UNDERSTANDING MOTION CAPTURE FOR COMPUTER ANIMATION

UNDERSTANDING MOTION CAPTURE FOR COMPUTER ANIMATION — SECOND EDITION

ALBERTO MENACHE

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann Publishers is an imprint of Elsevier

MK MORGAN KAUFMANN

Acquiring Editor: Jenifer Niles
Development Editor: Robyn Day
Project Manager: Heather Tighe
Designer: Alisa Andreola

Morgan Kaufmann is an imprint of Elsevier
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

Library of Congress Cataloging-in-Publication Data
Menache, Alberto, 1961-
Understanding motion capture for computer animation / Alberto Menache. – 2nd ed.
p. cm.
ISBN 978-0-12-381496-8 (acid-free paper) 1. Computer animation. I. Title.
TR897.7.M4597 2011
006.6′96–dc22
2010035604

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

ISBN: 978-0-12-381496-8

Printed in the United States of America
11 12 13 14 15 10 9 8 7 6 5 4 3 2 1

For information on all MK publications visit our website at www.mkp.com

Dedication

I dedicate this book with infinite gratitude to my dear friends who looked out for me and my family through difficult times.

To Ronit, *mi gordita preciosa*.

PREFACE

One could say that motion capture technology hasn't evolved much in the past twenty years. It still relies mostly on cameras, electromagnetics, or mechanics. Of course, the components are much better and we are now able to obtain superior measurements. Data that took weeks to process in the early days can now be obtained in real time with the brand-new, ultra-high-resolution cameras or microelectromechanical systems. Components are now small enough and cheap enough for us to have consumer products that rely on human motion.

What has evolved the most is the understanding of the medium. We can now say that *Motion Capture* and *Performance Capture* are two very different things. At the very beginning of computer graphics, only engineers were capable of creating imagery. It was not until tools became friendly enough for artists that computer graphics flourished and became an integral piece of filmmaking. The same has occurred with Performance Capture as now real artists have embraced it and we are starting to see what the real potential is.

Besides updating the old material, I have now included descriptions of brand-new technologies that could potentially become the new standard. Those technologies will enable new and exciting applications in medicine, sports, security, and other areas.

There is also a fresh look at facial capture, descriptions of the newest file formats, and accounts of the latest projects where the technology was used successfully in creating full body and facial performances. I have also included an updated list of related resources available throughout the world.

The Updated Edition is aimed at the new Motion and Performance Capture artists and technicians, to give them an understanding of the history, the controversy, and what goes on under the hood.

Alberto Menache
mocap@menache.com

ACKNOWLEDGMENTS

Illustrations by David House
Motion capture data files by House of Moves

Thanks to the Following Persons for Their Contributions

Updated Edition

Jerome Chen
D.J. Hauck
Steven M. Ilous
Jon Landau

Original Edition

Robert Abel—Rest in Peace
Larry Bafia
Graham Bruce
Gordon Cameron
Benjamin Cheung
Richard Chuang
Peter Conn
Eric Darnell
Karen Goulekas
Rex Grignon
Tim Johnson
Jeff Kleiser
Fred Nilsson
George Merkert
Evan Ricks
Carl Rosendahl

Thanks to the Following Persons for Their Support and Assistance

Albert Hastings
Alexandra Menache
Brian Rausch

1

MOTION CAPTURE PRIMER

CHAPTER OUTLINE

Understanding Motion Capture for Computer Animation.

Motion Capture and Performance Animation

Motion capture is the process of recording a live motion event and translating it into usable mathematical terms by tracking a number of key points in space over time and combining them to obtain a single three-dimensional (3D) representation of the performance. In brief, it is the technology that enables the process of translating a live performance into a digital performance. The captured subject could be anything that exists in the real world and has motion; the key points are the areas that best represent the motion of the subject's different moving parts. These points should be pivot points or connections between rigid parts of the subject. For a human, for example, some of the key points are the joints that act as pivot points and connections for the bones. The location of each of these points is identified by one or more sensors, markers, or potentiometers that are placed on the subject and that serve, in one way or another, as conduits of information to the main collection device. From now on, when speaking generally about these, I will refer to them as "markers."

Performance animation is not the same as motion capture, although many people use the two terms interchangeably. Whereas motion capture pertains to the technology used to collect the motion, performance animation refers to the actual performance that is used to bring a character to life, regardless of the technology used. To obtain it, one must go through the whole process of motion capture and then map the resulting data onto a 3D character. In short, motion capture is the collection of data that represents motion, whereas performance animation is the final product of a character driven by a performer.

There are different ways of capturing motion. Some systems use cameras that digitize different views of the performance, which are then used to put together the position of key points, each represented by one or more reflective markers. Others use electromagnetic fields or ultrasound to track a group of sensors. Mechanical systems based on linked structures or armatures that use potentiometers to determine the rotation of each link are also available. Combinations of two or more of these technologies exist, and newer technologies are also being tested, all aiming for one result: real-time tracking of an unlimited number of key points with no space limitations at the highest frequency possible with the smallest margin of error. This is the Holy Grail of motion capture and probably the mission statement of every motion capture hardware manufacturer's research department. I later discuss how each of the current technologies falls short in this respect.

History of Performance Animation in the Entertainment Field

The Rotoscope

Motion capture in the entertainment field is the descendant of rotoscoping, a technique still used by some traditional animation studios to copy realistic motion from film footage onto cartoon characters.

The rotoscope device was invented and patented by cartoonist Max Fleischer in 1915, with the intent of automating the production of cartoon films. The device projected live-action film, a frame at a time, onto a light table, allowing cartoonists to trace the frame's image onto paper. The first cartoon character ever to be rotoscoped was Koko the Clown. Fleischer's brother, Dave, acted out Koko's movements in a clown suit. Fleischer wanted to use Koko to convince the big studios to use the new process for their cartoon projects. The sale was difficult because it had taken Fleischer about a year to produce the initial 1-min cartoon using the technique, so he couldn't market it as a mass production tool. Eventually, Fleischer realized that rotoscoping would be a viable technique only for certain shots that required realistic motion.

Walt Disney Studios used some rotoscoping in 1937 to create the motion of human characters in *Snow White*. Snow White herself and the Prince were partially rotoscoped. The decision to use rotoscoping wasn't a matter of cost, but of realistic human motion. In fact, *Snow White* went tremendously over budget due to the complexity of the animation.

Rotoscoping has been adopted over the years by many cartoon studios, but few actually admit using it because many people in the animation industry consider it cheating and a desecration of the art of animation.

A two-dimensional (2D) approach, rotoscoping was designed for traditional, hand-drawn cartoons. The advent of 3D animation brought about the birth of a new, 3D way of rotoscoping. Hence, motion capture.

Brilliance

Some of the current motion capture technologies have been around for decades, being used in different applications for medical and military purposes. Motion capture in computer graphics was first used in the late 1970s and early 1980s in the form of research projects at schools such as Simon Fraser University, Massachusetts Institute of Technology, and New York Institute of Technology, but it was used in actual production only in the mid-1980s.

In late 1984, Robert Abel appeared on a talk show and was asked if he would soon be able to counterfeit people digitally. "We are a long ways from that," he replied. "We haven't even figured out human motion, which is the basis, and that's a year away." A week and a half later, Abel received a visit on a Friday afternoon from a creative director from Ketchum, a prominent advertising agency. The visitor brought six drawings of a very sexy woman made out of chrome. She was to have Kathleen Turner's voice and would be the spokesperson for the National Canned Food Information Council, an association formed by Heinz, Del Monte, Campbell's, and a number of big players that sold canned food. They felt they had to make a powerful statement because the idea of buying food in cans was becoming obsolete, so they wanted to do something really different and outrageous, and they wanted it to air during the Super Bowl in January 1985. "Can you do it?" asked the client. "You're certainly here a lot earlier than I would have planned," replied Abel, and asked the client to wait until the end of the weekend for an answer.

At that time most computer graphics consisted of moving logos, landscapes, and other hard objects, and Robert Abel and Associates had already become a player in that market, along with MAGI, Triple-I (Information International, Inc.), John Whitney's Digital Productions, and PDI, all of which had their own proprietary software, because at that time there was almost no off-the-shelf animation software and whatever was available was still in its infancy. Abel's software was initially based on bits and pieces from Bell Labs, Evans and Sutherland, JPL, and other places, and was augmented over time by his group.

The next step would be to animate a digital character. "For storytelling, which was really our goal, we had to have human characters," recalled Abel, "because nobody better than a human character is able to convey emotion and story. We come from a long line of storytellers that go back maybe 35,000 years, and although the forms may change from cave paintings to digitally made motion pictures, it's still the same thing, it's the passing on of stories." Creating the first animated digital character would open a Pandora's Box of many new challenges, such as creating realistic skin, hair, and expression. But first they had to deal with the motion.

Abel and his team decided to lock the doors and not leave the building until Monday morning. If by then they didn't have the solution figured out, they would have to pass on the project. Robert Abel and Associates' background in shooting miniatures with motion control cameras since the late 1960s and early 1970s was the key to the solution. They knew that the answer to their problem would have to do with motion and control, except this time it would be human motion. Keyframe character animation was not an option at the time, so they decided to find a way to track the motions of a woman acting the part of the character. It made sense to shoot the woman with several cameras from different points of view, and then use this footage to create a motion algorithm.

Seven people worked throughout the weekend. "Several of us got into our underwear," recalled Abel. "We got black adhesive dots and we put them on our bodies and we would photograph each other with Polaroid cameras and then we would lay out these Polaroids so we could see how they changed from angle to angle." They continued this slow deductive reasoning process until Sunday at 3 A.M., when they decided that it would take a few weeks to digitize all the motion. It would be close, but they felt that they could do the job in the 8-week schedule that the client had established.

Among the people involved in this project besides Bob Abel were Bill Kovacs and Roy Hall, who later became cofounders of Wavefront Technologies; Frank Vitz, now a Senior Art Director at Electronic Arts; Con Pederson, cofounder of Metrolight Studios a few years later; Randy Roberts, who directed the project and still a commercial director; Richard Hollander, former president of Blue Sky/VIFX and of the feature film division at Rhythm and Hues, now at Pixar; Neil Eskuri, currently visual effects director at Rainmaker in Vancouver; Charlie Gibson, Oscar-winning visual effects supervisor for *Babe* and *Pirates of the Caribbean*; and John Hughes, who later became cofounder and president of Rhythm and Hues.

They found a woman who was very pretty and graceful and had been a dancer and a model. They had decided that motion on 18 hinge points would be necessary to achieve the desired result, so with black magic markers they put black dots on each of her 18 joints. A stool with a 360° axis of rotation in the middle was assembled so that the model could sit and perform the moves without obstacles. The team photographed her from multiple points of view, and then managed to import the images to SGI Iris 1000 systems. These workstations appeared in early 1984 and were the first model produced by Silicon Graphics. They were then able to analyze the difference in measurement between pairs of joints (e.g., the elbow and the wrist) for each point of view and to combine them to come up with a series of algorithms that would ultimately be used to animate the digital character. This process was done on a frame-by-frame basis and took 4½ weeks.

At the same time, the wire-frame model was built in separate sections, all rigid parts. The motion algorithms were applied to all the combined moving parts, and the animation was output as a vector graphic. "We then had to deal with the basic issue of wrapping her body in chrome," said Abel. "Of course, there is no way in the world we could do ray-tracing to real reflective chrome the way those guys do it at SIGGRAPH, with those multimillion dollar supercomputers. We had VAX 750s, which were early DEC computers." This problem was solved by Charlie Gibson, who figured out a way of texture mapping the body so that when it moved, the map would animate following the topology of the body. Today we call this a reflection map.

The last challenge was to render the final spot, all 30 s of it, in the 2 weeks that they had left. "The good and the bad news is this," Abel announced. "We don't nearly have the horsepower, but the VAX 750 is a staple of almost every university, laboratory, and engineering place in the country." They ended up using 60 additional VAX 750s around the country, from Florida to Alaska to Hawaii, and even a few places in Canada. The final spot, "Brilliance," was rendered and pieced together about 2 days before the delivery date. It is now known by most people in the industry as "Sexy Robot." Abel, who passed away in 2001, was a pioneer in visual effects, computer graphics, and motion capture.

Pacific Data Images

Pacific Data Images (PDI) was the oldest operating computer animation studio until it was completely acquired by DreamWorks in 2000, and it played a big part in the history of performance animation. Founded in 1980 in Sunnyvale, California, by Carl Rosendahl, and later joined by Richard Chuang and Glenn Entis, it wasn't until 8 years later that the studio would produce its first project using some kind of human tracking technology.

Over the years, PDI used different types of tracking devices that fit particular project needs, ranging from custom-built electromechanical devices to electromagnetic and optical tracking systems, but it wasn't PDI's intention to specialize in motion capture. "We use technology where it is appropriate," says Richard Chuang. "We are not a one-technique company; our goal is not to be a master of any one thing, just be good at a lot of them."

The Jim Henson Hour

In 1988, PDI began collaboration with Jim Henson and Associates to create a computer-generated character for *The Jim Henson Hour*. Henson had already done some tests with a digital character at Digital Productions and had been holding the idea until the technology was mature enough to produce the result he wanted. Graham Walters, who later became a technical supervisor at Pixar, was the technical lead for PDI on the project.

The character was called Waldo C. Graphic. "One of the criteria on the project was that they wanted to be able to perform the character live when they were videotaping the TV show," recalls Carl Rosendahl. "It didn't mean that the final render needed to be live; they had to be able to record him live because of the spontaneity of the performance and such." Figure 1.1 is a photograph of the taping of *The Jim Henson Hour*.

Henson already had been doing some work with wireless telemetric robotics to animate characters for shots in which there was no way of controlling them directly, such as for a scene in *The Muppet Movie* in which Kermit is riding a bicycle and the audience sees the whole character in frame with his head moving around and singing a song. "It was a robot Kermit, and Jim had this foam Kermit head with some potentiometers on it that transmitted a radio signal to the robot on the bike," says Rosendahl. That device, shown in Figure 1.2, was used for the Waldo C. Graphic character, with some modifications. PDI built a mechanical arm to record the position of the character in space. Instead of the radio link, the robot had a direct hardware

Figure 1.1 Taping the *Jim Henson Hour*. Photo courtesy of Rex Grignon.

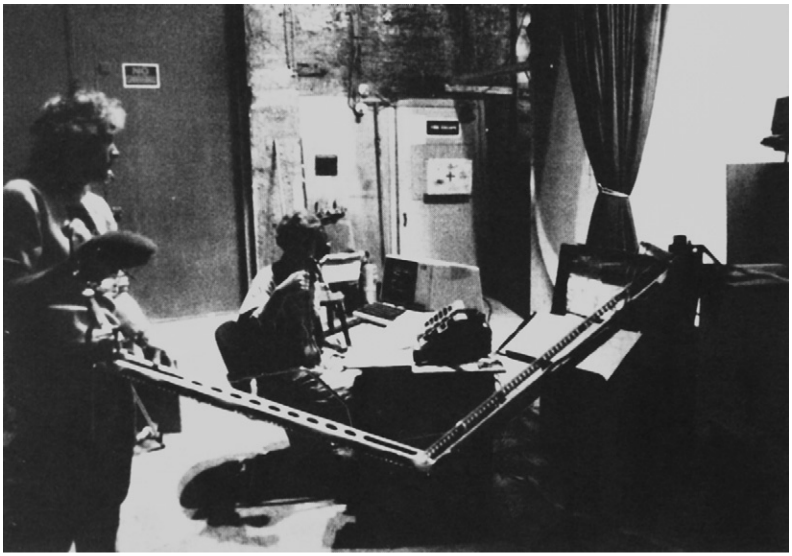

Figure 1.2 Steve Whitmayer (L) and Rex Grignon operating the device used to control Waldo C. Graphic. Photo courtesy of Rex Grignon.

link to one of PDI's Silicon Graphics systems. A control box with buttons allowed them to do eye blinks.

"All the videotape recording was done in Toronto, so we moved all the equipment up there and basically got it to work live on the set," says Rosendahl, "so that we took a feed of the video camera and then comped in Waldo on top of it, and that's what the performance would look like, even though the video was recording the final image without Waldo in there, but the performers would look at the one with the rough-version Waldo."

Steve Whitmayer, who after Henson's death was in charge of bringing *Kermit the Frog* to life, was the puppeteer operating the device. "Essentially, it was a puppet, something that the puppeteer was familiar with at the end," recalls Rex Grignon, Head of Character Animation at PDI/DreamWorks Animation. "He could put his hand in this and just fly it anywhere in the space within that region." During the show, the puppeteer's motion data was read in and interpreted to control Waldo on the screen as a low-resolution version. "We modeled a 50-polygon version of Waldo," says Grignon. "It was all within screen space. The puppeteers, when doing a normal puppet, would have their hand in the puppet, and they'd be looking down at a monitor, so they'd be able to gauge their performance, so they were absolutely used to this." The new technique had the added benefit that the puppeteer could bring the puppet down and not worry about hiding from the camera.

Grignon was in charge of recording the motion. "As we started a performance I'd essentially start the system recording, do the blinks, and when the director said cut, I'd cut and I'd save that take in a similar language to what they were using on the show," says Grignon. They kept track of the takes that Henson liked, because they needed to ensure that the data stayed in sync with the rest of the characters in the show. This was necessary so that after the editing of the show, PDI could use the data matching the chosen takes to produce the animation.

The device generated some interference, but although the character tended to jump and pop around, it looked pretty good for the live session. After the recording session, the data would be sent via phone line to the PDI studio, then located in Sunnyvale, California, and a PDI team would massage it to eliminate all the noise and add the necessary secondary animation, such as the belly wiggle. The team would also add several costume changes that required transitions, such as scenes in which the character turned into other objects (e.g., a book or an electric saw), and would render the final high-resolution character.

They were doing one episode a week with about 1 min of the Waldo character. "We'd go up on the weekend, we'd shoot the show on Monday, we'd send the tape back on Tuesday, and then Michelle Choi, who was here, would basically render the tape. We'd come back, we'd all work on this for about 5 days rendering a minute of stuff and adding any extra props and extra animation," recalls Grignon. "We lived here, we just lived here. We slept here every night."

"I still think this is one of the best uses of motion capture," notes Grignon, "You can take advantage of the puppeteer's skills, because these guys are masters. It's just amazing seeing them put their hand in a puppet and it just comes to life. I just remember Graham and I, both were just continually astounded with the subtleties that these guys could bring to these characters."

Waldo C. Graphic was brought back to life by PDI in 1991 for the MuppetVision 3D movie, still being shown at Disney-MGM Studios in Orlando, Florida.

Exoskeleton

In 1988, PDI commissioned Rick Lazzarini's The Creature Shop to build a Waldo device for the upper body and head. They called this mechanical device an "exoskeleton"; it was based on optical potentiometers on each joint (see Figure 1.3). "It took us two passes to get one that really worked well," recalls Carl Rosendahl. "Analog parts are too noisy," notes Richard Chuang. The second version had digital parts.

Figure 1.3 Jamie Dixon wearing the PDI exoskeleton as Graham Walters makes adjustments during a shoot for *Toys*. Photo courtesy of PDI.

The second exoskeleton was used for the Barry Levinson movie *Toys*, in a sequence for which PDI had to create an X-ray view of an ongoing meeting. "Jamie Dixon, who was our lead guy in the LA office, was the effects supervisor, and he actually was the one who did the performing," says Rosendahl. "And because there were multiple characters, he actually did it in multiple passes." Rosendahl also recalls that there were some glitches in the data that had to be cleaned up after the performance, but usable data was available in a reasonable amount of time. *Toys* was the first motion picture in which a digital character was successfully created using motion capture.

PDI later used a Flock of Birds, an electromagnetic device manufactured by Ascension Technology Corporation, on several projects in their Los Angeles office. One of them was a bit called “The Late Jackie Lenny” for a Comedy Central pilot in which a skeleton was a talk show host and would interview live comics or do a stand-up act.

For all the projects involving that device, PDI fabricated accessories to help hold the sensors in place and make the capture experience easier on the performer. Most of these were based on Velcro strips, but the head sensor was placed on top of a baseball cap so that when the performer took off the cap, the character would look like it had taken off its head. The device had some problems with interference. For example, the studio was located on the third floor of a three-story building, and the air conditioning equipment was located on the roof, right above the stage. When the air conditioning kicked on while the device was being used, the characters on screen would sway up and down.

deGraf/Wahrman

Brad deGraf also experimented with performance animation while working at Digital Productions, using Jim Henson’s Waldo device. After Digital Productions was purchased by Omnibus, Brad left and joined forces with Michael Wahrman, also from Digital Productions, and founded deGraf/Wahrman Production Company.

In 1988, Silicon Graphics contracted the services of deGraf/Wahrman to create a demonstration piece for their new 4D models. The piece was an interactive animation called “Mike the Talking Head,” which was showcased at SIGGRAPH’88. For the first time, an animated character was able to interact with the audience. The controls that animated the character in real time were operated by a puppeteer during the conference. deGraf/Wahrman’s proprietary software was used to create an interface between the controls and the rendering engine and to produce interpolated instances of the character’s geometry. The new Silicon Graphics 4D workstation had the horsepower to render the character in real time.

When deGraf and Wahrman dissolved their partnership, Brad joined Colossal Pictures and started rewriting the performance animation software. In 1993, he developed “Moxy,” a character for the Cartoon Network that was operated in real time using an electromagnetic tracker. In 1994, he founded Protozoa, a spin-off from Colossal’s performance animation studio. His focus has been on real-time performance animation solutions, including software and character development for different media, including television and the Web. ALIVE, Protozoa's proprietary performance animation software, supports multiple input devices, including motion capture systems, joysticks, and MIDI controllers. It also outputs to different formats, including live television and the World Wide Web via VRML (Virtual Reality Modeling Language).

Kleiser-Walczak Construction Company

In 1986, Jeff Kleiser began experimenting with motion capture while he was at Omnibus Computer Graphics. "We used the optical system from Motion Analysis in Santa Rosa, California, to encode martial arts movement for use in a test for Marvel Comics," recalls Kleiser. "Results were disappointing due to the alpha code we were working with."

In 1987, after Omnibus closed, Kleiser joined forces with Diana Walczak, who had been sculpting human bodies, and founded Kleiser-Walczak Construction Company. Their specialty would be to build and animate computer-generated actors, or Synthespians.

"After creating our first Synthespian, Nestor Sextone in 1988, we got together with Frank Vitz and went back to Motion Analysis to capture motion for our second digital actor, Dozo, in creating the film *Don't Touch Me*, in 1989," says Kleiser. "We were only able to get about 30 s of usable motion capture, and we had to recycle it to fill the 3.5 min of the song we had written." Vitz had been working with Robert Abel and Associates and had some motion capture experience, as he had been part of the team that created "Brilliance."

Over the years, Kleiser-Walczak has created digital actors for special venue, commercial, and feature film projects. They created dancing water people for the Doug Trumbull stereoscopic ride "In Search of the Obelisk," located inside the Luxor Hotel in Las Vegas. Using a Flock of Birds electromagnetic system by Ascension Technology Corporation, they also created digital stunt doubles for Sylvester Stallone, Rob Schneider, and others for the film *Judge Dredd*. In their most recent use of motion capture, Kleiser-Walczak produced "Trophomotion," a commercial spot for Stardox athletic braces in which two basketball trophies come to life. They used a combination of keyframe animation and motion capture, which they achieved with an optical system manufactured by Adaptive Optics in Cambridge, Massachusetts.

Their latest project is computer imagery for "The Amazing Adventures of Spiderman," a ride for Universal Studios' Islands of Adventure, in Orlando, Florida. "We tested mocap [motion capture] for this project, but it quickly became clear that superhero characters need to have super-human motion, and that keyframe animation gave us the look and flexibility we wanted," says Kleiser. "All the animation in the project was therefore done with keyframe animation."

Homer and Associates

In the early 1990s, projects utilizing motion capture in computer graphics were starting to become part of actual production work, so companies whose main business was based on this technology started to surface. Medialab, Mr. Film, Windlight Studios, SimGraphics, and Brad deGraf at Colossal Pictures concentrated their efforts on real-time applications that included character development and puppeteering, while Biovision, TSi, and Acclaim embraced the non-real-time technology for the up-and-coming video game market. At the same time, commercial production using the now traditional motion capture techniques was initiated by Homer and Associates.

Party Hardy

Although Homer and Associates had already created a shot for the film *Lawnmower Man* in 1991 using motion capture, they produced their initial entry in the advertising market in 1992: "Party Hardy," a spot promoting the Pennsylvania Lottery. It consisted of an animated crowd of lottery tickets at a costume party. The spot was especially challenging because it had to have humanoid motion and facial expressions, and each ticket had to be different. There also had to be a feeling of interaction among the characters.

Peter Conn, president and founder of Homer and Associates, decided to use a camera-based system to collect the motions for the spot. The system used was an Elite Motion Analyzer, a system manufactured by Bioengineering Technology Systems (BTS) in Italy primarily for medical and industrial applications, and provided by SuperFluo, a company dedicated to bringing these medical systems to the entertainment industry. SuperFluo added custom software geared to computer animation to the already existing configuration.

The spot was directed by Michael Kory, who also performed the motions for all the characters. For the body motions, markers were placed on human-size foam rectangles, shown in Figure 1.4, which Kory held while acting. The facial motions were also performed by Kory using the Elite system, but this time using smaller markers placed in specific areas of his face.

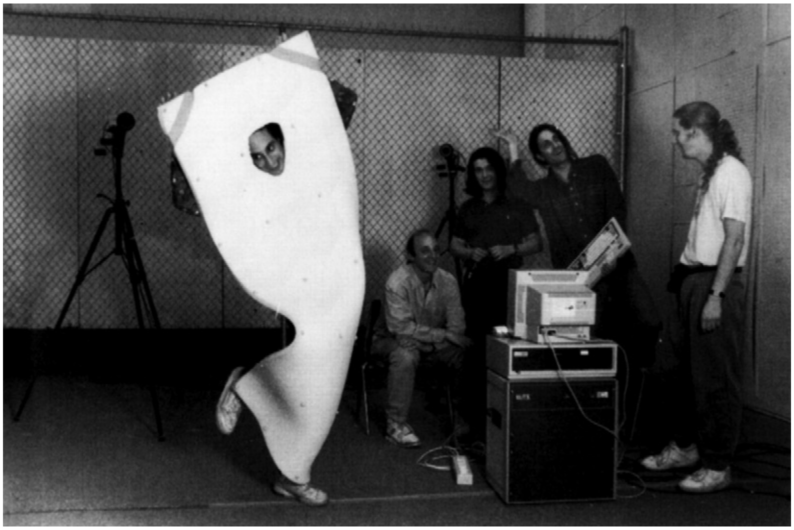

Figure 1.4 The Homer & Associates team with the BTS motion capture system for the Pennsylvania Lottery commercial “Party Hardy”: (left to right) Michael Kory, director (behind foam); Peter Conn, Producer; Francesco Chiarini (SuperFluo, motion capture specialist); Umberto Lazzari (SuperFluo, motion capture specialist); John Adamczyk, technical director.

The captured facial motion was used to help interpolate between facial shapes that were built by Kory.

“Party Hardy” is a good example of a project in which motion capture was used to collect data from an object or character puppeteered by a performer, as opposed to data collected from the actual performer’s body.

Steam

Peter Gabriel’s music video “Steam” was a coproduction between Colossal Pictures and Homer and Associates. The video was produced in 1992 and was directed by Stephen Johnson, director of other award-winning Peter Gabriel videos, including “Big Time.”

“There was a period of several months when we were actively working with Brad deGraf and Stephen Johnson,” recalls Peter Conn, president of Homer and Associates. “He [Stephen Johnson] had a fertile imagination and an equally formidable lack of decisiveness and reality orientation. I remember that every meeting we had about what the video would include always involved going through dozens of storyboards of very elaborate effects. Although the beginning of the song was precisely worked out bar by bar, we never ever got through even half the song. There were dozens of elaborate concepts, which he seemed to want in the video. Since he was having an inability to downsize the scope, he would fly to London or Senegal to get Peter’s input. When he came back, there was never any consensus, just more and more concepts and ideas. A lot would have been almost achievable had the months of prep been actual production, but by the time the video was officially green-lighted there was no more than 4 weeks or so left to do everything. Motion capture was always specified as the great technique that would somehow make it all possible."

The motions would be collected using the same BTS optical system used on "Party Hardy." "By the time Peter Gabriel showed up for the motion capture sessions, the SuperFluo guys, Umberto Lazzari and Francesco Chiarini, had set up in our usual place, the large abandoned white storage room in the basement of the adjacent building," says Conn.

They spent 2 days capturing motion data from Peter Gabriel and a dancer. About 150 different movements were collected. "Peter had one rule for the room: no spectators," recalls Conn. "When the playback rolled, everyone had to dance and get into the music. He liked so much doing the motion samplings that he refused to quit. Knowing that we had only time left to animate a few scenes, it was way over the top, but hey, it was Peter Gabriel and we were getting data."

In the end, there were only a couple of shots in which captured motion data was used. The longest shot was one in which Peter was supposed to be made of ice, with dancing fire girls alongside. "He was to melt then re-emerge as water," says Conn. Michael Kory animated the scene and Scott Kilburn wrote the particle software, which was based on Homer's proprietary code that had been written by John Adamczyk for *Lawnmower Man*.

A crew of eight undertook the task of putting together the shots in a short period of time, while the director was in London. "Memorable moments included a conference call after they had seen a test over in the UK," says Conn. "Stephen was upset because he said that he had wanted the girls 'not on fire' but 'of fire.' We didn't really know what he meant, but we kept adding more particles. Then after the scene was delivered, he was still upset. It turned out that the girls weren't voluptuous enough, so we took the models and pulled the breasts out an ungodly much and then rendered just some breast fire and comped it on."

The other scene with captured motion data was the "Garden of Eden," in which both Gabriel and the dancer turn into digital characters after walking through an imaginary plane. "Actually, Brad [deGraf] did a great mocap piece with Peter Gabriel as a puppet but for some reason it never made the final cut," recalls Conn. "Despite the multitudinous mistakes and us knowing what it could have been given more time, the video went on to win some major awards, like the Grammy for Video of the Year in 1993," he notes.

Other notable motion capture works involving Homer and Associates include shots from the film *Lawnmower Man* and music videos such as Vince Neil's "Sister of Pain" and TLC's

"Waterfalls," produced for Atomix with motion data from TSi, my former company. This was my first collaboration with Homer and Associates and Atomix.

Motion Capture Service Bureaus

In the early 1990s, a few companies started offering motion capture services to other production companies. The early players were Biovision in San Francisco and TSi in Los Angeles and San Francisco. The systems used were optical and the software to solve the motion data was all proprietary, because the few companies that sold optical motion capture equipment didn't have biomechanical solvers that could deliver data useable for animation to any of the current 3D animation software programs. Entertainment was a brand new market for them as they have been servicing mostly the Life Sciences sector at that time, so they looked at the service bureaus to help them access the market.

Owning and operating an optical system at the time was not only very expensive, but extremely technical and not user friendly by any means. The data was very difficult to collect and there was always a risk that the final data would be unusable. Verifying the data during the shoot wasn't possible as it took a very long time to process. Initially, at TSi we had mostly clients in the video gaming industry. Video game consoles were becoming more powerful and could render more complex motions, so most game developers started using motion capture when service bureaus started doing business. We were servicing clients from all over the world for a few years until motion capture systems became more user friendly. In 1997, TSi sold their biomechanical solver and animation software plug-ins to Motion Analysis Corporation. Optical systems started including the necessary software to produce performance animation and other service bureaus started opening in many countries. Also, larger video game developers such as Electronic Arts started acquiring systems to produce in-house graphics. Today it is a given that any video game that has human characters is expected to have used motion capture to generate the performance data.

Types of Motion Capture

Human motion capture systems are classified as outside-in, inside-out, and inside-in systems. These names are indicative of where the capture sources and sensors are placed.

- An outside-in system uses external sensors to collect data from sources placed on the body. Examples of such systems are camera-based tracking devices, in which the cameras are the sensors and the reflective markers are the sources.
- Inside-out systems have sensors placed on the body that collect external sources. Electromagnetic systems, whose sensors move in an externally generated electromagnetic field, are examples of inside-out systems.
- Inside-in systems have their sources and sensors placed on the body. Examples of these devices are electromechanical or inertial suits, in which the sensors are potentiometers, powered goniometers or accelerometers and gyroscopes, and the sources are the actual joints inside the body.

The principal technologies used today that represent these categories are optical, electromagnetic, and inertial human tracking systems.

Optical Motion Capture Systems

Optical motion capture is a very accurate method of capturing certain motions when using a state-of-the-art system. It is not a real-time process in most cases; immediate feedback is not possible on the target character unless motions are not too complex and there aren't too many performers. Data acquired optically can require extensive postprocessing to become usable, so operating costs can be high.

A typical optical motion capture system is based on a single computer that controls the input of several digital CCD (charge-coupled device) cameras. CCDs are light-sensitive devices that use an array of photoelectric cells (also called *pixels*) to capture light, and then measure the intensity of the light for each of the cells, creating a digital representation of the image. A CCD camera contains an array of pixels that can vary in resolution from as low as 128×128 to as high as millions of pixels. The state of the art today is 16 million pixels (megapixels), but that number will continue to increase every year.

Obviously, the higher the resolution, the better, but there are other trade-offs. The samples-per-second rate, or frame rate, has to be fast enough for capturing the nuances of very fast motions. In most cases, 60 samples per second are more than enough, but if the motions are very fast, such as a baseball pitch, a lot more samples per second are required. By today's standards, a CCD camera with a resolution of 16 megapixels would be able to produce up to 120 samples per second at that resolution. To capture faster motions, the resolution has to be dropped. Another important feature is shutter synchronization, by which the camera's shutter speed can be synchronized with external sources, such as the light-emitting diodes (LEDs) with which optical motion capture cameras are usually outfitted.

Some of the most modern cameras process the imagery locally before sending it to the computer, saving broadband and processing time of huge amounts of data. This can be very useful, especially when the system has a large number of ultra-high-resolution cameras, each capturing hundreds of images per second.

The number of cameras employed is usually no less than 8 and no more than 32, but there are cases where hundreds of cameras are used as I will explain later. They capture the position of reflective markers at speeds anywhere between 30 and 2000 samples per second. The cameras are normally fitted with their own light sources that create a directional reflection from the markers, which are generally spheres covered with a material such as Scotch-Brite tape. Red light sources are preferred because they create less visual distortion for the user. Infrared is also used but it is slightly less effective than visible red. The marker spheres can vary from a few millimeters in diameter, for facial and small-area captures, to a couple of inches. The Vicon system, shown in Figure 1.5, is an example of a state-of-the-art optical system that can accommodate up to hundreds of cameras.

Most optical systems were designed and manufactured for medical applications; as such, they lacked many features that are important to computer graphics applications. The Vicon 8 system was the first system designed with computer graphics in mind. It was the first able to support SMPTE time code, a time stamp used by most film and television applications. Even if you videotaped your capture session, there was no easy way to match the video to the actual motion data. Having time code in the motion data allows you to edit the motion files, as you would do with live-action video, and properly plan the association of the characters with background plates. Another very

Figure 1.5 The Vicon MX optical motion capture system. Photo courtesy of Vicon Motion Systems.

useful feature that the Vicon 8 introduced was the fact that reference video of the session could be synchronized, or genlocked, with the actual data collection. In addition to video genlock, the Vicon 8 software could shoot AVI movie files at the same time as it captured. These movie files are great reference for data postprocessing and application.

The optical system must be calibrated by having all the cameras track an object with known dimensions that the software can recognize, such as a cube or a wand with reflective markers. By combining the views from all cameras with the known dimensions of the object, the exact position of each camera in space can be calculated. If a camera is bumped even slightly, a new calibration must be performed. It is a good idea to recalibrate the system many times during a capture shoot, since any kind of motion or vibration can shift the position of a camera, especially if the studio is located on unstable ground.

At least two calibrated views are needed to track a single point's 3D position, and extra cameras are necessary to maintain a direct line of sight from at least two cameras to every marker. That doesn't mean that more cameras are better, because each additional camera increases postprocessing time. There are other methods for minimizing occlusions that are implemented in software and used during postprocessing. The most time- and cost-effective solution is different for every case, depending on the type, speed, and length of the motion, as well as on the volume of capture and the available light. Figure 1.6 shows a performance being filmed on an optical motion capture stage.

Once the camera views are digitized into the computer, it is time for the postprocessing to begin. The first step is for the software to try to produce a clean playback of only the markers. Different image processing methods are used to minimize the noise and isolate the markers, separating them from the rest of the environment. The most basic approach is to separate all the groups of pixels that exceed a predetermined luminosity threshold. If the software is intelligent enough, it will use adjacent frames to help solve any particular frame. The system operator has control over many variables that will help in this process, such as specifying the minimum and maximum lines expected per marker, so the software can ignore anything smaller or bigger than these values.

The second step is to determine the 2D coordinates of each marker for each camera view. This data will later be used in combination with the camera coordinates and the rest of the camera

Figure 1.6 A performance in an optical motion capture stage. Photo courtesy of House of Moves.

views to obtain the 3D coordinates of each marker. The third step is to actually identify each marker throughout a sequence. This stage requires the most operator assistance, since the initial assignment of each marker has to be recorded manually. After this assignment, the software tries to resolve the rest of the sequence until it loses track of a marker due to occlusion or crossover, at which point the operator must reassign the markers in question and continue the computation. This process continues until the whole sequence is resolved and a file containing positional data for all markers is saved.

The file produced by this process contains a sequence of marker global positions over time, which means that only each marker's Cartesian (x, y, and z) coordinates are listed per frame and no hierarchy or limb rotations are included. It is possible to use this file for computer animation, but a more extensive setup is required inside the animation software in order to resolve the final deformation skeleton to be used. Experienced technical directors can benefit by using this file's data, because it allows more control over what can be done in a character setup. For the average user, however, the data should be processed further, at least to the point of including a skeletal hierarchy with limb rotations. Most optical systems include data-editing systems that allow the user to produce the rotational hierarchical data before importing it to animation software. Figure 1.7 shows a sample screen of the Vicon Blade integrated motion capture platform that is shipped with the Vicon MX optical motion capture system.

Advantages of Optical Systems

- Optical data can be extremely accurate.
- A larger number of markers can be tracked.
- It is easy to change marker configurations.
- It is possible to obtain approximations to internal skeletons by using groups of markers.

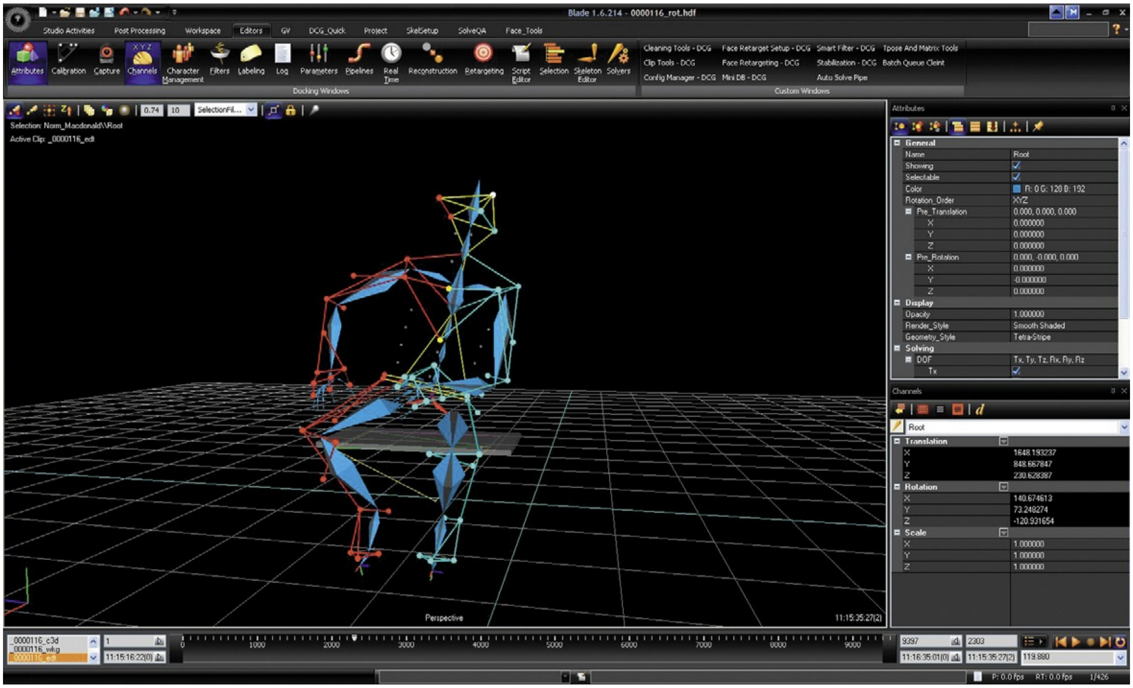

Figure 1.7 Vicon Blade motion capture software. Photo courtesy of Vicon Motion Systems.

- Performers are not constrained by cables.
- Optical systems allow for a larger performance area than most other systems today.
- Optical systems have a higher sample rate of capture, resulting in more measurements per second.

Disadvantages of Optical Systems

- Optical data requires extensive postprocessing, so operating costs are high.
- Hardware is expensive. An entry-level optical system can cost over $50,000 and a high-end system can cost over a million.
- Optical systems cannot capture motions when markers are occluded for a long period of time.
- Capture must be carried out in a controlled environment, away from yellow light and reflective noise.

Radio Frequency Positioning Systems

For the past couple of decades, there have been no major changes in the way motion capture is done. Since the late 1980s, optical has been the preferred method in the entertainment and life sciences fields, as accuracy is more necessary than real-time measurement.

Technological advances in hardware since optical systems emerged have been mostly in camera count and camera resolution. Meanwhile, other technologies have remained stagnant in their advances. Software improvements have been mostly in finding clever ways to obtain somewhat real-time feedback and better ways of using and managing data.

Radio frequency (RF) is the new emerging technology in position tracking. Not to be confused with using RF to deliver data wirelessly, but rather using RF to calculate position measurement. Currently, there are several types of RF-based systems, but none of them has the accuracy needed for performance capture. However, it is clear that this will be the next state of the art in motion capture.

Global Positioning System (GPS)

The best example of RF positioning that has been around for a while is GPS. GPS was originally developed by the US Department of Defense in the 1970s as a tool to aid in the location of ballistic missile submarines, because it is necessary to know the exact location of the submarine at the time of launch in order to make sure the missile hits its target. The first GPS satellite was launched in 1978 and the full constellation was completed in 1995. In 1983, President Reagan made GPS available for nonmilitary use after Soviet fighter jets shot down Korean Air flight 007, a passenger jet that had accidentally entered Soviet airspace, killing everyone on board. The nonmilitary version of GPS was made intentionally less accurate by the US government until the year 2000, when it stopped distorting signals for security reasons. As a result, civilian uses of GPS started to flourish and demand has kept increasing. It is estimated that the worldwide civilian market for GPS devices will reach $75 billion by 2013.

The United States' GPS constellation consists of 24 satellites that travel in a very precise orbit at 12,000 miles above the earth. They complete a full cycle around the globe in about 12 h. The system is designed so that at least five satellites have a simultaneous view of any point on the surface of the earth.

GPS units receive the signals of the satellites and use them to compute their location relative to a coordinate system. The coordinate system varies by manufacturer, but it is usually based on longitude/latitude like DMS (Degree/Minute/Second) or some decimal version of that. At least four satellites are required to make the position calculation.

Some GPS satellites have been around for a couple of decades and are already due for replacement. Meanwhile, the European Union is due to launch its own version of a satellite navigation system called Galileo in 2013.

Today, GPS can be found in vehicles, mobile phones, computers, shipping containers, wristwatches, dogs, and even troops in the battlefield. It isn't accurate enough or has enough samples per second to use as a motion capture tool, but other systems based on similar technology could achieve the accuracy and frequency required if they could be localized. In other words, instead of satellites sending signals through the atmosphere, it would use local transmitters in a contained environment and receivers placed on the subject.

Real-Time Location Systems (RTLS)

RTLS were originally conceived as an addition to RFID technology. RFID is an electronic replacement of barcode to be used as an asset tracking and inventory device. The idea was that in addition to having the information that the RFID chips contain, the exact position within the warehouse or retail facility would also be available.

RFID hasn't caught up in the market as fast as it was originally estimated, but RTLS proved to be a valuable technology in its own right. Today, RTLS encompasses a large percentage of the total RFID business and is growing fast. There are many types of RTLS but the most popular is the scheme based on wireless Internet (WiFi), as it can use existing WiFi deployments, so it is fairly cost effective. Other types of RTLS schemes include ultra-wideband (UWB), UHF, GPS, GSM, and even ultrasound.

RTLS is not meant to be exact. Most systems have accuracies of around a meter, so it isn't well suited for motion capture. Its main applications are in tracking assets. Hospitals are one of the largest users of RTLS as there's a lot of portable equipment moving around and most of it is quite expensive.

As RTLS technology evolves, it may become viable for motion capture applications; however, the current markets are so large and undeveloped, that it isn't a priority to exploit the entertainment market as of yet.

