

Second Skin: Motion Capture with Actuated Feedback for Motor Learning

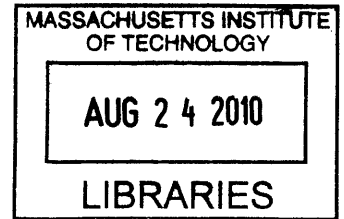
ARCHIVES

by

Dennis Miaw

B.S., Music, MIT., 2007

B.S., EECS, MIT., 2008



Submitted to the Department of Electrical Engineering and Computer
Science

in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2010

© Massachusetts Institute of Technology 2010. All rights reserved.

Author

Department of Electrical Engineering and Computer Science

February 2, 2010

Certified by

Ramesh Raskar

Associate Professor, MIT Media Lab

Thesis Supervisor

Accepted by

Dr. Christopher J. Terman

Chairman, Department Committee on Graduate Students

11/11/11

Second Skin: Motion Capture with Actuated Feedback for Motor Learning

by

Dennis Miaw

Submitted to the Department of Electrical Engineering and Computer Science
on February 2, 2010, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Second Skin aims to combine three-dimensional (3D) motion tracking with tactile feedback for the purpose of improving users' motor-learning ability. Such a system would track a user's body and limb movements as he or she performs an action, and then give the user automatic, real-time tactile feedback to aid in the correction of movement and position errors. This thesis details the development of a robust and low-cost optical 3D motion capture system along with versatile and flexible tactile feedback hardware. The vision is that these technologies will facilitate further research and the future development of motor-learning platforms that fully integrate 3D motion tracking and tactile feedback.

Thesis Supervisor: Ramesh Raskar

Title: Associate Professor, MIT Media Lab

Acknowledgements

Many people have contributed greatly to the success of this project. I would like to thank my advisor, Ramesh Raskar, for his guidance, support, and encouragement over the past two years.

In addition, I would like to thank Tyler Hutchison, Josh Wang, and Aoki Takafumi for their continued efforts towards the project, as well as Shinsaku Hiura, Ankit Mohan, Ahmed Kirmani, and Tom Cuypers for their expertise in fields unknown to me.

I would also like to thank Jeff Lieberman, Angela Chang, Jesse Gray and Matt Berlin from Personal Robots, Todd Farrell from Biomechatronics, Leah Buechley and Hannah Perner-Wilson from High-Low Tech, Rob Lindeman and Paulo de Barros from Worcester Polytechnic Institute, and the people from Roving Networks tech support for all of their assistance along the way.

Contents

1	Introduction	11
1.1	System Overview	11
1.2	Related Research	14
1.3	Applications	16
2	Motion Capture System	17
2.1	Motion Capture Hardware	17
2.1.1	Hardware Overview	17
2.1.2	Projectors	18
2.1.3	Photodetector Tag	23
2.1.4	Motion Capture Suit	26
2.2	Motion Capture Software	29
2.2.1	Software Overview	29
2.2.2	Projector Calibration	31
2.2.3	Three-Dimensional Point Computation	35
2.2.4	Projector Data Filtering	36
2.2.5	Least Squares Filtering	37
2.2.6	Bilateral/Median Smoothing Filter	40
2.2.7	Motion Capture GUI	41
2.3	Motion Capture Setup and Performance	42
3	Skeleton Modeling	45
3.1	Overview	45

3.2	Skeleton Calibration	46
3.3	Skeleton Generation	50
3.4	Skeleton GUI	51
4	Tactile Feedback System	53
4.1	Overview	53
4.2	Tactile Control Board	54
4.3	Tactors	57
5	Conclusion	59
5.1	Future Work	59
5.2	Summary	62
6	Bibliography	65

List of Figures

1-1	System overview with wireless communication. The computer computes the 3D coordinates of the photodetectors and sends commands to the tactile control board.	14
2-1	A single Second Skin infrared projector.	18
2-2	Creating the mask from a 3-bit Gray Code. A binary 0 corresponds to an opaque portion of the mask, while a binary 1 corresponds to a clear portion of the mask.	19
2-3	The complete mask slide used for the projectors.	19
2-4	Expanded view of a projector displaying its individual components. .	21
2-5	OpenGL visualization to check the accuracy of the projectors. Time is on the x -axis, and projector value is on the y -axis. This shows data from a projector with a properly aligned mask.	22
2-6	A photodetector tag board.	24
2-7	The bottom side of a photodetector tag board, showing the connector for the eighteen photodetectors.	24
2-8	A subject wearing the motion capture suit.	28
2-9	A closeup of the arm, showing the conductive thread and photodetectors. The thread is zig-zagged to allow it to stretch with the fabric. .	28
2-10	Backside of the motion capture suit, showing the tag boards.	29
2-11	A normal TSOP7000 photodetector (left), and one with the sensor lens covered in epoxy (right).	30
2-12	The 4'x4' calibration grid in its lowered position.	34

2-13	The 4'x4' calibration grid in its raised position.	34
2-14	Screenshot of the real-time motion capture GUI showing the 3D locations of the photodetectors on the motion capture suit.	41
2-15	A pair of projectors mounted as a single unit on a tripod.	43
3-1	Calibration tool used for the skeleton calibration.	49
3-2	Screenshot of the real-time motion capture skeleton along with the 3D photodetector positions. The coordinate system for each joint is also shown.	52
4-1	Tactile control board with tactors. TactaPack DC vibration motors on the left (white), and AEC voice coil actuators on the right (blue). . .	54

Chapter 1

Introduction

1.1 System Overview

When people develop new skills that require coordination, such as dance or sports, the process of learning the correct motions or the correct form is often time-consuming, and requires dedication and practice. Second Skin aims to enhance and quicken the process of motor-learning through the combined use of three-dimensional (3D) motion tracking and automatic, real-time tactile feedback. The ultimate goal is to use the motion tracking system to track a user's movements as he or she performs an action, analyze and compare the user's motion data to a reference dataset, such as the motions of an expert, and give the appropriate tactile feedback to the user to indicate how he or she should correct his or her motions and positions. While such a complete system is not yet fully realized, this thesis presents key technologies in motion capture and tactile feedback that have been developed to help achieve such goals.

The Second Skin 3D tracking system is an extension of the Prakash motion tracking system originally developed at Mitsubishi Electric Research Labs (MERL). This is an innovative design that uses infrared projectors to track the locations of infrared photodetectors, rather than using cameras to track passive white markers, which is typical with many of today's motion tracking systems [10]. This new type of tracking system inherently provides a number of promising qualities: it is unaffected by

changes in ambient lighting, each photodetector has a unique ID so there are no issues with marker swapping or reacquisition, and it is relatively low-cost since all the electronics are off-the-shelf components. Currently, the system is on the order of a few thousand dollars to manufacture. These are promising features because it enables motion capture to be used in a variety of environments that normally would not be able to support a motion capture setup. The system's robustness to lighting changes means it can be used outside of controlled studio environments and in more dynamic settings such as outdoors, in complete darkness, or on a stage with theatrical or unpredictable lighting. The low cost will make such systems available to consumers and businesses that otherwise would not be able to afford a motion capture setup.

For Second Skin, the basic Prakash technology was expanded and modified to support full three-dimensional tracking using multiple projectors. A motion capture suit with embedded photodetectors covering the left arm, upper body and torso was developed to facilitate motion capture of a human user. For minimal bulk, the majority of the electrical traces were sewn into the fabric using conductive thread. This makes for a lightweight and uncluttered design that handles very much like normal clothing. The photodetector tags were redesigned such that they could be integrated into fabric for use with the motion capture suit. The photodetectors receive 1D position data from the projectors, and the tag transmits this data to a computer wirelessly over Bluetooth, where the computer software calculates the 3D locations for all the photodetectors from the received 1D position data in real time.

In addition, for a user wearing the motion capture suit, the software also generates a real-time skeletal model of the user based on the photodetector positions on the motion capture suit. This model consists of the joint locations and orientations for the left wrist, left elbow, left shoulder, chest and torso. Such a model is often used for motion capture applications since it provides useful information about the user's motion and position. A simple OpenGL program, which is integrated with the rest of the software, renders the 3D positions of the photodetectors and the skeletal model

in a simple 3D scene. This makes it easy to monitor and visualize the resulting data.

The tactile feedback system consists of a tactile control board which controls sixteen tactile actuators, or tactors, simultaneously. Most of the tactile feedback research for Second Skin has focused around vibrotactile actuation, which creates the tactile stimulus by applying vibrations to the skin. The tactile control board is capable of driving both voice coil type actuators which require an oscillating input voltage, as well as direct current (DC) vibration motors which require a DC input voltage. Testing was performed with voice coil tactile actuators purchased from Audiological Engineering Corporation (AEC) [1], and TactaPack DC vibration motors borrowed from Robert Lindeman and his group at Worcester Polytechnic Institute [7]. These types of tactors are lightweight, small in size, and easy to use, which makes them suitable for this application as compared to other types of actuation, such as physical movement of limbs through large motors, or electrical stimulation of muscles. Producing physical movement requires bulky and heavy motor systems, and electrical stimulation is invasive and potentially painful [5].

While the low-level hardware control for the tactors is in place, there is not yet a system in place to analyze the motion capture data and determine the appropriate high-level control for the tactors so as to generate an effective stimulus for the user. In addition, the tactors and tactile control board also have not been integrated into the motion capture suit. These are all future goals for the continued development Second Skin. Currently the project establishes important developments in motion capture and tactile feedback to hopefully make such future goals not only possible, but also affordable and useable in any number of common, everyday environments.

The embedded software for the projectors, tags, and tactile control board was written in C using the MPLab 8 design environment and the CCS C compiler. The computer software was written in C++ using the Microsoft Visual Studio 2005 design environment. The circuit boards were designed in EAGLE.

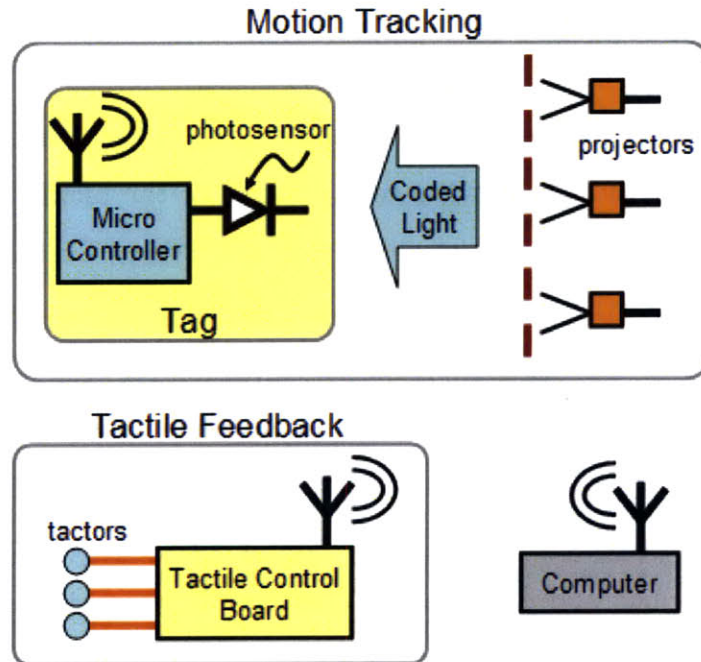


Figure 1-1: System overview with wireless communication. The computer computes the 3D coordinates of the photodetectors and sends commands to the tactile control board.

1.2 Related Research

It has previously been demonstrated that motion capture combined with vibrotactile feedback can be useful for patients undergoing physical therapy [7]. In experiments conducted by Robert Lindeman et. al. at Worcester Polytechnic Institute, motion sensing was performed using accelerometers, and the vibrotactile feedback was used to indicate to patients when they were in danger of injuring themselves by moving their joints outside of a safe range of motion [7].

