.

Chapter 5. Shape from Shadows. Strong lighting typically causes severe changes in appearance and is seen as a nuisance for image understanding. In this chapter we show that rather than causing problems, strong lighting can be exploited to improve human pose and shape estimation. Here we concentrate on cast shadows.

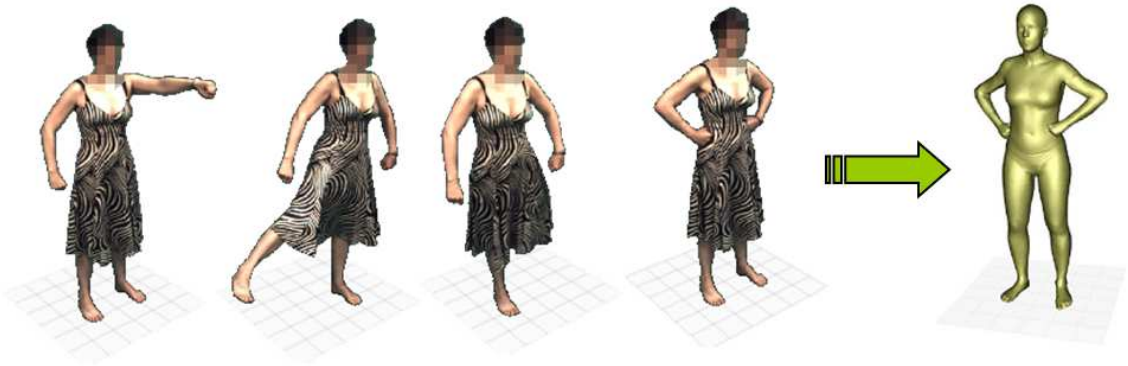


Figure 1.7: **Shape Under Clothing.** One way to improve the reliability of shape estimation in the presence of clothing is to observe a human subject in multiple poses, thereby deriving multiple pose-dependent constraints on body shape that can be combined to infer the most likely shape under clothing.

Chapter 6. Shape under Clothing. In this chapter we relax the assumption that clothes are tight-fitting and generalize the image matching formulation to handle loose clothing.

Chapter 7. Conclusions. We summarize the contributions of the thesis and propose extensions and directions for future research.

1.10 List of Published Papers

The thesis is based on material from the following published papers, listed in the order of relevance:

Alexandru O. Bălan, Leonid Sigal, Michael J. Black, James E. Davis and Horst W. Haussecker. Detailed human shape and pose from images. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007.

Alexandru O. Bălan and Michael J. Black. The naked truth: Estimating body shape under clothing. In *European Conference on Computer Vision*, volume 5303, pages 15–29, October 2008.

Alexandru O. Bălan, Michael J. Black, Leonid Sigal and Horst W. Haussecker. Shining a light on human pose: On shadows, shading and the estimation of pose and shape. In *IEEE International Conference on Computer Vision*, October 2007.

Chapter 2

State of the Art: A Review

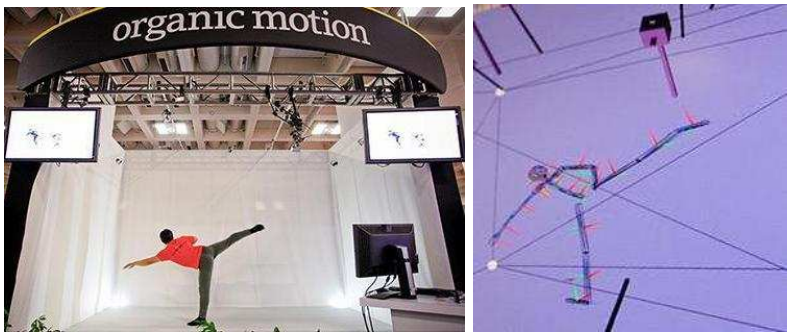
2.1 Introduction

This chapter gives a brief overview of the state of the art of the field of human shape and motion capture. Traditionally, human motion capture has been considered separately from shape capture, mainly because most viable approaches were only capable of recovering one or the other, and the technologies employed were very different. Early successful motion capture systems assumed that the shape of the human subject was known *a priori*, that the subject was wearing tight-fitting clothing, and that the body parts were moving rigidly with respect to each other. Interestingly, achieving the highest level of accuracy meant that a marker-based system had to be used in which the human subject was required to wear markers attached to the body. Such systems have difficulties performing motion capture in the presence of loose garments, and do not reconstruct realistic and detailed body shape models for unknown subjects. In contrast, existing 3D body scanners are designed to work with static objects. They capture fine details of the whole human body surface and appearance, but it can take 15 seconds to acquire a complete scan. While this makes it possible to capture the shape of people with arbitrary clothing in fixed poses, 3D body scanners cannot capture dynamic events such as the motion of the skeleton or the clothes.

Recent advances in 2D and 3D image capture and processing technologies have brought about a paradigm shift in the state of the art of motion capture, with new applications becoming possible in recent years. First, the advent of 3D whole body scanning has enabled modeling and animation of the human body by example. Such a data-driven approach can be used to recover realistic body shapes from image data during motion capture. Second, it is now possible to record high-resolution images from many cameras at high frame-rate with hardware synchronization, making it possible to capture not only the articulated pose, but also detailed appearance, dynamics, and fine geometric details of the skin or garments, in a process that is now being referred to as *human performance capture*.

(a) Gypsy Gyro - Meta Motion™

(b) Marker-based Motion Capture



(c) Marker-less Motion Capture - Organic Motion™

Figure 2.1: **Motion Capture Technologies.** (a) A portable and relatively inexpensive inertial motion capture system. (b) Marker-based optical motion capture for character animation (Reprinted from Wikipedia – Activemarker2.png; accessed December 14, 2009). (c) An image-based motion capture system.

2.1.1 Commercial Technologies

Motion Capture

Most existing commercial systems for motion capture (often referred to as *MoCap*) employ either active or passive sensing devices to track spatial coordinates of segments or joints over time (e. g., Vicon™, Motion Analysis™, Meta Motion™). Active sensing usually involves attaching active (mechanic, electro-magnetic or acoustic sensors, as well as accelerometers) or pseudo-passive (reflective markers) devices to the moving body parts, from which some kind of a signal is obtained that directly relates to the relative configuration, motion acceleration or 3D location (though triangulation) of the devices, requiring minimal post-processing for inferring joint motion (cf. Figure 2.1a,b). Because active sensing MoCap can generally provide sub-millimeter precision at high frame-rate (120fps and above), it is used in medical, bio-mechanics and sports applications for measuring and analyzing motion patterns of humans, as well as Human Computer Interaction (HCI) and animation of virtual characters.

The downside of marker-based motion capture is that the devices the subject has to wear are usually cumbersome and hamper movement, and are even impossible to wear in certain scenarios.

This makes marker-less motion capture approaches more attractive and general, enabling many other applications like visual surveillance or gesture recognition. While this is a desirable goal, the problem of marker-less motion capture is significantly more difficult because it relies on inherently ambiguous images from one or more cameras to capture the motion. As such, marker-less motion capture solutions are only slowly emerging in recent years (e. g., Organic Motion^{TM}, Figure 2.1c).

Shape Capture

Successful shape capture technologies are designed for capturing shape information and possibly the appearance (i. e. color), mostly of generic objects. 3D scanners can also be classified as passive or active, depending on whether they use images as the only source of information or require additional specialized emitting and sensing hardware.

Active 3D scanners emit some kind of radiation (laser rays, structured light pattern, ultrasound or X-ray) and detect the distorted reflection off the object in order to infer shape. A single scan typically produces only a partial view, requiring multiple scans from different views that subsequently need to be merged into a common coordinate system. Due to the interference between radiation emitted from multiple directions, complete shape reconstructions are difficult to achieve in real time. The shape is recovered as a point cloud of the outer surface or a depth map from which a complete surface mesh can potentially be reconstructed though interpolation.

While active scanning techniques produce high resolution shape estimates, most of them are expensive, require specialized hardware in a controlled environment, can only produce partial views in real-time, and cannot capture generic dynamic objects in 360°. They also do not provide any temporal surface correspondences, necessary for shape editing or texture-mapping. Pattern projection techniques often impose restrictions on the surface reflective properties or color, encountering difficulties with shiny, mirroring, transparent or dark objects unless they are coated with some kind of white powder. Since active 3D scanners produce very detailed shape reconstructions, they are extensively used for graphics applications, character modeling and rendering, as well as for industrial design.

A solution to capturing complete shapes in real-time is to use vision-based passive scanning techniques which are not as accurate. Passive scanners do not emit any kind of radiation themselves, but instead rely on detecting reflected ambient radiation (i. e. visible light) simply using regular cameras, making such systems much cheaper. In computer vision, the techniques to recover shape are called Shape-from-X techniques, where X can be silhouettes, stereo, motion, texture, shading, focus etc. Many of these techniques are particularly important in the single camera case where the problem of reconstructing shape is not well constrained. Currently no complete solution exists to the single-camera 3D shape extraction problem. In the multi-camera case, which is more popular, typical approaches work in carefully controlled settings. They include shape-from-silhouette and multi-view stereo techniques that reconstruct complete 3D object models using volume intersection or ray intersection based a collection of images taken from known camera viewpoints. These techniques are accurate enough for free view-point re-animations and, when combined with a reference body model, enable capturing human performance.

2.1.2 Vision-based Human Shape and Motion Capture

Vision-based human motion capture is a well established and very active field with a long history motivated by real-world applications like automatic image understanding. Numerous surveys have been written that span the period 1978-2007 [Gavrila and Davis (1996); Moeslund and Granum (2001); Moeslund *et al.* (2006)] and have focused on image detection and tracking of people as well as articulated motion estimation, analysis and recognition.

In the last decade, the field has evolved to include not only kinematic pose estimation, but also the closely related problems of acquiring a human body model of shape from images, useful for extracting anthropometric measurements for the clothing industry, as well as capturing human performance, in the form of detailed appearance, dynamics, and fine geometric details of the skin or garments. While no surveys exist that focus on extracting human body shape directly from images, there are some that analyze modeling of virtual humans and clothing [Magnenat-Thalmann *et al.* (2004)] or time-varying scene capture technologies [Stoykova *et al.* (2007)].

2.2 Analysis by Synthesis

Most motion-capture approaches employ an analysis by synthesis framework which will be the focus of our review. The problem is framed as an optimization of an objective function over some model parameters. The objective function measures the similarity between the features extracted from image observations (image evidence) and a reconstruction given a set of parameters controlling things like a parametric model of the human body, a camera projection model, and other models of entities in the scene (background, lighting, etc.).

Some of these parameters can be known in advance, given a calibrated camera or a known subject (dimensions of body parts) or *a priori* knowledge about the range of motion. Having a smaller set of unknown parameters makes the estimation more tractable but also imposes limitations on the visual input that can be appropriately analyzed. On the other hand, as the number of unknown parameters increases the problem quickly becomes severely under-constrained.

For image-based human shape and motion capture, a parametric model of the human body is needed. It needs to be adjustable in terms of kinematics (articulated pose) and body dimensions (shape). Different levels of complexity exist for designing body models depending on the application. For pose estimation only, simplistic (approximate) body models based on simple geometric primitives are generally sufficient and computationally more efficient. For virtual reality animations and for accurate dense shape measurements, more realistic humanoid models are often necessary.

2.3 Human Body Models

Human body models describe both the kinematic properties of the body (skeleton) and the shape (human tissue/ the flesh and skin, and sometimes the clothing). In some scenarios like virtual reality applications an appearance is also defined.

2.3.1 Kinematic Models

Kinematic models are almost always represented as kinematic trees consisting of segments that are linked by joints, although for some vision applications that perform bottom-up articulated pose reconstruction it is sometimes convenient to represent the kinematic structure of the body as a collection of individual body parts with soft constraints between joints.

The pose parameters of a 3D kinematic tree are given by the position and orientation of the root joint in the world coordinate system, as well as the joint angles between adjacent parts encoding the orientation of a part in the coordinate system of its parent (Figure 2.2). In general a joint can have 3 degrees of freedom (DOFs) like a ball-and-socket joint, but it is common to model special types of joints with fewer DOFs: like modeling the knee as a hinge joint with 1 DOF. Overall the dimensionality of the pose parameters can vary anywhere from 25-60 DOFs. Note that this simple model of relative joint rotations does not easily extend to model more complex joints such as the shoulder or structures such as the neck/spine. Equivalent 2D planar models have also been defined to be used in the image domain with a single DOF for each joint.

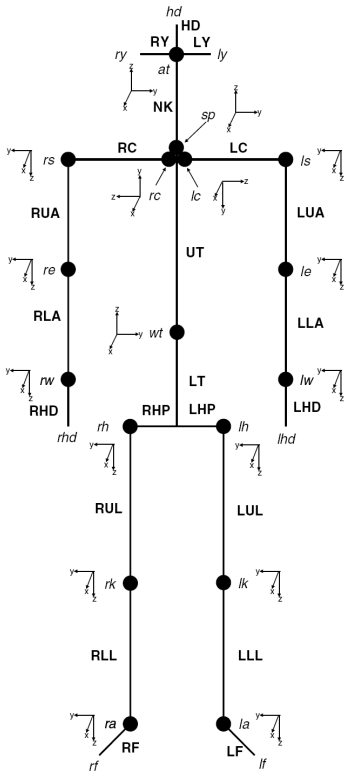
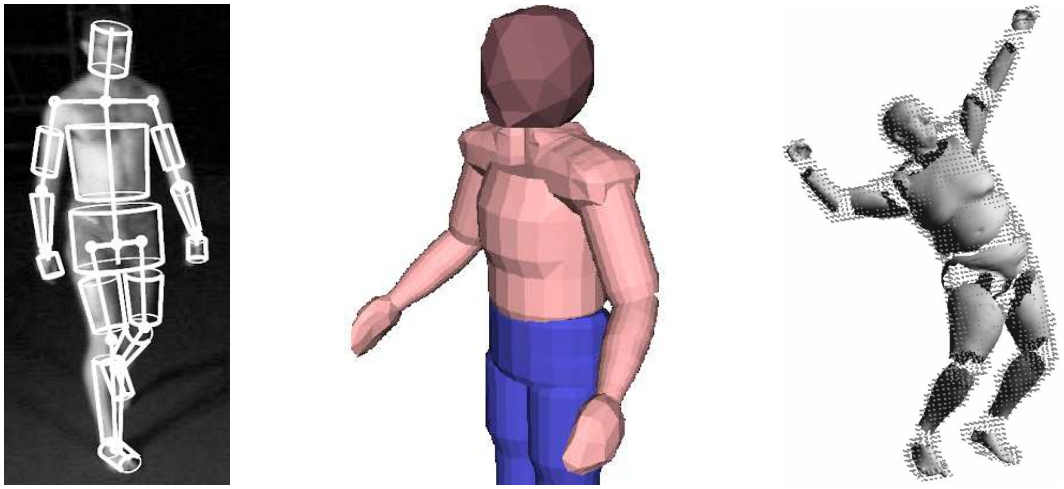


Figure 2.2: **Kinematic Articulated Model.** The skeleton of a person can be represented as a kinematic tree with bone segments linked by joints. Each bone has associated with it a local coordinate system. The relative rotations between consecutive bone segments are expressed using joint angles. (Reprinted from [Barrón and Kakadiaris (2003)])

Many parameterizations are possible for representing 3D joint angles including rotation matrices, Euler angles, quaternions and exponential maps (see Appendix B). Each suffers from different limitations (many-to-1 representations, excess dimensionality, instability due to the 'Gimbal lock') and the choice often depends on the optimization strategy.

2.3.2 Shape Models

A variety of 2D and 3D models have been proposed to approximate a subject's shape. In early vision approaches, 2D models based on quadrilateral or elliptical patches to approximate body parts and 2.5D models that add a depth ordering of the part patches to handle self-occlusions have proved



(a) (b) (c)

Figure 2.3: **Part-based Body Models.** (a) Body model built using tapered cylinders. (Reprinted from [Deutscher and Reid (2005)]). (b) Body model built using superquadrics. (Reprinted from [Grest *et al.* (2005)]). (c) Laser-scanned body model partitioned into rigid body parts and used for pose estimation (Reprinted from [Mündermann *et al.* (2007)]).

effective for surveillance applications that sought only an approximate pose estimate and where the motion was mostly parallel to the image plane. Inherently however such 2D models live in the image space and are not appropriate for recovering volumetric shape measurements or for reasoning about 3D events like inter-penetration the way actual 3D models are. Based on the level of complexity, 3D shape body models can be classified as part-based and whole-body shape models.

Part-based Shape Models

Part-based shape models represent each part as a rigid shape attached to a joint of the kinematic tree. These models are easy to animate and have been used successfully for articulated human body pose estimation and tracking. Most commonly used representations include simple geometric primitives like cylinders, truncated cones or ellipsoids (e. g. [Deutscher and Reid (2005)]), include some scaling parameters to make it fit different human shapes, and are typically built by hand (e. g. [Pentland and Horowitz (1991)]). In particular, the use of generalized cylinders for representing shapes have been proposed as early as the 70's by Nevatia and Binford (1973) and Marr and Nishihara (1978). Such models are too simplistic to fit the body shape well (see Figure 2.3a).

More complex parametric shapes like superquadrics [Gavrila and Davis (1996); Grest *et al.* (2005); Pentland and Horowitz (1991); Sminchisescu and Triggs (2003); Terzopoulos and Metaxas (1991)] are still too crude to allow for precise recovery of both shape and pose (e. g. Figure 2.3b). Detailed but fixed, subject-specific, laser-scanned body parts represented as free-form surfaces or polygonal meshes have also been used for motion estimation [Mündermann *et al.* (2006, 2007); Rosenhahn *et al.* (2007)] (e. g. Figure 2.3c).

Part-based shape models do not deform during motion, rigidly moving together with the skeleton segment; as such, they introduce artifacts at the joints where the surface geometry is not modeled.

The scaling parameters of the shape model (such as the limb lengths and widths) are often assumed known [Deutscher and Reid (2005); Mündermann *et al.* (2006, 2007)] or are estimated in calibration phase prior to tracking [Gavrila and Davis (1996); Grest *et al.* (2005); Rosenhahn *et al.* (2006); Sminchisescu and Triggs (2003)] by having the subject assume a set of pre-defined canonical poses.

Whole-body Shape Models

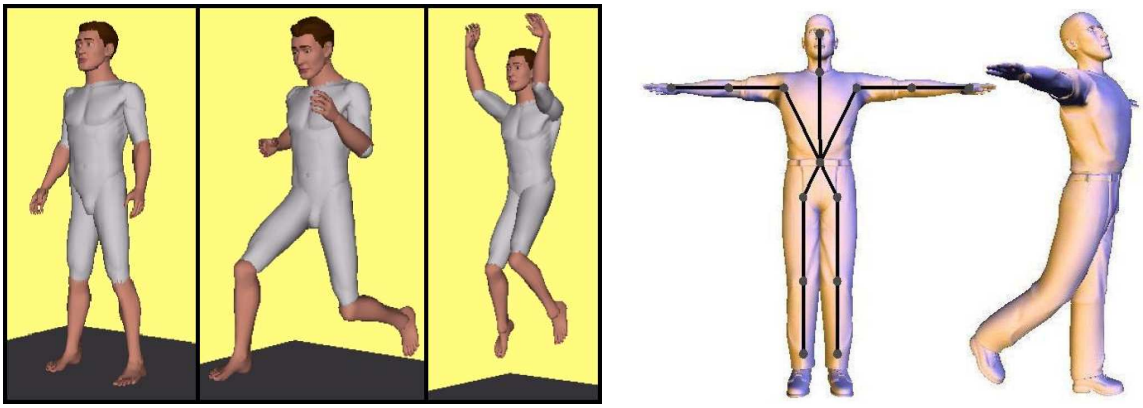
Realistic body models can be designed by modeling the shape as a single deformable surface for the entire body that avoids discontinuities at the joints. Typically represented as a mesh of polygons, fine anatomic details of the skin can be captured.

Whole-body shape models have originally been developed in the computer graphics (CG) community for animations and virtual reality applications. While more realistic, these models are more complex to design and animate. Given the recent computing power advances, humanoid models are slowly being adopted by the computer vision community for improving the capture of motion and shape of humans from images. We identify three categories of CG models that can be animated: surface-based, anatomically-based, and data-driven statistical models.

Skeleton-driven Surface-based Models

Classical surface-based modeling for animation is done in two stages. First, the surface geometry needs to be modeled. When designed by a graphics artist using specialized software (e. g. 3D Studio Max™ - Autodesk, BodyBuilder™ - Vicon, Maya™ - Alias Wavefront, Poser™ - Smith Micro Software), creative characters emerge, but often look cartoon-ish. In contrast, reconstructive approaches build the 3D geometry automatically by capturing the shape of real people using 3D scanners. The second stage of the process is called *rigging* or *skinning*, in which the skin is fitted with a skeleton for synthesizing skin deformations due to articulated motion. Each vertex of the mesh is assigned to one or more affecting bones with corresponding weights. As the bones are transformed, the vertices move in order to stay aligned with them, effectively following a weighted interpolation of rigid transformations. Arguably the most widely used technique for skeleton-driven skin deformations is linear blend skinning and its variants, spherical and dual-quaternion blend skinning; they also come under various other names such as joint-dependent local deformations (JLD), sub-space deformations (SSD), skeleton-driven deformations (SDD) and enveloping. The increased realism comes from creating associations that allow for surface vertices to be influenced by more than one joint. While such procedural methods are computationally efficient, they are notorious for generating not-always-anatomically-correct deformations, including collapsing-joints artifacts and the absence of muscle bulging.

The creation of several standards (VRML, BVH, X3D, MPEG-4, H-Anim) in the CG community for representing animated humanoids (both the joint hierarchy and the surface geometry) has



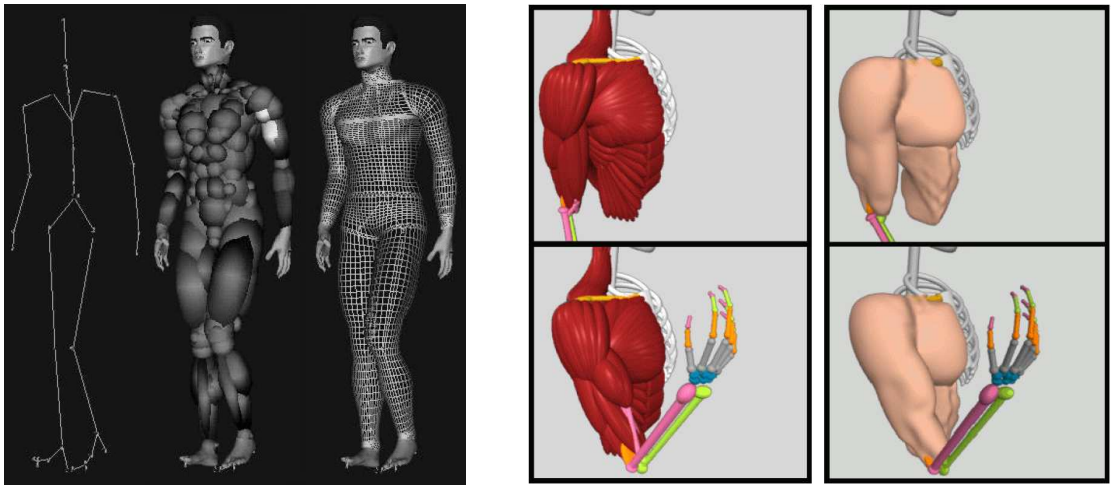
(a) (b)

Figure 2.4: **Skeleton-driven Surface-based Models.** (a) Generic humanoid model defined using the H-Anim standard and animated using linear blend skinning (Reprinted from [Hilton *et al.* (2000)]). (b) Artist-designed model from Poser^{TM} and rigged with a skeleton for animation (Reprinted from [Kehl *et al.* (2005)]).

facilitated the adoption of artist-designed models by the computer vision community to be used for capturing human shape and motion from images. For example, Kehl *et al.* (2005) use a body model exported from Poser^{TM} with $22,000$ vertices, rigged with a skeleton exhibiting 24 DOFs, and employing linear blend skinning deformations (see Figure 2.4b). The model is fitted to a volumetric reconstruction from up to 16 silhouettes. Estimates of shape and initial pose are obtained in a calibration stage in which the user adopts a 'Da Vinci' pose. Statistics of ratios between different limb lengths are also utilized. Closely related is the work of Hilton *et al.* (1999, 2000) who use a VRML body model, as illustrated in Figure 2.4a. Their approach requires the subject to stand in a known pose for the purpose of extracting key features from their silhouette contour which allows alignment with the 3D model. Vlasic *et al.* (2008) employ a laser-scan of the subject that is initially deformed using linear-blend skinning based on a kinematic skeleton fitted into visual hulls. The shape is subsequently refined by letting the vertices move toward the contours of the silhouettes. An extension of the method is provided by Gall *et al.* (2009).

Anatomically-based Models

Anatomically-based models provide an approximation of the components inside the body like major bones, muscle and fat tissue and their dynamics. During movement, the deformation of the interior structures induces a corresponding deformation of the skin that is wrapped over them. An example of this is given by Plänkers and Fua (2001a,b, 2003) who define a complex body model consisting of 3 layers: a kinematic tree model, soft objects ("meta-balls") attached to the skeleton structure and a polygonal skin surface (see Figure 2.5a). Having these "meta-balls" expressed as field functions based on 3D Gaussian blobs makes the skin be defined implicitly as a level set surface. This approach eliminates surface discontinuities at joint locations, but requires the relative shapes



(a) (b)

Figure 2.5: **Anatomically-based Body Models.** (a) Muscle and fat tissue are simulated using simple volumetric primitives (meta-balls) and attached to a skeleton to induce an implicit skin surface (Reprinted from [Plänkers and Fua (2003)]). (b) Physical simulation of a contracting muscle together with the induced skin deformation (Reprinted from [Scheepers *et al.* (1997)]).

and locations of these "meta-balls" be defined *a priori* by hand. Their relative scale is based on an estimated length and width of the corresponding limb. An iterative optimization method is proposed to fit each limb segment to silhouette and stereo data, constraining the left and right limbs to have the same measurements.

More complex, physically realistic, body models have been proposed [Aubel and Thalmann (2001); Dong *et al.* (2002); Scheepers *et al.* (1997); Shen and Thalmann (1995); Wilhelms and Van Gelder (1997)], performing multi-layered physical simulations of muscles contracting or skin stretching (e. g., Figure 2.5b). These models are tedious to design even by a modeling expert, requiring considerable user intervention, and are computationally expensive to simulate.

Example-based Statistical Body Models

With the advent of 3D whole body scanner technologies, example-based techniques for modeling human shape have become increasingly popular for generating and animating realistic human models. By relying on scan data and some interpolation scheme, realistic new shape models can be generated efficiently and without (significant) user intervention. Data-driven modeling can be used either for generating realistic body shapes [Allen *et al.* (2003); Seo and Magnenat-Thalmann (2003); Seo *et al.* (2003); Seo and Magnenat-Thalmann (2004)] or for predicting realistic skin deformations [Allen *et al.* (2002); Wang *et al.* (2007); Weber *et al.* (2007)], or both [Anguelov *et al.* (2005b); Hasler *et al.* (2009b)].

Two stages can be identified: (1) a training stage in which the scans are acquired, pre-processed and a model of deformations is learned, and (2) a morphing stage in which new examples are

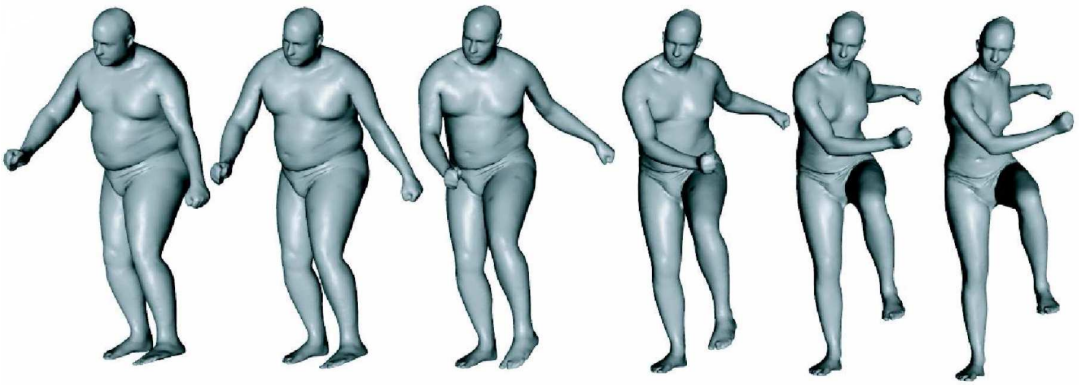


Figure 2.6: **Example-based Statistical Body Models.** Animation of the SCAPE model jointly varying body shape and pose. SCAPE uses a deformation model of shape and pose learned from example body scans. (Reprinted from [Anguelov *et al.* (2005b)])

generated by leveraging the learned deformation model. The key to learning a deformation model is bringing all scans into full correspondence using a process called registration or alignment. This is typically done by non-rigidly deforming a shape template mesh (also called the shape reference mesh) to match each of the scanned shapes [Allen *et al.* (2003); Anguelov *et al.* (2005a); Seo and Magnenat-Thalmann (2003)]. The output is a mapping between every point on the template surface and its corresponding point in the scan. The template can be artist-designed or a cleaned-up scan that may be rigged with a skeleton. The registration may take as input several sparse corresponding features between the scans and the template to facilitate the process and obtain more accurate alignments.

Having the scans in full correspondence has many benefits, including being able to interpolate between existing shapes or learning a statistical model that captures correlations of shape deformations between different body parts as well as correlations between articulated poses and skin deformations. For example, the SCAPE model proposed by Anguelov *et al.* (2005b) follows this approach. As illustrated in Figure 2.6, SCAPE generalizes to new shapes not present in the training data set while being capable of handling changes in pose without introducing artifacts at the joints. These features make SCAPE a highly flexible and realistic body model.

Being able to synthesize realistic shapes of arbitrary people in various poses that can match the appearance of humans in photographs makes data-driven shape modeling the method of choice for image-based fitting. Data-driven shape modeling eliminates the need for manual shape design, predicts more accurately skin deformations at the joints than classical skinning, and is more efficient than anatomical approaches running physical simulations.

2.3.3 Generic Shape Modeling

Finally, it is worth remarking that modeling the entire human body constitutes only a small subset of the vast amount of literature on modeling generic shapes. Pioneering works by Terzopoulos

and Metaxas (1991) and Cootes *et al.* (1995) have proposed the concept of using prior models of shape of same-class non-identical objects to assist in the recovery of shape information from noisy image observations. Early examples include deformable superquadrics [Terzopoulos and Metaxas (1991)], active shape models (ASM's) [Cootes *et al.* (1995)], or eigen-models of 3D objects [Sclaroff and Pentland (1995)].

There is also a large body of literature dedicated to modeling the geometry of faces. Here we briefly mention the seminal work of Blanz and Vetter (1999). They describe a *morphable face model* in which new faces are modeled by forming linear combinations of 200 example scanned face models. They also provide the ability to manipulate an existing face model according to changes in certain facial attributes.

2.4 Sources of Information

In order to extract human models from images, generative approaches need to compare synthesized instances of a parametric model with image observations. Since the appearance of humans in images can be affected by many factors such as lighting conditions or clothing, various image descriptors have been used in the literature for matching human models to features from the image domain. These include silhouettes (contours), edges, appearance (texture), optical flow (motion boundaries), and 3D reconstructions (visual hulls, stereo, shape-from-X).

2.4.1 2D Image Features

Silhouettes and Contours

When the contour of the human silhouette can be extracted reliably, it can be very powerful at constraining both shape and articulated pose, making them one of the most exploited image cues. A matching function in the image domain is often based on area overlap between the observed image silhouette and the silhouette of the projected model [Deutscher and Reid (2005); Sminchisescu and Triggs (2003)], but matching just the silhouette contours is also common. Hilton *et al.* (1999, 2000); Lee *et al.* (2000); Seo *et al.* (2006) use the silhouette contours from orthogonal views for generating 3D human models. Silhouettes from multiple views can also be combined to obtain 3D volumetric reconstructions.

Silhouettes are very good for localizing the person in the image, providing a strong global lock on the subject during tracking that prevents drifting over time. Because they do not convey information about the internal structure, they are susceptible to self-occlusions when limbs are projected inside the body contour for certain camera views (e. g., having an arm between the torso and the camera).

Silhouettes can be extracted robustly and with low computational cost from images when then background is relatively stationary and known. The popular background subtraction technique detects large pixel value differences between a background image and images containing the foreground

object (the human subject). Variants exist that handle slowly changing backgrounds by using adaptive background models, or perform segmentation based on motion. Contour tracking processes assume that the appearance of the foreground is vastly different than that of the background and attempt to find a separation of the two using snakes, active contours, or graph-cut methods. Recognizing the fact that such separation can be ambiguous, a final determination may be postponed by incorporating the image segmentation process in the model fitting stage. Guan *et al.* (2009) and Hasler *et al.* (2009a) perform joint image segmentation/pose estimation using 3D body models as shape priors for silhouette extraction.

Silhouettes cannot always be estimated robustly in the presence of shadows or moving background, making them unusable in these situations.

Edges

Edges are often used for human tracking because they are easy to compute and are often useful for more precisely localizing individual body parts [Deutscher and Reid (2005); Gavrila and Davis (1996); Kakadiaris and Metaxas (2000); Sminchisescu and Triggs (2003); Wachter and Nagel (1999)]. Matching functions typically compute a distance between the model's apparent edges when projected into the image and the closest observed edges detected in the image. Exploiting edges is advantageous because they provide some invariance to viewpoint, lighting conditions and local contrast, and can be used to complement the silhouettes by providing internal contours. However, many spurious edges can be detected in cluttered backgrounds or textured clothing, providing false matches and hiding the relevant ones. Using edge cues by themselves is problematic because it is easy to get confused by spurious edges, leading to tracking failures. Moreover, edges do not help with the recovery of limb rotations around the central axis.

Texture

In contrast to edges that are sparse in nature, image texture captures dense information about the appearance of objects in the scene. Template matching can be used for finding regions in an image which match a template image of a face or other body parts. A reference texture can also be mapped onto the model surface and matched against the texture in the image. Such a reference texture can be provided *a priori* and kept fixed during tracking, or can be updated over time using the estimated models from previous frames [Wachter and Nagel (1999); Sidenbladh *et al.* (2000); Bregler *et al.* (2004)]. Matching can be done using convolution or cross-correlation over raw pixel intensities, or by matching color Gaussian distributions or histograms. Texture-based methods are affected by intensity variations due to changes in lighting or orientation. They also suffer in the presence of loose deformable clothing or in large smooth texture-less regions, causing tracking systems to drift into the background over time.

Texture is being exploited for computing optical flow between consecutive frames, for structure-from-motion, and for 3D stereo reconstructions between different camera views.

Optical Flow

Optical flow is a 2D motion field in the image plane that densely captures the relative motion between frames. Texture information is used to compute smooth pixel displacements from one frame to the next to match the color, thereby relying on the brightness constancy and spatial consistency assumptions. Optical flow can be computed robustly using a multi-scale approach based on image pyramids [Black and Anandan (1996)]. Optical flow could be used for human tracking by matching the estimated optical flow field of the model to the observed optical flow field in images, but it is prone to the accumulation of error and drift, making tracking particularly unstable. Sminchisescu and Triggs (2003) use optical flow to construct outlier maps representing motion boundaries which are used to reinforce select edges.

Switching from the image domain to 3D, optical flow from multiple camera views can be combined to compute a 3D motion field called scene flow [Vedula and Baker (2005)] which captures the 3D motion of points in the scene. Scene flow can be used to constrain the estimation of 3D body motion. Theobalt *et al.* (2003) uses scene flow to augment a silhouette-based fitting method. They present a method for extracting hierarchical rigid body transformations from the motion fields and show that it is best used in conjunction with silhouette-based tracking. A generic body model is first fit to image silhouettes and then pose is refined to conform with estimates from the computed motion field.

Scene flow is also useful for establishing correspondences between 3D shape reconstructions at discrete time instances. Vedula *et al.* (2005) use the scene flow to perform continuous spatio-temporal shape modeling of dynamic objects in the scene. They capture the time-varying geometry of moving objects at discrete time instances using a shape-from-silhouettes approach and use "scene flow"-based interpolation to compute non-rigid changes in shape as a continuous function of time. de Aguiar *et al.* (2007) use a deformable mesh to capture the motion and dynamic shape of humans. They track surface deformations of an actor wearing wide apparel by evolving an *a priori* shape model according to 3D correspondences given by the scene flow. A Laplacian tracking scheme is incorporated to achieve robustness against errors in the 3D flow.

2.4.2 Depth Information from 3D Reconstructions

The image features presented so far, with the exception of scene flow, are inherently 2D, do not encode any depth information, and encounter difficulties with self-occlusions. Being able to extract 3D information from images can help a great deal with human shape and pose recovery. In the case of single images, this is particularly challenging, being an ill-posed problem. Nonetheless, shape-from-X techniques exist that attempt to recover shape information of generic rigid objects (e. g. a statue, building or toy) from single images in carefully-controlled environments. The shape recovery problem becomes much easier when combining information from multiple calibrated camera views. Here we focus our discussion on shape-from-silhouettes and multi-view stereo reconstruction methods for capturing people.

Shape-from-Silhouettes

Image silhouettes from multiple camera views can be used to compute 3D reconstructions of objects in the scene. These reconstructions are called *visual hulls* [Laurentini (1994)] and are the maximal possible 3D volume that still projects inside the image silhouettes. Two basic techniques exist for computing shape-from-silhouettes (SfS): one is based on voxel carving, while the other is based on intersecting generalized cones. Both methods require that camera calibration parameters (both intrinsic and extrinsic) be estimated in a calibration step, and rely on the fact that image silhouettes are the 2D projection of the actual 3D object onto the camera image plane. A silhouette, together with the camera calibration parameters, define a back-projected generalized cone that tangentially contains the actual object. The visual hull can be obtained by taking the volume intersection of several silhouette cones from multiple camera views [Franco *et al.* (2006)]. Alternatively, a bounding box of the capture volume of interest can be discretized into 3-dimensional equal-sized cubes called voxels (the 3D equivalent of 2-dimensional pixels in the image domain) and a carving procedure is employed that eliminates all voxels whose projections do not fall inside the image silhouettes in each camera view. The remaining voxels or the visual hull provide an upper-bound on the volume occupied by the actual object in the scene. The bound becomes tighter as more camera views are employed, but artifacts will remain in concave regions regardless of the number of views. A variation of voxel carving is given by *voxel coloring* [Cheung *et al.* (2003a)] in which the maximal volume also has a consistent foreground color across all views. Reconstructing the visual hull for the human body can be done with as little as 3 or 4 camera views, but more acceptable reconstructions require between 8 and 16 camera views. Shape-from-silhouette methods are susceptible to errors in the extracted image silhouettes and are usually used in combination with a chroma-key technique that makes foreground segmentation trivial. Chroma-keying involves making the background green or blue such that it is distinct from the color of skin or the clothes worn by the subject. Reconstructions based on voxel carving [Cheung *et al.* (2000, 2003a); Mikić *et al.* (2003); Kehl *et al.* (2005)] and visual hulls [Chu *et al.* (2003); Cheung *et al.* (2003b); Menier *et al.* (2006)] have been used extensively for human pose estimation.

Stereo Reconstruction

In addition to silhouettes, texture can also be used for extracting depth information. Stereo reconstruction algorithms use texture to establish pixel correspondences between multiple calibrated camera views before computing the 3D intersection of back-projected rays through corresponding pixel locations. Color calibrated cameras are required for matching pixels based on color between different cameras. The output is a 3D point cloud that can be post-processed into a surface mesh. The resulting mesh will contain holes in regions occluded from the cameras.

Classical binocular stereo uses epipolar geometry between two cameras with a narrow baseline to simplify the search for image correspondences, but only produces one-side reconstructions. In [Grest *et al.* (2005)], a 3D body model is then fit to such stereo reconstructions to estimate frontal human motion by employing an Iterative Closest Point (ICP) method that finds correspondences

(a) Input image (b) Shape from silhouettes (c) Multi-view Stereo (d) Refined shape

Figure 2.7: **Multi-view 3D Reconstructions.** Images (a) from 8 wide-baseline camera views can be used to reconstruct shape from silhouettes (b), or using multi-view stereo (c). Starck and Hilton (2007) compute refined shape reconstructions (d) by fusing silhouette and stereo cues with matched surface features between views. (Reprinted from [Starck and Hilton (2007)])

between points on the body model and the stereo reconstructed points.

Multi-view stereo techniques [Seitz *et al.* (2006)] can reconstruct all-around volumetric models by using three or more images taken from known camera viewpoints in a 360° configuration. Establishing correct pixel correspondences for triangulation between views becomes more challenging due to the wide baseline between the cameras. In addition to individual pixel correspondences based on intensities, other image features (e. g., texture discontinuities or SIFT[1] descriptors) can also be used for sparse matching, assisting in the 3D reconstruction process [Starck and Hilton (2007)].

Bradley *et al.* (2008) capture the geometry and motion of garments worn by people in action using multi-view stereo from 16 cameras to extract initial 3D meshes for each frame. Since the stereo meshes contain holes and have different connectivity, a template-based mesh completion technique is then applied that both completes the surface geometry and establishes temporal correspondences.

More accurate and robust reconstructions (cf. Figure 2.7) can be obtained by combining silhouette-based and stereo-based methods [Furukawa and Ponce (2009); Starck and Hilton (2007)].

2.5 Human Body Model Acquisition for Motion Capture

For the purpose of this thesis, the problem of Markerless Human Motion Capture consists of several phases: 1) body model acquisition, 2) pose estimation, and 3) tracking. Body model acquisition deals with generating a body model specific to the subject being tracked. Given a body model, pose estimation consists of recovering the articulated kinematic structure in a single frame

[1]The Scale-Invariant Feature Transform features (SIFT [Lowe (1999)]) are local textural image descriptors of particular interest points in the scene. SIFT is invariant to image scale and rotation, robust to changes in illumination and minor changes in viewpoint, and highly distinctive, making it appropriate for robust image matching.

using one or more camera views, which can be used to initialize a tracking procedure. Tracking then refers to the estimation of the articulated pose and shape over an image sequence that exploits the temporal correlations between consecutive frames to simplify the estimation problem. Tracking can be followed by a motion or shape analysis and recognition phase.

2.5.1 Shape Initialization

Estimating the kinematics of a body model from images relies heavily on having access to the correct body shape measurements, which are often assumed known or estimated in a pre-processing step prior to tracking. For generic part-based shape models that represent each part as a rigid volumetric object attached to a skeleton, the shape is often described in terms of a few scaling proportions that capture the lengths of the bone and the widths of the limb. Many previous approaches use manually measured shape parameters of the body parts to be tracked [Deutscher and Reid (2005)], or manually tune them to match the body shape in an image [Bandouch *et al.* (2008)]. Others extract the shape parameters (semi)-automatically directly from images with the subject standing in several canonical poses [Gavrila and Davis (1996)].

In [Gavrila and Davis (1996)], a cooperating subject assumes a standing pose in two orthogonal views (frontal and side) and a model based on tapered superquadrics is fitted to silhouette contours in a coarse-to-fine fashion. First the head is detected using color histograms and the major axis of the silhouette is used to obtain the torso-head configuration. Multiple views provide a 3D estimate of the major torso axis, followed by an incremental search for the orientation and dimensions of the limbs.

While it is simpler to estimate the scaling parameters for part-based shape models, such models are limited in fidelity. More sophisticated models can be obtained from images by relying on a reference humanoid model that is deformed to match silhouette contours.

Hilton *et al.* (1999, 2000) transform a standard 3D generic humanoid model to approximate a person's shape and anatomical structure observed from 4 orthogonal views (front, sides and back). With the person standing in the same pose as the humanoid model, feature points along the silhouette contours at extrema locations are used to determine dense interpolated correspondences with synthetic contours of the reference model. The vertex displacements in the planes are then interpolated to obtain vertex displacements over the entire body. The kinematic skeleton is estimated from the principal axes of the image contours. The method is simple and fully automatic, but the shape reconstructions are lacking. Following a similar approach, Lee *et al.* (2000) use a seam-less humanoid model to increase the realism of the recovered shape model and manually specify feature points along the silhouette contours taken from orthogonal views for better accuracy. They perform body cloning using a 2 stage body modification, first based on feature points and second on silhouette contours. The kinematic skeleton is also estimated from the contours. Such simple methods produce reasonable but not great body shapes. Instead of a simple reference humanoid model, Seo *et al.* (2006) employ a statistical model of human shape deformations to reconstruct human body models from orthogonal silhouettes.

Other approaches can reconstruct surface models without relying on a reference body model. Kakadiaris and Metaxas (1998) define a protocol of controlled movements that, when performed by the subject, reveal the structure of the human body. 3D surface shape models are reconstructed from 2D contours from 3 orthogonal views. In [Cheung *et al.* (2005)], the subjects performs a series of specific actions that are used for identifying the joint centers of several main joints of the human body using voxel data.

For body models using detailed surface-based representations, 3D surface models may be obtained using laser-scanning or shape-from-silhouette and stereo methods. A laser scan may provide a high resolution mesh model of the body, but the articulations are not explicit. The 3D mesh is then rigged with a kinematic skeleton either manually [Vlasic *et al.* (2008)] or automatically [Corazza *et al.* (2008)] using a learning approach. The scan can also be segmented into body parts which are kept rigid during tracking [Rosenhahn *et al.* (2005); Mündermann *et al.* (2007)]. For providing a better fit with the image observations during tracking, it is better to have the subjects scanned with the same clothes as during the motion capture phase. This makes tracking dynamic surfaces tractable [de Aguiar *et al.* (2007); Vlasic *et al.* (2008); de Aguiar *et al.* (2008)].

Finally, Mikić *et al.* (2003) use a body part localization procedure based on template fitting and growing, and optimize the scaling parameters of the ellipsoids and cylinders used as body parts. Instead of estimating the shape parameters in a pre-processing step, they perform shape refinement using a Bayesian network that is incorporated into the tracking framework for several frames in the beginning of the sequence. They use prior knowledge of average body part shapes and dimensions to impose human body proportions onto the body part size estimates. Tracking is performed using an Extended Kalman Filter (EKF).

2.5.2 Pose Initialization

Analysis by synthesis is typically implemented as a local search around an initial estimate of the state parameters. When tracking an entire image sequence, the estimate at the previous frame can be used as initialization for the current frame. This leaves open the question on how to initialize the model in the first frame of the sequence.

Many approaches assume the initial kinematic pose is known (e. g. [Wachter and Nagel (1999)]) or use a manual procedure for initialization. Some methods benefit from a cooperative subject that strikes a predefined canonical pose to be detected by the tracking system and then proceeds to perform the actions of interest [Kehl *et al.* (2005)]. Others acquire the initial pose using a marker-based motion capture system [Sigal *et al.* (2010)].