Local Positioning System

Local positioning systems (LPS) are still in their infancy and not in use yet. These are systems that use RF to measure highly accurate positioning data at a very large sample rate. These systems will be viable for motion capture applications and will most likely become the state of the art in this field in the next few years.

I have spent most of my last few years working on the development of the first LPS system. Most RF positioning systems (GPS, RTLS) obtain their measurements by calculating the time it takes for signals to travel from a transmitter to a receiver. The speeds of these signals are so high that the accuracy needed for motion capture cannot be achieved, even if the signals are sent back to a transceiver. The most difficult problem to solve is in using the signals in a smart way to achieve the accuracy needed.

The LPS system that I've been working on already is capable of achieving measurement accuracy comparable to that of optical systems (1–2 mm). It still needs further development in order to be useable in production, such as miniaturization and automatic calibration, but the main point is that we now know that this is viable technology that will eventually become the motion capture standard.

LPS systems are similar to GPS, but work in an inverse way. The difference is that the object captured sends signals as opposed to receiving them. There is an array of antennas around the area to be captured. The moving objects are fitted with small transmitters that send hundreds of bursts per second. The antennas receive the signals and send them to a signal processing unit, which calculates the position, using the known location of each antenna, the signal travel time, and some additional characteristics of the signal in order to achieve the accuracy needed.

The current system is already capable of calculating the position measurements of up to 32 transmitters or tags 250 times per second. The system is capable of working in an area of about 25,000 m^3, which is about 20 times larger than the current capture volume of a typical optical system.

Figure 1.8 shows the tags and antennas developed for the Menache LPS prototype system.

The most interesting part of LPS systems is their ability to develop new markets and applications. Besides improving on the current uses in performance capture and life sciences research, LPS could enable the following applications:

- *Sports*—LPS systems could be used to generate sports analysis data and injury diagnosis at the actual time of the sporting

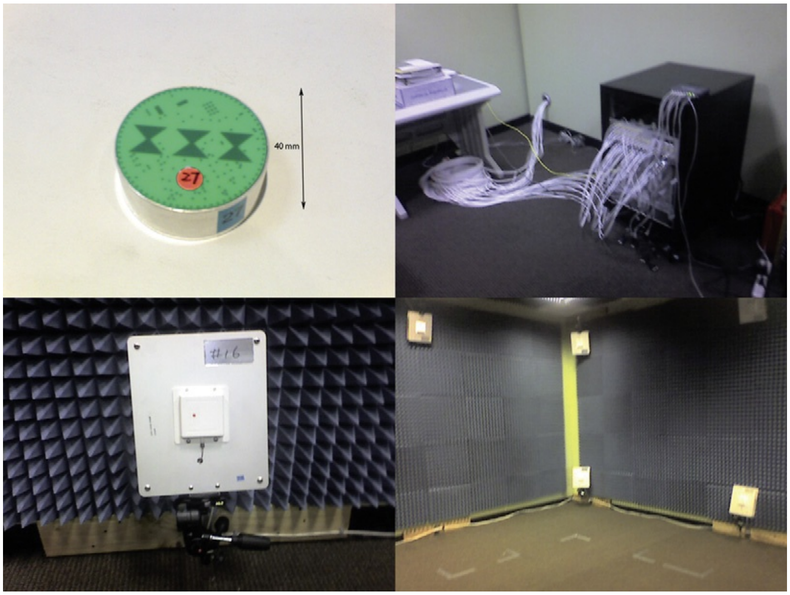

Figure 1.8 Menache LPS prototype.

event as opposed to the way it is done now, which is in a contained laboratory. Also, broadcast replay and analysis graphics could be generated in real time and used to aid in the chronicle of a sporting event. Figure 1.9 shows an example use of LPS in a soccer field. The proposed system would

Figure 1.9 An example of the use of an LPS system in a soccer match broadcast.

include 16 antennas placed around the stadium and a single tag on each player. In the future, multiple tags could be used and an accurate render of the event could also be produced.

- *Healthcare*—Some of the applications of LPS in healthcare are very interesting. Imagine being able to align or correct imagery for patient movement during an imaging session or even during surgery.
- *Gaming*—LPS systems could enable a new level of virtual reality that could be used at the arcade and at the consumer level. Since hardware could be inexpensive to manufacture in large volumes, an input device for home gaming devices could be developed. Also, location-based games such as Laser Tag could be transformed into a new kind of virtual or augmented reality experience.

Advantages of LPS Systems

- Tags transmit an ID pattern so no post identification is required.
- LPS systems calculate all accurate measurements in real time.
- LPS capture volumes are only limited by computing power.
- LPS can work outdoors as well as indoors.
- LPS tags can be worn under clothing.
- LPS systems will be cheaper to operate, as no postprocessing will be necessary.
- LPS systems will be cheaper to manufacture as no optical components are required.
- Automatic calibration will make these systems more portable and useable on film sets.
- LPS will be able to capture thousands of tags in real time.

Disadvantages of LPS systems

- LPS signals do not travel through metallic surfaces.
- Potential for RF interference. In its initial stages, LPS systems will require a characterization of the capture volume prior to use. Systems will be able to work using different frequencies to avoid this problem.

Electromagnetic Trackers

Electromagnetic motion capture systems are part of the six degrees of freedom electromagnetic measurement systems' family and consist of an array of receivers that measure their spatial relationship to a nearby transmitter. These receivers or sensors are placed on the body and are connected to an electronic control unit, in most cases by individual cables.

Also called *magnetic trackers*, these systems emerged from the technology used in military aircraft for helmet-mounted displays (HMDs). With HMDs, a pilot can acquire a target by locating it visually through a reticule located on the visor. A sensor on the helmet is used to track the pilot's head position and orientation.

A typical magnetic tracker consists of a transmitter, 11–18 sensors, an electronic control unit, and software. A state-of-the-art magnetic tracker can have up to 90 sensors and is capable of capturing up to 144 samples per second. The cost ranges from $5000 to $150,000, considerably less than optical systems.

The transmitter generates a low-frequency electromagnetic field that is detected by the receivers and input into an electronic control unit, where it is filtered and amplified. Then, it is sent to a central computer, where the software resolves each sensor's position in x, y, and z Cartesian coordinates and orientation (yaw, pitch, and roll).

Magnetic trackers such as the Flock of Birds by Ascension Technology Corporation use direct current (DC) electromagnetic fields, whereas others, such as the Polhemus Fastrak, use alternating current (AC) fields. Both these technologies have different problems associated with metallic conductivity. AC trackers are very sensitive to aluminum, copper, and carbon steel, but not as sensitive to stainless steel or iron, whereas DC trackers have problems with ferrous metals, such as iron, but not with aluminum and copper.

Many of these conductivity problems are caused by the induction of a current in the metal that creates a new electromagnetic field that interferes with the original field emitted by the tracker. These new fields are called *eddy currents*. Some magnetic trackers use special algorithms to compensate for these distortions by mapping the capture area, but these calibrations only work with static, predefined problem areas such as metallic structures in buildings. In most cases, it is better to avoid any high-conductivity metals near the capture area. This limitation makes the magnetic tracker difficult to transport to different stages or sets.

The latest use of magnetic systems is in consumer video gaming. Sixense Entertainment, a Los Gatos, CA–based company, is developing a wireless gaming controller (Figure 1.10) based on a low-power AC field that will be compatible with computer and console video games. Devices such as these will take the performance gaming trend started by the Nintendo Wii to the next level.

Magnetic trackers in the entertainment industry are used mostly for real-time applications such as live television, live performances, and location-based or Internet virtual reality implementations. Sometimes, they are used as part of puppeteering devices. A performer can author the body motions of a character with the magnetic tracker while someone else is performing the facial expressions and lip syncing using a face tracker or a data glove. At the same time, a puppeteer can be animating the character's eyes using a simple mouse. They are also used to obtain global positioning for other systems, such as inertial motion capture suits.

Figure 1.10 Sixense Truemotion Input Device. Photo courtesy of Sixense.

Advantages of Magnetic Trackers

- Real-time data output can provide immediate feedback.
- Position and orientation data is available without post-processing.
- Magnetic trackers are less expensive than optical systems, costing between $5000 and $150,000.
- The sensors are never occluded.
- It is possible to capture multiple performers interacting simultaneously with multiple setups.

Disadvantages of Magnetic Trackers

- The tracker's sensitivity to metal can result in irregular output.
- Performers are constrained by cables in some cases.
- Magnetic trackers have a lower sampling rate than some optical or LPS systems.
- The capture area is smaller than is possible with other systems.
- It is difficult to change marker configurations.

Electromechanical Performance Capture Suits

Electromechanical performance capture suits have been around for a while. They are inside-in systems based on a group of structures linked by potentiometers, gyroscopes, or similar angular measurement devices located at the major human joint locations. The newer versions use MEMS (Microelectromechanical Systems) inertial sensors placed over a Lycra or spandex suit. The idea is that the suit measures all human limb rotations.

Potentiometers are components that have been used for many years in the electronics industry, in applications such as volume controls on old radios. A slider moving along a resistor element in the potentiometer produces a variable voltage-potential reading, depending on what percentage of the total resistance is applied to the input voltage.

MEMS inertial sensors are also angular measurement devices, but they are much smaller and more accurate. They are basically integrated circuits combined with micromachined moving parts. Good examples of devices that use MEMS technology are the Nintendo Wii input device and the iPhone.

One big drawback of performance capture suits is their inability to measure global translations. In most cases, an electromagnetic or ultrasound sensor is added to the configuration to solve this problem, but that subjects the setup to the same disadvantages as the electromagnetic or ultrasonic systems, such as sensitivity to nearby metals or bad accuracy and drift. In addition, the design of most of these devices is based on the assumption that most human bones are connected by simple hinge joints, so they don't account for nonstandard rotations that are common to human joints, such as in the shoulder complex or the lower arm. Of course, this can actually be a benefit if the mechanical setup of a particular digital character calls for such types of constraints.

A good example of a performance capture suit is the Xsens MVN, shown in Figure 1.11. This suit uses inertial sensors based on MEMS technology.

Figure 1.11 The MVN Inertial Suit by Xsens. Photo courtesy of Xsens Technologies.

Advantages of Electromechanical Body Suits

- The range of capture can be very large.
- Electromechanical suits are less expensive than optical and magnetic systems.
- The suit is portable.
- Real-time data collection is possible.
- Data is inexpensive to capture.
- The sensors are never occluded.
- It is possible to capture multiple performers simultaneously with multiple setups.

Disadvantages of Electromechanical Body Suits

- The systems have a low sampling rate.
- They can be obtrusive due to the amount of hardware.
- The systems apply constraints on human joints.
- The configuration of sensors is fixed.
- Most systems do not calculate global translations.

Digital Armatures

Digital armatures can be classified into two types: (1) keyframing or stop-motion armatures and (2) real-time or puppeteering armatures. Like the mechanical suit, both types consist of a series of rigid modules connected by joints whose rotations are measured by potentiometers or angular sensors. The sensors are usually analog devices, but they are called *digital* because the resulting readings are converted to digital signals to be processed by the computer system. These armatures are typically modular in order to accommodate different character designs.

Keyframing armatures were initially used to help stop-motion animators animate digital characters; they are not really considered motion capture systems because they are not driven by a live performer. I mention them because most commercially available armatures can also be used as real-time armatures. Some proprietary armatures are dual purpose such as a device initially called the *Dinosaur Input Device* (DID), which was devised by Craig Hayes at Tippett Studio in Berkeley, California. The name was conceived because this unit was used to animate some of the digital dinosaurs in *Jurassic Park*. Later, the device was used to animate the bugs in *Starship Troopers* and it was called the Digital Input Device (Figure 1.12).

The basic concept behind keyframing armatures is that the animator poses the device manually to generate each keyframe in the animation. The character in the animation software is set up with a mechanical structure equivalent to the armature's. By pressing a key, the animator uses the computer to record the armature's pose into the digital character for a particular frame in time. This is done via a driver program that connects the device, typically plugged in to a serial port, to the animation software. Once all the key poses are recorded, the software treats them as regular keyframes that might have been created with the software by itself.

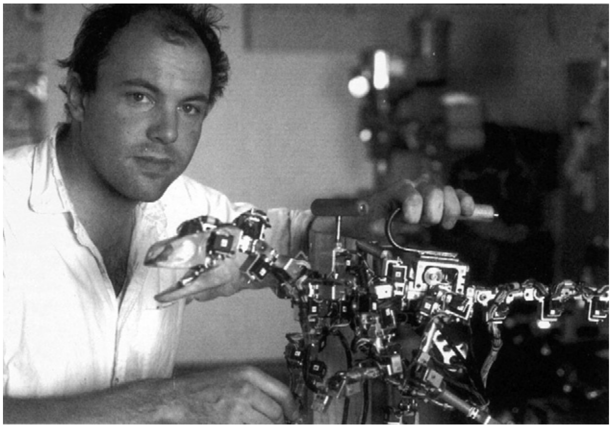

Figure 1.12 Digital Input Device (D.I.D) designer Craig Hayes and the Raptor D.I.D. puppet. *Jurassic Park* 1992-1993. Photo courtesy of Tippett Studio.

Puppeteering armatures are very similar to keyframing armatures, except the motion is captured in real time as performed by one or more puppeteers. An example of such a setup is the proprietary armature developed by Boss Film Studios to capture the motion of Sil, the alien character that H.R. Giger designed for the film *Species*.

Digital armatures are not very popular any more. The last commercially available example of a digital armature was the Monkey 2 by Digital Image Design in New York City (Figure 1.13). This unit could be used as both a keyframe and real-time armature. It was modular, so it could be assembled in different joint configurations. The first-generation Monkey had a fixed configuration, which made it unusable for any nonhumanoid applications. The typical cost for a 39-joint Monkey 2 setup was approximately $15,000, which included all the necessary parts as well as driver software for most well-known animation packages.

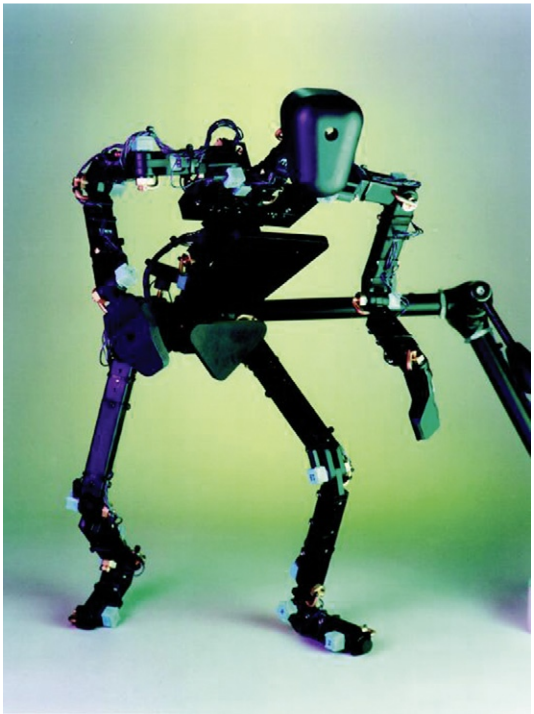

Figure 1.13 The Monkey 2 armature. Photo courtesy of Digital Image Design.

Facial Motion Capture Systems

There are several mechanical and optical devices for capturing facial motion. The most popular are the real-time optical face trackers, consisting of a camera that is placed in a structure attached to the performer's head so that it moves with the performer. The device captures the motion of small markers placed in different areas of the face. Unfortunately, these are 2D devices that cannot capture certain motions such as puckering of the lips, so the data is all in one plane and not very realistic. Three-dimensional facial motion data can be captured with an optical system using two or more cameras, yielding a much better result, but not in real time.

There are also newer optical systems that don't require markers, but are based on optical flow analysis. Such a system was used for the facial performance capture for the film *Avatar*.

The state of the art in facial motion capture is based on a combination of optical capture and software algorithms. The three major types of modern software algorithms to process facial optical data are muscle/skin simulators, dynamic blend shape solvers, and photogrammetry builders. There can also be combinations of two or three of these techniques. Most such algorithms exist as proprietary developments of visual effects companies, such as the muscle simulation software Sony Pictures Imageworks developed for films like *The Polar Express* or the dynamic blend shape solver used for *Avatar*, developed by Weta in New Zealand.

Facial Muscle and Skin Simulators

Facial muscle and skin simulators are algorithms that take optical data of the face and use it to solve facial poses based on anatomical rules and restrictions. This is a very good method to animate human faces, as it is based on real anatomy.

Initially, a model of the animated performer's face is built. The internal structure of the skull, jaw, and major facial muscles is also defined. It is ideal to have an MRI or a similar imagery of the performer to determine the exact location of the jaw. Most of the time, it is impossible to get such imagery, so the performer is asked to capture a series of expressions that are used to determine the muscle positions, movement, and jaw joint. We call this a *training set*.

The training set is used by the software to define what each muscle can and cannot do. The skull model serves as a collision object. The performance data is processed based on these restrictions, taking into account the dynamics of the skin.

Once the performance data is processed, the resulting stream is a series of normalized values per muscle. This means that a muscle in its relaxed state will have a value of zero, fully extended will have a value of one, and fully compressed minus one. The greatest benefit of using such a system is that it is extremely easy to apply this data to different human faces with different muscle layouts or limits.

Dynamic Blend Shapes

Blend shapes are probably the most popular method for animating facial performances. The method entails building a series of models of the character's face in different expressions and phonemes, which are the various mouth poses used for speech. The animator uses sliders to blend these expressions and create a facial pose.

Depending on the number of poses, this method can be very expensive to set up. The benefit is that it can be used to animate any kind of character, including humans.

It is possible to drive these blend shapes using facial motion capture data. The process starts with deciding on a list of shapes. These can be divided by phonemes, emotions, areas of the face, etc. There are a few popular lists of expressions that may work better depending on the nature of the character. A well-known system to categorize facial expressions is the Facial Action Coding System or FACS. The FACS research was first published in 1978 by Paul Ekman and Wallace Friesen, and it is widely used in the field of psychology to help in the interpretation of emotions. The list of shapes for a human face could be a combination of FACS units and phonemes.

Once the list is defined, it is necessary to capture the corresponding data. Each of the facial expressions in the list needs to be associated with a real expression from the performer, so regardless of the method used for facial motion capture, a dataset with every one of the expressions in the list needs to be created. This is similar to the training set used for facial muscle and skin simulation. Also, the corresponding digital model for each of the expressions needs to exist, and this is where the method can get really expensive.

Once all the assets are in place, there needs to be an algorithm that will convert an actual performance into a stream of shapes. The algorithm has to find the best fit of shape combinations; that is, the combination of blend shapes that will look closest to the performer's expression at that point in time. This algorithm can be a brute-force recursive algorithm or one of a few existing "approximation algorithms" that already can solve the "best fit" problem, such as the "least squares" method or "Kalman filtering."

Obviously, the more expressions and models are predefined, the better is the end result.

Computer Vision

Photogrammetry is the method of building geometry based on photographic images. It has been around for a long time and has been used for many applications. The first application of

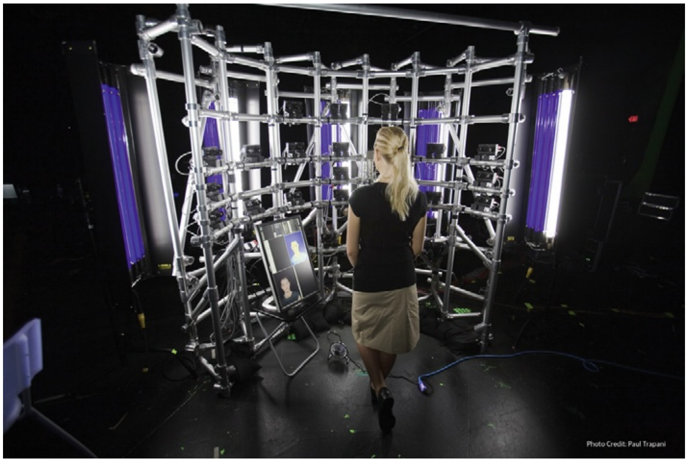

Figure 1.14 Mova's array of special fluorescent lights and synchronized cameras. Photo courtesy of Mova.

photogrammetry in 3D animation was to create computer models that would be animated later in some other way. Capturing facial motion using photogrammetry is a relatively new application, as it requires significant computing power. It is called *videogrammetry* because it uses a set of video images instead of photographs. Further developments are labeled as *computer vision* methods, as they not only use the discrete video images, but also an analysis of their changes frame after frame to achieve the optimal result.

The idea is that a model is created dynamically for each frame of the performance. This model is used directly as the animated face or to extract data to drive another character model.

There are a few companies that specialize in computer vision facial performance capture. The leading ones are US-based Mova and UK-based Image Metrics. Some films that have ventured into the use of this technique are *The Incredible Hulk* and *The Curious Case of Benjamin Button*. Figure 1.14 shows Mova's array of special fluorescent lights and synchronized cameras.

Other Motion Capture Systems

The Waldo

The Waldo is a telemetric device that imitates the motions of whatever it is controlling. It was named after Waldo Farthingwaite Jones, a fictional disabled scientist in Robert A. Heinlein's 1942 short story "Waldo" who invents a device that allows him to act beyond his biomechanical possibilities. Telemetry is by definition the transmission of measurements; consequently, the Waldo transmits measurements of the motion of joints and other body parts to a mimicking device. There are many kinds of Waldos, controlled by different body parts, such as the head, arms, or legs, or even the body as a whole. Their main applications in entertainment are the control of digital or animatronic characters.

The name "Waldo" is rumored to have been first used at NASA to name an early telemetry device, and it was used for years as a generic name for such machines. It is a trademark of

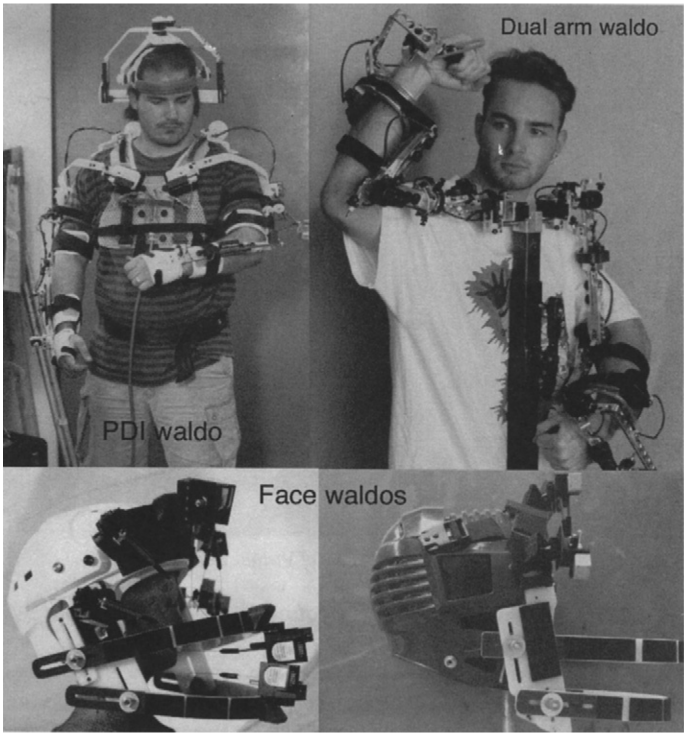

Figure 1.15 Different types of Waldos manufactured by The Character Shop. Photos courtesy of The Character Shop.

The Character Shop, a company that specializes in animatronic characters. Figure 1.15 shows different kinds of Waldos manufactured by The Character Shop. The Jim Henson Creature Shop uses a device of this genre called the *Henson Performance Control System* to bring their animatronic or digital creatures to life.

Hand Trackers

To capture the motion of small body parts, specialized trackers are needed because full-body tracking devices lack the necessary resolution. For example, it is practically impossible to track facial movements and movement of fingers with an electromagnetic tracker because the sensors are bigger than the locations where they need to be placed. It is possible to handle such situations with optical trackers in some cases.

For hand motions, there are several types of gloves that have small form-factor technologies, such as the very expensive Cyberglove II or the more reasonable 5DT Data Glove. One of the first commercially available hand trackers was the VPL

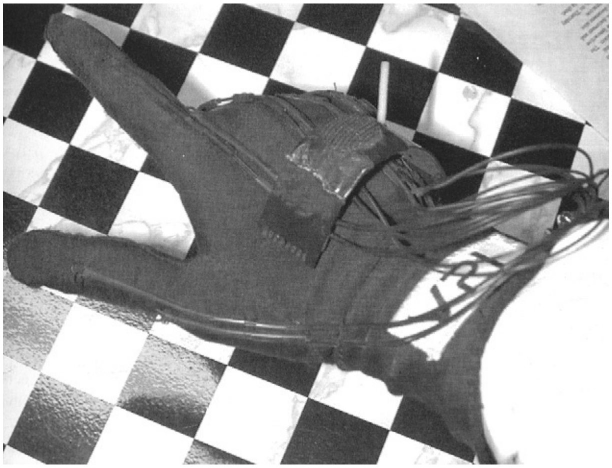

Figure 1.16 The VPL DataGlove.
Photo courtesy of Zak Zaidman.

DataGlove, shown in Figure 1.16, which was released in 1987. It was manufactured by VPL Research, a company founded by Jaron Lanier, a pioneer in the field of virtual reality, but the company ran into financial trouble and its technical assets and patents were acquired by Sun Microsystems in February 1998. The glove was based on fiber-optic sensors placed along the back of the fingers. As fingers rotated, the fibers were bent and their transmitted light was attenuated. The strength of the light was turned into a signal that could be measured by the processor to calculate the rotation of the fingers. Most DataGlove models had 10 sensors that measured the rotations of each of the two upper joints of the fingers. Some versions also had measurements for abduction (the angle between adjacent fingers). It could measure a minimum of 1° at up to 60 samples per second. The average cost was $8000.

Another historically significant hand tracker was the Mattel PowerGlove, introduced in 1989 as an add-on controller for the Nintendo Entertainment System. It was conceived at VPL Research as a low-cost alternative to the DataGlove, called *Z-Glove*. VPL Research licensed it to Abrams Gentile Entertainment (AGE), the company that manufactured it for Mattel. The PowerGlove sold for about $100.

The difference that made the PowerGlove's cost so much lower than the DataGlove's was in the components. The optic fibers were replaced with less expensive conductive ink that was used as a resistor whose impedance variations indicated degrees of flexion. The global position and orientation were measured via ultrasonic pulses that were emitted from a few locations on the glove. The time taken for these pulses to reach the receivers was used to calculate in a very loose way the position and orientation of the glove. In fact, the glove emitters always had to be pointing at the receivers, which limited its range of motion. The PowerGlove only measured flexion of the thumb and three fingers, at a much lower resolution than the DataGlove.

The PowerGlove was a commercial failure, but it eventually became very popular as an inexpensive virtual reality input device. Since it was conceived as a peripheral device for the

Nintendo game console, it had no standard interface. Thus, there was no easy way to connect it to anything else. When the popularity of the device became apparent, interfaces to use it with an IBM-compatible PC or an Apple Macintosh started to surface, either released by AGE, third-party manufacturers, or people experimenting with the device.

Figure 1.17 The Cyberglove II. Photo courtesy of Cyberglove Systems LLC.

The Cyberglove II (Figure 1.17) is manufactured by Cyberglove Systems, LLC. The first Cyberglove was launched in 1991 by Virtual Technologies, Inc. (VTI), based on their patented piezo-resistive bend-sensing technology. VTI was acquired by Immersion Corporation in 2000. Immersion sold the Cyberglove product line in 2009 and that is how Cyberglove Systems was born.

The Cyberglove II is wireless via Bluetooth and is available in 18- and 22-sensor models. The 18-sensor model measures most finger rotations, abduction, thumb crossover, palm arch, and wrist rotations. The 22-sensor model adds a sensor for the distal joint of the index, middle, ring, and pinkie fingers. The sensors can capture a minimum rotation of less than one degree and can work at up to about 90 samples per second. Like many other similar devices, it requires a third-party device in order to measure global positioning and orientation. Depending on its configuration, each glove can cost many thousands of dollars.

Applications of Motion Capture

Most motion capture equipment was originally developed with applications other than entertainment in mind. Such devices have been used for many years before becoming viable tools for 3D computer graphics.

The main markets that currently benefit from motion capture are medicine, sports, entertainment, and law, but there are smaller markets that are also taking advantage of the technology. For example, motion capture equipment is used to help design ergonomic environments. Also, it is used for automobile safety tests: The motion of crash test dummies is captured and analyzed.

Medicine

In clinical circles, motion capture is called 3D *biological measuring* or 3D *motion analysis*. It is used to generate biomechanical data to be used for gait analysis and several orthopedic applications, such as joint mechanics, analysis of the spine, prosthetic design, and sports medicine. It has always been the largest segment user of the technology, but entertainment uses are growing faster.

There have been a great number of studies of gait performed with patients of all ages and conditions. The first ones were made by Etienne Jules Marey and Eadweard Muybridge in the late 1800s using photographic equipment.

Muybridge's studies started when he was hired by Leland Stanford, governor of California, to study the movement of his race horses in order to prove that all four hooves left the ground simultaneously at a given point during gallop. In 1876, Muybridge succeeded by photographing a galloping horse using 24 cameras (Figure 1.18). He then continued his studies of human and animal motion for many years. His paper on the subject, "Animal Locomotion," was published in 1884 and is still one of the most complete studies in the area.

A professor of natural history, Etienne Marey used only one camera to study movement, as opposed to the multiple-camera configuration used by Muybridge. Even though they met in 1881, Marey and Muybridge followed separate paths in their research. Studies continued in this fashion for a century, but until the introduction of optical motion capture systems in the 1970s, the research yielded almost no benefits.

Of the current life sciences' applications, gait analysis is very useful because it accurately separates all the different mechanisms that are used during the multiple phases of a walk cycle in a way that makes it easy to detect certain abnormalities and changes. For example, gait analysis helps to measure any degree of change in conditions such as arthritis or strokes. It is also used along with other tests to determine treatment for certain pathological

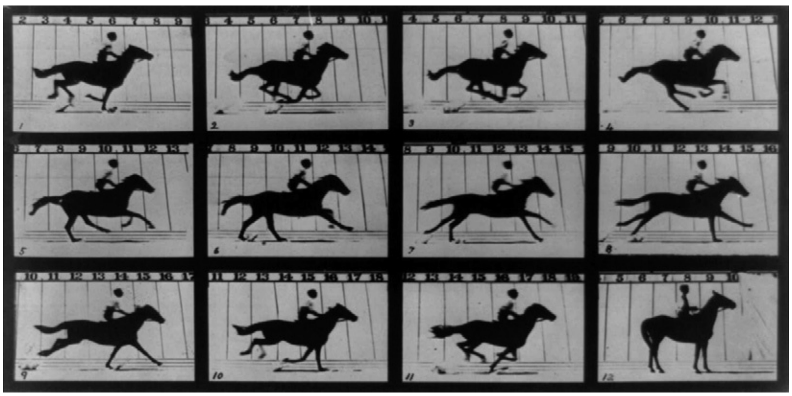

Figure 1.18 Muybridge's galloping horse photographs. Photo courtesy of Eadweard Muybridge Collection, Kingston Museum.

conditions that affect the way we walk, such as cerebral palsy. Rehabilitation by gait training is used for patients with pelvis, knee, or ankle problems.

There are many other potential applications for motion capture in the medical field. Once the current systems' limitations are surpassed, it will be possible to use motion capture to align patients and imagery in operating rooms, to compensate patient movements during MRI sessions, or even to make widely available the current applications that are possible only in special labs. The requirement will be a real-time motion capture system with great accuracy that can function in any environment.

Sports

Sports analysis is a major application of motion capture. 3D data is being used extensively to improve the performance of athletes in sports such as golf, tennis, gymnastics, and even swimming by studying the performance of professionals and dissecting and classifying the different components of the movement. As motion capture technology improves to the point at which undisturbed data collection is possible, the potential uses will become even greater in this field.

There are a few motion capture studios across the country dedicated exclusively to the analysis of sports, especially golf. For a few hundred dollars, any golfer can have his or her swing analyzed or compared with the swing of a professional golfer. Visualization software allows the studios to study the athlete's motions to find any problem areas.

The market for sports analysis is mostly amateur sports enthusiasts, but professionals look for help as well. Biovision, a company that used to operate three sports analysis studios across the country, claimed to have helped professional golfer John Daly to find a problem related to opposite movements of his hips and shoulders that happened for one-tenth of a second during his swing. Tiger Woods has done his share of motion-captured golfing for all his PGA video games. His swing has been analyzed and broken down to the smallest component in order to teach others what a perfect swing must look like.

The benefit of using motion capture rather than video for this kind of sports application is that motion capture yields a 3D representation of the motion that can be examined from any possible angle. This is true even if the video could be recorded at higher speed than the normal 30 frames per second. To analyze a fast sports action such as a golf swing or a baseball pitch, it is required to have at least a few hundred frames or samples per second.

The future of motion capture in sports is very bright. Newer systems will be able to carry out sports analysis outside of the lab. All track and field events will be analyzed at the track, its actual venue, and so will all other sports. In addition, the data captured will have many additional uses, such as real-time integration into video games via gaming networks such as X-Box live. It will be possible to watch an auto race from any desired camera in real time rendered on the gaming console and introduce a virtual car to participate in the race remotely.

Broadcasters are always looking for ways to enhance the sports watching experience. Having accurate position data of athletes in any sport is a very attractive prospect for generating eye-catching graphics and previously unavailable statistics. Once a game of football can be captured in real time, we will be able to watch rendered instant replays from any angle. The data will be used as an ongoing biomechanical profile for each player. It could even be used to diagnose injuries in real time.

Figure 1.19 shows how motion capture data could be used to enhance the broadcast of a soccer match. For the application in the image only one captured marker is required per player and that makes this application one of the closest ones to being possible. There are a few systems that analyze video and deliver motion data of such events, but technology is still not up to par for real-time applications. Instead, most current systems are mostly used to calculate estimated speed and total distance covered by each player. In a year or two, this is the way we'll be starting to look at broadcast sports.

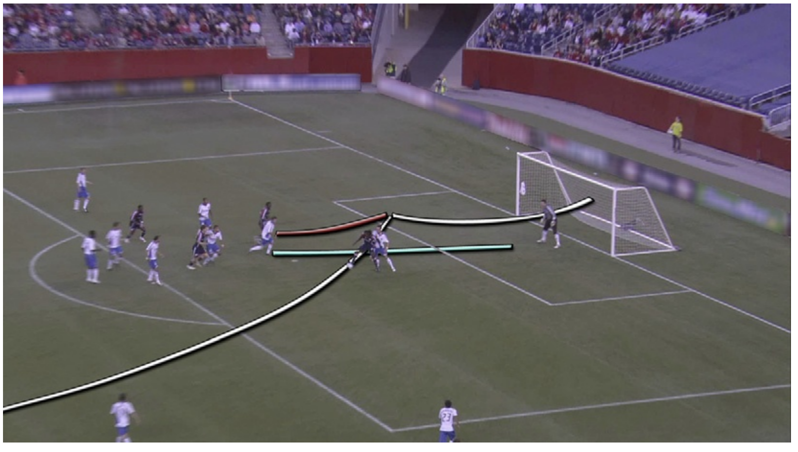

Figure 1.19 Motion capture will be used to generate graphics to enhance any sporting event broadcast.

Entertainment

The entertainment industry is the fastest growing market segment for motion capture; although not yet the largest, it certainly is the segment with the highest profile. Of the different entertainment applications, the use of motion capture in video games is currently the most widespread and the best accepted and understood. Television and feature film applications of motion capture are still migrating from an experimental market to a viable medium.

Video Games

Even though the use of motion capture in entertainment started in television and film, those markets remain resistant to widespread use of the tool. The video game industry was the first segment of entertainment to embrace motion capture as a viable tool for character motion, whereas motion capture is used on almost every video game that involves human motion. It started doing so at a time when the motion capture hardware and software was still producing low-quality results at a high, but not extreme cost. At the time, video games didn't require or rather couldn't handle high-quality motion data because they lacked the memory and rendering power to create complicated graphics, so the two industries evolved together and as rendering power increased, so did the camera resolution needed to produce the better motion quality. Video games today can handle just as high-resolution motion data as feature films.

While applications in video gaming are mostly in game production, actual performance gaming is growing fast. In October 2003, Sony released the Eye Toy, a camera to accompany the PlayStation 2. It came with a few games driven by the players' motions. The games were quite basic and weren't very compelling, but the device found a small market and over the years has been evolving, selling more than 10 million units to date. The Eye Toy is still a single camera that can capture 2D motion under certain circumstances, so games still need to be developed expressly for the device in order to work properly. Until true 3D data can be achieved, players won't be able to use their performances to power their current favorite games that are not designed for a particular input device.

Nintendo released the Wii at the end of 2006 and by the end of 2009 had sold close to 70 million units, making it the undisputed leader in console gaming. This is relevant because the Wii is designed for performance gaming and its players are willing to give up certain performance and graphic capabilities in favor of better immersion. The included input device contains an accelerometer and an infrared optical sensor that enables it to calculate an estimate of the player's movements. The device does not exactly capture the position of the player, but just the motion. The difference is that it won't tell you where it is in space at any given point in time, but it will tell you where it's going and how fast. In other words, the motion captured by the Wii Remote is relative to an arbitrary coordinate system that is controlled by the game. The Eye Toy has a fixed coordinate system but it's only 2D, while the Wii Remote has a 3D arbitrary coordinate system. Both systems' restrictions make it necessary to develop games that can operate within their capabilities.

In reaction to Wii's success, Microsoft and Sony have been busy developing their own performance gaming devices. Microsoft has demonstrated its Project Natal, which is an optical-based system. The device contains an RBG camera and a depth sensor, which is a monochrome camera working with an infrared projector to calculate 3D depth. Microsoft claims that the device will be capable of capturing movement, orientation, and gestures, but no specifications as far as accuracy, speed, or latency are available as of yet.

Sony has demonstrated its PlayStation Move device, which is a motion-sensing wand not unlike the Wii Remote, except it also uses their existing Eye Toy technology to enhance the motion complexity capabilities of the device.

Television

The use of performance animation in animated TV shows is a small market that had been growing slowly but has met with several obstacles. The main problem is that motion capture has been used primarily as a cost-cutting tool, resulting in an unexpected quality trade-off. Shows that used performance animation in the past have usually not included a budget for modifying the animation beyond the performance. An example of a show with this problem was *The Real Adventures of Jonny Quest*, in which only a few minutes per show contain any kind of computer animation.

Other cartoon shows have been successful in using performance animation in their entirety. Most of these shows have been produced by the company that is actually collecting the motion, and the motion is captured in real time in most cases, so there are no surprises in what the quality of the final animation will be. Companies that have succeeded in producing these kinds of shows are Medialab Technology, with Donkey Kong Country, and Modern Cartoons, with Jay-Jay the Jet Plane.

In television, commercials and music videos remain a small but consistent market as their budgets are larger than those of cartoon shows. Music videos have benefited from motion capture since the early 1990s, when Homer and Associates produced computer graphics based on performance animation for music videos by Peter Gabriel, Vince Neil, and TLC.

Feature Films

The main applications of motion capture in feature films used to be digital extras, digital stunts, and digital crowds, but as technology advances, main characters have been created using performance capture. Digital extras are background characters that for one reason or another have to be digitally generated. An example of such a case is a shot of the ship in the feature film *Titanic* as it leaves port. A live shot like this would be filmed with a helicopter and a lot of extras walking on deck, but since the actual ship did not exist, a miniature ship was shot, using a motion control camera. Because of the size of the shot, filming real people in green screen to composite over the ship would have been impossible, so the only solution was to create digital extras.

Digital stunts are actions that are either not humanly possible or need to be seen on screen performed by a person other than the stunt person. After a stunt is captured, it can only be enhanced to a certain level before the realistic motion is lost; thus, if a stunt is not humanly possible, it is best to hand it to a good character animator who specializes in realistic motion. Another kind of digital stunt involves applying motion data from a stunt person to a digital version of a known actor.

Motion capture is a good money-saving tool to use for the digital crowd scenes in which no interactions between characters occur. A perfect scenario would be a crowded stadium where some kind of sport is being played. A computer-generated crowd could be procedurally animated by first capturing different versions of each crowd reaction. As the game proceeds, the crowd may go into a cheering frenzy or be bored or angry. When one of the teams playing the sport scores, part of the crowd will be angry, others will be excited, and some will be ambivalent. A percentage of the crowd would be animated by using randomly selected angry cycles, another group would be animated by the cheering cycles, and the rest by the ambivalent cycles.

Animated films using motion capture have been also popular. The film *Happy Feet* was produced in Australia and went on to win an Academy Award for Best Animated Feature. Another example of an animated feature using motion capture is *Monster House*, produced by Robert Zemeckis.

The latest and, by far, most popular use of performance capture in any medium is for main characters and creatures. Films like *The Hulk*, *The Lord of the Rings* trilogy, *King Kong*, and the *Pirates of the Caribbean* series all used it for main characters.

Films like *The Polar Express*, where seven digital characters were based on Tom Hanks' performance, broke ground for what is now known as *motion-captured films*. The latest and most successful to date is James Cameron's *Avatar*. Budgets for these films are still extremely high; this is why only directors like Robert Zemeckis and James Cameron have been able to experiment in this medium. Eventually, as technology improves, budgets will become more accessible.

Law

Motion capture is applied in law to produce reconstructive videos of events. These videos are used as evidence in trials to aid the jury in understanding a witness's opinion about a particular order of events. According to a study conducted by the American Bar Association, this kind of evidence is much more effective with jurors than any other demonstrative evidence.

An example of an effective use of motion capture in the legal context is the video produced by Failure Analysis to recreate the events of the murder of Nicole Brown Simpson and Ronald Goldman during the O.J. Simpson trial. The video wasn't actually used as evidence in the trial, but it serves as an example of the kind of animation that could be used in a court of law.

To be admissible, the animation must be very plain, without any complex lighting, texture mapping, or detailed models, and, like any other evidence, it must meet the local and federal rules of evidence. In most cases, it has to be accompanied by supporting testimony from the animation creators, during either trial or deposition. Many animation companies specialize in all aspects of evidence animation, from its preparation to the presentation in court.

Security and Defense

Motion capture has been used in the military for many years, mainly in tracking helmets for pilots and other operators of combat vehicles. Tracking the line of sight of pilots has proven to be a great way to help aim weapons and to receive critical visual information.

Other uses in this field include troop training aid through simulation. Currently, it is mostly used for production, but eventually will be widely used to acquire exact troop position in the virtual or real training field. Positioning of troops and equipment in large area exercises will allow the instructors to evaluate everything that happens in the exercise by looking at a large real-time computer-generated visualization. Troops then will be debriefed using the 3D created simulation, so they can learn where to improve.

In secured installations or amusement parks, personnel and visitors can carry a tag at all times, allowing the tracking of all their movements in real time. Tagging will allow the system to provide entry privileges to certain areas and will also keep a history of personnel and visitor flow in the facility. Certain items and equipment can also be tagged in order to keep real-time account of its position at all times. There are already some GPS and RTLS versions of this application. Legoland in Denmark has had an RTLS child-tracking system for several years now. For child safety at the mall or anywhere else, parents can track their children by having them carry a mobile phone that they can track online.

Granted this is not performance capture, but it is capture of motion and it is relevant because it potentially is a much larger market than any other that exists today, even in entertainment. In fact, asset tracking is already a fast-growing market in hospitals and other facilities using RTLS.

Engineering

Motion capture is used in the engineering process for design, safety, and ergonomic testing. Through simulated human interaction with engineering models, engineers determine the functionality of their design prior to building expensive prototypes. Also, through better enabling augmented reality technology, motion capture will open the door to more efficient work flow and training methods.

Ford Motor Company has developed a system called *Human Occupant Package Simulator* (HOPS). The system is basically a large database of captured movements of drivers and passengers inside vehicles. Ford designers use digital humans driven by these motion datasets inside virtual vehicle designs to analyze their interactions with the vehicles. This helps them improve the ergonomics of their designs as much as possible before having to build models and prototypes.

Future Applications

There are many applications that do not currently utilize motion capture yet due to the limitations of existing systems. These markets include the social networking, location-based gaming, engineering, and sports broadcast industries.

Potentially, a location-based game such as laser tag could be carried out in a virtual world. Players would wear motion capture tags and a wireless stereoscopic head-mounted display. The virtual world's obstacles would match the real venue's, but all the decorations would be very different and the players' digital counterparts could take many different creature shapes. Weapons would also be greatly enhanced, resulting in a very realistic and immersive experience. An application like this is exactly what for many years has been the promise of what virtual reality is supposed to feel like. Players wouldn't even have to be in the same physical venue to participate. In the same manner, social networking could evolve into virtual reality applications where people meet using their chosen avatar without leaving their physical home.

Another interesting future application is in machine and vehicle service and maintenance. Imagine a head-mounted display capable of guiding a technician to a problem area inside an aircraft, showing internal parts based on actual CAD models. By tracking the display with great accuracy, the system will match the position of the real and virtual worlds, and it would make it very easy to find an exact location of a problem inside a very complex machine. It would be exactly like looking through walls or the fuselage of a plane.