Other similar research in motion capture and tactile feedback has also been conducted at the MIT Media Laboratory by Jeff Lieberman and Cynthia Breazeal using the Vi-con motion capture system and the same vibrotactile actuators developed by AEC

[5]. Here, it was demonstrated that precise control over the timing and intensity of the vibrations could indicate incorrect joint angles and rotations to a human user [5]. The tactile feedback in their experiments was used in conjunction with visual feedback from a computer screen in tests that asked participants to match the position of one of their arms to one displayed on the screen. Analysis of the tests showed that those who were aided both visually and tactilely demonstrated a decrease in motion errors of 21% over those who only had the visual aid [5].

These studies demonstrate the potential effectiveness of these types of systems, but present some issues as well. Accelerometers, while suitable for specific applications, are not ideal for many general purpose motion tracking applications due to their tendency to accumulate errors over time, and the lack of a defined global coordinate system makes it more difficult to determine where points are in relation to each other. Also, while Vicon is accurate and highly engineered, it is also very expensive and requires controlled scene lighting [9], limiting the number of scenarios in which it can be used. These are all issues that the Second Skin motion capture system inherently overcomes, which is an indication of its potential in the field of motion capture applications.

Previous research in wearable electronics also demonstrates the potential for clothing with embedded electronics. The Georgia Tech Wearable Motherboard was a project that integrated optical fibers, and various sensing techniques into a vest that could detect bullet wounds and monitor the user's body conditions such as body temperature or heart rate [15]. Leah Buechley at the MIT Media Lab has outlined and developed a broad array of techniques to incorporate basic electronic components such as LEDs and IC chips onto normal clothing, with more of an arts and crafts focus [13].

1.3 Applications

The Second Skin project presents a number of exciting future possibilities. The ability of the motion tracking system to function under any lighting conditions and its automatic marker IDs makes for a very robust system. This, combined with its low cost will make it useable in any number of different environments and facilities that normally would not be able to operate a motion capture setup. This could sustain an entirely new approach to sports training, dance practice, or healthcare applications. Gyms, dance schools, or other training centers could acquire these systems for their clients and customers to use. Hospitals could use them to aid with patient rehabilitation or with elderly individuals who might have difficulty performing daily activities. Such a system could even be installed in a home for personal use.

Chapter 2

Motion Capture System

2.1 Motion Capture Hardware

2.1.1 Hardware Overview

The motion tracking hardware is comprised of two components: infrared projectors and infrared photodetector tags. The projectors emit a sequence of structured light Gray code patterns which effectively codes the space into which they are projecting. Multiple projectors are mounted such that they surround and project into the tracking volume. The points to be tracked are very small Vishay TSOP7000 infrared photodetectors, and a single tag board supports up to eighteen of these photodetectors. A photodetector placed in the scene receives a unique light pattern from all of its visible projectors. For each projector, the received light pattern is then decoded into an 8-bit binary number which corresponds to the photodetector's 1D position relative to the location and orientation of that projector. The tag stores this set of 1D position data for every photodetector, and then wirelessly transmits this data to a computer via Bluetooth. The software on the computer calculates the 3D coordinates of each photodetector from this data.

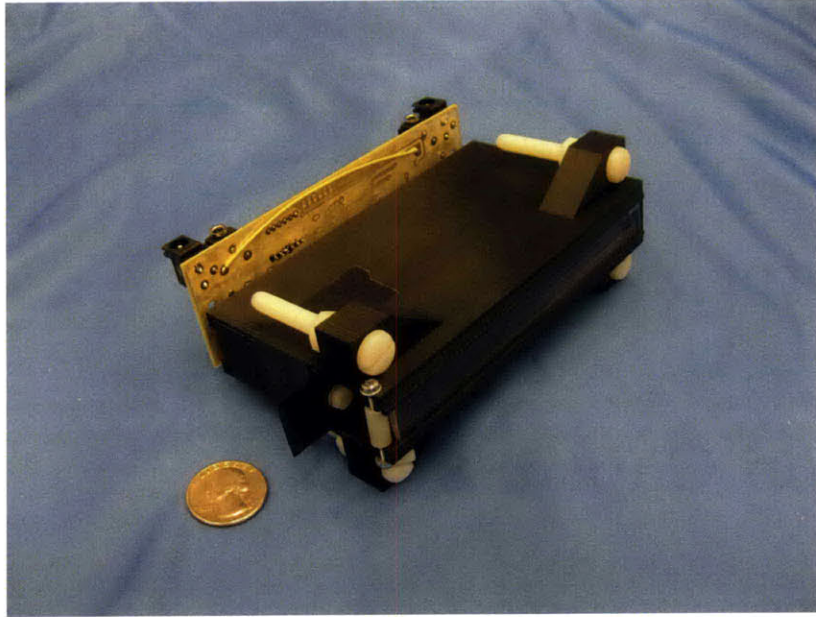


Figure 2-1: A single Second Skin infrared projector.

2.1.2 Projectors

The projectors are controlled by a PIC16F876 microcontroller. Each projector emits a sequence of eight frames. Each frame consists of a pattern of infrared stripes that corresponds to one of the bits of an 8-bit Gray code. This technique is sometimes referred to as structured light. The patterns are emitted using eight near-infrared LEDs placed side-by-side in a single row. The LEDs emit at a wavelength of 870 nm. In front of each LED is a transparency mask with opaque and clear stripes such that when the LED is turned on, the mask pattern is emitted into the scene. The correct mask is placed in front of each of the eight LEDs such that when the LEDs are turned on in succession, the Gray code structured light sequence is projected into the scene. See Figure 2-2 for an example of the mask pattern using a three-bit Gray code. For each frame, the corresponding LED is on for $33 \mu\text{s}$, and then it is off for $33 \mu\text{s}$, for a total time of $66 \mu\text{s}$. Each frame follows this same sequence, so all eight frames are complete in $528 \mu\text{s}$.

Additionally, a synchronization signal or sync bit is projected prior to the eight-bit

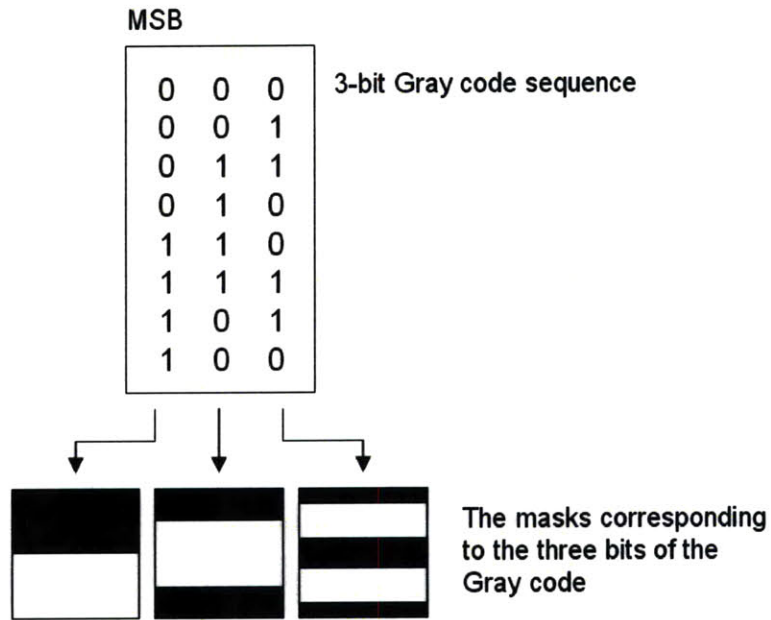


Figure 2-2: Creating the mask from a 3-bit Gray Code. A binary 0 corresponds to an opaque portion of the mask, while a binary 1 corresponds to a clear portion of the mask.

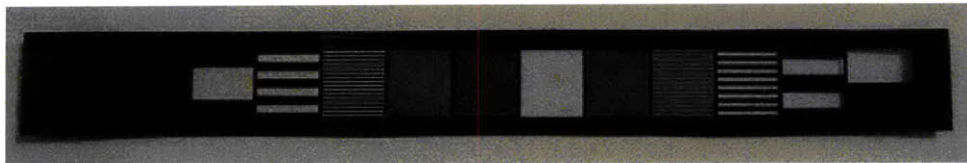


Figure 2-3: The complete mask slide used for the projectors.

Gray code sequence. This sync bit is on for $100\ \mu\text{s}$, then off for $33\ \mu\text{s}$, and is projected by an LED with a completely clear mask, so that any photodetector within the projector's field of view will receive it. The purpose of the sync bit is to signal the start of the projection sequence to the photodetector tag. If a photodetector receives a signal that is at least $90\ \mu\text{s}$ long, then it knows that that is the sync bit, since all the data bits are on only for $33\ \mu\text{s}$. The sync bit plus the Gray code sequence brings the total time to run one projector up to $661\ \mu\text{s}$. However, dead time is intentionally added after the LEDs have completed emitting, to bring the total time to run one projector to $1000\ \mu\text{s}$. This extra time is to allow for the photodetector tag to complete its data processing and wireless data transmission to a computer via Bluetooth.

Another key aspect of the LED projection is that all the LEDs are modulated at $455\ \text{kHz}$. This means that whenever an LED is on, it is actually flashing at a frequency of $455\ \text{kHz}$ and with a duty cycle of 50% . The photodetectors are only sensitive to light modulated at this carrier frequency, and this helps filter out ambient infrared sources such as light bulbs or sunlight that might otherwise interfere with the system. Each Gray code frame consists of fifteen pulses at $455\ \text{kHz}$, which translates to $33\ \mu\text{s}$.

When running multiple projectors, as would be necessary to perform 3D tracking and/or to cover a large area and field of view, the projectors are time-multiplexed such that only one projector is projecting its Gray code pattern at a time. The sync bit, however, is projected by all of the projectors simultaneously at the beginning of the entire cycle. To facilitate this functionality, the first projector in the chain operates in Master Mode, while all other projectors operate in Slave Mode. To start the cycle, the master projector transmits a wired signal to all of the slave projectors to indicate to them to project the sync bit. The master also projects its sync bit at this time, such that all the projectors emit the sync bit simultaneously. After the sync bit, the master then transmits its Gray code sequence. Then, the master sends a wired signal to the next slave in the chain to indicate to that projector to transmit its Gray code pattern. After the slave projector finishes projecting, it sends

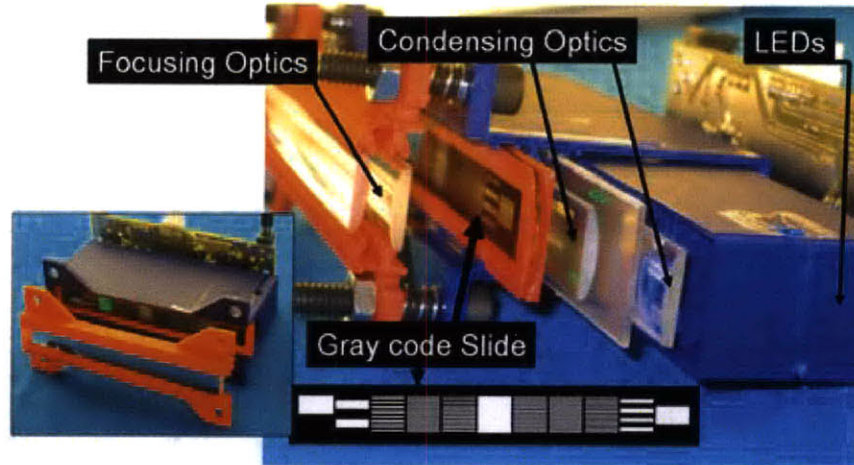


Figure 2-4: Expanded view of a projector displaying its individual components.

a wired signal to the next slave in the chain, and the process repeats for all slaves. The master does not receive any signal from the last slave to indicate that the cycle is finished. Rather, it simply waits $1000 \mu\text{s}$ for each slave, and then begins the next cycle automatically by signaling the sync bit again. A rotary selector switch on the master circuit board is used to set the number of slaves.