More general pose estimation methods can be categorized into generative and discriminative. Generative methods are model-based and use a top-down approach, matching projections of hypothesized body models to image observations. As such, they fall into the analysis-by-synthesis framework (Section 2.2) and require a good initial estimate of the pose. In contrast, discriminative methods, which we describe next, predict the body model by analyzing the image data directly. They either work in a bottom-up fashion, where candidates for individual body parts are located in

the image and then assembled into a consistent human body configuration, or use training examples to learn a direct mapping from images to pose. Such methods can be fully automatic, but are not sufficiently precise in general, making them appropriate for initializing generative methods.

Semi-Supervised Geometric Methods

Semi-supervised approaches based on geometry reasoning have been proposed both for estimation of anthropometry and pose estimation from a single calibrated or uncalibrated image. Geometric approaches use manually clicked points in a 2D image corresponding to major joints and some statistical prior over bone length proportions to recover a family of 3D skeleton configurations consistent with the 2D constraints, up to a scale factor. Taylor (2000) assumes known limb lengths and takes the image foreshortening of the segments into account, but has to specify the depth ordering for each bone manually. Since it assumes a scaled orthographic camera, the approach is limited to far views, although Lee and Chen (1985) presented a perspective solution earlier. Various extensions exist with different assumptions about the anthropometric measurements of the skeleton [BenAbdelkader and Davis (2006); BenAbdelkader and Yacoob (2008)] or the geometry of the pose [Barrón and Kakadiaris (2001, 2003)]. Mori and Malik (2006) extend the method to obtain the 2D joint locations automatically using shape context features. Geometric approaches remain particularly unstable to minor perturbations of the marked joint locations in 2D.

Automatic Bottom-up Discriminative Methods

One way to estimate the initial pose automatically is to use a bottom-up approach which works by first locating candidates for individual body parts in the image domain or within volumetric data and then assembling them into a consistent human body configuration.

The human body is decomposed into parts which are modeled using simple 2D or 3D geometric primitives. Part-specific detectors are devised to provide initial hypotheses for the placement of each limb. In general, the detectors are not very precise due to lack of strong identifying features, missing some parts and providing many false positives for others. A set of pair-wise constraints between parts, encoded directly in terms of compatibility, or probabilistically, need also be defined before a consistent body configuration is assembled. The process requires no manual intervention however the solution is often not well localized due to parts not being detected or to the inability to incorporate constraints other than kinematic constraints. A decomposition into body parts is also assumed, which may not be appropriate for people wearing loose clothing such as dresses.

Hybrid methods combine bottom-up and top-down approaches to complement each other. Bottom-up strategies provide a way to initialize and recover from drift during motion tracking, while top-down methods can recover more precise pose estimates by being able to provide robustness against noisy image observations, reason about the human body and its interaction with the environment, incorporate various types of constrains, and handle non-rigid deformations.

Sigal *et al.* (2003) represent the 3D human body as a graphical model in which the relationships between the body parts are encoded by conditional probability distributions and formulate the

problem as a probabilistic inference over a graphical model using approximate belief propagation. Rodgers *et al.* (2006) follow a similar approach, but deal with estimating pose from 3D range-scan data and relying on 3D part detectors. Nonetheless their methods are applicable to any type of volumetric input data (e. g. shape-from-silhouettes or stereo depth reconstruction). They also refine the pose estimates using a top-down Iterative Closest Point (ICP) approach. Gavrila and Davis (1996) use a combination of bottom-up and top-down for pose initialization. They employ a search-space decomposition strategy that first uses part detectors to find the head and torso, and then localize the limbs incrementally along the kinematic tree.

Example-based Discriminative Methods

Example-based discriminative approaches do not use an explicit model of the human body but rather attempt to learn a mapping directly from 2D image features on the body to 3D kinematic pose. The mapping can be exemplar-based [Shakhnarovich *et al.* (2003)], in which a database of example feature-pose associations is queried for nearest neighbor matches, or probabilistic-based, in which a more compact representation is learned from training feature-pose pairs using various flavors of regression [Agarwal and Triggs (2006)] or Bayesian Mixture of Experts [Sminchisescu *et al.* (2005); Sigal *et al.* (2008)]. Most commonly used image features include: shape contexts [Sminchisescu *et al.* (2005); Agarwal and Triggs (2006)] and histogram of oriented gradients [Shakhnarovich *et al.* (2003)], etc. Such approaches are computationally efficient, but require a training database that spans the set of all possible poses, body shapes, and/or scene conditions (lighting, clothing, background etc.) to be effective. Moreover, the performance degrades significantly when the image features are corrupted by noise or clutter. In such cases, a generative approach is more appropriate as it models the image formation process explicitly.

Sigal *et al.* (2008) combines a discriminative approach with a generative approach for model verification and refinement. Their technique uses shape context features not only for predicting pose, but also for predicting shape parameters from silhouette contours.

2.5.3 Model Tracking

Tracking the human body is a simplified version of the pose estimation problem since the pose estimate in the previous frame gives a strong indication on what the body configuration might be in the current frame. Tracking is not the focus of this thesis. For an extensive review on the subject, please refer to [Moeslund *et al.* (2006)].

Chapter 3

SCAPE: A Deformable Body Model of Shape and Pose

3.1 Introduction

Central to this thesis is a recently proposed deformable body model called SCAPE (Shape Completion and Animation of PEople) that is more suitable for analyzing humans in images than traditional models. Originally, SCAPE was designed for graphics applications to produce realistic animations. We aim to use this model for computer vision tasks such as matching the shape and pose of arbitrary, previously unseen, subjects directly from images. In this chapter we describe the main features of the SCAPE model as proposed in [Anguelov (2005)] and highlight the key differences between the original formulation and ours.

SCAPE is a morphable body model that is able to capture both variability of human shapes between individuals as well as pose-related skin deformations (see Figure 3.1). The model is learned from example shapes acquired with 3D laser scanners and is able to reconstruct shapes that are a lot more realistic than the part-based body models used in the computer vision literature. Specifically, it models both articulated and non-rigid surface deformations and captures the correlations between sizes and shapes of different body parts. Equally important for computer vision applications, the model factors changes in shape due to identity from shape changes due to pose. Compared to the physics-based simulation models, it is also more efficient and relatively low dimensional.

Statistical modeling of surface deformations requires known correspondences between semantically equivalent locations on the example bodies. To that end, we describe a non-rigid iterative closest point technique that registers a reference mesh with the laser scans and in the process deals with missing surfaces in the raw scans.

Unlike previous implementations, our work relies on an extensive dataset of high resolution laser scans comprising more than 2000 individuals. This allows us to more accurately capture the variability in human form between individuals than previously possible. Dealing with such a large

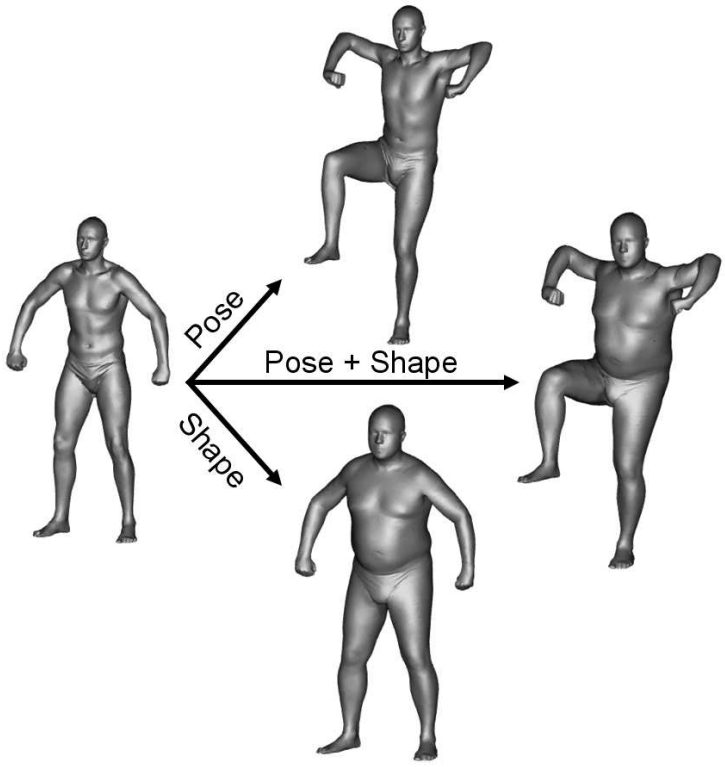


Figure 3.1: **SCAPE Synthesized Human Models.** The SCAPE model can be used to transform a reference body (*left*) into different poses (*top*), different body shapes (*bottom*), or a combination of different shapes in different poses (*right*).

amount of data however pushes the limits of current computer hardware capabilities. We use an incremental Principal Component Analysis approach (Appendix D) to address computer memory limitations and make learning a shape deformation model practical. We also go beyond previous work and learn three different shape models: one for men, one for women, and one gender-neutral model combining both men and women, which becomes useful for gender-constrained shape estimation from images.

3.2 Related Work

Simpler body models exist in the graphics literature. For example, synthetic humanoid models can be designed using specialized commercial software tools, with the shape controlled though a number of scaling parameters and pose varied by associating the surface mesh with a kinematic skeleton. While such models are easy to animate, and allow for pose and shape to be altered independently, the resulting shapes often lack realism.

Until very recently, models gained realism by learning either the deformations due to pose or due to identity changes from example 3D body scans, but not both. They used incompatible

representations that made merging the two deformation models difficult. For example, Allen *et al.* (2002) learn a model of pose deformations using point displacements from an underlying articulated model but can only handle a single subject, while Allen *et al.* (2003) and Seo and Magnenat-Thalmann (2003) model identity changes as point displacements from an average shape which are embedded in a linear subspace, but need to keep the pose fixed. The latter can be animated using procedural skinning techniques that cannot capture muscle bulging and that introduce twisting artifacts at the joints.

The SCAPE model [Anguelov *et al.* (2005b)] was the first to change the representation to triangle-based deformations, allowing for models of pose and body shape deformations to be learned separately, and using simple matrix multiplication to combine them, thus producing body models of different people in different poses with realistic skin deformations.

Since SCAPE, two new models have been proposed that combine pose and identity shape changes. Allen *et al.* (2006) learn a complex system that combines corrective skinning learned from examples with a latent model of identity variation. Unfortunately the complexity of the proposed training phase limits the amount of training data that can be used, which consequently impairs the model's realism. Hasler *et al.* (2009b) propose a representation that couples pose and identity shape deformations into a single linear subspace, where the deformations are based on an encoding that is locally invariant to translation and rotation. However, their model lacks the property of being able to factor changes due to pose from changes due to identity, which is necessary for estimating a consistent shape across different poses (Chapter 6).

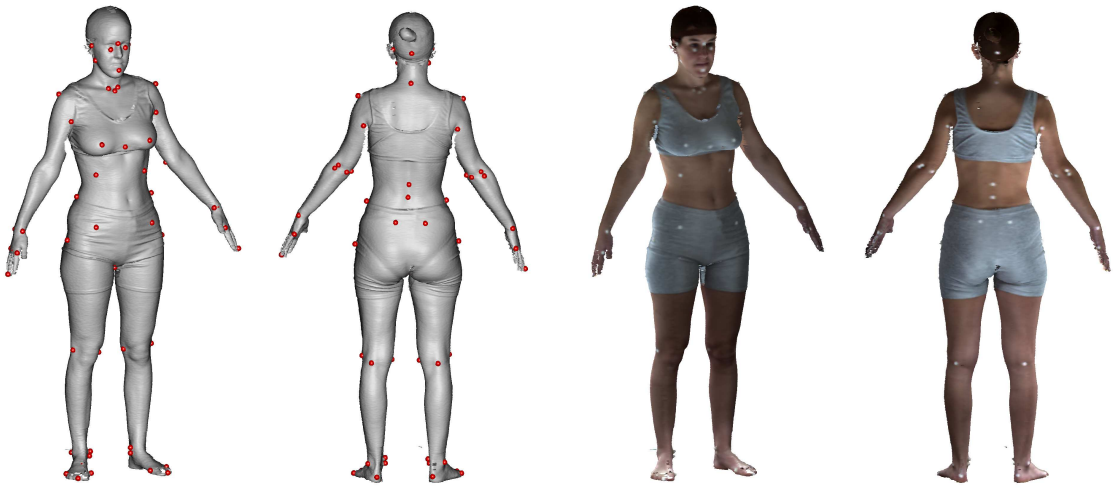
3.3 3D Scan Dataset Acquisition

Models of human body deformations can be learned from examples acquired in the form of high resolution 3D laser scans of real people. We use scans acquired using Cyberware^{TM} whole-body scanners that capture two to four simultaneous scans from orthogonal views and stitch them together into all-around surface meshes (Figure 3.2). The reconstructed meshes are not complete; holes remain in the reconstruction due to self occlusion and grazing angle views (Figure 3.2b). The subjects are scanned wearing minimal skin-tight shorts and a latex cap to cover the hair, with the women also wearing sports bras.

SCAPE can be thought of having 2 components. The first component is a *pose deformation model* which captures how the body shape of a person varies as a function of their pose; for example, this captures how the bicep muscle budges as the arm is bent. The second component is a *shape deformation model* that captures the variability in human shape across people using a low-dimensional linear representation. We obtain separate training sets for the two models.

For the *pose training set*, we use the same data as in [Anguelov (2005)]. Specifically, a single male subject is scanned in 71 diverse poses (Figure 3.3b). For the *shape training set*, we acquired the American edition of the publicly available CAESAR dataset (Civilian American and European

(a)



(b) (c)

Figure 3.2: **Shape Acquisition using Laser Scanning.** (a) A Cyberware whole-body laser scanner measures the 3D position and appearance for hundreds of thousands of points on the surface of the scanned body. (b) A typical surface scan is shown without texture, revealing common scanning artifacts. Holes remain where the surface cannot be estimated due to self occlusion or grazing angle views. (c) The same scan is shown front and back with the captured texture. Since the measured points come with no semantic meaning, sparse white markers (visible in (c)) can optionally be placed at anthropometric landmark locations on the body during scanning, providing context for body measuring and scan registration. Their 3D locations are estimated and manually labeled in a post-processing step and displayed as red dots in (b).

Surface Anthropometry Resource, SAE International). This dataset contains 2,384[1] range scans of adults from North America, aged 18-65 and with about equal gender representation. All subjects in the CAESAR dataset are scanned in a very similar, but not exactly identical, standing pose (Figure 3.3c). Prior to scanning, 74 white markers are placed on each subject at anthropometric landmarks and their 3D location computed (Figure 3.2b,c).

Our use of the extensive CAESAR dataset for capturing shape variability between individuals is in sharp contrast with the number of subjects used in the original implementation of the SCAPE model [Anguelov (2005)]. Their shape deformation model was built using only 45 different subjects, which is arguably insufficient to adequately represent the complex space of human shapes.

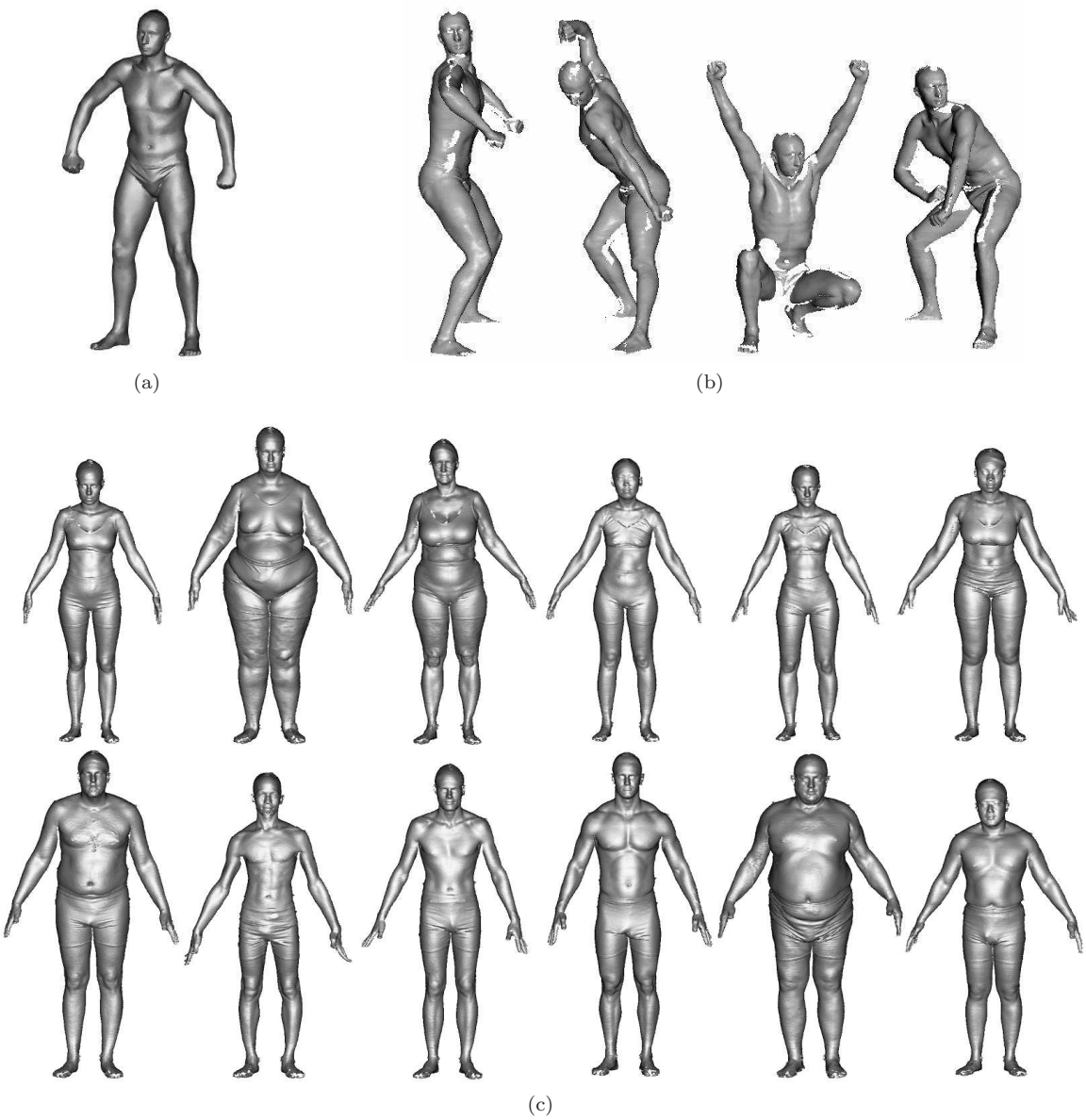
The raw scans contain missing surface regions due to limitations in the scanning process, provide no semantic correspondences between surface points necessary for shape modeling, and are at very high resolution, having between 200,000 and 350,000 triangles. However, for computational and algorithmic reasons, it is desirable to have lower resolution meshes that share the same vertex connectivity. The standard approach for jointly addressing these problems is to use a template-based non-rigid registration technique that brings all human laser scans in the training sets into full correspondence with respect to a reference mesh (Figure 3.3a), and implicitly with each other. By this, what is meant, for example, is that a mesh vertex on the right shoulder in one person corresponds to the same vertex on another person's shoulder. It also means that all aligned meshes have the same number of vertices and triangles. We discuss this process of surface registration and hole filling next.

3.4 Surface Registration

Surface registration is the process of establishing point-to-point correspondences among similar objects that exhibit the same overall structure but substantial variations in shape. In our case, we are interested in registering scans that exhibit deformations due to changes in pose and to variations in shape among different people. The general idea of template-based mesh registration techniques is to smoothly deform a reference mesh with a desired topology toward the example scans such that the resulting meshes have the geometry of the raw scans, but the topology of the reference mesh.

One approach is to use the technique of Allen *et al.* (2003) that employs a non-rigid Iterative Closest Point (ICP) algorithm. While computationally efficient, such an approach generally requires a set of sparse correspondences between the template and the raw scan be specified in advance to initialize and guide the alignment in the presence of large deformations. Alternatively, the Correlated Correspondence algorithm of Anguelov *et al.* (2005a) can be used to register significantly deforming surfaces in a mostly unsupervised manner. The algorithm is based on a probabilistic model over the set of all possible point-to-point correspondences and relies on matching the local geometry and preserving geodesic distances and local surface deformations. The search in the combinatorial space of correspondence assignments is done using probabilistic inference over a Markov network.

[1] Due to various scanning artifacts, we only use the scans from 1,051 male subjects and 1,096 female subjects.



(a) (b) (c)

Figure 3.3: **Training Scans.** (a) Template mesh, hole-filled and sub-sampled to $12,500$ vertices and $25,000$ triangles. (b) Subset of the pose training set, containing raw scans of a single subject in 71 poses. (c) Subset of the body shape training set, consisting of raw scans of 1000+ female subjects and 1000+ male subjects from the CAESAR dataset (SAE International).

With exact inference being infeasible, and approximate loopy belief propagation requiring significant amounts of computer memory and converging too slowly over large Markov networks, the Correlated Correspondence algorithm becomes practical only for sub-sampled meshes with approximately 250 vertices [Anguelov (2005)]. However, because it is designed to work in more general settings without requiring markers or other prior knowledge about the object shape, it remains effective at

coarsely aligning surfaces that undergo significant articulated deformations and subsequently use the alignment to initialize a standard non-rigid ICP algorithm (i. e. [Allen *et al.* (2003)]).

3.4.1 Marker-based Non-Rigid Iterative Closest Point Registration

Here we describe the method we applied for aligning a template mesh $\mathcal{T}$ to a deformed target mesh $\mathcal{D}$ given some corresponding marker locations, as proposed by Allen *et al.* (2003). The template mesh acts as a reference mesh that is smoothly deformed into other poses and body shapes to establish correspondences between all training meshes. Each of these shapes are represented as triangular meshes (although any polygon mesh representation can be used) consisting of a set of V vertices and a set of T triangle faces sharing common vertices. An optimization problem is formulated that solves for the 4×4 affine transformation $\mathbf{T}_i$ for each vertex $\vec{v}_i$ of the template mesh $\mathcal{T}$ using an objective function that trades off fit to the raw scan data, fit to known markers, and smoothness of the transformation. For the purpose of mesh registration, vertex locations are expressed in homogeneous coordinates: $\vec{v}_i = [x_i, y_i, z_i, 1]^\mathsf{T}$.

Data Error

Our first objective is for the aligned template surface to be as close as possible to the target surface. As such, we encourage vertices of the template mesh to move toward the closest respective points on the surface of the target mesh in order to acquire the geometry of the raw scan. We use the data error term E_d to penalize the remaining gap between the transformed vertices $\mathbf{T}_i\vec{v}_i$ and the target surface $\mathcal{D}$:

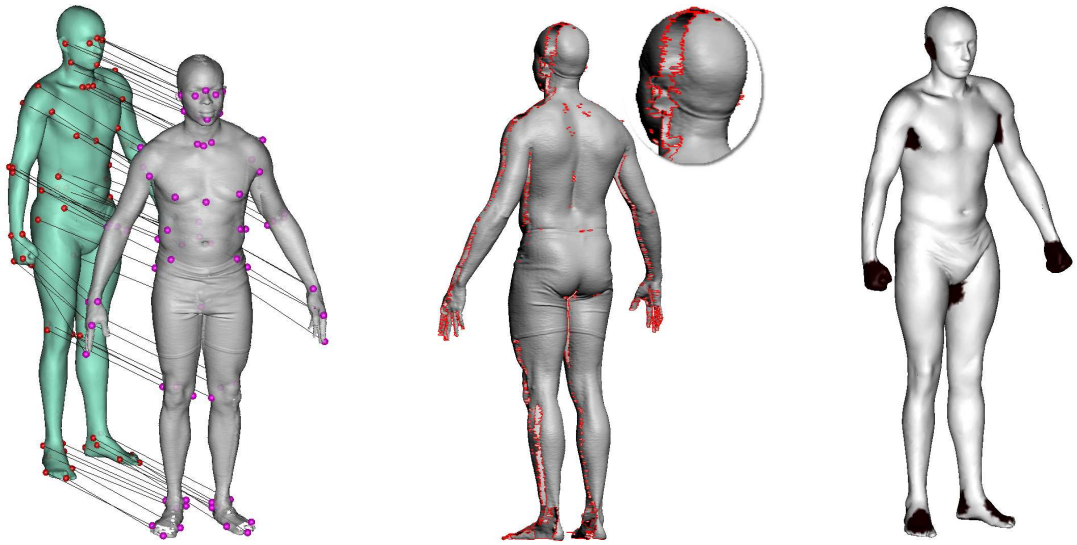
$$E_d = \sum_{i=1}^{V} w_i \, \mathrm{gap}^2(\mathbf{T}_i\vec{v}_i, \mathcal{D}) \, . \tag{3.1}$$

Here, V denotes the number of vertices for the template mesh $\mathcal{T}$ and w_i is used to control the influence of the data term in the presence of holes in the target mesh. The function $\mathrm{gap}(\cdot, \cdot)$ computes the distance from a point to the closest compatible vertex of a surface and it is implemented using a KD-tree data structure for computational efficiency. The compatibility restriction safeguards against front-facing surfaces being matched to back-facing surfaces and is measured in terms of the angle between the surface normals. It also restricts the distance between them to a threshold to avoid matching through holes in the target mesh.

Smoothness Error

Matching points to a wavy surface independently using the closest point strategy alone introduces unnatural folding and stretching artifacts. The solution can be regularized by adding a deformation smoothness constraint E_s that require affine transformations applied to adjacent vertices on the surface to be as similar as possible:

$$E_s = \sum_{\{i,j|(\vec{v}_i,\vec{v}_j)\in \mathrm{edges}(\mathcal{T})\}} ||\mathbf{T}_i - \mathbf{T}_j||_F^2 \, , \tag{3.2}$$



(a) (b) (c)

Figure 3.4: **Mesh Registration Process.** (a) Sparse landmark correspondences between the template mesh (green) and a raw scan (gray). (b) Vertices and edges located on the boundary of a hole in the scan are highlighted in red. (c) Vertex weights w_i for the data term. Black regions correspond to vertices from the template mesh with a zero data-fitting weight, while white regions have weight one.

where $||\cdot||_F$ denotes the Frobenius norm.

Applying this constraint is not the same as applying a surface smoothness constraint. This is important because what we want is to preserve the surface details present in the template mesh, particularly in regions where the target mesh contains holes.

Marker Error

In the presence of large surface deformations of the target mesh with respect to the template, the data and smoothness constraints are often insufficient for achieving a correct alignment. To assist the alignment process and help guide the deformation of the template mesh into place, a set of points on the target surface $\vec{m}_i$ that correspond to known points on the template are identified. We need to encourage the correspondences at marker locations to be correct. We use the marker error term E_m to minimize the distance between each marker's location on the template surface and its location on the target surface:

$$E_m = \sum_{i=1}^{M} ||\mathbf{T}_{\kappa_i}\vec{v}_{\kappa_i} - \vec{m}_i||^2 \,. \tag{3.3}$$

Here $\kappa_{1\cdots M}$ is a list of vertex indices from the template mesh that correspond to the markers on the target surface (Figure 3.4a).

Objective Function - Optimization Strategy

The complete objective function is a weighted sum of the three error terms:

$$E = \alpha E_d + \beta E_s + \gamma E_m \tag{3.4}$$

and can be optimized using gradient descent.

The optimization is initialized using a global transformation that aligns the center of mass of the template to the target mesh and the overall orientation and scale. The optimization continues in stages, following different weighting schedules that emphasize matching landmarks (10,000 iterations with $\alpha = 0, \beta = 10, \gamma = 20$), overall shape fitting (600 iterations with $\alpha = 2, \beta = 10, \gamma = 2$), and refinement of the surface geometry (400 iterations with $\alpha = 1, \beta = 0.1, \gamma = 0.1$).

Hole-filling

Large holes in the scanned mesh pose problems to the method described so far. In this case many vertices on the template have no correct correspondence on the scanned mesh. The data term encourages these vertices to move instead to the closest existing patch from the target mesh causing undesirable stretching. Fortunately, at each iteration we can easily identify the vertices $\vec{v}_i$ with no true correspondence as the ones whose closest point on the target mesh is located on the boundary edge of a hole (Figure 3.4b). For these vertices, we set the weight w_i in E_d to zero so that the transformations $\mathbf{T}_i$ will be driven by the smoothness constraint E_s. The effect is that holes are filled in by seamlessly transformed parts of the template surface.

3.4.2 Processing Pipeline

We apply the non-rigid mesh registration technique to our two training data sets, one containing the same subject in different poses, and the other containing different subjects in the same canonical standing pose.

Template Mesh

We choose a mesh from the pose training set standing in the canonical pose to be the template mesh. The template mesh is hole-filled and subsampled to contain $25,000$ triangles with $12,500$ vertices (Figure 3.3a). The remaining *instance meshes* are brought into full correspondence with the template mesh.

Acquisition of Marker Locations

Marker locations are obtained differently for the two training sets. For the shape training set, we use the location of the 74 anthropometric markers from the CAESAR dataset (see Figure 3.2b,c) to establish sparse correspondences with the template mesh. In the case of the pose training set in which the surface deformations due to articulated pose changes are more significant, many more marker

locations are obtained using the Correlated Correspondence algorithm [Anguelov *et al.* (2005a)]. This algorithm uses only 4-10 pairs of manually-placed initial markers to break the scan symmetries and produces a larger marker set of about 200 (approximate) sparse correspondences. These markers are subsequently used in Equation 3.3.

Initialization of the Mesh Registration

The registration algorithm is initialized using a global transformation that aligns the center of mass of the template to the target mesh and the overall orientation and scale. For the pose training set, all scans are of the same subject so the scale is kept constant. To estimate the initial overall orientation with respect to the template, the landmarks can be used to compute the optimal rigid global rotation between two corresponding point sets (see Appendix C.1). In the case of the shape training set, the subjects were already scanned in matching orientation. The scale factor is set to fit the apparent height of the CAESAR scans to the height of the template. The apparent height is just the z-coordinate difference between the highest and the lowest vertex of the mesh since the subjects were all scanned in the same standing pose. The translation is obtained as the difference between the centroids of the vertex sets of the two meshes. We use the global rotation $\mathbf{R}$, translation $\vec{t}$ and scale s to form the global rigid transformation $\mathbf{T}^0$ which is used to initialize the transformation for each vertex $\mathbf{T}_i$:

$$\mathbf{T}^0 = \begin{bmatrix} s\mathbf{R} & \vec{t} \\ 0 & 1 \end{bmatrix}. \tag{3.5}$$

Exclusion of Poorly Scanned Regions

The registration method handles large holes by automatically detecting vertices that are attracted to boundary edges and ignoring the data term. Special care must also be taken in regions that contain scattered surface fragments instead of a large hole since it is difficult to correctly match small fragments reliably to the template mesh. Poorly scanned regions include the ears, fingers, feet, arm pits and crouch. Additionally, the subjects in the CAESAR dataset were scanned using open hands while the template and the pose dataset contain closed hands as fists[2]. We manually identify these specific regions on the template mesh (see Figure 3.4c) and set w_i to zero to favor the template over the scanned data in these regions.

3.4.3 Results and Applications for Shape Registration

Figure 3.5 illustrates typical results for the shape registration process. The resulting meshes have the geometry of the raw scans, but the topology of the reference mesh. Having all the meshes in full vertex and triangle correspondence enables several applications. Some particularly trivial applications include texture transfer and interpolated shape morphing by linearly combining vertex locations of registered meshes (Figure 3.6).

[2]Our implementation explicitly ignores landmarks 37 and 49. These are landmarks at the ends of the fingers. This is done because the template model has its hands in a fist pose while the CAESAR models are open hand.

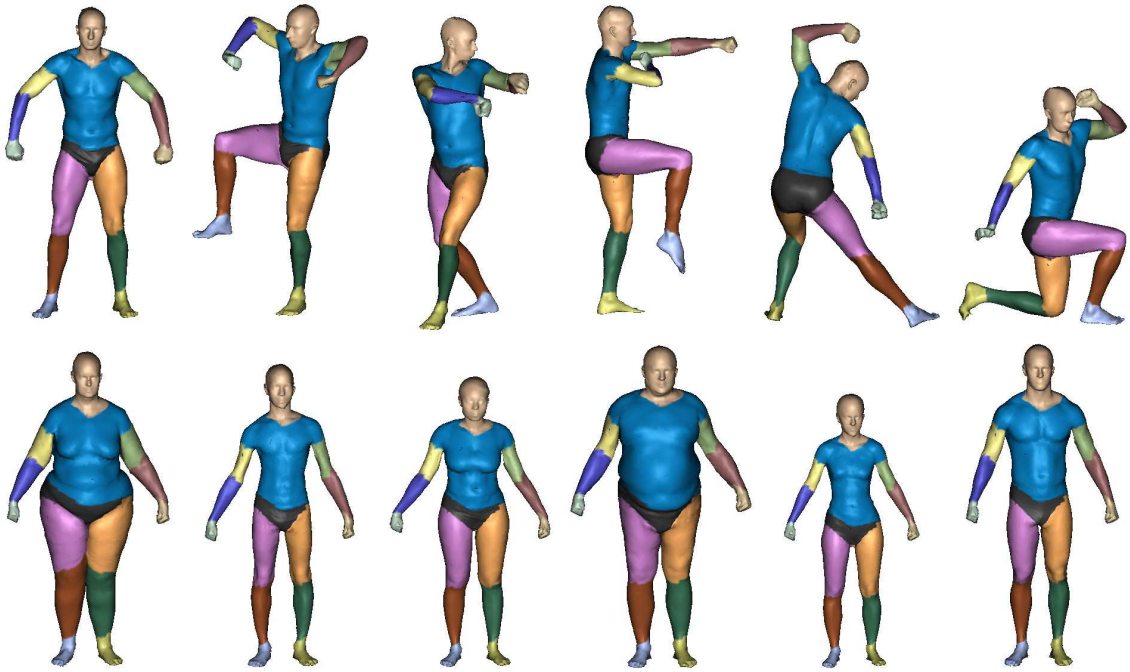


Figure 3.5: **Mesh Registration Results.** Example meshes obtained using the mesh registration process, where the template mesh (top left) has been brought into alignment with the scans. By construction, these meshes maintain full per-vertex and per-triangle correspondence with the template mesh and are hole-filled. Given a segmentation of the template mesh into 15 body parts, the segmentation transfers naturally to the example meshes using the learned correspondences. We illustrate the transfer by assigning a different color to each body part. (Top) example meshes from the pose training set; (Bottom) example meshes from the shape training set.

The registered pose training set can be used for unsupervised learning of the articulated structure of the body. Anguelov (2005) proposes an algorithm that uses a set of meshes corresponding to different configurations of an articulated object to automatically recover a decomposition of the object into approximately rigid parts. It defines a graphical model to capture the spatial contiguity of parts and performs segmentation using the EM algorithm. The method produces an initial segmentation of the template mesh into 18 parts, where the front and back of the pelvis area and the torso were split into several pieces. After manually merging some of them, we obtain a segmentation of the template triangles into 15 body parts corresponding to pelvis, torso, head, upper and lower arms and legs, hands and feet. Since the training data is in full triangle correspondence, the division into parts naturally applies to all meshes. Figure 3.5 shows the aligned meshes with the individual body parts color-coded.

Next, we are using the meshes in full correspondence for shape morphing and statistical modeling of the shape deformation space.

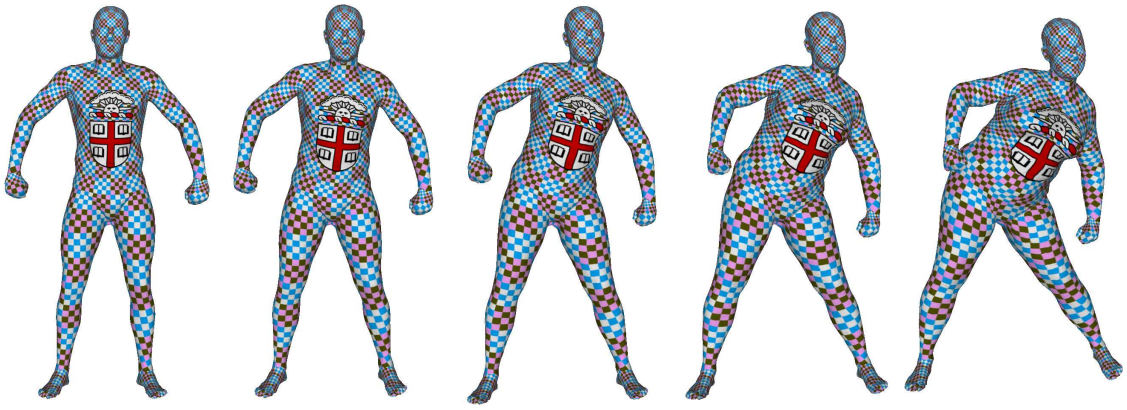


Figure 3.6: **Example Applications of Shape Registration.** The leftmost and rightmost meshes are in full vertex and triangle correspondence. This allows us to easily generate the intermediate meshes, which vary both in body shape and in pose, by linearly interpolating between corresponding vertex locations. Texture transfer is also immediate when meshes are in full alignment.

3.5 Deformation Modeling

From the pre-processing step, we have obtained a dataset of meshes consisting of a template mesh[3] $\mathcal{X}$, a set of example meshes of the same subject as the template but in different poses $\mathcal{A} = \{\mathcal{Y}^i\}$, and a set of example meshes of different subjects $\mathcal{B} = \{\mathcal{Y}^j\}$. All meshes share the same connectivity structure, with $V = 12,500$ vertices and $T = 25,000$ triangles in complete correspondence. The triangles are clustered into $P = 15$ body parts.

In order to model the deformations between example meshes, we need to choose an appropriate representation capable of encoding important shape properties like pose-dependent non-rigid deformations and body shape variations. Simple representations based directly on vertex coordinates in a global frame or on vertex displacements from a template shape fail to capture local shape properties and the relationship between neighboring vertices. More importantly, they make combining vertex deformations models of different types difficult.

The strength of SCAPE comes from the way it represents deformations, using shape deformation gradients instead of vertex displacements. This gives SCAPE the ability to model pose and body shape deformations separately and then combine the two different deformation models in a natural way.

3.5.1 Shape Deformation Gradients

In this section, we first define the shape deformation gradients in general as it applies to any given two triangular meshes in complete correspondence.

We want to model the deformations that morph a source mesh $\mathcal{X}$ into a target mesh $\mathcal{Y}$ in the training set. We use the *shape deformation gradients* to encode the deformations at a triangle level,

[3]Note that the template mesh used for modeling deformations can be different from the template mesh used for aligning meshes in Section 3.4.1.

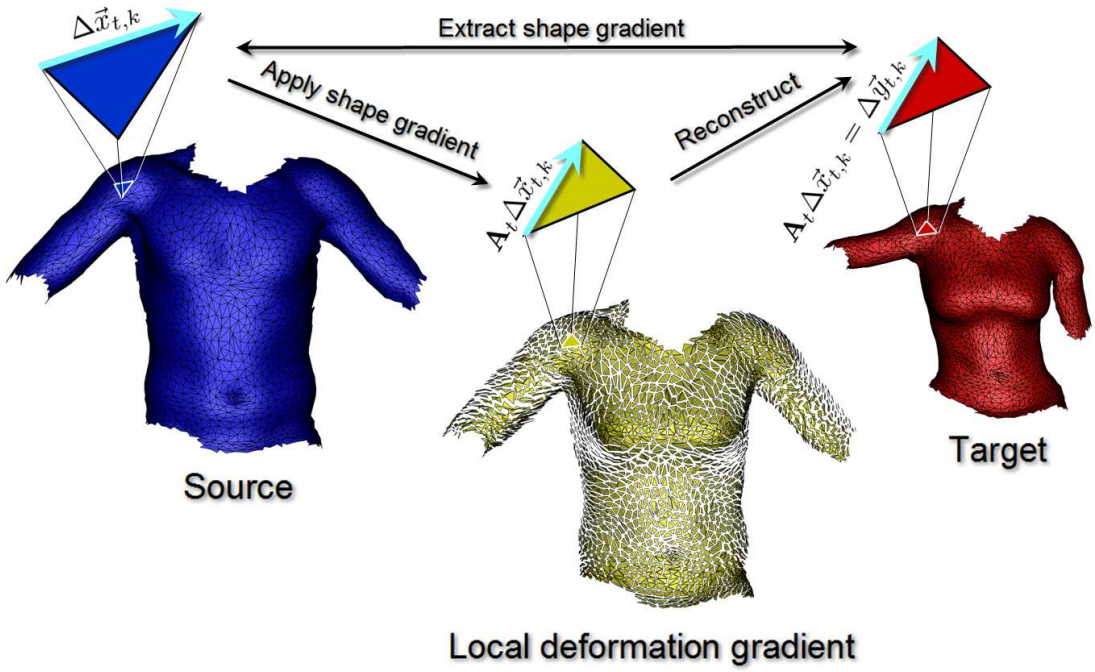


Figure 3.7: **Deformations Based on Shape Gradients.** Shape deformation gradients are the non-translational component $\mathbf{A}_t$ of an affine transformation that align the edge vectors $\Delta\vec{x}_{t,k}$ of a source mesh (left) to a target mesh (right). Since the shape deformation gradients are translation invariant, applying the deformations to each triangle of the source mesh independently results in a disconnected triangle "soup", but with the desired orientation, scale and skew (middle). A consistent mesh (right) can be obtained by solving a least squares problem over the shared triangle vertices which implicitly encodes the mesh connectivity.

using inspiration from the work of Sumner and Popović (2004) on mesh deformation transfer.

The deformation gradients are based on the affine transformations that align corresponding triangles between $\mathcal{X}$ and $\mathcal{Y}$. For a given triangle t of the source mesh $\mathcal{X}$ containing the vertices $(\vec{x}_{t,1}, \vec{x}_{t,2}, \vec{x}_{t,3})$ and the corresponding triangle of the target mesh $\mathcal{Y}$ with vertices $(\vec{y}_{t,1}, \vec{y}_{t,2}, \vec{y}_{t,3})$, we consider an affine transformation defined by a 3×3 matrix $\mathbf{A}_t$ and a displacement vector $\vec{t}_t$ such that

$$\mathbf{A}_t\vec{x}_{t,k} + \vec{t}_t = \vec{y}_{t,k}\,, \qquad k \in \{1,2,3\}\,. \tag{3.6}$$

By substructing the first equation from the others to eliminate the translation component $\vec{t}_t$ we obtain

$$\mathbf{A}_t\,(\vec{x}_{t,k} - \vec{x}_{t,1}) = \vec{y}_{t,k} - \vec{y}_{t,1}\,, \qquad k \in \{2,3\}\,, \tag{3.7}$$

which can be rewritten in matrix form as

$$\mathbf{A}_t\,[\Delta\vec{x}_{t,2}\ ,\ \Delta\vec{x}_{t,3}] = [\Delta\vec{y}_{t,2}\ ,\ \Delta\vec{y}_{t,3}]\ , \tag{3.8}$$

where the Δ operator computes the edge vector: $\Delta\vec{x}_{t,k} = \vec{x}_{t,k} - \vec{x}_{t,1}$.

The deformation gradient for each triangle is given by $\mathbf{A}_t$, the non-translational component of the affine transformation, which encodes only the local change in orientation, scale and skew induced

by the deformation of the triangle edges (Figure 3.7). The set of deformation gradients tabulated for each triangle provide only an intrinsic representation of the mesh geometry. Therefore we need address two problems: how to compute the deformations for example meshes and how to reconstruct meshes from shape gradients.

Since for a given triangle only two of its edges are actually constraining the shape gradient in Eq. 3.8, $\mathbf{A}_t$ is not uniquely determined. Sumner and Popović (2004) propose adding an *ad hoc* forth vertex to each triangle along the direction perpendicular to the triangle plane to implicitly add a third constraint for the subspace orthogonal to the triangle. We follow a more principled approach and regularize the solution by introducing a smoothness constraint which prefers similar deformations in adjacent triangles. We formulate a least-squares linear regression problem and solve for all deformations gradients at once:

$$\underset{\{\mathbf{A}_1,\cdots,\mathbf{A}_T\}}{\arg\min} \sum_{t=1}^{T} \sum_{k=2,3} ||\mathbf{A}_t \Delta\vec{x}_{t,k} - \Delta\vec{y}_{t,k}||^2 + w_s \sum_{t_1,t_2 \text{ adj}} ||\mathbf{A}_{t_1} - \mathbf{A}_{t_2}||_F^2 \,. \tag{3.9}$$

This approach effectively removes high-frequency noise from the mesh geometry and leads to better generalization when modeling the deformations in subsequent steps.

We now consider the problem of reconstructing a mesh from a set of shape gradients. Due to the local nature of the deformations gradients that contain no notion of translation or connectivity, if we were to transform individual triangles of the template by the corresponding deformation $\mathbf{A}_t$, we would get inconsistent edge vectors and unknown placement with respect to the neighbor triangles (Figure 3.7). Given a set of deformation gradients, reconstructing a consistent mesh requires solving a linear least squares problem over shared vertex coordinates:

$$\underset{\{\vec{y}_1,\cdots,\vec{y}_V\}}{\arg\min} \sum_{t=1}^{T} \sum_{k=2,3} ||\mathbf{A}_t \Delta\vec{x}_{t,k} - \Delta\vec{y}_{t,k}||^2 \,. \tag{3.10}$$

This approach computes the best possible connected mesh whose edges $\Delta\vec{y}_{t,k}$ are best aligned in a least-square sense to the individually predicted deformed edges within each triangle $\mathbf{A}_t \Delta\vec{x}_{t,k}$. Since all constraints are local, a global translational degree of freedom remains over the entire mesh. This can be accounted for by anchoring one of the vertices of the target mesh (assuming that the mesh is a single connected component).

We have shown how to extract shape deformation gradients from the example shapes and how to reconstruct meshes from a set of deformation gradients. We note however that for a mesh topology with $25,000$ triangles, each associated with a 3×3 matrix $\mathbf{A}_t$, the space of shape deformations is very high dimensional with $225,000$ dimensions and highly redundant.

In order to reduce the dimensionality, we take advantage of the deformation gradients being local and translation invariant and decouple the deformation transformations into a rigid and a non-rigid pose component and a body shape component, each modeled independently. We then express the triangle deformations as a sequence of linear transformations

$$\mathbf{A}_t = \mathbf{R}_{p[t]} \mathbf{D}_t \mathbf{Q}_t \,. \tag{3.11}$$

$\mathbf{Q}_t$ is a 3×3 linear transformation matrix specific for triangle t corresponding to non-rigid pose-dependent deformations such as muscle bulging. $\mathbf{D}_t$ is a linear transformation matrix corresponding to changes in body shape between individuals and is also triangle specific. Finally, $\mathbf{R}_{p[t]}$ is a rigid rotation matrix applied to the articulated skeleton and specific to the body part p containing the triangle t. We discuss each component next.