Sports broadcasting will be revolutionized by newer positioning technologies. Players' performances will be captured in real time, allowing for instant replays from any angle and even real-time injury diagnosis. The data generated will be also used to create previously unavailable statistics, much loved by sports fans, and to make sure referees make fewer mistakes. Eventually, data from the game could be sent to video game consoles to allow home players to participate in some way in the event, for example, by driving a virtual car around a racetrack at the same time as a real race is happening.

It is clear that knowing where things are and how they move is a big part of the future. It only takes a little imagination to come up with new and exciting ways to use motion capture once it is unrestricted and cost effective.

2

MOTION CAPTURE CASE STUDIES AND CONTROVERSY

CHAPTER OUTLINE

Motion capture, performance capture, Satan's rotoscope—call it what you may, it is a technology that rose to the top of the hype machine among entertainment industry watchers, and then became a topic of major controversy for various reasons.

The first of many problems was the initial perception of what could be done with motion capture. Years ago, when the technology was first used to generate animated characters in three dimensions, the results amazed most people because nobody had ever seen realistic human motion used for 3D character animation. Many people in the industry thought they had found the replacement for the traditional character animator. However, because the captured motion data available in those days was very noisy and expensive, it quickly became clear that it wasn't going to replace manual labor any time soon. In addition, off-the-shelf computer animation software with user-friendly interfaces such as Maya started to emerge, which helped more artistic and less technically oriented people enter the business. Character animators started to work in three dimensions.

Understanding Motion Capture for Computer Animation.

In the late 1980s, the concept of the motion capture service bureau was born. Equipped with the latest in optical and electromagnetic technology, these studios detected the need for cheaper methods of character animation. The video game industry quickly became the first entertainment unit to sign up, finding the data fast and cost effective. In fact, video game developers are still the number one consumers of optical motion capture services, with about 70% of the total usage. Larger video game publishers, such as Electronic Arts, have in-house motion capture studios that their game developers use, but smaller developers still use the service bureaus.

The next logical market for service bureaus to pursue was film and television, but there was one problem: The applications were very different from video games. Video game characters were extremely low detail, and the motion data didn't need to be very clean. Most characters didn't even need to stay firmly on the ground. Also, most service bureaus didn't have anyone familiar with computer animation or even performance animation, and motion capture equipment manufacturers had even less knowledge of the subject, especially since their main market wasn't entertainment. This lack of knowledge combined with the desire to break into the entertainment industry drove manufacturers and service providers to make the same outrageous claims that had been made years ago: "Motion capture can replace character animators." This time around, the technology was more affordable and readily available for purchase, so the studios and the press started listening. The media still loves to show people geared up with markers, especially if it is a celebrity performing for a digital character. On almost every motion capture job that I have worked on, there has been a "behind the scenes" crew videotaping the session. Motion capture would be the headline even if it wasn't ultimately used on the project. Many media reports on the subject also claimed that the technology would eventually replace traditional animators.

Some studio executives who got wind of the "new" technology through the media decided to study the possibility of integrating it into their special effects and CG animation projects. Most of these studies consisted of having some data captured by a motion capture studio and then testing it on a particular character. The results were misleading, because the studies didn't really include what the motion capture people went through to clean up the data and apply it to the character. They only considered cost, but didn't take into account that motion capture studios were often discounting the price of the data in order to get in the door. In addition, the motions studied were usually walk cycles and other simple moves that are less complicated to process than the average movements for a typical film or game project.

This motion capture frenzy was further fueled by the success of *Toy Story* as the first CG animated feature-length film. Performance animation was seen as a way to achieve the same success at a fraction of the cost. It was also heralded as the key to cost-effective CG television cartoons. At TSi, there was a period when studios would show up and ask us to bid on capturing data for full-length feature film projects or television shows without even having a specific project yet. It really didn't matter what the content was, as long as it was motion captured. In the studios' views, such a project would cost less and guarantee great press exposure, especially if an actual movie star performed the character.

Project-driven purchases of expensive motion capture equipment happened in many studios, such as Warner Brothers, DreamWorks, and smaller studios. The equipment in almost every case did not end up being used for the project for which it was intended, resulting in budget and creative problems and sometimes even the cancellation of projects. These studios were led to believe that any character-animated shot could be achieved with motion capture at a lower cost, even those with cartoon characters and requiring stylized motion and complex interactions with objects and other characters. In many cases, character animators were handed captured motion data and asked to modify it, a big mistake in my opinion, which only served to generate more heat against the technology.

For many years, motion capture couldn't find a place in the film industry. Some studios used it to create digital stunts and some background character work, but the technology had an overall bad reputation in the industry. It wasn't until 2004, when Robert Zemeckis released *The Polar Express* that motion capture started to find its place in the film industry, although not without some new controversy.

The Screen Actors Guild first noticed performance capture when James Cameron released *Titanic* in 1997. They started to be concerned about it during the shooting of *The Polar Express* and that concern has grown with the proliferation of performance capture films to the point that SAG has recently formed a national Performance Capture Committee. The guild will soon try to renegotiate its master contract with producers to cover performance capture under the same terms as other acting work.

Furthermore, none of the actors on *Avatar* was nominated for an Oscar, even though the film broke all box-office records and was nominated for Academy Awards in most categories. And to make things even more interesting, the Academy has recently ruled that performance capture films can no longer be considered for Academy Awards in the Best Animated Feature category.

Regardless of who authors or owns the performance and whether performance capture should be treated as any other acting job, there is still the issue of look and appropriateness. What kinds of characters should be performed versus animated? Is it photo-real or stylized humans? Stunts only or also main performances? Nonhuman characters? Cartoons?

Performance capture has been used successfully in creating photo-real humans for films like *Titanic* and others that didn't include main performances. It has also been used successfully in creating stylized humans on films like *Monster House*. Nonhuman photo-real creatures are clearly a great use of performance capture, as evidenced by Gollum and the Na'vi in *Avatar*. Even cartoony characters have been successfully created and the perfect example of that is the film *Happy Feet*, which even went on (and will be the only one) to win an Academy Award for Best Animated Picture. Films like *The Polar Express* and *Beowulf* had success but some say the look came short and that is attributed to the fact that the characters are intended to look as real as possible.

Digital Humans and the Uncanny Valley

During my career in visual effects, I sometimes must make suggestions as to whether a certain shot should be carried out using performance animation. Even though I am very careful in suggesting the use of motion capture and do so only when I feel that it is vital for the shot, I still meet with some resistance from the character animation group. Years ago, suggesting the use of motion capture was like telling a character animator that his or her job could be done faster and better in some other way. Also, motion capture had a reputation of being unreliable and unpredictable, and animators were afraid of being stuck with having to fix problems associated with it. Many of today's experienced character animators who work with computer-generated (CG) images have had such encounters. Many of them have stories about having to clean up and modify captured motion data, only to end up replacing the whole thing with keyframe animation. I saw this first hand on *The Polar Express* and I've heard animators claim that most mocap data was dropped in favor of keyframe animation in projects like the *Lord of the Rings* films and even *Avatar*. Also, many animators dislike the look of performance animation. Some say it is so close to reality that it looks gruesome on CG human characters; others say it is far from real. In the past few years, things have changed and motion capture has repaired most of its reputation. It is also no longer such a dividing, almost religious issue in the industry, but there are still some that resist it in a massive way.

The end result of applying performance animation to digital human characters can indeed look strange, but it isn't only because of the performance data. Creating a digital human is a very difficult endeavor. Since we are used to what a human must look like to the last detail, anything that is not perfect will jump out. We have seen this effect in most digital humans and there's even a name for it: the "Uncanny Valley." It isn't that clear what causes it but some people think that it is all due to the eyes. Our brains are so well trained to know what a human is supposed to look like that something as small as a twitch in the eye or the wrong dilation of the pupil can make it all seem strange.

Jerome Chen was the visual effects supervisor for *Beowulf* and *The Polar Express*. He traces the origins of the term "Uncanny Valley" to Japanese robotics, where the intention was to create a synthetic clone of a human and make it as lifelike as possible. "The theory of uncanny valley is the closer you approach to making something artificially human the higher the level of revulsion occurs in the human observing," points out Chen, "which is why if you have very stylized animation, like there's no issue of uncanny valley in terms of mocap being used for *Monster House*, because the characters are really stylized. The problem with [*The*] *Polar* [*Express*] and *Beowulf* is not the fact that they were performance captured, I think it had to do to the fact that maybe the characters weren't stylized enough. It was the level of realism in the designs of what the characters looked like."

In most digital human attempts, animators end up creating the eyes' performance. There have been attempts to capture eye lines and eye motion using electronic sensors, but most of those attempts haven't worked because the final eye line almost always differs from the one at the capture stage. When animators work on the eye movements they usually animate the voluntary motions, but eyes have many involuntary movements that end up being left behind, and without those it is difficult to sell a digital human. Also, how many people know how to create a gaze versus a look? Sounds easy, but it really is quite complicated. There are several kinds of eye motions and lots of research on the subject. Saccadic motions, for example, are fast simultaneous motions of both eyes in the same direction. These occur differently depending on whether the person is listening to someone else or talking to someone. There are other involuntary motions such as microsaccades, oculovestibular reflexes, and others. Also, body language may be linked to eye motions and eye line in a much deeper way than we know and our subconscious can tell the difference.

Eyes and perhaps skin and minor facial expressions are the reason why we have been able to take digital humans 98% of the way, but the remaining 2% is the most elusive. Creatures that look close to human but not quite have been very successful because there's no point of reference for the last 2%. A perfect example is the Na'vi creatures in *Avatar*. "If we had a human character we would shoot them live action," says Jon Landau, producer of *Avatar*, "that's how we would approach it."

Jerome Chen thinks that it also has to do with the fact that when we are looking at synthetic humans on screen we wonder why they weren't photographed for real. "Essentially you become frustrated, because all you're doing is picking things that are missing," he says. "It's an artistic choice. If the director wants to pursue the photorealistic human and you are in the position of helping him create that vision you have to go along for that ride, and you learn a lot. There're some valuable lessons you learn in trying to accomplish that, which is why digital stunt doubles have become state of the art now."

Relevant Motion Capture Accounts

In the recent past, the use of performance animation in a project was largely driven by cost. Many projects failed or fell into trouble due to lack of understanding, because the decision to use motion capture was taken very lightly and made by the wrong people and for the wrong reasons. Some of these projects entered into production with the client's informed knowledge that motion capture may or may not work, and a budget that supported this notion. Other clients blindly adopted performance animation as the primary production tool, based on a nonrepresentative test or no test at all, and almost always ran into trouble. In some cases, the problems occurred early in production, resulting in manageable costs; in other cases, the problems weren't identified until expenses were so excessive that the projects had to be reworked or cancelled.

Projects such as *Marvin the Martian in the 3rd Dimension,* the first incarnation of *Shrek, Casper,* and *Total Recall* were unsuccessful in using performance animation for different reasons.

Others, such as *Titanic* and *Batman and Robin,* were successful to a certain degree: They were able to use some of the performance animation, but the keyframe animation workload was larger than expected. A few projects managed to successfully determine early in production whether it was feasible to use motion capture or not. Disney's *Mighty Joe Young, Dinosaur,* and *Spiderman* are good examples of projects in which motion capture was tested well in advance and ended up being discarded.

In live-action films, large crowd scenes and background human stunt actions are some of the best applications. Character animation pieces, such as *Toy Story* or *Shrek,* will almost never have a use for performance animation, but let's not forget *Happy Feet*.

Total Recall

In 1989, Metrolight Studios attempted what was supposed to be the first use of optical motion capture in a feature film. The project was perfect for this technology and was planned and budgeted around it, but it somehow became the first failure of motion capture in a feature film.

The project was the futuristic epic *Total Recall,* starring Arnold Schwarzenegger. The planned performance animation was for a skeleton sequence in which Arnold's character, the pursuing guards, and some extras, including a dog, would cross through an X-ray machine. The idea was for characters to have an X-ray look.

Metrolight hired an optical motion capture equipment manufacturer to capture and postprocess the data. The motion capture session had to be on location in Mexico City, and the manufacturer agreed to take its system there for the shoot. The manufacturer also sent an operator to install and operate the system.

"We attached retro-reflective markers to Arnold Schwarzenegger, about 15 extras, and a dog," recalls George Merkert, who was the visual effects producer for the sequence, "then we photographed the action of all of these characters as they went through their performances." Regarding the placement of the markers, Merkert recalls that the operator "advised Tim McGovern, the visual effects supervisor, and me about where the markers should be on the different characters. We placed the markers according to his directions."

They captured Arnold's performances separately. The guards were captured two at a time, and the extras were captured in groups of up to 10 at a time. Even by today's standards, capturing that many performers in an optical stage is a difficult proposition unless the equipment is state of the art. “It seemed a little strange to me; I didn’t see how we could capture that much data, but the operator guaranteed us that everything was going to be okay,” recalls Merkert. The motion capture shoot went smoothly as far as anybody could tell, but nobody knew much about motion capture except for the operator sent by the manufacturer. Therefore, nobody would have been able to tell if anything went wrong.

After the shoot, the operator packed up the system and went back to the company’s headquarters in the United States to process the captured data, but no usable data was ever delivered to Metrolight. “He [the operator] needed to do a step of computing prior to providing the data to us so that we could use it in our animation, and we never got beyond that step. He was never able to successfully process the information on even one shot. We got absolutely no usable data for any of our shots,” says Merkert. “They had excuses that it was shot incorrectly, which it may well have been. My response to that was, ‘Your guy was there, telling us how to shoot it. We would have been happy to shoot the motion capture any way that you specified. The real problem is that you never told us what that right way was even though we asked you every few minutes what we could do to make sure this data gets captured correctly.’” The production even sent people to the motion capture manufacturer to see if anything could be done to salvage any data, but all of it was unusable.

“I think what happened is that their process just entirely melted down, didn’t work. They couldn’t process the data and they were unwilling to say so because they thought they would get sued. It wound up costing my company maybe three hundred thousand dollars extra,” notes Merkert. “Regardless that the motion capture simply didn’t work, we were still responsible to deliver to our client. The only way we could do that was by using the videotapes, which were extremely difficult to use for motion tracking. Unfortunately, because the motion capture company advised that we do it in this way, we had lit the motion capture photography in such a way that you could hardly see the characters to track them. You could see the retro-reflective balls attached to the characters very well, but you couldn’t see the body forms of the characters very well.” Optical systems in those days worked only in very dark environments. Almost no lighting could be used other than the lighting emitted from the direction of the cameras. “Our efforts were very successful, however. Tim McGovern won the 1990 Academy Award for Best Visual Effects for the skeleton sequence in *Total Recall*,” he adds.

Many lessons can be learned from this experience. First, a motion capture equipment manufacturer is not an animation studio, and thus may have no idea of the requirements of animation. In this case, the operator didn't even know the system well enough to decide whether it should be used to capture more than one person at a time. This brings us to the second lesson: Motion capture always seems to be much easier than it really is. It must be approached carefully, using operators with a track record. Extra personnel are also needed to make sure nobody is creating any noise or touching the cameras. Third, if possible, never capture more than the proven number of performers at a time. In this example, it was clear that each performer could have been captured separately because they didn't interact, but it was decided not to do so. Fourth, both the captured data and calibration must be checked before the performers take off the markers and leave the stage, and extra takes of everything must be captured for safety. Fifth, marker setups have to be designed by experienced technical directors and not by operators who know little about animation setup requirements. Sixth, when working on an unknown remote location, the motion capture system has to be tested before and during the session. Extra care must be taken because lighting conditions and camera placements are always different from those in the controlled environment of a regular motion capture studio.

Shell Oil

After 1992, many companies were involved in commercial work using motion capture. A campaign that was especially prominent was one for Shell Oil that featured dancing cars and gasoline pumps. It was produced by R. Greenberg and Associates in New York using a Flock of Birds electromagnetic device manufactured by Ascension Technology Corporation.

Fred Nilsson, now a senior animator at PDI, worked at R. Greenberg in 1994 when the Shell campaign was produced. "We got the main skeleton guys from Softimage to come down and they taught us how to build a skeleton for motion capture," recalls Nilsson. "We set up a skeleton that had goals where all the motion capture points were, and then there was another skeleton on top that was a parent of all the joints of the other skeleton, so that we could capture the motion and then animate on top of the other skeleton." The main complaint that Nilsson had about the motion capture data was that the characters were never firm on the floor, but the second skeleton was used by the animators to ground the characters, with the help of inverse kinematics.

The skeleton that contained the final motion was used to deform a grid, which was used to deform the final character—a car or gas pump. Because of the strange proportions of the characters, a lot of cleanup work was necessary. They also had to deal with interference. “We were on a sound stage and there was some big green screen wall that was all metal and it was really interfering,” says Nilsson.

“A year later we did three more Shell spots and all of us said no, we don’t want to use motion capture,” recalls Nilsson. They animated the last three spots by hand and it took about the same time to produce them. “They turned out a lot better,” says Nilsson.

In this particular case, the biggest benefit gained by the use of performance animation was the extensive press coverage that Shell obtained because of the “new” technology that was used to animate the cars and pumps. The free publicity didn’t dissipate with the creation of the last three spots by keyframe animation, since most people thought that performance animation was used on those as well.

Marvin the Martian in the 3rd Dimension

Marvin the Martian in the 3rd Dimension was a 12-min stereoscopic film that was produced for Warner Brothers Movie World theme park in Australia in 1996. It was meant to look like a traditional cartoon, but because it was stereoscopic, it had to be created using two different angles that represent the views from both the right and left eyes. The best way to achieve this was to create all the character animation using a computer. The challenge was to make sure the rendering and motion looked true to the traditionally animated cartoon.

In 1994, Warner Brothers acquired a Motion Analysis optical motion capture system, thinking that they could have actors perform the parts of the Warner Brothers’ cartoon characters. Among the companies involved in the project were Will Vinton Studios, PDI, Metrolight Studios, Atomix, and Warner Brothers Imaging Technology (WBIT). “They asked if we could do Daffy Duck with motion capture,” recalls Eric Darnell, director of *Madagascar.* After several months of testing, Warner Brothers decided to scratch the motion capture idea and use traditional animation after all. They ended up providing keyframes on pencil to all the 3D animation studios to use as reference for motion. The Warner Brothers motion capture equipment was auctioned in 1997, after WBIT closed its doors.

The Pillsbury Doughboy

In 1996, TSi was approached by Leo Burnett, the advertising agency that handles the Pillsbury account, to create two spots with the Pillsbury Doughboy using performance animation. The agency had already explored the technology once by doing a spot using an electromagnetic tracker at Windlight Studios, and now they wanted to try optical motion capture.

The first step was for us to produce a test with the Doughboy performing some extreme motions. We decided to bring in a dancer/choreographer we had often employed, because she had a petite figure and was a good actress who could probably come as close to performing the Doughboy as any human could. Since the client didn't specify what kinds of motions they were looking for, we decided to have the Doughboy dance to the tune of M.C. Hammer's "You Can't Touch That." We captured the main dance and a couple of background dances we would use for background characters and, after post processing, converted them to the Alias format.

The main dance was applied to the Doughboy, and the background dancers were mapped to models of Pillsbury products. The resulting piece was quite disturbing: the Pillsbury Doughboy performing a sexy dance surrounded by three or four cylindrical chocolate-chip cookie-dough dancers doing a pelvic-motion-intensive step. It wasn't exactly what you'd imagine the Doughboy would do, but they did say "extreme motions." I was a little worried about what the client's reaction would be, but I figured that if they really wanted to use motion capture for the Doughboy, they had a right to see the kind of realism in the motion they would get. We sent the test to Chicago.

The day after we sent the test we got a call from producers at Leo Burnett. Their first comment was that the dark lighting that we used would never be used on a Doughboy commercial. We explained that the point of the test was for them to see the motion, not the lighting, and that the lighting had been approached that way to match with the sexy nature of the motion. As for the motion, Leo Burnett said that it was offensive to their client (Pillsbury), but that it had proven the point. They gave us two Doughboy spots to do simultaneously.

Part of the deal was that we would set up a motion capture session that the Leo Burnett and Pillsbury people could attend. We used the same woman who performed the sexy Doughboy for the test. One of the shots involved the Doughboy vaulting over a dough package using a spoon as a pole. We knew that at the time we couldn't use captured motion data for the interactions of the spoon and the Doughboy, but we captured it anyway, because our client wanted this method used on all their shots. We figured we could use it for timing, and that's exactly what we did. All the other shots were standard Doughboy fare: the Doughboy pulling a dish, smelling the steam from baked products, just standing and talking, and, of course, being poked in the belly. The session went very smoothly. Our clients and their clients were very interested in the technology that we were using and had a lot of fun watching the performance.

The data was processed and converted to the Alias format, where we already had set up the Doughboy model for the test. A big problem we had on the test was that the legs of the Doughboy were so short that even retargeting them as best as we could at the time they would never stay on the ground and look right. We knew we would run into this problem again while working on the shots for the commercials. Benjamin Cheung was the technical director in charge of applying the data and lighting the Doughboy. He used only portions of the data, throwing away most of the legs' and some of the arms' motions, particularly when they interacted with the spoon. We probably used only 15% of the captured data on the character, and the rest just as reference for poses and timing. We were already expecting that, so it wasn't a surprise and we didn't miss our deadline.

The client, as we had agreed, stayed faithful to the captured performance and didn't make any major animation changes. It was a very smooth pair of projects, and although we were charging very low rates, we managed to make a decent profit. The project would have cost a little more if it had been keyframed, but it wouldn't have made a huge difference.

The main risk on a project such as this concerns the art direction, such as when a client who is not familiar with performance animation suddenly discovers that the motion looks different from what was expected and starts making changes. In this case it didn't happen because we did our homework to educate the client on what our system could do; however, they did not return for more motion-captured Doughboys, preferring to go back to keyframe animation. I was not surprised.

Batman and Robin

PDI had provided a CG Batman stuntman for *Batman Forever,* using optical motion capture as a means to collect the performance of a gymnast performing on a stage doing flips and other movements on rings. *Batman and Robin* required much more elaborate digital stunts, many of them happening in the air. PDI was again contracted to create these stunts using motion capture.

The motion capture services were provided by Acclaim Entertainment, the computer game company that had the video game rights for the Batman property and that had earlier provided the motion capture services for *Batman Forever.* They used their proprietary four-camera optical system to capture all the necessary motions for the stunt sequences.

Acclaim's team captured several of the shots at Acclaim's motion capture stage in New York, except for the shots that belonged to the sequence in which Batman and Robin skyboard away from an exploding rocket while in pursuit of Mr. Freeze (played by Arnold Schwarzenegger), trying to recover a diamond he had stolen. Because all the action in this sequence happened in the air, John Dykstra, visual effects supervisor, decided to bring Acclaim's optical motion capture equipment to a vertical wind tunnel used for military training. The tunnel was located at the Fort Bragg military base in North Carolina, and the performers were not actors, but members of the US Army's Golden Knights parachuting team. Dykstra directed the performances in the tunnel, and Acclaim processed the data and delivered it to PDI.

"We took the skeleton files from Acclaim and converted them to a setup," explains Richard Chuang, who was the digital effects supervisor for PDI. "We also converted all the motion data into our own animation space." Dick Walsh, character technical director at PDI, had set up a system that allowed them to adjust the differences in physical attributes between the digital model and the performer (Figure 2.1). "We actually had a certain flexibility with the use of motion capture data from different performers on the same character," recalls Chuang. The data was then used to move a skeleton that would drive PDI's muscle system, also written by Walsh. In addition, animators had the ability to blend the captured data with keyframe animation, with or without inverse kinematics. The system didn't require animators to modify the motion data. They worked on clean curves and were able to blend the animation and the captured data using animatable ratios. The system was especially useful when blending between captured motion cycles.

Figure 2.1 Dick Walsh working on the Batman character setup. Photo courtesy of PDI.

"In both cases we ended up using probably around 20% motion capture and 80% animation," notes Chuang, referring to both Batman films. "What we found is that because the motion is always captured in a stage, it really has nothing to do with the final performance you need for the film. For the director to have control of the final creative, we ended up taking motion capture as a starting point, and then animating on top of it. Then we'd be able to dial between motion capture and animation whenever we needed to."

Part of the problem with motion capture in this case was that the motions were for the main action of the sequence and the director wasn't present to direct the performance. When you capture motion for background characters, it is okay for the director to delegate the task, but in a case such as this, the director would expect to have first-hand creative control over the final performance. That meant the performance would have to be modified by the PDI team to his specification. Larry Bafia, animation director at PDI's commercial and film effects division, worked on the air sequence. "It was the first time that I ever had to deal with motion capture data at all, so I had no idea what I was getting into at the time," recalls Bafia. "This is a character that comes from a comic book and it really has to look like a superhero," he adds. "Some of the poses had to be exaggerated and it wasn't something that a human could do in a wind tunnel." The team kept most of the captured data for the motions of the character's torso, especially the actions of weight shifting on the skyboard. They used keyframe animation on many shots to enhance the pose of the arms and to animate the fingers, which were not included in the motion data. Inverse kinematics was used mostly on the legs.

The motion capture session was directed by the special effects supervisor, John Dykstra, who tried to capture as many different shots as possible for the director to choose from. This was done even before background plates were available, so no final camera angles and framing were available. "What you do is you capture all the possibilities, and then based on that you try to edit something that will fit your film," says Chuang. "But by the time you get your background plates you can end up with something completely different."

PDI technical directors prepared desktop movies of all the captured motions so that, after the plates became available, the team could decide which actions would apply for each particular shot. "One of the problems was the fact that a lot of the shots were long enough where the particular motion capture cycle didn't have quite enough activity," recalls Bafia. Because of this, the team had to blend several captured motions into single shots. An example of this is the shot in which Robin catches the diamond after Mr. Freeze loses it. “I had to start the front end of the shot with one piece of motion capture where [Robin] is basically staying alive and balancing himself on the board and getting ready for spotting this diamond and catching it,” recalls Bafia. After catching the diamond, Robin had to do a spin-around, not unlike that of a surfer. “Because of the restrictions inside the wind tunnel, the guy could do the spin but he had to pull up early so that he wouldn’t fall too far and drop down toward the fan that was actually suspending him,” says Bafia. PDI had to use inverse kinematics to maintain the character’s feet on the board, and keyframe animation to blend between two motion files that had a reversed feet stance—all this while keeping the elements in the captured data that John Dykstra was interested in maintaining.

The fact that the performers weren’t aware of where the final camera would be located was a problem that translated into the performance. “I would like to see a situation where you actually try motion capture after you shoot the film versus before, so you know where the background plate is going to be, and the director directs the action accordingly,” notes Chuang.

Regardless of the expected problems with captured performances, the resulting shots were extremely successful. PDI had the experience to use whatever they could out of the motion data and discard the rest. Therefore, the project was completed in a timely manner, the clients were happy, and PDI made a good profit. The difference between this and other projects using motion capture was that for *Batman and Robin,* motion capture was used to achieve a certain look, not to save money.

Shrek

Before the original *Shrek* animated film started production, DreamWorks Feature Animation had been developing it to be mostly based on performance animation.

The decision to use performance animation on *Shrek* was based not only on cost (which indeed played a big part in the decision), but also on the desire to achieve a certain look. Before *Shrek* was even in preproduction, PropellerHead Design (Rob Letterman, Jeffrey Abrams, Andy Waisler, and Loren Soman) produced a test of a similarly shaped character called “Harry” for a potential television show for HBO. The motion capture for the test was done at TSi.

For the Harry test, the propellerheads (as they were known in the *Shrek* production) prepared an appliance that was worn by their performer. The appliance would add the volume that the performer needed to approach the proportions of Harry. This was a new idea that we had never seen done before, and it seemed like a good one. During our first of two sessions, we placed the markers on the appliance and captured the body motions with our optical system. The tracking of the data was a challenging feat because it was an uncommon setup, but we managed to get it done with our ancient six camera system.

In the second session, we were supposed to capture Harry's finger motion. We had never tried to capture the motion of fingers with an optical system, and we didn't guarantee that it would work. It was strictly a developmental session. We placed the small markers that we usually used on face capture on the finger joints. The performer sat down in a chair, and we immobilized his hands as much as possible because the wrist movement had already been captured in the previous session and we knew we couldn't capture the fingers if the wrists were moving. We captured several passes of the actor moving his fingers with the timing of the dialogue, but in the end it was impossible to track those motions, and we had to scratch the idea of optical finger capture. Facial capture wasn't even tested for this project.

The propellerheads finished the Harry test, filling in the additional animation, plus texture and lighting. The end result was stunning. The Harry test laid the foundation for the production of *Shrek* at DreamWorks. The propellerheads would coproduce the film, which would be animated using techniques similar to those in the Harry test: optical motion capture with body extensions or appliances.

The DreamWorks studio in Glendale was geared up with Silicon Graphics computers running different animation platforms and a brand new 10-camera Motion Analysis optical motion capture system. A production staff was assembled and production of a 30-s test began. Character animators started to work with the optical motion data. Several months later, the test was shown to Jeffrey Katzenberg, DreamWorks' cofounder and the partner responsible for animation films. Shortly thereafter, the production was halted and the project went back into development. The propellerheads were no longer involved. Exactly what happened is still a mystery, but motion capture was ruled out for this project. Among other things, the production made a mistake by using experienced character animators to work with the captured motion data, even from the stage of retargeting. It is not certain if that actually affected the test, but it would have eventually become a big problem.

Godzilla

Another character that was supposed to be animated by performance was Godzilla, in the 1998 film with the same name. Future Light had developed a real-time motion capture system based on optical hardware by Northern Digital. The system was connected to a Silicon Graphics Onyx with a Reality Engine, and it could render an animated creature in real time. Future Light was a division of Santa Monica Studios and a sister company of Vision Art, one of the CG studios involved in the visual effects production for *Godzilla*. Director Roland Emmerich liked the system because he could see the creature motion in real time, so the production proceeded with the idea of using the system as much as possible for the animation of the creature.

The character design was proportionally close to the human body, so it was possible to map human motion into the character's skeleton without much retargeting. According to Karen Goulekas, associate visual effects supervisor, the motion capture worked technically, but it didn't yield the results they were looking for esthetically, so they decided to abort that strategy and use keyframe animation instead. In the end, the animation for all the creatures, including Godzilla and its offspring, was done with keyframe animation.

"The reason we pulled the plug on using the motion capture was, very simply, because the motion we captured from the human actor could not give us the lizard-like motion we were seeking. The mocap could also not reflect the huge mass of Godzilla either," adds Goulekas. "During our keyframe tests, we found that the Godzilla motion we wanted was one that maintained the sense of huge mass and weight, while still moving in a graceful and agile manner. No human actor could give us this result." In addition, they had Godzilla running at speeds as high as 200 mph, with huge strides; this proved to be impossible to capture.

In general, using human motion to portray a creature like Godzilla, even if it is proportionally close to a human, is a mistake. In previous Godzilla films, it was always a guy in a suit performing the creature. In this version as initially conceived, it would also be a guy in a suit, except it would be a digital suit.

Sinbad: Beyond the Veil of Mists

Sinbad: Beyond the Veil of Mists was the first full-length CG feature that used mostly performance animation. The film was produced by Improvision, a Los Angeles production company, and Pentafour Software and Exports, one of India's largest and most successful software companies.

The story concerns an evil magician named Baracca, who switches places with a king by giving him a potion. The king's daughter, who is the only one who knows that the switch took place, escapes, eventually running into Sinbad. The Veil of Mists refers to a mythical place where Sinbad and the princess have to go to find the answer to defeating Baracca and saving the king. The main character, Sinbad, was voiced by Brendan Fraser. Additional cast included Leonard Nimoy, John Rhys-Davies, Jennifer Hale, and Mark Hamill.

With an initial budget of $7 million, it is no secret that motion capture was adopted by the producers primarily as a cost-effective alternative to character animation. They had originally planned to finish the whole film in 6 months, thinking that using motion capture would be a huge timesaver. Evan Ricks, codirector of the picture, was approached by the producers with a rough script. "They already knew that they wanted to do motion capture," recalls Ricks, who is a cofounder of Rezn8, a well-known CG production and design studio in Los Angeles. He told the producers that a film of this magnitude would probably take at least 2 years of production, but they insisted on trying to finish it on schedule.

Given the constraint that the production would have to utilize motion capture, Ricks initially explored the possibility of using digital puppeteering to animate the characters in real time, using that as a basis for animators to work on. After conducting a few tests, they found that there weren't enough off-the-shelf software tools available to produce this kind of project in real time, so they decided to use an optical motion capture setup.

Motion capture would be used for human characters' body motions only. They couldn't find a good solution at that time to capture facial gestures optically, so they tried to use digital puppeteering instead. Finally, they decided that all facial animation would be achieved by shape interpolation. Pentafour had acquired a Motion Analysis optical motion capture system that was sitting for some time unused in India, but the production decided to use the services of the California-based House of Moves and their Vicon 370E optical system. The reason was that in Hollywood there were more resources and expertise readily available for the shoot, such as stages, actors, set builders, and motion capture specialists.

The production rented a large sound stage at Raleigh Studios and started building all the props and sets that the actors would interact with during the capture session. They even built a ship that sat on a gimbal that could be rocked to simulate the motion of the ocean. "That introduced all kinds of problems in itself," notes Ricks, “because all of a sudden you don’t know exactly where the ground is and you have to subtract that all out.”

In addition to the people from House of Moves, the production had the services of Demian Gordon, who was brought in as motion capture manager. Gordon had worked extensively with motion capture before when he was at EA Sports. His job was to help break down the script, work closely with House of Moves, help design marker setups for all characters, and find solutions to problems that arose during production.

The House of Moves’ Vicon system was brought into the sound stage and installed using a special rig that rose about 30 ft off the floor, which kept all seven cameras away from the action on the stage. In addition, special outfits were made for the performers to wear during the sessions.

Ricks wanted to shoot the motion capture sessions on tape from many angles as if it were a live television show. He was planning to edit a story reel after the motion capture was done and to use it while producing the CG images. A camera crew was hired initially, but was eventually dismissed by the production company because of its high cost. “We paid the price later,” says Ricks. “The camera can really help the director to visualize what’s going to be on the screen. I guess you can stand by and use a finder, but with live action, the very next morning you can see dailies. You can see what you actually shot and if the angle works. With motion capture, you get a single reference and end up with mechanics more than cinema.”

Even if they had been able to tape all the captured shots, they had to deal with the motion capture system’s inability to synchronize video and capture scan rates. They couldn’t use SMPTE time code, because at that time no optical system was able to generate it, although it is a very important aspect of production work. According to Ricks, the lack of a good way to synchronize all the elements played a big part in the extension of the production schedule.

Because of the compressed schedule, the crew had to capture thousands of shots in about 2 months. According to Ricks, 4 months would have been ideal. “We weren’t able to finish all the storyboarding before we started the capture, so sometimes we would be literally looking over the editor’s shoulder, telling him what we wanted, or looking at new storyboards that had just been drawn, and then go over to the stage around the corner and capture it. Not the ideal way to work.”

Typical problems during the motion capture sessions included performers losing or relocating markers. On projects of this magnitude, it becomes very difficult to keep track of exact marker placements, which creates inconsistencies in the motion data. They also had to deal with obstruction problems, where cameras didn't have a direct line of sight for all markers.

Many of the shots had more than one performer interacting at the same time. They were able to capture up to four characters simultaneously, although it became somewhat problematic to clean up the data. "A lot of it took place aboard ship, and there're six people on the crew, so there's a lot of interactions," says Ricks. Fighting sequences were especially difficult to capture, particularly when somebody would fall on the floor and lose markers.

The production was set up so that all design work, preproduction, initial modeling of characters, and motion capture work were done in the United States. The animation side, including the application of captured data, modeling of sets and secondary characters, shading, and rendering, would be handled by the Pentafour CG studio in India, which was a relatively new studio whose experience was mainly in the production of art for CD-ROM titles. Ricks wanted to use very experienced people to handle the most challenging parts of the project, creatively and technically, leaving the supporting tasks to the inexperienced crew.

Overall, the first 6 months of production had been extremely productive. Evan Ricks and Alan Jacobs had managed to rewrite the script practically from scratch, while Ricks and the production designer developed the look. They hired the staff and cast, recorded the voice track, storyboarded, built sets to be used for motion capture, tested motion capture alternatives, and designed and built all the digital characters and sets. They even stopped regular production for over a month to create a teaser for the Cannes Film Festival. All the motion capture was also finished within those first 6 months. Everything was looking great and the staff was energized. "I was continually told by visitors and investors how beautiful our designs were, and how new and different as opposed to a 'Disney' look," says Ricks.

Unfortunately, after the completion of the motion capture, the producers had a mistaken perception of the work left and didn't understand what had been achieved in those 6 months. In their view, most of the technical hurdles in the production had been surpassed. Not only did they not see the need to fulfill repeated requests to hire additional experienced personnel, but they also started laying off their most experienced staff members. Finally, the remainder of the production was moved to the Pentafour studio in India.

The difficult part started after House of Moves finished postprocessing and delivered the data. "There was a serious underestimation in how long it takes to weight the characters properly," notes Ricks, who describes some bizarre problems with the deformations of the characters. The deformation setup was much more time consuming than they had originally estimated. The problem was augmented by the fact that they relied exclusively on off-the-shelf software, and there's a lack of commercial tools for handling captured motion data. A project of this scale would normally have some sort of in-house R&D support, but the budget didn't provide for it.

The mechanical setups for the characters were created using rotational data. For a large project like this, with so many characters and interactions, it would have been beneficial to create customized setups using also the translational data, using markers as goals for inverse kinematics or as blended constraints. When using rotational data, the mechanical setup has already been done by the motion capture studio, and not necessarily in the best possible way for the project at hand. "[House of Moves] hadn't done tons of this stuff at that point either, and they had to staff up. It was a learning experience for everybody," notes Ricks, "but they did a very good job under the circumstances."

A typical problem with captured motion data is in the unmatched interactions between characters and props or other characters. The crew in India has been responsible for cleaning up all these problems, including the inability to lock the characters' feet on the floor. This particular problem can be fixed by using inverse kinematics, but it can also be hidden by framing the shots in such a way that the characters' contact with the floor is not visible. Ricks was trying not to let this problem rule the framing of shots.

As Ricks predicted, the length of the production of the film was close to 2 years. "If you deconstructed just about any recent film, you'd find out what the director originally wanted and what actually took place are very often two different things, and that was no different here," says Ricks.

"The key is to use this experience to identify the strengths of mocap, to realize that its value is currently not as broad as many think. It is a new camera, not an end in itself," concludes Ricks.

The Polar Express

Robert Zemeckis had teamed up with Tom Hanks to bring the famous children's book *The Polar Express* to the screen. The original idea was to give the film the same painterly look as Chris Van Allsburg's book. The methodology to achieve the look would be to use live-action characters shot in front of green screen in order to place them in digitally generated backgrounds. Once the elements were integrated, some kind of filter would be applied to give it the look.

A day was scheduled to bring Tom Hanks to a soundstage at Sony Pictures to do a green screen test. Ken Ralston and Scott Stokdyk were the visual effects supervisors for the test. I was helping Scott with the preparations and we thought it would be interesting to try to capture Tom's face and see how accurate looking we could get it. One must remember that while there had been some interesting facial motion capture examples at the time, there still wasn't a believable side-by-side test of the face of a known person, so we weren't exactly sure what the result would look like.

We had Vicon bring a system to the soundstage for the day of the test. Once Tom was there, we spent most of the time shooting the green screen. Afterward, we were given a few minutes to do our thing. We scanned Tom's head with a laser scanner and slapped about 50 markers in his face on places where I thought would give us the movement we needed. We then had Tom say a few lines in front of the cameras. Nobody but Scott and I had any hope that this test would turn into something. Ken said to me that he was sure it wasn't going to succeed and Zemeckis didn't even pay attention to us when we were doing it. The day after the test almost everyone had forgotten we captured anything. I was the only person working on the motion capture test. Everyone else in the crew was working on the green screen test.

After several days, we started seeing the first green screen tests in dailies. They weren't looking good at all. The static elements in the shot were fine with the painterly look, but the live-action actors looked like they had a skin disease.

In the meantime, I was working away on the mocap test. I had already written a facial muscle-based solver that was used on *Stuart Little II* for talking cats and I was modifying it to work with captured data. Once I set up the system with Tom's facial model and started driving it with the data we captured, I was happily surprised. The test actually looked like Tom and it was moving the right way. It clearly required more data but you could see Tom in there.

After several weeks, the painterly look tests weren't looking any better. I hadn't been asked to show anything as most people had forgotten or assumed what I was doing didn't work. When I finally showed it in dailies, Ken Ralston looked at it and said "this may actually work. Let's show it to Bob." Once Zemeckis saw it, he decided to pull the plug on the green screen test and use mocap for *The Polar Express*. I wasn't expecting that reaction.

I still didn't know how extrapolate from that test into a whole 90-min film with lots of characters interacting with each other. This was a scary time, but it got worse when I was told that I had to have the ability to capture face and body together and we'd start shooting in about 5 months.

Producer Jody Echegaray and I had to start by choosing the right equipment and assembling the team. We started talking to the different equipment manufacturers and also interviewing all the possible candidates. We knew that we pretty much had to hire almost anyone who had any experience with motion capture to cover the scope of this project. Our first hire was Demian Gordon, who would be our motion capture supervisor. He had just finished working on the *Matrix* sequels and was able to bring with him a lot of good people to fill many of the slots in our team. Part of his expertise was also the design of camera layouts for capture stages, so I asked him to calculate how many cameras would be needed to meet the spec that we were asked to deliver. He estimated that we needed no less than 70 to 80 cameras to cover the face and body markers of up to four people at a time.

None of the existing manufacturers had the capabilities we needed, so we started to talk to them about working with us to daisy chain several of their systems together. As we told some of them what we were thinking, most decided to pass and at least one of them told us we didn't know what we were doing. The only manufacturer willing to work with us was Vicon, and between them and us we started designing the final layout, calibration methodology, and pipeline to achieve the final spec. Brian Nilles, Vicon's CEO, was instrumental in having his engineering team work on their software to allow us to hook up five Vicon systems together. All of us were on constant communication and as soon as they could ship some equipment we started to assemble a massive truss for placing cameras inside a soundstage at the Culver Studios. At the beginning, the assembled system was very delicate and just the wrong setting would make it malfunction. There was also the issue of calibration, as there was no known way to do this with a daisy-chained system. Initially, we had to calibrate five systems separately and then merge the calibrations, but eventually Vicon provided us with software to merge them on the fly. The problem was that the calibration had to be seen by all cameras and apparently in some particular order. We had to come up with a "dance" that D.J. Hauck, our head mocap operator, had to perform perfectly every time in order for the calibration to take. In the next couple of months, we came up with protocols and procedures for everything that could happen and started educating the production team on what to expect and what was possible and what wasn't.

We ended up with three different mocap stages: the main stage, where body and facial data would be captured, and two larger stages, where only body data would be captured. The main stage ended up with 84 cameras; 70 of them were Vicon cameras and would be used strictly for face capture. The rest of the cameras were operated by Giant Studio and would be used for body capture. Giant Studio was also responsible for the operations of the two larger sets, one of which was 36,000 ft^3 (30 × 60 × 20 ft), the largest ever done at the time and used for large captures with many performers. The other large stage was 27,000 ft^3 (30 × 30 × 30 ft) and was used mostly for stunt action because it had more height. The main stage was only 1300 ft^3, which is 10 × 10 × 13 ft high.

We felt the stage was ready to shoot only about a week before the actual production would start. We held a massive meeting with all the different groups that would be on set and with the director and we went step by step over what was to happen on set. I also had to answer hundreds of questions from people who had never even heard about motion capture and were about to be dropped in the middle of the largest motion capture effort ever done. After that day, we started training the different groups about their new tasks. The makeup artists' job was to apply face markers and also to make sure that the marker sets were always complete during the shoot. Wardrobe people were in charge of mocap suits and body markers. Grips were in charge of bringing in and out of stage props that had to be built mostly with chicken wire. Also, the first assistant director and production assistants had to be educated about their new roles to get the stage ready for shooting.

I have to say that the crew for that shoot was ideal. Zemeckis understood immediately what he could and couldn't do. The makeup crew, led by Dan Striepeke, had to place 152 markers on up to 24 faces every day before shooting could begin, and they learned to do it very quickly and very accurately. Demian Gordon, our stage supervisor, had no problem stopping the shoot if any problem arose. He was never intimidated by Zemeckis, Hanks, or anyone else and because of that we never had any data surprises. We shot for 45 days without a major problem and everyone enjoyed the process. I heard Zemeckis say that he would never shoot a live-action movie again. He was so excited that he brought lots of people to see the stage. Among the people I saw, there were several studio heads and famous actors and directors like Steven Spielberg, James Cameron, and others.

An important aspect of the way Zemeckis decided to shoot his film is that he didn't want to frame shots on set or even look at real-time graphics of the performances. The small set was designed as an intimate setting where he could work with the actors. Not many people were allowed to stand inside the set.

When the shoot was done, we had to move into postproduction. We set up three departments to deal with the captured data. First was the Rough Integration Group, whose job was to create an assembled sequence from all the different pieces captured in different passes and stages. This sequence was played in real time and using a virtual camera setup the director of photography would design the shots. During *The Polar Express* we called that process Wheels, and it was specifically designed for Zemeckis' style of filmmaking.