The timing is designed such that each projector, regardless of whether it is a master or slave, completes in $1000 \mu\text{s}$. Thus, for slave projectors, there is more dead time since they do not have to transmit their own individual sync bit. There is only one sync bit which occurs as part of the master projector's $1000 \mu\text{s}$ time slot. Because the projectors are time-multiplexed, this unfortunately means that the more projectors in the system, the lower the overall frame rate, since all the projectors must complete before starting the next frame.

The casings for the projectors were 3D printed on a Stratasys Dimension Elite printer, using black ABS Plus material. Black-colored material was chosen for its opaqueness to visible and near-IR light. The optics consists of four cylindrical glass lenses and a plastic lenticular lens sheet used as a 1D diffuser. Most of the parts are designed such that they fit together with the proper alignment and placement. However, the

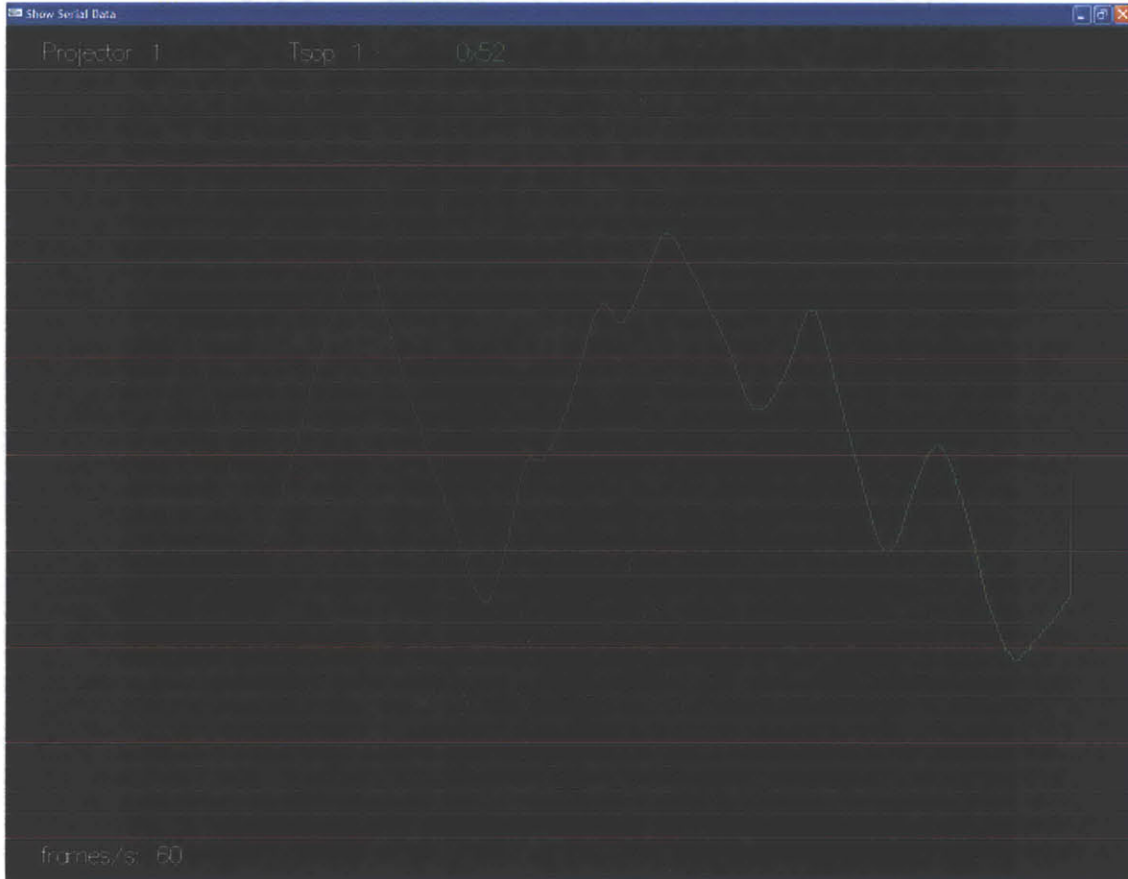


Figure 2-5: OpenGL visualization to check the accuracy of the projectors. Time is on the x -axis, and projector value is on the y -axis. This shows data from a projector with a properly aligned mask.

one component that must be manually aligned is the Gray code mask. Currently, multiple masks are printed on a transparency sheet and individually cut out by hand. The individual transparency mask is then fixed into a 3D-printed mask holder using nylon machine screws. This mask holder then slides into the projector casing. If the mask pattern is not exactly or very close to parallel to the lenses, then the projected patterns will be out of focus, resulting poor tracking accuracy. The alignment of the mask can be adjusted by loosening the machine screws that fix it to the mask holder, adjusting the mask, and then re-tightening the machine screws. To aid in obtaining the proper mask alignment, an OpenGL visualization program is used which displays the projector value for a photodetector over time. A properly aligned mask results in smooth data as the photodetector is moved across the projector's range, while a misaligned mask results in jumpy and unreliable data.

2.1.3 Photodetector Tag

The key components in the photodetector tag consist of a PIC18F25K20 microcontroller, a Roving Networks RN-41 Bluetooth transceiver, and connections for eighteen Vishay TSOP7000 photodetectors. Each photodetector contains a built-in automatic gain control circuit, bandpass filter, and demodulator. This makes for a very convenient package, as this circuitry provides a digital output corresponding to whether or not the photodetector is receiving light modulated at a 455 kHz carrier frequency. This carrier frequency is set by the manufacturer and is not adjustable. The photodetector outputs a digital high on its output pin if it is not receiving any light signal and it outputs a digital low if it is receiving a light signal. It must receive at least ten pulses at 455 kHz before it will output a digital low. The photodetectors are most sensitive to light with wavelengths between 850 and 900 nm. The tag board can be powered from any voltage supply between six and fifteen volts, including common 9V batteries. Onboard voltage regulators maintain the proper voltages required for the circuit components.

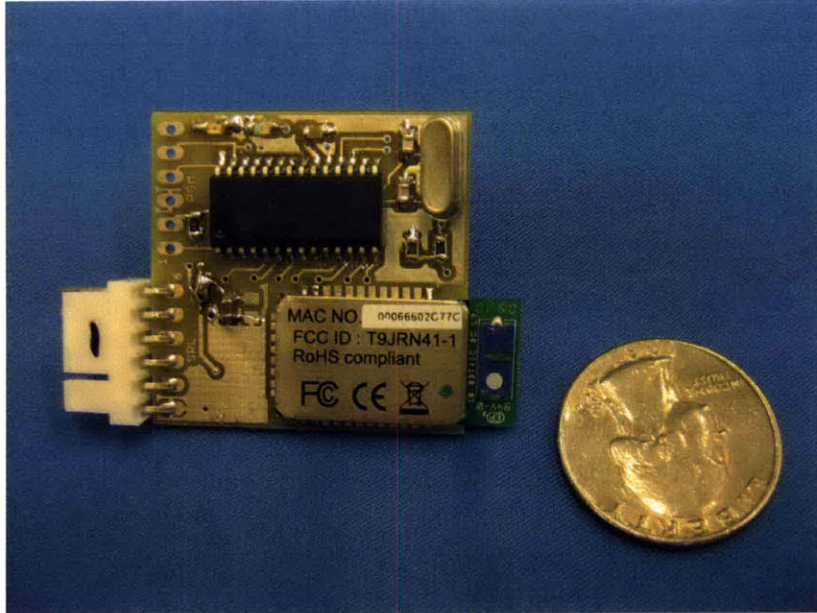


Figure 2-6: A photodetector tag board.

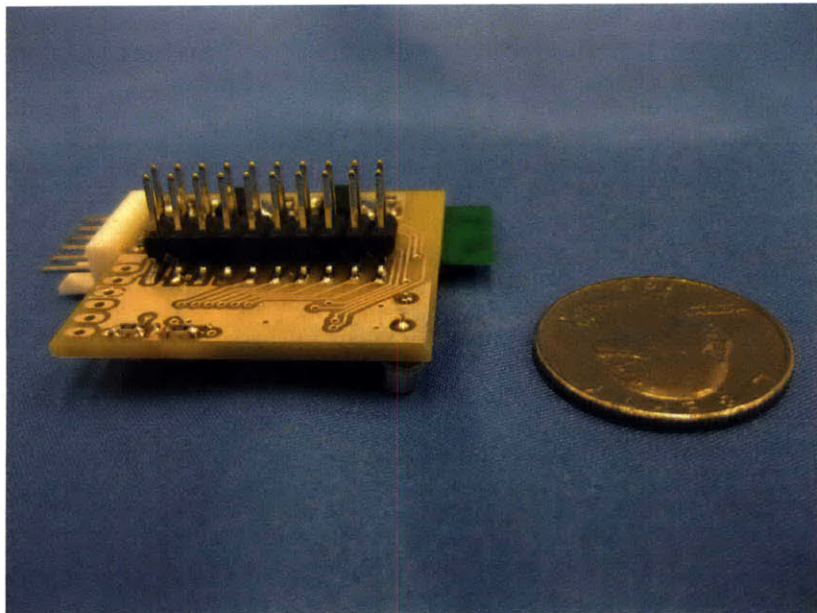


Figure 2-7: The bottom side of a photodetector tag board, showing the connector for the eighteen photodetectors.

If a photodetector is placed in the scene with a projector, then the photodetector will receive a unique light sequence from the projector which corresponds to the photodetector's location relative to the projector. In the 3-bit example described in Figure 2-2, if the three masks were flashed from left to right, a received pattern of *dark-dark-dark* would indicate that the tag is at the very top of the projector's field of view, whereas a received pattern of *dark-dark-light* would indicate that the tag is slightly lower. The next pattern down would be *dark-light-light*, and so on. These received sequences are simply Gray code numbers which correspond to the plane in which the photodetector is in relative to the position and orientation of the projector. Thus, for any single projector, a photodetector resolves its location in one dimension perpendicular to the direction of the stripes. Motion parallel to the stripes will not change the light sequence that the photodetector receives. In order to resolve location in higher dimensions, multiple projectors must be used. Having a second projector oriented perpendicular to the first will allow the resolution of two dimensions. A minimum of three projectors are needed to resolve all three dimensions.

The microcontroller first reads the photodetector outputs until at least one of them receives a sync bit of at least 90 μ s. After the sync bit finishes, the microcontroller is programmed to read the photodetectors at the correct times, corresponding to when the projectors are emitting each of their Gray code frames. This timing is handled by an onboard timer in the microcontroller. No synchronization is done between the projectors and tags other than the initial sync bit. The number of projectors currently must be hardcoded into the microcontroller code. A change to the number of projectors requires reprogramming the microcontroller.

The microcontroller reads in all the eight-bit Gray code values for each projector for each photodetector. These values are converted to binary using a look-up table, and then the binary data is sent to the RN-41 Bluetooth transceiver for wireless transmission to a computer. A start byte of 0xFF (255 in decimal) is transmitted prior to each set of data to indicate the beginning of the set. One set of data refers to all

the projector values for all the photodetectors for one cycle of all the projectors. In other words, it is all the data received in between successive sync bits. When using multiple photodetectors with a single tag, valid data is sent only for photodetectors that successfully received the sync bit. For ones that did not receive a sync bit because they were occluded or were otherwise not visible, zeros are sent for all of their projector values. Since a value of 255 is used for the start byte, and a value of zero is used to indicate invalid data, the range of values corresponding to valid data is from 1-254.