3.5.2 Articulated Rigid Deformations

We begin by addressing the articulated pose deformations. Let's assume that each body part is moving rigidly during changes in pose. In this case, the triangles experience only changes in orientation. The deformations can be modeled using rotation matrices. We start by defining 3×3 rotation matrices $\mathbf{R}_{p[t]}$ for each of the 15 body parts, where the triangles belonging to the same part experience the same rotation. Optimal rotation matrices for each body part can be computed in closed form using the known point correspondences between the template mesh $\mathcal{X}$ and example meshes $\mathcal{Y}$ (see Appendix C.1).

By letting $\mathbf{A}_t = \mathbf{R}_{p[t]}$, we can attempt to reconstruct a mesh $\hat{\mathcal{Y}}$ using

$$\underset{\{\vec{y}_1,\cdots,\vec{y}_V\}}{\arg\min} \sum_{t=1}^{T} \sum_{k=2,3} \left|\left|\mathbf{R}_{p[t]} \Delta\vec{x}_{t,k} - \Delta\vec{y}_{t,k}\right|\right|^2 . \tag{3.12}$$

In Figure 3.8 we show one of the example meshes (b), for which we have computed the rotation matrices for each body part relative to a template mesh (a), and its reconstruction (c) based on these matrices. We note that the reconstruction captures well the overall articulated structure of the body considering that we have reduced the dimensionality of the shape representation from $225,000$ to 45 (the rotation matrix for each of the 15 body parts has 3 degrees of freedom). Nonetheless, such a model is too simplistic and needs to be extended to account for non-rigid deformations associated with complex joints such as the shoulder, muscle bulging, and local deformation of soft tissue during pose changes.

3.5.3 Non-rigid Pose-dependent Deformations.

We use the set of example meshes in different poses $\mathcal{A} = \{\mathcal{Y}^i\}$ to train a non-rigid pose-dependent deformation model. Since the 70 meshes in the *pose set* belong to the same person as the template, the resulting deformations can only be attributed to pose changes. We let the body shape deformation $\mathbf{D}_t^i$ be the identity matrix $\mathbf{I}_3$ in Equation 3.11 and write the triangle deformations for each example mesh i as

$$\mathbf{A}_t^i = \mathbf{R}_{p[t]}^i \mathbf{Q}_t^i , \tag{3.13}$$

where $\mathbf{Q}_t^i$ is the residual triangle deformation after accounting for the part-based rigid rotation $\mathbf{R}_{p[t]}^i$, computed using point correspondences for each part (Appendix C.1). We can estimate the $\{\mathbf{Q}_t^i\}$

(a) (b) (c) (d)

Figure 3.8: **Articulated Rigid and Non-rigid Deformations.** (a) Template mesh. (b) Example scanned mesh for which we compute the part-based rotation matrices $\mathbf{R}_{p[t]}$. (c) Reconstructed mesh that assumes only articulated rigid deformations, using the rotation matrices as shape gradients (Equation 3.12). Since non-rigid deformations are not captured at this stage, significant artifacts remain mainly at the joints. The torso and shoulders retain the orientation of the template as the arms are raised, making the skin fold on the boundary between adjacent body parts. (d) Reconstructed mesh using articulated rigid and predicted non-rigid pose-dependent deformations (Equation 3.16), closely resembling the example mesh in (b), albeit slightly smoother in regions such as armpits, abdomen and knees.

matrices for each mesh $\mathcal{Y}^i$ directly from training data using the same idea as in Equation 3.9:

$$\underset{\{\mathbf{Q}_1^i,\cdots,\mathbf{Q}_T^i\}}{\arg\min} \sum_{t=1}^{T} \sum_{k=2,3} \left|\left| \mathbf{R}_{p[t]}^i \mathbf{Q}_t^i \Delta\vec{x}_{t,k} - \Delta\vec{y}_{t,k}^i \right|\right|^2 + w_s \sum_{\substack{t_1,t_2 \text{ adj} \\ p[t_1]=p[t_2]}} ||\mathbf{Q}_{t_1}^i - \mathbf{Q}_{t_2}^i||_F^2 \,. \tag{3.14}$$

As before, since the $\mathbf{Q}$ matrices are not fully constrained by the triangle edge vectors, we regularize the solution by adding smoothing constraints that require similar deformations in adjacent triangles. However, due to the foldings occurring at the boundary between adjacent body parts, as noted in Figure 3.8c, we do not enforce smoothing constraints across the boundaries, bur rather on adjacent triangles belonging to the same part. The smoothing factor w_s is set to equal 0.001ρ, where ρ is the mesh resolution, computed as the median value of the template mesh edge lengths.

Similar to [Anguelov (2005)], we assume that the non-rigid deformations $\mathbf{Q}$ can be expressed as a linear function of the pose parameters $\mathbf{R}$. We use the example set of non-rigid deformations to learn prediction models of these deformations given arbitrary new poses not present in the training set. Such non-rigid deformations are induced by rotations of the joints whose effect is localized to the body parts connected to moving joints. We learn a linear regression function for each triangle t which expresses the transformation matrix $\mathbf{Q}_t$ as a function of the relative joint rotations at the

closest joints. Let $\mathcal{N}_{p[t]}$ be the list of body parts connected through a joint to the limb containing triangle t. We compute the relative joint rotation $\Delta\mathbf{R}_{(p[t],c)}$ for each joint between the limbs $p[t]$ and c, $c \in \mathcal{N}_{p[t]}$, from the absolute part rotations $\mathbf{R}_{p[t]}$ and $\mathbf{R}_c$ using

$$\Delta\mathbf{R}_{(p[t],c)} = \mathbf{R}_{p[t]}^{\mathsf{T}}\mathbf{R}_c. \tag{3.15}$$

Using the axis-angle representation of a rotation, we encode each joint rotation $\Delta\mathbf{R}_{(p[t],c)}$ as a 3-element column vector $\Delta\vec{\omega}_{(p[t],c)}$ (see Appendix B.4). For each body part $p[t]$ we concatenate all adjacent joint rotations into a tall column vector $\Delta\vec{w}_{\left(p[t],\mathcal{N}_{p[t]}\right)}$.

Each of the 9 values of the 3×3 matrix $\mathbf{Q}_t$ is denoted by $q_{t,lm}$, with $l, m \in \{1, 2, 3\}$. We express the elements of the non-rigid deformation matrices as a linear function of the relative joint rotations adjacent to part $p[t]$:

$$q_{t,lm} = \mathbf{F}_{t,lm}^{\mathsf{T}} \cdot \begin{bmatrix} \Delta\vec{\omega}_{(p[t],\mathcal{N}_{p[t]})} \\ 1 \end{bmatrix}, \qquad l, m \in \{1, 2, 3\}. \tag{3.16}$$

We learn the linear coefficients $\mathbf{F}_{t,lm}$ from the example set of non-rigid deformations $\mathbf{Q}_t^i$ and pose parameters $\mathbf{R}_{p[t]}^i$ by solving a standard least-squares problem:

$$\underset{\mathbf{F}_{t,lm}}{\arg\min} \sum_i \left(\mathbf{F}_{t,lm}^{\mathsf{T}} \cdot \begin{bmatrix} \Delta\vec{\omega}^{\,i}_{(p[t],\mathcal{N}_{p[t]})} \\ 1 \end{bmatrix} - q_{t,lm}^i \right)^2. \tag{3.17}$$

Given a newly specified pose $\mathbf{R}'$, we compute for each limb the vector of adjacent relative rotations $\Delta\vec{\omega}'_{\left(p[t],\mathcal{N}_{p[t]}\right)}$, and obtain the corresponding non-rigid deformations $\mathbf{Q}^{\mathbf{F}}(\mathbf{R}')$ using Equation 3.16, where $\mathbf{F}$ is pre-computed from examples according to Equation 3.17. Note that this formulation of the relative rotation between parts does not require known joint locations or a skeleton-based pose representation. Also note that the original formulation [Anguelov (2005)] embeds the relative joint rotations in a lower dimensional subspace using Principal Component Analysis, motivated by the fact that certain joint angles exhibit less than three degrees of freedom. While this is a good idea in general, we feel that a sample size of 70 poses is insufficient for capturing a representative subspace. Absent a larger sample set, we forgo implementing this extra step.

Figure 3.8d shows a reconstructed mesh using just a set of pose parameters $\mathbf{R}$ from which non-rigid pose-dependent deformations are predicted. The non-rigid predictive model is able to capture shoulder and elbow deformations much better than in the articulated rigid case (c), although some smoothing in the arm pit areas, knees and abdomen remains. Additional synthesized results of shapes in various poses not in the training dataset are presented in Figure 3.12.

3.5.4 Alternative Pose Parameterization

While the current parameterization of articulated pose is 45-dimensional, consisting of 3 degrees of freedom specifying global orientation for each of the 15 body part rotations, it is still too broad, allowing for pose configurations that are anatomically infeasible. With applications involving pose estimation in mind, we may seek a more restrictive search space. We can model the skeleton of

the body as a 3D kinematic tree and parameterize $\mathbf{R}$ in terms of the relative joint angles between neighboring limbs and the orientation of the root part, jointly denoted by $\vec{\theta}$. In certain scenarios, we may still need to model the hips, shoulders and pelvis as ball and socket joints with 3 DoF, but could otherwise assume the knees, ankles, elbows, wrists and head to be hinge joints with 1 DoF, effectively reducing the dimensionality to 27D.

Such parameterization makes it easy to impose anatomical limits on joint angles, although one can define more elaborate pose priors using motion capture training data (e. g., [Sidenbladh *et al.* (2000); Urtasun *et al.* (2006)]).

3.5.5 Non-rigid Body Shape Deformations.

We turn our attention to modeling deformations due to the changes in body shape between individuals. The shape of a person is changed by applying a linear 3×3 shape deformation $\mathbf{D}_t$ to each triangle in the mesh. Given a template mesh aligned with example bodies, the deformation for each triangle in the template is computed to the corresponding triangle in each example mesh after accounting for variations in pose. A low-dimensional, parametric, model is sought that characterizes these variations within a population of people.

We use the set of example meshes of different people $\mathcal{B} = \{\mathcal{Y}^j\}$ as training data. As before, for each mesh $\mathcal{Y}^j$ we first estimate the rigid alignment $\mathbf{R}^j$ between parts point correspondences and use those to predict the pose-dependent deformation $\mathbf{Q}^{\mathbf{F}}(\mathbf{R}^j)$ with the linear mapping from Equation 3.16. After accounting for pose related deformations, the residual triangle deformations can be attributed to changes in body shape. We factor out the deformation gradient for a triangle t as

$$\mathbf{A}_t^j = \mathbf{R}_{p[t]}^j \mathbf{D}_t^j \mathbf{Q}^{\mathbf{F}}(\mathbf{R}^j) \ . \tag{3.18}$$

Estimating the residual body shape example deformations $\mathbf{D}_t^j$ for each mesh $\mathcal{Y}^j$ involves solving the usual least-squares problem

$$\underset{\{\mathbf{D}_1^j, \cdots, \mathbf{D}_T^j\}}{\arg\min} \sum_{t=1}^{T} \sum_{k=2,3} \left|\left| \mathbf{R}_{p[t]}^j \mathbf{D}_t^j \mathbf{Q}^{\mathbf{F}}(\mathbf{R}^j) \, \Delta\vec{x}_{t,k} - \Delta\vec{y}_{t,k}^j \right|\right|^2 + w_s \sum_{t_1, t_2 \text{ adj}} ||\mathbf{D}_{t_1}^j - \mathbf{D}_{t_2}^j||_F^2 \tag{3.19}$$

whose solution is regularized by including smoothing constraints that enforce similar deformations in adjacent triangles. The smoothing factor w_s is set to equal the mesh resolution ρ, computed as the median value of the template mesh edge lengths.

For a given mesh $\mathcal{Y}^j$, the body shape deformations $\mathbf{D}_t^j$ for all T triangles can be concatenated into a single column vector $\vec{d}^j$ of size $(3 \cdot 3 \cdot T) \times 1$, and every example body $\mathcal{Y}^j$ in the training set $\mathcal{B}$ becomes a column in a matrix of deformations: $\mathbf{D}^{\mathcal{B}} = \left[\cdots, \vec{d}^j, \cdots\right]$. Specifically, for a mesh with $T = 25,000$ triangles, the body shape is specified using $225,000$ parameters that are highly correlated. We use principal component analysis (PCA)[4] to find a reduced-dimension subspace that

[4] Our implementation uses incremental principal component analysis (iPCA) [Brand (2002)] to cope with computer memory limitations. Additional details are provided in Appendix D.

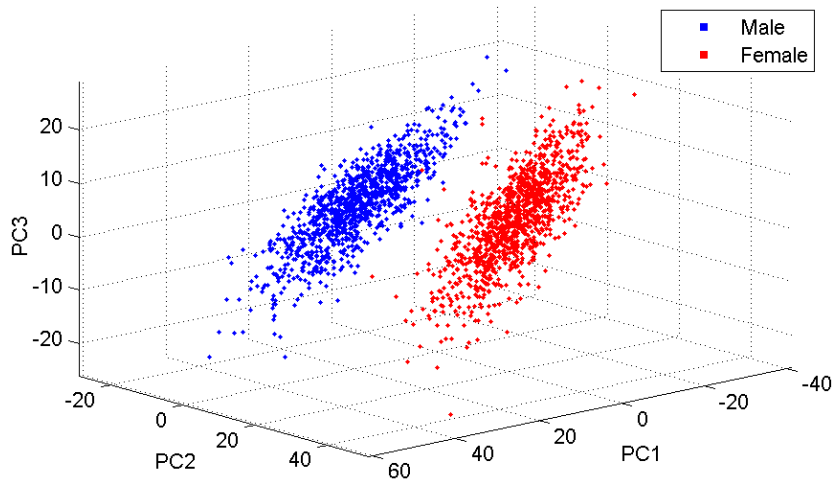


Figure 3.9: **PCA Gender Separation.** Analyzing body shape variations of the aggregate dataset of men and women. When the shape data is projected onto the first 3 principal components (PC), the plot reveals the presence of two clusters induced by gender.

retains most of the variance in how body shapes deform (see Appendix D):

$$\mathbf{D}^{\mathcal{B}} \xrightarrow{\text{PCA}_r} \mathbf{UB} + \vec{\mu}\,\vec{1}_{1\times|\mathcal{B}|}\,. \tag{3.20}$$

Here $\vec{\mu}$ is the mean body shape deformation and the columns of the PCA basis matrix $\mathbf{U}$ are the first r principal components given by PCA which are the directions of maximum variance in the training data. The matrix $\mathbf{B}$ constitutes a compact representation of the shapes in the training data using only r parameters per body. Each vector of shape deformations $\vec{d}^{\,j}$ in the training set is approximated by

$$\hat{\vec{d}}^{\,j} = \mathbf{U}\vec{\beta}^{j} + \vec{\mu}\,, \tag{3.21}$$

where $\vec{\beta}^{j}$ is a vector of r linear coefficients that characterizes a given shape j. The variance of each shape coefficient $\vec{\beta}_b$ is given by the eigen-value $\sigma^2_{\beta,b}$ obtained by PCA (Equation D.11). Note that this formulation effectively provides a prior on body shape in the form of a Gaussian distribution over the shape coefficients $\vec{\beta}_b$ with estimated mean $\vec{\mu}_b$ and variance $\sigma^2_{\beta,b}$.

Gender Separation

One immediate application of PCA is to visualize the training data by projecting the shape data onto the first three principal components. In Figure 3.9 each body in the CAESAR dataset becomes a 3D point. We discover that the aggregate shape data forms two separate clusters induced by gender, confirming that men and women have very distinctive body shapes. Because the shape data over the entire population is clearly non-Gaussian, PCA will not find axes of variation that are independent; in this case it just de-correlates the dimensions. To alleviate this, we depart from the original SCAPE formulation and learn separate eigen-models for men and women respectively in addition to the gender-neutral model with all the subjects combined. We use the variable χ to

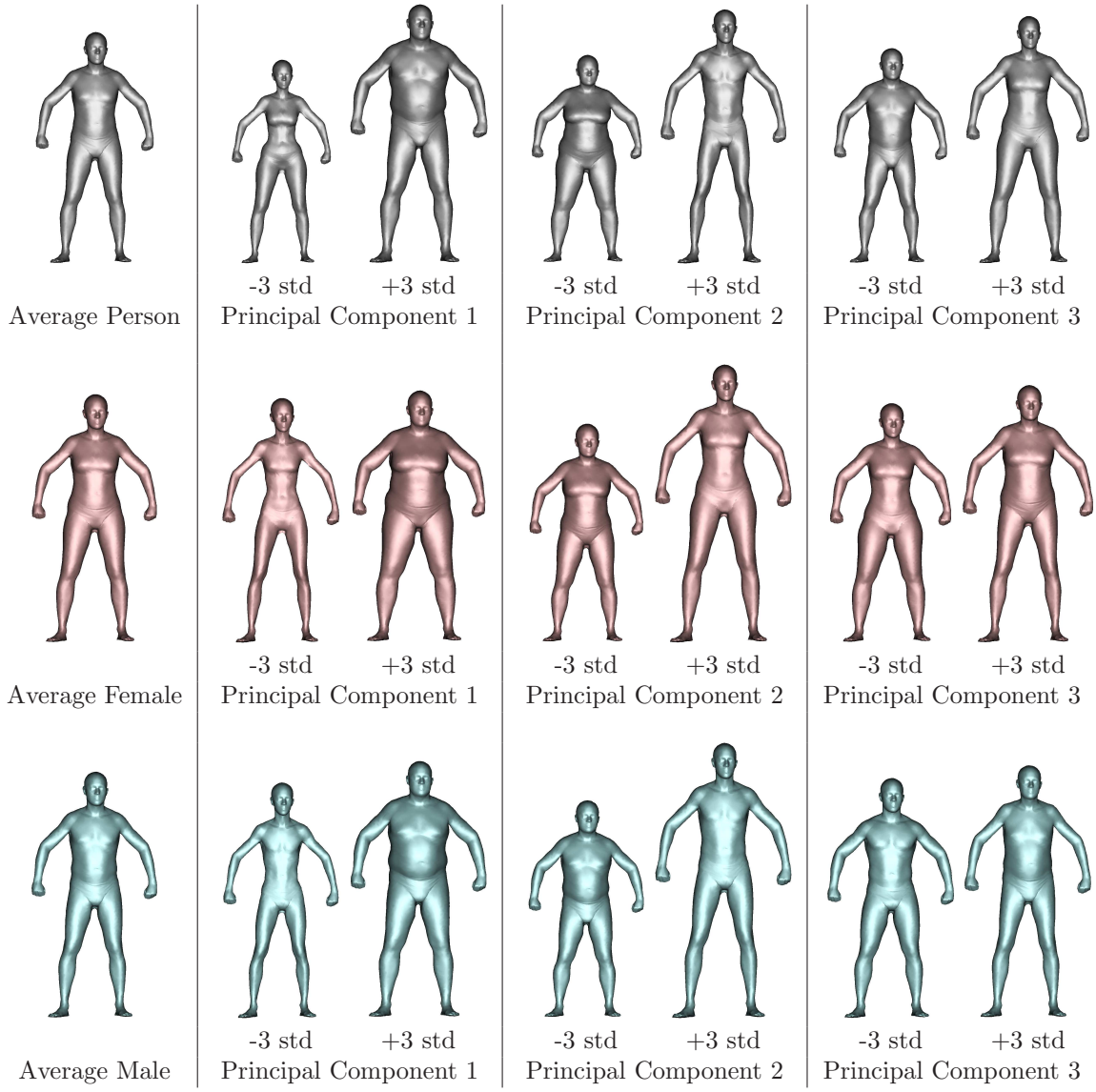
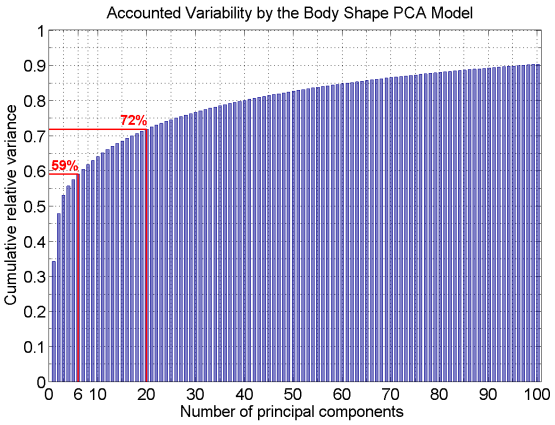


Figure 3.10: **PCA Shape Bases.** The mean body shape (*left*) followed by deviations from the mean along the first three principal components (± 3 standard deviations) are illustrated for three different eigen-shape models. The gender-neutral shape model is demonstrated on the top row, the female-only model in the middle, and the male-only model on the bottom row.

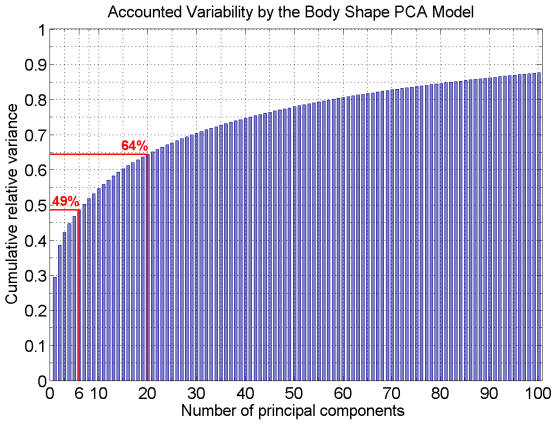
denote the corresponding eigen-shape deformation model:

$$\left(\mathbf{U}^{\chi}, \vec{\mu}^{\chi}, \sigma^2_{\vec{\beta},\chi}\right), \qquad \chi \in \{\text{male}, \text{female}, \text{neutral}\}.$$

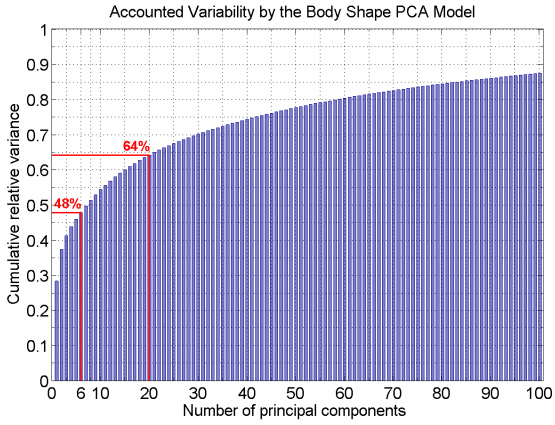
For the remainder of the thesis, whenever the choice of gender model can either be inferred from the context or is not critical to the discussion, the χ gender superscript is omitted.



(a) Gender-neutral Shape Model



(b) Female Shape Model



(c) Male Shape Model

Figure 3.11: **Accounted Shape Variability by PCA.** Principal Component Analysis (PCA) aligns the data with the directions of maximum variance (also called principal components). Dimensionality reduction is achieved by retaining a few of the directions of maximum variance and throwing away the rest, thus expressing shapes compactly using a few shape coefficients. The bar plots above display the proportion of the entire variability in a given training dataset that is accounted for when retaining the top r principal components. We show bar plots for the entire CAESAR dataset combining over 1000 men and 1000 women (a), as well as separately for each gender (b,c).

Major Shape Variations

The PCA representation allows us to express new shapes not in the training data as deviations from a mean shape by specifying only a few shape coefficients $\vec{\beta}$

$$\vec{d}(\vec{\beta}) = \mathbf{U}\vec{\beta} + \vec{\mu}\,. \tag{3.22}$$

Figure 3.10 shows the mean shapes and deviations from the mean along the first three principal components. For example, the first principal component of the gender-neutral model captures an axis of variation from small, slim, women to tall, heavy, men. The shape bases account for variations in gender, height and weight, as well as more specific fat and muscle distributions over the body.

In contrast, the gender specific models keep the shape variations within the same gender class. The shape of a particular individual is then represented as a linear combination of several of these bases, whose coefficients form a pose-independent descriptor of the intrinsic body shape.

In Figure 3.11 we observe the relationship between the number of shape coefficients r and the proportion of the total variability accounted for. We note that 6 principal components account for about half of the total variability in the gender-specific models, while 20 principal components raise the proportion to 65%. We typically use between 6 and 20 bases, though more can be used to increase shape reconstruction accuracy. It would take more than 100 principal components to achieve a 90% level of captured variance, increasing the dimensionality of the shape representation considerably. Please note that while the gender-neutral PCA model appears more expressive, it is only an artifact of the aggregate shape data being split into two distant clusters which increases the overall variance.

3.5.6 New Body Mesh Generation

In the previous sections we have shown how to express pose and body shape deformations compactly in terms of a few parameters. Given new joint angles $\vec{\theta}$, shape coefficients $\vec{\beta}^{\chi}$ and gender χ, a new mesh $\mathcal{Y}$, not present in the training set, can be computed by solving

$$\mathcal{Y}(\chi, \vec{\beta}^{\chi}, \vec{\theta}) = \underset{\{\vec{y}_1, \cdots, \vec{y}_V\}}{\arg\min} \sum_{t=1}^{T} \sum_{k=2,3} \left|\left| \mathbf{R}_{p[t]}(\vec{\theta})\, \mathbf{D}_t^{\mathbf{U}^{\chi}, \vec{\mu}^{\chi}}(\vec{\beta}^{\chi})\, \mathbf{Q}_t^{\mathbf{F}}\Big(\mathbf{R}(\vec{\theta})\Big)\, \Delta\vec{x}_{t,k} - \Delta\vec{y}_{t,k} \right|\right|^2 . \tag{3.23}$$

This optimization problem can be expressed as a linear system that can be solved very efficiently using linear least-square regression techniques. We note that this formulation leaves unconstrained three translational degrees of freedom. Therefore the global position of the mesh also needs to be specified and, for notational convenience, these parameters are included in the parameter vector $\vec{\theta}$.

We demonstrate the expressiveness of the SCAPE model in Figure 3.12. It shows examples of synthesized meshes for different subjects in a variety of poses and with realistic skin deformations, none of which are represented in the training data. These meshes are specified compactly using joint angles derived from marker-based motion capture data and randomly sampled PCA coefficients from the gender-neutral shape model.

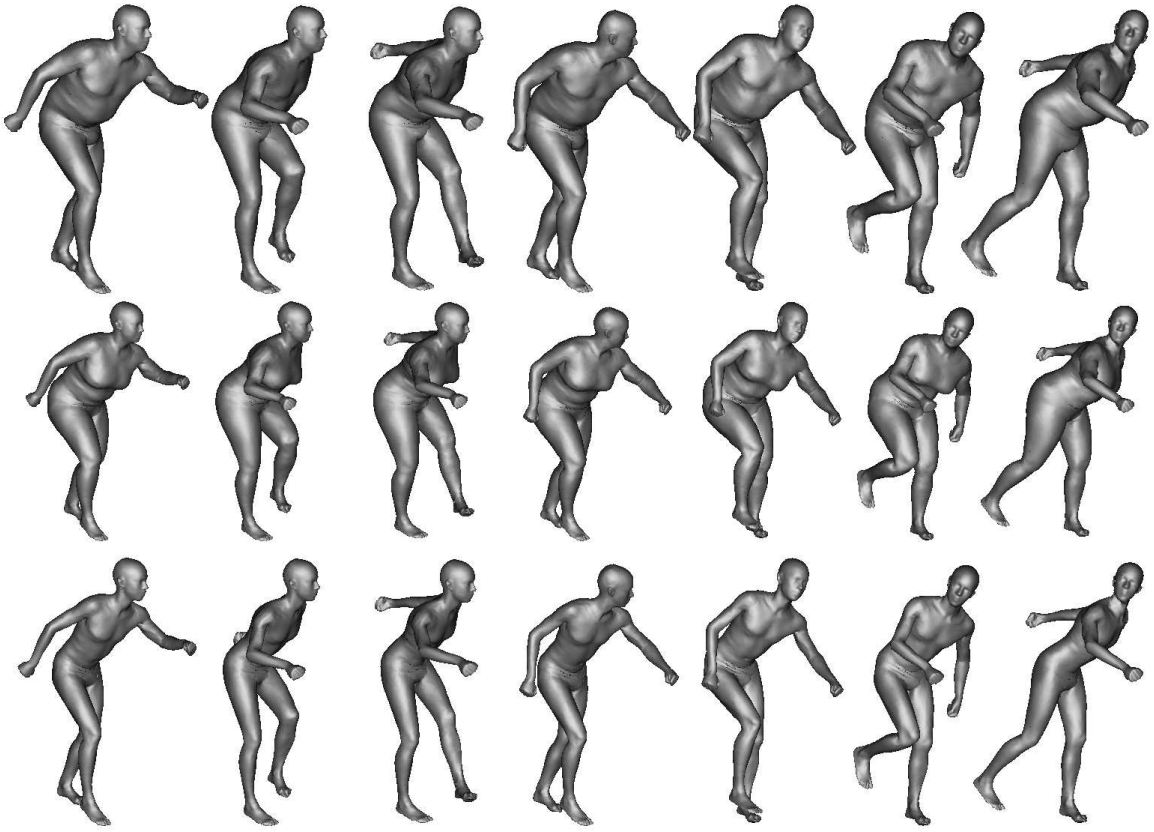


Figure 3.12: **SCAPE Animations.** Novel meshes synthesized using the SCAPE model. Each row represents a different individual dancing the Charleston. None of the individual shapes and poses are present in the SCAPE training sets.

Chapter 4

A Framework for Model Fitting to Images

4.1 Introduction

This chapter describes our basic approach to estimating the pose and shape of a person from multi-camera images. Much of the research on video-based human motion capture assumes the body shape is known *a priori* and is represented coarsely (e.g. using cylinders or superquadrics to model limbs). These body models stand in sharp contrast to the richly detailed 3D body models used by the graphics community. Detailed models of 3D human shape are useful not only for graphics applications, but also for extracting accurate biometric measurements. Here we propose a method for recovering such models directly from images. Specifically, we represent the body using the SCAPE model described in Chapter 3. It employs a low-dimensional, but detailed, parametric model of shape and pose-dependent deformations that is learned from a database of range scans of human bodies. Previous work [Anguelov *et al.* (2005b)] showed that the parameters of the SCAPE model could be estimated from marker-based motion capture data. Here we go further to estimate the parameters directly from image data.

Our current implementation estimates the parameters of the body model using image silhouettes computed from multiple calibrated cameras (typically 3-4). The learned model provides strong constraints on the possible recovered shape of the body which means that pose/shape estimation is robust to errors in the recovered silhouettes. Our generative model predicts silhouettes in each camera view given the pose/shape parameters of the model. A fairly standard Euclidean distance transform measure is used to define an objective function that we seek to optimize in terms of the pose and shape parameters. Our results show that the SCAPE model better explains the image evidence than does a more traditional coarse body model.

We obtain an automated method for recovering pose throughout an image sequence by using body models with various levels of complexity and abstraction. Here we exploit previous work on

3D human tracking using simplified body models. In particular, we adopt the approach of Deutscher and Reid (2005) which uses anneal particle filtering to track an articulated body model in which the limbs are approximated by simple cylinders or truncated cones. This automated tracking method provides an initialization for the full SCAPE model optimization. By providing a reasonable starting pose, it makes optimization of the fairly high-dimensional shape and pose space practical.

Our results show that such rich generative models can be used for automatic recovery of detailed human shape information from images. We compare the performance of the SCAPE model with a standard cylindrical body model and show that a more realistic body representation improves the accuracy of human pose estimation from images. Results are presented for multiple subjects (none of whom were present in the SCAPE training data) in various poses.

4.2 Related Work

We exploit the SCAPE model of human shape and pose deformation [Anguelov *et al.* (2005b)] but go beyond previous work to estimate the parameters of the model directly from image data. Previous work [Anguelov *et al.* (2005b)] estimates the parameters of the model from a sparse set of 56 markers attached to the body. The 3D locations of the markers are determined using a commercial motion capture system and provide constraints on the body shape. Pose and shape parameters are estimated such that the reconstructed body is constrained to lie inside the measured marker locations. Closely related is the work of Park and Hodgins (2006) who are able to capture more detailed human skin deformations by using a much larger set of markers ($\sim$350). Both methods assume that a 3D scan of the body is available. This scan is used to place the markers in correspondence with the surface model of the subject.

We go beyond these methods to estimate the pose and shape of a person directly from image measurements. This has several advantages. In particular, video-based shape and pose capture does not require markers to be placed on the body. Additionally, images provide a richer source of information than a sparse set of markers and hence provide stronger constraints on the recovered model. Furthermore, we show shape recovery from multi-camera images for subjects not present in the shape training set.

Previous methods have established the feasibility of estimating 3D human shape and pose directly from image data but have all suffered from limited realism in the 3D body models employed. A variety of simplified body models have been used for articulated human body pose estimation and tracking including cylinders or truncated cones (e.g. [Deutscher and Reid (2005)]) and various deformable models such as superquadrics [Gavrila and Davis (1996); Pentland and Horowitz (1991); Sminchisescu and Triggs (2003)] and free-form surface patches [Rosenhahn *et al.* (2006)]. These models do not fit the body shape well, particularly at the joints and were typically built by hand [Pentland and Horowitz (1991)] or estimated in a calibration phase prior to tracking [Gavrila and Davis (1996); Rosenhahn *et al.* (2006); Sminchisescu and Triggs (2003)]. Detailed but fixed, person-specific, body models have been acquired from range scans and used for tracking [Mündermann *et al.*

(2006)] by fitting them to voxel representations; this approach did not model the body at the joints. Gavrila and Davis (1996) fit tapered superquadrics to the limbs by using two specific calibration poses.

Kakadiaris and Metaxas used generic deformable models to estimate 3D human shape from silhouette contours taken from multiple camera views [Kakadiaris and Metaxas (1998)] and tracked these shapes over multiple frames [Kakadiaris and Metaxas (2000)]. Their approach involved a 2-stage process of first fitting the 3D shape and then tracking it. In contrast, pose and shape estimation are performed simultaneously in our method. Their experiments focused on upper-body tracking in simplified imaging environments in which near-perfect background subtraction results could be obtained.

In related work Plänkers and Fua (2001a) defined a "soft" body model using 3D Gaussian blobs arranged along an articulated skeletal body structure. The relative shapes of these "metaballs" were defined *a priori* and were then scaled for each limb based on an estimated length and width parameter for that limb. Left and right limbs were constrained to have the same measurements. The surface of the body model was then defined implicitly as a level surface and an iterative optimization method was proposed to fit each limb segment to silhouette and stereo data. Most experiments used only upper body motion with simplified imaging environments, though some limited results on full body tracking were reported in [Plänkers and Fua (2001b)].

Also closely related to the above methods is the work of Hilton *et al.* (1999, 2000) who used a VRML body model. Their approach required the subject to stand in a known pose for the purpose of extracting key features from their silhouette contour which allowed alignment with the 3D model. Their model has a similar complexity to ours ($\sim$20K polygons) but lacks the detail of the learned SCAPE model.

In these previous models the limb shapes were modeled independently as separate parts. This causes a number of problems. First, this makes it difficult to properly model the shape of the body where limbs join. Second, the decoupling of limbs means that these methods do not model pose dependent shape deformations (such as the bulging of the biceps during arm flexion). Additionally none of these previous method automatically estimated 3D body shape using learned models. Learning human body models has many advantages in that there are strong correlations between the size and shape of different body parts; the SCAPE model captures these correlations in a relatively low-dimensional body model. The result is a significantly more realistic body model which both better constrains and explains image measurements and is more tolerant of noise. In previous work, generic shape models could deform to explain erroneous image measurements (e.g. one leg could be made fatter than the other to explain errors in silhouette extraction). With the full, learned, body model, information from the entire body is combined to best explain the image data, reducing the effect of errors in any one part of the body; the resulting estimated shape always faithfully represents a natural human body. The SCAPE representation generalizes (linearly) to new body shapes not present in the training set.

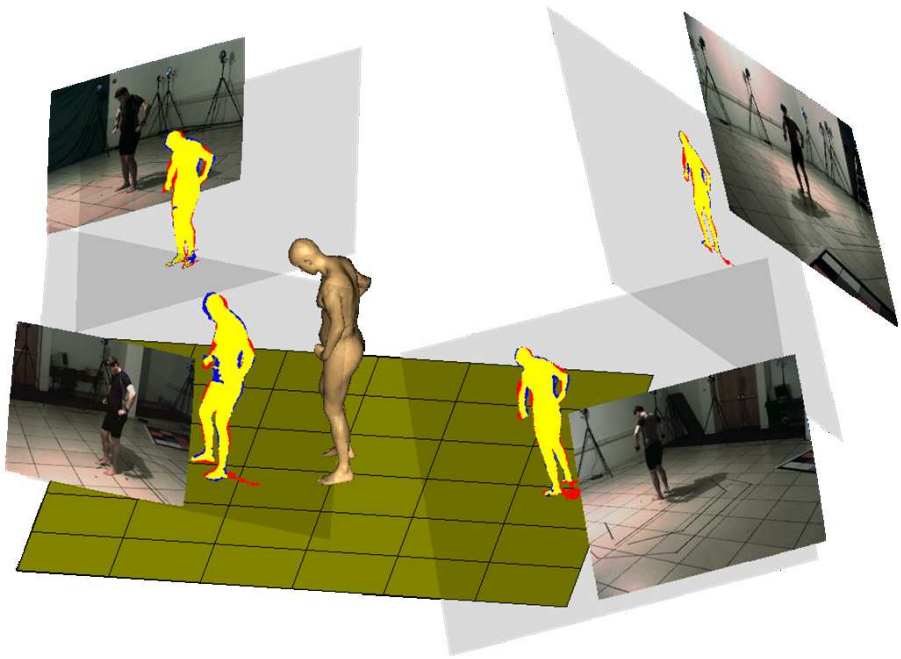


Figure 4.1: **SCAPE from Images.** The estimation of the body pose and shape parameters using image silhouettes (red) extracted using a known background. The estimation involves searching for the SCAPE parameters such that the body model projected into each image (blue) best overlaps (yellow) the observed image silhouettes.

Finally, there have been several non-parametric methods for estimating detailed 3D body information using voxel representations and space carving [Cheung *et al.* (2000, 2005); Chu *et al.* (2003); Mikić *et al.* (2003)]. While flexible, such non-parametric representations require further processing for many applications such as joint angle extraction or graphics animation. The lack of a parametric shape model means that it is difficult to enforce global shape properties across frames (e.g. related to the height, weight and gender of the subject). Voxel representations are typically seen as an intermediate representation from which one can fit other models [Mündermann *et al.* (2006); Starck and Hilton (2003)]. Here we show that a detailed parametric model can be estimated directly from the image data.

4.3 System Overview

In this chapter we describe a generic system for capturing the pose and shape of a person from image data. We also provide alternative instantiations of the system in the following two chapters. The major components of the system are given by: 1. environment instrumentation and data acquisition; 2. data pre-processing; and 3. model fitting to image observations. Figure 4.1 illustrates the process.

Sensor data is acquired using one or more digital cameras that have been calibrated and synchronized. In the pre-processing step, images are segmented into regions of interest, including foreground and background regions, though other image features may also be extracted to assist in the body model estimation. Finally, using information about the precise camera placement in the world, a 3D parametric model is matched to the image observations such as foreground silhouettes.

Environment instrumentation refers to the process of controlling the environment to simplify data pre-processing and model fitting. Image segmentation is easier when assuming a static known background and static cameras or when chroma-keying the environment. We also initially assume the subject wears tight-fitting clothing. This is because clothing is not represented by the SCAPE body model. We relax this assumption in Chapter 6. Furthermore, it is common to acquire images from multiple cameras simultaneously using hardware synchronization to ensure the pose has not changed between views, while camera calibration is performed so that matching the 3D model to images is done in a common coordinate system.

4.4 Camera Model and Calibration

In order to extract precise measurements from 2D image data and to combine information from multiple camera views, it is common for many computer vision applications to assume that the cameras have been calibrated in a pre-processing step. Camera calibration is the process of inferring certain properties of the cameras which relate 3D locations in the world to corresponding 2D projection locations in images.

Following a pinhole camera model with no lens distortion, a 3D point P in the scene with world coordinates $[x, y, z]^\mathsf{T}$ is mapped to a 2D image point with sub-pixel coordinates $[u, v]^\mathsf{T}$ using the standard camera projection equation in homogeneous coordinates [Hartley and Zisserman (2004)]:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K}_c \begin{bmatrix} \mathbf{R}_c & \vec{t}_c \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \tag{4.1}$$

where λ is an arbitrary scaling parameter related to depth, and $\mathbf{K}_c$ is the camera intrinsic parameter matrix encoding focal length (f_u, f_v), principal point (c_u, c_v) and skew coefficient α:

$$\mathbf{K}_c = \begin{bmatrix} f_u & \alpha & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.2}$$

The camera extrinsic parameters are represented by the rotation matrix $\mathbf{R}_c \in SO(3)$ and translation vector $\vec{t}_c \in \mathbb{R}^3$, encoding a rigid body transformation of the camera in the world coordinate system (Appendix B).

We use the Camera Calibration Toolbox for Matlab [Bouguet (2000)] and a planar checkerboard pattern to estimate the internal and external parameters of the cameras. This procedure relies on

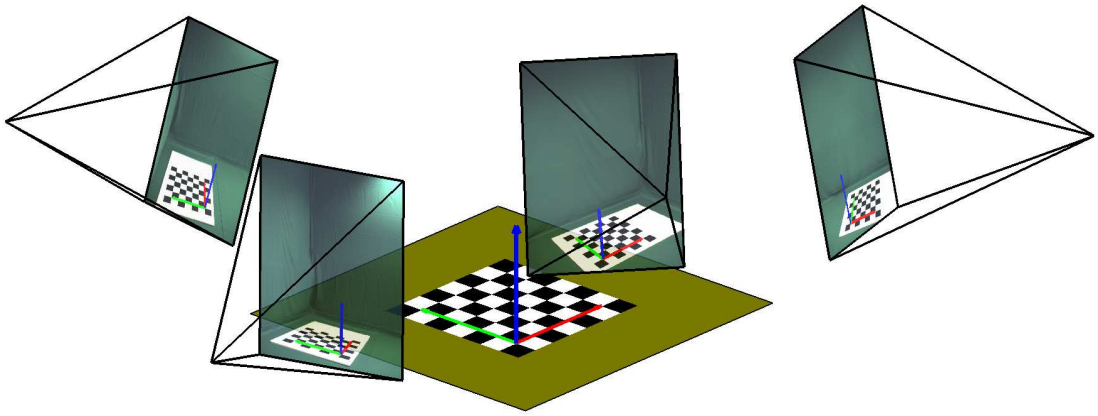


Figure 4.2: **Camera Calibration in a Controlled Environment.** A checkerboard pattern can be used to calibrate the cameras. The regular pattern provides easy correspondences between local 3D coordinates of the grid corners and the 2D pixel locations in each image. These correspondences are used to solve for the transformation that relates a camera local coordinate system to the grid local coordinate system. Having a static checkerboard pattern visible from all camera views helps in defining a common world coordinate system for all of the cameras. Note that by attaching the calibration pattern directly to planar surfaces in the scene, walls and the ground floor can also become calibrated in the world coordinate system. The same checkerboard pattern can be used to also estimate the camera intrinsic parameters (field of view, lens distortion, pixel skew, image center offset) using corner constraints derived from observing the grid at multiple orientations (at least 2) [Bouguet (2000)].

constraints derived from correspondences between 3D locations in the world and 2D pixel locations, manually selecting in each image the outer four corners of the calibration grid of known size, and discretizing the space between them to automatically detect all internal corners with sub-pixel accuracy. The internal parameters are estimated first by integrating constraints derived from observing the calibration grid at multiple orientations. Afterward, the extrinsic parameters for each of the cameras are estimated with respect to a common coordinate system by placing a single calibration grid on the floor viewable from all camera views (Figure 4.2). Furthermore, by attaching the calibration pattern to any planar surfaces in the scene, the position and orientation of these surfaces can also be expressed in the same common coordinate system. This can be useful for testing if the body model is floating in the air or is penetrating hard surfaces like the walls or the floor. We will be taking advantage of the known scene geometry in Chapter 5 for casting synthetic shadows in the scene.

The pinhole camera model presented above can be extended to model lens distortion effects, whose parameters are also estimated during calibration. Such a model introduces a high-order polynomial filter that depends on the distance from the principal axis of the lens. In order to reduce the computational complexity of algorithms that employ repetitive image projections, we reverse the process and rectify the images to remove any radial or tangential image distortions [Bouguet (2000)].

Assuming calibrated cameras and images that have been corrected for known lens distortions,

we provide a shorthand notation for the transformation in Equation 4.1 that maps any 3D point P in the world to a 2D image location $[u, v]^{\mathsf{T}}$ in the k^{th} camera view:

$$\text{Proj}_{C^k}(P) = [u, v]^{\mathsf{T}}. \tag{4.3}$$

4.5 Foreground Image Segmentation

We use foreground segmentation to identify the region in the image depicting the person. The most common approach is to use statistical measures of image difference between an image with and without a person present. For example, a standard method is to fit a Gaussian distribution (or mixture of Gaussians [Stauffer and Grimson (1999)]) to the variation of pixel values taken over several background images. For a new image with the person present, a statistical test is performed that evaluates how likely the pixel is to have come from the background model. Typically a probability threshold is set to classify the pixel. After individual pixels have been classified as foreground or background, several image processing operations can be applied to improve the segmentation, including dilation and erosion, median filtering, and removal of small disconnected components.

An extension of this approach that handles shadows robustly is described in detail in Section 5.5.

4.6 Problem Formulation

The SCAPE model is parameterized by a set of pose parameters $\vec{\theta}$, including global position and orientation, shape coefficients $\vec{\beta}$, and gender χ. The problem of estimating human body shape from image data is reduced to one of solving for the optimal body model parameters that minimize some error function $E(\chi, \vec{\beta}^{\chi}, \vec{\theta})$ given image measurements. Our approach uses foreground image silhouettes obtained from multiple calibrated cameras for estimating the body pose and shape parameters and assumes the subject wears minimal or tight fitting clothing. The framework is general however and can be augmented to exploit additional image features such as edges and shading [Guan *et al.* (2009)], shadows (Chapter 5), optical flow [Sminchisescu and Triggs (2003)], etc. A method for dealing with the more challenging case involving clothing is proposed in Chapter 6.

A generative approach is adopted in which predicted model parameters are used to construct a 3D body model from which a silhouette is computed and compared to the image silhouette. The model is projected into a camera view k assuming known extrinsic and intrinsic camera calibration (Equation 4.3), which produces a predicted foreground silhouette of the estimated model $F_k^e\left(\chi, \vec{\beta}^{\chi}, \vec{\theta}\right)$. This silhouette is compared with the observed silhouette, F_k^o, in camera view k, obtained by foreground segmentation (Section 4.5).