The second group was the data-tracking group, whose responsibility was to track and deliver facial data from the shots created and used by editorial. The third group, called "Final Integration," would bring each shot into Maya and integrate body and face data with the camera created by the DP. The rest of the pipeline was more typical of an animated feature. Shots went on to animation, layout, cloth, effects, color and lighting, and finally compositing.

"[*The*] *Polar Express* was in terms of motion capture probably the most groundbreaking, not necessarily on the imaging side, but because of the technology that had to be put together. Nobody had actually done body and face together at that point," says Jerome Chen, visual effects supervisor.

Beowulf

Robert Zemeckis' second directorial effort in the performance capture arena was *Beowulf*. His initial goal was, as with *The Polar Express*, to create stylized environments and characters, but again the project evolved into the almost-photo-real realm. "Zemeckis wants to see realistic textures, realistic aspects to the environment and to the characters," recalls Jerome Chen. "He doesn't gravitate toward the stylized designs; he likes it to be realistic. . . . Once again you are starting out trying to be stylized and ending up wanting to see more detail in the characters once you see realistic motion on them."

Beowulf's pipeline was almost identical to the one used for *The Polar Express*. The main difference was in the data acquisition. Imageworks upgraded to Vicon's latest hardware included cameras with much higher resolution and the ability to accommodate many more of them, over 250 to be more specific.

The volume was bigger so all the capture was done in a single stage. Also, the crew had a lot more experience from having done *The Polar Express* and *Monster House*. "A lot of the protocols established in the stage capture were now perfected," says Chen. "We budgeted more time in *Beowulf* to do facial animation, particularly preserving volume in the face and on the lips." The facial system in *Beowulf* was different from the one used on *The Polar Express*. It was mostly driven by FACS (Facial Action Coding System, described in Chapter 1), where actors would have a number of predetermined expressions and an algorithm would decide the optimal combination of expressions for each frame. They also paid a lot of extra attention to the eyes.

The Wheels process, now called Camera Layout, was still a major piece in the postproduction process as all the shots were designed there. For Zemeckis' style of filmmaking, that process worked very well because he is a big fan of complex camera work. Camera Layout allowed him to experiment and tweak the camera angles as many times as he wanted.

Avatar

The main difference between the production pipelines of *The Polar Express*/*Beowulf* and *Avatar* is in the way the camera work is designed. Zemeckis likes to have a director of photography design the shots well after the actual capture, while Cameron likes to design his cameras while he's capturing the performances. For that purpose, he used a virtual camera he called "Simulcam" that was captured at the same time and in the same way as the performers. It was very important to have real-time feedback in order to achieve this and Giant Studio was able to provide it.

Avatar had been in the works for many years but it didn't start production until 2005. "*Avatar* wasn't done before because the technology didn't exist to do the movie before," says Jon Landau, producer. "People get confused about what the technology was. They think it's the 3D. It has nothing to do with the 3D. It was the facial performance of the CG characters. We knew that for this movie to work the characters needed to be emotional and engaging, and that wasn't there."

The second large difference between *Avatar* and its predecessors was the fact that the facial data acquisition was done using a helmet-mounted camera and no facial markers. Image-based analysis was used to obtain the facial sampling. "Over the years, we did some testing ourselves in another project called Brother Termite, where we did some image based facial capture. That proved to us that there was some validity there. Then in seeing things like Gollum in *Lord of the Rings* and what other people were doing we felt in 2005 that if we pushed the technology that we could be the impetus to get it to where we needed it to go," says Jon.

I asked Jon about his take on motion capture and what was their intention when they decided to use it for *Avatar*. "We don't call it motion capture, we call it performance capture. We wanted the creative choices of performance to be made by actors and not made by animators, and that takes nothing away from great animators, because the animators are a very important step in the process, but the decisions that are made of when to stare and not blink and when to twitch—those are made by actors because that's what they do. The animators were responsible for showing that performance came through in the CG characters and that is a huge skill into itself. Performance capture is not a money saver to us, but a performance saver. If an actor gets that great performance, they have to only do it once. In fact it's a pure way of acting. Sigourney Weaver, on her first day on the set I said to her, so what do you think, and she said to me, It's easier than working on a theatre stage, because on a theatre stage you have to remember to play out to the audience, you can't always play off of your actors. Here I can play off of my actors and I can move and do what I want. It's liberating and I know that what I do, you will realize," he adds.

The Polar Express, *Monster House*, and *Beowulf* all had performers wear facial markers. It's just a matter of taste, as we did give Zemeckis the choice of having a helmet with cameras. He thought actors wouldn't like it and it would hurt the facial performance. The drawback of not using helmets is in the capture volume size, as cameras surrounding it need to be able to see all the facial markers. Zemeckis was happy with this compromise because he wanted a small, intimate space where he could interact with the actors better. Cameron's style is different in the sense that he likes to edit his film as he's shooting it, in fact, he would stop shooting for days or weeks to edit what was shot and then he would continue shooting again. That is why cameras had to be already done, and he needed to have body and facial real-time feedback as well in order to build his cut so early in the process.

"We stand on the shoulders of those that came before us. We ride on top of what [*The*] *Polar Express* did and *Polar* was an incredible first step in this arena," he concludes.

3

THE MOTION CAPTURE SESSION

CHAPTER OUTLINE

This chapter covers everything that pertains to the motion capture session, starting with the decision whether to use motion capture or not. It is important to learn how to make that choice in an intelligent manner so that it doesn't backfire later during production.

Deciding to Use Motion Capture

A client has sent storyboards and character designs for your review. You are supposed to come up with a budget for a digital character performing certain actions. The project does not need to be rendered in real time. The client has a very small budget and there is no way you'll be able to compete with other studios if you use keyframe animation. You don't have a lot of experience using captured motion data. Do you suggest performance animation immediately?

If you said "no," you're right. This scenario has been proven to result in chaos time after time. It is safe to assume that in most cases performance animation will not save time, except when used to create a product with expected quality trade-offs, such as a motion library for an interactive video game, a virtual reality experience, or a live television show. Similarly, it is safer to believe that it won't save you money. I'm not saying that it is impossible to save time and money using motion capture, but that you should never assume it will unless you have demonstrated this by testing. Testing, of course, will cost time and money. What you should do in a case such as the one outlined in the previous paragraph is propose exploring the possibility of using performance animation and then proceed cautiously.

I believe that a decision to use motion capture has to be initially based on look alone. If the realistic look of human motion is not what you want, do not even consider motion capture. This is the first bridge that you should cross. Once you have determined that this is the look you are aiming for, you need to start worrying about the second consideration: How can you capture reliable motion data for your project?

Stage 1: Do You Want Realistic Motion?

It is sometimes difficult to visualize what your character will look like after realistic motion has been applied to it. If the character is not shaped like a human, the motion might look strange and even disturbing. It is impossible to predict if this will happen because it depends on the character and the performance. Even if the character is human shaped and the data is clean, the animation might look weird if the context is not right. For example, my first tests using optical motion capture were done with sample data of a dancer's performance captured with an optical system, which I imported into Side Effects' Prisms. I applied the data to a human skeleton that I had from a previous job. I didn't want to acknowledge it then, but the skeleton motion looked strange. I wasn't used to seeing that kind of motion associated with a make-believe character. Nevertheless, I was still awed by the result and decided to look further into the technology.

On one occasion, we did a test for the now-defunct Boss Film Studios in which a chimpanzee had to move along a branch of a tree, holding on to it with its hands and swinging its legs. The purpose was to create a demo for a feature film client. During that process, we would find out if a human could perform the animal's motion in a convincing way, and also if we could collect motion from a chimpanzee.

In preparation for the test, we found an animal trainer who had two well-behaved chimpanzees with acting experience, and a very flexible human gymnast. Our motion capture stage supervisor spent time with the monkeys at the suggestion of the trainer, helping them familiarize themselves with the markers that they would have to wear during the session. At some point, he had problems with the monkeys putting the markers in their mouth. This could have been dangerous because the markers are covered with tiny crystals that, if swallowed, could cause internal bleeding, but after a couple of days the trainer was able to control this behavior and we thought we were ready for our session.

The day of the session, Boss brought to our studio a structure that would represent the branch. Before starting the capture, we had the chimpanzees rehearse the motion without the markers; this rehearsal was without major problems. The chimps were very friendly and even allowed all of us to take photos with them. Then, the gymnast performed the motion and we collected her data. It looked very convincing as we saw it at the studio.

After a few takes with the gymnast, it was time to put markers on the first chimpanzee. We wanted to use the bigger chimp because the markers could be placed farther away from each other. We had made Velcro belts that would strap around his arms, legs, neck, and chest, plus a little cap with markers for the head. Our stage supervisor tried time after time to put the markers on the monkey with the help of the trainer, but the chimpanzee wouldn't allow it and broke several markers in the process. We decided to try using the small chimpanzee. He was more docile and accepted the markers without problems. He came to the stage and performed the motions several times, following exact directions from the trainer. He was better than many of the humans we had captured over the years.

The data postprocessing was no walk in the park. We had minimum problems with the human data, but the animal data wasn't totally clean. The markers were very close to each other in some areas and the software couldn't follow them very well, so a lot of operator-assisted tracking was required. When all the data was clean, we noticed that the Velcro strap in the chimpanzee's belly had been sliding around. Fortunately, it had two markers, and it didn't really matter in what part of the perimeter of his belly the two markers were because we could always calculate the center based on their diameter. We managed to stabilize this data as well, and it worked.

We converted all the data to every available format and delivered it to Boss. They in turn gave us a chimpanzee mesh they had created. From this point on, we would do our own tests and they would do theirs. I used Prisms and Softimage software to do our tests, and I found that although the human motion looked convincing when I saw it performed, it didn't when it was applied to the monkey mesh. However, the monkey data looked perfect. The monkey with the human data looked like a chimpanzee in a human suit.

The problem with animal performance is the same as with human performance: The talent has to be able to perform in a convincing way. If a particular motion is not feasible for an animal to do, then you cannot capture it. You can do small enhancements after the fact, like increasing the distance of a jump, but not much more. Animals cannot perform a part other than their own, and humans cannot perform convincing animal motion in most cases. Recall the use of motion capture in *Godzilla,* discussed in Chapter 2. The production of the digital character was initially based almost totally on performance animation, but later in production it was reworked to mostly keyframe animation because the final character looked "like a guy in a Godzilla suit." The special effects producers of *Mighty Joe Young* also considered performance animation, but did the right thing by testing before production actually started. They had a better excuse for considering it, because their character actually *was* a guy in an ape suit for most of the film. They decided to drop the motion capture idea because there was no easy way of modifying the captured data. Because this decision was made before production started, they managed to avoid a costly mistake.

As a rule of thumb, you should not use motion capture to animate characters that should have cartoon-style motion. To be more specific, take Disney's 12 basic principles of animation as outlined by Frank Thomas and Ollie Johnston. These principles are standard learning material for character animators and should be taken into consideration when deciding whether to use motion capture. Some of them cannot be achieved by a performer no matter how talented, and others can be easily accomplished realistically with motion capture. The principles are as follows.

- *Squash and stretch.* The character goes through extreme shape changes but maintains its volume. This is the first principle that cannot be achieved by a performer. Some people have attempted to add this property to captured motion data either by hand or procedurally, but the results have not been promising.
- *Timing.* The performance, whether animated or acted, has to have the right timing to convey the necessary perception.
- *Anticipation.* Anticipation is an indication of an action to come. This is typical of cartoon characters and not necessarily of human performance, but in some cases it is consistent with realistic actions, such as bending your knees before jumping. A good performer can show anticipation to a certain degree, but is limited by the laws of physics.
- *Staging.* A principle of filmmaking in general, the layout of the scene and positioning of the camera and characters are equally important in animation and live-action performance.
- *Follow-through and overlapping action.* Follow-through is the opposite of anticipation. The reaction happens after the action, such as bending the knees as one reaches the floor after a high jump. Again, a live performer can accomplish physically feasible follow-through. Overlapping action is inherent to live performance, but in animation, it is easier to start an action after another one is finished, resulting in either paused or rigid motion.
- *Straight-ahead action and pose-to-pose action.* These are two animation methods. Straight-ahead action calls for the animation of a scene on a frame-by-frame basis, whereas pose-to-pose action entails the creation of key poses scattered over time periods. The frames in between these key poses are drawn later. In computer animation, most of the character work is done using a variation of the pose-to-pose action method, creating key poses for different parts instead of posing the whole character at a particular frame. This is done by creating keyframes and letting the software produce the in-between frames by some kind of interpolation defined by the animator. This method is easier to manage because there

is less data to deal with. Motion capture is completely straight-ahead action; as such, it generates keyframes at every frame. This makes it very difficult to modify. Pose-to-pose action can be achieved through motion capture by selecting significant keyframes, deleting the rest, and allowing the computer to do the in-betweening as before.

- *Ease-in and ease-out.* It is very rare for an object to become active without a period of acceleration, or to become static without a stage of deceleration. Ease-in and ease-out are principles based on real-world physics, so they can easily be achieved by capturing the motion of a live performance.
- *Arcs.* Most actions are not linear. When animating, you almost never want to use linear interpolation between keyframes. This is another principle aimed at emulating realistic movement, which can be represented as a set of different types of arcs. When using keyframe animation, these curves are usually smooth between keyframes. With motion data, however, they are coarse and noisy, representing the natural nuances of realistic motion.
- *Secondary motion.* When animating, you first create the primary motion of the character, which is usually the motion of limbs and face. You then create the motion of other parts or objects that react to the primary motion, such as hair and clothing. Secondary motion represents a lot of extra work with hand animation, whereas with motion capture it is a part of the performance. One has to be able to collect it, however, which may not be possible with some systems. For example, clothing motion can be captured by an optical system if markers are added to the clothes, but an electromagnetic tracker or electromechanical suit would not be able to collect that kind of data easily.
- *Exaggeration.* The principle of exaggeration implies approaching or crossing the boundaries of physical reality in order to enhance or dramatize the character's performance. You must decide if capturing a live performance would be acceptable, or even feasible, for the level of exaggeration needed.
- *Appeal.* The audience must find the characters interesting and appealing. This principle applies for both live action and animation.
- *Personality.* Two identical characters can appear totally different by conveying different personalities. This is a principle of acting in general that should be applied to animation. When using motion capture, this is the number one reason to use a talented performer, as opposed to just anybody who can move.

The following principles of animation cannot be accomplished with motion capture:

– Squash and stretch
– Anticipation beyond physical boundaries
– Follow-through action beyond physical boundaries
– Exaggeration beyond physical boundaries

The following principles of animation are natural to live performances:

– Overlapping action
– Straight-ahead action
– Ease-in and ease-out
– Arcs
– Secondary motion

Finally, the following principles of animation require work whether a character is animated or performed:

– Timing
– Appeal
– Personality

Procedural and manual methods exist for adding some of the principles to motion data after the fact, such as squash and stretch, anticipation, follow-through, and exaggeration. The question is, Why would you want to capture realistic data if you want a cartoony look? Modifying captured motion data by hand can be more expensive than keyframe animation, and a procedural solution usually doesn't yield an interesting performance. Of course, there are exceptions, but you shouldn't rely on them.

I'm not saying that you should never capture human motion and apply it to a cartoony character, but it is safer to go through this process as a test to find out if it will yield the result you want before you commit to a high expense.

Some examples of high-end performance capture used to animate nonhuman characters in feature films are *Monster House* and *Happy Feet*. *Monster House*'s characters are humanoid and in my opinion they were successful in achieving the look that was intended, although it isn't the same style as hand-animated characters in a film like *Shrek*. *Happy Feet*'s characters are penguins and normally I would think that motion capture wouldn't be the right methodology to use for such a project; however, the performances weren't overacted as it happens with most performance capture characters and the penguin characters weren't based on realistic looking penguins. Also, the main focus of the film was in the musical performances, which were very compelling. That and an excellent story helped a lot in the project's success, so much that it won an Academy Award for Best Animated Feature.

Many TV cartoon characters have been animated through performance over the years, mostly with real-time feedback systems in combination with other controls. An example of such a TV show is *Jay Jay the Jet Plane.* These kinds of projects are usually created by studios that specialize in that kind of medium and have experience with real-time character puppeteering. Companies such as Jim Henson, Modern Cartoons, and the erstwhile Medialab and Protozoa created a business from the real-time rendering of characters and established it as a medium separate from the rest of character animation; it is not accurate to call this medium *motion capture* or even *performance animation,* because it involves many kinds of manipulations other than performance. This type of animation is called *digital puppetry,* and it involves a different decision-making process from what I describe here.

A good example of a project in which human motion data was applied to a cartoony character was the 1996 set of spots that TSi did for the Pillsbury Doughboy, which are explained in more detail in Chapter 2. We decided to use captured motion data on the Doughboy because (1) we determined that his motion didn't require any squash and stretch or extreme exaggeration or anticipation, and (2) we supplied our client with a test of the Doughboy performing extreme motions mapped from a human, so they knew what to expect. Some of the shots were not well suited to the subject, but we knew we could collect the data anyway and use it as timing reference. As long as the client did not change the timing of the motion we would be fine, and they agreed not to do that.

Stage 2: Can You Successfully Capture the Data?

I will start by separating the two kinds of projects that you are likely to deal with. Film or video productions and game cinematic sequences follow a predetermined continuity, so I refer to them as *linear* projects. On the other hand, user-controlled video games are *nonlinear* projects, since the player's actions determine the sequence of events that, in real time, shape the character's motions in different patterns. These two types of projects are handled differently in some aspects when it comes to performance animation. There is a third type of project that I do not cover specifically: real-time digital performance applications, such as live television and trade-show performances. These kinds of projects have to be based totally on performance animation. I do feel, however, that the concepts I outline here can be used at the planning stage of such a project.

If a character design is already available, and it has been determined that realistic motion is the way to go, it is time to evaluate if it is possible to capture the data you are looking for. For this purpose, I like to use a divide-and-conquer approach, breaking the project into small, similar groups of shots and using these categories to evaluate strategy and plan the sessions. A *shot* is an uninterrupted camera take that is later combined with other shots to form a scene. It will fit our purpose to define a shot as an uninterrupted piece of character motion, because this definition applies to all media, including linear and nonlinear projects. A step-by-step explanation of how to perform the evaluation follows.

Create a Candidate Table

I'm assuming that if you are planning a motion capture session, you already have a script, production boards, or a game design from which you are supposed to determine which characters to animate via performance. For the characters you are considering, you have already concluded that realistic data is the way to go.

The candidate table is a preliminary step that is designed to save you time, helping you avoid creating storyboards for shots that are not even in consideration. As you create the table, define if a digital character is supposed to be visible and decide how it is supposed to be animated, if at all. For example, if a character is talking, it is possible that it isn't facing the camera. Similarly, a character may be so tiny in the camera viewpoint that you may not have to animate it at all, or maybe you only see the character's foot in the shot, so animating the rest would be irrelevant.

If all you have is a shooting script, you should be able to extract and list those shots that appear to be candidates for performance animation. You will be lacking valuable information, as no visual reference is yet available, but you must assume that if the digital character is present in the shot, it is visible. If possible, talk to the director to find out what he has in mind for each shot. Later in the production, shot breakdowns and production boards are created; as you get hold of them, you will have to revise the candidate table based on this new information. If the project is a nonlinear video game, the design must already include a list or a flow chart of motions for each character. Such a list is required before starting to create the table, because no script is available for a nonlinear project.

The table should include at a minimum a shot name (which will be used to designate the data file as well), a list of the character(s) involved, and a description of each shot. You can also include a client reference name, which could be the board number or a page number in the script. You don't need to include other information such as timing just yet, since it is not the purpose of this list to calculate costs and schedules. A *Categories* column is also necessary, but it should be left blank for now. I will get to that later.

Table 3.1 is a fictitious example of what an actual candidate table would look like. The candidate table is not supposed to be a final table, so don't be afraid to include shots that are questionable. Be careful not to exclude any shots that have any possibility of qualifying.

I didn't have the luxury of knowing about this method when I was starting, and I often ended up capturing a lot of data that was discarded for a variety of reasons, from not being able to track the desired motion to not being able to use the data in the shots. As you become more experienced, more of the shots cataloged in the candidate table will actually make it to the motion capture session because you will be able to tell which ones are not feasible without going through the whole evaluation process.

Prepare Motion Capture Blueprints

Storyboards are commonly used for representing and understanding live-action and animation shots, but performance animation requires more information than storyboards alone provide. You need a certain kind of blueprint that includes storyboards and much more. You need to know what your subject is interacting with, starting from the ground itself to other characters and props. You also need to know the placement of

Table 3.1 Sample Candidate Table

#	Name	Client Reference	Characters	Categories	Description
1	SO-02	p. 1-3	Stan, Ollie		Ollie is sitting in the dune buggy, driving recklessly. Stan is being pulled and is skiing behind it. Camera is in front of car.
2	SO-04	p. 2-1	Stan		Stan hits a dune and flies in the air, losing control. Camera follows from the side.
3	SO-08	p. 4-2	Stan, Ollie		Stan and Ollie sit comfortably in car, enjoying their drink. Camera is in front of car.

each of these interactors. If the floor is uneven or sloped, you need to know. If the character leans on the wall or touches anything, that is also important information. So are measurements of all props and the volume that will envelop all the action. All these variables may seem unimportant at the stage of planning a shot, but at the time of capturing and importing data into a character they become big issues that one ends up wishing had been dealt with at the beginning.

For nonlinear applications, you most likely don't need some of the information I describe in this section, especially the camera orientation, because the camera will be controlled by the user. You do need other items, however, such as a list of shots that will be tied with the shot in question, because you will want to make sure that the attachment point is similar for all of them.

The blueprints can be created in a database, spreadsheet, or project management software and should include the following items:

Shot Description

A small description of what happens in the shot.

Client Reference Name

A name by which your client refers to this shot (if it applies).

Character Names

This is a listing of all the characters that will be captured simultaneously or separately in this particular shot.

Storyboards

It is imperative to know exactly what has to happen for each shot that you are evaluating. The best way of doing this is actually videotaping the shot using live actors, but this is quite expensive and in some cases impossible; thus, the most common way is to use storyboards. Whether you create them or your client supplies them, they should be detailed enough that you should be able to visualize the shot perfectly.

Some of the things that depend on good understanding of the performance are the sample rate at which it is best to collect, the placement of the markers or sensors on the performer's body, the character setup, and the topology of the 3D model. In addition, if you will be using an optical system, this information will affect the number of cameras to use and the placement of cameras according to the area of capture.

A typical storyboard is a drawn representation of a scene, which is a combination of shots that take place in a particular setting. A shot is represented in a storyboard by one or more drawings that are commonly used to block character movements and camera work in relationship with dialogue and timing. The storyboards needed for motion capture pertain only to a particular shot and not to a whole scene, and they need to be more specific for complicated motions, sometimes showing various views of the action.

Shot Timing, Including Handles

Motion data can be expensive, so you don't want to capture more data than you need. Make sure that your timings include *handles,* that is, a few frames at the beginning and a few at the end. The number is up to you, but it should be anywhere from a third of a second to a couple of seconds. I measure it in seconds because the number of samples per second will vary according to the type of project or motion capture system used. To obtain the frame count, you can always multiply the timing by the sampling rate. If you use a motion capture service provider, this number can be key to avoiding cost surprises.

It is very common when designing video games to create a list of motions and just capture them without going through this step. In many cases, I would receive a motion list without timings or handles and be asked to quote the cost of capturing the data. I would calculate the cost, making sure to try to time the shots with reasonable handles, but the difficulty of envisioning what the client had in mind occasionally resulted in very different final costs from what was estimated.

It is also common to slap timing on a shot without even thinking about it. For example, I once had a client who wanted to collect data for an ice hockey game. I asked him to provide in advance a list of motions with timings, but all the timings he provided were inaccurate because most actions on skates require a certain degree of build-up, which translates into long handles. You cannot start a skating cycle like a walk cycle: You have to slowly pick up the right speed until you have the speed and rhythm you need. This was one of my earliest motion capture jobs, my first video game job, and perhaps the first motion capture session ever done on a skating arena, so I couldn't detect the problem immediately. I figured we could cut the files to the right length after the session, but this became impossible because we couldn't decide what segment of each shot to delete without consulting with the client, who happened to be based out of the country, and we were afraid of deleting the point where motions would tie into each other. Also, we had close to 500 files, so it would have been impossible for the client to help remotely. We decided to track most of the data, removing only parts in which the skater left the capture volume. The resulting cost was about triple what I had originally estimated, and although we had only provided an estimate, the client refused to pay the extra amount.

The Measured Boundaries of the Performance

Make sure you know the volume and area limitations of the motion capture stage. Most systems have a maximum volume that cannot be breached, so you need to ensure that each shot will comply with it. With some systems, you can sometimes play with the volume to add area in a certain plane while decreasing the area in another plane, maintaining the volume constraint. For example, the capture of a slam dunk will require much more height than the capture of a baseball batter, so you may want to trade some stage area for more altitude.

When using an optical system, there is always an optimal camera placement for each motion, and sometimes it becomes necessary to switch configurations in the middle of a job in order to obtain better and less expensive motion data. Similarly, with a magnetic tracker, you need to make sure the action will happen within the electromagnetic field, so the placement of the transmitter needs to be planned in advance. It is important to limit your setups to a minimum as it takes a long time to calibrate and relocate equipment. There is always a balance that will achieve the best efficiency and data quality.

Including all the performance's measurements in the storyboard will ensure that you can divide your shots into groups based on stage volume and configuration. This makes the tracking postprocess more efficient. If you don't include the measurements, the stage will usually be set up for the maximum possible capture volume. It could be very time consuming to reconfigure a setup, especially if it is optically based. Nevertheless, if you are well prepared, a long session can be divided into two or three different configurations.

The ice hockey game discussed earlier is a good example of how one can benefit from good planning. The scenes had very low and tight motions and very wide and high motions. Most of the goalkeeper actions could be collected with a very tight camera setup, which made the postprocessing stage easier. Other motions (such as skate cycles) required a huge volume because the cycle itself used up a long distance. The height had to cover a tall skater, his skates, and sometimes even his arm raised with the hockey stick in it. Of course, we didn't have to collect the whole stick since it was a rigid object, so we used three markers close to the handle that would represent the whole thing. However, it still was very challenging to collect all the necessary data. We certainly had a hard time cleaning up those motions. Had the client provided us with the critical timing information, we could have optimized our volume. A corridor-like configuration (a narrow, but long and tall, volume) would have been appropriate for this situation. It would have saved a lot of time and money in the postprocessing of the data.

There are cases where you would want to have different setups all installed and ready to go. For example, during *The Polar Express* we had three motion capture stages. The first stage was only a small cube with 10 ft per side. It was meant for all the close-ups and facial performance shots. Around the volume we had 70 cameras with long lenses to cover all the facial markers and about 10 more with wider lenses to cover the body motions. The second stage was a larger (30 × 60 × 10 ft) volume that was used to capture wide shots and performances that occurred in a large area, like a long walk or many characters interacting in a wide space. The third volume was a taller volume (30 × 30 × 30 ft) that was used almost exclusively for stunts that required rigs and other props. The second and third volumes had about 100 cameras each.

Since we were shooting only in one stage at a time, why would we want to have three available setups? Well, it had to do with cost. Surprisingly, it was less expensive to have three stages ready to go than having one stage and having to reconfigure it several times and have the hyperexpensive cast and crew waiting.

Character Setup

Character setup and marker setup are two different things. Marker setup pertains to the locations of markers, sensors, or tags that are used to collect data. Character setup deals with the locations of joints or bones in the body of a character that will provide the final motion and deformations. This is usually done using the 3D animation software, such as Maya or 3d Studio Max. The character setup depends on the marker setup, because the data collected must be enough to calculate all the information needed by the joints. For example, if you have a knee joint in your character setup, you will have to add the necessary markers at collection time to calculate the rotations of the knee.

You don't need to determine a marker setup at this time, but you must prepare a character setup design that will later allow you to come up with a marker configuration for your character.

Only a design is necessary since in a perfect world you will not actually model your character until after the motion capture session. Even if you already have a character, it shouldn't be set up for motion until the captured base position is available.

If your system can only handle one marker configuration, then your character setup will have to be designed based on this setup constraint. However, if you can design different marker setups, then you can work the opposite way, which is much better. With an optical system, you can define where you put the markers and you can use combinations that will allow you to determine the motion of any location on the subject that you may want.

World Axis

It is important to maintain a common world axis throughout the project, especially if you are capturing data from several characters that must interact in some way. Always use the same orientation and position of the world axis as in your target animation software or game engine. Looking at the front viewport in most off-the-shelf animation programs, the positive *Y*-axis is oriented toward the top of the screen, and the positive *X*-axis is directed toward the right side of the screen. The positive *Z*-axis is sometimes facing the outside of the screen; other times, it points toward the inside of the screen. These two variations represent right-handed and left-handed space, respectively.

If your software uses right-handed space, you would want your talent looking toward the positive *Z*-axis as you collect the base position (a file in which the talent is standing in a pose similar to the 3D character's neutral pose). For left-handed space, the performer should be oriented toward the negative *Z*-axis.

Some off-the-shelf animation programs use left-handed or right-handed space with variations in world axis orientation. A good example is Maya, in which you can specify either X, *Y,* or *Z* as the up axis.

Try to assume you will place the axis at the origin and use it as the root of your character's setup. If necessary, you can add a translation offset in case you need to place the character elsewhere in a set.

Measurement Units

You can add a scale offset to the world axis in order to size the character proportionally with other elements in your project, but proper planning calls for uniform units across the project. It is best to base digital models on real-world units, especially when using captured motion data. Make sure to specify the measurement units you'd like your data to conform to and use those units when modeling characters, sets, props, and any other elements in your shots, especially if they will interact with the digital character. For example, if the character has to avoid certain digital obstacles, you could conceivably mark the position of the obstacles in the capture area. This is not guaranteed to work exactly as planned because you still have to deal with character versus data proportions, but it will definitely be close.

Obviously, having the right units doesn't mean that your captured data will match perfectly for a specific character. You may have to manipulate it before and after applying it to the character, but at least you will have a common starting point for your whole project.

Main Camera Angle and Framing

Unless your data is to be used for a nonlinear application, chances are you will have a camera angle that will be used for the final rendering of the shot. This is important for two reasons. First, when you capture the performance, you always want to have a taped reference. This applies to both linear and nonlinear projects. If the project is linear, it is always best to have at least one camera placed with the angle and framing of the final camera to be used. Of course, this rule probably won't apply if the camera of the final shot is moving, but the camera should be at least placed in the general direction. The taped reference will be used in some cases to decide between takes, or as a placeholder for editing purposes, and definitely as reference when applying the data to the final character.

Second, the performer needs to know where the camera is. When acting, you perform to the camera, which is where the audience is. If you don't know where the audience is, you cannot establish contact with it through body language, and the performance becomes flat. If it isn't possible to place the video camera in the general direction of the final camera, you need to let the performer know where that final position is.

The camera position should not be confused with the world axis of the motion. Camera placement can change from shot to shot, but all shots have to maintain a common world axis. Thus, when you capture a base position that will be used for character setup, it must always face your preferred front axis (see Figure 3.1).

All Interactions Explained with a High Level of Detail

Anything that the character touches or reacts to has to be accounted for in advance in order to decide if it is possible or feasible to use motion capture on that particular shot. Some shots will require more human involvement than others at the time of applying data to a character; that time is usually based on the number of interactions.

Let's imagine we are capturing a baseball batter waiting for a pitch. The first capture is the batter walking with the bat in one hand. In this case, the only interaction is the bat in one hand and the floor, but the floor is flat, so we don't worry about it. We put markers on the batter, including his hand, and two markers on the bat (except for the base, because we know it is in his hand). This type of object is called a *constant interactor,* because it remains for the length of the shot. These kinds of interactions are solved mostly by character setup only and can be occasionally simplified by using fewer markers during the capture session. Applying this data to a character is not plug and play, but it is fairly uninvolved, assuming the size of the character is proportional to the size of the performer.

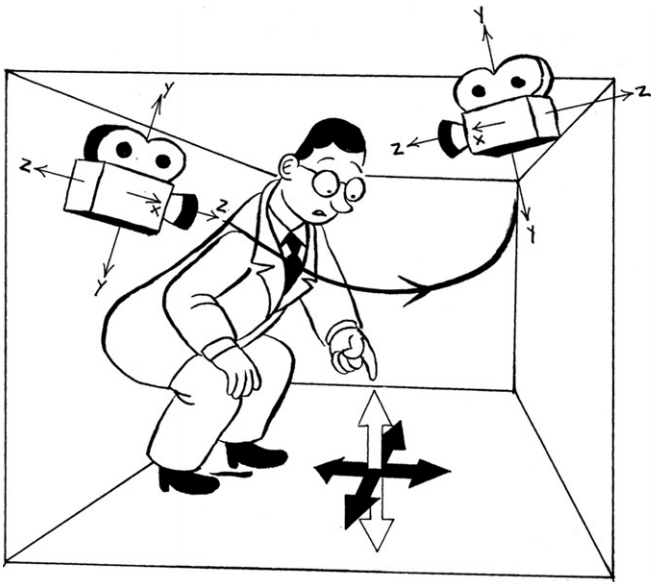

Figure 3.1 World axis versus camera placement.

The second capture is the batter putting the bat under his arm as he rubs powder in his hands. This situation is more difficult because there are two interactions to worry about: the bat under the arm and the two hands. In addition, none of them is constant, since the bat is first in the hand, then under the arm, and the hands are first apart and then together. These are called *variable interactors*, and we cannot use shared markers with them. For example, we cannot use the hand marker to capture the position of the bat anymore because the bat is not in the hand for the length of the shot. Therefore, we end up using three markers on the bat. At the time of applying the data, the bat most likely will not end up under the arm of the batter, so we'll need to either animate the motion by hand or add a parent joint to the bat that we can use to add variation to the original motion. We never want to modify the captured motion data directly. We'd also use markers on each hand. At the time of application, if the hands don't end up together due to size problems, we'll have to discard the elbow data and use the wrist positional data as a constraint for inverse kinematics goals.

The third capture is the following scenario: As the batter waits, he hits his shoe twice with the bat, holding it with both hands. Then he holds it with the left hand and passes it to the right one, starting to swing it into the ready position. He finally swings, hitting the ball. He drops the bat as he starts running. If we don't care about the bat after it leaves his hand, the interactors are as follows:

1. The bat in both hands
2. The bat in both hands and the shoe
3. The bat in both hands
4. The bat in both hands and the shoe
5. The bat in both hands
6. The bat in the left hand
7. The bat in both hands
8. The bat in the right hand
9. The bat in both hands
10. The bat in both hands and the back of his neck
11. The bat in both hands
12. The bat in both hands and the ball
13. The bat in both hands
14. The bat in one hand

You may think I'm being repetitive, but there really are 14 variable interactors in this shot, without counting the floor. Initially the batter has the bat in both hands, then he hits his shoe, then the bat is again in both hands, and then it hits his shoe again. It is back in both hands before it goes to the left hand, and as it goes to the right hand, it passes in between both hands again. Then it goes to both hands. Later it touches the back of the neck. As he swings, the bat is in both hands, then it hits the ball, and then it is in both hands again. As the bat is tossed, it leaves one hand first and then the other one.

Interactions need to be listed in this manner: cataloged as either constant or variable, and, if variable, in order of occurrence. You need to be very thorough in determining that all the interactions are listed, including the ones that are in-betweens. If you do this, it will be easy to estimate the operator time that will be used, and no money will be lost by miscalculating the actual cost of a difficult shot. It is even possible to come up with a cost-and-time formula based on the number of constant and variable interactors combined with the shot length.

All Props, with Size and Location

If your character must interact with any props, you have to specify the prop's exact size and position, based on the units and world axis provided. This is an area in which common sense needs to be used because there are many factors at play. Assuming you have a well-organized project in which all units are the same and in which the character's proportions match closely the ones of the performer, the correct placement of these items should result in a very close resemblance of reality once you translate the action into the digital realm.

When using a real-time capture device such as an electromagnetic tracker, you can usually place the digital prop along with the digital character for real-time feedback. The performer can interact with it by watching a screen. This technique avoids the need for creating a live version of the object for the performer to interact with.

In the case of non-real-time capture devices, such as most optical systems, there is no way of looking at the digital prop at the time of the performance. Therefore, a live stand-in object must be placed in the stage area that will represent the digital prop. This object could be anything from a replica of the digital prop to a painted area that represents where the object is located.

Depending on the complexity of the interaction, you must decide what level of detail is needed for representing the object. If the prop is a column that the performer needs to avoid without touching, perhaps a painted profile on the floor will be enough. Maybe a few ropes or thin pipes from the floor to the ceiling indicating the corners of the column are needed, depending on how close the interaction will be. Keep in mind that these objects could hinder the actual collection process by occluding markers or introducing interference.

Rigs

Some motions cannot be achieved by the performer alone, so special contraptions need to be created to help. These are not really props, because they will not form part of the digital scene. For example, a wire rig for flying requires the performer to wear a type of harness that will affect the marker configuration. Some rigs can even interfere with the data collection, so it is very important to be aware of this as you evaluate the project. If any of the shots to be evaluated require a special rig, you must figure out what that will be as you create the motion capture blueprint.

File-Naming Convention

When you're dealing with lots of motion files, you can't afford to be disorganized. Each shot that you capture requires a predetermined name and number that will give you exact information as to what character the file belongs to, what setup was used, the calibration file to be used when tracking, what base position the file is compatible with, and other specific items defined by the grouping that you will come up with afterward.

Since this is a preliminary evaluation stage and you probably still don't know most of the information, you can start designing the file names using the character or characters' names, a sequence number, and a description of the motion. The name will grow as you find out more information, and at the time of collecting the data, the full name will be listed here and used for saving the file.

If the data is for a client, you must find out their naming convention because you will have to establish a cross-reference table to be synchronized with them. Also, one shot in a production could represent more than one shot in the motion capture list, and vice versa; that relationship will have to be kept organized as well. Finally, if file name length is limited, a detailed index must be kept to correlate file names to their contents.

Dialogue

When capturing a facial performance, it is obvious that the entire dialogue is needed beforehand, but the dialogue also has importance when collecting full body data. In real life, body language enhances communication when one speaks. Thus, if a shot has dialogue, it is important to consider it when collecting full body data because it affects timing and performance.

The nuances that you are able to capture are so realistic that if you omit a simple item like this, the final performance will not look believable. The eye is so well trained for the real thing that a little deviation will nullify the whole effect.

Shots to Be Blended

When capturing data for a nonlinear project, each shot will most likely have to blend with many others. For example, in a video game, a player controls what the characters do, and the game engine plays back the motion corresponding to the player's action in almost real time, blending it with the preceding motion. If these two motions don't fuse well, the game loses fluidity.

A list of shots to be blended and a flow chart should already be included in a good game design, and this material should be transferred to the blueprint, making sure to specify if it should match from the head or tail. Whether a shot is a cycle or a loop must also be specified, because the shot's head must blend with the tail. The start and end positions must be specified in these kinds of shots, preferably by drawings or photos.

A shot in a linear project doesn't usually have to blend with any other motion, but there are special cases. If the motion has to occur in a volume greater than the one available for collection, it is possible to break the motion into two and blend it together later. On the other hand, if a production storyboard lists two or more shots that represent the same motion from different camera angles, they should be performed and captured in one step, splitting it later into different shots.

Number of Files

To be used later, this is the number of files in which this shot will be divided. It could be more than one file if there are multiple characters to be captured separately or if a particular motion needs to be collected in two or more passes.

Sampling Rate

Sampling rate is also referred to as *frames per second*. There are two kinds of sampling rates that you need to worry about: the capture sampling rate and the delivery sampling rate. Capture sampling rate is the frequency at which you collect the data, and delivery sampling rate is the frequency at which you apply the data to a character. At least the delivery sampling rate should be included in the blueprint. The capture sampling rate is possibly not known yet, but it will become important later.

If the project will be rendered at NTSC video resolution, the delivery sampling rate will probably be 30 frames per second. For film, it is likely 24 frames per second. The sampling rate for video games is defined by the engine capabilities and should be included in the game design. You may need the final data delivered at a higher sampling rate than the final medium requires in order to do certain operations, such as motion blur, field rendering, or time distortion effects. This data should be delivered at a rate divisible by the final product's rate. That is, if film is the final medium, the delivery sampling rate could be 24, 48, or any similar number divisible by 24, because it is easy to down-sample data when the divisor is an integer number. Converting film to NTSC video or vice versa always presents a problem because every 24 frames of film correspond to 30 frames of video. Add to that the fact that NTSC frame rate is really not 30 but 29.97. High definition can be broadcast in any of various different frame rates, some nice round numbers and some not. This is why having the motion capture system synchronized to time code is a must for most film or television projects.

Configuration Name

This field is for future use and will list the name of the marker configuration to be used. In large projects, it is common to have several different configurations, each of which will include a marker setup map and a base position file. You must keep track of what configuration matches each shot because you will need the information later in the process, especially during character setup and data application.

Calibration Name

When you collect data that will be tracked as a postprocess, such as with an optical system, you must take note of what calibration name pertains to each shot. At the time of data analysis, the operator will have to associate the calibration with the actual shot raw data file in order for the software to start tracking.

The calibration of an optical system is kept in a file and is used by the software to calculate the exact position of each camera in space. Without this information, it is impossible to calculate anything else's position in space. Several times during a daily session, a calibration file must be recorded by capturing data from a known object, such as a large cube with markers.

For systems in which the data is tracked in real time, it isn't necessary to save the calibration information because once the motion data is tracked, the calibration parameters are no longer needed for a particular shot. The exception in this case is when the real-time data is not captured at full quality and may need to be processed further.

This field will be left blank at this stage, as it will only be used at the time of data collection.

Talent Used

It is important to always use the same performer when you capture data for a character within a particular project. Otherwise, it would be like switching actors in the middle of a film. Later in the process, you will have a casting session, where you will determine the name or names to put in this field.

Special Instructions

Based on the number of interactors, camera framing, props, and other special considerations, here you will write notes that pertain to the collection, character setup, and data application.

You don't need to—and cannot—write all your notes at this point, but you will use this field throughout the project.

Among other things, this field's objective is to pass on pertinent information to the next levels of the project. For example, if you have a motion file in which, for session economy, you collected data for markers that are not supposed to be used in that particular shot, the technical director in charge of applying the data must be informed. Another scenario would be to pass specific instructions to the data collection and analysis team to follow at collection or tracking time.

It sounds like a lot of work, and realistically, you probably will not have all the information available in advance, but knowing the items that could be an issue later is important in order to finish a project on time and budget. If the project is large enough, it will make sense to prepare a user interface with a database to handle all the data. Figure 3.2 shows an example of a motion capture blueprint.

Catalog Your Shots

Once your motion capture blueprints are complete and you are sure that you have all the pertinent information about each shot, it is time to organize them in different categories. You can forget about the chronology of the shots, because you have already assigned shot and file names to each of them, so everything should come together nicely when you are done.

The object in this step is to group the shots into categories based on motion capture parameters that will allow you to organize your session, postprocessing, and data application, and to reject shots that don't meet your requirements. This is the final bridge to cross; any shot that survives will be captured.

First, you have to come up with categories based on the differences of the particular shots in your project. I cannot tell you exactly what categories these will be, but I can show you the general areas that are significant in order to help you come up with them. Second, you need to be able to sort your data based on these new categories. If you use a spreadsheet or a database program, you should be able to do this without a problem.

The listing of significant areas that follows is not in any particular order, as each project has different variables and priorities.