In order to send data to the computer, the RN-41 Bluetooth transceiver on the tag board wirelessly connects to a Roving Networks RN-USB-X transceiver on the computer side. The RN-USB-X is a Bluetooth transceiver that plugs into a USB port on the computer and appears as a serial COM port to computer applications. The RN-41 on each tag is configured to automatically connect to a specific RN-USB-X on power-up. When using multiple tags, the RN-41 transceiver on each tag connects to its own RN-USB-X transceiver.

2.1.4 Motion Capture Suit

To facilitate the use of motion capture with a human user, a motion capture body suit was developed with the required electronics integrated into the suit. The suit is an Under Armor Heat Gear garment, modified to contain all the required electronics. There are 26 photodetectors placed around the left arm, chest, back and torso, and two tag boards located at the left side of the lower back. The tag boards can be plugged and unplugged from the suit via connectors that are sewn into the fabric. The Under Armor garment is reasonably skintight, such that the photodetectors undergo minimal shifting relative to the wearer's skin during typical use. This is important for creating skeleton fits from the photodetector locations that remain accurate as the user is moving.

The photodetectors are sewn in place, with six photodetectors located around the wrist, four located on the forearm, six located on the upper arm, six on the back, and four on the chest and stomach. The majority of the electronic traces are sewn into the fabric using a 4-ply conductive thread purchased from SparkFun Electronics. The thread has a resistance of about 12 ohms per foot, which is significantly higher than typical wire, but is perfectly suitable for the digital data lines between the photodetectors and the microcontroller on the tag. However, this resistance is unfortunately too high for use as the power and ground traces. The high resistance plus the large number of photodetectors connected to the power and ground lines results in too much voltage drop across the thread, and the photodetectors are unable to receive the proper operating voltage. In general, it is important that the photodetectors receive close to five volts, if not slightly more. As the supply voltage decreases, their light sensitivity decreases which reduces their viewing angle. However, when the thread was tested, the voltage ended up so low that most of the photodetectors were unable to function at all. So, 30AWG wire is used instead for the power and ground lines. Because the conductive thread is not insulated, care must be taken to avoid accidental shorts between the traces. Aleene's Flexible Stretchable Fabric Glue is a fabric adhesive that remains flexible and elastic when dry. To insulate the conductive thread, this glue was applied to areas where there was a high risk of accidental shorts.

A small surface-mount $4.7 \mu\text{F}$ capacitor is soldered in between the power and ground pins on each photodetector. This helps to eliminate fluctuations and noise on the power wires which otherwise might affect the performance of the photodetectors. Additionally, the sensor lens of each photodetector is covered with Devcon 5-minute Epoxy. The epoxy acts as a lens and/or diffuser which actually increases the viewing angle of the photodetectors. Several adhesives were tested, including Instant Crazy Glue, Aleene's Flexible Stretchable Fabric Glue, Elmer's All-Purpose White Glue, and E6000 Industrial Adhesive. The two that provided the best performance were the Devcon 5-minute Epoxy and Aleene's Fabric Glue. The epoxy was ultimately chosen for its rapid curing time of 5-15 minutes, ease of application to the photode-



Figure 2-8: A subject wearing the motion capture suit.



Figure 2-9: A closeup of the arm, showing the conductive thread and photodetectors. The thread is zig-zagged to allow it to stretch with the fabric.



Figure 2-10: Backside of the motion capture suit, showing the tag boards.

tectors, and its hard and smooth surface finish. The fabric glue, while offering a similar performance, requires a 24-hour dry time, has a viscosity which makes it more difficult to apply uniformly, and remains slightly tacky after fully dried which gives it an undesirable tendency to pick up dust and particles.

2.2 Motion Capture Software

2.2.1 Software Overview

The primary role of the motion tracking software is to compute the 3D positions for all the photodetectors from the raw 1D projector data. To compute the 3D points, calibration data must first be obtained for each projector. There are seven parameters that must be computed for each projector. These parameters are used to map a 1D projector value to a plane in 3D space. Then, if a photodetector receives three or more projector values, a least squares solution for the intersection of those planes can be solved. That point of intersection is the 3D location of the photodetector. While,

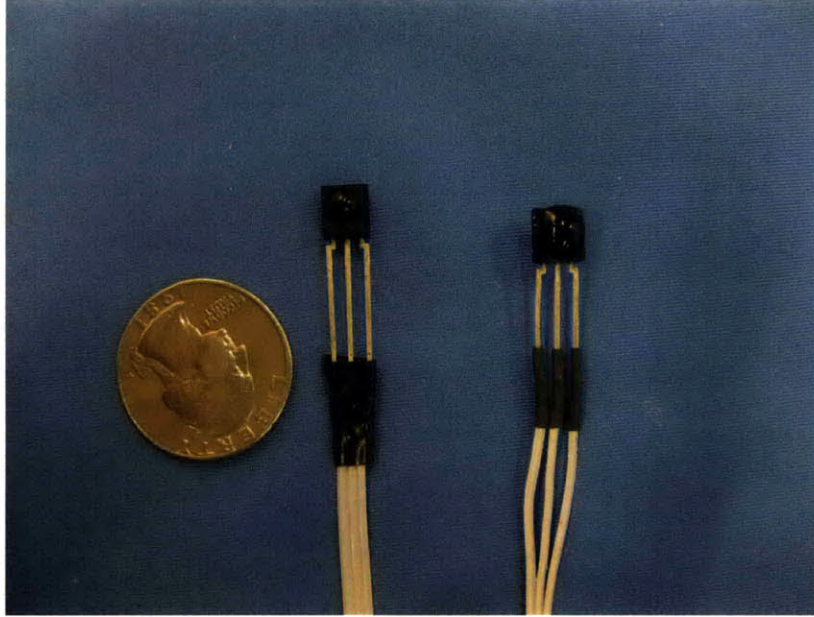


Figure 2-11: A normal TSOP7000 photodetector (left), and one with the sensor lens covered in epoxy (right).

mathematically, only three planes are needed to compute a point of intersection, the software requires at least four planes (four projector values) so as to improve accuracy.

One of the challenges in performing the 3D point computation is in eliminating noisy data from the projectors. If a photodetector is pointing towards a particular projector, then it will get accurate data; if a photodetector is pointing away from a projector, then it will receive no data; but if the photodetector is at an angle such that a projector is just barely in the photodetectors field of view, then often the photodetector will not receive all of the Gray code bits, which results in highly incorrect and erratic position readings. If these incorrect values are incorporated into the 3D point calculation, then the resulting locations will be inaccurate. For this reason, much of the software is dedicated towards detecting and ignoring invalid projector data. The projector data filtering and least squares filtering stages are designed to remove invalid data from the 3D computation, while the point smoothing filter is used simply to refine and smooth the resulting 3D data.

Additionally, for convenience, a simple OpenGL visualization renders the computed positions of the photodetectors in 3D space.

2.2.2 Projector Calibration

In order to compute a 3D point from multiple 1D projector values, the software needs information that relates the projector values to the 3D space. Essentially, a known projector value will constrain the location of the photodetector to a plane which passes through the center of the projector and is parallel to the direction of the stripes of the Gray code patterns. If the photodetector moves anywhere along this plane, it will receive the same projector value.

The derivation for determining the parameters for each projector begins with the following matrix equation which maps a 3D coordinate to a 1D projector value. This is analogous to the more common camera matrix equation which maps a 3D coordinate to a 2D camera sensor.

$$hV = PX$$

$$h \begin{bmatrix} v \\ 1 \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

h is a scalar representing the distance of the photodetector to the projector.

V is the projector value received by the photodetector.

P is the matrix containing the seven unknown projector parameters.

X contains the 3D coordinates of the photodetector.

The goal is to find P , given V and X for multiple photodetectors. First, the matrices

$hV = PX$ are multiplied out:

$$\begin{aligned}P_{11}x + P_{12}y + P_{13}z + P_{14} &= hv \\P_{21}x + P_{22}y + P_{23}z + 1 &= h\end{aligned}$$

From these two equations, h is eliminated:

$$P_{11}x + P_{12}y + P_{13}z + P_{14} = P_{21}xv + P_{22}yv + P_{23}zv + v$$

This is rewritten as:

$$P_{11}x + P_{12}y + P_{13}z + P_{14} - P_{21}xv - P_{22}yv - P_{23}zv = v$$

In matrix form this becomes:

$$\begin{bmatrix} x & y & z & 1 & -xv & -yv & -zv \end{bmatrix} \begin{bmatrix} P_{11} \\ P_{12} \\ P_{13} \\ P_{14} \\ P_{21} \\ P_{22} \\ P_{23} \end{bmatrix} = v$$

If there are many photodetector calibration points (at least 7), P can be solved using

standard least squares methods:

$$\begin{bmatrix} x_1 & y_1 & z_1 & 1 & -x_1v_1 & -y_1v_1 & -z_1v_1 \\ x_2 & y_2 & z_2 & 1 & -x_2v_2 & -y_2v_2 & -z_2v_2 \\ x_3 & y_3 & z_3 & 1 & -x_3v_3 & -y_3v_3 & -z_3v_3 \\ & & & & etc. & & \end{bmatrix} \begin{bmatrix} P_{11} \\ P_{12} \\ P_{13} \\ P_{14} \\ P_{21} \\ P_{22} \\ P_{23} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ etc. \end{bmatrix}$$

To perform the calibration process, a large square grid is used, approximately 4x4 feet, with twelve photodetectors placed at precisely spaced locations along the grid. Each photodetector is assigned a specific coordinate in 3D space. The units for the coordinates can be arbitrary; but units of 0.1 inch were chosen for convenience. So, for example, two photodetectors located at (10,0,0) and (20,0,0) would be one inch apart from each other in the x -direction. The grid lies along to the x - z plane, which is parallel to the floor. The y -axis points upwards. This grid is placed in the center of the motion capture space, surrounded by the projectors. Because the photodetectors have a limited field of view, they are mounted on posts that can rotate 360 degrees. This allows a photodetector to be rotated so that it can receive data from any surrounding projector without altering its position in 3D space. With this setup, the projector values V for all the projectors are obtained for all the assigned photodetector locations X .

After the projector data has been obtained, the grid is then raised 36 inches (360 units in the y -direction), by placing it on a specially designed base and tower. This puts all of the photodetectors into new 3D locations that are 360 units higher in the y -direction. Projector readings are then obtained for all of these locations. So in the end, V and X data is obtained for every projector for many locations within the 3D space. From here the required matrices are created, and P is solved for every projector. After all the parameters are computed, they are written to a text file so

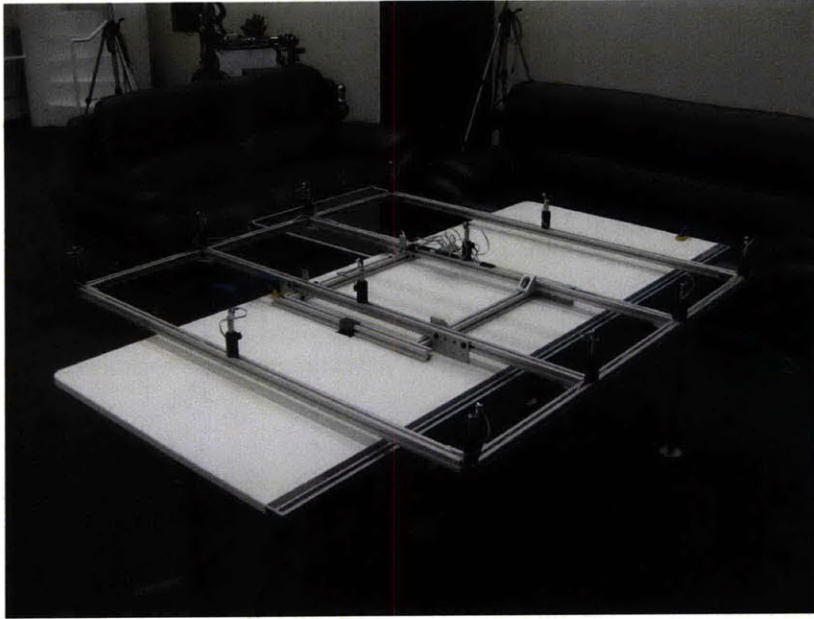


Figure 2-12: The 4'x4' calibration grid in its lowered position.