4.6.1 Silhouette Similarity Measure

Measures have been proposed in the literature for computing (dis)similarity of silhouettes. For instance, one of the most widely used measures is based on silhouette overlap, computed by summing

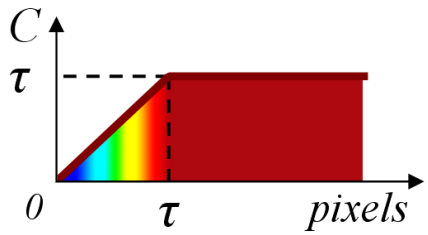


Figure 4.3: **Thresholded Distance Transform.** The silhouette similarity measure is made robust to outliers in the extracted image silhouette by thresholding the distance transform after τ pixels.

the non-zero pixels resulting from a pixel-wise XOR (exclusive OR) between the two image masks (predicted and observed). While computationally efficient, this measure is not very informative in guiding the search during optimization. Alternatively, the distance between the outline of one silhouette and the outline of the other and vice-versa can be used.

Our approach uses a modified version of the Chamfer distance. First an asymmetric distance from silhouette S to silhouette T is defined as

$$\tilde{d}^{\tau}(S,T) = \frac{\sum_{i,j} S_{ij} \cdot C^{\tau}_{ij}(T)}{\left(\sum_{i,j} S_{ij}\right)^{\xi}}, \tag{4.4}$$

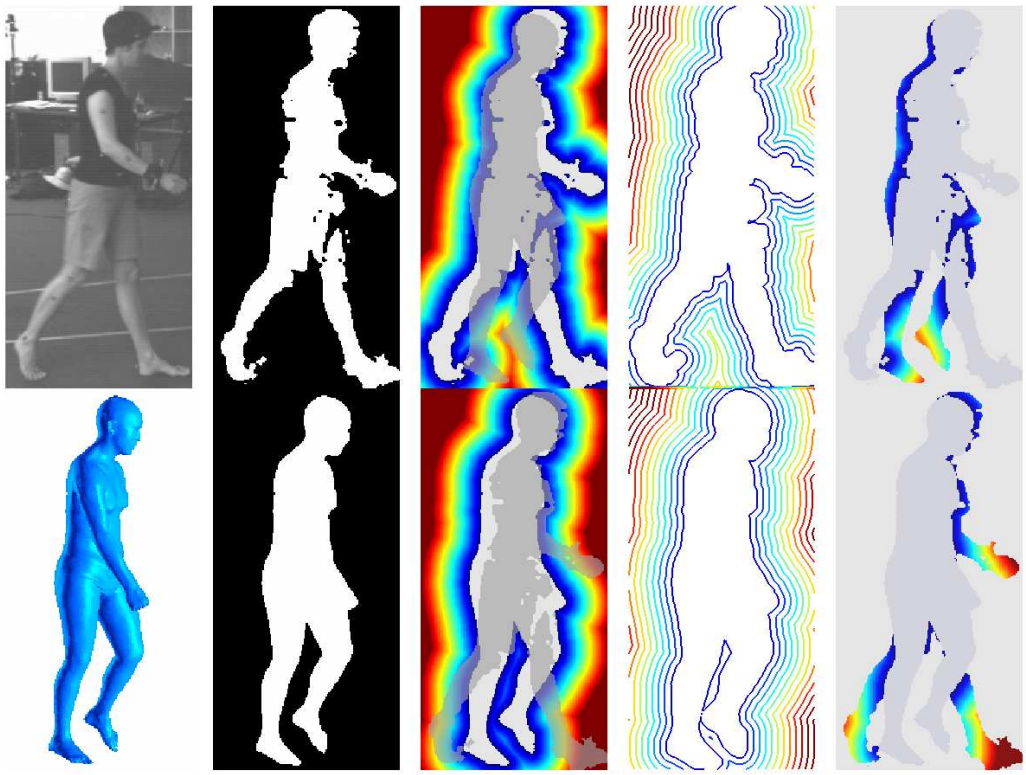
where $S_{ij} = 1$ for the pixels inside S and 0 otherwise; $C^{\tau}_{ij}(T)$ is a distance transform function which is zero if pixel (i,j) is inside T and is a robust Euclidean distance to the closest point on the boundary of T for points outside (see Figure 4.4c). In order to cope with errors in the image silhouettes, $C^{\tau}_{ij}(T)$ is made robust by capping the Euclidean distance at a certain threshold τ (e.g. 20 pixels for an image size of 800 by 600 pixels). As illustrated in Figure 4.3, for pixels (i,j) that are more than τ Euclidean distance away from T, $C^{\tau}_{ij}(T) = \tau$. The denominator is a normalization term based on the size of the silhouette S. Usually $\xi = 1$, in which case $\tilde{d}(S,T)$ measures the expected Euclidean distance of a pixel in S to the closest pixel in T. Note however that in this case the measure is not invariant to variations in camera depth. Everything else equal, moving the camera away from the subject makes not only the silhouettes smaller, but also the distance from pixels in S to T. Since distances in the image are linearly related to depth, but the number of silhouette pixels is quadratically related, we can optionally raise the denominator to a power of $\xi = \frac{3}{2}$ to achieve the effect of depth invariance.

4.6.2 Objective Function - Minimal Clothing Case

We define a bi-directional objective function [Sminchisescu and Telea (2002)] that uses a symmetric distance to match the estimated and observed foreground silhouettes for a given camera view k:

$$D(F^e, F^o) = \tilde{d}^{\tau}(F^e, F^o) + \tilde{d}^{\tau}(F^o, F^e)\ . \tag{4.5}$$

As illustrated in Figure 4.4, in effect this objective function equally penalizes the regions of the model silhouette that fall outside the image silhouette and the regions of the image silhouette that



(a) (b) (c) (d) (e)

Figure 4.4: **Silhouette Matching.** The process of measuring the (dis)similarity between the observed foreground silhouette (top) and the estimated silhouette of the body model (bottom) is illustrated: **(a)** input image and hypothesized mesh; **(b)** image foreground silhouette F^o and mesh silhouette F^e, with 1 for foreground and 0 for background; **(c)** silhouette distance transforms $C^\tau_{ij}(F^o)$ and $C^\tau_{ij}(F^e)$, which are 0 inside the silhouette; the opposing silhouette is overlaid transparently to illustrate the overlap between silhouettes; **(d)** contour maps for visualizing the silhouette distance transforms; **(e)** per pixel silhouette distance map $F^e_{ij} \cdot C^\tau_{ij}(F^o)$ used to compute $\tilde{d}^\tau(F^e, F^o)$ (top), and $F^o_{ij} \cdot C^\tau_{ij}(F^e)$ used to compute $\tilde{d}^\tau(F^o, F^e)$ (bottom).

are not covered by the model's projection. This is appropriate for the case where the subject wears tight-fitting clothing.

Using multiple synchronized camera views where the images are taken at the same time instant, the constraints over the K camera views are integrated to optimize a consistent set of model parameters

$$E\left(\chi, \vec{\beta^\chi}, \vec{\theta}\right) = \sum_{k=1}^{K} D\left(F^e_k\left(\chi, \vec{\beta^\chi}, \vec{\theta}\right), F^o_k\right). \tag{4.6}$$

4.7 Optimization Strategy

Recovering shape and pose by minimizing an objective function of the form $E\left(\chi, \vec{\beta^\chi}, \vec{\theta}\right)$ can be challenging given that the number of parameters to estimate is large and the objective function has

local optima. Body pose may be described by approximately 40 parameters while shape may be described by 6-20 or more. We describe several strategies that can be used to effectively find good solutions.

First, initial estimates of the parameters are obtained, providing a good starting point for the optimization (Sections 4.7.1 and 4.7.2). An optional stochastic search method (Section 4.7.3) can be used to generate more hypotheses of possible shape and pose parameters and explore the search space more widely. Finally, estimates of pose and shape can then be refined using a direct search method (Section 4.7.4).

4.7.1 Initialization of Pose

One way to make the optimization of body shape and pose practical is to initialize the search relatively close to the true solution. This initialization component can take several forms depending on the application domain.

The simplest approach involves directing the subject to stand in a particular canonical pose; for example, a **T**-pose or a relaxed standing pose. The initial pose is refined in the optimization process. This is appropriate in controlled environments with a cooperative subject. We utilize this approach in Chapter 6.

Another approach uses coarser body models which allow for an efficient, albeit less accurate, search over a larger space of poses, and then initializes the present model from the coarser method's result. For example, an existing human tracking algorithm [Bălan *et al.* (2005); Deutscher and Reid (2005)] based on a cylindrical body model can be employed for a motion sequence. This method is initialized in the first frame from marker data, and the position and joint angles of the body are automatically tracked through subsequent frames. In particular, it uses an annealed particle filtering technique for inference, uses fairly weak priors on joint angles, enforces non-inter-penetration of limbs and takes both edges and silhouettes into account. The recovered position and joint angles together with the mean body shape parameters can be used to initialize the optimization of the SCAPE parameters. This approach works well in a multi-camera setup, but still requires initialization at the first frame. Some of the experiments in this chapter make use of this approach.

In contrast, *discriminative* methods take image features and relate them directly to 3D body shape and pose. In particular, the method described by Sigal *et al.* (2008) is fully automatic and uses segmented foreground regions to produce a pose and shape estimate by exploiting a learned mapping based on a mixture of linear regressors [Sminchisescu *et al.* (2005)]. Such methods are not very precise, but are appropriate for initialization. In Chapter 5 we rely on this approach to cope with pose ambiguities in monocular frames.

4.7.2 Initialization of Shape and Gender

A good starting point for optimizing the shape of the subject is the gender-specific mean shape of the SCAPE model. In many applications, the gender of a person being estimated may be known

or the user may specify that information. In these cases, body shape using the appropriate gender-specific body model is estimated (Section 3.5.5). When gender is not known there are several options. One can fit a gender-neutral body model that is capable of representing male or female bodies. Second, one can fit using both male and female body shape models and select the one that achieves a lower error of the objective function (Chapter 6). Third, one can fit a gender-neutral model and then classify gender directly from the estimated shape coefficients. Once gender is known, a refined shape estimate using the appropriate gender-specific shape model can be produced.

4.7.3 Stochastic Optimization

Exploration of the non-convex search space of shapes and poses can be done within an *Iterated Importance Sampling* framework [Deutscher *et al.* (2002)]. Note that we do not make any rigorous claims about our probabilistic model, rather we view the formulation here as enabling an effective method for stochastic search. For a state vector $\vec{s} = \left(\vec{\beta}^{\chi}, \vec{\theta}\right)$ encoding the body parameters, we define a state space probability distribution $f(\vec{s}) = \exp\left(-E(\vec{s})\right)$ (i. e. based on Equation 4.6) and represent it non-parametrically using N particles with associated normalized weights $\{(\vec{s}_{(i)}, \vec{\pi}_{(i)})\}_{i=1}^{N}$. Such a particle set can be estimated by randomly sampling particles from a separate importance density function $g(\vec{s})$ and adjusting the weights as follows:

$$\vec{s}_{(i)} \sim g(\vec{s}) \quad , \quad \vec{\pi}_{(i)} = \frac{f(\vec{s}_{(i)})}{g(\vec{s}_{(i)})} \,. \tag{4.7}$$

The resulting Gaussian mixture density estimator for the $f(\vec{s})$ distribution:

$$\hat{f}(\vec{s}) = \sum_{i=1}^{N} \vec{\pi}_{(i)} \mathcal{N}_{\vec{s}_{(i)},\Sigma}(\vec{s}) \tag{4.8}$$

is known to be a better approximation for $f(\vec{s})$ the more similar the importance density function $g(\vec{s})$ is to $f(\vec{s})$. We therefore adopt an iterative strategy that uses the density estimator $\hat{f}^{(r-1)}$ obtained at iteration $r-1$ as the importance density function $g^{(r)}$ at iteration r:

$$\vec{s}_{(i)}^{(r)} \sim \hat{f}^{(r-1)} \,, \qquad i = 1, 2, \cdots, N \tag{4.9}$$

$$\vec{\pi}_{(i)}^{(r)} = \frac{f\left(\vec{s}_{(i)}^{(r)}\right)}{\hat{f}^{(r-1)}\left(\vec{s}_{(i)}^{(r)}\right)} \tag{4.10}$$

$$\hat{f}^{(r)}(\vec{s}) = \sum_{i=1}^{N} \vec{\pi}_{(i)}^{(r)} \mathcal{N}_{\vec{s}_{(i)}^{(r)},\Sigma^{(r)}}(\vec{s}) \,. \tag{4.11}$$

The procedure is initialized using a mixture Gaussian model with covariance $\Sigma^{(0)}$ around initial estimates of pose and shape $\{\vec{s}_{(i)}^{(0)}\}_{i=1}^{N^0}$ obtained as described in Sections 4.7.1 and 4.7.2:

$$\hat{f}^{(0)}(\vec{s}) = \sum_{i=1}^{N^0} f(\vec{s}_{(i)}^{(0)}) \mathcal{N}_{\vec{s}_{(i)}^{(0)},\Sigma^{(0)}}(\vec{s}) \,. \tag{4.12}$$

An annealing approach is used to initially search over a wide region of the state space and then gradually focus in on a specific local minimum. The sampling covariance Σ controls the breath of the search at each iteration, with a large Σ spreading sampled particles more widely. From iteration to iteration we scale Σ by a scaling factor α:

$$\Sigma^{(r)} = \alpha \Sigma^{(r-1)} \,. \tag{4.13}$$

This parameter is used to gradually reduce the diffusion covariance matrix Σ during the later iterations in order to drive the particles toward the modes of the state probability distribution. Typically α is set to 0.5.

Additionally, to avoid becoming trapped in sharply peaked local optima, predicted particles are re-weighted differently than in Equation 4.10 using a smoothed version of the state probability f:

$$\vec{\pi}_{(i)}^{(r)} = \frac{1}{Z^{(r)}} \frac{\left(f\left(\vec{s}_{(i)}^{(r)}\right)\right)^{t^{(r)}}}{\hat{f}^{(r-1)}\left(\vec{s}_{(i)}^{(r)}\right)} \qquad \text{s.t.} \qquad \sum_{i=1}^{N} \vec{\pi}_{(i)}^{(r)} = 1 \,, \tag{4.14}$$

where $t^{(r)}$ is an annealing temperature parameter optimized so that approximately half the particles get re-sampled at least once from the previous iteration [Deutscher and Reid (2005)], while $Z^{(r)}$ is a normalizing parameter ensuring the particle weights integrate to 1.

The expected, as well as the most likely parameter states, can be computed from the resulting particle set using:

$$\hat{\vec{s}} = \sum_{i=1}^{N} \vec{\pi}_{(i)}^{(r)} \vec{s}_{(i)}^{(r)} \tag{4.15}$$

$$\hat{\vec{s}}^{\text{MAP}} = \vec{s}_{(j)}^{(r)} \,, \qquad \vec{\pi}_{(j)}^{(r)} = \max_i \left(\vec{\pi}_{(i)}^{(r)}\right) \,. \tag{4.16}$$

4.7.4 Shape and Pose Refinement

Estimates of pose and shape can be refined locally using direct search methods. Using some variant of the steepest descent method is challenging as it requires an estimate of the gradient of the objective function. However, expressing the gradient analytically is difficult, and estimating it numerically using finite differencing is computationally expensive in high dimensions. Another limiting aspect of gradient-based methods is that one needs to carefully redesign the objective function to ensure it is continuous and differentiable. For example, an objective function based on the silhouette distance transforms causes discontinuities when an arm is suddenly occluded by the torso. Dealing with discontinuities at occlusion boundaries is problematic, although de la Gorce *et al.* (2008) propose an elaborate solution to address this issue.

Instead we use the Nelder-Mead simplex method [Lagarias *et al.* (1998)], which is a gradient-free direct search method minimizing an objective function in a many-dimensional space without employing numerical or analytical gradients. As such, the objective function is not required to be differentiable everywhere. It works by evaluating the objective function at the vertices of a simplex, then iteratively transforming and shrinking the simplex as better points are found until

some convergence criterion is satisfied. Compared to stochastic search methods based on particles (Section 4.7.3), this method is deterministic and therefore more consistent, moves faster toward the local optima, and is more precise at localizing it.

Faster convergence is obtained by partitioning the search space. For a given frame and gender value, it is desirable to alternate between optimizing pose and optimizing shape in an incremental fashion to help avoid local optima: after initializing with an initial pose and shape model, the process of optimizing the global position of the torso and the first few shape coefficients (e.g. the first 6) corresponding to the shape variation directions with the largest eigenvalues is commenced. The rotation of individual body parts is then estimated, starting with those closest to the torso (upper arms and upper legs) followed by lower arms and legs. Then all part rotations together with additional shape coefficients (e.g. the first 12) are jointly optimized. In the last phase, the full set of unknown variables including all pose parameters and shape coefficients are optimized.

4.8 Experiments and Evaluation

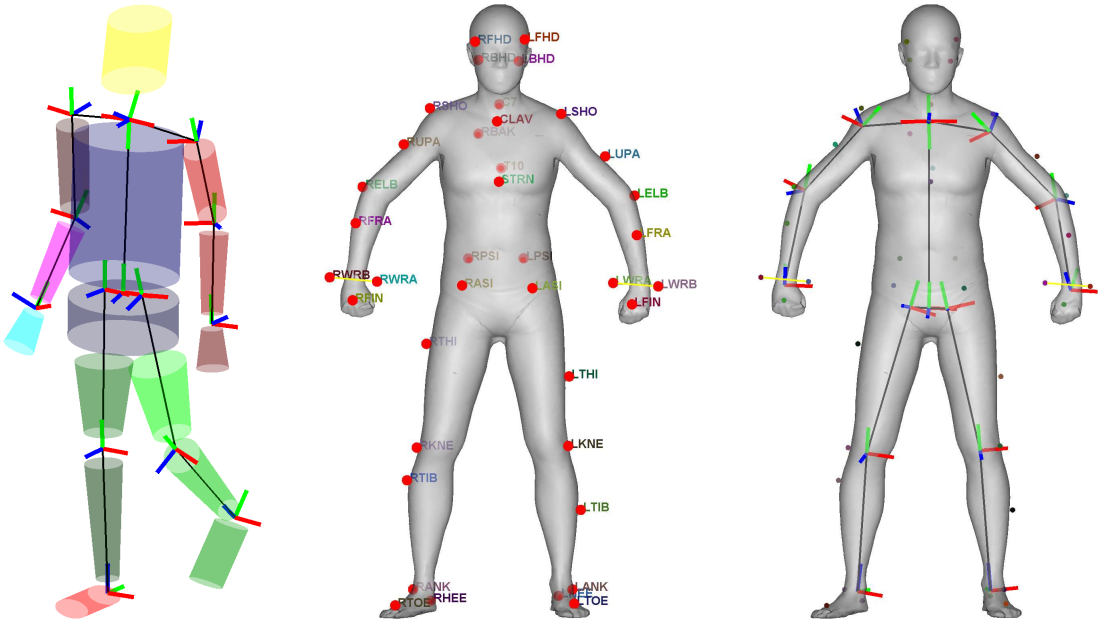
We show that when the subject wears tight fitting clothing, the silhouettes from multiple views provide tight constraints on the contour of the body that are sufficient to estimate the shape of the subject. Moreover, the learned model itself provides strong constraints on the possible recovered shapes making pose/shape estimation robust to holes in the recovered silhouettes. In addition, we find that using a more realistic body model improves pose estimates when compared with a more traditional body model based on generalized cylinders.

4.8.1 Dataset

In order to quantitatively evaluate the ability of our proposed method to estimate human shape and pose from image data, we use a video dataset with associated ground truth human motion [Bălan *et al.* (2005)] in which the subjects wear tight fitting clothing. For our experiments we use images depicting circular walking motion and ballet poses.

Ground truth motion is captured by a commercial Vicon System (Vicon Motion Systems Ltd, Lake Forest, CA) that uses reflective markers and six 1M-pixel cameras to recover the three-dimensional pose and motion of human subjects. Video data is captured simultaneously from four Pulnix TM6710 cameras (JAI Pulnix, Sunnyvale, CA). These are grayscale progressive cameras with a resolution of 644×488 pixels and a frame rate of 120Hz (though to achieve better image quality we capture video at 60Hz). Video streams are captured and stored to disk in real-time using a custom PC-based system built by Spica Technologies (Maui, HI). The Vicon system is calibrated using Vicon's proprietary software while the video cameras are calibrated using the Camera Calibration Toolbox for Matlab [Bouguet (2000)]. Offline, the coordinate frames of the two systems are aligned and temporal synchronization is achieved by tracking visible markers in both systems.

As an alternative to the SCAPE body model proposed in this thesis, we also construct a more traditional body model built around the Vicon motion capture data. It approximates the body parts



(a) Cylindrical Model (b) Virtual Markers (c) Vicon Skeleton from Markers

Figure 4.5: **Kinematic Skeletons for Two Body Models. (a)** Motion capture skeleton outfitted with generalized cylinders as body parts. Motion capture software uses marker locations on the body to infer the location and orientation of the local coordinate systems at each joint. **(b)** Virtual motion capture markers defined for a reference SCAPE mesh. **(c)** Vicon kinematic skeleton obtained for a SCAPE mesh from the virtual markers. For both the cylindrical model and SCAPE, the derived joint locations are used for evaluating the pose estimates from images against ground truth.

with generalized cylinders. The cylindrical body model represents the skeleton of the body as a 3D kinematic tree with 15 truncated cones as body parts (see Figure 4.5a). Such a model is useful for obtaining a coarse initialization and also for quantitatively evaluating the merits of the two shape models for pose estimation.

The Vicon motion capture system tracks the 3D location of 39 reflective markers physically attached to the body of the subject. The recovered marker locations are used by Vicon's proprietary software to infer the kinematic structure of the skeleton, specified through the location and orientation of the local coordinate system at each joint (see Figure 4.5a).

There are two types of parameters that describe the pose and shape of the cylindrical body model. The shape is given by the lengths and widths of the limbs which are assumed known and fixed. The pose is parameterized in terms of the relative joint angles between neighboring limbs as well as the position and orientation of the root part. This pose parameterization is the same as the one used for the SCAPE body model in Section 3.5.4.

The subjects are measured using a standard Vicon protocol to obtain their height, weight, limb width and shoulder offsets. Motion capture data is then used to estimate limb lengths for each subject. Limb lengths are computed as the median distance between pairs of corresponding joint

locations and are kept fixed for the cylindrical model during testing.

4.8.2 Evaluation Metric for Pose Estimation

We evaluate pose estimation results using an error measure based on the location of major body joints [Bălan *et al.* (2005)]. Given a body model described by a state vector $\vec{s} = \left(\vec{\beta}^{\chi}, \vec{\theta}\right)$, we use $\vec{m}^j(\vec{s})$ to denote the 3D location of the j^{th} joint. The error between the estimated state $\hat{\vec{s}}$ and the ground truth state $\vec{s}$ is expressed as the average Euclidean distance between $J = 14$ individual joint locations:

$$e(\hat{\vec{s}}, \vec{s}) = \frac{1}{J} \sum_{j=1}^{J} \left|\left| \vec{m}^j(\hat{\vec{s}}) - \vec{m}^j(\vec{s}) \right|\right| . \tag{4.17}$$

For a sequence of T frames we compute the average performance using the following:

$$e_{seq} = \frac{1}{T} \sum_{t=1}^{T} e(\hat{\vec{s}}^t, \vec{s}) . \tag{4.18}$$

For the cylindrical body model, finding the joint locations is immediate. The joints are explicitly represented by the skeleton model (Figure 4.5a). In contrast, SCAPE does not have an explicit model of the joints or a kinematic skeleton. Instead, the pose is given by the global orientation of each part and by imposing constraints to preserve mesh connectivity (see Equation 3.23). Note however that for evaluation purposes we can infer the joint locations for a given SCAPE mesh by following the Vicon protocol. We attach virtual markers to a mesh model much the same way reflective markers are physically attached to a subject's body (Figure 4.5b). Following the same approach to obtaining the ground truth joint locations from actual markers using the Vicon proprietary software, we obtain the joint locations for an image-fitted SCAPE mesh from the location of the virtual markers (see Figure 4.5c). Because SCAPE meshes are all in correspondence, the virtual markers need to be associated with points on the surface of the mesh only once for the template mesh.

4.8.3 Optimization Pipeline

We illustrate the steps of our method in Figure 4.6 on a circular walking video sequence. After foreground silhouettes are segmented from input images using background subtraction (Figure 4.6, 2^{nd} row), an existing human tracking method [Bălan *et al.* (2005)] based on the cylindrical body model is employed to provide coarse initial pose estimates (Section 4.7.1) for every frame in the sequence (Figure 4.6, 3^{rd} row). These pose estimates are refined together with the shape of a SCAPE model starting from an average shape, where the SCAPE model used is based on a gender-neutral body model obtained from a reduced shape training dataset[1].

[1] Unless explicitly stated otherwise, the experiments in Chapters 4 and 5 were performed using a preliminary implementation of the SCAPE model that used a body shape training set consisting of only 10 people, 4 of which were women, with distinctive body shape characteristics. After modeling the shape variations jointly for men and women using PCA, we kept the first 6 eigenvectors which accounted for 80% of the total shape variance in this shape training set. Gender-specific models were not learned due to insufficient data. The results are obtained by optimizing the first 6 gender-neutral shape coefficients.

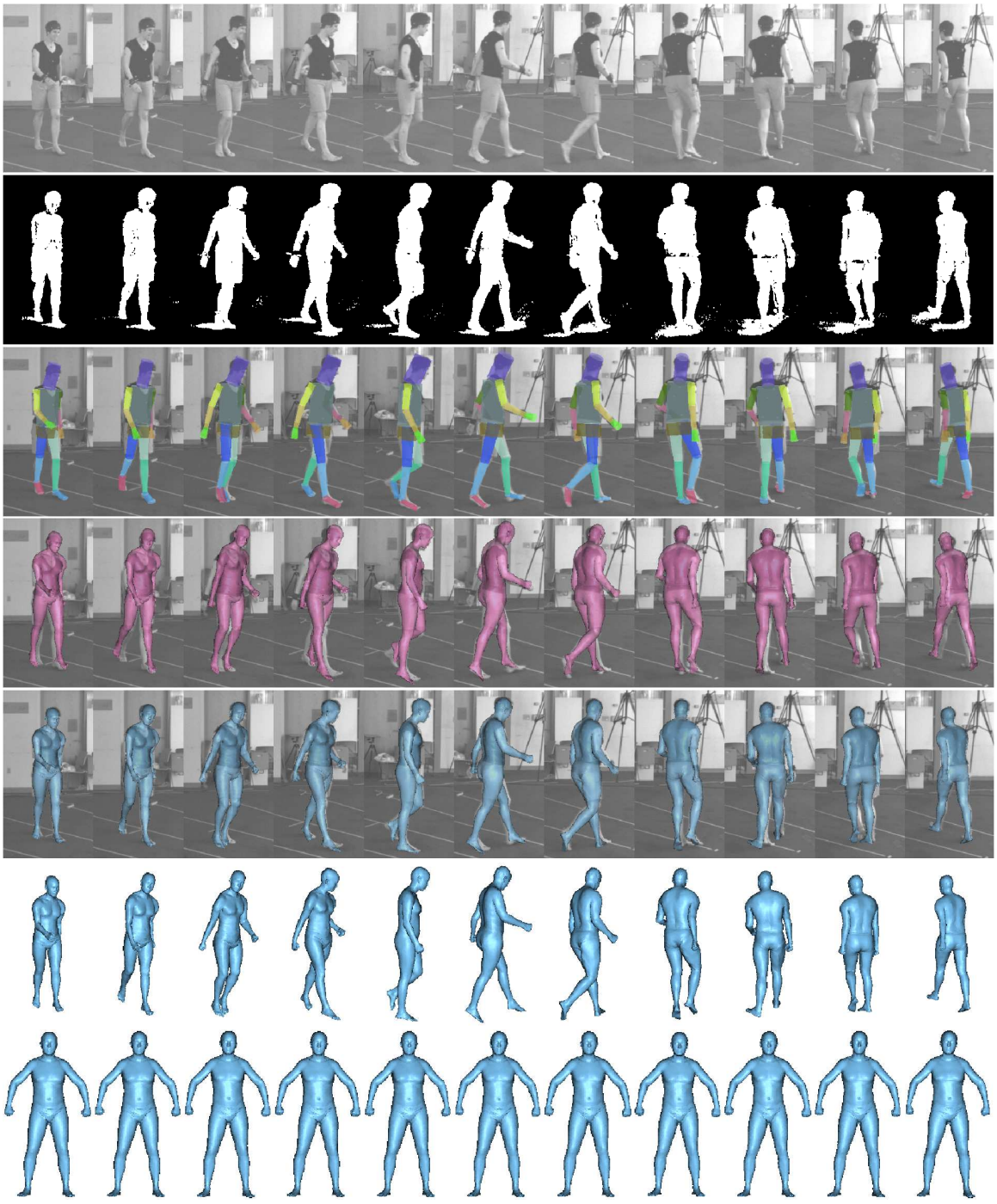


Figure 4.6: **Sequence of Poses.** 1st row: Images from one of four cameras. 2nd row: Extracted silhouettes with some shadow artifacts. 3rd row: Cylindrical body model tracking result [Bălan *et al.* (2005)]. 4th row: SCAPE initialized with the pose from the cylindrical model and an average shape. Initially, the arms and legs of the model do not align well with the images and the shape is bigger. 5th row: Image-fitted SCAPE models overlaid over input images. Alignment between the model and the images is improved, albeit the shadows-corrupted silhouettes in the later frames misguide the left leg. 6th row: Estimated 3D models. 7th row: Estimated shape morphed in a canonical pose.

4.8.4 Qualitative Results

Some body models obtained before and after fitting the SCAPE model to images are shown in Figure 4.6, rows 4-5 respectively. Additional representative results obtained with our method are illustrated in Figure 4.7. With 3 or 4 camera views we recover detailed mesh models of people in various poses and wearing sports and street clothing; none of the subjects are present in the SCAPE training set. In contrast, voxel carving techniques require many more views to reach this level of detail. The results illustrate how the SCAPE model generalizes to shapes and poses not present in the training data. We also note that the optimization can tolerate a significant amount of noise in the silhouettes due to shadows, clothing and foreground mis-segmentation.

While the proposed model does not take clothing into account, we find that the body shape can be recovered in some cases in the presence of clothing. For example, in Figure 4.7c the subject's loose pants cannot be fit by the SCAPE model. The correlations in body shape help predict the shape of some parts from those of others. As long as some parts of the body are seen un-occluded, these provide strong constraints on the body shape; this is an advantage of a learned shape model. We dedicate Chapter 6 to the problem of estimating body shape under loose clothing robustly.

4.8.5 Consistent Shape Estimation

For the results shown in Figure 4.6, even though the optimization is performed in each frame independently of the others frames, the body shape remains consistent between frames. To illustrate this, the bottom row in Figure 4.6 shows what the template mesh looks like when the shape parameters estimated in each frame are applied to it. The shapes are visually similar. By applying the shape parameters recovered from 33 frames to the template mesh placed in a canonical pose, we obtained a shape deviation per vertex of $8.8 \pm 5.3mm$, computed as the mean deviation from the average location of each surface vertex.

In general, our framework is capable of explicitly enforcing shape consistency between frames. We can either process several frames in a batch fashion where the shape parameters are shared across frames (see Chapter 6) or employ a prior in tracking that enforces small changes in shape over time.

4.8.6 Shape Estimation – Anthropometric Measurements

We use extracted anthropometric measurements to quantitatively evaluate shape accuracy. Once the shape parameters have been estimated in each frame, we can then place the mesh with the corresponding shape in an appropriate pose for extracting anthropometric measurements. From the T-pose in Figure 4.8 we can easily measure the height and arm span for each shape.

33 frames	Actual	Mean	StDev
Height (mm)	1667	1672	15
Arm Span (mm)	1499	1492	16

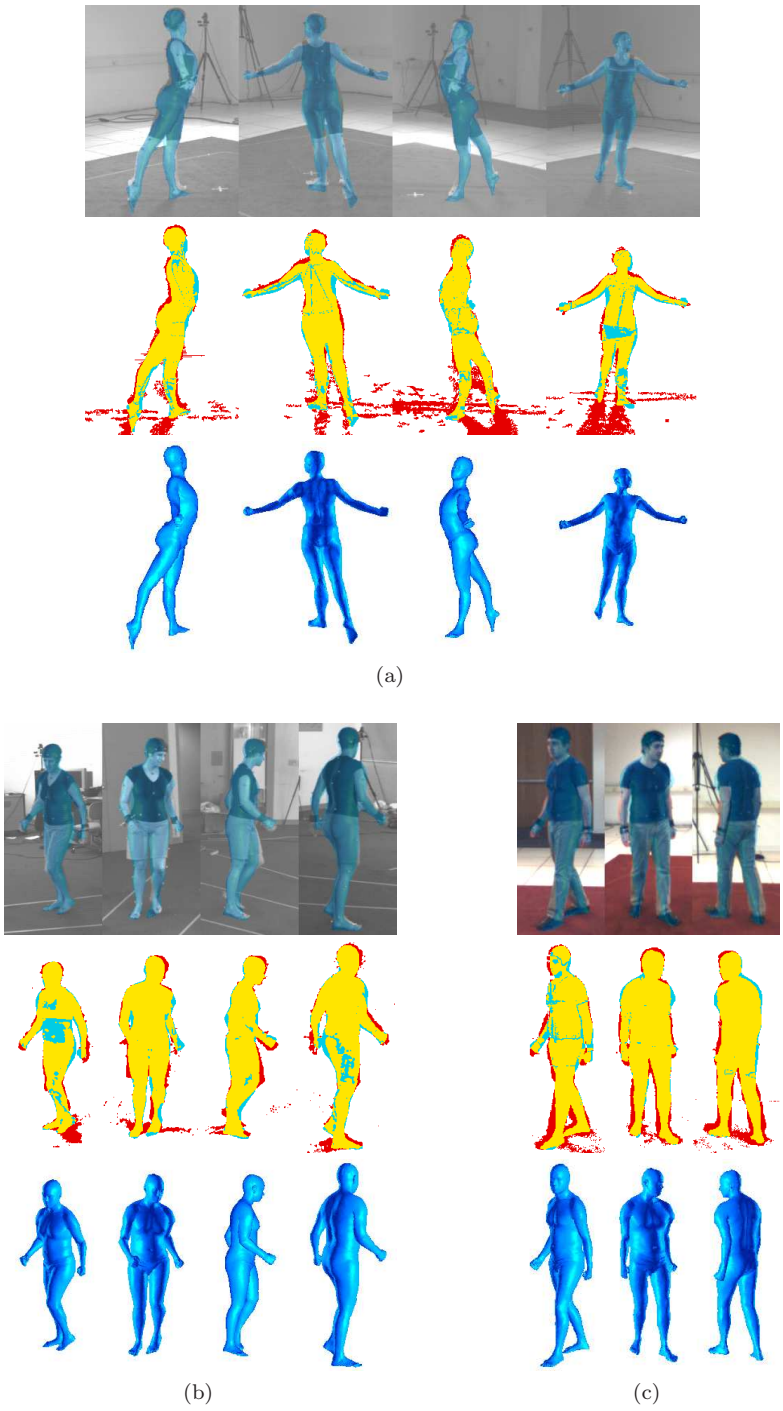


Figure 4.7: **SCAPE-from-image Results.** Reconstruction results based on the views shown for one male and two female subjects, in walking and ballet poses, wearing tight fitting as well as street clothes. *(top)* Input images overlaid with estimated body model. *(middle)* Overlap (yellow) between silhouette (red) and estimated model (blue). *(bottom)* Recovered model from each camera view. Note that these results use a smaller training set of 10 body shapes. Note also that (c) uses 3 color cameras rather than 4 grayscale cameras.

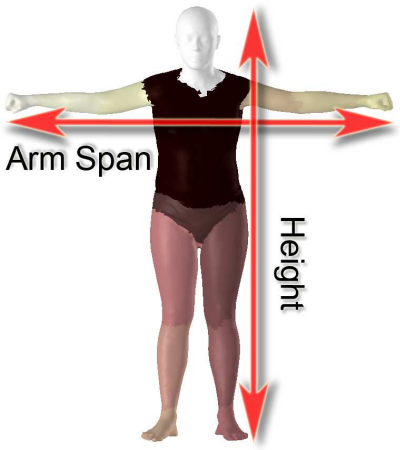


Figure 4.8: **T-pose.** Pose useful for extracting anthropometric measurements once shape was recovered from images.

The actual values for the height and arm span are within half a standard deviation from the estimated values, with a deviation of less than $7mm$. For reference, one pixel in our images corresponds to about $6mm$. This suggests the potential use of the method for surveillance and medical applications.

This analysis of shape recovery is done here only for one subject. A more extensive analysis for multiple subjects and additional measurements is provided in Chapter 6. Other measurements that could also be estimated are leg length, abdomen and chest depths, shoulder breadth etc., by measuring distances between relevant landmark positions on the template mesh, or mass and weight by computing the mesh volume.

4.8.7 Quantitative Pose Estimation Analysis

In order to test the hypothesis that a more realistic shape model improves pose estimation, we compare the performance of estimating pose using the SCAPE body model with that obtained using a traditional body model having generalized cylinders as body parts.

Figure 4.9 presents results obtained using each of the two body models for one frame using 4 different camera views. The figure illustrates how the fitted SCAPE body model is capable of explaining more of the image foreground silhouettes than the cylindrical model. This can potentially make the image matching function better behaved for the SCAPE model.

One way to quantify this is to compute how much the predicted silhouette overlaps the actual foreground (*precision*) and how much of the foreground is explained by the model (*recall*).

33 frames	Precision	Recall
Cylinder Model	91.07%	75.12%
SCAPE Model	88.13%	85.09%

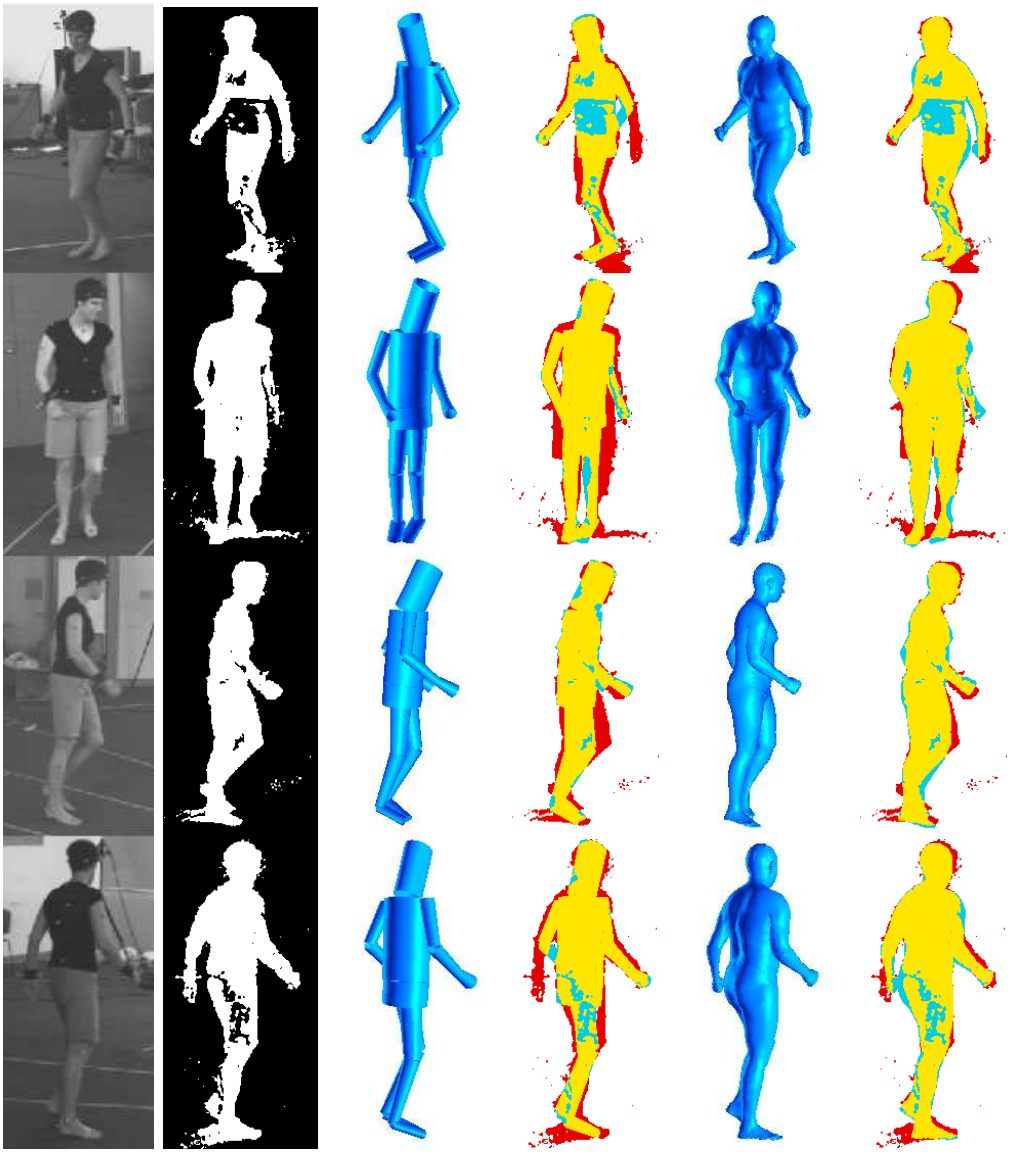


Figure 4.9: **Same Pose, Different Camera Views.** Each row is a different camera view at the same time instant. (**a**) Input images. (**b**) Image silhouettes. (**c**) 3D cylindrical model. (**d**) Overlap between image silhouettes and cylindrical model. (**e**) 3D shape model. (**f**) Overlap between image silhouettes and SCAPE model.

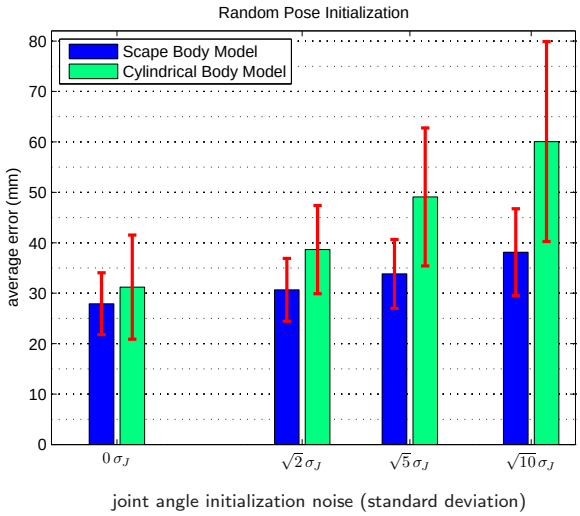


Figure 4.10: **Convergence from Random Pose.** Pose estimation error after fitting the SCAPE model and the cylindrical body model to images given different levels of initialization noise added to the true pose. The SCAPE model always converges closer to the optimal pose and diverges slower than the cylindrical body model as the distance between the initial and the true pose increases.

We find that the cylindrical model has 3% better precision because it is smaller and consequently more able to overlap the image silhouettes. On the other hand, the SCAPE model has 10% better recall because it is able to modify the shape to better explain the image silhouettes.

We also evaluate the accuracy of the pose estimates using the two body models by comparing against ground truth pose data. Recall however that our optimization pipeline relies on using the cylindrical body model to obtain initial pose estimates. To ensure fairness in comparing the performance using the two body models, we provide both methods with the same initial pose estimate derived from ground truth[2] by perturbing the true joint angles with Gaussian noise at multiple levels. In particular, we add Gaussian noise with a variance σ_j^2 equal to the maximum inter-frame difference for each joint during a walking sequence, multiplied by a factor of 0, 2, 5 and 10 for each experiment, respectively. We also assume the shape parameters are known and fixed for both models during this experiment. We evaluate the performance for every fifth frame of 300 frames containing a subject walking in a full circle and compute the average joint error using Equation 4.18.

Detailed per-frame results are shown in Figure 4.11, while aggregate results are summarized in Figures 4.10. They illustrate that the SCAPE body model converges closer to the optimal pose for

[2] Note that there is a certain systematic error for the SCAPE model when converting from ground truth pose representation to SCAPE pose representation and then back to joint locations. Ground truth pose is represented through the location and orientation of the local coordinate system at each joint, while for the SCAPE model only the orientation of each part is used to specify pose. Since SCAPE does not have an explicit model of joints, we estimate joint locations for a 3D SCAPE model from virtual markers on the mesh as described in Section 4.8.2. The resulting joints deviate from the ground truth joint locations by $21.7mm$ on average. There is no such systematic error for the cylindrical body model. This puts the SCAPE model at a slight disadvantage when comparing performance based on predicting joint locations. This becomes apparent in the top row of Figure 4.11 where the cylindrical model is effectively initialized with zero pose error while the SCAPE model produces an initial average joint error of $21.7mm$ absent any random error in initialization.