Volume

With any software you use it should be easy to produce charts that will visually help you identify the shots that either fall outside of the maximum volume or have deviations that would

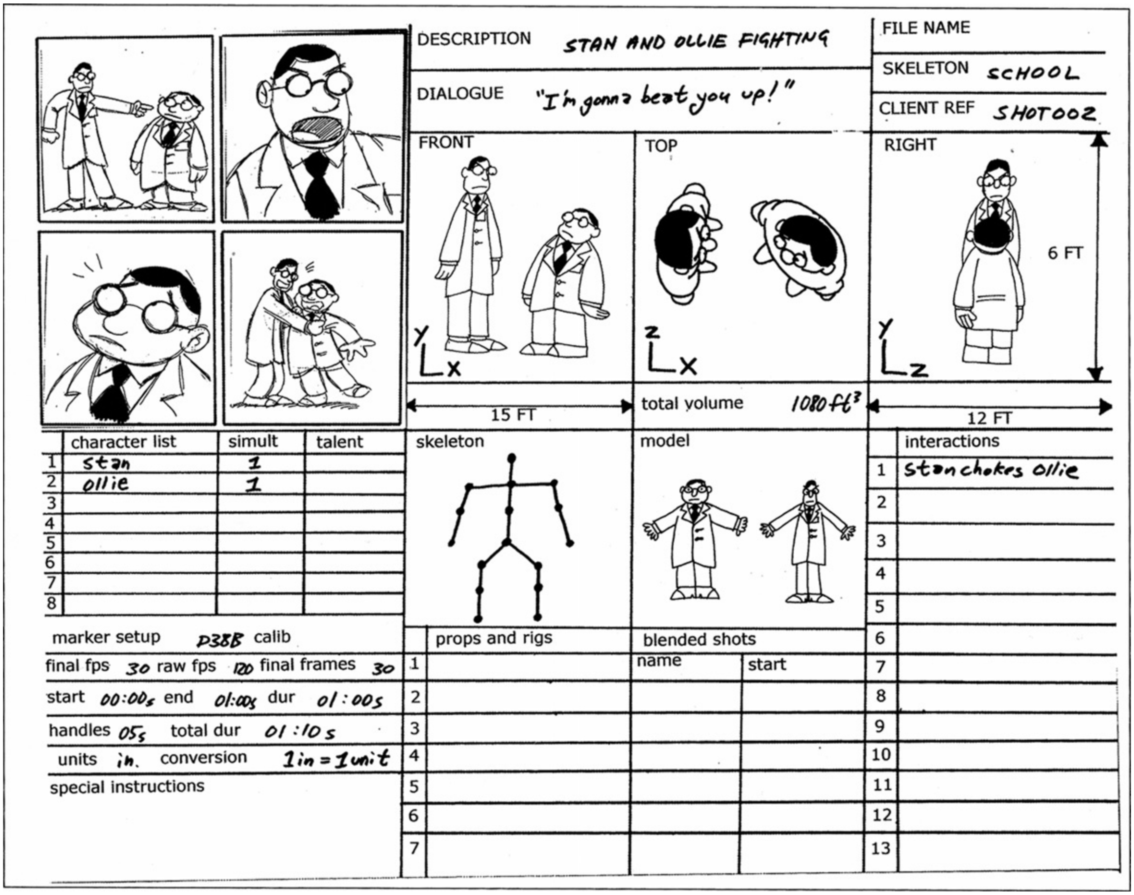

Figure 3.2 Sample motion capture blueprint.

cause them to become special cases of stage configuration. Let's look at the group of shots shown in Table 3.2. For simplicity's sake, Table 3.2 has only 17 shots, but the example can be applied to any project size. In addition, we will assume that the system in question can handle a maximum volume of 2000 cubic feet. You can reject shots that fall outside the range of capture of your particular system by charting the total volume per shot. The bar chart in Figure 3.3 shows that shot 9 surpasses the maximum volume; therefore, it must be discarded from the list.

For optical systems, with which the capture volume can be reconfigured, you can divide the remaining shots into volume groups. If all shots fall into a narrow volume range, this may not be necessary. Depending on the spread between the

Table 3.2 List of Possible Shots

Shot	Width ft	Length ft	Height ft
1	12	8	8
2	4	5	7
3	7	11	9
4	7	7	7
5	9	11	9
6	12	21	7
7	15	4	8
8	20	17	7
9	18	12	7
10	12	12	8
11	5	3	9
12	3	4	12
13	13	12	9
14	8	12	7
15	6	12	7
16	4	6	7
17	9	6	7

minimum and maximum area, you may have to divide your shots into two or three basic groups, adding special groups to encompass special cases such as corridors or tall setups.

A project with 500-plus shots could easily end up with three rectangular volume groups, plus several special groups. An easy way to see if you have any special cases that require additional configurations is to plot the width/length ratio and the height per shot. The chart in Figure 3.4 shows the proportions of the floor area of the stage. The perimeter values represent the shots, and the aligned numbers represent the ratio between width and length. The higher the value, the less square the stage needs to be. This value is given by the following code:

```
if (width/length < 1) {
   ratio = length / width;
else
   ratio = width / length;
}
```

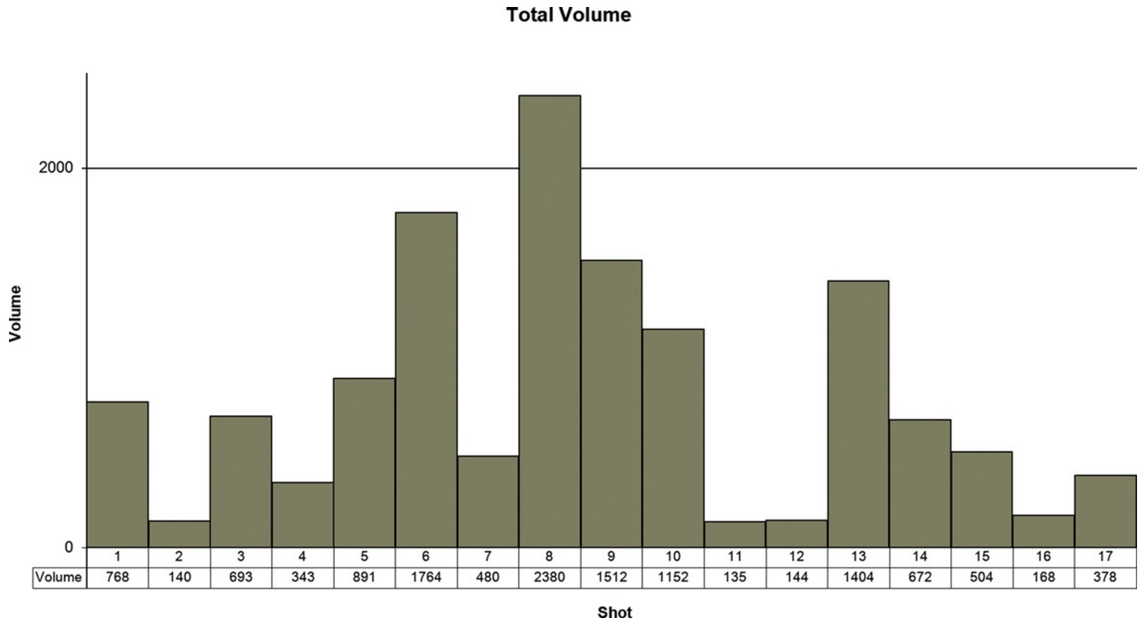

Figure 3.3 Volume chart.

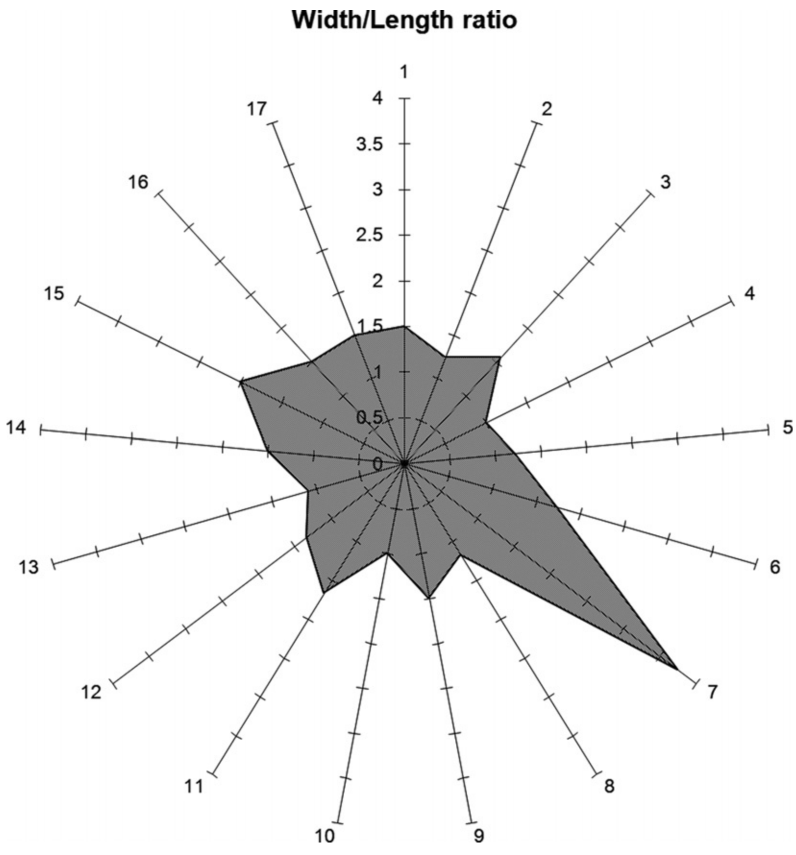

Figure 3.4 Width/length ratio chart.

Out of 17 shots, notice how 16 fall within the 1:2 ratio value. Shot 7, however, is noticeably different from the rest and will have to be cataloged as a special case, requiring its own stage configuration.

Figure 3.5 shows a similar scenario based on the height requirements of each shot. In this case, 16 shots fall within the 6–8 ft range, whereas shot 12 requires 12 ft. It is possible that this shot will require another special configuration.

Characters

A listing of all shots by character is always a good idea, especially for talent scheduling and rehearsing purposes. When scheduling your capture session, it may also help to calculate a certain performer's cost, or to calculate the character's total screen time.

Three very important categories that pertain to characters are multiple, split shot, and split multiple. A *split shot* is the type of shot in which more than one character is present, but which can

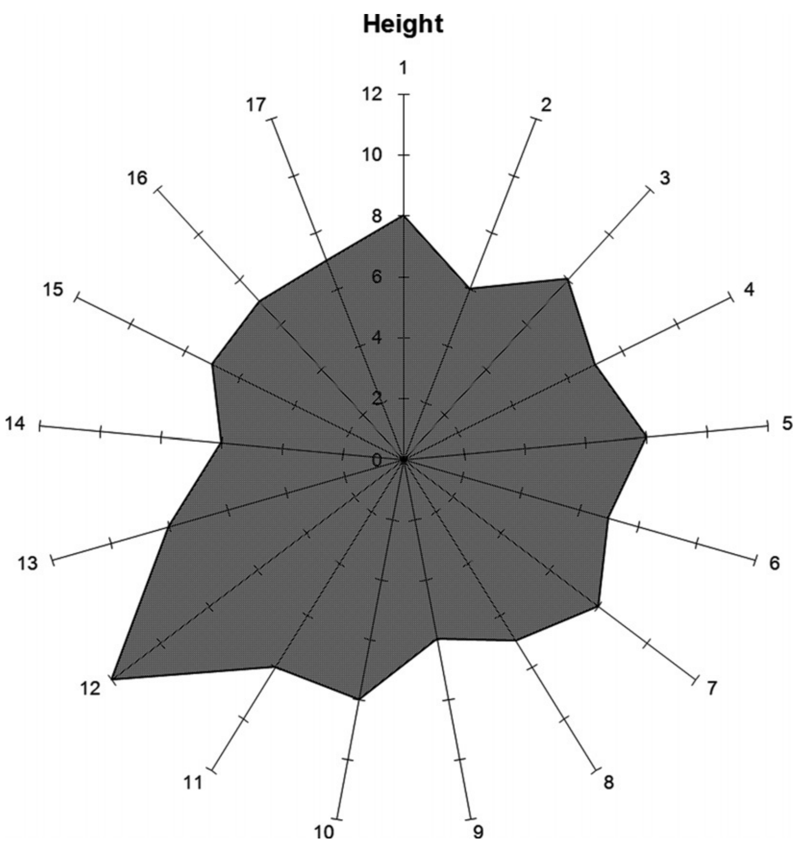

Figure 3.5 Height chart.

be split into two different subshots in order to collect the data of each character individually. When this is done, a postfix needs to be added to the shot name to represent the fact that it is divided. A shot with multiple characters can only be split in this manner if there are minimal or no interactions between characters. If there are minimal interactions, the timing between the performances might need to match perfectly, a match that will only be achieved by extensive rehearsing and maybe multiple data captures. Never assume that you can modify the timing of a performance after the fact.

Multiple refers to a shot in which more than one character's motion has to be collected simultaneously. There are shots in which characters interact with each other in ways that would make it impossible to split the shot into several parts. Think about it in terms of video. If you could shoot each character separately performing this action and then merge them, being able to reconstruct the action perfectly, then you should split the shot; otherwise, it falls into the multiple category. If you have an electromagnetic or an electromechanical system that can only handle one performer at a time and you don't have access to a second setup, you must now reject all the shots that fall into this category. With an optical system you have a lot more leeway, since you can add markers to more subjects; however, depending on the interaction, you will have to design creative ways of positioning markers so that they don't get occluded for lengthy periods of time. Most state-of-the-art optical systems should be able to handle various performers and props at a time.

Split-multiple shots are performed by multiple performers, not all of whom have markers. This is a good idea when timing is a consideration, and it helps the talent achieve a certain movement. This is done all the time, especially during sports games projects. A good example is capturing the motions of a soccer goalkeeper, because there has to be somebody shooting the ball. For a martial arts game, a martial artist performed certain moves, such as air kicks and flying turns, with the help of another artist without markers, who also served as the target of punches and kicks. We later captured the action of receiving these punches and kicks, again with both martial artists on the stage but with only the receiver wearing the markers.

Interactions, Props, and Rigs

You must have a category for each type of shot based on interactions. For example, all shots in which there are variable interactions involving one arm should be grouped together, as should all the shots that have constant interactions with a certain prop. You need to define what is the maximum number of interactions that you will have to handle manually and decide if it is worthwhile or even possible to capture that performance.

For example, suppose you have a shot in which a character has variable interactions with both his arms and legs. It is likely that you will have to eliminate the captured data on all those limbs and animate them manually. You still have the body left. It may be worthwhile for you to capture the data just to have the body motions, but if the arm and leg interactions will affect the body in any way, such as in timing, it is best to reject that particular shot. In most cases, the legs' motion affects the body motions, unless the character is swimming or sitting without his feet touching the ground.

Another example would be the baseball player mentioned previously, who moves the bat back and forth between his hands several times. This kind of shot should be captured, as the bat interactions do not affect the body data in any way. It will require human involvement to finalize this shot, but you can still retain a good amount of the data without modification.

Constant interaction with a prop can also be tricky. Suppose a character is climbing a rock of ambiguous shape. Unless you have a replica of the actual rock on stage and your character is proportionally equal to your talent, you are most likely looking at problems. You will have to turn off the legs' data and perhaps change the body's timing, and that makes that shot a reject.

Props and rigs can also be problematic without interactions if they prevent the collection device from capturing the data. This includes anything that would introduce noise or occlude cameras in a way that would make it impossible to collect the data. In any case, all props and rigs that interact, occlude, or even exist in a shot must be in a separate category.

Blending

Two categories need to be created based on blending, namely, head and tail blending. A particular shot could belong to either of these categories or both. This list will be used to make sure during the capture session that the shots have similar starting and/or ending points, keeping in mind that as the sampling resolution of a shot increases, blending will be more difficult and will require extra work after character application.

Character Setup

This category is particularly important, as it will be key to designing the marker setup. You may have one particular setup for each character or even a few different ones per character.

If you will do facial and body captures for a particular character, you need two separate setups. You must define one category per character setup. This will most likely yield a similar number of marker setups, but it may also result in more or less, so it has to be independent.

Imagine you have already designed one particular character setup in Softimage that is used in all your character's shots, but one shot calls for the character to sit on a couch. If an optical system is used, the couch will obviously occlude all the markers placed on the back of the performer, so they need to be moved to the front. The marker setup for this shot will be different, but in the end, the same character setup can be used in Softimage if the motion capture data analysis and conversion was done correctly. This book includes a practical example of this particular case.

If you have any kind of control over the placement of the markers, I advise continuing only if the character setup design is ready, because then the marker setup will be dependent on the character setup and not vice versa. Try to use this benefit to your advantage.

Priority

Some shots have a priority for reasons not related to the motion capture technical process. Most of the time these priorities concern delivery restrictions. In the past, my clients have divided shots into different delivery categories with different due dates. This would dictate the order of postprocessing that we'd follow.

Depending on the size of the project, you may have many categories to consider. Use a database or spreadsheet program to help. It will help you get organized and you will save money in the process. Make sure you have candidate tables sorted by every one of these categories. Doing so will help you come up with your final motion capture schedule. Also, make a single list with all the categories you defined. Table 3.3 shows the same candidate table as in Table 3.1, but with categories added.

You are now finished with the evaluation process, and much of the work you've done will prove very valuable at the collection and character-mapping stages. I'm sure you've also managed to reject some shots that you didn't consider as problematic to begin with. You should have rejected all shots that didn't meet the system's maximum capture volume, shots that had more characters than is possible to collect per shot and that were not feasible for splitting, shots with many interactions that affect

Table 3.3 Sample Candidate Table with Categories

#	Name	Client Reference	Characters	Categories	Description
1	SO-02	p. 1-3	Stan, Ollie	Split shot, Sitting setup, Skiing rig	Ollie is sitting in the dune buggy, driving recklessly. Stan is being pulled and is skiing behind it. Camera is in front of car.
2	SO-04	p. 2-1	Stan	Rejected	Stan hits a dune and flies in the air, losing control. Camera follows from the side.
3	SO-08	p. 4-2	Stan, Ollie	Multiple, Sitting setup	Stan and Ollie sit comfortably in car, enjoying their drink. Camera is in front of car.

limbs and body, and other shots that would have required too much human manipulation later. Now you can start planning your motion capture session.

Preparing for the Session

You have successfully prepared a list of shots that will be performed at the motion capture stage and are getting ready to schedule your session. Whether you have an in-house system or are hiring a service provider, some preparations are still needed. You need to find a good performer, organize all your props and rigs, order all your shots as efficiently as possible and, if necessary, come up with marker configurations that will fit your character's setup.

Using an External Motion Capture Service Provider

There are several service companies in the field of data collection. Some specialize in the entertainment business and others in areas such as sports analysis, forensic science, or biomechanics. Obviously, you would prefer to deal with a studio that has some experience in the field that will correspond to your project, whether it is a video game or a film effect shot. Also, you want them to have the kind of system that you think is best suited for your application, plus a capture volume greater than or equal to your maximum required volume. You would like them to be as close as possible to these criteria. In Appendix B, I have listed the primary service providers, the type of equipment they use, their specialty, the value-added services offered, and their supported animation and motion capture data formats. Most studios will provide you with sample data in the format you need. Some will even capture a free test if your project is large enough.

Find out if the studio can provide the data already mapped to your character model using your preferred software package, and if so, figure out to what level. Simply applying data to a character is very different from applying data plus solving all interactions and size problems. Applying data takes just a second once the character setup is in place, but all the extra tweaking that will be needed can take a long time. Make sure they have the qualified technical directors required to do this job. If you decide to have the service bureau do this work for you, you will need to provide them with all your models and character rigs, including props.

A state-of-the-art motion capture studio should have the ability to deliver time code along with your motion data, all synchronized with any reference video and movie files.

Depending on who your talent is, you may want to choose a studio that can provide services such as catering, green room, and other on-set amenities. If this is not important, you shouldn't have to pay for it, but you may want the studio to help with casting. Most studios have a good list of performers who are familiar with motion capture. Also, some studios will provide props and different kinds of rigs that you may need, such as a boxing ring, harnesses, and weapon props. Some even have the capability to build set pieces for you.

If your intention is to record audio during the shoot, the service bureau will be required to either have a sound proof stage, or be able to relocate their equipment to one. Most studios don't have soundstages but are able to bring their equipment anywhere you may want to shoot as long as it is a proper environment for their equipment, and, of course, at an extra cost.

When you negotiate a price with the service bureau, make sure you show them as much information as possible to avoid misunderstandings. Have them sign a nondisclosure agreement if necessary, but make sure they know what they're talking about when giving you a quote, especially if it is only an estimate. It is very common for a client to call a motion capture studio and say, "Give me a ballpark figure for 200 moves for a 3D shooter." Based on their experience with these kinds of game projects, the studio may give you an estimate that could be close but could also be totally off. After doing dozens of these types of games, I've learned that there's nothing standard about doing game character's motions. They no longer involve just walk and shoot cycles, but very complex movements that designers introduce to make their games different from the rest. Sports games are no exception, and some get extremely complicated.

If your capture project is for a linear animation, you will probably require more services from the studio because these types of projects tend to have more time, and revisions are very usual. In addition to your main studio session, try to allow your budget to cover an extra maintenance date, because you will probably have to modify your animation. It is always better to collect the performance again than to modify captured motion data.

When the data collected is to be used on a nonlinear project with a long development schedule, it is wise to break the capture schedule into two separate sessions. That way, you can correct errors from the first session that will become known only after data was applied to the character models.

Different service bureaus have different billing practices. Some studios charge for the day and the price includes studio and data. Others charge a bulk rate per project; still others charge a fee for stage time plus postprocess operator hours, but give you an estimate before the session. I prefer the method of charging an hourly fee for the stage, plus a fixed rate by the finished second of data. As a client, you thus know exactly what you will pay, and it is fair to the studio because they will set the fee per second based on the type and complexity of the project. Paying a bulk rate is also fine, but because it can be risky for the studio, it usually is higher than other billing plans. The daily charge is usually reserved for real-time systems, where you can walk out with your data when the session is over.

The worst type of billing, in my opinion, is by operator post-process hour. Even if you get an estimate at the beginning, it doesn't mean the studio is obligated to charge that amount, and, believe me, it will almost never be less than the estimate. Try to avoid this system at all cost if you don't want to be horrified at invoice time. This type of billing puts you at the mercy of the studio, and, even assuming that everybody is honest in this world, you want to know in advance what you will be paying for a service. In addition, the studio may give you a low estimate to ensure you will come there for business.

You don't want to do business with a motion capture service bureau that doesn't deliver the quality of data you expect. A demo session is always recommended, but if you couldn't arrange for one, you should, at a minimum, arrange for the studio to provide you with sample data files in the format(s) that you will be using for your project. Unless your final animation solution is proprietary, the service bureau should be responsible for delivering data compatible with your off-the-shelf animation software. When you inspect the data, make sure it has no imperfections, such as sudden pops and snaps, and that it isn't noisy, but fluid and realistic. Remember, if this is a sample file that the service bureau is distributing to clients, it probably represents the best quality that the studio has ever achieved, not necessarily its average quality, which is why a custom test is always better.

The inspection of the data should first be carried out visually, using a simple object such as a stick figure or just a skeleton within your animation program, making sure when you set it up that the data is not being scaled or modified in any way. You should also look at the data itself with the aid of a curve visualization tool, such as the one included with any off-the-shelf animation program. In rotational data files, look for sudden multiple of 90° jumps in rotation angles for any axis. You may think a 360° jump from one frame to the next will not cause any problems because the axis's orientation remains the same, but when you start tweaking the data—especially if you want to blend between motions or reduce the number of keyframes—you will find that it does cause major problems. It also causes problems when assigning vertices to joints for deformations. Some of these jumps are due to a problem called *parametric singularity*, commonly known as *gimbal lock*, in which a rotational degree of freedom is lost due to the alignment of the axes. Other jumps are caused simply by the tracking software as it converts transformation matrices to Euler angles (x, y, z). Any motion capture service bureau should be able to deliver data without these kinds of defects. Finally, make sure there is a keyframe in every frame. If the data has many gaps of data without keyframes, it is likely that it has been heavily massaged to achieve the final result.

Another important item to consider when capturing data that requires postprocessing is capacity. Whether or not the studio is able to deliver the data you need in a reasonable time frame has to do with the number of data analysts it employs and the number of other jobs that will be processed during the same period. If this becomes a problem, try to work out a schedule whereby you will receive partial deliveries. Also, arrange to receive your data in all its stages: from the global translation file to the hierarchical rotations file to the file compatible with your animation software. You never know when you will need these files, and it probably doesn't cost the studio any extra money to provide them to you.

Before closing a deal with the service bureau, make sure to reserve the dates needed, and put the agreement on paper. Also, find out about liability insurance. If the studio is not covered for any accidents that may occur at the stage, you will have to arrange coverage for your session.

In summary, these are some of the items you should require from a motion capture service bureau:

- Experience in your particular field
- Desired motion capture equipment
- A capture volume as large as your needs require
- Motion data without noise and other defects such as gimbal locks and continuity of keyframes
- Support for your animation format
- Ability to deliver data synchronized with video and audio in both tape and digital movie files
- Ability to apply data to characters, including interactions and other improvements
- Other value-added services
- Enough capacity to fit your deadline
- Insurance

Props and Rigs

If your session requires any special prop, now is the time to put it together. When a certain item has a digital counterpart, it can be critical that both match perfectly. In other cases, only a small representation of the actual prop is needed. You should follow the guidelines concerning units of measurement that I discussed previously.

It is always a good idea to capture data from props. In an optical stage, this should always be done no matter what. If an item is to remain stationary during the capture session, there is no need to collect its data at the same time as the actual performance; the more markers there are, the more postprocessing is needed for a shot. A single capture of only the stationary props can be layered with the performance after the fact. You can do this before the actual session in order to make sure everything will match afterward. Having this data will help the digital artists in charge of the scene layout to know exactly where everything is on the stage. All moving props, such as guns, swords, and footballs, should be captured during the performance, but prepared in advance.

Some rigs are simple, such as a rope and a harness, but others are very complex mechanical contraptions that can interfere with a motion capture session either by introducing unwanted interference or by occluding reflecting markers. Make sure this won't happen by testing the rig at the stage. Familiarize yourself with it before designing the marker setup, as you may have to place markers on the rig as opposed to the performer.

Talent

Performance animation is so named because the animation emanates from a performance, but the importance of the actual performance is often underestimated. If one has to produce a filmed project, you usually search for good acting talent. Likewise, if one puts together an animated cartoon, you'd want to have trained animators working on the characters. For some reason, when the two are combined, people just don't believe such experience is needed. This may be true if you are dealing with a video game that will play back at four frames per second, but for any other kind of performance animation project, talent is as important as any other artistic source, whether it is acting, animating, painting, singing, or sculpting.

This train of thought started because a few years ago, the final animation of a character created by performance animation was very different from the actual performance. Today this still happens in some cases, but we already have the tools to create exact replicas of the realistic motion, which makes the performer all the more important.

Take your time finding the right talent for your project. Do not simply decide to use your lead programmer just because you want to get some bang for your payroll buck, or your level designer because he had some martial arts training when he was a child. Performance is the source of the art in performance animation, and as such, it should not be the place to cut corners.

Match the right project with the right type of performer. If your project is a football video game, use football players; for a fighting game, use martial artists. If you have to collect the performance of a particular celebrity, try to use that celebrity. Stunt people are very good if you are doing rough motions, but you shouldn't use them as they do in film, only for dangerous shots. For each character, you must use the same performer throughout the project to avoid multiple character setups.

It is always best to modify the character model to approximate the proportions of the performer, but many times this is not possible, so you must try to find talent that matches as close as possible. Take all measurements from the people you are considering and compare their proportions to the model's. It will save time later in the process.

The best way to find talent is to prepare a test audition, where you will have a few applicants perform certain motions that will best represent the content of the project, perhaps involving the required props and rigging. Make a short list of the best ones and, if possible, schedule a second session in which you will capture their performance. The data by itself reveals more subtleties that are not obvious when looking at the live performance, like small idiosyncrasies particular to specific people that would not be acceptable for your digital character. Try looking at the data without knowing who the performer is, and you will probably make the best choice. If you are using a motion capture service bureau, this step may seem like overkill, but you could combine it with the test session and kill two birds with one stone. You may also think that it is a big waste of your time to put markers on each of the applicants, but the fact is that the performance will suffer if the talent doesn't feel comfortable dressed in black leotards or carrying a heavy tethered contraption, so a dress rehearsal is in order. When hiring a performer, make sure he or she will be available for follow-up sessions. Also keep in mind that one performer can play many parts if necessary.

Other issues that are important relate to insurance, union requirements, rights over the performance, and other contractual questions. I never had to deal with some of these, especially unions, because the medium was pretty new at the time, but today at least the Screen Actors Guild (SAG) has established guidelines for its members' participation in digital performances. The ownership of the captured data remains a gray area. The norm still in force today is that the owner of the data is whoever pays for the session; however, this assumption is bound to be questioned sooner rather than later, as performers acquire more practical understanding of the medium. In the near future, the use of a performance will probably be equated with the use of a person's likeness. Granted that a performance may not be as descriptive of a person as a photograph or a painting would, but when the performer's movements are widely known it could be, as is the case with world-renown celebrities such as Marcel Marceau or Michael Jackson.

A performer with motion capture experience is also very desirable. There are special issues that a performer must understand when at the motion capture stage. In any situation, the talent must look at the stage in terms of the animation space, knowing where the origin is and being able to move within the boundaries of the capture area. These restrictions are not common to other media and sometimes present difficulty to the inexperienced performer. Previous experience also helps in the capture of motions that are intended to be blended together. The performer must be good at starting and ending his or her performance in a predetermined pose.

If your talent happens to be a celebrity, spend some time explaining to him or her how things work at a motion capture shoot. The performer has to be willing to abide by the guidelines or no good data will be collected. Sometimes bringing in a director that the celebrity can trust is a lifesaving move. If the director is well educated in the motion capture process, everything should go smoothly, so invest some of your time to make sure that is the case. When working with a celebrity, you almost never get rehearsal time, but try to push for it.

Designing the Marker Positions

If you are using an optical system, chances are you can define the marker locations. You should take advantage of that, especially if you have a preliminary character setup design.

At TSi, we had a default human marker setup that basically covered all the main bone groups. Most of our clients' projects were covered by this configuration, especially the real-time video game projects. For those projects, we almost always captured more data than needed because the client usually did not have a preliminary character setup design. As far as clients are concerned, the service bureau is responsible for obtaining all the information that their particular project requires, so it is important to know as much as possible about the characters in advance. The marker configuration should be as efficient as possible for the capture session, but by the same token, it should enable you to collect all the data that your character setup will require. That is why I recommend a character setup design in advance.

A default marker setup is a great starting point because it will cover major joints, and you can add and remove from it as needed. Figure 3.6 shows a default marker setup for an optical system. As opposed to an external device that collects rotational values, such as an electromechanical suit, an optical system captures translational data for individual points. The advantage is that you can calculate internal joint rotations by grouping specific markers. The default optical marker setup should be able to capture all internal rotations of the major joints, including at least four sections of the spine (see Figure 3.7).

To collect all the needed information for each of these moving segments, you need to add enough markers to form a triangle. The markers need to be placed in locations where the skin is

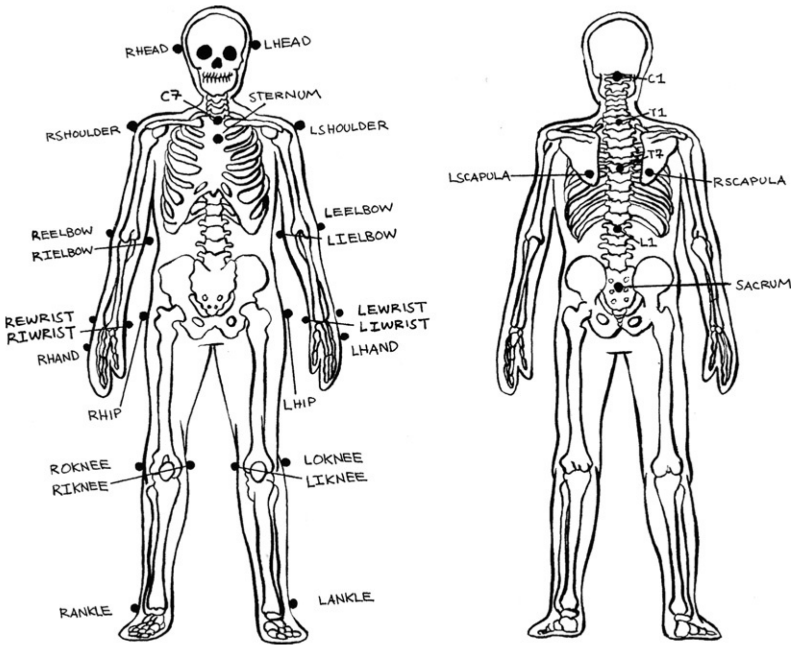

Figure 3.6 Default optical marker setup.

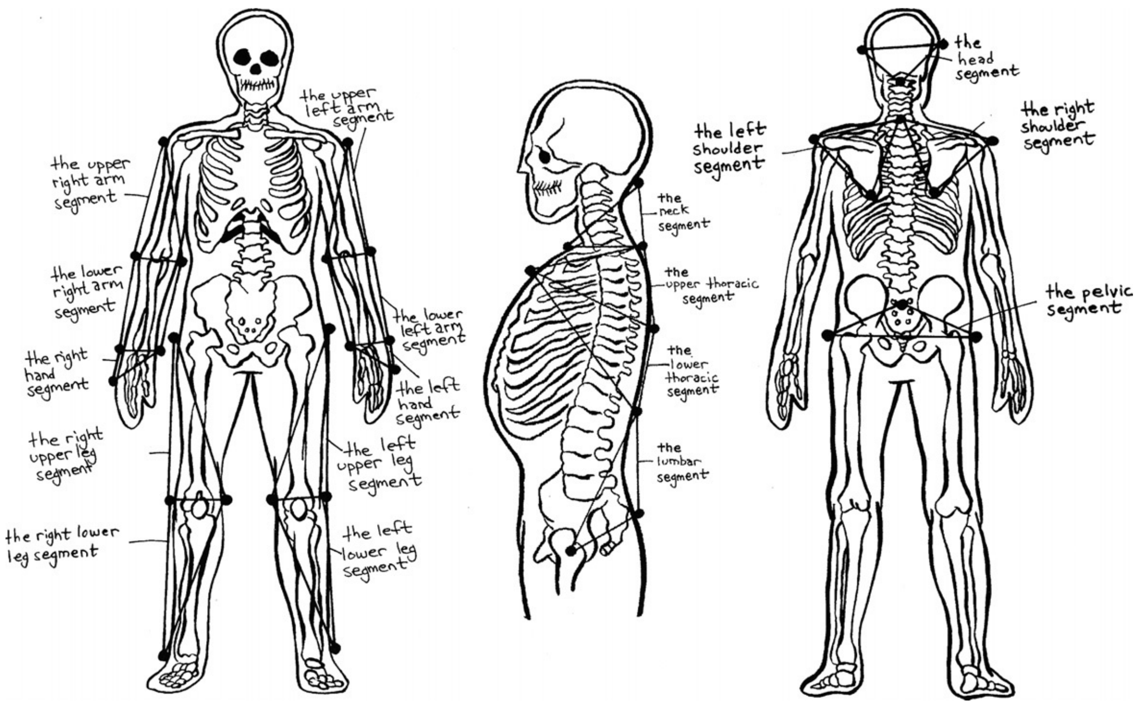

Figure 3.7 Major bone groups covered by the default marker configuration.

close to the bone, because the more tissue and fat between the marker and the bone, the more distortion of the initial triangle is induced. The size of the triangle can be unified for all frames as a postprocess, but the less modification is applied to the data, the better the result will look.

The default marker setup allows for the collection of all possible motions of the rigid segments composed of the pelvis, vertebral column (in four separate segments, including the neck), head, left and right shoulder, upper arm, lower arm, hand, upper and lower leg, and foot (in two segments). Although they share markers, I will cover each of these segments separately after describing the terminology of motion.

Terminology of Motion

There are up to six degrees of freedom, or possible movements, available in whole or in part to any given joint. In biomechanical terms, these movements are called *flexion*, *extension*, *abduction*, *adduction*, *medial or internal rotation*, and *lateral or external rotation*.

Flexion is the motion of bending between two adjacent segments that decreases the angle between them, whereas *extension* is the opposite or reverse bend, whereby the segments return to their initial position. Examples of flexion and extension are shown in Figure 3.8. In all the character setups that I describe, flexions and extensions are represented by the *X*-axis.

The next pair of movements, abduction and adduction, pertain to the *Z*-axis of character setups. *Abduction* is the movement away from the center of the segment, such as a leg rotating toward the outside of the body; *adduction* is the returning motion. Figure 3.9 illustrates typical abductions and adductions.

The last group of motions are the rotations along the axis that travels through the segment (see Figure 3.10). These are normally called *medial* (or *internal*) and *lateral* (or *external*), depending on whether the rotation is taking the segment and its children closer or farther away from the body, respectively. They are also known as *twists*. The body and head rotations are called *left* and *right rotations*. The *Y*-axis represents these rotations in character setups.

Pelvis

If you trace a straight line from the center of the skull, passing through the center of the atlas down to the ankle joint, and you traverse it with a line going through the center of the sacrum, the intersection point will be close to the human body's center of

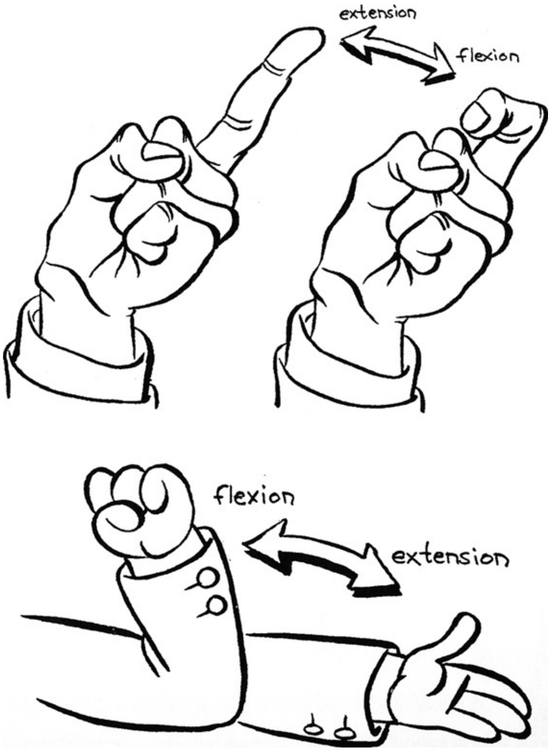

Figure 3.8 Examples of flexion and extension.

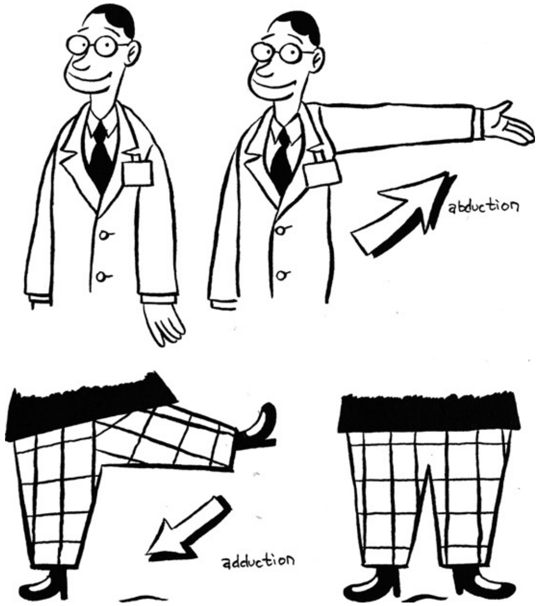

Figure 3.9 Examples of abduction and adduction.

gravity (Figure 3.11). This point is very important for any character setup, especially when using a hierarchical chain of nodes, because it will be the parent of the body-based node chain. As such, it will hold the global translational data for the body.

Figure 3.12 shows the triangle that depicts the motion of the pelvis, which is formed by the markers LHIP, RHIP, and SACRUM. Both LHIP and RHIP are placed along the sides at the same height as the head of the femur. The SACRUM marker is located in the back around the middle of the sacrum.

Vertebral Column

The human vertebral column is formed by 24 vertebrae that comprise its three divisions: lumbar spine, thoracic spine, and cervical spine. The lumbar spine is the lowest segment, consisting of vertebrae L1 through L5, which are similar in shape and motion characteristics (Figure 3.13). Left or right rotation is not easy for the lumbar spine vertebrae, but all other movements are common. The whole lumbar segment can be labeled as one rigid

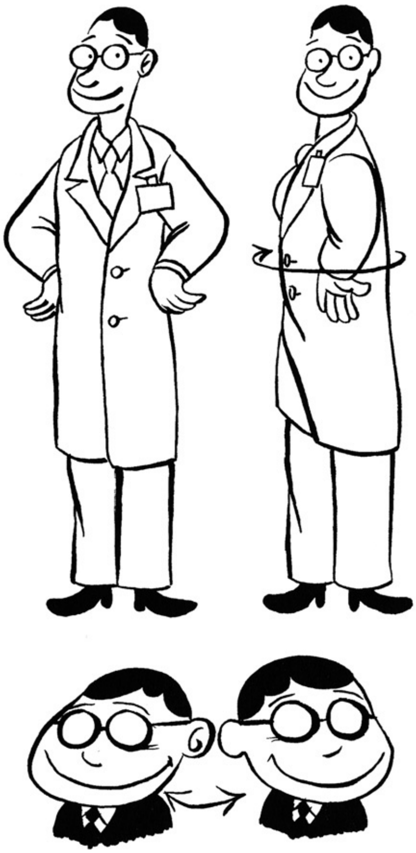

Figure 3.10 Examples of rotations.

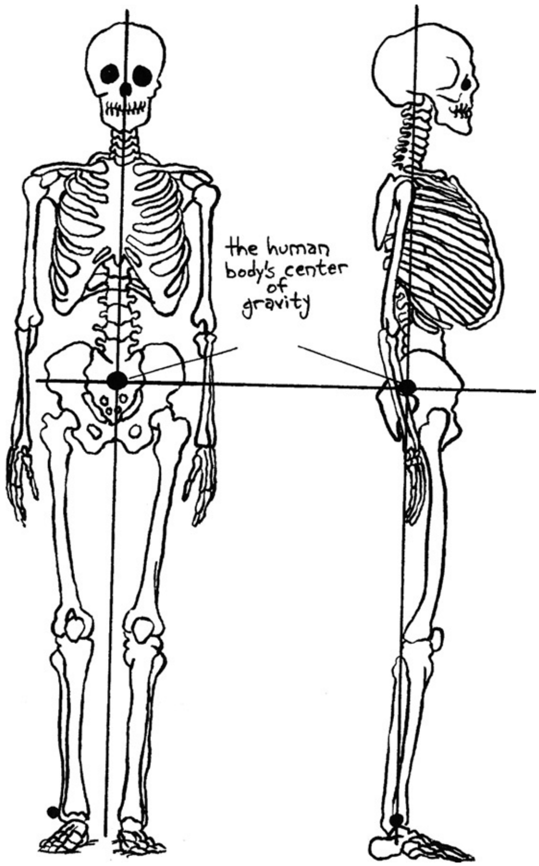

Figure 3.11 Human body's center of gravity.

motion segment defined by markers LHIP, SACRUM, and L1. The L1 marker, as its name indicates, should be placed in the back of the L1 vertebra. Other combinations of markers can be used, but the one I describe here will work fine.

The thoracic spine contains 12 vertebrae, T1 through T12. I divide this part of the column into two motion segments called the *lower thorax* and the *upper thorax*, based on their motion characteristics (see Figure 3.14). The vertebrae in the lower thorax (T8-T12) have much more mobility than the ones in the upper thorax (T1-T7), because the latter have the first seven ribs attached to them. These ribs are called *true ribs* and are connected directly to the sternum, which constrains their motion. The next three ribs are called *false ribs* and are attached to the front part of rib 7 instead of to the sternum. The last two ribs are called *floating ribs* and are not attached to the anterior or front side at all. Even though the thoracic spine could be divided into three motion groups based on its motion characteristics, I will only divide it into two segments. This division is based on the fact that only two vertebrae compose the smallest segment, so the lower thoracic segment will be formed by the vertebrae that hold the false and floating ribs.

The lower thoracic segment is calculated by the combination of markers L1, STERNUM, and T7; the upper thorax is determined by T7, STERNUM, and T1. The STERNUM marker should be placed at the bottom of the sternum, whereas T7

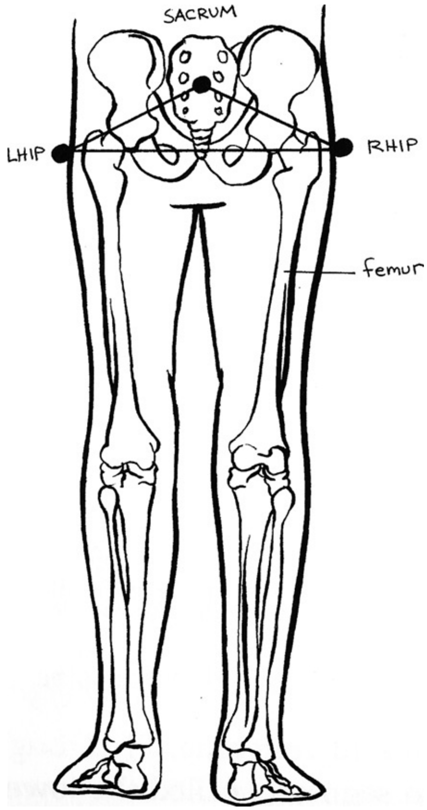

Figure 3.12 The triangle that depicts the motion of the pelvis.

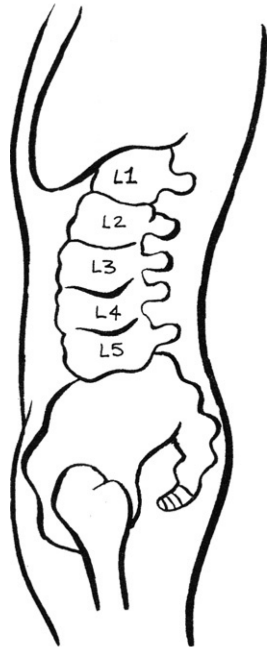

Figure 3.13 Lumbar spine.