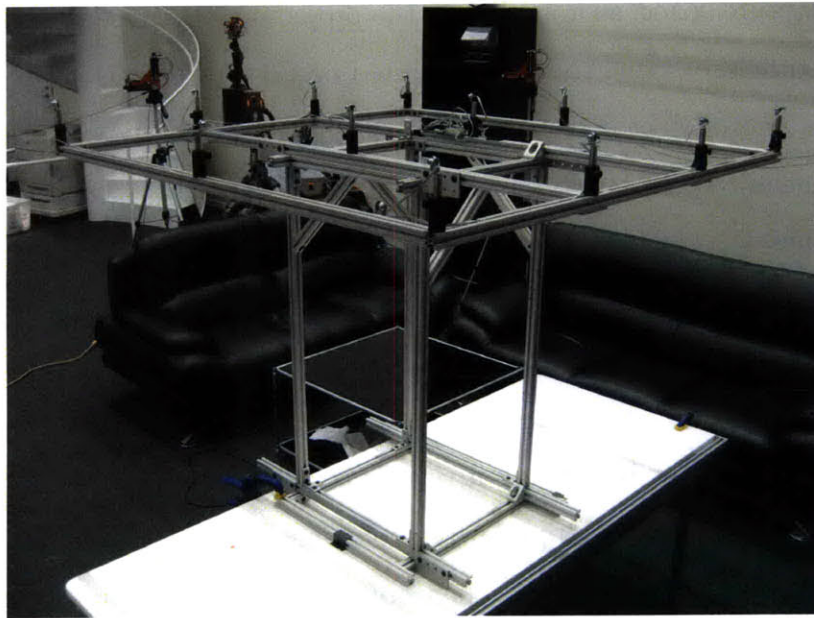


Figure 2-13: The 4'x4' calibration grid in its raised position.

that they can be easily loaded each time the software runs.

The calibration grid is built from 1"x1" 80/20 aluminum extrusion. This allows for easy construction and adjustment of the grid, and it is relatively lightweight. While a 4x4 foot square grid is somewhat cumbersome to manipulate and transport due to its size, larger grids generally result in more accurate calibration data. Smaller calibration objects were also used during the course of the project, including a 2x2 foot grid, and a 30x30 cm grid. Both of those provided considerably worse 3D point accuracies.

2.2.3 Three-Dimensional Point Computation

After the seven parameters are calculated for each projector, they are used to compute a 3D point for any photodetector that receives data from at least three projectors. Each projector value constrains the location of the photodetector to a plane in 3D space, and the least squares solution is obtained for the intersection of three or more planes.

Starting again from the 1D projector matrix equation described in section 2.2.2 Projector Calibration:

$$hV = PX$$

$$h \begin{bmatrix} v \\ 1 \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Again, as in section 2.2.2 Projector Calibration, this is multiplied out and rewritten as:

$$P_{11}x + P_{12}y + P_{13}z + P_{14} - P_{21}xv - P_{22}yv - P_{23}zv = v$$

This is rewritten as the equation of a plane, $Ax + By + Cz = D$:

$$(P_{11} - P_{21}v)x + (P_{12} - P_{22}v)y + (P_{13} - P_{23}v)z = v - P_{14}$$

In matrix form:

$$\begin{bmatrix} P_{11} - P_{21}v & P_{12} - P_{22}v & P_{13} - P_{23}v \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = v - P_{14}$$

Given three or more planes (three or more projector values v), and knowing P for all the projectors, the location of the photodetector is then solved using least squares methods. The system of equations is written in the form $Ax = b$:

$$\begin{bmatrix} P_{111} - P_{121}v_1 & P_{112} - P_{122}v_1 & P_{113} - P_{123}v_1 \\ P_{211} - P_{221}v_2 & P_{212} - P_{222}v_2 & P_{213} - P_{223}v_2 \\ P_{311} - P_{321}v_3 & P_{312} - P_{322}v_3 & P_{313} - P_{323}v_3 \\ P_{411} - P_{421}v_4 & P_{412} - P_{422}v_4 & P_{413} - P_{423}v_4 \\ \text{etc.} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} v_1 - P_{114} \\ v_2 - P_{214} \\ v_3 - P_{314} \\ v_4 - P_{414} \\ \text{etc.} \end{bmatrix}$$

2.2.4 Projector Data Filtering

The projector data filtering stage attempts to detect invalid data by looking at the recent history of received values from a projector. If the history of values is highly discontinuous, then that projector is likely providing inaccurate data for that particular photodetector, and its current value is ignored for the 3D point computation. If the history is smoothly changing, then this projector is likely providing valid data, and its current value is passed on to be used for the 3D point computation.

A history of the past m values is kept for every projector for every photodetector. Each time a new set of projector data is acquired, the new values are loaded into their appropriate histories, and the oldest values are removed. For each history, each

entry in the history is checked against its previous entry, keeping track of the number of times that the difference between two successive values is larger than n . If there are more than p of these discontinuities in the history, then it is concluded that this photodetector is not receiving consistent data from this particular projector, and the projector data is not used for the 3D point computation of this photodetector. If there are fewer than p discontinuities in the history, then the current value gets passed on to be used for the 3D point calculation. The data is also put through a small median filter to eliminate the occasional stray outlier for a projector that is otherwise providing good data. The histories contain the original projector values and not the median-filtered values.

The values for the history size m , the jump value limit n , and the jump count limit p are 15, 6, and 3, respectively. These were chosen experimentally by evaluating the effectiveness of the filter as these parameters changed.

This technique is computationally simple and is able to remove the majority of noise-related issues. However, bad projector values do occasionally make it through. The least squares filtering section discusses the next filtering stage which employs a different technique to attempt to remove any remaining inaccuracies from the 3D point computation.

2.2.5 Least Squares Filtering

A least squares computation is performed on the matrix equation described at the end of section 2.2.3 Three-Dimensional Point Computation in order to determine a 3D location for a photodetector given a set of projector data. However, a standard least squares solver will simply return the best fit solution to all the input equations. In some cases, the best solution may actually be quite far away from the correct solution if some of the input data is extremely noisy or inaccurate.

The *filtered least squares algorithm* returns a solution only if one is computed with a variance of less than a specified threshold value. For the matrix equation $Ax = b$, the algorithm solves for the best fit x' given the matrix A and the column vector b . First, the least squares solution x' is found by solving the normal equations:

$$x' = (A^T A)^{-1} A^T b$$

The error column vector is computed:

$$err = Ax' - b$$

The variance is computed by taking the square root of the sum of the squares of all the entries in the error column vector. If this variance is smaller than the specified threshold value, then x' is returned as the answer. However, if the variance is larger than the threshold value, then there is likely to be one or more rows in A and b that were generated from noisy outlier data and are causing the large variance. The algorithm now tries to discover and remove those rows. New A' and b' matrices are created by removing single rows from A and b , and the new solution and variance are computed. The algorithm checks if removing one or possibly more rows brings the variance to below the threshold. For example, if the original $Ax = b$ equation is as follows:

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \\ A_{41} & A_{42} & A_{43} \\ A_{51} & A_{52} & A_{53} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

The algorithm removes the first row of A and b and computes the new least squares

solution and variance:

$$\begin{bmatrix} A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \\ A_{41} & A_{42} & A_{43} \\ A_{51} & A_{52} & A_{53} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

Then the algorithm removes the second row and computes the new solution and variance:

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{31} & A_{32} & A_{33} \\ A_{41} & A_{42} & A_{43} \\ A_{51} & A_{52} & A_{53} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} b_1 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix}$$

Then the third row is removed, and so on, for all of the rows in A and b . The algorithm saves the solution that produced the smallest variance, and if this new variance is below the threshold then the corresponding solution is returned. If the smallest variance is still larger than the threshold, then the row-removal process is performed again using the A' and b' matrices that produced the smallest variance. This is repeated until a solution with a variance below the threshold is found or until A' and b' no longer have enough rows to continue the process. If A or A' is a square matrix, then the error and variance will always be zero, regardless of whether or not the rows contain accurate data, so the algorithm requires A and A' to always contain at least one more row than column. If no solution is found before A' is reduced to a square matrix, then the coordinates (9999,9999,9999) are returned, which indicates that there is not enough reliable data from which to compute a 3D location for this photodetector.

A suitable value for the variance threshold was determined experimentally by examining how stable the 3d points were as the threshold was changed, and by examining the variances produced under controlled test conditions. Variances that were produced

when it was known that all the received projector data was valid were compared to variances that were produced when it was known that noisy outlier data was being received. If the threshold is too large, then a noisy outlier may incorrectly be used to compute the solution. If the threshold is too small, then valid data may incorrectly be thrown out, reducing the accuracy and stability of the solution. For this particular system, the variances produced when all the projector data is valid are in the single digits or less, whereas the variances produced when there are one or two outliers are typically in the thousands. So, there is a fairly clear distinction and thus a decently large range of acceptable values that can be used for the variance threshold. Values between 10 and 100 were found to produce good results.

This algorithm, while not one hundred percent fail proof, improves the stability and accuracy of the resulting 3D data quite noticeably over using just the *projector data filtering* technique described in the section 2.2.4 Projector Data Filtering. In fact, it is possible to use this algorithm completely on its own without the initial *projector data filtering* stage and still get accurate results. However, this algorithm is much more computationally intensive. Therefore, it is more efficient to first use the *projector data filtering* to remove the majority of the invalid projector data, and then to use this algorithm to remove any remaining outliers.