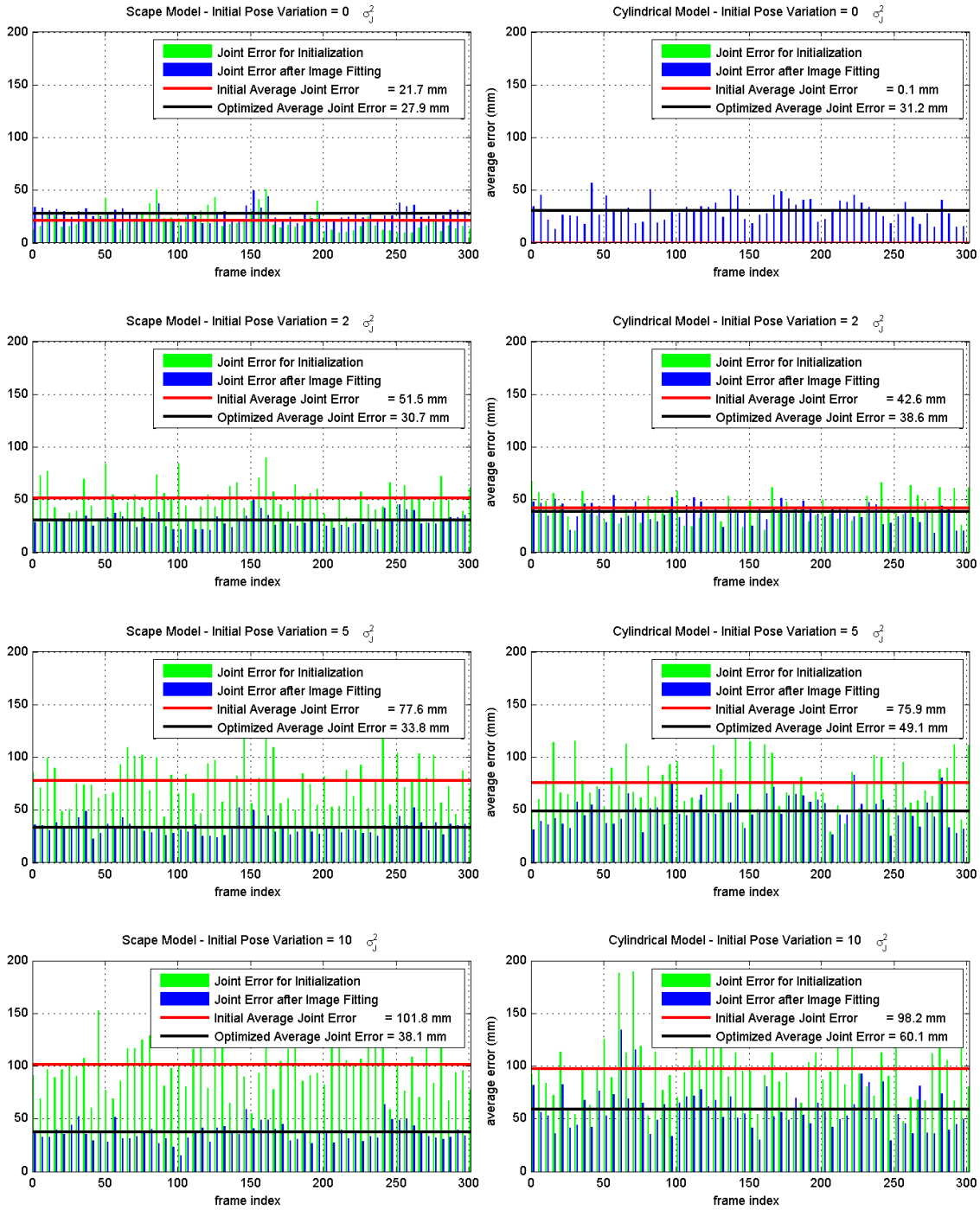


Figure 4.11: **Pose Estimation Comparison of SCAPE vs. Cylindrical Model.** Average joint error shown for individual frames of a walking sequence, before and after the optimization. Each row contains a different level of noise added to initial poses. *Left column:* Results using the SCAPE body model. *Right column:* Results using the cylindrical body model. The further away from the true pose the cylindrical model is initialized, the easier it is for it to get stuck on local optima.

all levels of noise added to the initial pose. The further away from the true pose the cylindrical model is initialized, the sooner it gets stuck in local optima. We formally confirmed this using Welch's t-test and found the improvement in performance for the SCAPE body model relative to the cylindrical model to be statistically significant at 95% confidence level for the zero noise case, and at 99.99% confidence level whenever noise was added to the initialization. The results confirm that the cylindrical body model is a poor generative model of human form and as such cannot explain the image evidence very well, leaving room for extra ambiguities. Because the SCAPE body model can better conform to the shape of the observed person, the ambiguities are reduced and the convergence to the optimal pose is easier, achieving a lower average joint error. Additionally, we observe in Figure 4.10 that the difference in performance between the cylindrical model and SCAPE increases at a higher rate as more noise is added to the initialization, with the cylindrical model performing much worse and diverging faster at higher levels of noise.

Finally, we note that the fitting procedure diverges slightly from the true pose for both models even when initialized at the ground truth pose with zero noise (Figure 4.11, top row). This is due to the presence of errors in image segmentation as well as some uncertainty in collecting the ground truth data and estimating the joint locations.

4.9 Discussion

We have presented a method for estimating 3D human pose and shape from images. The approach leverages a learned model of pose and shape deformation previously used for graphics applications. The richness of the model provides a much closer match to image data than more common kinematic tree body models based on simple geometric primitives. The learned representation is significantly more detailed than previous non-rigid body models and captures both the global covariation in body shape and deformations due to pose. We have shown how a standard body tracker can be used to initialize a search over shape and pose parameters of this SCAPE model. Using a state of the art model from the graphics community we are better able to explain image observations and make the most of generative vision approaches. Additionally, the model can be used to extract relevant biometric information about the subject.

Chapter 5

Shape from Shadows

5.1 Introduction

Strong illumination is often seen as a problem for pose estimation and tracking; this is particularly true for human pose estimation. In contrast, we show that, rather than hinder human pose and shape estimation, strong illumination can actually make it more robust. With a known light source, shadows provide additional constraints for pose estimation. Conversely, if one has accurate pose estimates, we can estimate the light source location. Putting both of these observations together results in a complete framework for incorporating strong illumination in human body estimation. These ideas, however, are applicable to object detection and tracking in general.

Consider the situation in which the scene is illuminated by a single, known, point light source and is viewed through one, or more, calibrated cameras. Here we focus on indoor scenes where the light source distance is finite. The approach, however, easily generalizes to distant light sources, the most common being the sun in outdoor scenes. Our first observation is that a point light source and the ground plane form what we call a *shadow camera*. The point light acts like the focal point of a pinhole camera with the ground plane acting like the image plane. The image formed on the ground is the shadow cast by the body (Figure 5.1). This can be generalized to multiple light sources (which effectively produce a "camera" with multiple focal points). The cast shadow image acts like a foreground silhouette mask in the image plane of a regular camera. Note, moreover, the "image plane" of the shadow camera need not be planar but can be any calibrated surface (or surfaces) in the scene. This shadow image provides additional constraints on body pose which make it possible to estimate 3D pose from monocular camera views.

Making use of shadows requires the accurate segmentation of shadow regions in images. To that end we propose a novel approach that uses background subtraction data and checks whether putative shadow pixels are consistent with being on the calibrated ground plane.

For a complete framework, we must also estimate the lighting in the scene automatically. For this, we develop an approach that exploits 3D body pose and shape represented using the SCAPE

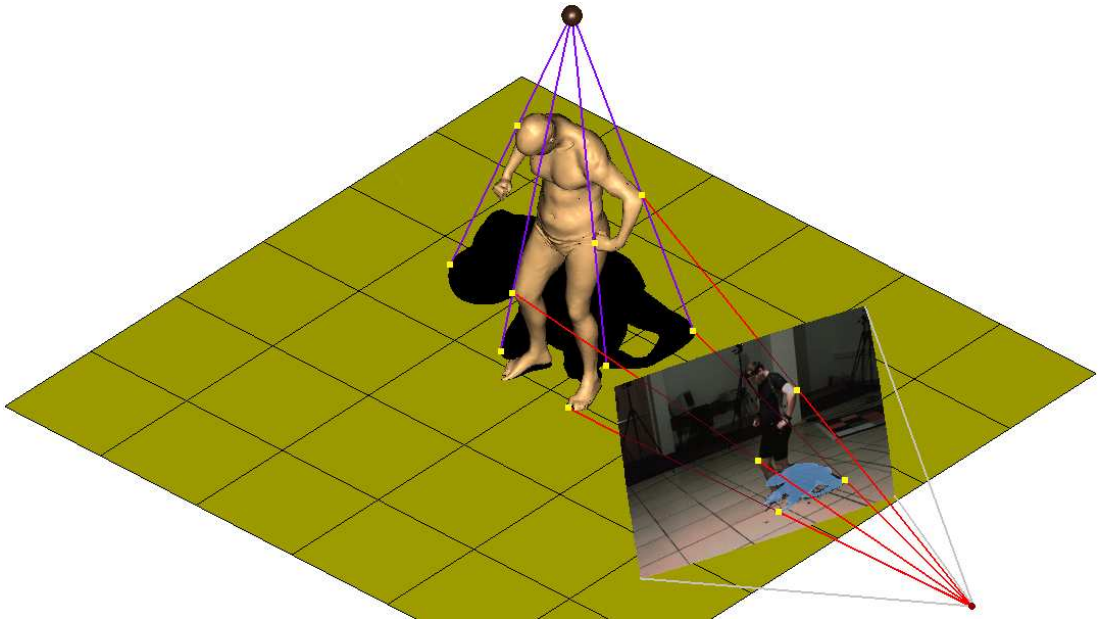


Figure 5.1: **The Shadow Camera.** The combination of one or more point light sources with a known ground surface forms a virtual "shadow camera". The light source is the focal point of a pinhole camera, while the ground surface acts as the image plane. The silhouette of the shadow provides an additional view of the body.

model (Chapter 3), the parameters of which are estimated directly from image foreground silhouettes without knowledge of scene illumination (Chapter 4). This approach recovers a point light position (or direction) from cast shadows. Using the known body pose in multiple frames and the detected shadow regions, we optimize for the light position that best explains the cast shadows. Combining many poses gives strong constraints on the location of the light. Hence by tracking an object with fixed lighting we can actually infer the lighting; in this way the human body becomes a *light probe* [Debevec (1998)].

We present results on several sequences with three light configurations and two subjects. A quantitative evaluation of pose estimation under different numbers of cameras and different numbers of point light sources is also provided. We assume the video cameras have been fully calibrated with respect to a global coordinate system, as well as the ground plane on which the cast shadows are visible. Finally, we assume we know the number of the light sources, but not their locations.

5.2 Related Work

There is a long history of recovering lighting and using it to infer 3D structure. This work includes shape-from-shading, photometric stereo, shadow carving, inverse lighting, and reflectance modeling. A thorough survey is beyond the scope of this thesis and the reader is referred to [Luong *et al.* (2002)] for an overview.

Our work is quite different from to the majority of work in shape, shading and lighting. Most approaches assume a fixed object which is viewed under different lighting conditions. The most common approaches attempt to estimate object shape from multiple images of a static object illuminated from different light locations (for example [Epstein *et al.* (1996); Yuille *et al.* (1999)]); in many cases these light locations are known. We turn this standard problem around and use multiple known poses (i.e. estimated from data) of the object to estimate the unknown lighting.

The most closely related work is that of Luong *et al.* (2002) which estimates light sources and albedos using multiple views of an object. They assume a rigid object but move the camera to track it. This is similar to our case where the camera is static but the object moves. We go beyond their work to deal with an articulated non-rigid object which casts shadows on the ground plane.

There has been little work on articulated pose estimation from cast shadows, with the focus on simple objects. Segen and Kumar (1999) describe a system to recognize basic hand gestures by tracking the 3D position and orientation of two fingers using the hand shadow captured with a single camera. More relevant is the work of Bruckstein *et al.* (2001) in which they geometrically recover the pose of an articulated human stick figure and the light position from shadows. The approach requires the skeletal joints, and their corresponding locations on the shadow, to be manually marked in the image.

We apply a different strategy and define an objective function over the parametric pose and shape of the subject and the point light source position such that the projection of the shape onto the image silhouette and the shadow best overlap the observed body regions. We believe this to be the first automatic procedure to estimate articulated human pose and shape by taking advantage of cast shadows.

5.3 Pose & Shape from Silhouettes & Shadows

Much of the work on human pose estimation and tracking employs generative models of human shape that are crude approximations of the body. In Chapter 4 we used a detailed graphics body model (SCAPE), learned from range scans of real people, to address the problem of markerless human pose and shape estimation in a multi-camera setting. The generative model predicts silhouettes in each camera view given the pose/shape parameters of the body and matches them to foreground silhouettes extracted from images using a fairly standard Chamfer distance measure. Here we extend this framework to take advantage of shadows cast from point light sources. These shadows provide additional constraints on pose and shape which are sufficient to disambiguate and effectively enable monocular 3D pose estimation.

The new framework consists of the following steps:

1. Segment the images into background, foreground and shadow regions (Section 5.5);
2. Estimate pose and shape parameters from foreground silhouette data alone and generate the surface meshes in each frame (Chapter 4);
3. Estimate light position from shadows (Section 5.7); and

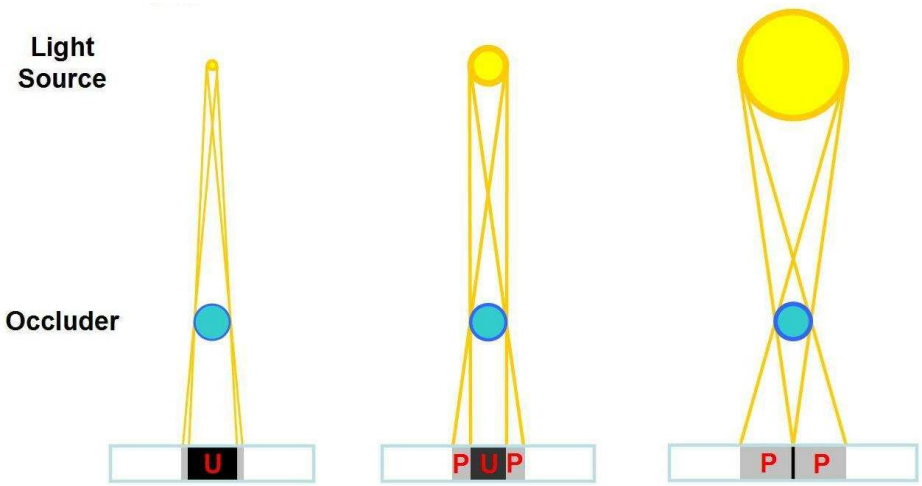


Figure 5.2: **Shadow Formation.** A shadow appears when an occluding object prevents a surface from receiving full contribution from the light source. (a) For a single point light source, the shadow is very dark and has sharp boundaries. (b) For a non-point light source, the shadow is divided into *umbra* and *penumbra*. Umbra (U) is the surface region where no part of the light source can be seen, while penumbra (P) is the surface region where only some portion of the light source is visible. (c) The wider the light source, the greater the penumbra, with the umbra eventually disappearing. Because the light source slowly disappears behind the occluder, the shadow boundaries become smooth and the shadow blurred.

4. Re-estimate pose and shape from foreground regions, shadow regions and the estimated light position (Section 5.6).

5.4 The Shadow Camera Model

In order to detect and exploit shadows in images, we first need to understand how shadows are formed. Objects can be seen only when they are illuminated. In the absence of light, everything would appear pure black. Therefore the presence of light in a scene is essential for observing it. The more light reaches the surface of an object, the brighter it appears. Conversely, a shadow appears when a surface area no longer receives direct illumination from a light source due to the obstruction caused by the presence of an opaque object in the scene.

The appearance of shadows depends greatly on the type of lighting in the scene, as illustrated in Figure 5.2. When the lighting is diffused, like in the case of an extended area light source, the shadows produced are very weak and with smooth boundaries, making shadow detection and segmentation ambiguous. Inferring light properties from the scene is also more difficult in this case. A typical example of a wide light source is the sunlight going through a thick cloud: as light traverses the cloud, water particles scatter the light rays in different directions, creating the equivalent of an extended area light source. Another example is given by the rectangular fluorescent lights found in tiled ceilings where the light passes through a semi-transparent diffuser screen. The wider the light

source, the more blurred and less useful the shadow.

Shadows become more structured and informative when strong light concentrates in a small volume relative to the observed scene. Direct, undiffused light casts strong shadows with crisp boundaries, increasing the color contrast and saturation and making shadow detection and extraction possible. As illustrated in Figure 5.2a, the transition from shadow to no shadow regions is done quickly, making the shadow boundaries more precisely localized. This is exactly the case we would like to exploit.

5.4.1 The Single Point Light Case

Consider the case where a single point light source exists in the scene and a human subject occludes the ground plane from the light (Figure 5.1). The shadow is just the silhouette of the subject seen from the location of the light onto the ground plane. The geometry of the shadow follows the same principles as for a pinhole camera. The light position is the focal point of the camera, while the ground plane is the image plane of the camera. Effectively the ground plane and the light form a virtual "shadow camera".

For a single light source at a finite distance, the shadow formation amounts to a perspective projection of the 3D body shape on the ground plane through the light position. Let L be the 3D position of a single point light source, and $\Pi_g = (O_g, \vec{n}_g)$ the ground plane specified by the origin O_g and a normal vector $\vec{n}_g$ to the plane. The shadow of point P onto the ground plane Π_g from the light L is given by the point $S = \text{Proj}_{\Pi_g, L}(P)$. This perspective projection is simply the intersection between the light ray through P and the ground plane. Deriving the solution for the point S requires satisfying the following constraints:

$$\begin{cases} S = L + (P - L)\lambda\,,\ \lambda \in \mathbb{R} & \text{(light ray constraint)} \\ \vec{n}_g^{\mathsf{T}}(S - O_g) = 0 & \text{(ground plane constraint)} \end{cases} \tag{5.1}$$

Substituting the light ray constraint into the ground plane constraint, we solve for λ:

$$\begin{aligned} \vec{n}_g^{\mathsf{T}}\left(L + (P - L)\lambda - O_g\right) &= 0 \\ \vec{n}_g^{\mathsf{T}}(P - L)\lambda &= \vec{n}_g^{\mathsf{T}}(O_g - L) \\ \lambda &= \frac{\vec{n}_g^{\mathsf{T}}(O_g - L)}{\vec{n}_g^{\mathsf{T}}(P - L)}\,. \end{aligned}$$

Finally, the shadow point is obtained by substituting λ back into the light ray constraint:

$$\text{Proj}_{\Pi_g, L}(P) = L + (P - L)\frac{\vec{n}_g^{\mathsf{T}}(O_g - L)}{\vec{n}_g^{\mathsf{T}}(P - L)}\,. \tag{5.2}$$

If the ray from the light to P is parallel to the ground plane, then $\vec{n}_g^{\mathsf{T}}(P - L) = 0$ and there is no shadow point.

For a body model represented as a 3D triangle mesh whose geometry is defined by a set of vertex locations and whose connectivity is stored using indexed arrays of vertex triplets, obtaining a

mesh model of the shadow amounts to simply taking the perspective projection through the shadow camera of the vertices of the body model while preserving the mesh connectivity. The shadow mesh, while planar when projected onto the ground plane, is still a 3-dimensional shape model, sharing the same topology as the body model. Hence, the process for computing the image shadow silhouette for a given camera view is going to be identical to the one used for computing the image silhouette of the actual body (see Equation 5.3).

5.4.2 Generalization of the Shadow Camera Model

Several extensions to the basic "shadow camera" model are possible.

Distant Point Light Source. When the light lies an infinite distance away from the scene, the shadow camera has infinite focal length and the perspective projection of the shadow becomes parallel projection. The sun in outdoor scenes is a common example of directional lighting. The light rays illuminating the scene are considered parallel to each other. Specifying directional lighting requires only two parameters (e. g., the elevation angle from a ground plane and the azimuth angle in the plane).

Multiple Point Light Sources. In the case of several point light sources in the scene, several possibly overlapping shadows are cast that depend on the light color and intensity of each light source. The shadows become lighter in non-overlapping regions; differentiating between the shadows is easier when the lights are covered by different color filters. When the shadow cast from multiple light sources is taken to be the union of the individual shadows, the shadow camera is the equivalent of a pinhole camera model with multiple pinholes (focal points).

Non-planar Projection Surfaces. Having a planar projection surface is optimal for computational reasons. However the image plane of the "shadow camera" need not be planar. The shadows can be cast onto arbitrary surfaces of known geometry. For example the walls and floors of an empty room form a multi-planar surface which can be established by affixing a calibration checkerboard pattern to each plane and estimating a coordinate system rigid transformation with respect to one of the cameras (Section 4.4). Handling more complex scenes comes at a greater computational cost. It requires finding the intersection between light rays and one or more arbitrary surfaces whose surface geometry also needs to be calibrated. A complex scene can be decomposed into objects whose shapes are specified mathematically using simple geometric primitives (spheres, cones, planar patches, polygonal meshes) and for which computing the point of intersection is simpler. Similar to the ray-tracing algorithm in computer graphics, each light ray going through an occluding point P can be tested for intersection with some subset of all the objects in the scene, identifying the nearest object and computing the point of intersection with it.

Composition of Two Projections. The shadow projection in Equation 5.2 can be composed with the camera projection in Equation 4.3 to directly obtain the shadow silhouette in one of the

images

$$p^i = \text{Proj}_{C^i}\left(\text{Proj}_{\Pi_g,L}(P)\right), \tag{5.3}$$

where p^i is the pixel location in camera view C^i of the shadow cast by the occluding point P.

Shadow Camera vs. Pinhole Camera. It is worth pointing out one important difference between a shadow camera and a pinhole camera. In the shadow camera case, the object (person) generally occludes part of the shadow (because it sits "inside" the camera). The information in the partial shadow however is usually complementary to the foreground silhouette and therefore well suited to be exploited jointly. Nonetheless, the amount of information extracted from the shadow greatly depends on the relative placement of the light and the camera with respect to the body. In the special case where the camera and light are in the same location, the body completely covers the shadow and no new information is gained.

5.5 Foreground and Shadow Segmentation

The detection of shadows in images is an important activity in computer vision. This is because shadows have long been viewed as an obstacle in reliably estimating the silhouettes of the foreground objects. Image foreground silhouettes are typically extracted using standard background subtraction methods that segment the image into two classes: foreground and background. Because both foreground and shadows differ significantly from the background, distinguishing between them is challenging. In this context, most work in shadow detection has focused on *removing* the shadows to improve foreground segmentation [Prati *et al.* (2003)]. Segmenting the shadows with the purpose of actually using them is also important because shadows can provide additional cues about the 3D structure of the scene.

The vast majority of existing shadow detection/removal techniques are concerned with images or videos taken with a single camera. Here we describe a new method to segment shadows from foreground that becomes very robust when additional views of the scene are available. Starting from an initial segmentation in each camera view, our method then employs homography constraints to jointly detect and correct the segmentation in all views. This method applies to both shadow detection and removal problems.

Multiple point light sources in the scene generate multiple shadows. Our segmentation approach does not attempt to separate individual shadows. Instead we detect the union of the background regions in the image that are in shadow due to the presence of the subject in the scene. Additionally, our approach does not require any knowledge about the light sources in the scene, relying instead on differences from images of the empty background.

5.5.1 Single-view Segmentation

The initial step involves classifying foreground, shadow, and background classes using a simple classifier. This is done independently in each view. Images are first transformed into the HSV

color space (Hue, Saturation, Value) with values between 0 and 1. The HSV color space can be used more effectively in discriminating foreground from shadows than the RGB (Red, Green, Blue). Alternative color spaces that also approximate the human visual system model include $YCbCr$ and $L*a*b$.

We assume a stationary background and model each pixel independently with a mean $\mu^{HSV} = [\mu^H, \mu^S, \mu^V]^\mathsf{T}$ and standard deviation $\sigma^{HSV} = [\sigma^H, \sigma^S, \sigma^V]^\mathsf{T}$ for each of the three HSV color channels, estimated from a set of images (B) of the empty scene. Note that circular statistics (the von Mises distribution [Bishop (2006)]) need to be used for the Hue channel because the Hue values wrap around:

$$\begin{aligned} \mu^H &= \text{atan2}\Big(\text{E}\big[sin(2\pi B^H)\big], \text{E}\big[cos(2\pi B^H)\big]\Big) \\ \sigma^H &= \sqrt{\text{E}\big[\min\{|B^H - \mu^H|, 1 - |B^H - \mu^H|\}^2\big]}, \end{aligned} \tag{5.4}$$

where E[·] denotes the expected value.

The basic idea is to detect where a new image I differs from an image of the empty scene and classify those regions as non-background, and then distinguish foreground from shadows. Good results are obtained by segmenting the foreground from non-background pixels using only the saturation channel. We made this choice particularly for its decreased foreground false positive rate relative to shadows. We employ a basic deterministic decision process:

$$Class = \begin{cases} Background: & \left\|[N^H, N^S, N^V]^\mathsf{T}\right\| < t^{HSV}, & else \\ Foreground: & N^S \geq t^S, & else \\ Shadow: & otherwise \end{cases} \tag{5.5}$$

where N^* denotes the normalized absolute difference between expected and observed value for each channel:

$$\begin{aligned} N^H &= \frac{\min\left\{\left|I^H - \mu^H\right|, 1 - \left|I^H - \mu^H\right|\right\}}{\sigma^H} \\ N^S &= \frac{\left|I^S - \mu^S\right|}{\sigma^S} \\ N^V &= \frac{\left|I^V - \mu^V\right|}{\sigma^V}. \end{aligned} \tag{5.6}$$

The thresholds t^S and t^{HSV} are determined empirically from data.

This initial segmentation is followed by several morphological operations, including median filtering, image dilation and erosion, removal of small disconnected components and hole filling. Alternative options include smoothing using regularization, anisotropic diffusion or other Bayesian approaches such as Markov Random Fields. Rows 1 and 2 in Figure 5.3 show the segmentation result before and after this procedure for four different views. In the next section we describe how to improve these segmentation results from multiple views jointly.

Of course, there are many other methods known in the literature that can be used to obtain this initial segmentation. The key novelty here lies in the combination of shadows across multiple camera views. The insight is that the foreground changes in each view while the shadow does not (relative to a planar homography).

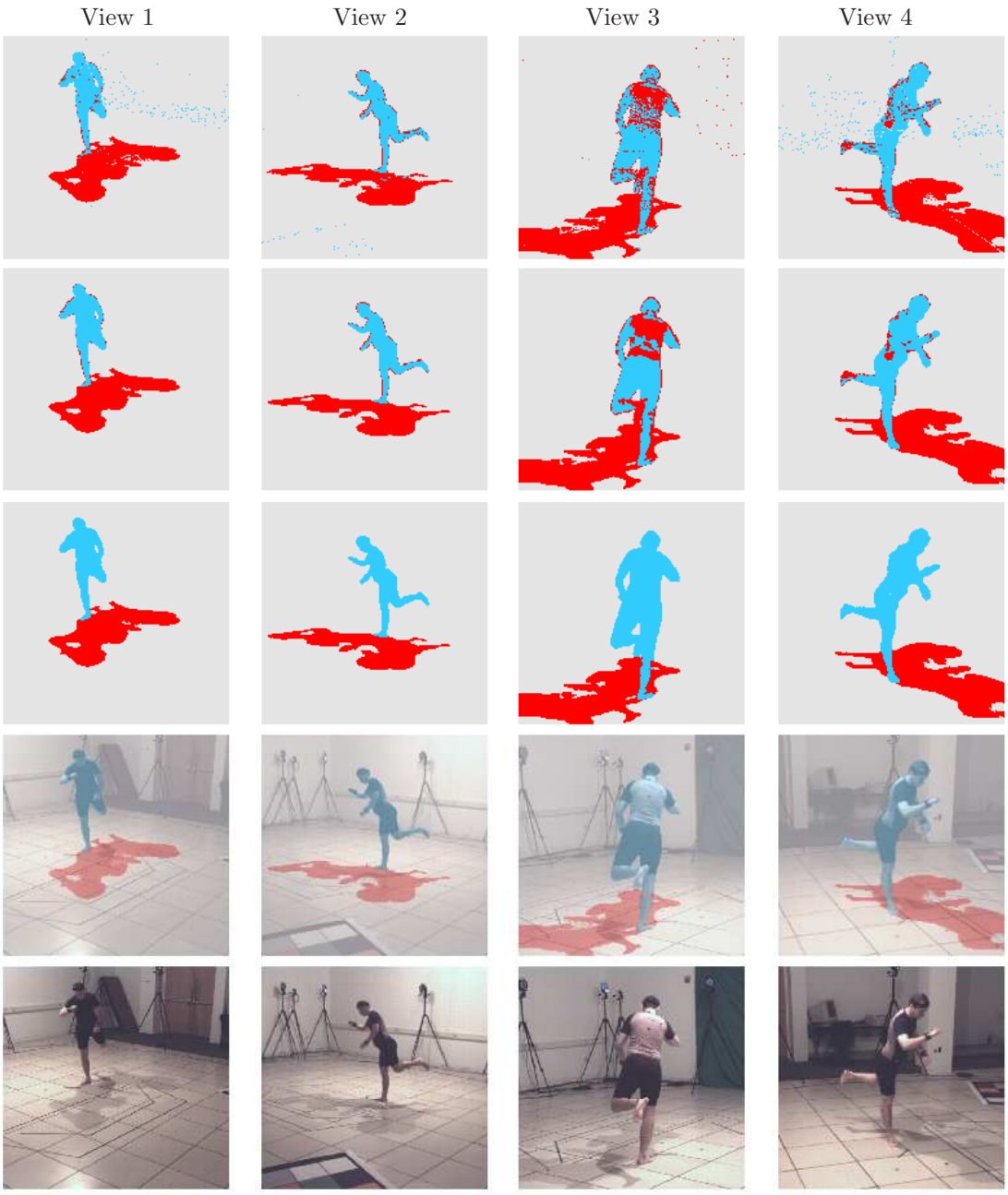


Figure 5.3: **Foreground and Shadow Segmentation.** The process of segmenting the images into foreground and shadow regions is demonstrated for the case of two lights present in the scene. Each column represents a different camera view. **Row 1:** Per pixel classification. Red denotes shadow and blue foreground. **Row 2:** Morphological operations. **Row 3:** Multi-view integration. Note the robustness introduced by this step. **Row 4:** Segmentation overlaid on original images. **Row 5:** Original images (with two light sources).

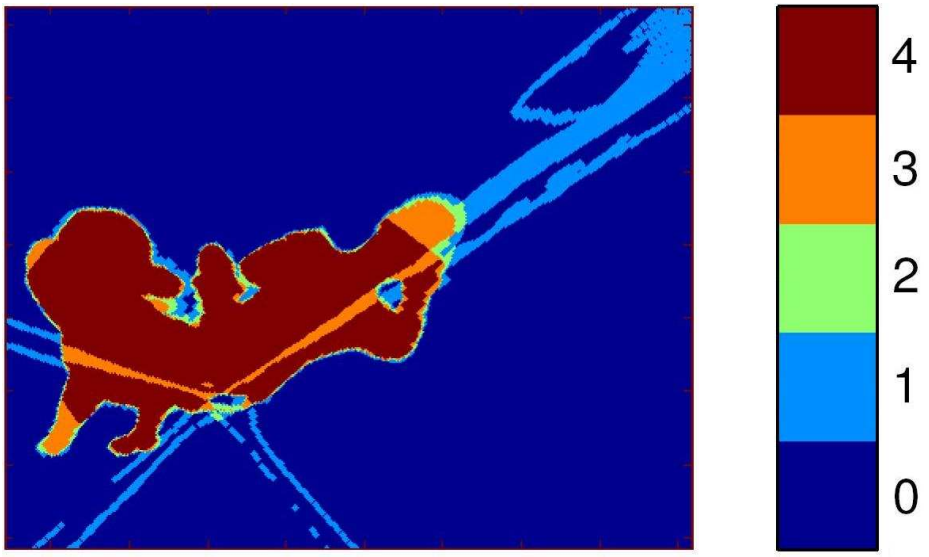


Figure 5.4: **Shadow Integration.** Segmented shadow regions from Figure 5.3 aligned in a single coordinate frame. Each registered shadow can be thought of as voting for the presence of a shadow. Regions that contain only one vote (light blue) can be classified as outliers and potentially relabeled as foreground pixels.

5.5.2 Multi-view Segmentation

We employ a novel shadow segmentation method that uses multiple synchronized and calibrated camera views of the scene and a calibrated ground plane on which the shadow is cast. By combining information about the shadow from different views, inconsistent shadow segmentations can be detected and corrected. This is first made possible by the fact that shadows can be easily registered in a common coordinate system. All we need to know is how to warp the shadows between views. Each camera view yields a 3D planar reconstruction of the shadow since each shadow pixel corresponds to a 3D ray which can be intersected with the ground plane with known coordinates. The reconstructed 3D shadow from one view can then be re-projected into any other view. This means that shadows can be aligned and compared in a common coordinate system. We can think of the segmented shadow in each view as voting for the true 3D shadow. For instance, Figure 5.4 shows the shadows from all 4 views warped to a virtual view directly above the ground plane and accumulated.

Ideally, the reconstructed 3D image of the shadow is the same in all views. In practice, they differ for two reasons: 1) the body occludes parts of the shadow in some camera views, and 2) the initial shadow detection in images is unreliable.

Usually the shadow regions occluded by the foreground in one view are still visible in the majority of the other views. On the other hand, foreground or background regions mis-labeled as shadow are not expected to match the shadows from other views. For example, in Figure 5.3, View 3, most of the torso is detected as shadow, yet it is not consistent with the shadow reconstructions in the other

views.

We adopt a conservative approach and attempt to relabel inconsistent shadow regions only when it leads to a spatially consistent foreground segmentation. More precisely, we re-label shadow pixels as foreground when they are not explained by the other shadow views *and* are adjacent to foreground pixels in the current view. Since this step alters the adjacency condition, the procedure is repeated until convergence (between 3 and 5 iterations in our experiments). This alone leads to robust and clean segmentations (see Figure 5.3).

Please note that this specific voting method with iterative removal is only one way to implement the multi-view shadow segmentation idea. We could formulate the problem probabilistically with each image giving a likelihood of foreground versus shadow. This can be combined with a spatial prior and optimized using Bayesian or maximum-likelihood inference.

Our multi-view segmentation procedure is effective at correcting foreground regions incorrectly detected as shadow. Typically there is a precision-recall trade-off in the initial classifier with respect to detecting foreground regions versus shadows. Therefore the initial classifier should be designed to maximize the precision of detecting foreground regions and not the recall.

5.6 Problem Formulation

Our goal is to estimate the shape and pose of the body from one or more images of the subject under different illuminations. This can be done within a framework of *Analysis through Synthesis* as illustrated in Figures 4.1 and 5.1. For a given set of body model parameters as well as the position of the point light source(s) and the equation of the ground plane, the entire scene can be reconstructed in 3D and subsequently rendered in 2D to simulate the camera imaging process. The model parameters can be fitted by defining a metric that compares features in this simulated image with features in the actual image captured with the camera. Following the approach introduced in Chapter 4, we rely only on image silhouettes which have been widely used in human pose estimation and tracking. The generative framework presented here, however, can be readily extended to exploit other features such as edges, shading or optical flow.

Given a a predicted body shape and pose described by the state vector $\vec{s} = (\chi, \vec{\beta}^{\chi}, \vec{\theta})^{\mathsf{T}}$, a 3D surface mesh of the body is constructed. For each active light source in the scene with an estimated position, a 3D mesh of the cast shadow on the estimated ground plane is also constructed.

We use K calibrated cameras to capture synchronized images. Both the body mesh and each shadow mesh are rendered into each camera view k using the camera calibration parameters C^k (Section 4.4), light parameters $\vec{\ell}$ and ground plane parameters $\vec{g}$. By rendering first the shadows and then the body model into the image, the occluded regions of the shadows can easily be determined. From this we obtain the estimated silhouettes of the subject $F_k^e(\vec{s})$ and the visible joint cast shadows $S_k^e(\vec{s}, \vec{\ell}, \vec{g})$. These can then be compared to the observed foreground and shadow silhouettes extracted from the input images F_k^o and S_k^o respectively.

To estimate the model parameters, we formulate an image error function in terms of a silhouette

dissimilarity measure $D(\cdot^e, \cdot^o)$ between estimated and observed silhouettes. Many such measures can be defined, and the exact choice is not critical. Following the approach in Chapter 4, the $D(\cdot^e, \cdot^o)$ measure is implemented as a bidirectional Chamfer distance between the estimated and observed silhouettes and vice-versa (see Equation 4.5).

The mismatch between the estimated silhouette of the subject observed in camera k is given by $D(F_k^e(\vec{s}), F_k^o)$, while the mismatch for the shadow is given by $D(S_k^e(\vec{s}, \vec{\ell}, \vec{g}), S_k^o)$. In Equation 4.6 we relied on foreground silhouettes alone to estimate pose and shape; here we add a "shadow camera" term $\frac{1}{K}\sum_{k=1}^{K} D\Big(S_k^e(\vec{s}, \vec{\ell}, \vec{g}), S_k^o\Big)$ to measure the shadow difference. Note that while there are K actual cameras, there is only one "shadow camera". The shadow contributions from different views need to be averaged together as they do not really provide independent constraints (see Section 5.5.2). To optimize pose and shape from silhouettes and shadows we minimize the image error function

$$E(\vec{s}) = \sum_{k=1}^{K} D(F_k^e(\vec{s}), F_k^o) + \frac{1}{K}\sum_{k=1}^{K} D\Big(S_k^e(\vec{s}, \vec{\ell}, \vec{g}), S_k^o\Big) \ . \tag{5.7}$$

5.7 Estimating the Light Position

Estimating the light position can be done within the same *Analysis through Synthesis* framework. The idea is to use the human body as a light probe. As a subject moves through the scene, the shadows on the ground plane provide independent constraints on the position of the light assuming a stationary light source. The shape and pose of a human subject, $\vec{s}^t$, at several time instants $t = 1 \ldots T$ can be initially estimated without relying on any lighting information in the scene, based solely on foreground silhouettes (Section 4.6.2), by optimizing the objective function

$$E(\vec{s}^t) = \sum_{k=1}^{K} D\big(F_k^e(\vec{s}^t), F_k^o\big) \ . \tag{5.8}$$

Keeping the estimated pose and shape parameters fixed, we then optimize for a consistent light position $\vec{\ell}$ over different body postures by minimizing an objective function based on the mismatch between the predicted shadow silhouettes $S_{k,t}^e(\vec{s}^t, \vec{\ell}, \vec{g})$ and the observed shadow silhouettes $S_{k,t}^o$ over several time instants t:

$$E(\vec{\ell}) = \sum_{t=1}^{T}\sum_{k=1}^{K} D\Big(S_{k,t}^e(\vec{s}^t, \vec{\ell}, \vec{g}), S_{k,t}^o\Big) \ . \tag{5.9}$$

This formulation assumes a calibrated ground plane (Section 4.4); the same objective function however can be used to also estimate the parameters of the ground plane $\vec{g}$, effectively performing camera calibration for the "shadow camera".

The location of the point light source can be parameterized as $\vec{\ell} = [\gamma, \phi, z]$, where ϕ and γ are the elevation angle from the ground plane and the azimuth angle in the plane respectively and z is the height of the light source above the ground. In the case of a directional light source assumed to be an infinite distance away from the scene, only the first two parameters need be estimated.

5.8 Experiments and Evaluation

We perform experiments on three sequences denoted by SEQ^{L1}, SEQ^{L2} and $SEQ^{\mathrm{L1,L2}}$, with one or two light sources (L1, L2), each captured by four synchronized and calibrated color cameras. We use 500W GE PhotoFlood light bulbs and compute ground truth light positions using a commercial motion capture system by affixing reflective markers to the top and the sides of the light bulbs while turned off. The first two sequences capture the scene illuminated with different individual light sources, while the third sequence has both lights turned on. While sequences SEQ^{L1} and $SEQ^{\mathrm{L1,L2}}$ are of the same subject, sequence SEQ^{L2} contains a different subject. In all cases we fit 6 shape parameters for a reduced SCAPE body model consisting of 10 example body shapes combining men and women. The fitting is done independently in each frame, using a stochastic search technique related to annealed particle filtering as described in Section 4.7.3. This optimization strategy requires an initial estimate of the pose that is relatively close to the actual configuration. Toward that end, we predict initial poses directly from individual silhouettes using a Bayesian Mixture of Experts (BME) framework that learns a direct non-linear probabilistic mapping from image features to 3D pose (Section 4.7.1). Here we make 3D predictions from monocular camera views using shape features computed from silhouettes [Sigal *et al.* (2008)] to cope with pose ambiguities in monocular sequences.

5.8.1 Light Estimation Results

We first show how the light position can be estimated from the body model and extracted shadows. We estimate the position of each light one at the time, using different subjects. For each experiment, we assume only one light is turned on. We estimate the shape and pose of each subject at several time instants using only the foreground silhouettes as image observations. Each pose results in a different shadow and provides different constraints on the light position. Given the estimated shape and pose, we optimize Equation 5.9 for the optimal light position using a direct search approach. To initialize the search we parameterize the light location by its height from the floor and its azimuth and elevation angles. We discretize the space in a reasonable range above the person and compute the value of (5.9) for a total of 288 light positions. We then select the best location, re-discretize around it using a $7 \times 7 \times 7$ grid with a finer sampling, and repeat down to a $5mm$ discretization.

We evaluate the estimated light positions in terms of both direction and position error. In particular, we report the relative distance error as a ratio of the placement error and the distance from the light source to the average location of the subject on the floor.

The results in Table 5.1 suggest that the cast shadows are very good at recovering the light direction, but not its precise location. This is due to the fact that small changes in the direction of incoming light induce large changes in the cast shadow while, at the distances found here, variations in the light distance to the subject produce smaller variations in cast shadows.

	Placement Error	Relative Distance Error	Direction Error
Light 1	140mm	4.64%	0.87°
Light 2	218mm	7.40%	1.83°

Table 5.1: **Estimated Light Position and Distance Accuracy.** The position of each light is estimated using different subjects. Light 1 was estimated with subject 1 (sequence SEQ^{L1}, 10 poses) and light 2 with subject 2 (sequence SEQ^{L2}, 10 poses).

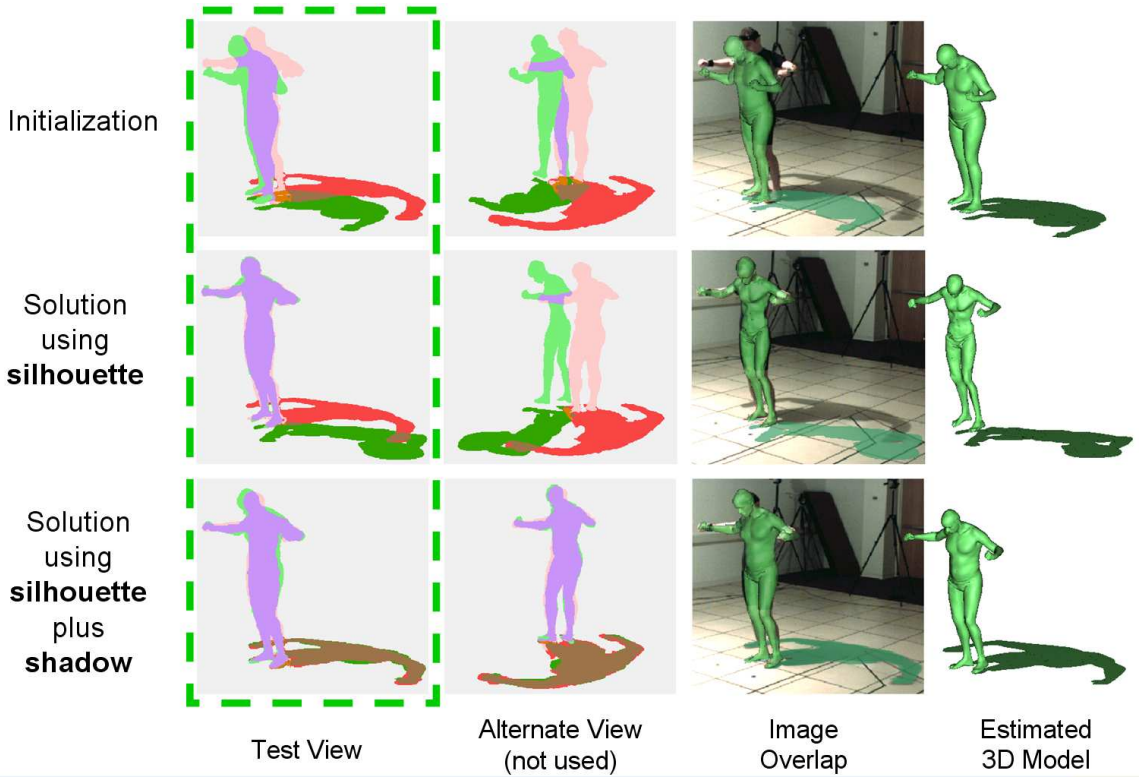


Figure 5.5: **Contribution of Shadows to *Monocular* Pose and Shape Estimation (Sequence SEQ^{L1}, Subject 1).** Fitting results are presented in the monocular case with and without shadows. Row 1 shows the automatic initialization from silhouettes ([Sigal *et al.* (2008)]). Row 2 shows estimated pose and shape based on monocular silhouette optimization. Row 3 shows the improvement obtained by adding shadows to the optimization. Color key: light green = model silhouette (F^e); light red = image silhouette (F^o); light purple = agreement between F^o and F^e; dark green = model shadow (S^e); red = image shadow (S^o); brown = agreement between S^o and S^e. The estimation uses only the left-most view of each frame, with the alternate view presented for visual inspection. The right two columns show the recovered 3D shape projected into the image and rendered in green along with its shadow.

5.8.2 Body Fitting Results Using Shadows

We now presents results for the problem of estimating pose and shape in the presence of shadows. To demonstrate the usefulness of the shadows, we first consider the less constrained monocular case. Figures 5.5 and 5.6 show examples of the initialization (top row), estimated pose and shape

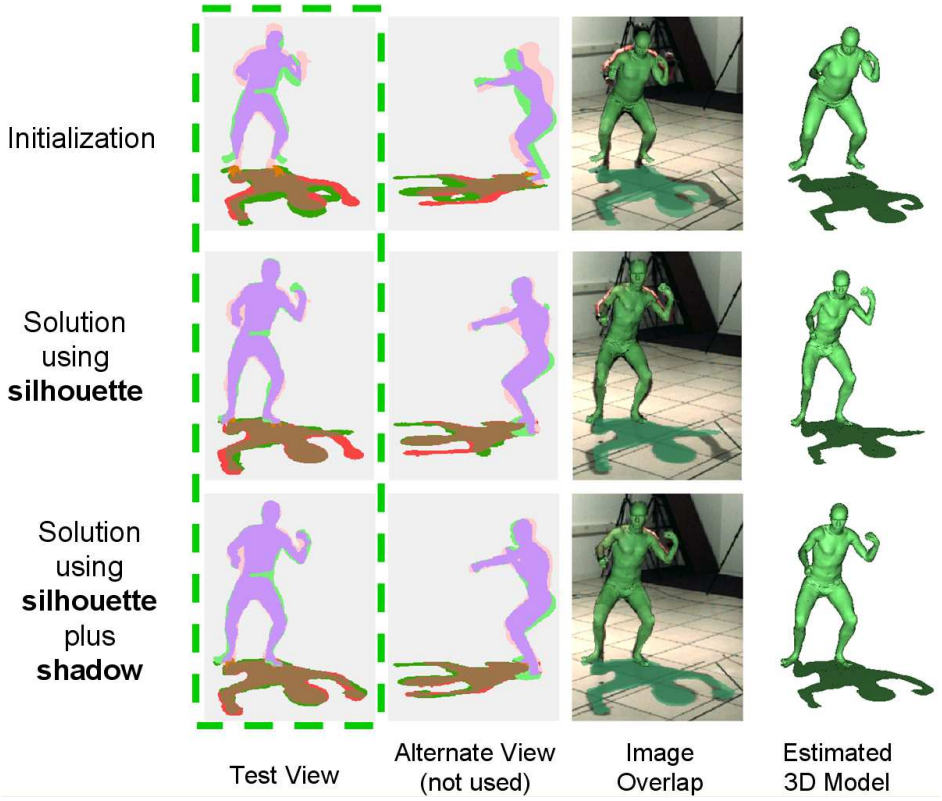


Figure 5.6: **Contribution of Shadows to *Monocular* Pose and Shape Estimation (Sequence SEQ^{L2}, Subject 2).** Fitting results are presented in the monocular case with and without shadows, but for a different subject and a different position of the light than in Figure 5.5. For details, see the caption of Figure 5.5.

based on monocular foreground silhouettes alone (middle row), and using both foreground and shadows (bottom). In particular, the example in Figure 5.5 illustrates why monocular views are inherently ambiguous and the optimization is under-constrained. While the fit of the foreground in the optimized view is almost perfect, an alternate camera view (middle column) reveals that the recovered pose is far from the truth; note also that the projected shadow (dark green) does not match the observed shadow (red). The entire body of the person was actually estimated closer to the camera and the body shape was reduced accordingly. The shadow in this case is sufficient to fully constrain the pose and shape estimation (bottom). The example in Figure 5.6 shows a less dramatic impact of the shadow which nonetheless correctly constraints the position of the left arm. This demonstrates that shadows can provide powerful constraints for human pose and shape estimation from monocular images. We note that in the absence of shadows the problem is clearly under-constrained in the monocular case, as the optimization is able to explain the foreground silhouette very well with a non-optimal solution. Using the shadows enables a more accurate recovery of pose.