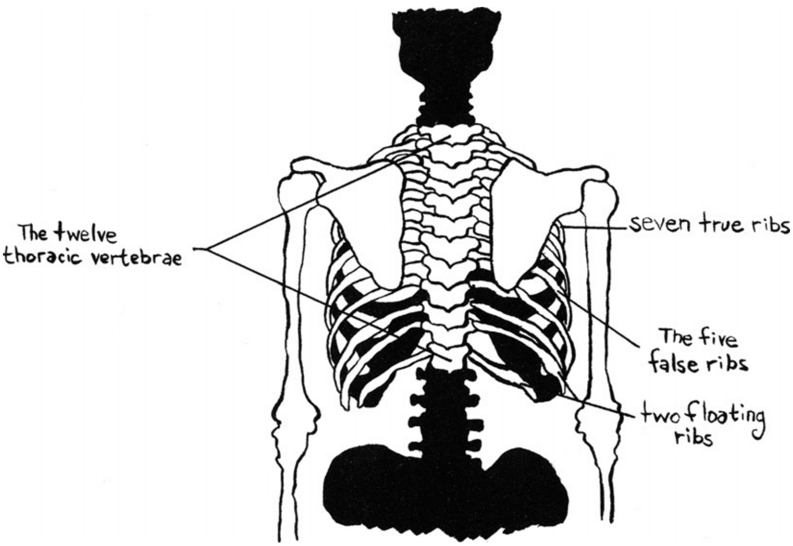

Figure 3.14 Thoracic spine.

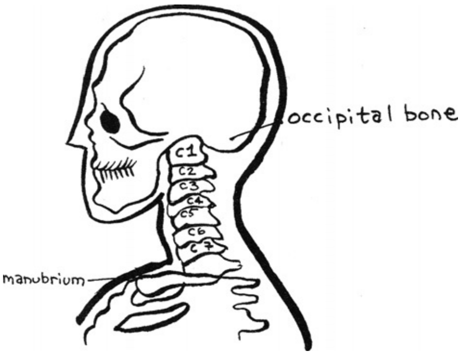

Figure 3.15 The neck.

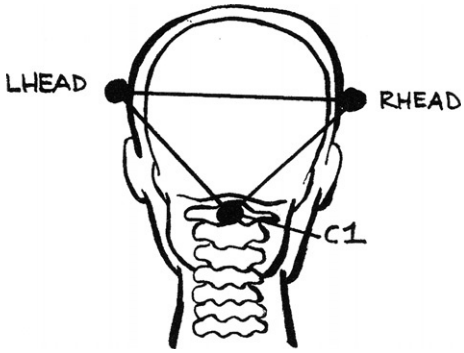

Figure 3.17 The head motion triangle.

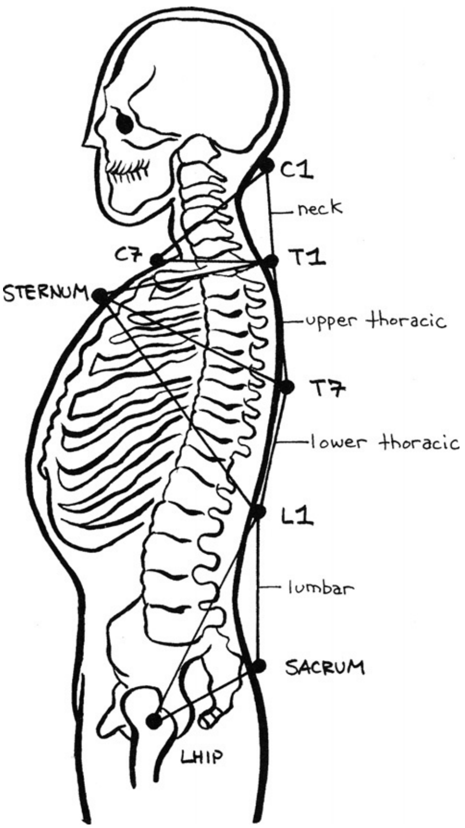

Figure 3.16 The vertebral column and its four motion segments.

and L1 should be located behind their respective vertebrae.

The final segment pertaining to the vertebral column is the neck (Figure 3.15). It is formed by vertebrae C1 through C7, which have increased mobility over the lumbar and thoracic vertebrae, and will be represented by markers T1, C1, and C7. Marker C7 should be placed on the anterior side, at the height of the manubrium. C1 should be located at the bottom of the occipital. Figure 3.16 shows the entire vertebral column and its four motion segments.

Head

The motion of the skull is easy to collect because it is rigid and has almost no tissue. I use markers C1, RHEAD, and LHEAD. The two latter markers can be placed on a cap and, if preferred, can be located in front and back instead of side by side. Figure 3.17 shows the triangle that represents head motion.

Shoulder

The shoulder is one of the most difficult segments to capture accurately. It is formed by a combination of three joints: the *glenohumeral joint* between the humerus and the scapula, the *acromioclavicular joint* between the distal clavicle and the acromion, and the *sternoclavicular joint* between the medial clavicle and the manubrium (see Figure 3.18).

The sternoclavicular and acromioclavicular joints are driven by the motion of the scapula, which adds range to the movements that the shoulder can do, along with the glenohumeral joint. The shoulder motions can be captured using the C7 marker along with the LSHOULDER and LSCAPULA or RSHOULDER and RSCAPULA markers, depending on the side. Each SHOULDER marker should be placed just above its respective acromion, trying to keep it unaffected by the glenohumeral joint's motion. Each SCAPULA marker should be positioned on the posterior side at the bottom of its respective scapula. Unfortunately, scapular motion is very difficult to collect because the scapula slides under the skin, so if this motion is really needed a procedural method based on biomechanical rules is required. Most motion capture software packages that are used in entertainment-related applications ignore the true mechanics of the shoulder complex. That works in many cases, but sometimes it results in nonrealistic shoulder motions. Figure 3.19 depicts the triangles that represent the shoulder segments.

Upper Arm

The humerus is the bone that supports the upper arm. It starts at the glenohumeral joint and has the capability of flexion, extension, abduction, adduction, and lateral and medial rotations. You need a triangle to define the motion of this important segment (see Figure 3.20). The first vertex is represented by

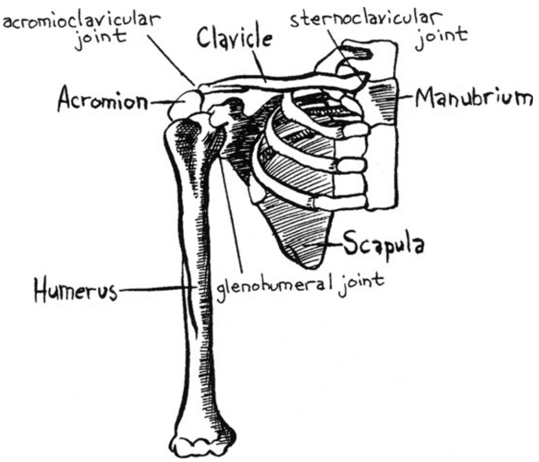

Figure 3.18 The shoulder complex.

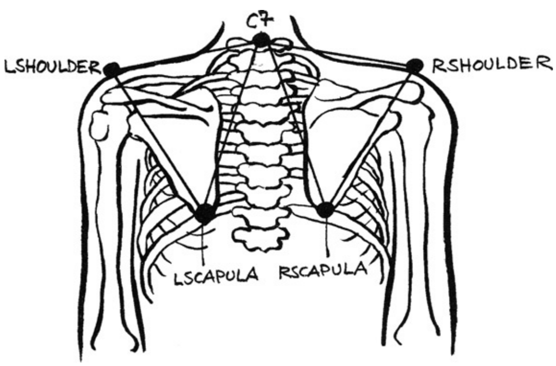

Figure 3.19 The triangles representing the shoulder segments.

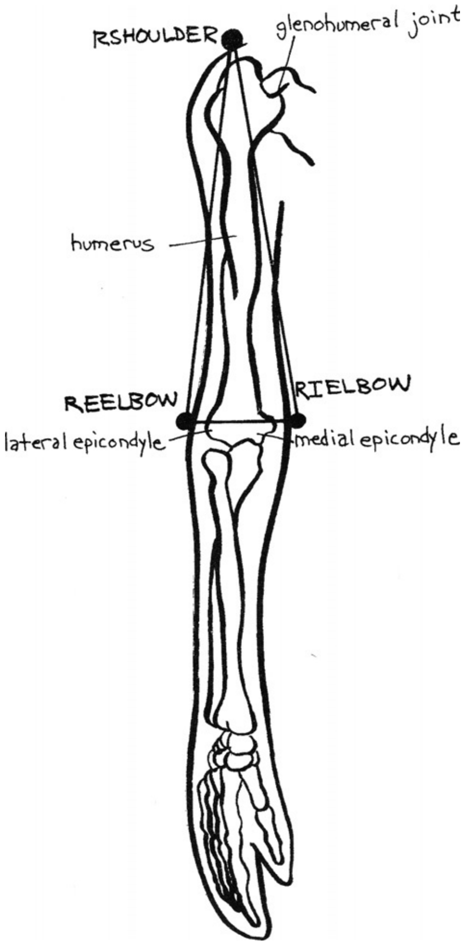

Figure 3.20 The upper arm segment.

the already known SHOULDER marker, while the other two are represented by IELBOW and EELBOW. These two markers should be placed at the elbow, over the medial and lateral epicondyle, respectively. The IELBOW marker in some cases may be occluded from the cameras. In such a case, it would be acceptable to place an additional marker somewhere else on the upper arm—perhaps along the lateral humerus, where muscle action is not too apparent, unlike over the biceps or triceps.

Lower Arm

The forearm is formed by the radius and ulna, and is linked to the humerus by the elbow joint, which is restricted to flexion and extension. It is easy to confuse the motion characteristics of this joint because as we see the rotation of the hand, we think we are rotating the elbow, or even the shoulder, but in reality we are crossing the radius over the ulna. This motion is called *pronation*, whereas the reverse motion in which the radius and ulna return to their parallel location is called *supination* (Figure 3.21).

Understanding the arm mechanics is key to the realistic setup of a humanoid character. Most marker setups that I've seen in the past do not take into account all of the arm's underlying structure, and they create data that assumes the whole arm to be formed by two segments that share lateral and medial rotations, while the wrist has its own. When the final character is set up for deformation, therefore, one usually sees horrible abnormalities along the shoulder and wrist areas that need to be dealt with.

The forearm's motion will not be represented by a triangle, because at least four points are needed to catalog its complexity. The lower arm quadrangle is defined by IELBOW, EELBOW, IWRIST, and EWRIST (see Figure 3.22). The IWRIST marker should be placed at the bottom of the ulna, and the EWRIST marker should be located at the bottom of the radius. It is also possible, if it suits the particular performance, to place these markers between the radius and ulna, at the anterior and posterior sides of the lower forearm. The idea is to place them as close to the hand as possible without affecting them by hand movements, so they should be located above the carpal bones.

Hand

The hand is usually captured as a whole segment when using a body capture system, but a bend-sensing glove, such as the Cyberglove, can be used simultaneously. If this is done, a parallel and synchronized stream of information will be collected and must be merged after the fact. For optical capture of the rigid hand segment, you can use both wrist markers (IWRIST and EWRIST) plus the HAND marker, located above the metacarpals. The use of both wrist markers will ensure that the elbow rotations stay where they belong.

Some state-of-the-art optical motion capture systems are able to capture finger motions when using small markers. Usually to achieve accurate and complete hand motions using such a system, a special camera setup focusing on hands is required. When capturing full body motions in a large stage, it may be possible to add a marker (FINGERS) on top of the last segment of the index finger that will allow to capture the finger rotations as a whole segment and that can be used to drive a procedural control to bend all the fingers. Figure 3.23 shows the hand motion segments.

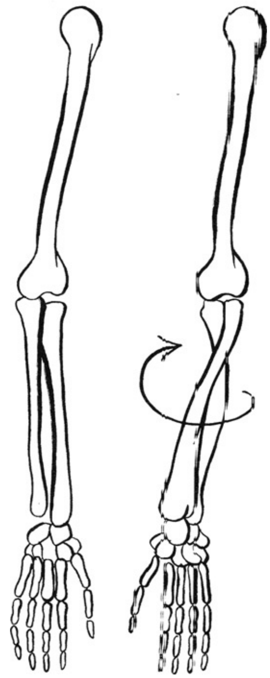

Figure 3.21 Pronation and supination.

Upper Leg

The hip joint is capable of flexion, extension, abduction, adduction, and both medial and lateral rotations, but it is not possible to capture its exact pivot point location because it is located internally. Its position can be adjusted in postprocessing. Use the HIP, EKNEE, and IKNEE markers to create the segment triangle, as shown in Figure 3.24. Both knee markers should be placed on the lower end of the femur, over the lateral and medial condyles. If the motion does not permit having two markers on each side of the knee joint, one of them can be moved to the midfemur area, but doing so will require more fine-tuning after the fact.

Lower Leg

The lower leg segment is based on the movements of the knee, which is capable of flexion and extension primarily, but can also rotate a few degrees in lateral and medial directions. Most of the rotational movements, though, are based on the hip rotations and abduction and adduction of the foot. To capture the motion segment of the lower leg, use EKNEE, IKNEE, and ANKLE, which should be placed over the lateral malleolus (see Figure 3.25). If one of the knee markers is not present, you can place another marker in the midfrontal area of the lower leg, where the tibia is closest to the skin.

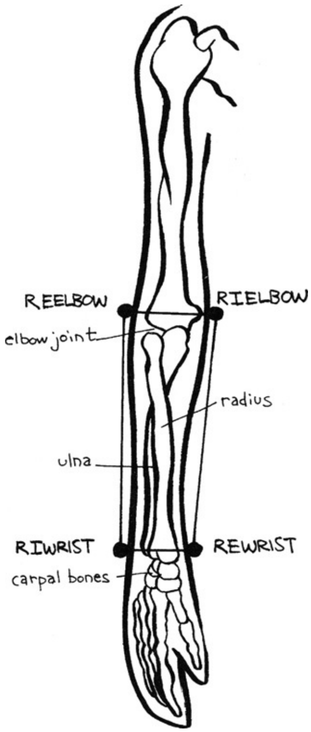

Figure 3.22 The lower arm motion segment.

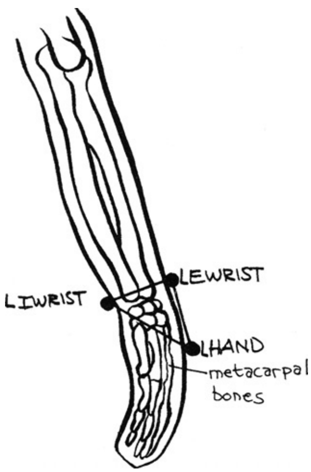

Figure 3.23 The hand motion segments.

Foot

The foot has a very large number of joints: 31, to be precise. Most marker setups consider the foot as one single motion segment, but I believe that a minimum of two segments is acceptable for most projects. It is best to wear hard shoes if possible when capturing the foot, since its entire complicated mechanism is filtered into two segments that can be captured by an optical system.

We cannot break the foot in exactly two segments based on its mechanical structure, but I will assume that the first segment starts at the ankle joint and ends somewhere along the metatarsals. The reason for this division line is that the rotation of the second segment is a combination of many joints, including the tarsometatarsal, metatarsophalangeal, and interphalangeal. The first segment is defined by ANKLE, HEEL, and FOOT. HEEL should be placed in the back of the shoe, as low as possible, because this marker will be used to define where the floor is for this segment. FOOT should be located also at floor level where the shoe bends when the foot is in flexion.

The second segment is assumed to be strictly capable of flexion and extension, so only a line is required, formed by FOOT and TOE. TOE should be placed in the front of the shoe, as close to the floor as possible, because it will define the location of the floor for this segment. Figure 3.26 shows the foot segments.

Using the Optimal Number of Markers per Application

What we discussed previously is a generic marker layout that will generate enough data for a high-resolution performance, suitable for a film or other high-end entertainment application. In life sciences projects, the marker layouts are designed differently, as the results there mostly need to be as exact as possible without regard to esthetics. Depending on your particular application, you may want to use a different number of markers per performer.

There are applications where the object of the motion capture is to line up digital elements with live action elements. For example, a character that is half digital and half live action would require much more accuracy than a whole digital performance, as any sliding in the data would be very noticeable. For applications such as these, it is recommended to use redundant markers that can aid in the correction of data. Also, using the best camera setup is highly recommended.

Capturing props on set sounds like an easy task. Just slap three markers on the prop and that's it. Not exactly. The biggest problem with props is that they are not well documented and can be forgotten easily. I've had props' data lost even in projects with as many as two dozen shots. A way to avoid this from happening is to keep extremely good notes for each trial. If the project is big enough, these notes should be kept in a database that the crew should have access to. The best solution, and one that hasn't been enacted yet as far as I know, would be to develop a system where everything is tagged with RFID chips, and by everything I mean everyone, too. As performers and props move into the stage, the RFID is read by a sensor at the entry point and whatever is being captured on a specific trial is automatically logged.

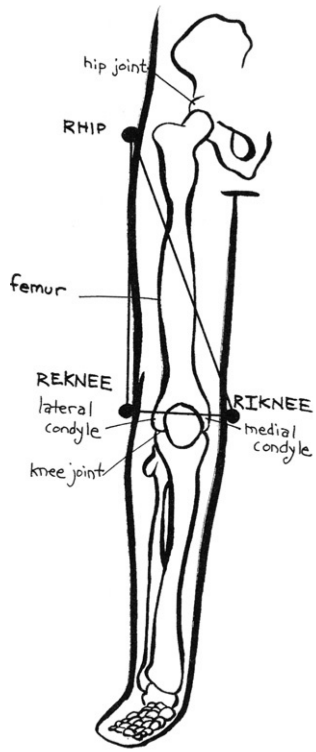

Figure 3.24 The upper leg motion segment.

Capture Schedule

You have prepared a candidate table, which you have sorted in different ways according to categories. You have hired the talent, built the props, looked at the rigs, and know how many different characters and configurations you have to deal with. Each shot has a detailed blueprint as well. It is time to put together the capture schedule.

You will now use some of those categories you created earlier, since you will build the schedule using the category constants and variables. A *constant* is something you cannot change in the time it has been scheduled, such as the availability of a performer. *Variables* are the categories you can schedule any time you want. The category types that are important for scheduling a session are the ones that have to do with volume, characters, interaction, props and rigs, and character setup. The latter category, character setup, is used indirectly, as each character setup will translate into one or more marker setups. The categories based on blending are used during the session and data application, and the ones based on delivery priority are used during postprocessing.

You can use different scheduling methods depending on the size of your project. In the early days at TSi, we generally used Microsoft Project. There are operations research methods such as Simplex or Network Flow Programming that can produce a mathematically optimal result, but unless the project is extremely complicated, such a solution would be overkill. Choosing a solution has more to do with the number of categories than with the number of shots, because each category creates either a new marker setup or a new stage configuration, and the idea is to minimize the number of times you reconfigure the stage and the marker setup. You could have a job for which you need to capture 500 shots, but if it involves no props, no interactions, no rigs,

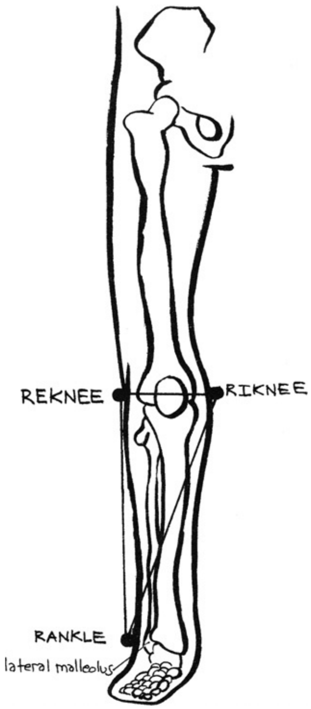

Figure 3.25 The lower leg motion segment.

and only one character, it becomes a very simple scheduling problem.

If you have access to a system with enough cameras and a large enough stage you may not need to reconfigure the volume and camera locations at all, but we will assume otherwise for the example.

The best way to illustrate the scheduling task is by using a low-tech model established by film production, where the shooting calendar is created by using a production board such as the one shown in Figure 3.27. I like this approach because it helps visualize the overall production timetable. The talent, character name, and marker setup are listed horizontally on the right side of the board, whereas the shots with their stage configuration are shown vertically. You can rearrange the shots because they are listed on moveable strips and color coded by stage configuration. Gaps between shot groups indicate days or half-days. This method is better illustrated by a simple example.

Suppose you have scheduled 2 days of motion capture at an optical studio. The whole project calls for three different camera configurations based on volume, plus there are two different characters, Stan and Ollie, performed by different actors. Stan has two shots in which he's wearing a harness and two in which he's interacting with a surfboard, so you will have different marker configurations for those groups of shots. Also, there are two shots in which two characters appear together, and you have decided to have them perform simultaneously.

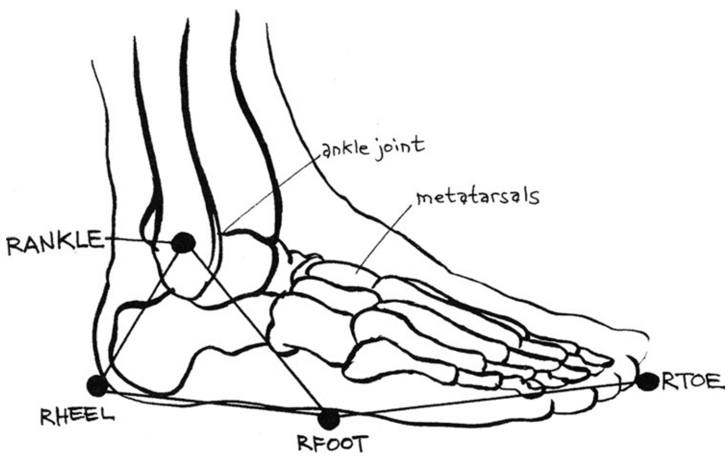

Figure 3.26 The foot motion segments.

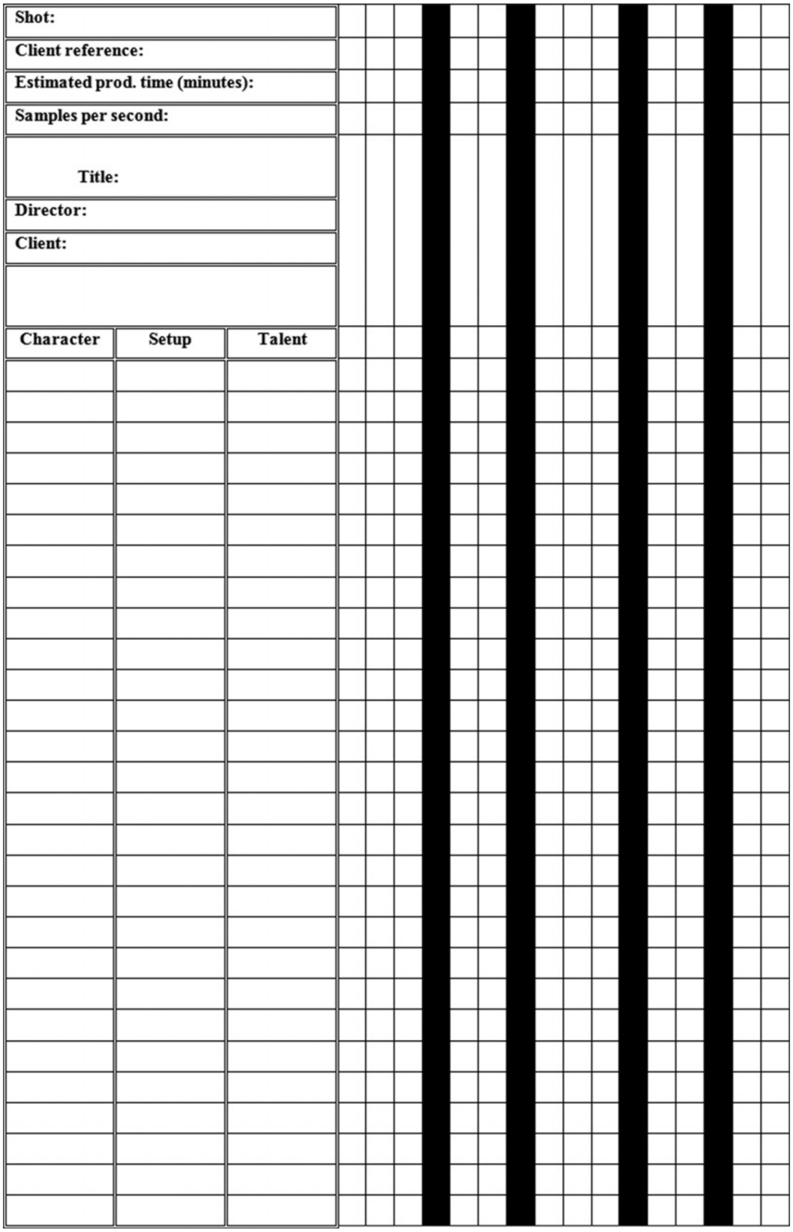

Figure 3.27 Production board.

You already know that your shots have the following marker setups:

1. Stan shots with harness: 2
2. Stan shots without harness: 8
3. Stan holding surfboard: 2

4. Ollie shots: 4
5. Multiple shots: 2
6. Total shots: 18

Notice that you have five different marker configurations. A marker configuration is defined by every time there is even a minor difference in the marker setup, such as a different character, a character and an interactor or prop, multiple characters, and so forth.

You will fill out a moveable strip with each shot's information. The strips are color coded by stage configuration, so you'll have three colors to work with. An example of a production board with the information already transferred is shown in Figure 3.28; you can see the two capture sessions divided by a black gap. The strips have been placed in a way such that stage configuration takes precedence over other variables, such as talent availability, but if we had restrictions or constants, we could easily reorganize the board to reflect them. All shots with the volume 1 configuration and most with volume 2 will be captured on the first day, finishing with all shots requiring a rig and harness. The second day, the remaining shots with volume 2 configuration will be shot first. These happen to be the two multiple shots. Then, all volume 3 shots are captured, starting with Stan's shots (including his surfboard ones) and finishing with the last Ollie shot.

The drawback to this schedule is that both actors have to be available for both sessions, but in this case, it is more cost effective to configure the stage only three times and have the talent come back. If there had been any talent restrictions, we would have had to rearrange the stage a few more times to accommodate.

Rehearsals

The importance of rehearsing can be easily underestimated, because producers sometimes think that the cost of having a performer show up in advance to get familiarized with the motions is higher than the cost of the extra time at the stage. Even if the stage is in-house, this may not be true, as it takes at least two extra people to operate whatever motion capture device is being used, but this is not the only reason.

Provide the talent with the script and all the blueprints in advance and have a preproduction meeting at which you discuss the character and all the motions. Listen to all the performer's concerns and comments about every shot and make sure he or she understands exactly what is expected from the technical and esthetic standpoints. The talent must know all there is to know about the character.

Shot:			0S02	0S03	0S07	0S08	0S09	0S18	0S01	0S05	0S06	0S10	0S12	0S13	0S15	0S04	0S11	0S14	0S16	0S17
Client reference:			S23	S32	S12	S09	S65	S37	S21	S98	S42	S45	S98	S04	S73	S11	S43	S19	S06	S91
Estimated prod. time (minutes):			30	30	30	30	30	30	30	30	60	60	30	45	45	30	30	45	45	30
Collection samples per second:			120	120	240	120	120	120	120	120	120	120	120	240	120	120	120	120	120	120
Title: *Stan & Ollie go surfing*			Volume 1	Volume 1	Volume 1	Volume 1	Volume 1	Volume 1	Volume 2	Volume 2	Volume 2	Volume 2	Volume 2	Volume 2	Volume 2	Volume 3	Volume 3	Volume 3	Volume 3	Volume 3
Director: *John Director*																				
Client: *Peter Client*																				
Character	Setup	Talent																		
Stan	*Normal*	*Paul Talent*	X	X	X	X	X				X					X	X			
Stan	*Harness*	*Paul Talent*										X	X							
Stan	*Surfboard*	*Paul Talent*																X	X	
Stan, Ollie	*Multiple*	*Paul & Tom*												X	X					
Ollie	*Normal*	*Tom Talent*						X	X	X										X

Figure 3.28 Production board with values filled in.

You want the motions to be rehearsed and learned by the performer in a comfortable environment, that is, without the constricting markers, sensors, or potentiometers that are common to all systems. This practice is consistent with other media, such as theater, where a rehearsal in costume is not done until all the lines and scenes are well learned by everybody. The idea is to get the talent very familiar with the part first; once that is achieved, get him or her familiar with the costume.

Another reason why it is a good idea to rehearse in advance is that the talent may not be capable of performing the motions required. Even though you had casting sessions and the performer you chose looked extremely qualified, it is always possible that some of the motions might be too complex for that particular performer. If you rehearse, you will have time to recast your session.

Finally, make the performer rehearse wearing the motion capture gear, because it will add an extra level of discomfort that you must be sure the performer can handle. Even if the data is to be captured with an optical system, the talent could feel confined by the markers or the black tights.

During the Session

If everything was planned as previously described, the actual session should flow very smoothly. You will start with an efficient list of shots, a very clear understanding of each one, well-rehearsed performers, and predefined marker setups. If the preparation was done correctly, there's no need to have everybody involved in the project supervising the session, because they have already given their input as the project evolved. The session is not the right time for changes in the setups, script, or shots.

The people handling the session should consist of a director; a mocap stage director; a technical supervisor; the mocap staff; the talent; riggers, choreographers, or related motion experts; and additional support personnel. I don't mean to exclude other people associated with the project: They can be present as long as they are not involved, although a big crowd of onlookers is never appropriate. A clear chain of command, headed by the director, is necessary. The mocap stage director should act as the first assistant director on a movie set, being the one calling all the instructions to the crew and giving recording cues to the mocap staff. I've often seen chaos erupt because of lack of planning and excess of opinions, which usually results in unusable data—not because it is unreadable, but because it is either the wrong performance or the wrong setup. It is best to leave changes to the follow-up session, the one that I suggested be scheduled for any corrections and problems that may arise.

Bring at least one video camera to the stage in order to shoot each take from a favorable angle. If the project is linear, try to use an angle similar to the final rendering. A monitor should be hooked up to the camera to allow the director to look at the framing and help in the blending and looping of shots.

A clapper like the kind used for motion picture shooting is essential; otherwise, you won't be able to distinguish the data from the video reference. Make sure you write in the clapper all the pertinent information, such as shot name, take number, file name, project name, character name, director name, and date, even if the video is synchronized with the captured data.

When dealing with stationary props that need to match with digital props, make sure to capture at least one file of only the props with enough markers to make out all of their dimensions. After this data is captured, there is no need to place markers on these props again, provided they remain in the same position throughout the session.

Sometimes it is better to collect a motion at a very high sampling rate, especially if the motion is very fast, because it helps if you need to do post-tracking (as you would with an optical system). The markers are closer frame after frame, which helps the software to find the continuity a lot better. Many motion capture operators believe that a higher capture sampling rate increases the postprocess work, but that is not the case if the cameras operate at the same resolution at the high and low sampling rates.

Try to use a capture frame rate that is divisible by the final frame rate for the project. For example, if you will ultimately need 24 frames per second, don't collect data at 30 or 60; instead, use 48, 96, or any other multiple of 24 if possible. If your equipment is only capable of multiples of 30, then use a higher sampling rate so there will be more data available for a cleaner down-sampling.

The Base Position

When you model a character, you choose a neutral pose that is appropriate for the kinds of deformations the model will go through. When the character is humanoid, a good neutral pose is standing up with arms about 45° apart from the body, palms

Figure 3.29 Character in a neutral pose.

facing in, and legs apart about 30°, as shown in Figure 3.29. Only if the character's performance will be in a predetermined position throughout the project would you want to model it in a non-neutral pose. If your character will be sitting in all shots, for example, you might as well model a sitting character, since you can come up with good deformations while modeling.

The base position in motion capture serves a similar purpose. When setting up the character, you need a neutral pose with which to begin—a pose that will match your modeled character's neutral position. It is also a definition of the character's initial set of coordinate systems, which I describe later. The neutral pose in mocap terms is also called the *t-pose*.

If the t-pose is different from the typical pose described earlier, the modeling of the character should ideally happen after the base position has been captured, but this rarely happens. Also, it is preferable to model a character using the performer's proportions, but this is sometimes impossible as well. If you have modeled the character in advance, it is necessary to capture a matching base position by measuring the angle of rotation of limbs and helping the performer adopt that stance.

Since all you need is one frame of matching data, the capture of the t-pose can include motion. As you set up the digital character, you can pick the closest frame to the model. Pose the performer's legs at the right angle of separation, and make sure the back is straight. Have the performer rotate his or her arms from a position close to the body to a total extension pose, making sure the palms are pointing in the right direction and that the arms are aligned with the body. That will most likely yield a frame that matches the arm rotations. Also, make sure that the performer is facing the right axis, whether it is negative Z or anything else, so that the collected data is facing you when looking at the front orthogonal view of your animation software. Finally, make sure the performer is standing at the origin.

When capturing a base position, place an extra marker on the floor. It could be at the front, back, or any side, as long as you remember where it is and you always do it the same way. This marker will tell you where the front and back of the data are.

Depending on the file format you will use for your final product, the base position may or may not be a pose of your character's skeleton in which all the joints have a transformation value of zero. I cover this in the next chapter when I talk about file formats.

Directing the Motion Capture Session

Directing a motion capture session is very similar to directing live action. A film director runs the acted performance and the first assistant director controls the technical aspects of the shoot, such as the camera roll and sound. The director in a motion capture set controls the performance and the mocap stage director controls the video shooting and the collection of the data. In many cases, especially small mocap projects, the two roles are filled by the same person. Directing motion capture includes more technical elements that apply exclusively to this type of performance, such as interactions, blending, loops, and nonlinear action in general, plus all the constraints of the virtual space in which the action takes place. A director must have a very clear vision of the performer's motion as filtered into the final character and must enforce the character's limitations on the performer.

Capturing motion for crowds in the background of a scene is very different from capturing the lead character, so if the shots to be captured are very important to the project, it is advisable that the director of the project be the one directing the session. In feature film, a second-unit director is responsible for running certain less important shots that the director cannot or will not direct, but he or she needs to know exactly what the director is looking for. As the director of the motion capture session, you need to be totally involved and aware of what the project is about.

The first thing you must do as director is to brief the crew, talent, and visitors about what is about to happen and what is not acceptable for them to do while the session is going on. You don't want anybody leaning on a camera stand if you are using an optical system, since this will disturb the calibration and make all the data unusable. Also, people must know not to bring bright objects to an optical stage, or certain metallic objects to an electromagnetic capture session. Make sure people stay within certain boundaries that will not distract you or the talent from the actual performance.

Time should be spent making the talent comfortable and trying to convey what the shot calls for. A well-trained performer is not necessarily used to acting in black tights with white dots or outfitted with sensors and a mechanical suit, and self-consciousness will hurt the performance. You can avoid this by establishing a rapport with the talent, inducing trust while maintaining authority, especially if the talent is a celebrity.

The pace of the session should be as slow as required. Rushing through the moves often doesn't result in good takes. You should look at the taped reference after every good shot to make sure it is a potential final shot; take notes that will allow you to decide this during or after the session. For safety, always capture more than one potential hero shot, as the data could have technical problems.

The monitor attached to the camera can be used to make sure that the blending and looping start and end poses are met. An experienced film director is already familiar with the concept of continuity, which in the case of motion capture can be taken to the extreme. As in film, you should draw or tape marks on the floor and the monitor's screen that will allow the matching of moves. It is possible to use more than one camera/monitor combination if the blending is very complicated. The higher the frame rate of the final medium, the more exact the blending must be at capture time.

When directing a session as captured by optical equipment, you must use the motion capture blueprint to make sure all the interactions are performed with minimum occlusion of markers, and you must make sure no markers are missing during a shot. If the performer needs to interact with nonmarked people on stage, those persons must be careful not to occlude or disturb any markers; however, ultimate responsibility always lies with the director. The technical supervisor should act as an advisor to the director in these and other technical matters.

On one occasion, I was asked to provide motion capture services for the opening of a well-known television sitcom. The performer was to be a famous actor, the star of the show. I asked the client for storyboards, but they declined to provide them, saying that they had to be kept confidential. They did describe very roughly what the project would entail and said that it would consist of five or six shots, most of them very simple, except for one in which the actor would have to sit on a chair. They would bring their own director and I wouldn't meet him or the actor until the day of the shoot. You may be asking why I walked into this situation when I should have known better. Well, it wasn't a difficult decision, because we were dealing with a famous star and a high-profile show. Also, it was a paying job and we were looking for those. The risk seemed calculated. On many occasions I had to walk into projects as blindly as in this case, but that is the risk you take when dealing with clients.

A few things went wrong during that session. First, the motions turned out to be much more complicated than expected. The actor had to sit down in a chair, so we had prepared a special chair with a clear back in order to avoid occluding the markers placed on his back. During the shoot, the client decided that a chair would not do and asked for a couch. To accommodate this new development, we had to add more markers to the actor's front, resulting in a different configuration from the rest of the already collected data. When motion data is delivered, a base position file is included to be used for character setup, and the rest of the data is supposed to be compatible with that file. This change would force the client to have two different character setups, so I decided instead to recompute all the couch data to match the first base position. This was a difficult process and at that time we didn't have an easy way of testing the results, so it took extra work to make it match.

Second, the director was actually a character animator rather than a director, and he had very little knowledge of motion capture issues and wasn't open to many of our suggestions. The performer was a famous actor and had a big attitude about taking direction from an unknown person. We tried to persuade him to do more takes of certain shots, especially the one with the couch, but he declined. We customarily use those extra takes as insurance in case the preferred take is not clean. As expected, the couch shot wasn't very clean because we lost many of the markers for a long period of time. Even though we had placed markers in the front, only two cameras could see them when he was sitting, which wasn't enough with the archaic optical system that was the state of the art at the time. We had to work long hours reconstructing some of the motion.

To make a long story short, we managed to finish the job on time, but ended up losing money because of the extra time it took to clean up and recalculate the data. Changing the special chair for a couch is an inconsequential change in a film or a television production, but for a motion capture session it meant the difference between profit and loss.

Summary

When deciding whether to use motion capture, you should answer the following questions:

1. Do you want realistic-looking motion?
2. Is the character human shaped?
3. Is the context right?
4. If the character is an animal, can a real animal perform the part?
5. If enhancements need to be made to the performance, are they reasonably small?
6. Will the character require squashing or stretching?

7. Will the character require anticipation beyond physical boundaries?
8. Will the character require follow-through action beyond physical boundaries?
9. Will the character require exaggeration beyond physical boundaries?
10. Is it likely that the director will require substantial changes after the capture session?
11. Will the shot fit within the available capture volume?
12. Will it be possible to capture enough data to feed to an existing character setup?
13. Are the interactions affecting the root of the motion?
14. Is the number of interactions low enough to make it feasible to use motion capture?
15. Will the necessary props interfere with the capture process?
16. Will the necessary rigs interfere with the capture process?
17. Is it possible to blend the shots together, if necessary, after the capture?
18. Is it possible to capture at the necessary sample rate?
19. Is it possible to capture the necessary number of performers simultaneously?
20. Is it possible to comply with special instructions?

When deciding whether to use motion capture, do not make the following assumptions:

- Motion capture will save you money.
- Motion capture will save you time.
- A human can perform a nonhuman character.
- You can do all the shots for a project with motion capture.
- You can do all interactions with motion capture.
- You can fix a bad performance after the fact.
- You can modify timing after the fact.
- A director will accept the captured performance even if he or she directed it.
- You can produce performance animation without advance planning.

Remember these facts about motion capture:

1. Cartoon characters almost never look good if performed.
2. If your data is not good, don't fix it: Recapture it.
3. Always videotape all captured shots.
4. If a move can't be done physically, it can't be captured.

5

SETTING UP YOUR CHARACTER

CHAPTER OUTLINE

Setting up a digital character involves two main steps: mechanical setup and deformation setup. I'm not talking about modeling or lighting, but about motion setup or rigging, which is the action of making a character controllable or poseable. *Mechanical setup* involves creating the skeleton that will drive the character and other skeletons that will be used to add motions of different types, as well as creating all the controls that will be used to animate the character. *Deformation setup* defines the relationship of each vertex of the character to the skeleton.

Understanding Motion Capture for Computer Animation.

This chapter deals only with mechanical setup, because deformation setup is very specific to each particular software package and is implemented in the same way for both keyframed and captured motion data. Nevertheless, when using captured motion data, you must keep in mind that deformations are pushed to the limit. Thus, it is very important to take extra care when creating the mechanical setup that will drive them. Usually, the rotational data that a motion capture studio delivers is not enough to drive a typical vertex-weighting deformation scheme. Keyframe animators can distribute transformations across several joints when posing a character; for example, in twisting an arm, rotations can be partially added to the shoulder, elbow, and wrist in such a way that the character's envelope will not be twisted. Captured motion data does not follow this distribution pattern, but rather behaves in an anatomically correct way, assigning most of the twisting to the segment between the elbow and the wrist. This is usually not friendly to most deformation schemes. Setting up the mechanics of a character to follow the properties of the actual raw data is the key to better deformations.

The rotational motion data formats that were discussed in Chapter 4 include a skeleton for the user. In those cases, the mechanical setup has already been done for you. All you need to do is import the file and, voilà, a skeleton is readily available for your character. I question most of these file formats because the creation of a skeleton is not something to take lightly. In some cases, the default skeleton in the motion file will do, but the highest quality jobs will require a very careful mechanical setup that can only be designed by the same person who is responsible for the setup of the deformations.

To maximize your control over captured motion data, it is best to obtain the data in the simplest format possible. All you need is the tracked marker data, so the translational file will do. This file contains only a stream of marker position coordinates over time. By using this file, you can design the skeleton inside your animation environment and not abide by what a motion capture studio delivers. This method is definitely more time consuming, so make sure you also obtain the rotational data, and determine first if it is of adequate quality and if the skeleton is appropriate enough for your project.

Setting up a Character with Rotational Data

When rotational data is used for motion capture, a skeleton is already included or assumed by the data stream. It is possible that all you need to do is import the data into your software and your character will start moving. It is also possible that the data may be incompatible with your software, or that you may have to change your software's settings in order to receive the data.

Depending on the software used, you may have to create a skeleton before importing the data. If you are dealing with data in the Acclaim format, chances are you only need to load the .asf file. When using other formats, you sometimes have to create the skeletal nodes and then load the data to pose them. If you must do that, you need to find out what the initial orientation and position of the bones should be.

If you load a skeleton in its base position and its pose doesn't match that of the character, the ideal solution is to pose the character to match using the modeling software, but in most cases this is impossible. Another solution is to modify the base position to match the pose of the character. Doing so entails recalculating all the motion data to compensate. A third and most frequently used solution is to add an expression that will add an offset to the data equal to the negative of the offset added to the base position. This is almost equivalent to recalculating all the data, but it is done on the fly. This technique has the advantage of being quick and easy, but it is a hack after all, so it isn't the cleanest solution and may even complicate Euler angle problems that had previously been fixed in the data.

When you import rotational data into animation software, you must make sure that the data is prepared to transform your character in the same order as the software does. In some programs, you are able to change this order—sometimes for each segment or for the whole project. Others are not so flexible. Some programs allow you to change the order of rotations only, whereas others allow you to change when translations, rotations, and scaling happen. You need to know what your particular software supports before preparing the rotational data.

Most programs and motion data formats assume translations will happen before rotations, and scaling will happen after rotations. Most problems that arise usually concern the order of rotations specifically. Three-dimensional rotations are non-commutative, which means you cannot interchange their order and obtain the same result.

A program that is used in a lot of pipelines is Autodesk MotionBuilder. This software primarily works with raw data but it will deliver rotational data in a skeleton for use in 3D animation packages such as Maya or video game engines.

Setting up a Character with Translational Data

The files you use to set up a character with translational data contain a set of markers with positional coordinates over time. You use combinations of these markers to create approximations of rigid segments that will define the skeleton of your character.

Creating the Internal Skeleton

The first step in setting up the character is finding the locations of all the local coordinate systems that will represent the new internal skeleton. There is at least one coordinate system associated with each of the rigid segments, and you need to find equations or formulas that will allow you to find these over time. Once you have these formulas, you can use them in your animation or video game software to position and orient joints.

The bones can be calculated using exclusively the markers that define the rigid segments associated with them, or you can use previously found internal points in combination with markers. The latter approach makes the process easier and ensures that the segments are linked to each other, considerable inaccuracy being common because markers are placed on the skin and do not exactly maintain their position relative to the bones.

Center of Gravity

The location of the character's center of gravity must be determined. Assuming the data was captured using the default marker setup defined in Chapter 3, the pelvis segment is defined by the SACRUM, LHIP, and RHIP markers. Figure 5.1 is a visualization of the center of gravity, shown in the figure as point PELVIS$(x_{\text{PELVIS}}, y_{\text{PELVIS}}, z_{\text{PELVIS}})$.