2.2.6 Bilateral/Median Smoothing Filter

After the 3D points are computed, they are then put through small bilateral and median filters to refine and smooth the 3D point data. Each x , y , and z value for each point has its own filter. Every time a new set of 3D points are computed, the x , y , and z values are pushed into their respective filters along with a timestamp, and the oldest values are removed. The bilateral portion of the filter replaces each entry in the filter with a weighted average of its neighboring elements in both position and in time. In other words, an entry in the filter will be most affected by other entries that are far away in value and close by in time. An entry will be least affected by other

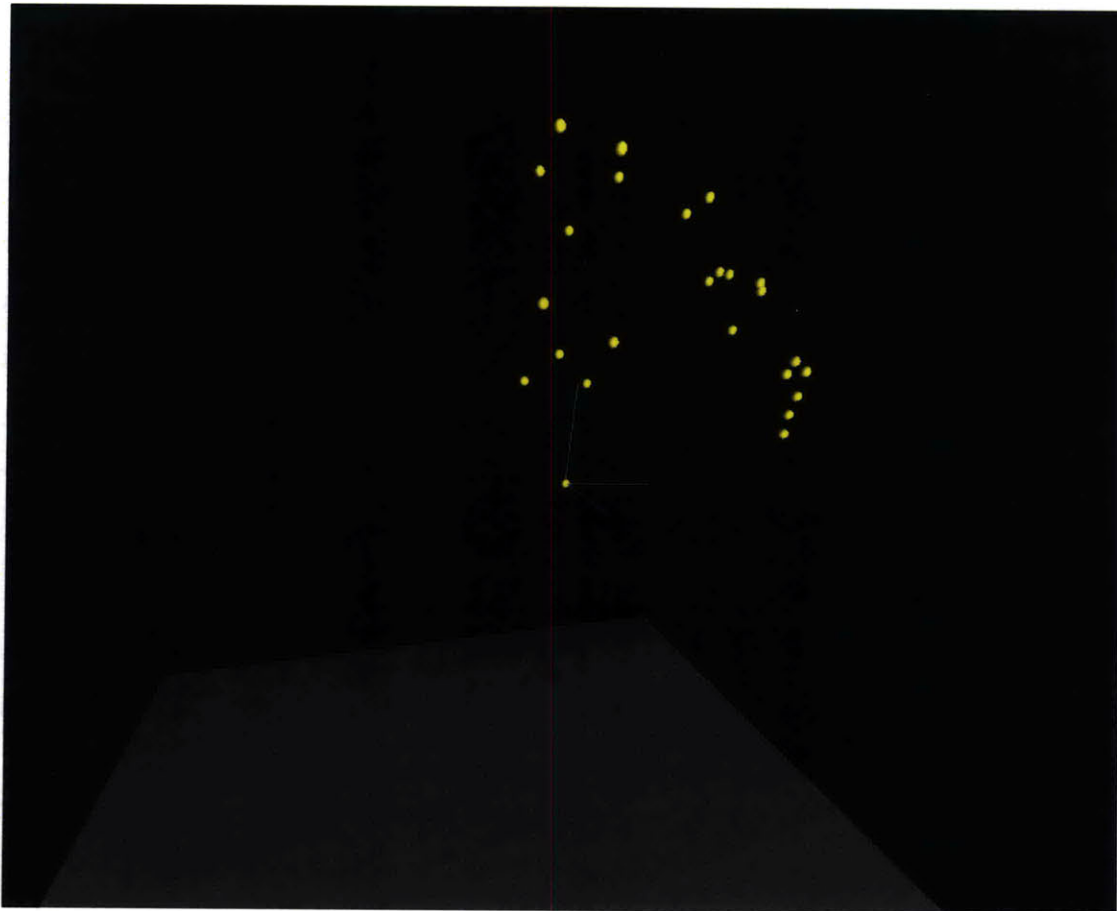


Figure 2-14: Screenshot of the real-time motion capture GUI showing the 3D locations of the photodetectors on the motion capture suit.

entries that are close in value and far away in time. The median of these smoothed values is then taken as the output of the filter.

2.2.7 Motion Capture GUI

A simple graphical user interface (GUI) is implemented in OpenGL in order to provide an easy way to visualize and monitor the 3D point data. A simple scene is rendered with a gray floor, coordinate axes to show orientation, and basic lighting. Each 3D point is rendered as a yellow sphere. The arrow keys on the keyboard are used to adjust the viewing position.

2.3 Motion Capture Setup and Performance

The current motion capture setup consists of sixteen projectors arranged in circle roughly 22 feet in diameter, all facing in towards the center. The projectors are mounted in pairs, with each projector in the pair oriented perpendicular to the other. The useable tracking space is approximately a 6 foot tall cylinder with a 6.5 foot diameter. Of course, the tracking volume can be adjusted by altering the projector placement. The useable range of distance for a single projector is between approximately ten to thirty-five feet, with a viewing angle of about 30 degrees in both the horizontal and vertical directions. The frame rate with sixteen projectors is 62.5 Hz, and the system demonstrates 3D accuracy to about one centimeter. The total latency is around 85 ms, which comes mostly from the Bluetooth transmission (approximately 15 ms), and the median filters which require future data before they generate a valid output (approximately 64 ms). The computer used to run the software features a Pentium D 2.80 GHz processor with 2 GB of RAM. With this machine and projector setup, the system successfully tracked 72 points in real-time (four tags with eighteen photodetectors per tag). However, this was not the upper bound, and it is estimated that real-time tracking of 100 points or more is feasible.



Figure 2-15: A pair of projectors mounted as a single unit on a tripod.

Chapter 3

Skeleton Modeling

3.1 Overview

For most motion tracking applications involving a human user, the locations of the photodetectors or markers are usually not of great interest. More important are the locations and orientations of the user's joints. Since different users have different body structures which affect the position of the markers relative to their actual bones and joints, a skeletal model is useful because it abstracts away the raw motion tracking data and provides more relevant joint information. This makes it easier for applications to determine properties of interest such as joint angles and body and limb positions. The skeletal models for Second Skin currently consist of joints for the left wrist, left elbow, left shoulder, upper body and lower body.

To generate a skeletal model for a user, the calibration phase first obtains information that relates the positions of the photodetectors to the positions and orientations of the joints. Then, the generation phase computes the locations and orientations of the joints in real-time, given the photodetector locations and the relational information obtained in the calibration.

3.2 Skeleton Calibration

Each photodetector on the motion capture suit gets associated with one or possibly more joints. These associations are based on whether or not a particular photodetector is rigidly attached to a particular joint. For example, a photodetector on the upper arm may be associated with the elbow joint, but a photodetector on the torso would not.

To determine the parameters relating the joint positions and orientations to the photodetector locations, the photodetectors associated with each joint are divided into groups of three. Three photodetectors make up the vertices of a triangle, and since the photodetectors have unique IDs, the orientation and position of the triangle will be uniquely defined given the locations and IDs of the vertices. The origin of the triangle is simply taken to be the location of the first vertex. Any vertex can be defined to be the first vertex, and this designation is somewhat arbitrary. A local coordinate system is then created for the triangle. The local x -axis is taken to be the vector from the first vertex to the second vertex. The local y -axis is created by taking the cross product between the local x -axis and the vector from the first vertex to the third vertex. The local z -axis is created by taking the cross product between the local x -axis and the local y -axis. This origin and local coordinate system completely defines the position and orientation of the triangle within the 3D space.

The triangle position and orientation is now known, and if the 3D position of the joint during calibration is also known, then it is possible to determine the positional offset between the origin of the triangle and the position of the joint, in terms of the triangle's local coordinate system.

The first step is to find the rotation matrix that converts the triangle coordinate system expressed in terms of the triangle coordinate system to the triangle coordinate system expressed in terms of the global coordinate system. For now, ignore any

translation between the two coordinate system expressions:

$$R_1 T_{sys,T} = T_{sys,G}$$

R_1 is the rotation matrix.

$T_{sys,T}$ is the triangle coordinate system in terms of the triangle coordinate system.

$T_{sys,G}$ is the triangle coordinate system in terms of the global coordinate system.

Since the triangle coordinate system expressed in terms of the triangle coordinate system is simply the identity matrix, this becomes:

$$R_1 I = T_{sys,G}$$

$$R_1 = T_{sys,G}$$

Where I is the identity matrix.

The next step is to find the positional offset from the triangle origin to the joint position, in terms of the triangle's local coordinate system. Multiplying this positional offset by R_1 found above will rotate the position offset to be in terms of the global coordinates. Adding the origin of the triangle to this will give the 3D joint position in terms of the global coordinate system:

$$R_1 J_{offset,T} + T_{origin,G} = J_{pos,G}$$

R_1 is the same rotation matrix as before.

$J_{offset,T}$ is the positional offset from the triangle origin to the joint position, expressed in terms of the triangle coordinate system.

$T_{origin,G}$ is the 3D position of the triangle origin, expressed in terms of the global coordinate system.

$J_{pos,G}$ is the 3D position of the joint, expressed in terms of the global coordinate sys-

tem.

Substituting R_1 for the expression found above, the postional offset in terms of the triangle coordinate system $J_{offset,T}$ is found:

$$T_{sys,G}J_{offset,T} + T_{origin,G} = J_{pos,G}$$

$$J_{offset,T} = T_{sys,G}^{-1}(J_{pos,G} - T_{origin,G})$$

Similarly, if the orientation or local coordinate system of the joint during calibration is also known, then it is possible to determine the transformation from the triangle coordinate system expressed in terms of the global coordinate system to the joint coordinate system expressed in terms of the global coordinate system.

$$T_{sys,G}R_2 = J_{sys,G}$$

$$R_2 = T_{sys,G}^{-1}J_{sys,G}$$

R_2 is the transformation.

$J_{sys,G}$ is the joint coordinate system expressed in terms of the global coordinate system.

$T_{sys,G}$ is the triangle coordinate system expressed in terms of the global coordinate system.

Once the positional offset $J_{offset,T}$ and the rotation matrix R_2 are known, the position and orientation of the joint can be computed given the locations and IDs of the triangle vertices. The procedure is demonstrated in section 3.3 Skeleton Generation.

With the design of the current motion capture suit, all the joints have more than three photodetectors associated with them. Thus, for each joint, multiple triangles are created in this manner from various combinations of three photodetectors. This provides increased accuracy and redundancy in case some photodetectors become oc-

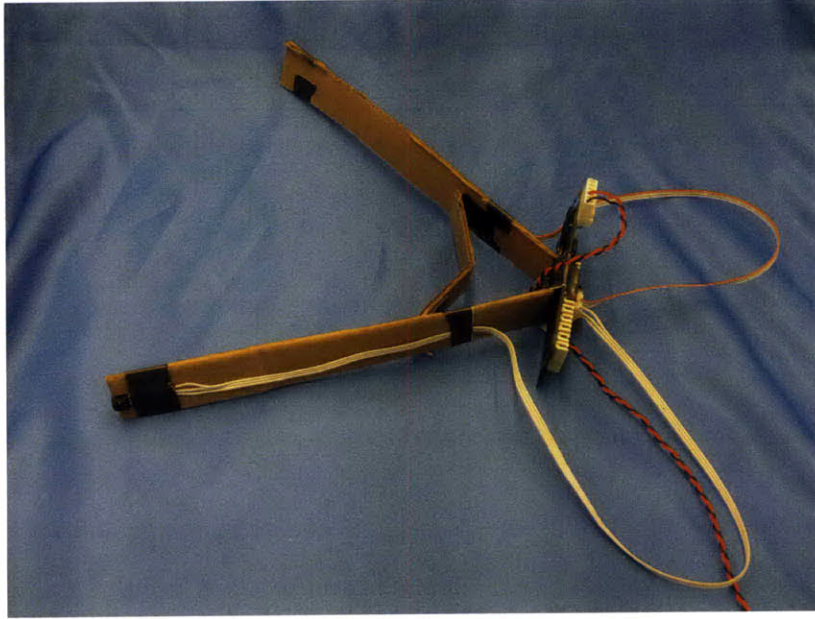


Figure 3-1: Calibration tool used for the skeleton calibration.

cluded. In addition, rather than using only photodetectors for the vertices of the triangle, computed joint positions are also used for some of the triangle vertices. This allows for increased flexibility in generating the skeleton.

The 3D location of each joint during calibration $J_{pos,G}$ is determined using one of two methods. The first method involves the use of a specially designed calibration tool that is essentially a pair of tongs with a photodetector placed on the end of each arm. During calibration of a joint, the tongs are placed around the relevant joint, and the joint location is simply taken to be the center point between the two photodetectors. The joints are calibrated one at a time, and the calibration tool is moved from joint to joint as each one is calibrated. The second method is similar, but the joint location is taken as the center of a set of photodetectors on the motion capture suit itself. The first method is used for the left shoulder and left elbow joints, while the second method is used for the lower body, upper body, and left wrist.

Currently, the joint coordinate system at the time of calibration $J_{sys,G}$ is always taken to be the same as the global coordinate axes, or in other words, the identity matrix.

This, however, is not an ideal method for establishing the joint coordinate systems because this causes the joint coordinate systems to be different relative to the orientation of the user depending on the user's pose during the calibration. Instead, the joint coordinate systems should be somehow based on the positions of the photodetectors.