We quantitatively evaluate pose recovery accuracy using joint placement error. Our video capture is synchronized with a marker-based motion capture system which is used to acquire ground truth

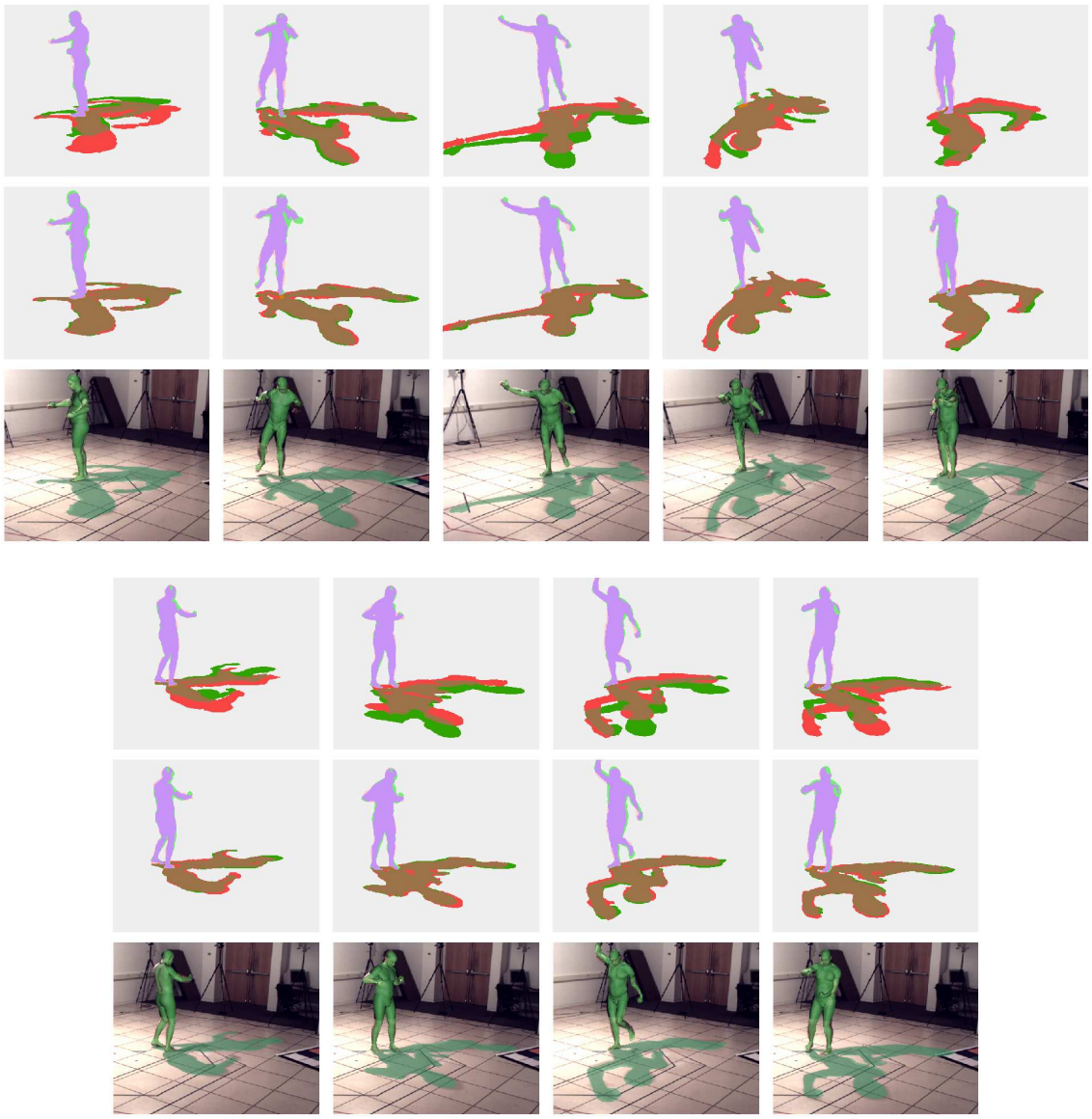


Figure 5.7: **Monocular Pose and Shape Estimation with Shadows from Two Lights (Sequence $SEQ^{L1,L2}$, Subject 1).** Fitting results are presented in the monocular case with and without shadows, with two lights illuminating the scene. The top row contains the results that do not take the shadows into account, while the middle row exploits the shadows during model fitting. While the silhouettes are explained well, shadows are not well matched unless they are explicitly taken into account. The bottom row contains the estimated 3D body model with shadows overlaid over the original images. See the caption of Figure 5.5 for description of the color coding.

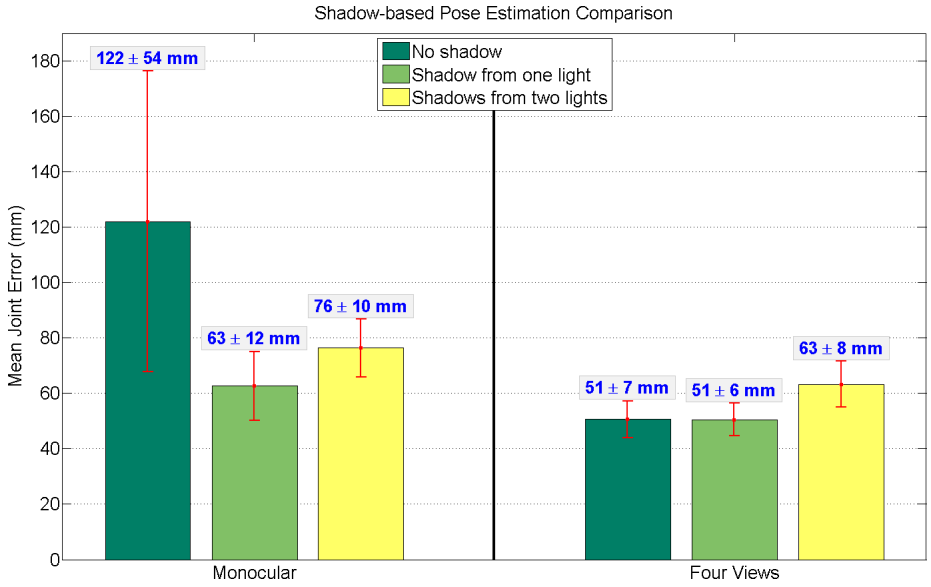


Figure 5.8: **Shadow-based Pose Estimation Comparison.** Quantitative evaluation of the pose reconstruction using different numbers of lights and camera views for optimization. The evaluation is based on the average 3D joint location prediction error. Shadows prove most useful for monocular sequences when generated by a single point light source, but provide no benefit when the subject is observed from 4 camera views.

joint locations for each frame. We compute the average joint location error over 12 joints (hips, knees, ankles, shoulders, elbows and wrists) according to Equation 4.17. Figure 5.8 shows the mean and standard deviation of the errors over all the joints in mm. The results suggest that a single shadow present in the scene offers a significant improvement in monocular pose estimation, decreasing the error from $122 \pm 54\,mm$ to $63 \pm 12\,mm$. The addition of a second point light source actually reduces accuracy to $76 \pm 10\,mm$, although the use of shadows in this case is still very beneficial to pose estimation as illustrated in Figure 5.7. In the limit, too many lights will cause shadows to overlap and lose their intrinsic information. It is important to note that these experiments, while monocular, use shadows and light positions computed with the multi-view methods described in Sections 5.5.2 and 5.7 respectively. These are likely to be much better estimated than in the single-view case. Thus these experiments represent an upper bound on the kind of performance that can be expected.

If all four camera views are used for model fitting, the shadows appear to offer no benefit (Figure 5.8). The "shadow camera" is the equivalent of the fifth camera to the system, providing minimal additional information. Representative examples are presented in Figures 5.9 and 5.10 for different subjects.

Note that the spatial configuration of the cameras, light sources and the subject affect the performance of the system. Intuitively, a cast shadow is most informative when the camera viewing direction is orthogonal to the plane containing the light and subject. If the light, camera and subject are relatively collinear, then there is little new information present in the shadow. This is

more likely to happen in a multi-camera case. In our case, one light was placed above one of the cameras, providing little contribution to the optimization cost function.

We conclude that cast shadows may provide an inexpensive means to generate a "second view" for body model estimation in a controlled environment with a single camera.

5.9 Discussion

We have presented a framework for exploiting strong lighting in the estimation of 3D human shape and pose. We have shown that a sufficiently rich model of the human body makes estimation of light sources practical. We have also shown that knowing the lighting in a scene can make human pose estimation more reliable. In contrast to the prevailing wisdom that strong lighting should be avoided, or that vision algorithms should be invariant to lighting, we show that strong lighting is actually beneficial to pose and shape estimation. These conclusions extend beyond the case of human pose considered here.

In particular we showed that cast shadows can be treated as an extra "camera" and used to improve pose and shape fitting even for monocular images. We also showed how these cast shadows can be used to estimate the light source position in the scene to effectively calibrate the "shadow camera". The results presented however rely on fairly accurate segmentation of the shadow regions in images, obtained using a novel approach that integrates shadow pixel constraints over multiple camera views and assumes a calibrated ground plane. Also assumed known is the number of the light sources, but not their locations.

In future work, we would like to extend the analysis for monocular images using single-view segmented shadows which are harder to estimate reliably and can negatively impact the performance. Additionally, we will experiment with jointly estimating the light, shape and pose in an iterative fashion, as well as exploit shadows in tracking monocular sequences. Future work will also consider extended light sources (which can be modeled as many point light sources) and will combine cast shadows with shading [Bălan *et al.* (2007b)] for light source estimation.

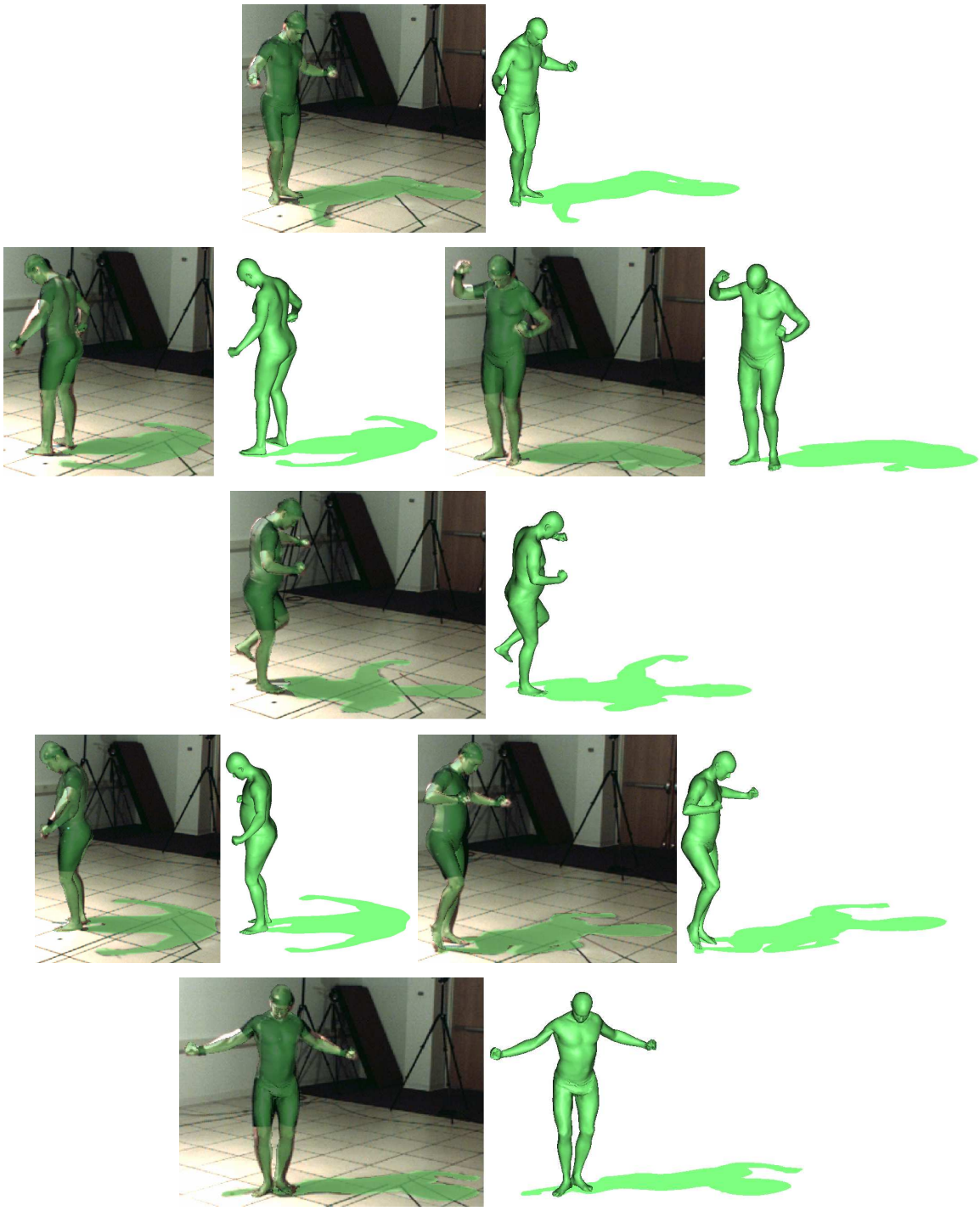


Figure 5.9: **Multi-camera Estimation of Shape and Pose from Silhouettes and Shadows.** SCAPE models estimated in a laboratory setting with 4 calibrated cameras plus the shadow cast on the ground from a single light source. Each body model was estimated independently in each frame. There are no shape constraints imposed between different poses of the body.

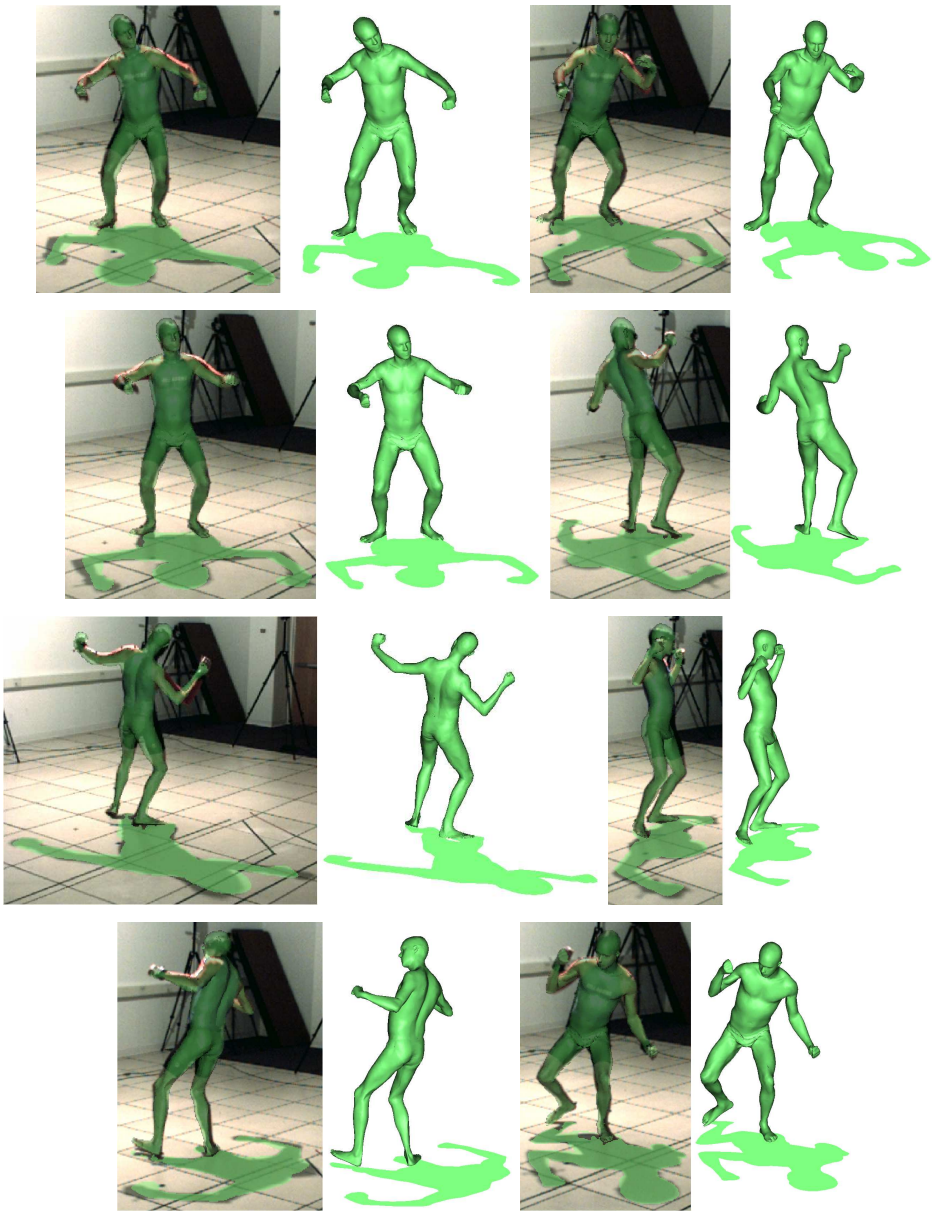


Figure 5.10: **Multi-camera Estimation of Shape and Pose from Silhouettes and Shadows.** Additional results obtained for a second subject in the same scenario as in Figure 5.9 but with a different light source.

Chapter 6

Shape under Clothing

6.1 Introduction

In this chapter we address the problem of reliably estimating a person's body shape from images of that person wearing clothing. Estimation of body shape has numerous applications particularly in the areas of tracking, graphics, surveillance and forensic video analysis. To be practical any method for estimating body shape must recover a representation that is invariant to changes in body pose. To that end, we exploit the SCAPE body model and show that the 3D body shape parameters are largely invariant to body pose. Additionally, such a model must be robust to clothing which obscures the true body shape. Here we build on the concept of the visual hull, which represents a bound on the underlying shape. In the case of a clothed person, the visual hull may only provide a loose bound on body shape. To gain tighter bounds we exploit the pose-invariance of the shape model to combine evidence from multiple poses. Specifically a clothed form provides constraints on the body shape underneath. As a person moves, the constraints provided by the visual hull change as the clothing becomes looser or tighter on different body parts. We combine these constraints to estimate a *maximal silhouette-consistent parametric 3D shape*. Using a unique dataset of subjects with both minimal and normal clothing we demonstrate that a person's body shape can be recovered from several images of them wearing clothes.

To our knowledge this is the first work to attempt to recover a detailed estimate of a person's 3D body shape from natural (i. e. standard CCD) images when the body is obscured by clothing. The approach is illustrated in Figure 6.1. In the previous two chapters we have shown that the parameters of the SCAPE body model can be directly estimated from image silhouettes, but those approaches were restricted to people wearing tight-fitting clothing. Here we go beyond that work to 1) estimate a single person-specific body shape by integrating information from multiple poses; and 2) infer the 3D body shape even when it is obscured by loose-fitting clothes.

Our method rests on two key hypotheses. First: human body shape can be recovered independently of body pose. We test this hypothesis using a unique dataset of "naked" subjects in

Figure 6.1: **Shape under Clothing.** Body shape recovery for two clothed subjects. For each subject we show (left to right) one of four input images, the 3D body model superimposed on the image (giving the impression of "X-ray" vision), and the estimated body model. The subjects' faces have been blurred for privacy considerations.

several poses captured with 4 calibrated cameras. We estimate their body shape in each pose both independently and in a batch fashion that combines information from multiple poses. We find that the variability in shape parameters across pose is small relative to the variability across subjects. We exploit this relative pose-independence of shape to combine multiple poses to more accurately estimate a single 3D body shape.

The second hypothesis is that images of the human body in clothing provide sufficient constraints to infer the likely 3D body shape. Of course a garment or costume could be worn which completely obscures or provides false information about the body shape. In normal "street clothes" however, we argue that many constraints exist that can be combined with a learned model of 3D body shape to infer the true underlying shape. Even when people wear clothing various parts of their body are often seen unobscured (face, neck, hands, arms, legs); when observed, these parts provide tight constraints on body shape. To formalize this, we define the notion of a maximal silhouette-consistent parametric shape (MSCPS) that generalizes the notion of a visual hull. A visual hull has two properties [Cheung

et al. (2003b); Laurentini (1994)]. First, the true 3D object lies completely within the visual hull (and its projection into images lies within the image silhouettes). Second, each facet of the visual hull touches the surface of the object. In the case of clothing, property 1 holds but property 2 does not. The object itself is obscured such that the silhouette contours may, or may not, correspond to true object boundaries. Rather, the silhouettes provide a bound on the possible shape which may or may not be a tight bound. Note also that, with clothing, in some poses the bound may be tight in some places and loose in others and these locations may change with pose.

In place of the visual hull, we define the maximal silhouette-consistent parametric shape (MSCPS) that optimizes the following weak constraints: 1) the shape lies inside the visual hull; 2) the volume of the shape is maximal; and 3) the shape belongs to a parametric family. In our case this family is the family of 3D human body shapes. Constraint 2 is required to avoid the trivial solution where the estimated shape is made arbitrarily small. In general, each of these constraints can be viewed as a weak constraint with the last one being a statistical prior over 3D shapes. We go a step beyond previous work to deal with time-varying constraints and non-rigid, articulated objects, by requiring constraint 1 hold over multiple poses. We also use the fact that portions of the body may actually provide tight constraints on shape. For example, when a person wears short sleeves, their bare arms provide cues not only about the arm shape, but also about the overall weight of the person. Consequently we automatically detect skin regions and exploit tight constraints in these regions.

Central to our solution is a learned human body model. We go beyond previous work to use three different models: one for men, one for women, and one gender-neutral model combining both men and women. We find that gender can be reliably inferred in most cases by fitting both gender-specific models to the image data and selecting the one that best satisfies all the constraints. Given this estimated gender, we then use a gender-specific model to produce a refined shape estimate. To our knowledge this is the first method to estimate human gender directly from images using a parametric model of body shape.

In summary, the key contributions described here include: a shape optimization method that exploits shape constancy across pose; a generalization of visual hulls to deal with clothing; a method for gender classification from body shape; and a complete system for estimating the shape of the human body under clothing. The method is evaluated on thousands of images of multiple subjects.

6.2 Related Work

There are various sensing/scanning technologies that allow fairly direct access to body shape under clothing including backscatter X-ray, infra-red cameras and millimeter waves. While our body fitting techniques could be applied to these data, for many applications, such as forensic video analysis, body shape must be extracted from standard video images. This problem is relatively unexplored.

Rosenhahn *et al.* (2007) proposed a method to track lower limbs for a person wearing a skirt or shorts. Their approach uses a generative model to explicitly estimate parameters of the occluding

clothing such as the cloth thickness and dynamics. In their work, they assume the shape of the body and cloth measurements are known *a priori* and do not estimate them from image evidence. There has been recent interest in generative models of cloth [Salzmann *et al.* (2007); White *et al.* (2007)] but the huge variability in clothing appearance makes the use of such models today challenging.

Most human shape estimation methods attempt to estimate the shape *with* the clothing and many of these techniques are based on visual hulls [de Aguiar *et al.* (2007)]. Visual hull methods (including voxel-based and geometric methods) attempt to reconstruct the *observed* 3D shape with the silhouette boundary providing an outer bound on that shape. A detailed review is beyond the scope of this thesis. We focus instead on those methods that have tried to restrict the shape lying inside the visual hull. Several authors have noted that, with small numbers of images, the visual hull provides a crude bound on object shape. To address this in the case of people, Starck and Hilton (2007) combine silhouettes with internal structure and stereo to refine the 3D surface. They still assume the true surface projects to match the image silhouette features.

More generally, Franco *et al.* (2006) impose weak assumptions on the underlying shape. They define a notion of a set of visual shapes that are consistent with the observed silhouettes (silhouette-consistent). As the number of unique views tends to infinity, this set approaches the visual hull (with the exception of concavities). The key contribution of their work is the idea of adding an assumption of shape smoothness which regularizes the set of possible 3D shape solutions. The observed silhouette is always considered as providing a tight bound on the surface with the priors compensating for an impoverished set of views. Note, however, in our problem, the visual hull is not the goal. Our case is different in that the object we care about (the human body) is obscured (by clothing) meaning that observed silhouette boundaries often do not provide tight constraints. We build on the notion of a visual shape set to define a person-specific prior model of the underlying shape.

In related work Mündermann *et al.* (2007) fit a SCAPE body model to visual hulls extracted using eight or more cameras. They do this in a single pose and assume tight-fitting clothing. We use a more detailed body model than they did and do not explicitly reconstruct the visual hull. Instead, we fit directly to image data and this allows us to use a smaller number of cameras (4 in our case).

Most visual hull reconstruction methods assume rigid objects. With non-rigid clothing we find it important to integrate information over time to constrain the underlying 3D shape. In related work, Cheung *et al.* (2005) combine information over time by performing rigid alignment of visual hulls at different time instants and then refinement of the hulls using more constraints to get a tighter bound on the shape. Knowing a rigid alignment over time effectively provides additional views. They also extend this idea to articulated body parts but focus only on recovering the bounding volume. Grauman *et al.* (2003a,b) estimate a 3D shape consistent with a temporal sequence of silhouettes using assumptions on the smoothness and shape transitions. They apply this method to silhouettes of humans and recover a visual hull using an example-based non-parametric model of body shapes. They do not use a parametric body model or explicitly attempt to infer the shape under clothing.

Most previous work on gender classification from images has focused on faces (e.g. [Moghaddam

and Yang (2002)]), but in many situations the face may be too small for reliable classification. The other large body of work is on estimating gender from gait (e.g. [Davis and Gao (2004); Huang and Wang (2007); Li *et al.* (2008)]). Surprisingly, this work typically takes silhouettes and extracts information about gait while throwing away the body shape information that can provide direct evidence about gender. We believe ours is the first method to *infer* a parametric 3D human body shape from images of clothed people and to use it for gender classification.

6.3 Clothing

In the previous chapters we have established the basic model, its optimization and its application to shape estimation in the absence of loose clothing. Estimating the human shape is made more challenging when the subject is wearing loose clothing that obscures the true form of the naked body.

We define an observation model that deals with clothing robustly using the concept that silhouettes in 2D represent bounds on the underlying body shape. Consequently the true body should fit "inside" the image measurements. In the case of a clothed person, the observations may only provide loose bounds on body shape. This makes the problem significantly under-constrained and therefore requires additional assumptions to regularize the solution.

Additionally, the objective function is made aware of the clothing, or lack of it, in different regions of the body. Regions in the image data are identified that are likely to be skin. In these regions, the optimization method constrains the fitted body model to match the silhouette contours. In the remaining clothed (or hair) regions, the objective function is modified so that it does not have to strictly match the observations.

Moreover, it is noted that clothing provides constraints on body shape that vary with pose as illustrated in Figure 6.4(*bottom*). In each posture depicted, the clothing is loose or tight on different parts of the body. Each posture provides different constraints on the possible underlying body shape. Constraints from multiple poses, such as these, are accumulated by a consistent body model across poses.

6.3.1 Maximal Silhouette-Consistent Parametric Shape

We introduce the concept of a maximal silhouette-consistent parametric shape that weakly satisfies the following constraints:

1. the projected model falls completely inside the foreground silhouettes;
2. the model attempts to fill the image silhouette mainly in regions with tight or no clothing;
3. the intrinsic shape is consistent across different poses; and
4. the shape of the object belongs to a parametric family of shapes (in our case human bodies).

Each aspect is discussed below.

The first constraint is satisfied by penalizing the regions of the projected model silhouette, $F_k^e\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right)$, that fall outside the observed foreground silhouette F_k^o. The silhouette match error in camera k from Chapter 4 is separated into two pieces:

$$E_{1pose}^k\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right) = E_{inside}^k\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right) + E_{expand}^k\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right) \tag{6.1}$$

For the "inside" term, the same distance function as defined in Chapter 4 is used:

$$E_{inside}^k\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right) = \tilde{d}^\tau\left(F_k^e\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right), F_k^o\right) . \tag{6.2}$$

For the second constraint, it is desirable that the projected model explain as much of the foreground silhouette as possible; if the subject were not wearing clothing this would just be the second term from the minimal-clothing case: $\tilde{d}^\tau\left(F_k^o, F_k^e\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right)\right)$. In the more general setting where people wear clothing or interact with objects, the observed foreground silhouettes will be too large producing a bias in the shape estimates. To cope with this, several strategies are employed. The first is to down-weight the contribution of the second constraint, meaning it is more important for the estimated shape to project inside the image silhouette than to fully explain it. The second is to use features in the image that are more likely to accurately conform to the underlying shape. In particular, skin-colored regions are detected (see Section 6.3.4) and, for these regions, the second constraint is given full weight. The detected skin regions are denoted by F_k^s and the non-skin regions of the observed foreground silhouette by $F_k^o \setminus F_k^s$. Third, in the non-skin regions a robust penalty function controlled by a parameter $\tau^c < \tau$ is employed. Recall that the distance function, $\tilde{d}^\tau$, already has a threshold τ on the maximum distance, which makes the term robust to segmentation errors. In putative clothing regions this threshold is reduced to τ^c. When the clothes are tight (or skin is being observed), it is desired that the error term increasingly penalize non-skin regions even when they are far from the model silhouette. In this case, a large threshold τ is appropriate. However, if the clothes are expected to be loose, a small threshold τ^c effectively disables the silhouette distance constraint in non-skin regions. It is possible to apply the robust operator also to the skin term (with a corresponding τ^s threshold greater than τ^c) to protect against errors in skin detection (but typically $\tau^s \doteq \tau$).

The "expansion" constraint is then written as

$$E_{expand}^k\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right) = \tilde{d}^{\tau^s}\left(F_k^s, F_k^e\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right)\right) + \lambda_c \tilde{d}^{\tau^c}\left(F_k^o \setminus F_k^s, F_k^e\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right)\right) , \tag{6.3}$$

with $\lambda_c \ll 1$ (e. g., 0.1).

Different parts of the body can be obscured by different pieces of clothing with different looseness characteristics. The above formulation can be extended to incorporate any additional knowledge about the looseness of clothing in G different regions of the body. More generally, imagine the image silhouette is segmented into regions corresponding to different classes of clothing with associated looseness / tightness properties. Such classes can represent broad categories such as skin versus non-skin regions as described above, or can include more refined categories such as hair, t-shirt,

jacket etc. Each category, g, has an associated looseness threshold τ^g and relative importance λ_g. The "expansion" constraint can be generalized as:

$$E^k_{expand2}\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right) = \sum_{g=1}^{G} \lambda_g \tilde{d}^{\tau^g}\left(F^g_k, F^e_k\left(\chi, \vec{\beta}^\chi, \vec{\theta}\right)\right) . \tag{6.4}$$

Segmentation of the image into G labeled regions can be obtained using general skin, clothing and hair classifiers described in the literature.

When a clothed subject is observed with clothing in only a single pose, the shape estimate may not be very accurate. Additional constraints can be obtained by observing the subject in different poses. This requires estimating a different set of pose parameters in each frame, but a single body shape consistent for every pose:

$$E^k_{multipose}\left(\chi, \vec{\beta}^\chi, \mathbf{\Theta}\right) = \sum_{p=1}^{P} E^k_{1pose}\left(\chi, \vec{\beta}^\chi, \vec{\theta}^p\right) , \tag{6.5}$$

where $\mathbf{\Theta} = \left(\vec{\theta}^1, \cdots, \vec{\theta}^P\right)$ represents the different body poses.

In the case of multiple synchronized camera views where the images are taken at the same time instant, we integrate the constraints over the K camera views to optimize a consistent set of model parameters given all sensor data

$$E_{sensor}\left(\chi, \vec{\beta}^\chi, \mathbf{\Theta}\right) = \sum_{k=1}^{K} E^k_{multipose}\left(\chi, \vec{\beta}^\chi, \mathbf{\Theta}\right) . \tag{6.6}$$

Finally, the sensor constraints are combined with domain knowledge constraints to ensure the shape remains within the family of human shapes by exploiting the availability of a large database of body shapes. It is not required that the estimated shape exist in the database; instead, computed statistics on shape variability are used to penalize unlikely shape parameters, $E_{shape}(\chi, \vec{\beta}^\chi)$. Pose priors $E_{pose}(\vec{\theta}^p)$ that penalize un-natural poses exceeding anatomical joint angle limits are also enforced. Details about the shape and pose priors are provided in Sections 6.3.2 and 6.3.3 respectively.

The final objective function is given by

$$E_{clothes}\left(\chi, \vec{\beta}^\chi, \mathbf{\Theta}\right) = E_{sensor}\left(\chi, \vec{\beta}^\chi, \mathbf{\Theta}\right) + E_{shape}(\chi, \vec{\beta}^\chi) + \sum_{p=1}^{P} E_{pose}(\vec{\theta}^p) . \tag{6.7}$$

It should be also noted that the terms in the objective functions can all be weighed by different scaling constants to change the relative importance of each term.

6.3.2 Shape Prior

A penalty is defined for body shapes that do not conform to the observed statistics of true human bodies. The SCAPE body shape model is learned from training bodies and the resulting PCA model includes the variance along each principal component direction. The variance $\sigma^2_{\beta,\chi}$

along these shape-deformation directions characterizes the shape of the population being modeled. A standard Gaussian noise assumption would lead to an error term defined by the Mahalanobis distance of a body from the mean.

To avoid biasing the estimates toward the mean we use a different penalty term. Specifically, a shape prior is formulated that penalizes extreme shapes while assigning the same fixed cost for more average shapes:

$$E_{shape}\left(\chi, \vec{\beta}^{\chi}\right) = \sum_{b} \max\left(0, \frac{|\beta_b^{\chi}|}{\sigma_{\beta,\chi,b}} - \sigma_{\beta}^{thresh}\right)^2, \tag{6.8}$$

where b ranges over all the shape parameters. Typically $\sigma_{\beta}^{thresh} = 3$ is chosen, thus penalizing only those shapes that are more than 3 standard deviations from the mean.

6.3.3 Pose Prior

There are some poses that are anatomically impossible or highly unlikely. The elbow, for example, cannot extend beyond a certain angle. To control this, a prior is enforced on body pose that is uniform within joint angle limits and only penalizes poses beyond those limits. Impossible joint angles are penalized, similar in formulation to the shape prior:

$$E_{pose}\left(\vec{\theta}\right) = \sum_{j} \left(\frac{\max\left(0, \theta_j^{\min} - \theta_j, \theta_j - \theta_j^{\max}\right)}{\sigma_{\theta,j}} \right)^2 + w \sum_{j} \left(\max\left(0, \frac{\left|\theta_j - \theta_j^0\right|}{\sigma_{\theta,j}} - \sigma_{\theta}^{\text{thresh}}\right) \right)^2, \tag{6.9}$$

where j ranges over all the pose parameters. Note that both the angle bounds $\left[\vec{\theta}_j^{\min}, \vec{\theta}_j^{\max}\right]$ and the variances $\sigma_{\theta,j}^2$ can be specified from anthropometric studies or learned from motion capture data. The second term penalizes poses that deviate more than $\sigma_{\theta}^{\text{thresh}}$ standard deviations (typically 3) from an initial pose θ_j^0. This second term is appropriate for cases where an approximate initial pose is pre-specified and known. In such cases, w is set to 1; if the initial pose is unknown, w is set to 0.

6.3.4 Image Skin Detection and Segmentation

There are many algorithms in the literature that perform skin detection (e.g. [Jones and Rehg (2002)]). Many of these deal with variations in lighting and skin tone across different people and can be quite accurate. Clothing detection is a harder problem due to the wide variability of materials, colors, and patterns used to make clothing. Hair detection has also received some attention. In our case, skin detection is sufficient to constrain the remainder of the foreground region to be classified as "clothing". Skin and clothing regions will be treated differently in the fitting process.

We describe a method for segmenting an image into skin and non-skin regions, although the precise formulation is not critical. In order to detect skin colored regions in an image, a skin detector can be built from training data using a simple non-parametric model of skin colors in hue and saturation space. Using a large dataset of images that have been segmented into skin or non-skin, a normalized joint histogram $P(H, S|skin)$ of Hue and $Saturation$ values is built for the skin

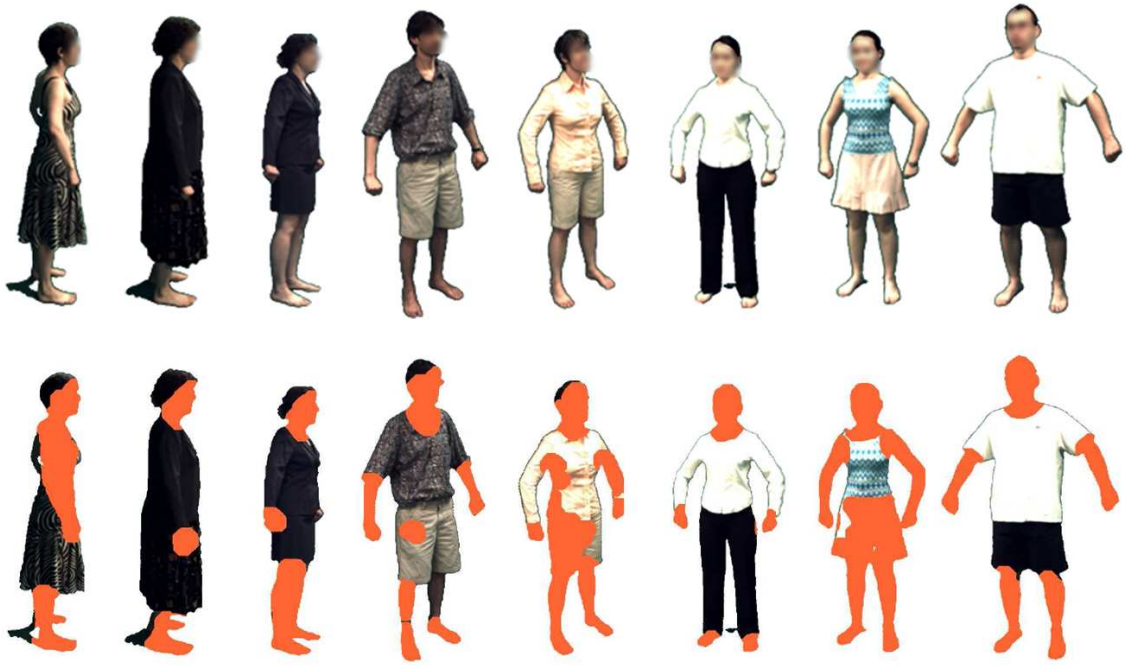


Figure 6.2: **Skin Segmentation.** Examples of segmented people and the regions identified as skin, shown in orange. Note that this segmentation need not be perfect to provide useful constraints on the shape fitting.

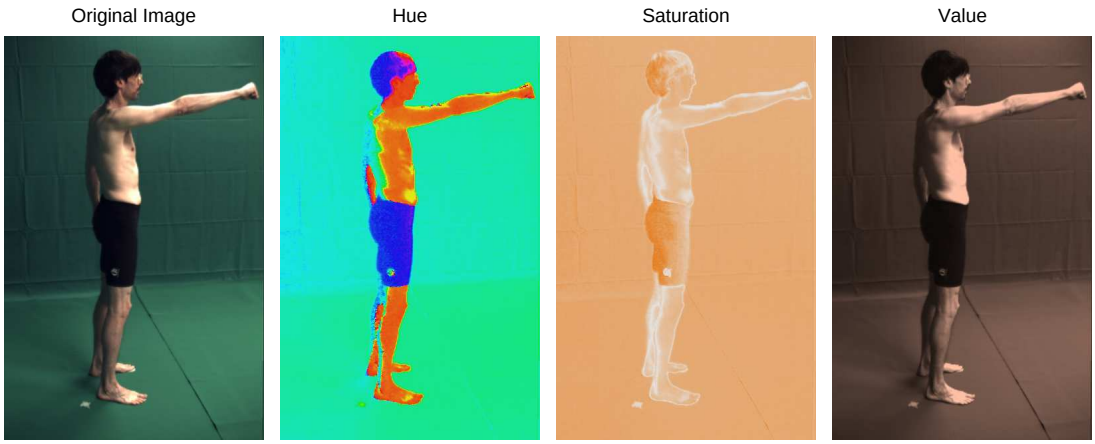
pixels. A threshold on the histogram is used to obtain a binary skin classifier for $(Hue, Saturation)$ pairs:

$$P(H, S|skin) \geq threshold\,. \tag{6.10}$$

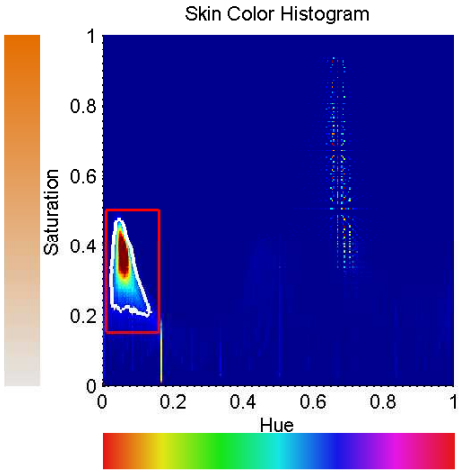
After individual foreground pixels have been classified as being skin or not skin, several standard image filters are applied to improve the segmentation, including dilation, median filtering, and removal of small disconnected components. Note that, as a final step, we dilate the skin regions by a few pixels within the foreground silhouette regions in an attempt to capitalize on the nearby true negative skin pixels lying along the edge of the foreground silhouettes. Figure 6.2 shows several examples of people in different clothing and the identified skin regions.

Skin Classifier Training Procedure

We automate the process of training the skin detectors from unlabeled training data. We rely on the subset of the images from the Clothing Dataset (Section 6.4.1) where the subjects wear minimal black clothing and build an aggregate color histogram of the foreground pixels. By switching from the RGB to the HSV color space, the $Value$ channel can be ignored, which captures mostly lighting intensity information (see Figure 6.3a). In Figure 6.3b a joint histogram is constructed for Hue and $Saturation$ channels. Hair and pants have Hue colors that can easily be pruned away as outliers. We use a threshold on the histogram inside the red box to obtain a binary classifier for $(Hue, Saturation)$



(a) RGB to HSV Conversion



(b) Skin Color Histogram

Figure 6.3: **Learning a Skin Classifier. (a)** Training images of subjects wearing minimal black clothing are separated into HSV color channels for skin detection. The $Value$ channel contains lighting information and is ignored. Foreground segmentation is also performed by chroma-keying the background. **(b)** A joint $Hue - Saturation$ histogram is build from foreground pixels. It is reasonable to expect the hue of the skin to be in the interval [0,0.15] and the saturation in [0.15,0.50] (red box). Values outside the box are attributed to clothing and hair, as well as green spill-over from the indirect reflected light off the background. The white contour defines the classification boundary of the skin detector. Note that the triangular shape of the classifier makes the non-parametric representation a better choice. The Hue values are better constrained for higher saturation values.

pairs (Equation 6.10).

For our experiments, multiple skin detectors are trained. We use a leave-one-out cross-validation method and train one classifier for each person in the Clothing Dataset using all the other people in the database. Hence the skin of the left-out subject is segmented using a skin model that excludes his/her data (Figure 6.2). We also train one skin model for each camera view, to account for color variations in different cameras.

6.4 Experiments and Evaluation

We perform experiments on a novel clothing dataset of thousands of images of clothed and "naked" subjects captured in a controlled environment using green-screening and in pre-defined poses. In addition, we also show results on the HumanEva dataset [Sigal *et al.* (2010)] in less than ideal circumstances. Given initial poses, the optimization is done using a gradient-free direct search simplex method as described in Section 4.7.4. This optimization over 27 pose parameters and 20 shape parameters for a SCAPE model in a single pose given 4 camera views with resolution 656 x 490 takes approximately $40min$ on a 2GHz processor with a Matlab implementation. In the case where integration of information across multiple poses is performed, the optimization process alternates between optimizing a single set of shape parameters applicable to all postures, and optimizing the pose parameters $\vec{\theta}^p$ independently for each posture.

6.4.1 Clothing Dataset

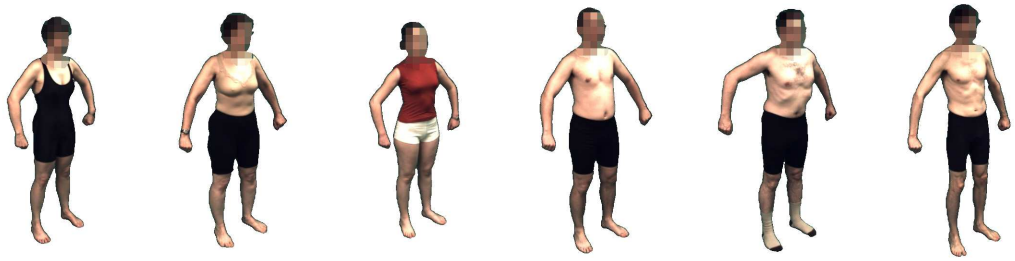
To test our ability to infer shape under clothing we collected a dataset of 6 subjects (3 male and 3 female); a small sample of images from the dataset is shown in Figure 6.4. Images of the subjects were acquired in two conditions: 1) a "naked condition" (NC) where the subjects wore minimal tight fitting clothing (Figure 6.4a), and 2) a "clothed condition" (CC) in which they wore a variety of different "street" clothes (Figure 6.4b). Each subject was captured in each condition in a fixed set of 11 postures, several of which are shown in Figure 6.4c. All postures were performed with 6 - 10 different sets of "street" clothing (trials) provided by the subjects. Overall, the dataset contains 53 trials with a total of 583 unique combinations of people, clothing and pose (a total of 2332 images).

For each of these, images were acquired with four hardware synchronized color cameras with a resolution of 656 x 490 (Basler A602fc, Basler Vision Technologies). A full green-screen environment was used to remove any variability due to imprecise foreground segmentation. The cameras as well as the ground plane were calibrated using the Camera Calibration Toolbox for Matlab [Bouguet (2000)] and the images were radially undistorted. Foreground silhouette masks were obtained using a standard background subtraction method, performed in the HSV color space to account for background brightness variations induced by the presence of the foreground in the scene (e.g. shadows).

6.4.2 Shape Constancy

In previous chapters we only optimized the body shape at a particular time instant. Here we take a different approach and integrate information about body shape over multiple poses. Our first hypothesis is that the SCAPE model provides a representation of body shape that is invariant to body pose. To test this hypothesis we optimize body shape and pose for each posture independently in the "naked condition" (NC).