We begin by defining the rigid segment formed by SACRUM, LHIP, and RHIP, using the following linear equation of a plane:

$$ax + by + cz + d = 0 \tag{5.1}$$

As we know the coordinates of three points in space, LHIP$(x_{\text{LHIP}}, y_{\text{LHIP}}, z_{\text{LHIP}})$, RHIP$(x_{\text{RHIP}}, y_{\text{RHIP}}, z_{\text{RHIP}})$, and SACRUM$(x_{\text{SACRUM}}, y_{\text{SACRUM}}, z_{\text{SACRUM}})$, the constants $(a, b, c,$ and $d)$ of Equation (5.1) can be calculated by the following determinants:

$$a = \begin{vmatrix} 1 & y_{\text{LHIP}} & z_{\text{LHIP}} \\ 1 & y_{\text{RHIP}} & z_{\text{RHIP}} \\ 1 & y_{\text{SACRUM}} & z_{\text{SACRUM}} \end{vmatrix} \tag{5.2}$$

$$b = \begin{vmatrix} x_{\text{LHIP}} & 1 & z_{\text{LHIP}} \\ x_{\text{RHIP}} & 1 & z_{\text{RHIP}} \\ x_{\text{SACRUM}} & 1 & z_{\text{SACRUM}} \end{vmatrix} \tag{5.3}$$

$$c = \begin{vmatrix} x_{\text{LHIP}} & y_{\text{LHIP}} & 1 \\ x_{\text{RHIP}} & y_{\text{RHIP}} & 1 \\ x_{\text{SACRUM}} & y_{\text{SACRUM}} & 1 \end{vmatrix} \tag{5.4}$$

$$d = \begin{vmatrix} x_{\text{LHIP}} & y_{\text{LHIP}} & z_{\text{LHIP}} \\ x_{\text{RHIP}} & y_{\text{RHIP}} & z_{\text{RHIP}} \\ x_{\text{SACRUM}} & y_{\text{SACRUM}} & z_{\text{SACRUM}} \end{vmatrix} \tag{5.5}$$

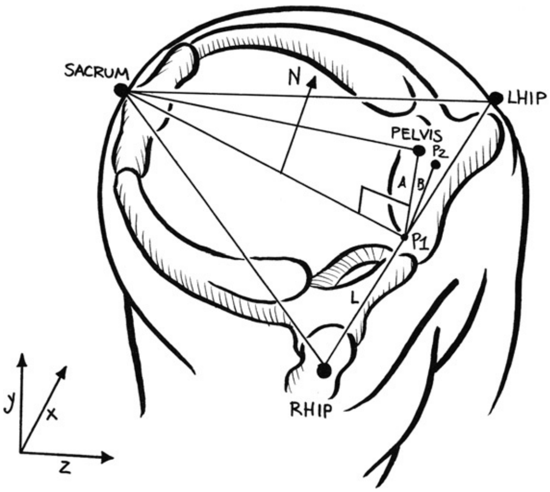

Figure 5.1 The center of gravity.

We can expand these equations, obtaining the following:

$$\begin{aligned} a = {} & y_{\text{LHIP}}(z_{\text{RHIP}} - z_{\text{SACRUM}}) + y_{\text{RHIP}}(z_{\text{SACRUM}} - z_{\text{LHIP}}) \\ & + y_{\text{SACRUM}}(z_{\text{LHIP}} - z_{\text{RHIP}}) \end{aligned} \tag{5.6}$$

$$\begin{aligned} b = {} & z_{\text{LHIP}}(x_{\text{RHIP}} - x_{\text{SACRUM}}) + z_{\text{RHIP}}(x_{\text{SACRUM}} - x_{\text{LHIP}}) \\ & + z_{\text{SACRUM}}(x_{\text{LHIP}} - x_{\text{RHIP}}) \end{aligned} \tag{5.7}$$

$$\begin{aligned} c = {} & x_{\text{LHIP}}(y_{\text{RHIP}} - y_{\text{SACRUM}}) + x_{\text{RHIP}}(y_{\text{SACRUM}} - y_{\text{LHIP}}) \\ & + x_{\text{SACRUM}}(y_{\text{LHIP}} - y_{\text{RHIP}}) \end{aligned} \tag{5.8}$$

$$\begin{aligned} d = {} & -x_{\text{LHIP}}(y_{\text{RHIP}}z_{\text{SACRUM}} - y_{\text{SACRUM}}z_{\text{RHIP}}) \\ & -x_{\text{RHIP}}(y_{\text{SACRUM}}z_{\text{LHIP}} - y_{\text{LHIP}}z_{\text{SACRUM}}) \\ & -x_{\text{SACRUM}}(y_{\text{LHIP}}z_{\text{RHIP}} - y_{\text{RHIP}}z_{\text{LHIP}}) \end{aligned} \tag{5.9}$$

We then proceed to find the direction cosines $(\sigma_x, \sigma_y, \sigma_z)$ of the normal N of the plane as follows:

$$\sigma_x = \frac{a}{\sqrt{a^2 + b^2 + c^2}} \tag{5.10}$$

$$\sigma_y = \frac{b}{\sqrt{a^2 + b^2 + c^2}} \tag{5.11}$$

$$\sigma_z = \frac{c}{\sqrt{a^2 + b^2 + c^2}} \tag{5.12}$$

We continue by defining a line segment L between points LHIP and RHIP. We will refer to the midpoint between LHIP and RHIP as P_1 and will calculate its coordinates by using the following linear parametric equations:

$$\begin{aligned} x &= a_x u + b_x \\ y &= a_y u + b_y \\ z &= a_z u + b_z \end{aligned} \tag{5.13}$$

u is the parametric variable that defines where in the line the point is located. Substituting $u = 0$, we obtain the coordinates of one of the end points; with $u = 1$, we obtain the coordinates of the second end point. $u = 0.5$ will give us the coordinates of P_1, the midpoint in the center of the line segment.

As we know the coordinates of the end points LHIP and RHIP, to solve the equations we start by substituting the end point coordinates. For the first end point, we assume $u = 0$ and we obtain

$$\begin{aligned} b_x &= x_{\text{LHIP}} \\ b_y &= y_{\text{LHIP}} \\ b_z &= z_{\text{LHIP}} \end{aligned} \tag{5.14}$$

where x_{LHIP}, y_{LHIP}, and z_{LHIP} are the coordinates of marker LHIP. For the second end point, we substitute $u = 1$ and obtain

$$\begin{aligned} x_{\text{RHIP}} &= a_x + b_x \\ y_{\text{RHIP}} &= a_y + b_y \\ z_{\text{RHIP}} &= a_z + b_z \end{aligned} \tag{5.15}$$

Substituting Equation (5.14) into Equation (5.15) gives us

$$\begin{aligned} a_x &= x_{\text{RHIP}} - x_{\text{LHIP}} \\ a_y &= y_{\text{RHIP}} - y_{\text{LHIP}} \\ a_z &= z_{\text{RHIP}} - z_{\text{LHIP}} \end{aligned} \tag{5.16}$$

Finally, we substitute Equations (5.14) and (5.16) into Equation (5.13) and obtain the coordinates of $P_1(x_1, y_1, z_1)$, where $u = 0.5$:

$$\begin{aligned} x_1 &= 0.5(x_{\text{RHIP}} - x_{\text{LHIP}}) + x_{\text{LHIP}} \\ y_1 &= 0.5(y_{\text{RHIP}} - y_{\text{LHIP}}) + y_{\text{LHIP}} \\ z_1 &= 0.5(z_{\text{RHIP}} - z_{\text{LHIP}}) + z_{\text{LHIP}} \end{aligned} \tag{5.17}$$

Next, we need to calculate the constant length of line segment A; to do so, we will need to assume that the system is in a base pose where L is aligned with the world x-axis and A is aligned with the world y-axis. In this state, we can further assume the following:

$$\begin{aligned} x_{\text{PELVIS}} &= x_1 \\ y_{\text{PELVIS}} &= y_{\text{SACRUM}} \\ z_{\text{PELVIS}} &= z_1 \end{aligned} \tag{5.18}$$

We make these assumptions because when the system is in its base state, PELVIS is located at the same height as SACRUM and at the same width and depth as P_1. The length of A is then calculated by

$$A = y_{\mathrm{SACRUM}} - y_1 \tag{5.19}$$

We have a line segment B from P_1 to P_2 that is parallel to N and of equal length to A. We know that any two parallel lines in space have the same direction cosines, so the direction cosines of B are the same as the direction cosines of N. The direction cosines for B are computed using the following equations:

$$\sigma_x = \frac{x_2 - x_1}{B} \tag{5.20}$$

$$\sigma_y = \frac{y_2 - y_1}{B} \tag{5.21}$$

$$\sigma_z = \frac{z_2 - z_1}{B} \tag{5.22}$$

As the only unknown variables are x_2, y_2, and z_2, we can obtain the frame-by-frame location of P_2 as follows:

$$x_2 = x_1 + B\sigma_x \tag{5.23}$$

$$y_2 = y_1 + B\sigma_y \tag{5.24}$$

$$z_2 = z_1 + B\sigma_z \tag{5.25}$$

Next, we need to find the value of θ, the angle between the plane normal and the z-axis. We know that

$$\sigma_z = \cos\theta \tag{5.26}$$

so for a base pose, we can calculate θ by

$$\theta = \arccos(\sigma_z) \tag{5.27}$$

The angle α between A and B is then found by

$$\alpha = -(90 - \theta) \tag{5.28}$$

To find the location of PELVIS, we need to rotate P_2 α degrees about the local x-axis at P_1. We perform this operation by first translating P_1 to the origin, then performing the rotation in x, and finally reversing the translation. We achieve this by the following set of matrix transformations:

$$[x_{\mathrm{PELVIS}}\ \ y_{\mathrm{PELVIS}}\ \ z_{\mathrm{PELVIS}}\ \ 1] = [x_2\ \ y_2\ \ z_2\ \ 1]\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ -x_1 & -y_1 & -z_1 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\sigma & \sin\alpha & 0\\ 0 & -\sin\alpha & \cos\alpha & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ x_1 & y_1 & z_1 & 1\end{bmatrix} \tag{5.29}$$

Vertebral Column

The vertebral column's four segments are used to calculate the position of the four main spinal joints in our setup. The lowest point is LUMBAR, followed by THORAX, LONECK, and UPNECK.

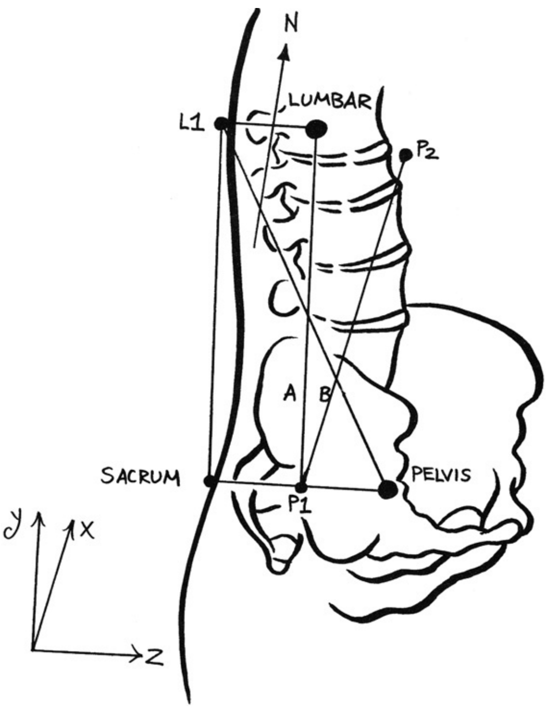

Figure 5.2 Elements used to calculate the position of LUMBAR.

Lumbar

Point LUMBAR($x_{\mathrm{LUMBAR}}, y_{\mathrm{LUMBAR}}, z_{\mathrm{LUMBAR}}$) is located at the same height as marker L1 and at the same width as point PELVIS. Its depth falls somewhere between L1 and PELVIS, so we will assume its location to be right in between them. We will use a very similar method to calculate LUMBAR as we used to calculate PELVIS, using the elements in Figure 5.2.

The location of the midpoint $P_1(x_1, y_1, z_1)$ is found by using Equation (5.17) as follows:

$$x_1 = 0.5(x_{\mathrm{SACRUM}} - x_{\mathrm{PELVIS}}) + x_{\mathrm{PELVIS}}$$
$$y_1 = 0.5(y_{\mathrm{SACRUM}} - y_{\mathrm{PELVIS}}) + y_{\mathrm{PELVIS}}$$
$$z_1 = 0.5(z_{\mathrm{SACRUM}} - z_{\mathrm{PELVIS}}) + z_{\mathrm{PELVIS}} \tag{5.30}$$

We then use Equation (5.19) to find the length of line segment A. This is done by assuming the system is at its base state:

$$A = y_{\mathrm{L1}} - y_1 \tag{5.31}$$

We know the coordinates of three points in space: L1($x_{\mathrm{L1}}, y_{\mathrm{L1}}, z_{\mathrm{L1}}$), PELVIS($x_{\mathrm{PELVIS}}, y_{\mathrm{PELVIS}}, z_{\mathrm{PELVIS}}$), and SACRUM($x_{\mathrm{SACRUM}}, y_{\mathrm{SACRUM}}, z_{\mathrm{SACRUM}}$). The constants ($a$, b, c, and d) of Equation (5.1) can be calculated by the determinants in Equations (5.2) through (5.5), which we can expand as follows:

$$a = y_{\mathrm{L1}}(z_{\mathrm{PELVIS}} - z_{\mathrm{SACRUM}}) + y_{\mathrm{PELVIS}}(z_{\mathrm{SACRUM}} - z_{\mathrm{L1}}) + y_{\mathrm{SACRUM}}(z_{\mathrm{L1}} - z_{\mathrm{PELVIS}}) \tag{5.32}$$

$$b = z_{\mathrm{L1}}(x_{\mathrm{PELVIS}} - x_{\mathrm{SACRUM}}) + z_{\mathrm{PELVIS}}(x_{\mathrm{SACRUM}} - x_{\mathrm{L1}}) + z_{\mathrm{SACRUM}}(x_{\mathrm{L1}} - x_{\mathrm{PELVIS}}) \tag{5.33}$$

$$c = x_{\mathrm{L1}}(y_{\mathrm{PELVIS}} - y_{\mathrm{SACRUM}}) + x_{\mathrm{PELVIS}}(y_{\mathrm{SACRUM}} - y_{\mathrm{L1}}) + x_{\mathrm{SACRUM}}(y_{\mathrm{L1}} - y_{\mathrm{PELVIS}}) \tag{5.34}$$

$$\begin{aligned} d = &-x_{\mathrm{L1}}(y_{\mathrm{PELVIS}}z_{\mathrm{SACRUM}} - y_{\mathrm{SACRUM}}z_{\mathrm{PELVIS}}) \\ &-x_{\mathrm{PELVIS}}(y_{\mathrm{SACRUM}}z_{\mathrm{L1}} - y_{\mathrm{L1}}z_{\mathrm{SACRUM}}) \\ &-x_{\mathrm{SACRUM}}(y_{\mathrm{L1}}z_{\mathrm{PELVIS}} - y_{\mathrm{PELVIS}}z_{\mathrm{L1}}) \end{aligned} \tag{5.35}$$

We proceed to find the plane's normal N direction cosines (σ_x, σ_y, σ_z), using Equations (5.10) through (5.12).

We have a line segment B from P_1 to P_2 that is parallel to N and of equal length to A. We know that any two parallel lines in space have the same direction cosines, so the direction cosines of B are the same as the direction cosines of N. We use Equations (5.20) through (5.22) to obtain the location of P_2:

$$x_2 = x_1 + A\sigma_x \tag{5.36}$$

$$y_2 = y_1 + A\sigma_y \tag{5.37}$$

$$z_2 = z_1 + A\sigma_z \tag{5.38}$$

We know the angle α between A and B is 90°, so to obtain LUMBAR's position, we need to rotate B 90° about the local z-axis at P_1. We use the following set of matrix transformations:

$$\begin{aligned} &[x_{\mathrm{LUMBAR}} \;\; y_{\mathrm{LUMBAR}} \;\; z_{\mathrm{LUMBAR}} \;\; 1] = \\ &[x_2 \;\; y_2 \;\; z_2 \;\; 1] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x_1 & -y_1 & -z_1 & 1 \end{bmatrix} \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x_1 & y_1 & z_1 & 1 \end{bmatrix} \end{aligned} \tag{5.39}$$

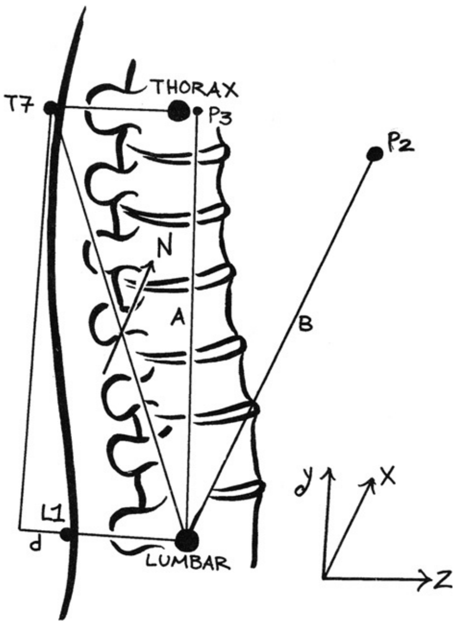

Figure 5.3 Elements used to calculate the position of THORAX.

THORAX

Point THORAX is calculated by the same method as LUMBAR, using T7, LUMBAR, and L1 instead of L1, PELVIS, and SACRUM. We don't use P_1 at all; instead, we use LUMBAR. The final transformation will yield point P_3, which can then be transformed by the depth difference d, which can be calculated from the base position. Figure 5.3 shows the elements used to calculate THORAX.

LONECK

Point LONECK is calculated by the same method as THORAX, using T1, THORAX, and T7 instead of T7, LUMBAR, and L1. Figure 5.4 shows the elements used to calculate LONECK.

UPNECK

For point UPNECK, we use the same method as for THORAX, using C1, LONECK, and T1. Figure 5.5 shows the elements used to calculate point UPNECK.

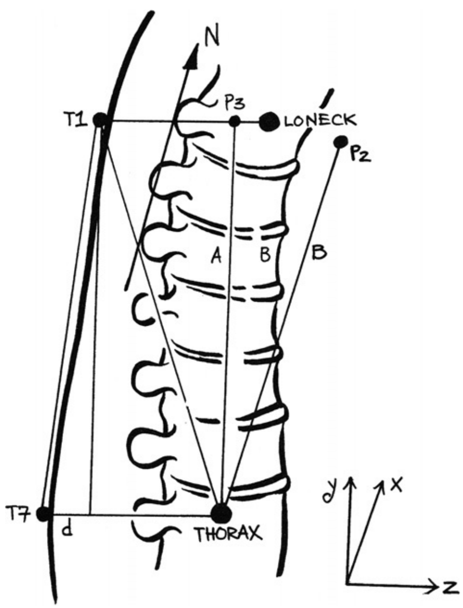

Figure 5.4 Elements used to calculate the position of LONECK.

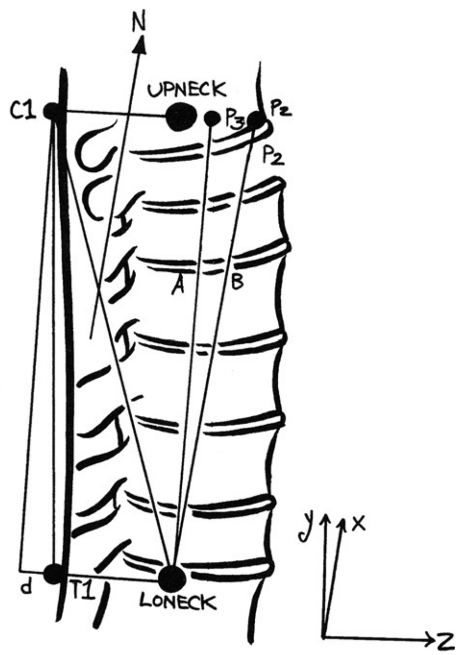

Figure 5.5 Elements used to calculate the position of UPNECK.

HEAD

Point HEAD is used to help orient the joint that has its root at UPNECK. We assume that it is located between LHEAD and RHEAD. We calculate the location of this point by simply applying Equations (5.13)–(5.17). Figure 5.6 shows the calculation of point HEAD.

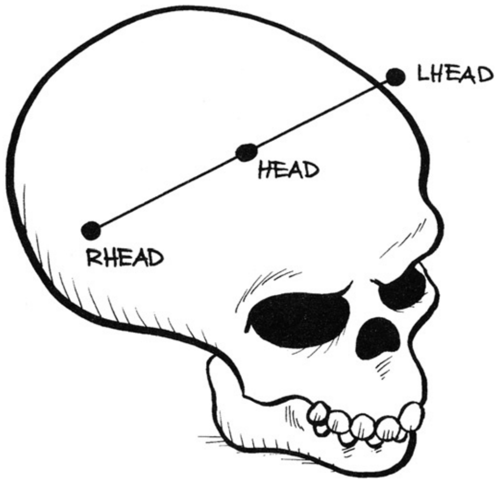

Figure 5.6 The calculation of point HEAD.

Torso

It is a good idea to separate the front part of the ribcage from the spine, as it is not as flexible, especially in the sternum area. We do this by using markers C7 and STERNUM, combined with point LONECK, as shown in Figure 5.7.

Point UPTORSO is simply calculated by obtaining a point in between C7 and LONECK, using Equations (5.13)–(5.17), with an appropriate value for parametric variable u that is just enough to internalize the point. A value of 0.05–0.1 will do in most cases, but it really depends on the width of the performer and the diameter of the marker.

Point LOTORSO is used to orient the joint with its root located at UPTORSO. It is calculated in a similar way, using a line between STERNUM and T7 and Equations (5.13)–(5.17). The parametric variable u should also be small enough to just internalize the point.

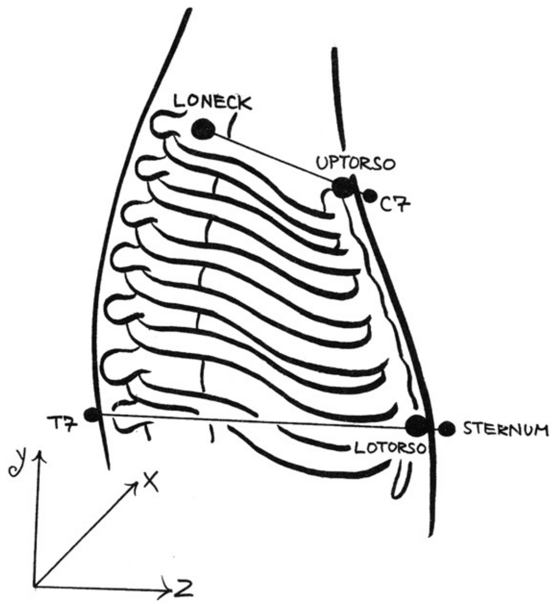

Figure 5.7 Calculating the position of the torso points UPTORSO and LOTORSO.

Shoulder and Arm

Most of the motion capture data files that I have seen do not account for the three joints that form the shoulder complex: the glenohumeral joint between the humerus and the scapula, the acromioclavicular joint between the distal clavicle and the acromion, and the sternoclavicular joint between the medial clavicle and the manubrium. None of the common deformation schemes works well with shoulders, but it helps to have a somewhat anatomically correct setup. If we use an internal deformation technique, such as a muscle-based system, the correctness is imperative.

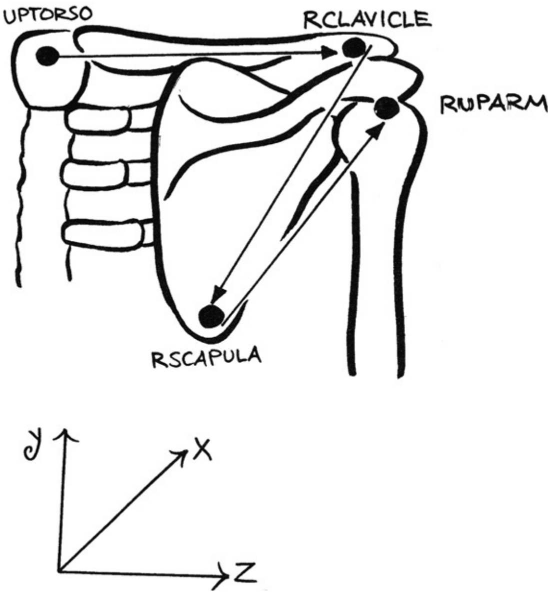

Figure 5.8 The points that define the shoulder joints.

The shoulder complex will be formed by joints that go from point UPTORSO to CLAVICLE, down to SCAPULA, and back up to UPARM (see Figure 5.8).

We first calculate point CLAVICLE, using markers SHOULDER and SCAPULA, and point LONECK. Using Equations (5.13)–(5.17), calculate a point P_1 between the SHOULDER marker and point LONECK. The parametric variable u should be defined so as to place P_1 almost above the acromioclavicular joint. We then calculate point CLAVICLE, using the same method for a line between P_1 and SCAPULA. Although it appears that the constructs of the shoulder complex are not well suited for animation, by using SCAPULA and SHOULDER to calculate the position of CLAVICLE, we are establishing links between the parts. This setup may not be optimal for keyframe animation, but it is well suited for captured motion data.

Marker SCAPULA can be used as is, because it only serves as an orientation point for CLAVICLE, but UPARM must be placed at the pivot point of the glenohumeral or shoulder joint. We do this by calculating a midpoint LOARM between markers IELBOW and EELBOW and using a line between SHOULDER and LOARM to find the location of UPARM (see Figure 5.9).

To replicate as faithfully as possible the rotation of the lower arm, where the radius rotates over the ulna, we will require two bones that we can originate at point LOARM. One will be aligned to point IWRIST and the other to EWRIST. Finally, the hand rotation is obtained by aligning a midpoint P_1, between IWRIST and EWRIST, to the HAND marker (see Figure 5.10).

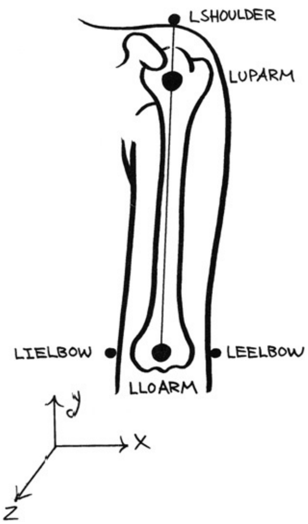

Figure 5.9 The upper arm.

Leg and Foot

The upper leg pivot point is located at the head of the femur. We trace a line between HIP and PELVIS, and place FEMUR at about 35% of the distance between the two. We then use the same line to create UPLEG, just inside the greater trochanter. Point LOLEG, located at the knee, is positioned by calculating the midpoint between markers IKNEE and EKNEE, as shown in Figure 5.11.

Point UPFOOT is obtained via a method similar to that used to calculate LUMBAR, using point LOLEG and markers HEEL and FOOT. We use marker ANKLE later to maintain orientation. LOFOOT is calculated using FOOT, TOE, and UPFOOT.

Orientation of the Bones

You now have a set of points in space where a set of local Cartesian coordinate systems will be located. These represent the roots of joints that will form the mechanical setup. The base position is assumed to be the laboratory frame in time where all these coordinate systems are in their initial state. You can further assume that the longitudinal axis of each segment is the line between the root and the target points, but the rotation about the longitudinal axis has to be locked down to a moving rigid segment, such as the ones defined in the marker setup design section in Chapter 3.

Some high-end animation programs, such as Maya, provide the ability to assign a point in space as a root and another as a target or an effector, using other points as orientation or aim constraints. If this is the case, you can now use these points, in combination with the marker sets that were defined as rigid segments in Chapter 3, to place your joints and constrain their primary orientation. If your software does not allow you to do this, you need to perform these calculations yourself, using the direction cosines described previously to calculate the angular changes in the plane and apply them to the bone. In this case, you might want to consider using a nonhierarchical system, as it would require fewer computations.

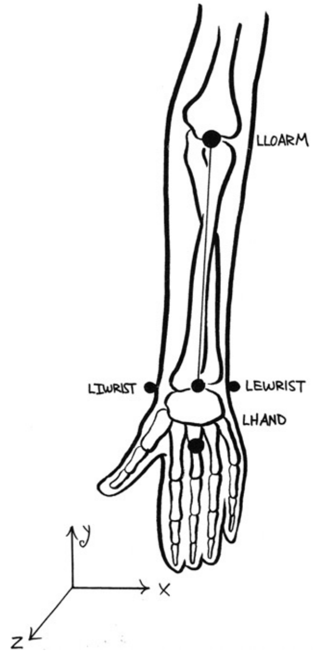

Figure 5.10 The lower arm.

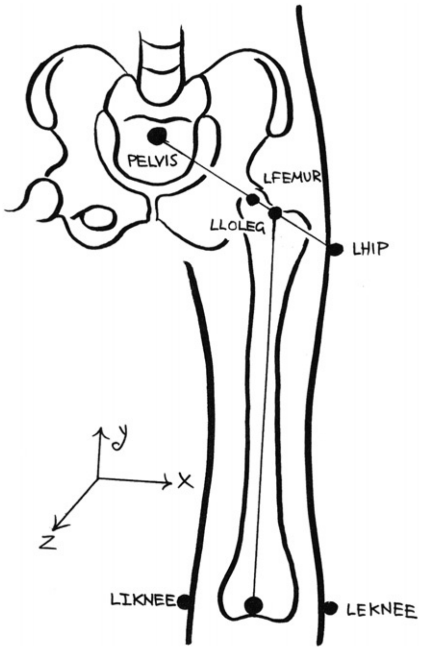

Figure 5.11 The upper leg.

Facial Motion Capture

Preparing the Data

Applying motion capture data to a digital model is not very user friendly. Facial data is even more of an undertaking. The main issue is that there aren't many software packages that have the necessary tools as they exist for body data.

To obtain decent facial data, one must apply enough markers to the face. A good layout will have markers placed along the lines of the main facial muscles. Figure 5.12 shows a facial marker setup that follows this concept.

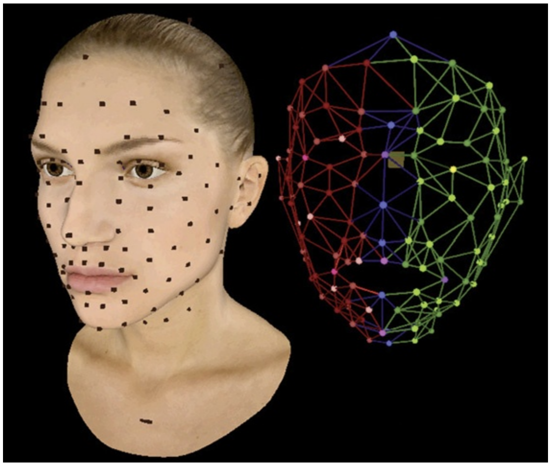

Figure 5.12 Facial marker layout. Photo courtesy of Digital Concepts.

Facial data is translational and it doesn't matter if the face is captured with the performer sitting still or moving; the data still needs to be stabilized before applying it to a digital character in order to remove any head motion from it. That process can be done by capturing the head motions and globally subtracting them from each of the markers that cover the face. It can be done in animation software or in packages such as Matlab.

An interesting software package that is used to manipulate motion data is Vicon Blade. Digital Concepts Group, a company founded by DJ Hauck and Steven Ilous, has created a toolset for Blade that greatly aids in the preparation of facial data, specifically in the labeling of markers, stabilization, cleanup, and retargeting. Figure 5.13 shows a screenshot of the DCG toolset.

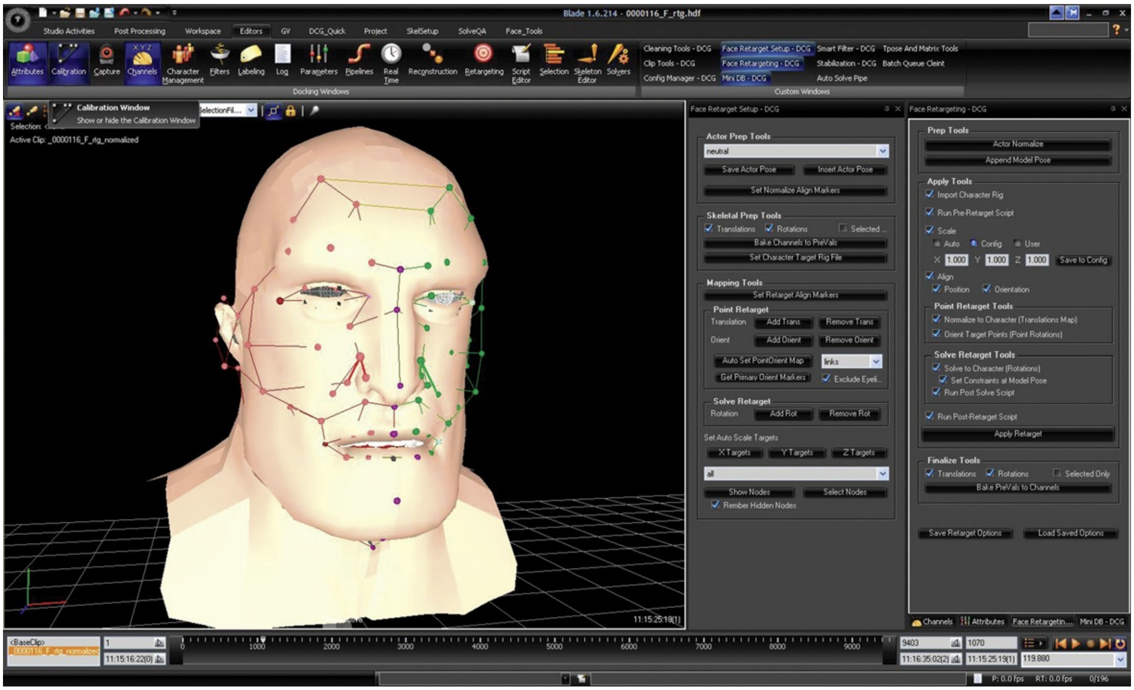

Figure 5.13 The DCG toolset. Photo courtesy of Digital Concepts Group.

Facial Character Setup

The two most widely used computer facial animation systems are the ones based on premodeled shape interpolation and the ones based on cluster or lattice deformations. Most of the facial systems today are a combination of both.

The state of the art in facial animation software is a muscle system combined with shape interpolation. This kind of system is not widely used because of the lack of available commercial technology as most of the currently existing systems are proprietary.

Once the data has been prepared as described in the previous section, it should be imported into the animation software where the character setup will be created.

Assuming a model with a body rig already exists, that data should be placed globally under the last node that pertains to the head motions. The data also should be offset globally to match as best as possible with the digital model's face. If the data was retargeted properly, it should be an easy match.

The data can then be used to drive a series of joints that will ultimately be used to deform the face. I particularly like to avoid the joints and drive a control mesh by constraining all its vertices directly to the data. I call this "The Mug" (Figure 5.14).

The mug can be used as one of many layers of controls that could be used to animate a face. It would deform the face using a deformer that allows a piece of geometry to ride on another. An example of a deformer such as this would be the Maya muscle.

By designing the rig in layers, an experienced setup artist can add different controls and deformers in modular ways, allowing the face to be animated by many possible methods such as blendshapes, manipulating muscles, and retargeted facial motion data.

Shapes can also be driven by motion data. This is very useful, especially if the character doesn't have facial features that resemble a human. For every blend shape, there would be a corresponding human expression. I call this group of human expressions the "training set." The final blend of shapes would be calculated frame by frame by an algorithm that would study the expression in the human data and calculate the blend in the training set that would achieve best the frame's expression. That blend could be applied to the nonhuman shapes, obtaining the corresponding final expression.

The facial systems that were used to create *Beowulf* and *Avatar* follow this principle. The shapes they use are based on the Facial Action Coding System list of expressions created by Paul Eckman. Obviously, the more shapes are used, the better will be the resulting expression.

Figure 5.14 The Mug.

The algorithm used to find the shape combination would analyze the options and calculate the final recipe. Mathematically, this would be considered an Inverse Problem, which is loosely defined as a problem where the answer is known but not the question. We know what the final expression looks like but we don't know what combination was used to create it. This kind of inverse problem would be solved by using a geometric fitting algorithm such as least squares or nonlinear regression. These methods are readily available in software packages such as Matlab.

Tips and Tricks

Switching the Order of Rotations

There are cases in which your data will be delivered based on a predefined order of rotations with which your software may not be compatible. If this is the case, you will have to recompute your data to reflect an acceptable order. Most of the common captured motion data files contain rotations specified in Euler angles. We will combine all the rotations into one matrix and then decompose it in a way that will yield the desired order of rotations.

The 3×3 rotation matrix of θ about the x-axis is given by

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \tag{5.40}$$

The 3×3 rotation matrix of ϕ about the y-axis is given by

$$\mathbf{R}_y = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix} \tag{5.41}$$

The 3×3 rotation matrix of ψ about the z-axis is given by

$$\mathbf{R}_z = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.42}$$

We begin by creating the matrix that will contain all the rotations. If the data file contains rotations in the *XYZ* order, we would multiply the matrices in the proper order as follows:

$$\mathbf{R}_{\theta\phi\psi} = \mathbf{R}_x\mathbf{R}_y\mathbf{R}_z \tag{5.43}$$

where $\theta =$ rotation about the x-axis, $\phi =$ rotation about the y-axis, and $\psi =$ rotation about the z-axis. We obtain the following matrix:

$$\mathbf{R}_{\theta\phi\psi} = \begin{bmatrix} \cos\phi\cos\psi & \cos\phi\sin\psi & -\sin\phi \\ \sin\theta\sin\phi\cos\psi - \cos\theta\sin\psi & \sin\theta\sin\phi\sin\psi + \cos\theta\cos\psi & \sin\theta\cos\phi \\ \cos\theta\sin\phi\cos\psi + \sin\theta\sin\psi & \cos\theta\sin\phi\sin\psi - \sin\theta\cos\psi & \cos\theta\cos\phi \end{bmatrix} \tag{5.44}$$

Because we know all the values of θ, ϕ, and ψ, we can solve the equations and represent the matrix as follows:

$$\mathbf{R}_{\theta\phi\psi} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{5.45}$$

We next build the matrix that we will use to extract the new order of rotations. If we had to convert the order of rotations to *ZYX*, for example, we would multiply the matrices in Equations (5.40)–(5.42) as follows:

$$\mathbf{R}_{\theta\phi\psi} = \mathbf{R}_z\mathbf{R}_y\mathbf{R}_x \tag{5.46}$$

where after decomposing, θ = rotation about the z-axis, ϕ = rotation about the y-axis, and ψ = rotation about the x-axis. We obtain the following equation:

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \tag{5.47}$$

$$\begin{bmatrix} \cos\phi\cos\psi & \cos\theta\sin\psi + \sin\theta\sin\phi\cos\psi & \sin\theta\sin\psi - \cos\theta\sin\phi\cos\psi \\ -\cos\phi\sin\psi & \cos\theta\cos\psi - \sin\theta\sin\phi\sin\psi & \sin\theta\cos\psi + \cos\theta\sin\phi\sin\psi \\ \sin\phi & -\sin\theta\cos\phi & \cos\theta\cos\phi \end{bmatrix}$$

We simplify the equations by multiplying both sides by $\mathbf{R}_z^{-1}$, which is equal to

$$\mathbf{R}_z = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.48}$$

obtaining

$$\begin{bmatrix} r_{11}\cos\psi - r_{21}\sin\psi & r_{12}\cos\psi - r_{22}\sin\psi & r_{13}\cos\psi - r_{23}\sin\psi \\ r_{11}\sin\psi + r_{21}\cos\psi & r_{12}\sin\psi + r_{22}\cos\psi & r_{13}\sin\psi + r_{23}\cos\psi \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\theta\sin\phi & -\cos\theta\sin\phi \\ 0 & \cos\theta & \sin\theta \\ \sin\phi & -\sin\theta\cos\phi & \cos\theta\cos\phi \end{bmatrix} \tag{5.49}$$

From Equation (5.49), we can extract

$$r_{11}\sin\psi + r_{21}\cos\psi = 0 \tag{5.50}$$

We further simplify Equation (5.50) to obtain

$$r_{11}\sin\psi/\cos\psi + r_{21} = 0 \tag{5.51}$$

which can be turned into

$$r_{11}\tan\psi + r_{21} = 0 \tag{5.52}$$

We finally obtain the value of ψ by the following:

$$\psi = \tan^{-1} - (r_{11}/r_{21}) \tag{5.53}$$

Having ψ, we can calculate the value of θ using Equation (5.49):

$$\begin{aligned} \sin\theta &= r_{13}\sin\psi + r_{23}\cos\psi \\ \cos\theta &= r_{12}\sin\psi + r_{22}\cos\psi \end{aligned} \tag{5.54}$$

which can be used to compute θ:

$$\theta = \tan^{-1}[(r_{13}\sin\psi + r_{23}\cos\psi)/(r_{12}\sin\psi + r_{22}\cos\psi)] \tag{5.55}$$

We now calculate ϕ using Equation (5.49) in a similar way:

$$\begin{aligned} \sin\phi &= r_{31} \\ \cos\phi &= r_{11}\cos\psi - r_{21}\sin\psi \end{aligned} \tag{5.56}$$

so that

$$\phi = \tan^{-1}[r_{31}/(r_{11}\cos\psi - r_{21}\sin\psi)] \tag{5.57}$$

Finally, we assign the following values:

$$\begin{aligned} &\text{Rotation about } x = \psi \\ &\text{Rotation about } y = \phi \\ &\text{Rotation about } z = \theta \end{aligned} \tag{5.58}$$

Distribution of Rotations

About the Longitudinal Axis

Some of the most common deformation systems do not yield great results when a joint or a bone rotates about its longitudinal axis. The lower arm is almost always a problem area in this respect.

If your character setup does not provide a mechanical solution for pronation and supination as specified in the character setup section, chances are that the longitudinal rotations of the lower arm are attributed to the wrist. An alternative to redesigning the skeletal structure is to distribute the wrist rotations equally between the wrist and the elbow. Better yet, distribute the total longitudinal rotations of the arm among the wrist, elbow, and shoulder joints. This is a very easy process, provided that the rotation along the longitudinal axis is the last transformation, which should be standard practice in any character setup.

When using captured motion data from a hierarchical rotational file, assuming that the longitudinal axis is y and that the rotation order is *ZXY* or *XZY*, you could simply solve the problem with a set of expressions as follows:

$$\begin{aligned}
&\text{TOTAL}_{\mathbf{R}_y} = \text{WRIST}_{\mathbf{R}_y} + \text{ELBOW}_{\mathbf{R}_y} + \text{SHOULDER}_{\mathbf{R}_y}\\
&\text{SHOULDER}_{\mathbf{R}_y} = \text{TOTAL}_{\mathbf{R}_y}/3\\
&\text{ELBOW}_{\mathbf{R}_y} = \text{TOTAL}_{\mathbf{R}_y}/3\\
&\text{WRIST}_{\mathbf{R}_y} = \text{TOTAL}_{\mathbf{R}_y}/3
\end{aligned} \tag{5.59}$$

You can distribute the rotations in different ways, perhaps applying less to the shoulder.

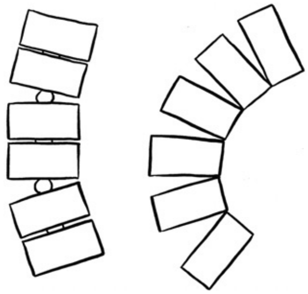

Figure 5.15 Distributed rotations with shifting pivot point.

With Shifting Pivot Point

Another interesting idea is to distribute rotations between joints in a way similar to the actual rotations of the spine, where a single vertebra has a maximum possible range of rotation. The rotation starts with the lowermost vertebra and continues until its maximum range has been reached. At that point, the rotation and the pivot point carry over to the next vertebra until it meets its maximum, and so on (see Figure 5.15). For example, if a vertebra has a range of rotation of 10° for a given axis, rotating the whole spine 36° would result in the rotation of the lowermost four vertebrae: the first three by 10° and a fourth by 6°. This method does not yield the same result as rotating the lowest vertebra in the spine by 35° along its single pivot point, but it is close to half, or 18°. Of course, this is dependent on the separation between vertebrae, which needs to be uniform.