3.3 Skeleton Generation

After determining the positional offset $J_{offset,T}$ and transformation matrix R_2 for every triangle associated with a joint, these parameters are used to compute the joint position $J_{pos,G}$ and orientation $J_{sys,G}$ given the IDs and locations of the vertices of at least one triangle associated with the joint.

From the given triangle vertices, the triangle coordinate system $T_{sys,G}$ and origin $T_{origin,G}$ are created in the same manner as described in section 3.2 Skeleton Calibration. The 3D position of the joint is found using the computed positional offset $J_{offset,T}$ for that triangle and joint that was obtained in the skeleton calibration phase:

$$J_{pos,G} = R_1 J_{offset,T} + T_{origin,G}$$

$$J_{pos,G} = T_{sys,G} J_{offset,T} + T_{origin,G}$$

Similarly, the orientation of the joint $J_{sys,G}$ is computed using the transformation matrix R_2 for that triangle and joint that was obtained in the skeleton calibration phase.

$$J_{sys,G} = T_{sys,G} R_2$$

If vertices for multiple triangles are known for a particular joint, then the joint position and orientation is taken as the average of the solutions produced from each triangle.

This produces the positions and orientations of the joints in terms of the global coordinate system. However, some third party applications that work with motion capture skeletons such as Autodesk Maya, require the orientation for each joint to be specified relative to its parent joint. A particular joint considered the parent of another joint if moving the parent joint also moves the other joint, also referred to as the child joint. For example, bending the elbow joint also causes the wrist joint to move, so the elbow would be the parent joint of the wrist. For the skeletal models developed for Second Skin, each joint has exactly one parent joint. The lower body joint is the parent of the upper body joint, which is the parent of the left shoulder, which is the parent of the left elbow, which is the parent of the left wrist.

To transform the orientation of each joint to be relative to its parent joint, the child joint coordinate system simply needs to be multiplied by the inverse of its parent joint coordinate system:

$$J_{child,sys,P} = J_{parent,sys,G}^{-1} J_{child,sys,G}$$

$J_{child,sys,P}$ is the child joint coordinate system in terms of its parent joint coordinate system.

$J_{parent,sys,G}$ is the parent joint coordinate system in terms of the global coordinate system.

$J_{child,sys,G}$ is the child joint coordinate system in terms of the global coordinate system.

3.4 Skeleton GUI

The same GUI as described in section 2.2.7 Motion Capture GUI is also used to display the motion capture skeleton. The visualization renders the 3D locations of the joints along with their local coordinate axes, and bones to connect the joints.

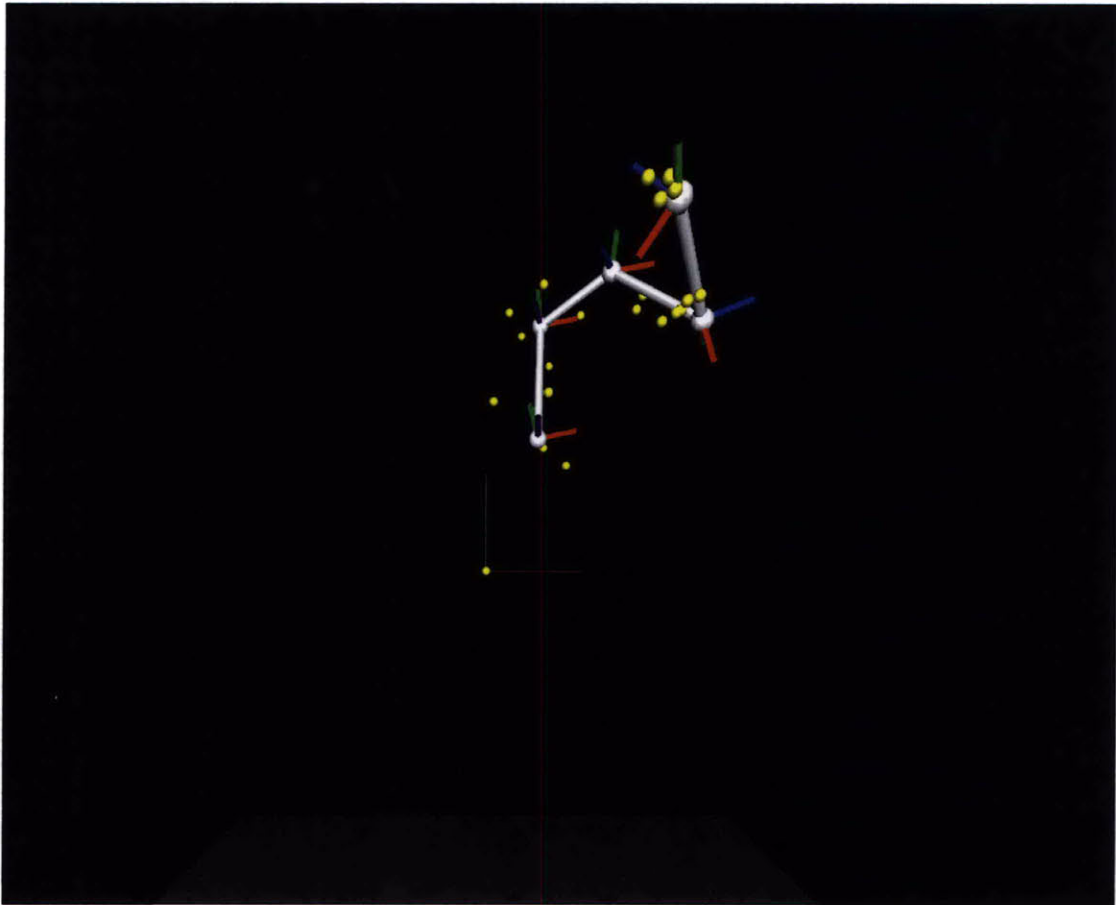


Figure 3-2: Screenshot of the real-time motion capture skeleton along with the 3D photodetector positions. The coordinate system for each joint is also shown.

Chapter 4

Tactile Feedback System

4.1 Overview

The tactile response system consists of a tactile control board which controls up to sixteen tactile actuators. The board supports two different actuation methods. The first method outputs voltage square waves up to five volts peak-to-peak to drive voice coil vibrotactile actuators. These actuators operate similar to speaker voice coils in that they vibrate in synchrony with an oscillating input voltage. In general, different types of oscillating input signals can produce different vibrotactile sensations on the skin, but square waves were chosen for their simple control and strong sensation. The frequency of the square waves is 250 Hz, which is the frequency that is most sensitive to the skin [1]. The second method outputs DC voltages from zero to five volts to drive vibration motors or any other DC actuator. DC vibration motors usually have an off-axis weight to create the vibrotactile sensation when the motor is run. Vibrotactile actuators are small, lightweight and easy to control, which makes them useful for this particular application where the user should remain as unhindered as possible. The tactile control board receives wireless commands from a computer via Bluetooth to activate any or all of the sixteen factors and to indicate the desired intensity level for each.

While the factors have not yet been integrated into the motion capture suit, the

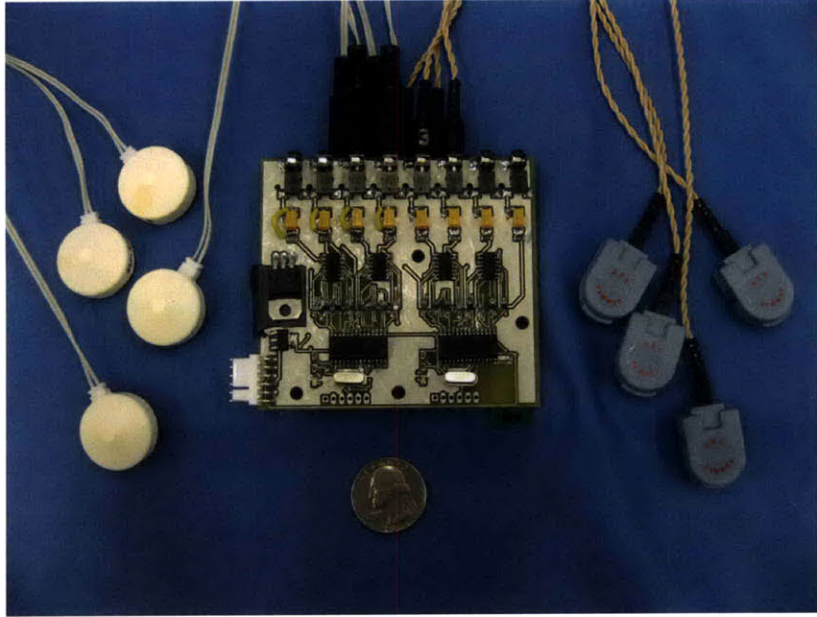


Figure 4-1: Tactile control board with tactors. TactaPack DC vibration motors on the left (white), and AEC voice coil actuators on the right (blue).

expected design was to have four tactors located around the wrist, and four tactors located around the upper arm close to the elbow, to be able to provide tactile feedback for an entire arm. The board-level control for the tactors has been implemented as well as a method to wirelessly control the board from a computer, but there is not yet software in place to analyze the motion capture data and determine how tactors should be activated to generate the corrective stimulus for the user. As such, the current state of the tactile response system is to provide functionality to easily explore various tactile feedback actuation techniques.

4.2 Tactile Control Board

The main components on the tactile control board are two PIC18F25K20 microcontrollers, sixteen Analog Devices AD5601 8-bit serial digital-to-analog converters, four Analog Devices AD8534 quad op-amps, and a Roving Networks RN-41 Bluetooth transceiver.

The control for the sixteen tactors is generated by two PIC18F25K20 microcontrollers, each of which controls eight of the tactors. Each microcontroller outputs commands to eight of the digital-to-analog converters. The analog signals are then sent through the op-amps which are configured as buffers to provide the higher current required by the actuators. For the square wave actuation, a 220 F DC-blocking capacitor is used to center the square wave at zero volts. For the DC voltage actuation, this capacitor is bypassed. The tactors plug into the board using 2.5 mm mono connectors, and the board can be powered from any voltage source between 6 and 15 volts. Onboard voltage regulators are used to deliver the proper voltages to the components.

The tactile board provides low-level control to generate several different types of vibrotactile stimuli. The first mode is simply a continuous vibration of the tactor, and the intensity is controlled by increasing or decreasing the voltage level. The second mode pulses the tactors on and off, and the intensity is controlled by increasing or decreasing the frequency of the pulsing and the voltage level. The third mode, and the most involved, pulses four tactors in sequence, but pulses each tactor twice before moving to the next one. For the four tactors located around the wrist, for example, pulsing each tactor twice in sequence will give the sensation of the pulses moving continuously around the wrist, in a phenomenon known as *cutaneous saltation* [2]. In short, if the timing is done correctly, giving two vibrations at one location on the skin and then a third vibration at a different location on the skin creates the sensation of three equally spaced vibrations moving from the first point to the second point. This method was intended to be used to indicate wrist and shoulder rotations. For this case, the intensity is controlled by increasing or decreasing the frequency of the pulsing and the voltage level.

To send commands to the microcontrollers from a computer, a Roving Networks RN-USB-X Bluetooth transceiver is plugged into a USB port on the computer. The RN-USB-X appears as a serial COM port to computer applications. The RN-USB-X

transceiver automatically connects to the RN-41 Bluetooth transceiver on the tactile board. Bytes sent to the RN-USB-X from the computer software or from a terminal program such as Hyperterminal or RealTerm are transmitted wirelessly to the RN-41 transceiver on the tactile board, which then are sent to the microcontrollers. A single command consists only of two bytes. The first byte is the ID value, and the second byte is the intensity value. The ID value indicates which tactor to activate, as well as which control method to use: continuous vibration, pulsing, or cutaneous saltation.