Figure 6.5a (top) shows three examples of the body shape recovered for one of the subjects in this fashion using the minimum clothing objective function from Equation 4.6. We plot in Figure 6.5c the aggregate variance v_b in the recovered shape coefficients across pose for all subjects (yellow) versus

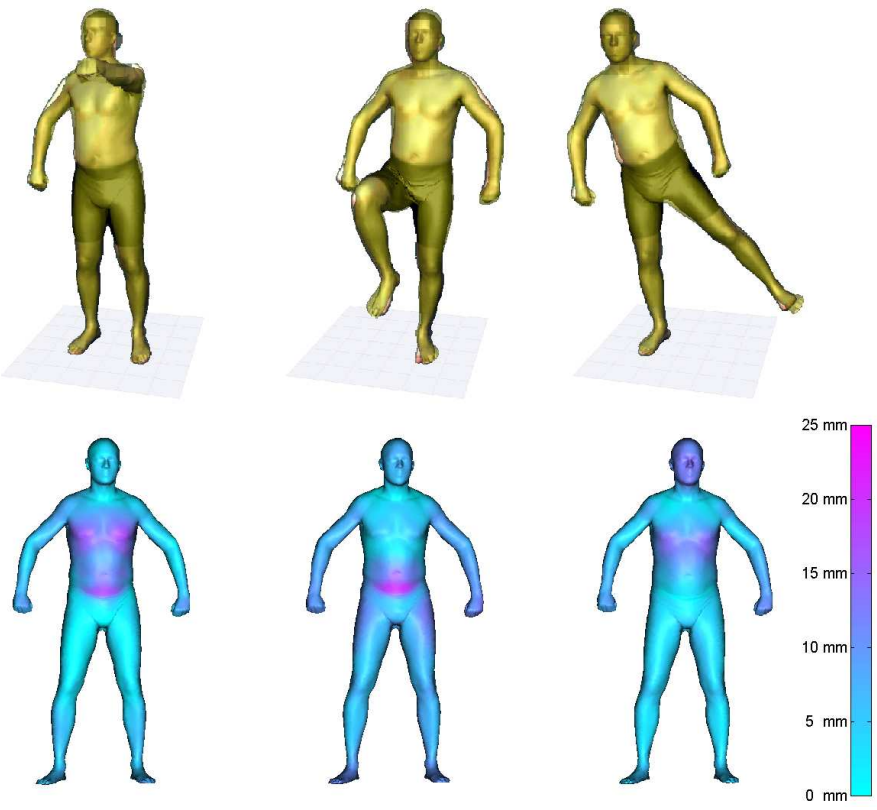


(a) Naked Condition (NC)

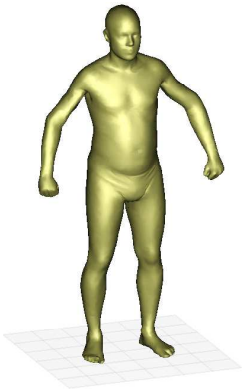
(b) Clothed Condition (CC)

(c) A Variety of Poses

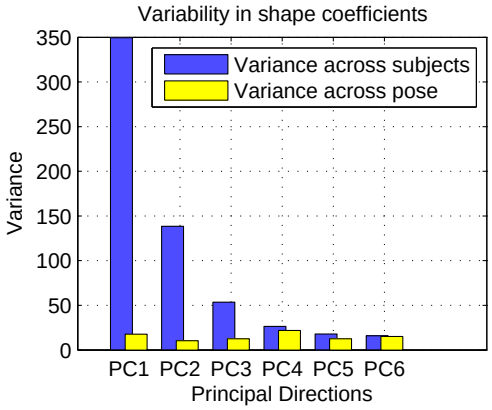
Figure 6.4: **Clothing Dataset.** Example images from the clothing dataset shown here after background subtraction. **(a)** All subjects in the "naked condition" (NC); **(b)** single subject in the "clothed condition" (CC); **(c)** subject in 11 different poses.



(a) Single Pose Fitting



(b) Batch Fitting



(c) Shape Variability

Figure 6.5: **Invariance of Body Shape to Pose.** (a) **Top row:** 3D body reconstructions independently estimated in several poses using the clothing-oblivious formulation from Chapter 4. Only one of the four camera views used in the optimization is shown, with the 3D model superimposed over the segmented image. **Bottom row:** The estimated shapes above displayed in the canonical pose and textured with an error map measuring deviations between shapes estimated in a single pose and the shape estimated in batch mode. (b) Model recovered in batch by combining constraints across 11 different poses. (c) Variance in shape coefficients across subjects and poses.

the variability in these shape coefficients $\sigma^2_{\beta,b}$ (Section 3.5.5) across all subjects in the CAESAR dataset (blue). The aggregate shape variability υ_b in our estimates is computed for each shape coefficient $\hat{\vec{\beta}}_b$ by integrating over all poses p and all subjects j:

$$\upsilon_b = \sum_{j=1}^{J} \sum_{p=1}^{P} \left(\hat{\vec{\beta}}_b^{j,p} - \mathrm{E}_p \left[\hat{\vec{\beta}}_b^{j,p} \right] \right)^2 , \tag{6.11}$$

where $\mathrm{E}_p[\cdot]$ denotes the expected value over poses. We find the variation of the major 3D shape parameters with pose to be small relative to the variation across people (Figure 6.5c). Moreover, our estimate of the shape variation with pose also encompasses the errors introduced by the image fitting process. In contrast, there is no fitting error in computing the shape parameters for the CAESAR dataset, suggesting that the variability due to pose might have been even smaller after factoring out image fitting error.

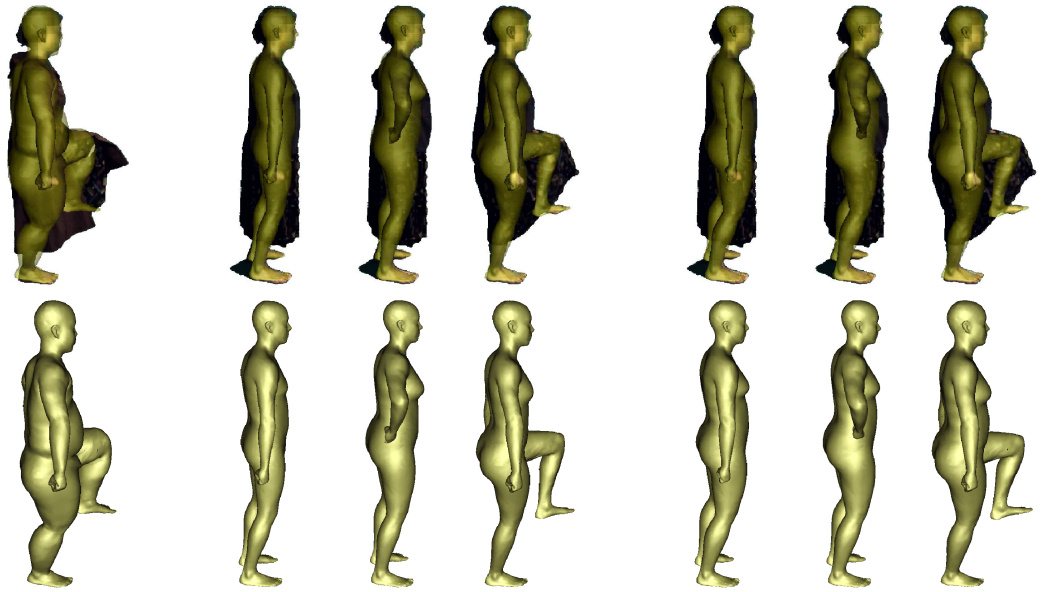
Having established the shape constancy across pose property for the SCAPE body model, we can exploit it by defining a "batch" optimization that extends the objective function to include P different poses. In the *naked case* this is simply:

$$E(\vec{\beta}, \vec{\theta^1}, \ldots, \vec{\theta^P}) = \sum_{p=1}^{P} \sum_{k=1}^{K} \tilde{d}^{\tau} \left(F_k^e(\vec{\beta}, \vec{\theta^p}), F_{k,p}^o \right) + \tilde{d}^{\tau} \left(F_{k,p}^o, F_k^e(\vec{\beta}, \vec{\theta}_p) \right) . \tag{6.12}$$

In the clothing case, the objective function in Equation 6.5 would be used instead. Figure 6.5b shows the body shape recovered by integrating across pose, obtained by alternating between optimizing a single set of shape parameters $\vec{\beta}$ applicable to all postures, and optimizing the pose parameters $\vec{\theta}^p$ independently for each posture. The examples in Figure 6.5a (bottom) demonstrate that shapes obtained with batch fitting differ from independently estimated body shapes in individual poses in subtle ways, mainly in regions close to the joints exhibiting large motions. The magnitude of the deviations however is indeed fairly small, not exceeding $2cm$. Note that establishing this property of shape invariance with respect to pose is useful for tracking applications and biometric shape analysis.

6.4.3 Qualitative Results in the Presence of Clothing

In Chapter 4 we demonstrated that the SCAPE model is able to provide a close match to image silhouettes when the person wears tight fitting clothing. With that in mind, we defined an objective function designed to maximize the overlap between the model and image silhouettes. In the presence of loose clothing however that approach is not very good. To illustrate this, consider the extreme case from Figure 6.6a depicting a women wearing a large coat. In this case the clothing-oblivious method from Chapter 4, when applied to an individual pose and without enforcing priors on shape coefficients, leads to unrealistic overestimates of shape and size. In contrast, the proposed clothing-robust and shape-constrained formulation in this chapter results in much more plausible shape estimates, although some shape ambiguity remains when estimating gender-neural shapes in different poses independently (Figure 6.6b). Formulating the problem in a batch fashion and optimizing for gender as well results in a plausible shape consistent across multiple poses (Figure 6.6c).



(a) (b) Clothing-aware Single Pose (c) Clothing-aware Batch

Figure 6.6: **Example Shapes Estimated by Three Different Methods.** (a) The clothing-oblivious method from Chapter 4 applied to an individual pose and without enforcing any priors on shape coefficients leads to unrealistic overestimates of size/shape. (b) Estimated shapes using the proposed clothing-robust formulation (6.7) but relying only on the gender-neutral shape model and applied only to individual poses. Significant variations in shape across different poses are still apparent, noting in particular inconsistencies in gender features. (c) A single gender-optimized shape model estimated using the multi-pose shape consistency constraint in Equation 6.7.

6.4.4 Shape under Clothing - Clothing Dataset

We recover the body pose and shape for all 583 independent poses and the 53 batch trials in the "clothed condition" (CC). A few representative batch results[1] are shown in Figures 6.7 and 6.8.

We quantitatively evaluate the accuracy of the estimated 3D body models using a variety of derived biometric measurements such as height, waist size and chest size. These body measurements are collected from the results of fitting the body in batch fashion from the NC data. We treat these as ground truth shape measurements for our quantitative evaluation. Figure 6.9a shows the recovered 3D body shape models used for ground truth. Displayed on the models in red are the locations used to compute derived measurements for chest and waist size. These are obtained by slicing the body mesh with a horizontal plane at a given location on the body and computing the perimeter of the convex hull of the body cross section.

We show quantitative results in Figure 6.9 and report the mean and variance of the error from ground truth measurements, both for single-pose fitting and for batch fitting. Figure 6.9 shows how errors in height, waist and chest size decrease by combining information across pose. In particular,

[1]See http://www.cs.brown.edu/research/vision/scapeClothing for results on the entire Clothing dataset.

Figure 6.7: **Clothing Dataset Batch Results.** *(top)* One of four input images after foreground segmentation. *(middle)* Estimated body model superimposed on the image. *(bottom)* Estimated 3D body model.

we find that height can be well recovered even in the presence of clothing, but that circumference measurements are somewhat impacted by loose clothing which obscures the true shape of the body. This is also supported by the larger variance observed in the case of chest and waist measurements.

6.4.5 Gender Classification

For gender classification, we estimate the pose and the first 6 shape parameters in each *test instance* using the gender-neutral shape model. After convergence, we keep the pose parameters fixed and re-estimate the shape parameters with both gender-specific shape models. The best fitting model according to the objective function corresponds to the true gender 86% of the time when the optimization is performed on individual poses (see Figure 6.10). By observing the same subject striking 11 different poses within each trial and adopting a voting strategy based on the predicted gender from each pose, the majority classification across all poses in a trial increases the accuracy

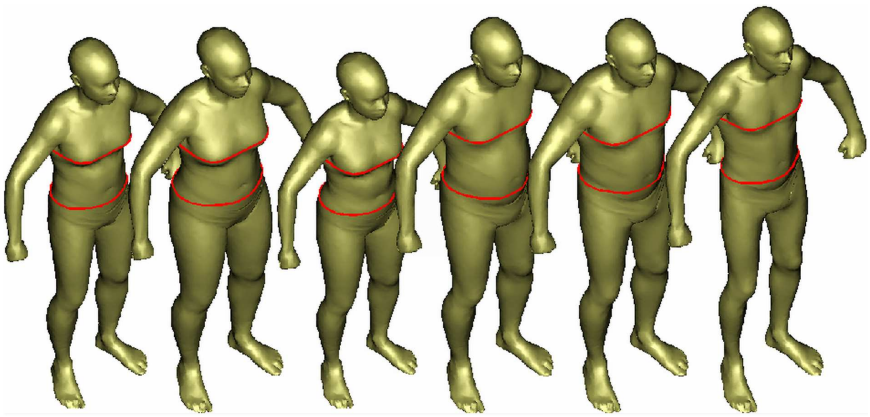
Figure 6.8: **Clothing Dataset Batch Results.** Additional results similar to Figure 6.7.

to 90.6%. Finally, by estimating the shape parameters for the two gender models in batch fashion over all poses in a trial, the gender classification improves to 94.3% accuracy on the dataset of 53 trials with natural clothing.

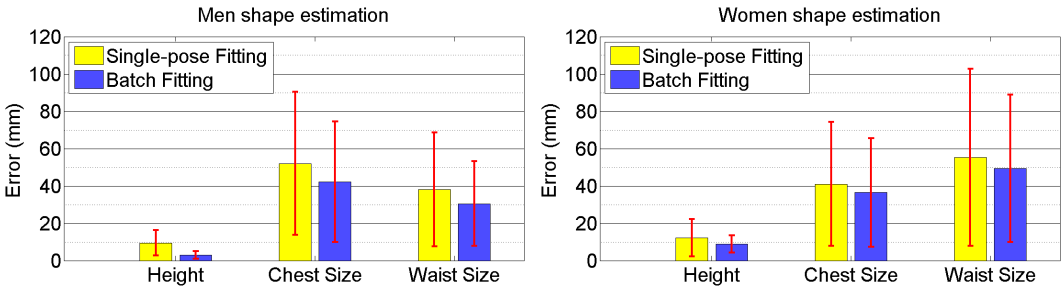
6.4.6 Shape under Clothing - HumanEva-II Dataset

To test the generality of our methods in less than ideal circumstances, we also perform experiments on the generic HumanEva dataset [Sigal *et al.* (2010)] where the poses are "natural" and background subtraction is imperfect. Specifically, we use the HumanEva-II subset which consists of two sequences with two subjects S2 and S4 walking and jogging in a circle, followed by a leg-balancing action. The subjects are wearing casual street clothing. A kinematic tree tracking algorithm using a coarse cylindrical body model is used to obtain rough initial pose estimates at each frame which were subsequently refined during shape estimation using our framework.

This dataset only contains 2 subjects, but we test our approach on approximately 200 frames in

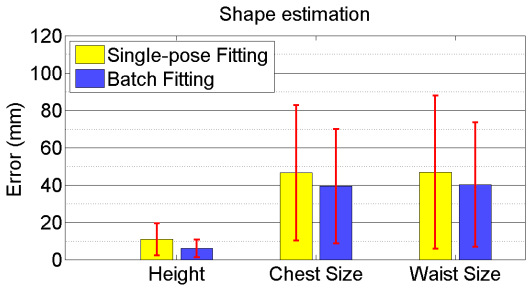


(a) Measuring Shape Estimates



(b) Shape Estimates - Males

(c) Shape Estimates - Females



(d) Aggregate Shape Estimates

Figure 6.9: **Quantitative Evaluation of Shape.** Accuracy of estimated body measurements (height, waist, chest) relative to ground truth. Batch estimation across pose decreases the errors.

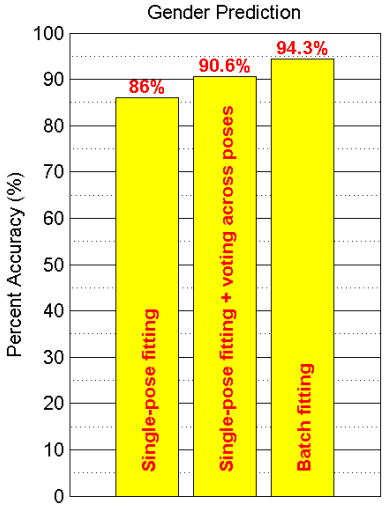


Figure 6.10: **Gender Classification.** Gender is best predicted in batch model when a single shape model is recovered that is consistent across image observations of subjects in multiple poses.

a wide variety of postures and with various levels of real silhouette corruption to estimate the body shape (Figure 6.11); these results suggest the method is relatively robust to errors in foreground segmentation. The optimization converges in all test cases we considered.

6.5 Discussion

We defined a new problem of inferring 3D human shape under clothing and presented a solution that leverages a learned model of body shape. The method estimates the body shape that is consistent with an extended definition of the visual hull that recognizes that shape bounds provided by the visual hull may not be tight. Specifically, the recovered shape must come from the parametric family of human shapes, it should lie completely within the visual hull, and it should explain as much of the image evidence as possible. We observed that by watching people move, we could obtain more constraints on the underlying 3D body shape than in a single pose. Consequently, we exploited the relative pose-independence of our body shape model to integrate constraints from multiple poses by solving for the body pose at each time instant and a single 3D shape across all time instants. We integrated a skin detector to provide tight constraints on 3D shape when parts of the body are seen unclothed. We also showed that gender could be reliably classified based on body shape and defined gender-specific shape models to provide stronger priors on the unobserved shape. The method was tested on a laboratory dataset of people in a fixed set of poses and two clothing conditions: with and "without" clothes. The latter gave us "ground truth" with which to evaluate the method. We also demonstrated the method for more natural sequences of clothed people from the HumanEva dataset [Sigal *et al.* (2010)].

We envision several extensions to this line of research. Based on the observation that in different

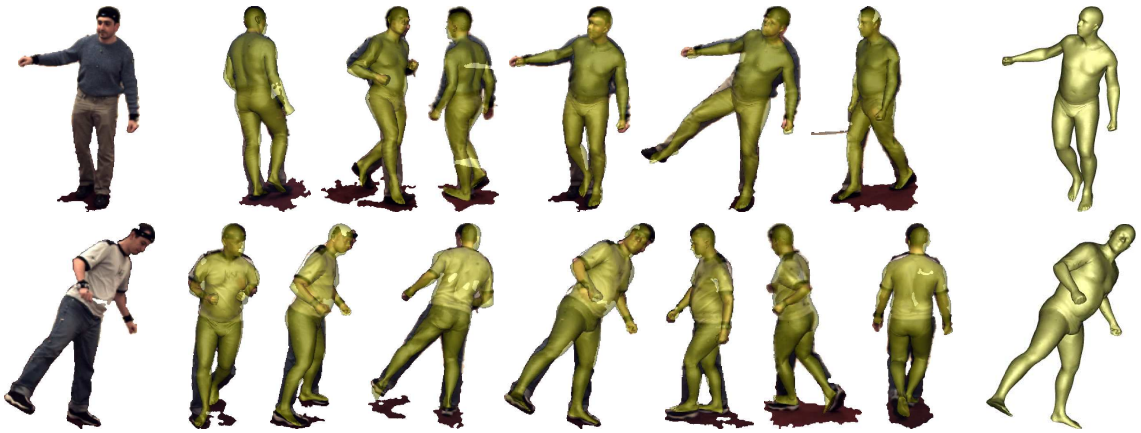


Figure 6.11: **Example Body Shapes for the HumanEva-II Dataset.** (*left*) One segmented frame; (*middle*) several frames with the estimated model overlaid; (*right*) estimated body shape.

poses clothing imposes tight silhouette constraints in different regions of the body, our batch experiments dealing with scanning the body shape under clothing relied on 11 pre-defined poses chosen in an ad-hoc manner. As future work, we would like to find a minimal set of poses that best constrain body shape. Additionally, in our example dataset none of the subjects are wearing shoes, nor are any wearing hats or big hairstyles. Future work will involve dealing with hair and shoes appropriately. As an example, the shape parameterization can be extended to include the shoe heel height as well and infer it during the optimization process. Finally, our current fitting method does not make use of any *a priori* knowledge of where clothing may be tight or loose. How tight a constraint might be in different regions on the body may be predicted from the pose of the body, type of clothing and properties of the material, and the effects of gravity and dynamics.

Chapter 7

Conclusions

7.1 Contributions

In this thesis we have presented several methods for jointly estimating shape and pose of a person from standard digital images. This has many applications in personal fitness, retail apparel and computer games. For forensic video applications, the extraction of body shape parameters could be useful in identifying suspects.

The common theme is the utilization of a state of the art graphics body model learned from examples that is very realistic and capable of representing both articulated and non-rigid deformations of the human body, as well as body shape variability between individuals. We have introduced methods for recovering the parameters of such a model directly from image data and for extracting relevant biometric information from the recovered body model, such as gender or height. A better body model enables a more robust estimation from imperfect image observations.

The proposed methods address different scenarios. The first scenario uses multiple synchronized camera views and expects the subject to wear tight fitting clothing. It uses extracted image silhouettes to match the 3D parametric model to image evidence. We are able to relax the tight fitting clothing assumption and propose an extended method for predicting shape under clothing by including a multitude of constraints: shape constancy across poses, tight constraints in skin regions and enforcement of a statistical model of human shapes. Finally, in cases where the scene is illuminated with strong lighting, we find that shadows contain valuable information that we can use to effectively reduce the number of cameras needed for model estimation, while at the same time the body can be used to calibrate the light.

7.2 Extensions

There are many future directions for this new line of research.

7.2.1 Going Beyond Silhouettes

In this work we rely on image silhouettes to match a low dimensional parametric model to image observations. In particular, we assume a hard segmentation of the background and foreground. Instead we could extract an alpha matte and modify the approach to take into account the uncertainty in the segmentation regions. Additionally, we use silhouettes from multiple views to derive view-dependent constraints on the outer contour. To potentially reduce the need of multiple camera views, additional image cues that capture internal image structure, such as edges, shading and optical flow, may be used as well [Guan *et al.* (2009)].

In [Bălan *et al.* (2007b)] we show that we can recover the albedo of the body for a known geometry and lighting. Similarly, for known lighting and known albedo, local shape orientation can also be estimated from images. This forms the basis for an iterative optimization procedure of pose, shape and albedo by taking into account not only the silhouette contours and apparent edges, but also the internal appearance and geometric information. It is also possible to apply these methods to single uncalibrated images if additional information about the subject is known (such as their height) [Guan *et al.* (2009)].

7.2.2 Monocular Estimation and Tracking in Video Sequences

One direction of future research involves extending our methods to extract body shape from monocular image sequences by integrating information over time. Tracking is a specialization of pose estimation to video sequences that exploits a motion model that describes the possible motion of the subject between consecutive frames. For human tracking in video, estimating limb lengths, body shape parameters and body mass can be useful as these could be used in the inference of dynamics. Body shape and mass clearly affect gait and influence how external loads may alter gait. Body shape parameters could be used in visual tracking applications to identify and re-acquire subjects who come in and out of the field view.

Monocular estimation can further benefit from the inclusion of constraints on preventing interpenetration of body parts, and even more from the addition of a statistical model of expected articulated poses by embedding the allowable poses in a much lower dimensional space. As an illustrative example, any articulated pose from a monotonous walking motion can be encoded using a single parameter, the phase in the periodic walking cycle.

7.2.3 Computing Time Considerations

Currently we have not exploited graphics hardware for the projection of 3D meshes and the computation of the image matching function; such hardware will greatly reduce the computation time required. Parallel computing is another way to speed up the processing time, particularly in the case of particle-based stochastic optimization that requires the evaluation of multiple randomly generated hypotheses within one iteration. The task of gender estimation by trying both gender specific models and choosing the one that better fits the image evidence is also inherently parallel.

Finally, a computational speedup can be achieved by adopting a coarse-to-fine fitting approach, both in terms of images and the body model. A body model with a low-resolution mesh can be fit to low-resolution images much more quickly than at full resolution. The solution is then locally refined at successively finer resolutions of the body mesh and the images. In order for this to work, a mapping of the body model parameters between successive mesh resolutions also needs to be learned in advance from training data.

7.3 Privacy Considerations

Privacy concerns must be addressed for any technology that purports to "see" what someone looks like under their clothes. Unlike backscatter X-ray and millimeter wave scanners, our approach does not *see through* clothing. It does not have any information about the person's body that is not available essentially to the naked eye; in this sense it is not intrusive. The unwanted production of a likeness or facsimile of a person's unclothed shape might still be considered a violation of privacy. It is important to maintain a distinction between body shape measurements and the graphical representation of those measurements as a realistic 3D model; it is the latter which has the greatest potential for concern but this may not be needed for many vision applications. We used such a model in this dissertation only as a graphical means of conveying the measurement information obtained by our method. There are numerous computer vision applications however where 3D shape properties of the human body could prove very useful *to the computer* without ever producing a graphical representation for human viewing. Provided the vision applications are not themselves a violation of privacy, then the use of body measurements internally would likely be acceptable. In many applications, people may even find this technology beneficial in that it can provide detailed body measurements without the need to disrobe.

7.4 Open Problems

In the long term, the goal is to exceed the level of accuracy available from current commercial marker-based shape capture systems [Park and Hodgins (2006, 2008)] by using images which theoretically provide a richer source of information. We expect that, with additional cameras and improved background subtraction, the level of detailed shape recovery from video will eventually exceed that of marker-based systems.

This work was motivated by the desire to capture the shape of humans from standard digital images. The general formulation proposed in this thesis however could be extended to use other types of sensor inputs, such as depth sensors or millimeter wave scanners, and can be applied to other objects as well. For example, a 3D deformable heart model in terms of the intrinsic shape or the contracting phase can prove useful for medical imaging analysis, while for traffic monitoring, a 3D deformable vehicle model [Leotta and Mundy (2009)] may be used for tracking and recognizing cars in surveillance video.

Appendix A

Mathematical Notation

We have tried to maintain a consistent mathematical notation throughout this thesis. Here we summarize the conventions followed.

A.1 Conventions

a, b, c, ... **Scalar** are typeset in regular italic lower-case

$\vec{a}$, $\vec{b}$, $\vec{c}$, ... **Vectors** are typeset in italic lower-case and assumed column vectors: $\vec{a} = [a_1, a_2, \cdots]^\mathsf{T}$

$\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, ... **Matrices** are typeset in non-italic boldface capitals: $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$

$\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, ... **Sets** are typeset in upper-case calligraphic font: $\mathcal{A} = \{a_1, a_2, \cdots\}$ or $\mathcal{A} = [\vec{a}_1, \vec{a}_2, \cdots]$

A.2 Nomenclature

$\mathbb{R}$ - Real numbers

s - Scaling factor $s \in \mathbb{R}, s > 0$

$\vec{t}$ - Translation vector $t \in \mathbb{R}^3$

$\mathbf{R}$ - Rotation matrix $\mathbf{R} \in SO(3)$

$\vec{q}$ - Quaternion rotation

$\vec{\omega}$ - Axis-angle vector rotation

$\vec{n}$ - Surface normal

$\mathbf{I}_n$ - Identity matrix $\mathbf{I}_n = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}$

$||\mathbf{A}||_F$ - Frobenius norm of a matrix $\mathbf{A}$

$\text{diag}(\mathbf{A})$ - Matrix diagonals are manipulated using the $\text{diag}(\mathbf{A})$ operator. If $\mathbf{A}$ is a vector of length n, an $n \times n$ diagonal matrix is produced. If $\mathbf{A}$ is an $n \times n$ matrix, the diagonal is extracted into a vector of length n.

$\text{tr}(\mathbf{A})$ - The trace of square matrix $\mathbf{A}$, which is the sum of the elements on its diagonal

$\bar{\mathcal{A}}$ - Arithmetic mean

$\hat{a}$ - Estimated value for a

E - Error energy

Σ - Covariance matrix

Λ - A diagonal matrix of eigenvalues

λ_i - The i^{th} eigenvalue

A.3 SCAPE notation

i - index on example meshes in different poses

j - index on example meshes for different subjects

k - index on vertices of a triangle

v - index on vertices of a mesh

t - index on triangles of a mesh

p - index on body parts

$p[t]$ - index of body part to which triangle t belongs to

V - number of mesh vertices

T - number of mesh triangles

P - number of body parts

J - number of joints

$\mathcal{J}[p[t]]$ - list of joints for part $p[t]$

$\mathcal{X}$ - template mesh

$\{\mathcal{Y}^i, \mathcal{Y}^j\}$ - set of example meshes

$\vec{x}_{t,k}$ - location of the k^{th} vertex of triangle t for the template mesh $\mathcal{X}$

$\vec{y}^i_{t,k}$ - location of the k^{th} vertex of triangle t for a deformed mesh $\mathcal{Y}^i$

$\vec{y}_v$ - location of the vertex v for a deformed mesh $\mathcal{Y}$

$\Delta\vec{x}_{t,k}$ - edge vector $(\vec{x}_{t,k} - \vec{x}_{t,1})$

$\mathbf{A}_t$ - 3×3 - affine transformation matrix associated with triangle t

$\mathbf{R}_p$ - 3×3 - rotation matrix associated with body part p

$\mathbf{Q}_t$ - 3×3 - non-rigid pose deformation matrix associated with triangle t

$\mathbf{D}_t$ - 3×3 - body shape deformation matrix associated with triangle t

$\Delta\mathbf{R}_{p,c}$ - 3×3 - relative rotation between two adjacent body parts p and c

$\Delta\vec{\omega}_{p,c}$ - 3×1 - axis-angle representation of the relative rotation between two adjacent body parts p and c

$\mathbf{F}$ - Linear coefficients used to predict non-rigid deformations $\mathbf{Q}$ from rigid part-based rotations $\mathbf{R}$

$\vec{\theta}$ - alternative pose parameterization in terms of joint angles between adjacent body parts instead of global rotations $\mathbf{R}_p$ for each part

$\vec{d}$ - $3 \cdot 3 \cdot T \times 1$ - column vector of body shape deformations containing vectorized 3×3 body shape deformations $\mathbf{D}_t$ for all T triangles of the mesh

$\vec{\beta}$ - eigen-shape coefficients

b - index for principal components

r - number of PCA principal components for the eigen-shape model

$\mathbf{U}$ - PCA basis matrix consisting of the top r eigen-shape column vectors

$\vec{\mu}$ - mean body shape deformation

$\sigma^2_{\vec{\beta},b}$ - variance for shape coefficient $\vec{\beta}_b$

χ - gender-specific shape model: $\chi \in \{\text{male}, \text{female}, \text{neutral}\}$

Appendix B

Representations of Rigid Body Transformations

Rigid body transformations are integral to many problems addressed in this thesis including pose parameterization, shape registration and camera calibration. Here we briefly elaborate on several representations employed in this thesis.

B.1 Standard Matrix Representation

Rigid body transformations are commonly represented by a translation component and a rotation component. Translation is represented as a vector displacement in 3D: $\vec{t} = [t_x, t_y, t_z]^\mathsf{T}$. Rotation can be expressed as a 3×3 rotation matrix $\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$ subject to the constraints that it is orthonormal and its determinant is positive. More formally, the set of all valid 3D rotations is denoted by $SO(3) = \{\mathbf{R} \in \mathbb{R}^{3\times3} | \mathbf{R}\mathbf{R}^\mathsf{T} = \mathbf{R}^\mathsf{T}\mathbf{R} = \mathbf{I}_3, \det(\mathbf{R}) = +1\}$.

A point in the local coordinate system $\vec{p}_1 = [p_x, p_y, p_z]^\mathsf{T}$ is transformed by

$$\vec{p}_2 = \mathbf{R}\vec{p}_1 + \vec{t}, \tag{B.1}$$

which can also be expressed in homogeneous coordinates as

$$\begin{bmatrix} \vec{p}_2 \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \vec{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \vec{p}_1 \\ 1 \end{bmatrix}. \tag{B.2}$$

While there are 9 variables representing the rotation, the orthonormality constraint leaves only 3 free variables. Therefore, the space of all rigid transformations has 3 degrees of freedom for the translation and 3 degrees of freedom for the rotation.

B.2 Euler Angles

Euler angles (α, β, γ) can be used to represent any general 3D rotation $\mathbf{R}$ as a composition of three rotations $\mathbf{R}_x(\alpha)$, $\mathbf{R}_y(\beta)$ and $\mathbf{R}_z(\gamma)$ about the orthogonal coordinate axes:

$$\begin{aligned}\mathbf{R}(\alpha,\beta,\gamma) &= \mathbf{R}_z(\gamma)\mathbf{R}_y(\beta)\mathbf{R}_x(\alpha) \\ &= \begin{bmatrix}\cos\gamma & -\sin\gamma & 0\\ \sin\gamma & \cos\gamma & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\cos\beta & 0 & \sin\beta\\ 0 & 1 & 0\\ -\sin\beta & 0 & \cos\beta\end{bmatrix}\begin{bmatrix}1 & 0 & 0\\ 0 & \cos\alpha & -\sin\alpha\\ 0 & \sin\alpha & \cos\alpha\end{bmatrix}.\end{aligned} \tag{B.3}$$

The order of the rotations needs to be pre-specified. This representation is used for parameterizing articulated pose in terms of the relative joint angles between a child part and a parent part about the x-, y- and z- axes of the parent coordinate system. The three angles are sometimes referred to as *roll*, *pitch* and *yaw*. Euler angles are known to suffer from the *Gimbal Lock* problem which causes one degree of freedom to be lost when two of the axes become aligned.

B.3 Quaternions

Quaternions are the generalization of complex numbers to 4 dimensions. A quaternion $\vec{q}$ consists of a *real part*, a scalar w, and its *imaginary part*, a vector $\vec{\omega} = [x, y, z]^\mathsf{T}$. It is customary to denote a quaternion by the notation $\vec{q} = w + \vec{\omega}$.

A unit quaternion $\vec{q} \in \{(w, x, y, z) | w^2 + x^2 + y^2 + z^2 = 1\}$ is associated with the following rotation matrix:

$$\mathbf{R}(\vec{q}) = \mathbf{I}_3 + 2\begin{bmatrix}-y^2 - z^2 & xy - wz & xz + wy\\ xy + wz & -x^2 - z^2 & yz - wx\\ xz - wy & yz + wx & -x^2 - y^2\end{bmatrix}. \tag{B.4}$$

It follows immediately that $\vec{q}$ and $-\vec{q}$ induce the same rotation $\mathbf{R}$. The rotation of angle θ about the unit vector $\vec{a}$ is given by the unit quaternion

$$\pm\,\vec{q} = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\vec{a}\,. \tag{B.5}$$

We use quaternions in the next section for computing the alignment between two point clouds.

B.4 Axis-angle Rotations

The axis-angle representation of a rotation parameterizes the direction of the axis of rotation using a 3D unit vector $\vec{u}$ and the amount of rotation about the axis using the scalar angle θ.

Converting a rotation matrix $\mathbf{R}$ to the axis-angle representation can be done using:

$$\theta = \arccos\left(\frac{\mathrm{tr}(\mathbf{R})-1}{2}\right) \tag{B.6}$$

$$\vec{u} = \frac{1}{2\sin(\theta)}\begin{bmatrix}\mathbf{R}_{32}-\mathbf{R}_{23}\\ \mathbf{R}_{13}-\mathbf{R}_{31}\\ \mathbf{R}_{21}-\mathbf{R}_{12}\end{bmatrix}. \tag{B.7}$$

The two parameters are typically combined into a compact representation $\vec{\omega} = \theta\vec{u}$ thereby encoding the angle of rotation as the magnitude of the rotation vector $\vec{\omega}$. This compact representation is used for learning a prediction model from pose parameters to non-rigid pose deformations in Section 3.5.3.

Appendix C

Rigid Registration of 3-D Point Clouds

Consider the problem of finding the relationship between two coordinate systems by using pairs of measurements of the coordinates of a number of points in both systems. The goal is to estimate the rigid transformation and sometimes the scaling factor that best aligns two shapes, represented as a collection of 3D surface points, into a common coordinate system, minimizing the distance between the shapes. One application of this can be found in Section 3.5.2 where the aim is to estimate the optimal rotations for each body part between a template mesh and other example meshes that are in full vertex correspondence.

We first discuss the special case in which there is a one-to-one correspondence between points in the two point clouds. In this case, a closed-form solution exist that can be computed efficiently. We then describe an iterative algorithm for computing the registration parameters for two point clouds for which no such point correspondences exist and do not share the same number of vertices.

C.1 The Alignment of Corresponding Point Clouds

Various methods have been proposed in the literature that solve for the 3-D rigid body transformation that aligns two corresponding data points, most notably in closed-form [Horn (1987); Besl and McKay (1992); Challis (1995); Eggert *et al.* (1997)].

Most deal with only computing the rotational and translational components of the transformation, though the extension to account for scale changes is immediate [Horn (1987)]. Deriving the solution for the rotation component is the most challenging step. The translation and scale are easy to determine once the rotation is known. Existing approaches differ in the rotation representation used and the mathematical derivation employed. Horn (1987) and [Besl and McKay (1992)] derive solutions based on unit quaternion representation for rotations. These are appropriate for points in 2 or 3 dimensions. When the data points have more than 3 dimensions, an SVD approach based

on the cross-covariance matrix of the two point distributions is preferred [Challis (1995)]. For an extensive comparison of different methods, see [Eggert *et al.* (1997)].

C.1.1 Least-squares Formulation

We present a method for computing the least-squares solution for the optimal rigid transformation and scaling that best aligns two corresponding point sets. Let $\mathcal{X} = \{\vec{x}_i\}_{i=1}^n$ and $\mathcal{Y} = \{\vec{y}_i\}_{i=1}^n$ be two sets of $n \geq 3$ 3-D points each for which $\vec{x}_i$ and $\vec{y}_i$ are known to be in correspondence. We seek the rotation matrix $\mathbf{R}$, the translation vector $\vec{t}$ and scale factor s that closely aligns $\mathcal{X}$ to $\mathcal{Y}$. If the data were perfect, we would have $\vec{y}_i = s\mathbf{R}\vec{x}_i + \vec{t}$.

We start by defining a least squares objective function

$$\arg\min_{s,\mathbf{R},\vec{t}} \sum_{i=1}^{n} \left(\sqrt{s}\mathbf{R}\vec{x}_i + \frac{1}{\sqrt{s}}(\vec{t} - \vec{y}_i) \right)^2 . \tag{C.1}$$

This particular choice of the objective function was proposed by Horn (1987) in order to distribute the uncertainty in the scale parameter equally between the two point sets. This choice is preferred over the more typical

$$\arg\min_{s,\mathbf{R},\vec{t}} \sum_{i=1}^{n} \left(s\mathbf{R}\vec{x}_i + \vec{t} - \vec{y}_i \right)^2$$

because the latter can be shown to introduce an asymmetry in the determination of the optimal scale factor. In other words, optimal scale factors when aligning $\mathcal{X}$ to $\mathcal{Y}$ versus when aligning $\mathcal{Y}$ to $\mathcal{X}$ are not exact inverses of each other as one would expect.

C.1.2 Solving for the Translation

We begin by computing the centroids of the two point set

$$\bar{\mathcal{X}} = \frac{1}{n}\sum_{i=1}^{n} \vec{x}_i \qquad ; \qquad \bar{\mathcal{Y}} = \frac{1}{n}\sum_{i=1}^{n} \vec{y}_i \,,$$

and define new coordinates for the two point sets, $\vec{x}_i' = \vec{x}_i - \bar{\mathcal{X}}$ and $\vec{y}_i' = \vec{y}_i - \bar{\mathcal{Y}}$, whose centroids are at the origin of the coordinate system. The objective function can be rewritten in terms of the new coordinates

$$\begin{aligned} \arg\min_{s,\mathbf{R},\vec{t}} \sum_{i=1}^{n} \left(\sqrt{s}\mathbf{R}\vec{x}_i' - \frac{\vec{y}_i'}{\sqrt{s}} + \sqrt{s}\mathbf{R}\bar{\mathcal{X}} + \frac{\vec{t} - \bar{\mathcal{Y}}}{\sqrt{s}} \right)^2 \quad = \quad & \arg\min_{s,\mathbf{R},\vec{t}} \sum_{i=1}^{n} \left(\sqrt{s}\mathbf{R}\vec{x}_i' - \frac{\vec{y}_i'}{\sqrt{s}} \right)^2 + \\ & 2\left(\sqrt{s}\mathbf{R}\bar{\mathcal{X}} + \frac{\vec{t} - \bar{\mathcal{Y}}}{\sqrt{s}} \right) \sum_{i=1}^{n} \left(\sqrt{s}\mathbf{R}\vec{x}_i' - \frac{\vec{y}_i'}{\sqrt{s}} \right) + \\ & n\left(\sqrt{s}\mathbf{R}\bar{\mathcal{X}} + \frac{\vec{t} - \bar{\mathcal{Y}}}{\sqrt{s}} \right)^2 . \end{aligned} \tag{C.2}$$

Since the first term does not depend on $\vec{t}$, the second term is equal to zero by construction ($\sum_{i=1}^n \vec{x}_i' = \sum_{i=1}^n \vec{y}_i' = 0$), and the third term cannot be negative, we find that the optimal translation is given

by

$$\vec{t} = \bar{\mathcal{Y}} - s\mathbf{R}\bar{\mathcal{X}}\,. \tag{C.3}$$

Once the rotation and scale parameters are estimated, the translation is immediately obtained using the above formula.

C.1.3 Solving for the Scaling

We have reduced the problem to one of solving for the rotation and scaling of two corresponding points sets that are centered at the origin:

$$\arg\min_{s,\mathbf{R}} \sum_{i=1}^{n} \left(\sqrt{s}\mathbf{R}\vec{x}_i' - \frac{\vec{y}_i'}{\sqrt{s}} \right)^2 . \tag{C.4}$$

We expand the square to obtain

$$\begin{aligned} \sum_{i=1}^{n} \left(\sqrt{s}\mathbf{R}\vec{x}_i' - \frac{\vec{y}_i'}{\sqrt{s}} \right)^2 &= s\sum_{i=1}^{n} \vec{x}_i'^{\mathsf{T}}\mathbf{R}^{\mathsf{T}}\mathbf{R}\vec{x}_i' + \frac{1}{s}\sum_{i=1}^{n}\vec{y}_i'^2 - 2\sum_{i=1}^{n}\vec{y}_i'^{\mathsf{T}}\mathbf{R}\vec{x}_i' \\ &= s\sum_{i=1}^{n}\vec{x}_i'^2 + \frac{1}{s}\sum_{i=1}^{n}\vec{y}_i'^2 - 2\sum_{i=1}^{n}\vec{y}_i'^{\mathsf{T}}\mathbf{R}\vec{x}_i'\,. \end{aligned} \tag{C.5}$$

We used the fact that rotational matrices are orthonormal and hence $R^{\mathsf{T}}R = \mathbf{I}_3$. At this point, we note that only the first two terms depend on s and only the third term depends on $\mathbf{R}$, which means s and R can be determined independently of each other:

$$s = \arg\min_{s} \left(s\sum_{i=1}^{n}\vec{x}_i'^2 + \frac{1}{s}\sum_{i=1}^{n}\vec{y}_i'^2 \right) \tag{C.6}$$

$$\mathbf{R} = \arg\max_{\mathbf{R}} \sum_{i=1}^{n} \vec{y}_i'^{\mathsf{T}}\mathbf{R}\vec{x}_i' \tag{C.7}$$

We consider the scale parameter first. We let $A = \sum_{i=1}^{n}\vec{x}_i'^2$ and $B = \sum_{i=1}^{n}\vec{y}_i'^2$ and minimize $sA + \frac{1}{s}B$. By completing the square, we write

$$sA + \frac{1}{s}B = \left(\sqrt{s}\sqrt{A} - \frac{1}{\sqrt{s}}\sqrt{B} \right)^2 + 2\sqrt{AB}\,. \tag{C.8}$$

Since only the first term depends on s and it cannot be negative, this expression is minimized when $s = \sqrt{\frac{B}{A}}$ or

$$s = \sqrt{\frac{\sum_{i=1}^{n}(\vec{y}_i - \bar{\mathcal{Y}})^2}{\sum_{i=1}^{n}(\vec{x}_i - \bar{\mathcal{X}})^2}}\,. \tag{C.9}$$

The main observation here is that the optimal scale factor can be computed independently of the translation and rotation. Moreover, the determination of the rotation in the next step is not affected by the choice of the scale parameter.

C.1.4 Solving for the Rotation

The derivation of a closed form solution for determining the rotation is by far the most difficult step. There are many representations possible for rotations. 3×3 rotation matrices are the most commonly used, but enforcing the non-linear orthonormality constraint in closed-form requires dealing with special cases. The unit quaternion representation however eliminates the need for handling special cases because it is simpler to enforce quaternion unit magnitude than it is to ensure that a matrix is orthonormal. Quaternions can easily be converted to rotation matrices.

We first provide a solution that computes the rotation matrix from the Singular Value Decomposition of the cross-correlation matrix of the two point sets. We then outline the preferred alternative procedure that uses the quaternion representation.

The problem we are solving is

$$\underset{\mathbf{R}\in SO(3)}{\arg\max} \sum_{i=1}^{n} \vec{y}_i'^{\mathsf{T}} \mathbf{R} \vec{x}_i' \,. \tag{C.10}$$

SVD Method

Since $\vec{y}_i'^{\mathsf{T}} \mathbf{R} \vec{x}_i'$ is a scalar, it is trivially identical to its trace, where the trace of a matrix is defined as the sum of the elements on its main diagonal. The objective function can be rewritten as

$$\begin{aligned} \sum_{i=1}^{n} \vec{y}_i'^{\mathsf{T}} \mathbf{R} \vec{x}_i' &= \sum_{i=1}^{n} \mathrm{tr}\left(\vec{y}_i'^{\mathsf{T}} \mathbf{R} \vec{x}_i'\right) \\ &= \sum_{i=1}^{n} \mathrm{tr}\left(\mathbf{R} \vec{x}_i' \vec{y}_i'^{\mathsf{T}}\right), \qquad \text{using } \mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA}) \\ &= \mathrm{tr}\left(\mathbf{R} \sum_{i=1}^{n} \vec{x}_i' \vec{y}_i'^{\mathsf{T}}\right) \end{aligned} \tag{C.11}$$

The matrix $C = \sum_{i=1}^{n} \vec{x}_i' \vec{y}_i'^{\mathsf{T}}$ is called the un-normalized correlation matrix (sometimes also called the cross-dispersion matrix or the cross-covariance matrix) and can be decomposed using Singular Value Decomposition (SVD) into

$$\mathbf{C} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathsf{T}} \tag{C.12}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices and $\mathbf{S}$ is diagonal.