The following pseudocode converts a single rotation group into a set of three distributed rotation groups with shifting pivot point. It assumes the rotations are stored in *rx*, *ry*, and *rz*, respectively; that y is the longitudinal axis; and that the rotation order is *XYZ*.

```
/* Declare various variables */
int num_joints = 3;                  /* Number of joints      */
float rot_vector[3];                 /* Incoming rotations    */
float rot_residual[3];               /* Carry over rotations */
float rot_joint_vector[num_joints][3];
                                     /* Outgoing rotations    */
float rot_joint_limit;               /* Maximum rot per joint */
int i;                               /* Loop variable         */
/* Write incoming rotations into array */
/* Double rx and rz to approximate result to single pivot */
rot_vector[0] = rx * 2;
rot_vector[1] = ry;
rot_vector[2] = rz * 2;
```

```
/* Set rotation limit per joint */
rot_joint_limit = 15;

/* Initialize residual array */
/* For longitudinal axis divide rotations equally */
rot_residual[0] = rot_vector[0];
rot_residual[1] = rot_vector[1] / num_joints;
rot_residual[2] = rot_vector[2];

/* Start processing rx */
/* If rx is greater than the limit, start distribution */
if (abs(rot_vector[0]) ] rot_joint_limit) {
   i = 0;

/* Start the distribution loop */
   /* Check if there is carry over and additional joints */
   while (abs(rot_residual[0]) ] 0 && i [ num_joints) {

      /* If rotations are positive */
      if (rot_vector[0] ] 0) {

           /* If the rotation left over is less than limit */
           /* apply it all to current joint*/

      if (abs(rot_residual[0]) [ rot_joint_limit)
           rot_joint_vector[i][0] += rot_residual[0];


      /* Otherwise, apply limit to current joint*/
      else
          rot_joint_vector[i][0] += rot_joint_limit;
      /* If joint is over limit, save extra in residual */
      if (rot_joint_vector[i][0] ] rot_joint_limit) {
          rot_residual[0] += rot_joint_vector[i][0] -
               rot_joint_limit;
          rot_joint_vector[i][0] = rot_joint_limit;
          }

      /* Subtract current rotation from new residual */
      rot_residual[0] -= rot_joint_limit;

      /* Make sure residual isn't negative */
      if (rot_residual[0] [ 0)
          rot_residual[0] = 0;
      }
```

```
    /* Process negative rotations */
    else {

        /* If the rotation left over is less than limit */
        /* apply it all to current joint*/

        if (abs(rot_residual[0]) [ rot_joint_limit)
            rot_joint_vector[i][0] += rot_residual[0];
/* Otherwise, apply limit to current joint*/
else

            rot_joint_vector[i][0] -= rot_joint_limit;

        /* If joint is over limit, save extra in residual */
        if (abs(rot_joint_vector[i][0]) ] rot_joint_limit) {
            rot_residual[0] -= (rot_joint_vector[i][0] -
                rot_joint_limit);
            rot_joint_vector[i][0] = -rot_joint_limit;
            }

        /* Subtract current rotation from new residual */
        rot_residual[0] += rot_joint_limit;
        /* Make sure residual isn't positive */
        if (rot_residual[0] ] 0)
            rot_residual[0] = 0;
        }
    i++;

    /* End of loop */
    }
}


/* If rx is smaller than the limit, apply all to first joint */
else {
      rot_residual[0] = 0;
      rot_joint_vector[0][0] = rot_vector[0];
      }

/* Process ry (longitudinal axis) */
i = 0;
while (i [ num_joints) {

        /* Rotate joint */
        rot_joint_vector[i][1] = rot_residual[1];
```

```
        i++;
        }

    /* Start processing rz */
    /* If rz is greater than the limit, start distribution */
    if (abs(rot_vector[2]) ] rot_joint_limit) {
        i = 0;

        /* Start the distribution loop */
        /* Check if there is carry over and additional joints */
        while (abs(rot_residual[2]) ] 0 && i [ num_joints) {

            /* If rotations are positive */
            if (rot_vector[2] ] 0) {

                /* If the rotation left over is less than limit */
                /* apply it all to current joint*/

                if (abs(rot_residual[2]) [ rot_joint_limit)
                   rot_joint_vector[i][2] += rot_residual[2];

                /* Otherwise, apply limit to current joint*/
                else
                   rot_joint_vector[i][2] += rot_joint_limit;

                /* If joint is over limit, save extra in residual */
                if (rot_joint_vector[i][2] ] rot_joint_limit) {
                   rot_residual[2] += rot_joint_vector[i][2] -
                       rot_joint_limit;
                   rot_joint_vector[i][2] = rot_joint_limit;
                   }

                /* Subtract current rotation from new residual */
                rot_residual[2] -= rot_joint_limit;

                /* Make sure residual isn't negative */
                if (rot_residual[2] [ 0)
                   rot_residual[2] = 0;
                }
        /* Process negative rotations */
        else {

            /* If the rotation left over is less than limit */
            /* apply it all to current joint*/
```

```
      if (abs(rot_residual[2]) [ rot_joint_limit)
         rot_joint_vector[i][2] += rot_residual[2];

   /* Otherwise, apply limit to current joint*/
      else
         rot_joint_vector[i][2] -= rot_joint_limit;

      /* If joint is over limit, save extra in residual */
      if (abs(rot_joint_vector[i][2]) ] rot_joint_limit) {
         rot_residual[2] -= (rot_joint_vector[i][2] -
            rot_joint_limit);
         rot_joint_vector[i][2] = -rot_joint_limit;
         }

      /* Subtract current rotation from new residual */
      rot_residual[2] += rot_joint_limit;

      /* Make sure residual isn't positive */
      if (rot_residual[2] ] 0)
         rot_residual[2] = 0;
      }
    i++;

    /* End of loop */
    }
   }
   /* If rz is smaller than the limit, apply all to first joint */
   else {
      rot_residual[2] = 0;
      rot_joint_vector[0][2] = rot_vector[2];
      }
   /* Apply new values to all joints */
```

Using Parametric Cubic Curves

Other methods for creating smoother relationships between joints involve the generation of curves, using markers or calculated points as control points. A parametric cubic curve can be generated using two or four points. To produce a parametric cubic curve using two points, use the following three polynomials of the third order:

$$\begin{aligned} x(u) &= a_x u^3 + b_x u^2 + c_x u + d_x \\ y(u) &= a_y u^3 + b_y u^2 + c_y u + d_y \\ z(u) &= a_z u^3 + b_z u^2 + c_z u + d_z \end{aligned} \tag{5.60}$$

where u is the parametric variable that has an interval from 0 to 1. $u = 0$ represents the first point, and $u = 1$ represents the second point. Any value in between can be used to calculate additional points as needed.

The slope of the curve is given by the two tangent vectors located at the ends of the curve. The components of each tangent vector are given by differentiating Equation (5.60) as follows:

$$\begin{aligned} x^u &= \frac{\mathbf{d}x(u)}{\mathbf{d}u} = \mathbf{3}a_x u^2 + \mathbf{2}b_x u + c_x \\ y^u &= \frac{\mathbf{d}y(u)}{\mathbf{d}u} = \mathbf{3}a_y u^2 + \mathbf{2}b_y u + c_y \\ z^u &= \frac{\mathbf{d}z(u)}{\mathbf{d}u} = \mathbf{3}a_z u^2 + \mathbf{2}b_z u + c_z \end{aligned} \tag{5.61}$$

By solving the equations using $u = 0$ and $u = 1$ where the two points are known, you can obtain the values of the coefficients (a, b, c, and d) and then proceed to calculate intermediate points.

When linking two point segments, such as in the vertebral column, make sure that the tangent vectors at the ends of the curves are matching, thus resulting in a single smooth curvature.

Interpolation

There are many instances when you will need to interpolate rotations between different captured motion data files. Common cases are the blending or stitching of different motions or creating a looping motion for actions like walking or running.

As I've stated in previous chapters, the success of blending or looping depends a great deal on the compatibility of the motion data. The points where the data files are to be blended need to be similar in their pose and have a similar timing of motion. If this is true, half the work is already done.

Quaternions

Rotations in captured motion data files are usually represented by Euler angles combined with a predefined order of rotations. This is not the best format to use when interpolating between two rotations because in order to reach the final orientation, three ordered rotations about separate axes have to occur. There are 12 combinations by which the final orientation can be reached, resulting in different representations of the 3 × 3 rotation matrix. Furthermore, Euler angles have the problem commonly known as *gimbal lock*, whereby a degree of freedom is lost due to parametric singularity. This happens when two axes become aligned, resulting in the loss of the ability to rotate about one of them.

Interpolating translations using Cartesian coordinates works mainly because translations are commutative, which means that to arrive at a certain location in space the order of translations is not important; thus, there are no dependencies between the transformations. Rotations represented by Euler angles are noncommutative, which means that their order cannot be changed and still yield the same final orientation. It also means that the transformations depend on each other.

Interpolation between orientations is best done by using *quaternions*, which, as their name implies (from Latin), are sets of four numbers, one of which represents a scalar part and three that represent a vector part. A quaternion q can be represented by the following equation:

$$q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k} \tag{5.62}$$

where the coefficients a, b, c, and d are real numbers, and $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ are the axes. As in complex numbers of the form

$$z = a + \mathbf{i}b \tag{5.63}$$

where $\mathbf{i}^2 = -1$, for quaternions, each of $\mathbf{i}^2$, $\mathbf{j}^2$, and $\mathbf{k}^2$ are also equal to -1. Thus, quaternions are extensions of complex numbers that satisfy the following identities:

$$\begin{aligned} &\mathbf{i}^2 + \mathbf{j}^2 + \mathbf{k}^2 = -1 \\ &\mathbf{ij} = -\mathbf{ji} = \mathbf{k} \\ &\mathbf{jk} = -\mathbf{kj} = \mathbf{i} \\ &\mathbf{ki} = -\mathbf{ik} = \mathbf{j} \end{aligned} \tag{5.64}$$

The condensed notation for a quaternion is

$$q(s, \mathbf{v}) \tag{5.65}$$

where s is the scalar part and $\mathbf{v}$ is the vector part, so that

$$\begin{aligned} &s = a \\ &\mathbf{v} = b\mathbf{i} + c\mathbf{j} + d\mathbf{k} \end{aligned} \tag{5.66}$$

Quaternion multiplication can be expressed as follows:

$$q_1 q_2 = (s_1 s_2 - \mathbf{v}_1 \cdot \mathbf{v}_2, s_1 \mathbf{v}_2 + s_2 \mathbf{v}_1 + \mathbf{v}_1 \times \mathbf{v}_2) \tag{5.67}$$

Quaternion multiplication is noncommutative, so $q_1 q_2$ is not the same as $q_2 q_1$. The magnitude of a quaternion can be determined by the following equation:

$$|q| = \sqrt{q\bar{q}} = \sqrt{s^2 + |\mathbf{v}^2|} \tag{5.68}$$

where $\bar{q}$ is the conjugate of the quaternion, defined as

$$\bar{q} = q(s, -\mathbf{v}) \tag{5.69}$$

A unit quaternion has a magnitude of 1, so from Equation (5.68) we can determine that

$$q\bar{q} = 1 \tag{5.70}$$

where q is a unit quaternion. Unit quaternions are used to represent rotations and can be portrayed as a sphere of radius equal to one unit. The vector originates at the sphere's center, and all the rotations occur along its surface. Interpolating using quaternions guarantees that all intermediate orientations will also fall along the surface of the sphere. For a unit quaternion, it is given that

$$a^2 + b^2 + c^2 + d^2 = 1 \tag{5.71}$$

The inverse of a quaternion is defined as

$$q^{-1} = \frac{\bar{q}}{|q|^2} \tag{5.72}$$

but for a unit quaternion it can be reduced to

$$q^{-1} = \bar{q} \tag{5.73}$$

A unit vector $\mathbf{p}$ can be represented in quaternion notation as one without a scalar part, also called a *pure* quaternion:

$$P(0, \mathbf{p}) \tag{5.74}$$

The rotation of $\mathbf{p}$ can be computed by the following equation:

$$P' = qPq^{-1} \tag{5.75}$$

where P' is also a pure quaternion. This rotation can also be achieved by applying the following matrix:

$$\mathbf{R} = \begin{bmatrix} 1-2c^2-2d^2 & 2bc-2ad & 2bd+2ac & 0 \\ 2bc+2ad & 1-2b^2-2d^2 & 2cd-2ab & 0 \\ 2ad-2ac & 2cd+2ab & 1-2b^2-2c^2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.76}$$

Before we start looking at the interpolation process, we need to establish that a rotation of θ about a unit vector $\mathbf{p}$ can be performed by using the following unit quaternion:

$$q(s, \mathbf{v}) = (\cos(\theta/2), \mathbf{p}\ \sin(\theta/2)) \tag{5.77}$$

which means that separate rotations about the x, y, and z axes can be represented as

$$\begin{aligned} q_x &= (\cos(\theta/2), \sin(\theta/2), 0, 0) \\ q_y &= (\cos(\phi/2), 0, \sin(\phi/2), 0) \\ q_z &= (\cos(\psi/2), 0, 0, \sin(\psi/2)) \end{aligned} \tag{5.78}$$

respectively. We can now convert our Euler angles from the captured motion data file into quaternion space by simply multiplying the quaternions in the proper order. For example, a rotation in *XYZ* order would be given by using Equation (5.67) to perform the following multiplication:

$$q_{xyz} = q_x q_y q_z \tag{5.79}$$

Converting a rotation matrix to a quaternion is a simple process. Let us assume we have a 4 × 4 transformation matrix of the form

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.80}$$

We obtain the *trace*, which is the sum of its diagonal elements r_{11}, r_{22}, and r_{33} so that

$$\mathrm{Tr}(\mathbf{R}) = r_{11} + r_{22} + r_{33} \tag{5.81}$$

To find the converted quaternion $q(W, X, Y, Z)$, we start by finding the value of the scalar W using Equation (5.81):

$$W = \frac{\sqrt{\mathrm{Tr}(R)}}{2} \tag{5.82}$$

The vector elements are calculated as follows:

$$X = (r_{12} - r_{21})/4W \tag{5.83}$$

$$Y = (r_{20} - r_{02})/4W \tag{5.84}$$

$$Z = (r_{01} - r_{10})/4W \tag{5.85}$$

Once we are in quaternion space, we can proceed with the interpolation. The most commonly used interpolation method for quaternions is called *spherical linear interpolation*, or *slerp*. Because of its spherical nature, this method guarantees that any intermediate quaternions will also be unit quaternions. Slerp computes the angle θ between both quaternions as vectors in two-dimensional space, using their scalar product. If we have two quaternions q_1 and q_2, we can find θ as follows:

$$\theta = \cos^{-1}(q_1 \cdot q_2) \tag{5.86}$$

To calculate quaternion q' in between q_1 and q_2 using slerp, we use a parametric variable u with interval from 0 to 1 and the following equation:

$$q' = q_1\left(\frac{\sin[(1-u)\theta]}{\sin\theta}\right) + q_2\frac{\sin u\theta}{\sin\theta} \tag{5.87}$$

If θ is equal to 0, the slerp will run into computational problems due to division by zero. If this is the case, instead of using slerp, we just use linear interpolation. Other problems with slerp occur when θ is equal to multiples of $\pi/2$. Those problems can be avoided by changing either one of q_1 or q_2 to an approximation. For example, we can use q_2 and q_3, an adjacent keyframe, to generate a third quaternion q_4 that is as close to q_2 as possible without falling in the problem area; then we could proceed with the original interpolation using q_1 and q_4 instead of q_1 and q_2.

We now must convert q' (W, X, Y, Z) back into a rotation matrix **R** of the form shown in Equation (5.80). We use the matrix in Equation (5.76), which now becomes

$$\mathbf{R} = \begin{bmatrix} 1-2Y^2-2Z^2 & 2XY-2WZ & 2XZ+2WY & 0 \\ 2XY+2WZ & 1-2X^2-2Z^2 & 2YZ-2WX & 0 \\ 2XZ-2WY & 2YZ+2WX & 1-2X^2-2Y^2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.88}$$

We can finally decompose **R** into Euler angles if we need to using the method explained in the section "Switching the Order of Rotations."

Keyframe Reduction

In some situations, you may want to reduce the number of keyframes in your data. This is not the same as downsampling, as you will keep the same sampling rate or frames per second at which the data is supposed to play back. Depending on what the purpose is, you may be able to reduce the number of keyframes automatically, but for the highest quality you may have to do it manually.

Removing every nth keyframe is not a good method because it doesn't take into account the ratio of change from one keyframe to the next. The preferred method is to find areas in your curve that are smooth and don't fluctuate too much, and then delete all keyframes that fall within two predetermined boundary keyframes. This is a long process if you have a character with dozens of joints, but you can accelerate it by using an algorithm that looks for differences in the curves.

There are many commercial software solutions that allow you to perform several types of fitting to your curves. Many load your data into a spreadsheet where you can perform a variety of manual or automatic operations, including smoothing and other noise reduction methods. A good method for curve fitting is the least squares method, which most software packages support. Matlab is probably the most popular software package that can be used to do all kinds of curve analysis and manipulations.

CONCLUSION

Motion capture is not a new technology, but it is still not completely evolved in the field of performance animation. There are now better tools to deal with the data, but it is still necessary to have specialized technical people to deal with it.

Because of the proliferation of performance capture projects, there are more and more people using motion capture on a regular basis. The bad reputation that used to surround motion capture in the computer animation industry is virtually gone and, although it isn't liked by everyone, it is a valuable tool in the visual effects arsenal and it will remain so for time to come.

It wasn't possible to cover each and every topic pertaining to motion capture, specifically, the software-specific aspects of using motion capture data in a production environment. My intention is to present a general view with enough low-level information for the reader to understand all the concepts when reading documentation of specific software or a high-level tool, or even when writing such a tool. The captured motion data files that are dissected in Chapter 4 should help you understand any other format that may be out there, and you should be able to write a converter for any combination of formats. The character setup example is definitely not best suited for every case, but the concepts should be useful to anybody setting up a character for motion capture, where external markers need to be converted to an internal skeleton. In addition, most common motion capture problems can be solved with similar math procedures as the ones used in Chapter 5. Nothing is as valuable, though, as the experiences and commentary from people that have used the tool. The best perspective of the process is gained by reading and using those experiences in your own projects.

I tried to write this book in a less academic and more production-oriented language, perhaps dealing with problems in a way that may not be mathematically perfect, but one that is well fitted for an animation production environment. After all, this is not medicine, and we are allowed to manipulate the numbers somewhat, as long as we obtain what we are looking for visually, and we must take advantage of that freedom.

One of the main purposes of the original edition of this book was to defuse the misconceptions in the entertainment industry. I feel the technology is about to reach an important point in its maturity where new and exciting methodologies will allow us to use it for far-reaching applications that may actually make the world a better place. The deployments of new consumer products, mainly in the gaming arena, that use human motion as a controlling device are evidence of what is to come.

APPENDIX A. MOTION CAPTURE EQUIPMENT AND SOFTWARE MANUFACTURERS

Optical Tracking Systems

Ariel Dynamics

http://www.arielnet.com
4885 Ronson Court, Suite A
San Diego, CA 92111
USA
Tel: +1 (858) 874-2547
Fax: +1 (858) 874-2549
E-mail: info@arielnet.com

Product:

APAS: Video-based 3D motion analysis system which provides objective mechanical data.

Codamotion

http://www.codamotion.com
Fowke Street
Rothley
Leicestershire LE7 7PJ
UK
Tel: +44 116 230 1060
Fax: +44 116 230 1857

Product:

CODA: Codamotion's measurement technology uses miniature infrared active markers whose signals are beamed to three linear arrays which provide a real-time 3D measurement.

Euclid Research

http://www.euclidres.com
2 North 1st Street, 6th Floor

San Jose, CA 95113-1201
USA
Tel: +1 (408) 283-9020
Fax: +1 (408) 283-9029

Product:

Motion Capture Card: PC-compatible card that provides interfaces to incremental optical encoders. Used to digitize angular/linear position, velocity, and acceleration in order to measure motion parameters.

GestureTek Technologies

http://www.gesturetek.com
530 Lakeside Drive, Suite 280
Sunnyvale, CA 94085
USA
Tel: +1 (408) 216 8087
Fax: +1 (408) 732 3977

Product:

3D gesture control technology: It uses cameras to capture depth information surrounding a user's movements. It measures changes in positioning and distance between the various parts of a user's image and other elements within the scene.

Innovision Systems Inc.

http://www.innovision-systems.com
3717 Peters Road
Columbiaville, MI 48421-9304
USA
Tel: +1 (810) 793 5530
Fax: +1 (810) 793 1714

Product:

MaxPro: Low-cost optical motion analysis system that uses nonproprietary cameras.

iPi Soft

http://www.ipisoft.com
E-mail: ipi@ipisoft.com

Product:

iPi Desktop Motion Capture: iPi Desktop Motion Capture is an entry-level markerless optical motion capture technology that is targeted at small studios, freelance animators, and motion capture enthusiasts.

Lutz Mechatronic Technology

http://www.lukotronic.com
Technologie- und Wirtschaftspark Innsbruck
Eduard-Bodem-Gasse 5-7
6020 Innsbruck
Austria
Tel: +43 512 363631-1
Fax: +43 512 36363-15
E-mail: office@lukotronic.com

Product:

LUKOtronic: Optical infrared cameras with active markers, force plates, accessories, and appropriate software for data acquisition and data processing.

Mikromak Service

http://www.mikromak.com
Bernhard-Lichtenberg-Str. 10
10407 Berlin
Germany
Tel: +49-30-42022-402
Fax: +49-30-42022-401
E-mail: info@mikromak.com

Product:

Motion Analysis System: High-speed cameras and measurement software (tracking, motion analysis) as complete systems in 2D and 3D.

Motion Analysis

http://www.motionanalysis.com
3617 Westwind Boulevard
Santa Rosa, CA 95403
USA
Tel: +1 (707) 579 6500
Fax: +1 (707) 578 8473
General Email: info@motionanalysis.com

Product:

Eagle-4 Optical Motion Capture System: Eagle-4 Digital Cameras and Cortex software.

Motion Capture Technologies

http://www.mctcameras.com
P.O. Box 1046
Valdese, NC 28690
Tel: +1 (828) 874 2284
Fax: +1 (828) 874 2285
E-mail: sales@mctcameras.com

Product:

Xcitex ProAnalyst: Software package for measuring motion from video. Used in engineering, design, medicine, and other sciences.

Natural Point

http://www.naturalpoint.com
P.O. Box 2317
Corvallis, OR 97339
USA
Tel: +1 (541) 753-6645
Fax: +1 (541) 753-6689

Product:

OptiTrack: Optical motion capture and tracking solutions for commercial, industrial, game development, and research.

NDI

http://www.ndigital.com
103 Randall Drive
Waterloo
Ontario, Canada N2V 1C5
Tel: +1 (519) 884 5142
E-mail: info@ndigital.com

Products:

Polaris and Optotrak: Optical tracking systems that support active and passive markers.

Organic Motion

http://www.organicmotion.com
336 West 37th Street, 7th Floor
New York, NY 10018
USA
Tel: 212-776-6100
Fax: 212-273-3777

Product:

Organic Motion Capture Technology: Optical markerless capture system.

PhaseSpace

http://www.phasespace.com
1933 Davis Street, Suite 304
San Leandro, CA 94577
USA
Tel: +1 (510) 633-2865
Fax: +1 (925) 945-6718

Product:

IMPULSE: Active LED marker optical motion capture system.

Phoenix Technologies Incorporated

http://www.ptiphoenix.com
4302 Norfolk St.
Burnaby, B.C.
Canada V5G 4J9
Tel: +1 (604) 321-3238
Fax: +1 (604) 321-3286

Product:

Visualeyez: Optical motion capture/tracking system.

Qualisys

http://www.qualisys.com
Packhusgatan 6, S-411 13
Gothenburg, Sweden
Tel: +46 31 336 94 00
Fax: +46 31 336 94 20
Email: sales@qualisys.com

Product:

Qualisys Motion Capture System: System for motion capture and analysis of movement data based on the Oqus cameras and the Qualisys Track Manager software.

Vicon

http://www.vicon.com
5419 McConnell Avenue
Los Angeles, CA 90066
USA
Tel: +1 (310) 306-6131
Fax: +1 (310) 437-4229
E-mail: moveme@vicon.com

Products:

Vicon MX: State-of-the-art optical motion capture system with cameras of up to 16 megapixels.

Vicon Bonita: Entry-level optical motion capture system with compact cameras.

Vicon Motus Video: Solution for both video-based and optical tracking.

Electromagnetic Trackers

Ascension Technology Corporation

http://www.ascension-tech.com
P.O. Box 527
Burlington, VT 05402
USA
Tel: +1 (802) 893-6657
Fax: +1 (802) 893-6659
E-mail: jscully@ascension-tech.com

Product:

Flock of Birds: Pulsed DC magnetic 6-degrees-of-freedom tracker with a universal interface.

Polhemus

http://www.polhemus.com
40 Hercules Dr., PO Box 560
Colchester, VT 05446
USA

Tel: +1 (800) 357-4777
Fax: +1 (802) 655-3159
E-mail: sales@polhemus.com

Products:

LIBERTY: 6-degree-of-freedom electromagnetic tracker with scalable support for up to 16 sensors.

FASTRAK: Real time 6-degree-of-freedom tracker for head, hand, and instrument tracking, biomechanical analysis, graphics and cursor control, digitizing and pointing, stereotaxic localization, and telerobotics.

MINUTEMAN: 3-degree-of-freedom orientation tracking system.

Sensor-Based Motion Capture Systems

Advanced Motion Measurement

http://www.amm3d.com
1202 E. Maryland Ave. Ste. 1J
Phoenix, AZ 85014
USA
Tel: +1 (602) 263-8657
Fax: +1 (602) 277-2326
E-mail: steve@amm3d.com

Product:

AMM Graph 3D: Real-time wireless Bluetooth-enabled 3D sensor. It provides motion data by employing three miniature tri-axial MEMS chips to capture acceleration, the earth's magnetic vector, and angular velocity.

InnaLabs

http://www.innalabs.com
Landyshevaya str. 12
Moscow
Russian Federation, 125466
Tel: +1 703 286 5427
E-mail: contact.sales@innalabs.com

Product:

Attitude and Heading Reference Systems: High-performance strapdown system that determines the full angular orientation of any vehicle or the carrier in the 3D space.

Intersense

http://www.intersense.com
4 Federal Street
Billerica, MA 01821
USA
Tel: +1 (781) 541 6330
Fax: +1 (781) 541 6329
E-mail: ISinfo@intersense.com

Product:

Inertia Cube sensors: The InertiaCube2+ integrates nine discrete miniature sensing elements utilizing advanced Kalman filtering algorithms to produce a full 360° sourceless orientation tracking sensor.

Measurand

http://www.measurand.com
2111 Hanwell Road
Fredericton, New Brunswick
Canada E3C 1M7
E-mail: sales@measurand.com
Tel: +1 (506) 462 9119
Fax: +1 (506) 462 9095

Product:

ShapeWrap III: Wireless real-time motion capture system based on fiber optic 3D bend and twist sensors.

Trivisio

http://www.trivisio.com
Karcherstrasse 10
67655 Kaiserslautern
Germany
Tel: +49 631-4125111
Fax: +49 631-4126127
E-mail: info@trivisio.com

Product:

Colibri: 3DOF inertial measurement unit that includes 3-axis sensors to measure acceleration, angular rate, and magnetic field.

Xsens

http://www.xsens.com
Pantheon 6a
7521 PR ENSCHEDE
The Netherlands
Tel: +31 88 97367 00
Fax: +31 88 97367 01
E-mail: sales@xsens.com

Product:

Xsens MVN Inertial Motion Capture System: Consists of MEMS (Micro-Electro-Mechanical Systems) inertial sensors attached to the body by a Lycra suit.

Ultrasonic Systems

Hexamite Ultrasound

http://www.hexamite.com
P.O. Box 24
Sunny Beach 8240
Bulgaria
E-mail: contactus@hexamite.com

Product:

Hx11: General-purpose ultrasonic positioning system.

Hand Capture Systems

CyberGlove Systems

http://www.cyberglovesystems.com
2355 Paragon Drive, Suite D
San Jose, CA 95131
USA
Tel: +1 (408) 451 9463
Fax: +1 (408) 689 4362

Product:

Cyberglove II: Wireless 18-sensor motion capture data glove system.

Fifth Dimension Technologies

http://www.5dt.com
15375 Barranca Pkwy, G-103
Irvine, CA 92618
USA
Tel: +1 (949) 450 9044
Fax: +1 (949) 450 9045

Product:

5DT Data Glove: Wired and wireless right- and left-handed real-time capture gloves with multiple application drivers.

noDna

http://www.x-ist.de
Kronenstr. 70, 10117
Berlin
Germany
Tel: +49 30-544915-38
Fax: +49 30-544915-39
E-mail: sales@nodna.com

Product:

X-IST DataGlove 3D System: X-IST DataGlove is made out of two gloves, an inner glove with the attached sensors and an outer glove to protect the electronics and provide the grip needed for ergonomic research and manipulation of objects.

Eye Tracking Systems

Applied Science Laboratories

http://www.asleyetracking.com
175 Middlesex Turnpike
Bedford, MA 01730
USA
Tel: +1 (781) 275 4000
Fax: +1 (781) 275 3388
E-mail: asl@ASLeyetracking.com

Product:

Mobile Eye: It collects eye movements and point of gaze information during the performance of natural tasks allowing the use of unconstrained eye, head, and hand movements under variable lighting conditions.

Sensomotoric Instruments

http://www.smivision.com
Warthestraße 21
D-14513 Teltow
Berlin
Germany
Tel: +49 (3328) 3955 0
Fax: +49 (3328) 3955 99
E-mail: info@smi.de

Product:

iView X HED: Mobile eye-tracking system for indoor and outdoor use. It includes a lightweight headset and laptop PC.

Smart Eye

http://www.smarteye.se
Första Långgatan 28 B
413 27 Göteborg
Sweden
Tel: + 46 31 60 61 60
Fax: + 46 31 701 05 15
E-mail: info@smarteye.se

Product:

Smart Eye Pro: Optical eye tracking technology used mostly in the automotive industry.

SOFTWARE

Autodesk

http://www.autodesk.com
111 McInnis Parkway
San Rafael, CA 94903
USA
Tel: 800-578-3375

Product:

MotionBuilder: Real-time 3D character animation software for high-volume animation, virtual cinematography, previsualization, and performance capture.

Contemplas

http://www.contemplas.com
Albert-Einstein-Straße 6
87437 Kempten (Allgäu)
Germany
Tel: +49 (831) 254369-20
Fax: +49 (831) 5645328
E-Mail: info@contemplas.com

Product:

Templo motion analysis software: Specialized for the markets of biomechanics, sports, and medicine.

Innovative Sports Training

http://www.innsport.com
3711 N. Ravenswood Ave., Suite 150
Chicago, IL 60613
USA
Tel: +1 (773) 244-6470
Fax: +1 (773) 244-6473
E-mail: support@innsport.com

Product:

The MotionMonitor: Data acquisition, analysis, and visualization system for third-party motion capture systems. Used mostly for life-sciences applications.

Mocap.CA

http://www.mocap.ca
E-mail: al@mocap.ca

Products:

Peelsolve: Global Optimization Motion Capture Solver plugin for Autodesk Maya.
Peelanim: Animation Clip Importer and Exporter Plugin for Autodesk Maya.
Peelfarm: Distributed render/process management system.

Simi Reality Motion Systems

http://www.simi.com
Max-Planck-Str. 11

D-85716 Unterschleissheim
Germany
Tel: +49 89-321459-0
Fax: +49 89-321459-16

Products:

Simi MotionCapture3D: Software solution for capturing and manipulating real motion sequences in 3D for entertainment applications.

Simi MotionTwin: Video analysis software which provides support for animators. It permits simultaneous capturing and analysis from up to four video sources.

Vicon

http://www.vicon.com
5419 McConnell Avenue
Los Angeles, CA 90066
USA
Tel: +1 (310) 306-6131
Fax: +1 (310) 437-4229
E-mail: moveme@vicon.com

Products:

Vicon Blade: Software solution that includes most elements needed for motion capture, from system configuration to actor setup and delivery mocap data. Aimed at film, game, broadcast, and postproduction studios.

Vicon Nexus: Life-science–specific motion capture software.

Vicon Polygon: Software package designed to generate interactive multimedia reports for clinicians, surgeons, sports therapists, and researchers.

Markers and Mocap Suits

3X3 Designs

http://www.3x3.bc.ca/mocap.htm
#19-145 Schoolhouse Street
Coquitlam, British Columbia
V3K 4X8 Canada
E-mail: info@3x3.bc.ca
Tel: +1 (604) 777-2555
Fax: +1 (604) 777-2508

Products:

Motion capture markers, suits, and accessories.

Mocap Solutions

http://www.mocapsolutions.com
20661 Egret Lane
Huntington Beach, CA 92646
USA
Tel: +1 (714) 969 6520
Fax: +1 (714) 969 4808
E-mail: sales@mocapsolutions.com

Products:

Motion capture markers, suits, and accessories.

APPENDIX B. MOTION CAPTURE SERVICE PROVIDERS

Australia

Act3Animation

http://www.act3animation.com
9 Warwick Farm Road
Olinda, Victoria, 3788
Australia
Tel: +61 3 9510 8943
Fax: +61 3 9510 8953
E-mail: info@Act3Animation.com

Beyond Motion

http://beyondmotion.com.au
First Floor-94
Burswood Rd
Burswood, WA
Australia
Tel: +61 4 2218 1916
Facsimile: +61 8 9358 0062
E-mail: info@beyondmotion.com.au

Plastic Wax

http://www.plasticwax.com
Unit D8
27-29 Fariola St
Silverwater NSW 2128
Australia
Tel: +61 2 9743 0700
E-mail: info@plasticwax.com

Austria

7Reasons Motion Capture

http://www.motioncapture.at
Dr.-Adolf-Altmann-Straße 30, A-5020
Salzburg
Austria
Tel: +43-662-633 43 420
E-mail: salzburg@7reasons.at

Canada

Fast Motion Studios

http://www.fastmotionstudios.com
2 Denison Road West
Toronto, ON M9N 1C1
Canada
Tel: +1 (416) 244-3664
Fax: +1 (416) 241-1606

Rainmaker

http://www.rainmaker.com
Suite 500
2025 West Broadway
Vancouver, BC
Canada, V6J 1Z6
Tel: +1 (604) 714-2600
Fax: +1 (604) 714-2641

Czech Republic

Bohemia Interactive Studio

http://pro.bistudio.com
Stribrna Lhota 747
Mnisek pod Brdy
Czech Republic 25210
Tel: +420 606 292 422

France

Mocaplab

http://www.mocaplab.com
5, cité Riverin

75010 Paris
France
Tel: +33 9 54 05 38 17
E-mail : contact@mocaplab.com

Germany

MetricMinds

http://www.metricminds.de
Rüsselsheimer Strasse 22
60326 Frankfürt am Main
Germany
Tel: +49 69 759 3380
Fax: +49 69 759 338 29
E-mail: info@metricminds.com

Rocketbox Studios

http://www.rocketbox.de
Leonhardtstr. 10
30175 Hannover
Germany
Tel: +49 511 8984384
Fax: +49 511 8984386
E-mail: info@rocketbox.de

Greece

Motion Capture 3D

http://motioncapture3d.com
Spetson 1
Kifissia 14561
Greece
Tel/Fax: +30 210 8072271
E-mail: info@motioncapture3d.com

Hungary

3D Brigade

http://www.3dbrigade.com
Budapest
Hungary
E-mail: info@3dbrigade.com

India

Accel Animation Studios

http://www.accelanimation.com
Kinfra Film and Video Park
Kazhakuttam 695585
Thiruvananthapuram
India
Tel: +91 471 2417434
E-mail: murali@accelanimation.com

Mobility Art Studios

http://www.mobilityart.com
2A, Annapurna Studios
Road No. 2, Jubilee Hills
Hyderabad 500033
India
E-mail: info@mobilityart.com

Lithuania

Mocap.LT

Fabijoniskiu 90-60
Vilnius, LT-07100
Lithuania
Tel: +370 65 905 905
E-mail: info@mocap.lt

Malaysia

Inner Esteem Studios

http://www.motioncapturestudios.com
6-3, Jalan 15/48A
Off Jalan Sentul
51000 Kuala Lumpur
Malaysia

Mexico

Gyroscopik Studios

http://www.gyroscopik.com
224 Sv. Cubilete

Col. Chapalita Sur
Guadalajara, Jalisco
Mexico
Tel: +52 33 3335 2672
E-mail: info@gyroscopik.com

Netherlands

Motek Entertainment

Keienbergweg 75
1101GE Amsterdam
Netherlands
Tel: +31 20 4191111
Fax: +31 20 4192222
E-mail: info@motekentertainment.com

Russia

Animaccord Animation

http://mocap.animaccord.ru
Godovikova 9, bld.3
Moscow
Russia
Tel: +7 (495) 775-13-31
E-mail: mocap@animaccord.com

Icehill Entertainment

http://icehill.ru
408, 9a Kraula (BC “Olimp”)
Yekaterinburg, 620028
Russia
E-mail: info@ice-hill.com

Serbia

Centroid 3D Serbia

http://www.centroid3d.com
Dositejeva 12
11 000 Beograd
Serbia
Tel: +381 11 262 4129

Spain

Mocapservice

http://www.mocapservice.com
Fargaires 4A
08290/Cerdanyola del Vallés/Barcelona
Spain
Tel: +34 935 938-185
Fax: +34 935 790-320
E-mail: alinas@eddadesign.com

Sweden

Imagination Studios

http://www.imaginationstudios.com
Kungsgatan 30
S-753 11 Uppsala
Sweden
Tel: +46 18-10 6930
Fax: +46 18-14 6930
Email: info@imaginationstudios.com

Switzerland

Moka Studio

http://www.mokastudio.tv
Rue Marconi, 19
1920 Martigny
Switzerland
Tel: +41 27 566 70 17
E-mail: info@mokastudio.tv

United Kingdom

Animazoo

http://www.animazoo.com
Basin Road South
Brighton
West Sussex
UK
BN41 1WF
Tel: +44 1273 418 641

Audiomotion Studios

http://www.audiomotion.com
Holywell House
Osney Mead, Oxford
OX2 0EA
United Kingdom
Tel: +44 8701 600 504
Fax: +44 1865 728 319
E-mail: info@audiomotion.com

Centroid Motion Capture

http://www.centroid3d.com
Pinewood Studios
Pinewood Road
Iver Heath
Buckinghamshire
SL0 0NH
United Kingdom
Tel: +44 1753 630202

Imagemetrics

http://www.image-metrics.com
1 Portland Street
Manchester M1 3BE
United Kingdom
Tel: +44 161 242 1800
Fax: +44 161 242 1801

United States

8 Hats High Animation & Production

http://www.8hatshigh.com
23-27 West Main Street. 3rd Floor
Middletown, NY 10940
USA
Tel: +1 (845) 344-1888

Auvis Studios

http://www.auvisstudios.com
70 Glen Street
Glen Cove, Long Island, NY

USA
Tel: +1 (516) 749-3052
E-mail: bobd@auvisstudios.com

Captive Motion

http://www.captivemotion.com
302 S. River Dr.
Tempe, AZ 85281
USA
Tel: +1 (480) 557-6222

Center for Human Performance

http://www.sdmocap.com
3020 Childrens Way MC 5054
San Diego, CA 92123
USA
Tel: +1 (858) 966-8415
E-mail: info@sdchp.com

Cinematico

http://www.cinematico.com
1504 Bryant Street, Suite 333
San Francisco, CA 94103
USA
Tel: +1 (415) 896-5776
E-mail: info@cinematico.com

Critical Moves

http://www.criticalmovesusa.com
4925 Cadieux Rd
Detroit, MI 48224
USA
Tel: +1 (248) 505-4030
E-mail: bizdev@criticalmovesusa.com

Curious Pictures

440 Lafayette Street
6th Floor
New York, NY 10003
USA
Tel: +1 (212) 674-1400

Fax: +1 (212) 674-0081
E-mail: info@curiouspictures.com

Digital Double

http://www.digitaldouble.net
15814 Bear Creek Parkway, Second Floor
Redmond, WA 98052
USA
Tel: +1 (425) 867-0400
E-mail: info@digitaldouble.net

Elektrashock

http://www.elektrashock.com
1320 Main St.
Venice, CA 90291
USA
Tel: +1 (310) 399-4985
Fax: +1 (310) 399-4972
E-mail: info@elektrashock.com

Fay Media Group

http://www.mocapcolorado.com
5201 East 48th Avenue
Denver, CO 80216
USA
Tel: +1 (303) 333-5209

Fitwerx Cycling Specialists

http://www.fitwerx.com
4312 Main Street
Waitsfield, VT 05673
USA
Tel: +1 (802) 496-7570
E-mail: info@fitwerx.com

Giant Studios

http://www.giantstudios.com
12615 Beatrice Street
Los Angeles, CA 90066
USA
Tel: +1 (310) 839-1999

Fax: +1 (310) 862-5122
E-mail: projects@giantstudios.com

House of Moves

http://www.moves.com
5419 McConnell Avenue
Los Angeles, CA 90066
USA
Tel: +1 (310) 306-6131
Fax: +1 (310) 306-1351

IBC Mobile Motion

http://www.mobilemotionmocap.com
1945 Gardena Avenue
Glendale, CA 91204
USA
Tel: +1 (818) 244-3575
Fax: +1 (818) 244-5325
E-mail: Eric@mobilemotionmocap.com

Iseesoft Studios

http://www.iseesoft.com
1776 Mentor Ave.
Suite 179
Cincinnati, OH 45212
USA
Tel: +1 (513) 351-0000
Fax: +1 (513) 351-0610
E-mail: info@iseesoft.com

Just Cause Productions

http://www.for-the-cause.com
4096 Glencoe Ave
Marina del Rey, CA 90292
USA
Tel: + 1 (310) 439-9721
Fax: +1 (310) 827-2787
E-mail: info@for-the-cause.com

Kidz Korner Studio

http://www.kkstudio.us
8144 Ventana Azul Ave. NW
Albuquerque, NM 87114

USA
Tel: +1 (505) 615-2410
E-mail: bryan@kkStudio.us

Mods

http://www.themods.tv
437 North Varney St.
Burbank, CA 91502
USA
Tel: +1 (661) 478-4052
E-mail: contact@themods.tv

Motion Analysis Studios

http://www.mastudios.com
1041 North Mansfield Avenue
Hollywood, CA 90038
USA
Tel: +1 (323) 461-3835
Fax: +1 (323) 461-3837
E-mail: info@mastudios.com

Motion Capture NYC

http://www.motioncapturenyc.com
61 Rivington Street
New York, NY 10002
USA
Tel: +1 (917) 843-5647

Motion Golf

http://www.motiongolf.com
5100 PGA Boulevard, #101
Palm Beach Gardens, FL 33418
USA
Tel: +1 (561) 537-5484

Motion Reality

http://www.motionrealityinc.com
200 North Cobb Parkway, Suite 140
Marietta, Georgia 30062
USA
Tel: +1 (770) 424-8195

Motus Digital

http://motusdigital.com
2805 E. Plano Parkway, Suite 250
Plano, TX 75074
USA
Tel: +1 (972) 943-0008
Fax: +1 (972) 943-0009
E-mail: Info@MotusDigital.com

Mova

http://www.mova.com
2565 Third Street, Unit 311
San Francisco, CA 94107
USA
Tel: +1 (415) 947-5590
E-mail: mocap3@mova.com

NIMS Center

http://www.digitalcinemastudio.com
800-824 Distributors Row, Suite 201
New Orleans, LA 70123
USA
Tel: +1 (504) 430-8965
E-mail: nimscenterstudio@yahoo.com

Perspective Studios

http://www.perspectivestudios.com
70 Glen Street, Suite LL
Glen Cove, NY 11542
Tel: +1 (516) 247-6022
E-mail: info@perspectivestudios.com

Red Eye Studio

http://www.redeye-studio.com
215 Stonington Ave. #122
Hoffman Estates, IL 60169
USA
Tel: +1 (847) 843-2438
E-mail: info@redeye-studio.com

Retul Fit Studios

http://www.retul.com
2040 30th Street, Suite C
Boulder, CO 80301
USA
Tel: +1 (720) 406-1171
E-mail: info@retul.com

Taylormade Performance Labs

http://www.tmplabs.com
Corporate Headquarters
945 Hotel Circle South
San Diego, CA 92108
USA
Tel: (619) 297-8700
Fax: (619) 297-3180

Worley Works Productions

http://worleyworks.com
117 Grattan Room 203,
Brooklyn, NY 11237
USA
Tel: +1 (646) 797-2941
E-mail: info@worleyworks.com

APPENDIX C.
WEB RESOURCES

Motion Capture Associations

Mocap Club
http://www.mocapclub.com

Motion Capture Society
http://www.motioncapturesociety.com

Motion Capture Research Institutions and Organizations

Bio Motion Lab
http://www.biomotionlab.ca

C3D Home
http://www.c3d.org

Carnegie Mellon Master Motion Project
http://www.etc.cmu.edu/projects/mastermotion

Deakin Motion Lab
http://www.deakin.edu.au/motionlab

ETH Zürich Computer Vision Laboratory
http://www.vision.ee.ethz.ch

Gait and Clinical Movement Analysis Society
http://www.gcmas.org

Institute for Creative Technologies
http://ict.usc.edu

Miralab
http://www.miralab.ch

NYU Movement Lab
http://movement.nyu.edu

Ohio State University ACCAD Motion Capture Lab
http://accad.osu.edu/research/mocap

Riverside Graphics Lab
http://graphics.cs.ucr.edu

Stanford Movement Lab
http://movement.stanford.edu

UC Merced Computer Graphics Lab
http://graphics.ucmerced.edu/index.html

University of Iowa Virtual Soldier Research
http://www.ccad.uiowa.edu/vsr

University of Miami Biomechanics Research Lab
http://www.ie.miami.edu/brl

University of Michigan 3D Lab
http://um3d.dc.umich.edu/home.html

University of Washington Motion Capture Laboratory
http://grail.cs.washington.edu/mocap-lab

University of Wisconsin Computer Graphics
http://www.cs.wisc.edu/graphics

INDEX

Note: Page numbers followed by *f* indicate figures.