The ID values from 0-23 are used to specify the tactors controlled by the first microcontroller:

IDs 0-7: continuous vibration for the eight tactors.

IDs 10-17: pulsed vibration for the eight tactors.

ID 20: cutaneous saltation for the first four tactors.

ID 21: cutaneous saltation for the first four tactors in the reverse order.

ID 22: cutaneous saltation for the last four tactors.

ID 23: cutaneous saltation for the last four tactors in the reverse order.

The ID values from 30-53 are used to specify the tactors controlled by the second microcontroller:

IDs 30-37: continuous vibration for the eight tactors.

IDs 40-47: pulsed vibration for the eight tactors.

ID 50: cutaneous saltation for the first four tactors.

ID 51: cutaneous saltation for the first four tactors in the reverse order.

ID 52: cutaneous saltation for the last four tactors.

ID 53: cutaneous saltation for the last four tactors in the reverse order.

The intensity value indicates how strongly to activate the tactor specified by the ID value. An intensity value of zero turns off the tactor, while an intensity value of 0xFF

generates the strongest sensation for the tactors and control method specified by the ID value.

4.3 Tactors

The two vibrotactile actuators that were tested during the course of the project are voice coil tactors from Audiological Engineering Cooperation, and TactaPack DC vibration tactors developed by Rob Lindeman and his group at Worcester Polytechnic Institute.

During some early testing, it was found that when the tactors were located close together, such as around the wrist, it was often difficult to discern which tactor was vibrating if one of them was undergoing a continuous vibration. The tactor pulsing was designed to remedy this. When the tactors are pulsed, it was found that individual tactors could be distinguished much more easily. However, the pulsing technique tends to work well only for the voice coil tactors, due to their faster transient response. The startup and stop times for the DC vibration motors are too long to allow for pulsing at reasonable frequencies, which may go up to 15 Hz or higher. Slower pulse periods result in feedback that is less immediate, since there is no tactile stimuli when the tactors are in the 'off' portion of the pulsing. The AEC tactors feature a startup and stop time of only five milliseconds [1], as compared to the DC vibration motors which have startup and stop times of around 100 ms [7]. The benefit of DC vibration motors, however, is that they are usually capable of generating a stronger sensation when operating at similar voltage levels [7].

Chapter 5

Conclusion

5.1 Future Work

There are a number of elements that could be further developed to improve the system, as well as several areas of interesting continued research.

For the motion capture system, an easier and more user-friendly calibration method would definitely be a major step forward. The large 4x4 foot calibration grid, while effective, is large and difficult to manipulate and transport. The photodetectors must also be rotated manually during the course of the calibration phase so that they can receive data from every projector. There are calibration methods that exist that are easier from the standpoint of the user; the Vicon motion tracking system, for example, uses a small triangular grid and a wand with one marker on the end. The user places the small grid in the motion tracking space and waves the wand around such that all the cameras receive many corresponding marker locations from the wand and from the grid. Implementing a similar method or other simple procedure would make the motion capture setup and calibration much more hassle-free.

Also, the Vishay TSOP7000 photodetectors used for the project were unfortunately discontinued in mid-2009. While there are other photodetectors that operate in a similar fashion, the TSOP7000 was unique in that it was specially designed to handle

very high data rates. Resorting to other commonly available photodetectors would result in an unacceptable decrease in tracking rate. As such, an important area for further development would be in locating, or perhaps designing, a suitable replacement photodetector.

The motion capture suit could benefit from a redesign that allowed it to fit a larger range of body sizes, and made it easier to put on and remove. The current elastic Under Armor garment is a fairly tight pullover shirt designed to fit a small range of body sizes. Perhaps a design that utilized straps that could be tightened or loosened would allow for an easier, more adjustable fit as well as make it easier and faster to put on and take off. Also, if the position of the photodetectors on the motion capture suit were slightly adjustable, then more experimentation could be done in determining the optimal location for the best motion capture performance. To create a more complete system, the photodetectors would have to cover the entire body, and the tactors and tactile control board would also need to be integrated into the suit as well, ideally without adding much additional bulk to the suit. They may necessitate a slight redesign of the tactile control board and the other suit components.

The skeleton fitting accuracy and possibly the 3D point computation accuracy may also be improved through the use of better filtering techniques. Kalman filters are commonly used for motion data, and their application to the project may result in more reliable data.

Software to analyze the motion capture skeleton information and determine the high-level control for the tactile feedback also would need to be developed for the complete motion capture and tactile feedback motor-learning platform to be realized. Joint angles and orientations derived from the skeleton data could be compared against a reference data set or otherwise analyzed to determine the error of the user's body and limb positions. From the error information, the appropriate tactors would need to be activated so as to indicate corrective action to the user. Research would have to be

conducted to determine what sorts of stimuli are most effective, and to what extent such a system would be intuitive for a user, and to what extent a user would have to learn how to respond to the tactile feedback.

Another area of research interesting to pursue for future developments would be in other, new methods of tactile feedback that remain lightweight, but can provide stronger force. Vibrotactile actuators work well in the sense that they are small, lightweight, and effective in suggesting corrective motion, but they are somewhat limited in their ability to physically cause or guide motion; they will never be able to prevent a user from moving a certain way, or cause a user to move a certain way. Especially for activities that are higher speed, such as a golf swing, in which there is less time for a person to consciously react to a skin stimulus, a method of tactile feedback that can act as more of a physical guiding force would seem valuable. However, since fast motions are sometimes practiced slowly in order to develop proper form or technique, the current methods still may have value even for those types of applications.

Another possible area for future development is in researching what can be achieved by greatly increasing the density of photodetectors and the number of tactile feedback points. If a user was wearing a fabric covered with extremely small and densely packed sensors and actuators, perhaps an entirely new level of motion capture data and feedback control could be achieved. Fine motor movements of individual fingers and other subtle muscle movements could potentially be tracked and evaluated. It would apply similar techniques and principles but on a completely different scale. Perhaps such a system could be used for learning sign language or other fine motor control skills. This would likely require a revised approach to both the structure of the motion capture system and to the tactile feedback system to allow for the significantly higher sensor and tactor density.

5.2 Summary

In this modern age, technology puts a wealth of information and knowledge at peoples fingertips. Yet the way in which people must learn new abilities that require physical coordination, such as dance or martial arts, has remained largely unchanged. Regardless of the manner in which people acquire information about a topic of interest, or how much information they receive, if they wish to learn the movements they must ultimately dedicate the time to practice the motions themselves. Second Skin presents the development of a set of technologies designed to enhance this learning process, and improve one's motor-learning ability.

The Second Skin motion tracking system is unaffected by changes in lighting conditions, it has automatic marker IDs, and it is low-cost, which makes it robust, affordable, and useable in a variety of locations that are not suitable for other optical motion capture systems. Furthermore, complete privacy is ensured because there are no cameras, and no visual information is used. Real-time 3D motion capture of 72 points at 62.5 Hz was demonstrated within a cylindrical tracking volume 6 feet tall and with a 6.5 foot base. A motion capture suit with embedded photodetectors and electronics allows for convenient tracking of a human user. Photodetectors are located on the left arm, chest, torso, and back, which allows the skeleton modeling software to determine the joint positions and locations for the lower body, upper body, left shoulder, left elbow, and left wrist. The skeleton modeling software provides a more useful interface for motion tracking applications that do not want to process the raw motion capture data. These applications could be third party, or it could be software developed as part of future iterations of the project.

The tactile feedback hardware consists of a tactile control board capable of driving sixteen tactile actuators simultaneously. These actuators could be voice coil vibrotactile actuators that require an oscillating input voltage, or any actuator that requires a DC input voltage, such as a DC vibration motor. The board provides the low-level

control to continuously drive the tactors at varying intensities, pulse the tactors at varying rates, pulse a set of four tactors in a specialized manner so as to generate the cutaneous saltation effect, or any combination of the three simultaneously. While DC vibration motors are capable of creating stronger sensations, voice coil actuators feature very quick transient start and stop times which make them more effective for the pulsing and cutaneous saltation techniques.

While a fully functional motor-learning platform is not yet in place, the motion tracking and tactile feedback systems implemented here are strong developments towards the realization of such systems. The goal is that these technologies will facilitate further research and development of motor-learning systems that will be effective for a variety of real-world applications.

Chapter 6

Bibliography

- [1] Audiological Engineering Corporation. <http://www.tactaid.com/>

- [2] F.A. Geldard and C.E. Sherrick, The Cutaneous Rabbit: A Perceptual Illusion, *Science*, vol 178, 1972.

- [3] Ian R. Summers Ed. *Tactile Aids for the Hearing Impaired*. Whurr Publishers, London, 1992.

- [4] Lieberman, J. Accelerated and Improved Motor Learning and Rehabilitation using Kinesthetic Feedback. Masters Thesis, Massachusetts Institute of Technology, 2006.

- [5] Lieberman, J., Breazeal, C. TIKL: Development of a Wearable Vibrotactile Feedback Suit for Improved Human Motor Learning. 2006.

- [6] Lindeman, R.W., Page, r., Yanagida, Y., Sibert, J.L. Towards Full-Body Haptic Feedback: The Design and Deployment of a Spatialized Vibrotactile Feedback System. *Proc. of ACM Virtual Reality Software and Technology (VRST) 2004*.

- [7] Lindeman, R.W., Yanagida, Y., Hosaka, K., Abe, S. The TactaPack: A Wireless Sensor/Actuator Package for Physical Therapy Applications. *Proc. of the 14th Symp.*

on Haptic Interfaces for Virtual Environment and Teleoperator Systems (2006).

[8] Niwa, M., Noma, H., Yanagida, Y., Hosaka, K., Lindeman, R.W. Controlling the Perceived Vibrational Frequency and Amplitude of a Voice-Coil-Type Tactor. Proc. of the 14th Symp. on Haptic Interfaces for Virtual Environment and Teleoperator Systems (2006).

[9] Raskar, R., Nii, H., deDecker, B., Hashimoto, Y., Summet, J., Moore, D., Zhao, Y., Westhues, J., Dietz, P., Barnwell, J., Nayar, S., Inami, N., Bekaert, P., Noland, M., Branzoi, V., and Bruns, E. Prakash: Lighting Aware Motion Capture using Photosensing Markers Multiplexed Illuminators. SIGGRAPH (2007).

[10] Robertson, B. Big moves. Computer Graphics World 29 (2006).

[11] Tan, H., Lim, A., Traylor, R. A Psychophysical Study of Sensory Saltation with an Open Response Paradigm. Proc. of the 9th Int'l Symp. on Haptic Interfaces for Virtual Environment and Teleoperator Sys. (2000).

[12] Vicon Motion Systems. <http://www.vicon.com/>

[13] Buechley, L. and Eisenberg, M. Fabric PCBs, Electronic Sequins, and Socket Buttons: Techniques for E-textile Craft. Journal of Personal and Ubiquitous Computing. (2009).

[14] Park, S., Mackenzie, K., Jayaraman, S. The Wearable Motherboard: A Framework for Personalized Mobile Information Processing (PMIP). Design Automation Conference. (2002).

[15] Georgia Tech Wearable Motherboard: The Intelligent Garment for the 21st Century. <http://www.gtwm.gatech.edu/>

[16] Davis, R., Ounpuu, S., Tyburski, D., Gage, J. A Gait Analysis Data Collection and Reduction Technique. *Human Movement Science* 10. (1991).

[17] Kabada, M., Ramakrishnan, H., Wootten, M. Measurement of Lower Extremity Kinematics During Level Walking. *Journal of Orthopedic Research*. 1990.