By combining the last three equations and using commutativity for trace we obtain

$$\begin{aligned} \mathrm{tr}\,(\mathbf{RC}) &= \mathrm{tr}\left(\mathbf{R}\mathbf{U}\mathbf{S}\mathbf{V}^{\mathsf{T}}\right) \\ &= \mathrm{tr}\left((\mathbf{V}^{\mathsf{T}}\mathbf{R}\mathbf{U})\mathbf{S}\right) \end{aligned} \tag{C.13}$$

Because $\mathbf{S}$ is diagonal with singular values that are almost always positive, we seek

$$\underset{\mathbf{R}\in SO(3)}{\arg\max} \left(\mathrm{tr}\left(\mathbf{V}^{\mathsf{T}}\mathbf{R}\mathbf{U}\right)\right) \,.$$

We note that $\mathbf{V}^\mathsf{T}\mathbf{R}\mathbf{U}$ is a product of orthonormal matrices and hence orthonormal itself and the identity matrix is the only orthonormal matrix with maximal trace. As such

$$\mathbf{R} = \mathbf{V}\mathbf{U}^\mathsf{T} \tag{C.14}$$

In rare circumstances, the determinant of $\mathbf{V}\mathbf{U}^\mathsf{T}$ is not +1 but rather -1, which corresponds to a reflection instead of a rotation. To account for this special case, the following modification for obtaining the optimal rotation is sufficient:

$$\mathbf{R} = \mathbf{V}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(\mathbf{V}\mathbf{U}^\mathsf{T}) \end{bmatrix}\mathbf{U}^\mathsf{T}. \tag{C.15}$$

Unit Quaternion Method

We present the unit quaternion method due to [Horn (1987); Besl and McKay (1992)] without the derivation.

First we compute the 3×3 correlation matrix

$$C = \sum_{i=1}^{n} \vec{x}_i' \vec{y}_i'^\mathsf{T}. \tag{C.16}$$

Second, we build the column vector

$$\Delta = \begin{bmatrix} \mathbf{C}_{23} - \mathbf{C}_{32} \\ \mathbf{C}_{31} - \mathbf{C}_{13} \\ \mathbf{C}_{12} - \mathbf{C}_{21} \end{bmatrix}. \tag{C.17}$$

Third, we form the symmetric 4×4 matrix

$$\mathbf{Q} = \begin{bmatrix} \mathrm{tr}(\mathbf{C}) & \Delta^\mathsf{T} \\ \Delta & \mathbf{C} + \mathbf{C}^\mathsf{T} - \mathrm{tr}(\mathbf{C})\mathbf{I}_3 \end{bmatrix}. \tag{C.18}$$

Forth, the unit eigenvector $\vec{q}$ corresponding to the largest positive eigenvalue of the matrix $\mathbf{Q}$ is the unit quaternion corresponding to the optimal rotation $\mathbf{R}(\vec{q})$ as given by Eq. B.4.

C.1.5 Algorithm

1. The optimal rotation represented as a unit quaternion was shown to be the eigenvector associated with the largest positive eigenvalue of a symmetric 4×4 matrix derived from the correlation matrix of the two point sets. The elements of this matrix are simple combinations of sums of products of coordinates of the points.

2. The best scale is equal to the ratio of the root-mean-square deviations of the coordinates in the two systems from their respective centroids.

3. The optimal translation is found to be the difference between the centroid of the coordinates in one system and the rotated and scaled centroid of the coordinates in the other system.

C.2 The Iterative Closest Point Algorithm

We now describe an iterative algorithm for computing the registration parameters for two point clouds for which no such point correspondences exist. The standard approach for rigidly matching two point clouds is to use the Iterative Closest Point (ICP) algorithm. The algorithm is very simple: it iteratively updates the transformation between the two point sets as they move closer together.

1. Establish point correspondences between the two point sets using the closest-point criterion;
2. Estimate the rigid transformation using the closed-form solution for aligning two point sets with known correspondences (section C.1);
3. Transform the points using the estimated parameters;
4. Iterate until some criterion for stopping is met.

Appendix D

Large Scale Principal Component Analysis

Principal component analysis (PCA) is a mathematical method commonly used for dimensionality reduction that uses linear projection to relate a set of high dimensional vectors to a set of lower dimensional vectors. It does so by finding a lower-dimensional linear subspace which, for a set size, accounts for most of the variability in the input data.

The PCA derivation involves computing the eigenvalue decomposition of the data covariance matrix. This becomes problematic when the dimensionality of the input data is very large and the covariance matrix cannot fit into computer memory. The general accepted solution has been to express the PCA basis in terms of the singular value decomposition (SVD) of the data matrix, which has much lower memory requirements than using the covariance matrix when the number of data points is very small relative to the number of data dimensions. For cases when the data matrix itself is too large, we propose using a method for computing a reduced SVD in an incremental fashion in order to bypass computer memory limitations.

D.1 Principal Component Analysis

Principal component analysis (PCA) is a mathematical procedure often used for extracting meaningful information from overly redundant multi-dimensional data. PCA can be thought of as a pseudo-rotation of the standard axes for a set of data points around their mean in order to align them with the major axes of variation in the data set called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible after all correlation with the previous principal components has been subtracted out from the data. In effect, PCA un-correlates the data. While the top principal components accumulate most of the variance in the data, the bottom ones may be left with noise or be highly correlated with the top ones. By dropping the

bottom principal components, we can reduce the dimensionality of the input data with minimal loss of information.

D.1.1 PCA Derivation using the Eigen-decomposition of the Covariance Data Matrix

Let $\mathbf{X}_{d\times n} = [\vec{x}_1, \cdots, \vec{x}_n]$ be the data matrix whose columns represent n data points embedded in a d-dimensional space. PCA starts from the basic assumption that the data comes from a multivariate Gaussian distribution in order to find independent axes of variation in the data, and places a strong emphasis on the statistical importance of the mean and covariance of the data. Without loss of generality, the mean is factored out before further analyzing the covariance. A zero-mean data matrix $\mathbf{A}_{d\times n} = [\vec{a}_1, \cdots, \vec{a}_n]$ is obtained by subtracting the empirical mean $\vec{\mu}_{\mathbf{X}} = \sum_{i=1}^{n} \vec{x}_i$ from the original data:

$$\begin{aligned} \mathbf{A} &= \mathbf{X} - \vec{\mu}_{\mathbf{X}} \vec{1}_{1\times n} \\ \vec{\mu}_{\mathbf{X}} &= \sum_{i=1}^{n} \vec{x}_i \\ \vec{\mu}_{\mathbf{A}} &= \sum_{i=1}^{n} \vec{a}_i = \vec{0}_{d\times 1}\,. \end{aligned} \tag{D.1}$$

PCA seeks a new orthonormal basis $\mathbf{U}^{\mathsf{T}}_{d\times d}$ for $\mathbf{A}$ such that the covariance between different dimensions of the transformed data points, $\mathrm{cov}(\mathbf{U}^{\mathsf{T}}\mathbf{A})$, is diagonal [Shlens (2009)]. In other words, it wants to remove the correlated redundancy between different directions. The columns of $\mathbf{U}$ are the *principal components* of $\mathbf{A}$ and are the directions of maximum variance in the data set.

If $\mathbf{B}_{d\times n} = \mathbf{U}^{\mathsf{T}}\mathbf{A}$ is the transformed representation of the data set, then its covariance matrix is given by

$$\begin{aligned} \mathrm{cov}(\mathbf{B}) &= \frac{1}{n-1}\mathbf{B}\mathbf{B}^{\mathsf{T}} \\ &= \frac{1}{n-1}\left(\mathbf{U}^{\mathsf{T}}\mathbf{A}\right)\left(\mathbf{U}^{\mathsf{T}}\mathbf{A}\right)^{\mathsf{T}} \\ &= \mathbf{U}^{\mathsf{T}}\left(\frac{1}{n-1}\mathbf{A}\mathbf{A}^{\mathsf{T}}\right)\mathbf{U} \\ &= \mathbf{U}^{\mathsf{T}}\,\mathrm{cov}(\mathbf{A})\mathbf{U}\,. \end{aligned} \tag{D.2}$$

Using the fact that $\mathbf{U}$ is an orthonormal basis and hence $\mathbf{U}^{\mathsf{T}} = \mathbf{U}^{-1}$, we obtain

$$\mathbf{U}\,\mathrm{cov}(\mathbf{B}) = \mathrm{cov}(\mathbf{A})\mathbf{U}\,. \tag{D.3}$$

We express the fact that $\mathrm{cov}(\mathbf{B})$ needs to be diagonal explicitly

$$\mathrm{cov}(\mathbf{B}) = \mathbf{\Lambda}_{d\times d} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_d \end{bmatrix} \tag{D.4}$$

and further decompose $\mathbf{U}$ into its d-dimensional column vectors: $\mathbf{U} = [\vec{u}_1, \cdots, \vec{u}_d]$. Equation D.3 then becomes

$$\begin{aligned} [\vec{u}_1, \cdots, \vec{u}_d] \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_d \end{bmatrix} &= \text{cov}(\mathbf{A}) [\vec{u}_1, \cdots, \vec{u}_d] \\ [\lambda_1 \vec{u}_1, \cdots, \lambda_d \vec{u}_d] &= \text{cov}(\mathbf{A}) [\vec{u}_1, \cdots, \vec{u}_d] \ . \end{aligned} \tag{D.5}$$

Since $\lambda_i \vec{u}_i = \text{cov}(\mathbf{A})\vec{u}_i$ for every column i, it follows that each $\vec{u}_i$ is an eigenvector of the covariance matrix of $\mathbf{A}$ and λ_i the corresponding eigenvalue. This means one can obtain the new set of basis vectors for $\mathbf{A}$ by simply performing an eigen-decomposition of the covariance matrix $\text{cov}(\mathbf{A}) = \frac{1}{n-1}\mathbf{A}\mathbf{A}^\mathsf{T}$.

An eigen-decomposition is possible for any given square symmetric positive semi-definite matrix like $\text{cov}(\mathbf{A})$ and can be found using existing computer-based implementations[1]. It returns an orthonormal matrix $\mathbf{U}_{d\times d}$ as a row of column eigenvectors, and a diagonal matrix $\mathbf{\Lambda}_{d\times d}$ of non-negative eigenvalues, arranged in decreasing order along the diagonal, such that

$$\text{cov}(\mathbf{A}) = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\mathsf{T} \ . \tag{D.6}$$

Moreover, since $\mathbf{\Lambda}$ is $\text{cov}(\mathbf{U}^\mathsf{T}\mathbf{A})$, the diagonal covariance matrix of the data expressed using a new orthonormal basis (Equation D.4), each of the eigenvalues λ_i specifies the variance in the data set along the direction of the principal component $\vec{u}_i$. The columns of the basis $\mathbf{U}$ are arranged in the order of decreasing variance of the data they capture.

Given a point $\vec{x}$ in the original space, we can express it in the transformed space using

$$\vec{b} = \mathbf{U}^\mathsf{T} (\vec{x} - \vec{\mu}_\mathbf{X}) \ , \tag{D.7}$$

and map it back using

$$\hat{\vec{x}} = \mathbf{U}\vec{b} + \vec{\mu}_\mathbf{X} \ . \tag{D.8}$$

D.1.2 Alternative Solution using Singular Value Decomposition

The problem with the covariance eigen-decomposition approach however for computing PCA is that it requires to explicitly compute and store the covariance data matrix in the computer main memory. The covariance matrix is $d \times d$ in size and typically non-sparse. To provide some context, recall that in Chapter 3 we use PCA to find a reduced-dimension subspace that captures the variance in how body shapes deform between individuals. There, the data set contains $n = 2,000$ individuals and the shape deformations are encoded using $d = 225,000$ dimensions. The covariance matrix would contain over 5×10^{10}, 8 byte, elements and would necessitate 377 GB just to store. Clearly,

[1] The eigen-decomposition of a matrix can be computed, for example, using the Matlab$^{\text{TM}}$ function: $[\mathbf{U}, \mathbf{\Lambda}] = eig(cov(\mathbf{A}))$, followed by a re-arrangement of the column vectors of $\mathbf{U}$ such that the diagonal elements for $\mathbf{\Lambda}$ appear in non-increasing order.

this is currently computationally impractical. In contrast, the data matrix itself requires only 3.35 GB of memory storage.

An alternative approach exists that uses the Singular Value Decomposition (SVD) of the matrix $\mathbf{A}$ directly. SVD is a mathematical method for decomposing any arbitrary matrix $\mathbf{A}$ into a product of three matrices of the form

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}, \tag{D.9}$$

where $\mathbf{U}_{d\times d}$ and $\mathbf{V}_{n\times n}$ are orthonormal matrices and $\mathbf{\Sigma}_{d\times n}$ is a diagonal (albeit non-square) matrix whose elements s_i along the diagonal, called *singular values*, are arranged in decreasing order. Such decomposition can be computed for a given data matrix $\mathbf{A}$ using existing computer-based implementations[2].

When this decomposition is integrated into the computation of the covariance matrix of $\mathbf{A}$, we find that

$$\begin{aligned}
\mathrm{cov}(\mathbf{A}) &= \frac{1}{n-1}\mathbf{A}\mathbf{A}^{\mathsf{T}} \\
&= \frac{1}{n-1}\left(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}\right)\left(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}\right)^{\mathsf{T}} \\
&= \frac{1}{n-1}\mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}}\mathbf{V}\mathbf{\Sigma}^{\mathsf{T}}\mathbf{U}^{\mathsf{T}}, \qquad (\mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}) \\
&= \frac{1}{n-1}\mathbf{U}\mathbf{\Sigma}\mathbf{\Sigma}^{\mathsf{T}}\mathbf{U}^{\mathsf{T}} \\
&= \mathbf{U}\left(\frac{1}{n-1}\mathbf{\Sigma}^2\right)\mathbf{U}^{\mathsf{T}} \\
&= \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathsf{T}}, \qquad \mathbf{\Lambda} = \frac{1}{n-1}\mathbf{\Sigma}^2 .
\end{aligned} \tag{D.10}$$

It becomes pretty obvious when comparing Equation D.10 with Equation D.6 that the result of SVD on the data matrix $\mathbf{A}$ can be used to infer the eigen-decomposition of the covariance data matrix without actually computing $\mathrm{cov}(\mathbf{A})$. The SVD method applied to $\mathbf{A}$ directly returns the PCA basis $\mathbf{U}$, while the variance and standard deviation along the direction of the principal component $\vec{u}_i$ are given by λ_i and σ_i, respectively:

$$\lambda_i = \frac{s_i^2}{n-1} \quad , \quad \sigma_i = \sqrt{\frac{s_i^2}{n-1}} . \tag{D.11}$$

Note that $\mathbf{\Sigma}_{d\times n}$ is a non-square diagonal matrix which means there are at most $\min(d, n)$ non-zero singular values along its diagonal. In particular, when $n < d$, the length of its diagonal is n, which means only the first n columns of $\mathbf{U}$ are relevant for the SVD decomposition. The following reduced, but equivalent, form is called an economy (or thin) SVD decomposition[3]:

$$\mathbf{A}_{d\times n} = \mathbf{U}_{d\times n}\mathbf{\Sigma}_{n\times n}\mathbf{V}^{\mathsf{T}}_{n\times n} = [\vec{u}_1, \cdots, \vec{u}_n]\,\mathrm{diag}\,(s_1, \cdots, s_n)\,[\vec{v}_1, \cdots, \vec{v}_n]^{\mathsf{T}} . \tag{D.12}$$

[2] The singular value decomposition of a matrix can be computed, for example, using the Matlab™ function: $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = svd(\mathbf{A})$. An economy version is also available that computes only the basis vectors corresponding to the largest singular values and skips the ones whose corresponding singular values are guaranteed to be zero.

[3] A similar economy decomposition exists for the case when $d < n$, which uses only the first d columns of $\mathbf{V}$.

The reduced SVD form is significant because it is more economical for storage and faster to compute when $n \ll d$.

D.1.3 Dimensionality Reduction

For many practical applications, the sample points in the data set may span a much smaller dimensional sub-space than the one span by the $\mathbf{U}_{d\times \min(d,n)}$ basis. Assuming that the data truly comes from an r-dimensional linear manifold, the entire variance would concentrate in the subspace spanned by the first r principal components $\mathbf{U}_{d\times r} = [\vec{u}_1, \cdots, \vec{u}_r]$, ordered in decreasing order of the variance they capture. Ideally no variance is left in the complement space ($\lambda_i = s_i = 0$ for $i \in \{d-r+1, \cdots, d\}$), although in practice the sample data set is often corrupted by noise and no singular value along the diagonal of Σ is exactly zero ($r \leq \text{rank}(\mathbf{A}) \leq \min(d, n)$). Setting all the singular values in Σ except the r largest ones to zero results in the best r-ranked matrix approximation under the Frobenius norm difference measure:

$$\begin{aligned}\mathbf{A}_{d\times n} \xrightarrow{\text{SVD}_\text{r}} \mathbf{U} &\begin{bmatrix} s_1 & & & \\ & \ddots & & \\ & & s_r & \\ & & & 0 \end{bmatrix} \mathbf{V}^\mathsf{T} \\ &= [\vec{u}_1, \cdots, \vec{u}_r] \,\text{diag}\,(s_1, \cdots, s_r)\, [\vec{v}_1, \cdots, \vec{v}_r]^\mathsf{T} \\ &= \mathbf{U}_{d\times r} \mathbf{\Sigma}_{r\times r} (\mathbf{V}_{n\times r})^\mathsf{T}.\end{aligned} \tag{D.13}$$

Note that this representation admits a compact form consisting of the first r columns of $\mathbf{U}$ and $\mathbf{V}$ and only the first r diagonal elements of $\mathbf{\Sigma}$. The eigenvectors (principal components) $\vec{u}_i$ that correspond to the largest eigenvalues can be used to reconstruct a large portion of the variance of the original data. The original space can been reduced with minimum data loss to a space spanned by a few eigenvectors.

A d-dimensional data point $\vec{a}$ in the original space can be expressed with a few PCA coefficients $\vec{b}_{r\times 1}$ by projecting onto the lower dimensional space spanned by $\mathbf{U}_{d\times r}$:

$$\vec{b} = (\mathbf{U}_{d\times r})^\mathsf{T} \vec{a} \tag{D.14}$$

from which we can reconstruct the best linear approximation for $\vec{a}$ using

$$\hat{\vec{a}} = \mathbf{U}_{d\times r} \vec{b}. \tag{D.15}$$

D.2 Incremental Singular Value Decomposition

Standard SVD implementations are highly optimized to work in batch form, taking as input a full matrix containing the entire data set, and produce an exact answer. For many practical cases, the size of the data set may prove too large for the computer main memory. Moreover, for dimensionality

reduction problems, only an r-rank approximation of the SVD factorization is actually needed, with $r \ll \min(d, n)$.

We describe an r-ranked approximation method for computing the singular value decomposition of very large matrices in an incremental fashion, which can be obtained much quicker and with a smaller memory footprint than the exact batch SVD methods. The procedure relies on a subroutine that updates an SVD factorization of some data when additional data is taken into consideration [Brand (2002)]. Essentially, given a large dataset of points, we compute the exact SVD for the first r data points and then use the SVD-update subroutine to sequentially include small chunks of the remaining data points.

D.2.1 Updating an SVD

Given a portion of the data for which we already obtained an r-ranked SVD, we are interested in updating it given an additional chunk of data points. We assume the data has zero-mean or that the mean can be estimated *a priori* and subtracted from the incoming data points.

Let $\mathbf{A}_1 \in \mathbb{R}^{d \times n_1}$ be the r-ranked SVD approximation for a portion of n_1 data points

$$\mathbf{A}_1 = \mathbf{U}_{d \times r} \mathbf{\Sigma}_{r \times r} (\mathbf{V}_{n_1 \times r})^{\mathsf{T}} , \tag{D.16}$$

and let $\mathbf{A}_2 \in \mathbb{R}^{d \times n_2}$ be a new chunk of n_2 data points which has been mean-centered. Since $\mathbf{A}_2$ may not be completely spanned by $\mathbf{U}$, it can be decomposed into a component that lies within, and a component orthogonal to, the subspace spanned by $\mathbf{U}$. Specifically, let $\mathbf{B}_{r \times n_2}$ be the projection coefficients of $\mathbf{A}_2$ onto the orthogonal basis $\mathbf{U}$

$$\mathbf{B} = \mathbf{U}^{\mathsf{T}} \mathbf{A}_2 , \tag{D.17}$$

and let the residual $\mathbf{H}_{d \times n_2}$

$$\mathbf{H} = \mathbf{A}_2 - \mathbf{U}\mathbf{B} = (\mathbf{I} - \mathbf{U}\mathbf{U}^{\mathsf{T}})\mathbf{A}_2 \tag{D.18}$$

be the component of $\mathbf{A}_2$ orthogonal to the subspace spanned by $\mathbf{U}$. We define an orthonormal basis for $\mathbf{H}$ by applying a standard, economy size, QR-decomposition procedure[4]

$$\mathbf{H}_{d \times n_2} \xrightarrow{\text{QR}} \mathbf{J}_{d \times n_2} \mathbf{K}_{n_2 \times n_2} , \tag{D.19}$$

where $\mathbf{J}$ is an orthogonal basis of $\mathbf{H}$ and $\mathbf{K}$ are the projection coefficients of $\mathbf{A}_2$ onto $\mathbf{J}$:

$$\mathbf{K} = \mathbf{J}^{\mathsf{T}} \mathbf{H} . \tag{D.20}$$

By construction, for Equations D.18 and D.19 we have

$$\mathbf{A}_2 = \mathbf{U}\mathbf{B} + \mathbf{J}\mathbf{K} = \begin{bmatrix} \mathbf{U} & \mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{B} \\ \mathbf{K} \end{bmatrix} . \tag{D.21}$$

[4] The economy size QR-decomposition of a matrix can be computed, for example, using the Matlab$^{\text{TM}}$ function: $[\mathbf{J}, \mathbf{K}] = qr(\mathbf{H}, 0)$.

Similarly, from Equation D.16

$$\mathbf{A}_1 = \mathbf{U}(\mathbf{\Sigma}\mathbf{V}^\mathsf{T}) = \begin{bmatrix}\mathbf{U} & \mathbf{J}\end{bmatrix} \begin{bmatrix}\mathbf{\Sigma}\mathbf{V}^\mathsf{T} \\ \mathbf{0}\end{bmatrix}. \tag{D.22}$$

The last two expressions can be combined into

$$\begin{aligned}\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_2\end{bmatrix} &= \begin{bmatrix}\mathbf{U} & \mathbf{J}\end{bmatrix} \begin{bmatrix}\mathbf{\Sigma}\mathbf{V}^\mathsf{T} & \mathbf{B} \\ \mathbf{0} & \mathbf{K}\end{bmatrix} \\ &= \begin{bmatrix}\mathbf{U} & \mathbf{J}\end{bmatrix} \begin{bmatrix}\mathbf{\Sigma} & \mathbf{B} \\ \mathbf{0} & \mathbf{K}\end{bmatrix} \begin{bmatrix}\mathbf{V}^\mathsf{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}\end{bmatrix}.\end{aligned} \tag{D.23}$$

After diagonalizing the matrix in the middle using standard batch SVD

$$\begin{bmatrix}\mathbf{\Sigma} & \mathbf{B} \\ \mathbf{0} & \mathbf{K}\end{bmatrix} \xrightarrow{\text{SVD}} \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^\mathsf{T}, \tag{D.24}$$

we obtain

$$\begin{aligned}\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_2\end{bmatrix} &= \begin{bmatrix}\mathbf{U} & \mathbf{J}\end{bmatrix} \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^\mathsf{T} \begin{bmatrix}\mathbf{V}^\mathsf{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}\end{bmatrix} \\ &= \underbrace{\left(\begin{bmatrix}\mathbf{U} & \mathbf{J}\end{bmatrix} \mathbf{U}_1\right)}_{\mathbf{U}_2} \mathbf{\Sigma}_1 \underbrace{\left(\begin{bmatrix}\mathbf{V} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}\end{bmatrix} \mathbf{V}_1\right)^\mathsf{T}}_{\mathbf{V}_2}.\end{aligned} \tag{D.25}$$

Note that $\mathbf{U}_2 \in \mathbb{R}^{d\times(r+n_2)}$ and $\mathbf{V}_2 \in \mathbb{R}^{r\times(r+n_2)}$ are products of orthonormal matrices and hence orthonormal themselves, while $\mathbf{\Sigma}_1$ is a proper SVD diagonal matrix by construction. As such, the updated SVD is given by

$$\begin{bmatrix}\mathbf{A}_1 & \mathbf{A}_2\end{bmatrix} \xrightarrow{\text{SVD}} \mathbf{U}_2\mathbf{\Sigma}_1\mathbf{V}_2^\mathsf{T}. \tag{D.26}$$

As this decomposition has $r + n_2$ principal components, we obtain the best r-rank SVD by keeping the top r most significant principal components and truncating the remaining n_2. This is equivalent to replacing the exact SVD solution in Equation D.24 with an r-rank SVD$_r$ approximation (Equation D.13).

D.3 Incremental Principal Component Analysis

Incremental PCA is a direct extension of standard PCA where the input data needs to be partitioned into smaller chunks in order to compute a reduced r-ranked SVD of the data matrix in an incremental fashion as described in the previous section. Also necessary for large datasets is to estimate the mean of the data incrementally as well, which is straightforward using a sum accumulator and a count accumulator. In the end we obtain r principal components, the sample mean, and associated variance and standard deviation for each principal component.

Bibliography

Agarwal, A. and Triggs, B. (2006). Recovering 3D human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **28**(1), 44–58.

Allen, B., Curless, B., and Popović, Z. (2002). Articulated body deformation from range scan data. *ACM Transactions on Graphics (TOG), SIGGRAPH*, **21**(3), 612–619.

Allen, B., Curless, B., and Popović, Z. (2003). The space of human body shapes: reconstruction and parameterization from range scans. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, pages 587–594, New York, NY, USA. ACM.

Allen, B., Curless, B., Popović, Z., and Hertzmann, A. (2006). Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA*, pages 147–156.

Anguelov, D. (2005). *Learning Models of Shape from 3D Range Data*. Ph.D. thesis, Stanford University.

Anguelov, D., Koller, D., Srinivasan, P., Thrun, S., Pang, H.-C., and Davis, J. (2005a). The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces. In *Neural Information Processing Systems, NIPS*, volume 17, pages 33–40, Vancouver, Canada.

Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., and Davis, J. (2005b). SCAPE: Shape completion and animation of people. *ACM Transactions on Graphics (TOG), SIGGRAPH*, **24**(3), 408–416.

Aubel, A. and Thalmann, D. (2001). Interactive modeling of the human musculature. In *Conference on Computer Animation, CA*, pages 167–255.

Bălan, A. O. and Black, M. J. (2008). The naked truth: Estimating body shape under clothing. In *European Conference on Computer Vision, ECCV*, volume 5303, pages 15–29.

Bălan, A. O., Sigal, L., and Black, M. J. (2005). A quantitative evaluation of video-based 3d person tracking. In *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, pages 349–356.

Bălan, A. O., Sigal, L., Black, M. J., Davis, J. E., and Haussecker, H. W. (2007a). Detailed human shape and pose from images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*.

Bălan, A. O., Black, M. J., Sigal, L., and Haussecker, H. W. (2007b). Shining a light on human pose: On shadows, shading and the estimation of pose and shape. In *IEEE International Conference on Computer Vision, ICCV*.

Bandouch, J., Engstler, F., and Beetz, M. (2008). Accurate human motion capture using an ergonomics-based anthropometric human model. In *International conference on Articulated Motion and Deformable Objects*, pages 248–258. Springer-Verlag.

Barrón, C. and Kakadiaris, I. A. (2001). Estimating anthropometry and pose from a single uncalibrated image. *Computer Vision and Image Understanding, CVIU*, **81**(3), 269–284.

Barrón, C. and Kakadiaris, I. A. (2003). On the improvement of anthropometry and pose estimation from a single uncalibrated image. *Machine Vision and Applications, MVA*, **14**(4), 229–236.

BenAbdelkader, C. and Davis, L. (2006). Estimation of anthropomeasures from a single calibrated camera. In *International Conference on Automatic Face and Gesture Recognition, FGR*, pages 499–504, Washington, DC, USA. IEEE Computer Society.

BenAbdelkader, C. and Yacoob, Y. (2008). Statistical estimation of human anthropometry from a single uncalibrated image. In *Computational Forensics*. Springer Press.

Besl, P. J. and McKay, N. D. (1992). A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **14**(2), 239–256.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Black, M. J. and Anandan, P. (1996). The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding, CVIU*, **63**(1), 75–104.

Blanz, V. and Vetter, T. (1999). A morphable model for the synthesis of 3D faces. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, pages 187–194.

Bouguet, J.-Y. (2000). Camera calibration toolbox for Matlab. Technical report, CalTech.

Bradley, D., Popa, T., Sheffer, A., Heidrich, W., and Boubekeur, T. (2008). Markerless garment capture. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, pages 1–9, New York, NY, USA. ACM.

Brand, M. (2002). Incremental singular value decomposition of uncertain data with missing values. In *European Conference on Computer Vision, ECCV*, pages 707–720.

Bregler, C., Malik, J., and Pullen, K. (2004). Twist based acquisition and tracking of animal and human kinematics. *International Journal of Computer Vision, IJCV*, **56**(3), 179–194.

Bruckstein, A. M., Holt, R. J., Jean, Y. D., and Netravali, A. N. (2001). On the use of shadows in stance recovery. *International Journal of Imaging Systems and Technology, IJIST*, **11**(5), 315–330.

Challis, J. (1995). A procedure for determining rigid body transformation parameters. *Journal of Biomechanics*, **28**(6), 733–737.

Cheung, G., Kanade, T., Bouquet, J., and Holler, M. (2000). A real time system for robust 3D voxel reconstruction of human motions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 714–720.

Cheung, K. M., Baker, S., and Kanade, T. (2003a). Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 1, pages 77–84.

Cheung, K. M., Baker, S., and Kanade, T. (2003b). Visual hull alignment and refinement across time: A 3D reconstruction algorithm combining shape-from-silhouette with stereo. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 375–382.

Cheung, K.-M. G., Baker, S., and Kanade, T. (2005). Shape-from-silhouette across time part II: Applications to human modeling and markerless motion tracking. *International Journal of Computer Vision, IJCV*, **63**(3), 225–245.

Chu, C.-W., Jenkins, O. C., and Matarić, M. J. (2003). Markerless kinematic model and motion capture from volume sequences. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume II, pages 475–482.

Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J. (1995). Active shape models—their training and application. *Computer Vision and Image Understanding, CVIU*, **61**(1), 38–59.

Corazza, S., Gambaretto, E., Mündermann, L., and Andriacchi, T. P. (2008). Automatic generation of a subject specific model for accurate markerless motion capture and biomechanical applications. *IEEE Transactions on Biomedical Engineering*.

Davis, J. and Gao, H. (2004). Gender recognition from walking movements using adaptive three-mode PCA. In *IEEE Workshop on Articulated and Nonrigid Motion, CVPRW*, volume 1, pages 9–16.

de Aguiar, E., Theobalt, C., Stoll, C., and Seidel, H.-P. (2007). Marker-less deformable mesh tracking for human shape and motion capture. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2502–2509.

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

de la Gorce, M., Paragios, N., and Fleet, D. (2008). Model-based hand tracking with texture, shading and self-occlusions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*.

Debevec, P. (1998). Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, pages 189–198.

Deutscher, J. and Reid, I. (2005). Articulated body motion capture by stochastic search. *International Journal of Computer Vision, IJCV*, **61**(2), 185–205.

Deutscher, J., Isard, M., and MacCormick, J. (2002). Automatic camera calibration from a single Manhattan image. In *European Conference on Computer Vision, ECCV*, pages 175–205.

Dong, F., Clapworthy, G. J., Krokos, M. A., and Yao, J. (2002). An anatomy-based approach to human muscle modeling and deformation. *IEEE Transactions on Visualization and Computer Graphics*, **8**(2), 154–170.

Eggert, D. W., Lorusso, A., and Fisher, R. B. (1997). Estimating 3-D rigid body transformations: a comparison of four major algorithms. *Machine Vision and Applications, MVA*, **9**(5-6), 272–290.

Epstein, R., Yuille, A. L., and Belhumeur, P. N. (1996). Learning object representations from lighting variations. In *ECCV International Workshop on Object Representation in Computer Vision II*, volume 1144 of *Lecture Notes in Computer Science*, pages 179–199. Springer-Verlag.

Franco, J.-S., Lapierre, M., and Boyer, E. (2006). Visual shapes of silhouette sets. In *International Symposium on 3D Data Processing, Visualization and Transmission, 3DPVT*.

Furukawa, Y. and Ponce, J. (2009). Carved visual hulls for image-based modeling. *International Journal of Computer Vision, IJCV*, **81**(1), 53–67.

Gall, J., Stoll, C., de Aguiar, E., Theobalt, C., Rosenhahn, B., and Seidel, H.-P. (2009). Motion capture using joint skeleton tracking and surface estimation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*.

Gavrila, D. M. and Davis, L. S. (1996). 3-D model-based tracking of humans in action: A multi-view approach. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 73–80.

Grauman, K., Shakhnarovich, G., and Darrell, T. (2003a). A Bayesian approach to image-based visual hull reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 1, pages 187–194.

Grauman, K., Shakhnarovich, G., and Darrell, T. (2003b). Inferring 3D structure with a statistical image-based shape model. In *IEEE International Conference on Computer Vision, ICCV*, pages 641–648.

Grest, D., Woetzel, J., and Koch, R. (2005). Nonlinear body pose estimation from depth images. *Pattern Recognition*, **3663**, 285–292.

Guan, P., Weiss, A., Bălan, A. O., and Black, M. J. (2009). Estimating human shape and pose from a single image. In *IEEE International Conference on Computer Vision, ICCV*.

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

Hasler, N., Rosenhahn, B., Thormählen, T., Wand, M., Gall, J., and Seidel, H.-P. (2009a). Markerless motion capture with unsynchronized moving cameras. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*.

Hasler, N., Stoll, C., Sunkel, M., Rosenhahn, B., and Seidel, H.-P. (2009b). A statistical model of human pose and body shape. *Eurographics, Computer Graphics Forum, CGF*, **2**(28), 337–346.

Hilton, A., Beresford, D., Gentils, T., Smith, R., and Sun, W. (1999). Virtual people: Capturing human models to populate virtual worlds. In *Conference on Computer Animation, CA*, pages 174–185, Washington, DC, USA. IEEE Computer Society.

Hilton, A., Beresford, D., Gentils, T., Smith, R. J., Sun, W., and Illingworth, J. (2000). Whole-body modelling of people from multi-view images to populate virtual worlds. *The Visual Computer*, **16**(7), 411–436.

Horn, B. K. P. (1987). Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America*, **4**(4), 629–642.

Huang, G. and Wang, Y. (2007). Gender classification based on fusion of multi-view gait sequences. In *Asian Conference on Computer Vision, ACCV*, pages 462–471.

Jones, M. J. and Rehg, J. M. (2002). Statistical color models with application to skin detection. *International Journal of Computer Vision, IJCV*, **46**(1), 81–96.

Kakadiaris, I. and Metaxas, D. (2000). Model-based estimation of 3D human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **22**(12), 1453–1459.

Kakadiaris, I. A. and Metaxas, D. (1998). Three-dimensional human body model acquisition from multiple views. *International Journal of Computer Vision, IJCV*, **30**(3), 191–218.

Kehl, R., Bray, M., and Van Gool, L. (2005). Full body tracking from multiple views using stochastic sampling. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 129–136, Washington, DC, USA. IEEE Computer Society.

Lagarias, J. C., Reeds, J. A., Wright, M. H., and Wright, P. E. (1998). Convergence properties of the Nelder-Mead simplex method in low dimensions. *SIAM Journal of Optimization*, **9**(1), 112–147.

Laurentini, A. (1994). The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **16**(2), 150–162.

Lee, H. J. and Chen, Z. (1985). Determination of 3D human body postures from a single view. *Computer Vision Graphics and Image Processing, CVGIP*, **30**(2), 148–168.

Lee, W., Gu, J., and Magnenat-thalmann, N. (2000). Generating animatable 3d virtual humans from photographs. *Eurographics, Computer Graphics Forum, CGF*, **19**(3), 1–10.

Leotta, M. and Mundy, J. (2009). Predicting high resolution image edges with a generic, adaptive, 3-D vehicle model. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1311–1318.

Li, X., Maybank, S., Yan, S., Tao, D., and Xu, D. (2008). Gait components and their application to gender recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, **38**(2), 145–155.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *IEEE International Conference on Computer Vision, ICCV*, volume 2, pages 1150–1157. IEEE Computer Society.

Luong, Q.-T., Fua, P., and Leclerc, Y. (2002). The radiometry of multiple images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **24**(1), 19–33.

Magnenat-Thalmann, N., Seo, H., and Cordier, F. (2004). Automatic modeling of virtual humans and body clothing. *Journal of Computer Science and Technology*, **19**(5), 575–584.

Marr, D. and Nishihara, K. (1978). Representation and recognition of the spatial organization of three-dimensional shapes. *Biological Sciences, Series B*, **200**(1140), 269–294.

Menier, C., Boyer, E., and Raffin, B. (2006). 3d skeleton-based body pose recovery. In *International Symposium on 3D Data Processing, Visualization, and Transmission, 3DPVT*, pages 389–396.

Mikić, I., Trivedi, M., Hunter, E., and Cosman, P. (2003). Human body model acquisition and tracking using voxel data. *International Journal of Computer Vision, IJCV*, **53**(3), 199–223.

Moeslund, T. B. and Granum, E. (2001). A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding, CVIU*, **81**(3), 231–268.

Moeslund, T. B., Hilton, A., and Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding, CVIU*, **104**(2), 90–126.

Moghaddam, B. and Yang, M. (2002). Learning gender with support faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **24**(5), 707–711.

Mori, G. and Malik, J. (2006). Recovering 3D human body configurations using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **28**(7), 1052–1062.

Mündermann, L., Corazza, S., and Andriacchi, T. (2006). The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications. *Journal of Neuroengineering and Rehabilitation*, **3**(6).

Mündermann, L., Corazza, S., and Andriacchi, T. (2007). Accurately measuring human movement using articulated ICP with soft-joint constraints and a repository of articulated models. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*.

Nevatia, R. and Binford, T. O. (1973). Structured descriptions of complex objects. *International Joint Conference on Artificial Intelligence, IJCAI*, pages 641–647.

Onishi, N. (Jun 13, 2008). Japan, seeking trim waists, measures millions. *The New York Times*.

Park, S. I. and Hodgins, J. K. (2006). Capturing and animating skin deformation in human motion. *ACM Transactions on Graphics (TOG), SIGGRAPH*, **25**(3), 881–889.

Park, S. I. and Hodgins, J. K. (2008). Data-driven modeling of skin and muscle deformation. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, pages 1–6, New York, NY, USA. ACM.

Pentland, A. and Horowitz, B. (1991). Recovery of nonrigid motion and structure. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **13**(7), 730–742.

Plänkers, R. and Fua, P. (2001a). Articulated soft objects for video-based body modeling. In *IEEE International Conference on Computer Vision, ICCV*, volume 1, pages 394–401.

Plänkers, R. and Fua, P. (2001b). Tracking and modeling people in video sequences. *Computer Vision and Image Understanding, CVIU*, **81**(3), 285–302.

Plänkers, R. and Fua, P. (2003). Articulated soft objects for multiview shape and motion capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **25**(9), 1182–1187.

Prati, A., Mikic, I., Trivedi, M. M., and Cucchiara, R. (2003). Detecting moving shadows: Algorithms and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **25**(7), 918–923.

Rodgers, J., Anguelov, D., Pang, H.-C., and Koller, D. (2006). Object pose detection in range scan data. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2445–2452. IEEE Computer Society.

Rosenhahn, B., Kersting, U., He, L., Smith, A., Brox, T., Klette, R., and Seidel, H.-P. (2005). A silhouette based human motion tracking system. Technical Report CITR-TR-164, Centre for Image Technology and Robotics (CITR), University of Auckland.

Rosenhahn, B., Kersting, U., Powel, K., and Seidel, H.-P. (2006). Cloth X-ray: MoCap of people wearing textiles. *German Association for Pattern Recognition, DAGM*, pages 495–504.

Rosenhahn, B., Kersting, U., Powell, K., Klette, R., Klette, G., and Seidel, H.-P. (2007). A system for articulated tracking incorporating a clothing model. *Machine Vision and Applications, MVA*, **18**(1), 25–40.

Salzmann, M., Pilet, J., Ilic, S., and Fua, P. (2007). Surface deformation models for nonrigid 3D shape recovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **29**(8), 1481–1487.

Scheepers, F., Parent, R. E., Carlson, W. E., and May, S. F. (1997). Anatomy-based modeling of the human musculature. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, volume 31, pages 163–172.

Sclaroff, S. and Pentland, A. P. (1995). Modal matching for correspondence and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **17**(6), 545–561.

Segen, J. and Kumar, S. (1999). Shadow gestures: 3D hand pose estimation using a single camera. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 1, pages 479–485.

Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., and Szeliski, R. (2006). A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 519–528.

Seo, H. and Magnenat-Thalmann, N. (2003). An automatic modeling of human bodies from sizing parameters. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics, I3D*, pages 19–26, New York, NY, USA. ACM.

Seo, H. and Magnenat-Thalmann, N. (2004). An example-based approach to human body manipulation. *Graphical Models*, **66**(1), 1–23. Adding a link to the PDF from sciencedirect.com in JabRef confuses the LEd editor.

Seo, H., Cordier, F., and Magnenat-Thalmann, N. (2003). Synthesizing animatable body models with parameterized shape modifications. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA*, pages 120–125, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Seo, H., Yeo, Y. I., and Wohn, K. (2006). 3D body reconstruction from photos based on range scan. In *Technologies for E-Learning and Digital Entertainment*, volume 3942, pages 849–860.

Shakhnarovich, G., Viola, P., and Darrell, T. (2003). Fast pose estimation with parameter-sensitive hashing. In *IEEE International Conference on Computer Vision, ICCV*, page 750. IEEE Computer Society.

Shen, J. and Thalmann, D. (1995). Interactive shape design using metaballs and splines. In *Proc. Implicit Surfaces*.

Shlens, J. (2009). A tutorial on principal component analysis. `http://www.snl.salk.edu/~shlens/pub/notes/pca.pdf`. (Version 3.01).

Sidenbladh, H., Black, M. J., and Fleet, D. J. (2000). Stochastic tracking of 3D human figures using 2D image motion. In *European Conference on Computer Vision, ECCV*, pages 702–718, London, UK. Springer-Verlag.

Sigal, L., Isard, M., Sigelman, B. H., and Black, M. J. (2003). Attractive people: Assembling loose-limbed models using non-parametric belief propagation. In *Neural Information Processing Systems, NIPS*, pages 1539–1546.

Sigal, L., Bălan, A., and Black, M. J. (2008). Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Advances in Neural Information Processing Systems, (NIPS 2007)*, volume 20, pages 1337–1344. MIT Press.

Sigal, L., Bălan, A. O., and Black, M. J. (2010). Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision, IJCV*, **87**(1–2), 4–27.

Sminchisescu, C. and Telea, A. (2002). Human pose estimation from silhouettes a consistent approach using distance level sets. In *International Conference on Computer Graphics, Visualization and Computer Vision, WSCG*.

Sminchisescu, C. and Triggs, B. (2003). Estimating articulated human motion with covariance scaled sampling. *International Journal of Robotics Research, IJRR*, **22**(6), 371–393.

Sminchisescu, C., Kanaujia, A., Li, Z., and Metaxas, D. (2005). Discriminative density propagation for 3D human motion estimation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 390–397, Washington, DC, USA. IEEE Computer Society.

Starck, J. and Hilton, A. (2003). Model-based multiple view reconstruction of people. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 915–922.

Starck, J. and Hilton, A. (2007). Surface capture for performance-based animation. *IEEE Computer Graphics and Applications, CG&A*, **27**(3), 21–31.

Stauffer, C. and Grimson, W. (1999). Adaptive background mixture models for real-time tracking. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, volume 2, pages 246–252. IEEE Computer Society.

Stoykova, E., Alatan, A., Benzie, P., Grammalidis, N., Malassiotis, S., Ostermann, J., Piekh, S., Sainov, V., Theobalt, C., Thevar, T., and Zabulis, X. (2007). 3-D time-varying scene capture technologies-a survey. *IEEE Transactions on Circuits and Systems for Video Technology*, **17**(11), 1568–1586.

Sumner, R. W. and Popović, J. (2004). Deformation transfer for triangle meshes. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, pages 399–405, New York, NY, USA. ACM.

Taylor, C. J. (2000). Reconstruction of articulated objects from point correspondences in a single uncalibrated image. *Computer Vision and Image Understanding, CVIU*, **80**(3), 349–363.

Terzopoulos, D. and Metaxas, D. (1991). Dynamic 3d models with local and global deformations: Deformable superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **13**(7), 703–714.

Theobalt, C., Carranza, J., Magnor, M. A., and Seidel, H.-P. (2003). Enhancing silhouette-based human motion capture with 3D motion fields. In *PG '03: Proceedings of the 11th Pacific Conference on Computer Graphics and Applications*, page 185, Washington, DC, USA. IEEE Computer Society.

Urtasun, R., Fleet, D. J., and Fua, P. (2006). Temporal motion models for monocular and multiview 3d human body tracking. *Computer Vision and Image Understanding, CVIU*, **104**(2), 157–177.

Vedula, S. and Baker, S. (2005). Three-dimensional scene flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, **27**(3), 475–480.

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

Vlasic, D., Baran, I., Matusik, W., and Popović, J. (2008). Articulated mesh animation from multi-view silhouettes. *ACM Transactions on Graphics (TOG), SIGGRAPH*, **27**(3), 1–9.

Wachter, S. and Nagel, H.-H. (1999). Tracking persons in monocular image sequences. *Computer Vision and Image Understanding, CVIU*, **74**(3), 174–192.

Wang, R. Y., Pulli, K., and Popović, J. (2007). Real-time enveloping with rotational regression. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, pages 73:1–9, New York, NY, USA. ACM.

Weber, O., Sorkine, O., Lipman, Y., and Gotsman, C. (2007). Context-aware skeletal shape deformation. *Eurographics, Computer Graphics Forum, CGF*, **26**(3), 265–274.

White, R., Crane, K., and Forsyth, D. (2007). Capturing and animating occluded cloth. In *ACM Transactions on Graphics (TOG), SIGGRAPH*.

Wilhelms, J. and Van Gelder, A. (1997). Anatomically based modeling. In *ACM Transactions on Graphics (TOG), SIGGRAPH*, pages 173–180, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.

Yuille, A. L., Snow, D., Epstein, R., and Belhumeur, P. N. (1999). Determining generative models of objects under varying illumination: Shape and albedo from multiple images using SVD and integrability. *International Journal of Computer Vision, IJCV*, **35**(3), 203–222.