UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

ACTIVITY RECOGNITION AND CRAWLING ASSISTANCE USING MULTIPLE INEXPENSIVE INERTIAL MEASUREMENT UNITS

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

JOSHUA B. SOUTHERLAND
Norman, Oklahoma
2012

ACTIVITY RECOGNITION AND CRAWLING ASSISTANCE
USING MULTIPLE INEXPENSIVE INERTIAL MEASUREMENT UNITS

A THESIS APPROVED FOR THE
SCHOOL OF COMPUTER SCIENCE

BY

Dr. Andrew H. Fagg

Dr. David P. Miller

Dr. Thubi H.A. Kolobe

Acknowledgments

This work is supported by grant R21HD061678-01A1: *Locomotion in Infants with or at Risk for Disabilities* (PI: T.H.A. Kolobe), National Institute of Child Health and Development. I would like to thank the families who participated in the study.

Drs. Thubi H.A. Kolobe and David P. Miller for all of their guidance in my graduate work. The SIPPC was from Dr. Kolobe's lab, and she guided the experiment design and data analysis.

I would also like to thank Dr. Peter E. Pidcoe for his work in designing and building the SIPPC. Also, Billy Mays and Greg Williams of the OU AME Machine Shop, who helped with many modifications throughout the evolution of the SIPPC.

Alex Olivier investigated the initial sensing requirements for the suit and tested the idea with a chain of accelerometers.

My colleagues from OU, including Charles de Granville, Thomas Palmer, Kim Houck, Di Wang, Matthew Bodenhamer, Emmanuel Goossaert, Scott Hellman, and Dr. Mark Woehrer for all of their help and support over the years.

The OU Health Science Center group: Megan Warner, Laura Rauh, Amy Laizure, Dakota Brown, and Thomas Hsing have supported this project in many ways, including helping with data collection, analyzing data, and building and improving the suit and SIPPC.

I would like to thank my parents, Paul and Becky Southerland, who have given me unending support throughout my studies. They have nurtured my early interest in computers, robotics, and in science.

I would not have made it without the encouragement and support of my loving fiancé, Joelle Harbour.

Finally, I would like to give special thanks my advisor, Dr. Andrew H. Fagg, for many years of his inspiration, guidance, and support.

Table of Contents

List of Tables

List of Figures

List of Algorithms

Abstract

Infants with Cerebral Palsy learn to crawl at a later age than typically developing infants, with many never learning to walk. The delayed onset of crawling is linked to delays in other areas of development. Previously, robotic crawling assistants have been designed to aid in early infant locomotion, with the hope of decreasing the deleterious effects of Cerebral Palsy. I propose an inertial measurement unit based sensing suit, composed of twelve sensors, that allows us to estimate the kinematic configuration of the infant's trunk and limbs in real-time. This approach, which is part of a larger study on crawling, enables the quantitative measurement of the infant's behavior as she attempts to interact with her environment and to locomote. Furthermore, the actions of a robotic crawling assistant are commanded in real-time based on specific gestures that have been recognized. My results show that this approach is a viable method for measuring infant motion. I also show that by using the suit as a measurement tool, changes in limb activity and coordination can be observed as the infants develop.

Chapter 1

Introduction

Cerebral Palsy (CP) is a mobility-limiting disorder that is estimated to affect three to ten out of every 1,000 infants (Anderson et al., 2003). Infants with CP experience decreased muscle coordination and strength as the result of neurological damage (Olney and Wright, 2006). Although CP is non-progressive, motor difficulties caused by brain injury can become worse if the infant does not continue to move. There is a risk that infants with CP may abandon attempts at crawling due to the extra difficulty caused by decreased muscle strength and coordination, which could also affect their subsequent cognitive development (Chen et al., 2010).

Early intervention might help infants with CP learn to crawl, as well as to reduce CP's effects on other areas of development. Early intervention is the practice of providing assistance to infants at an early age in order to aid in the development process. Ulrich et al. (2010) show that that treadmill training reduces the onset age of walking by three months in children with Down Syndrome (DS). Early intervention could also reduce the effects of CP on other areas of development, such as cognition.

Our goal is to motivate the infant to continue practicing limb movements so that the infants may explore the environment and gain other rewarding experiences.

Assisted-motion devices for infants with CP or DS have been proposed that aid the infant based on actions with varying similarity to crawling. Chen et al. (2010) propose a robot assistant for children with CP or DS. With typically-developing children (ages 11 to 17 months) in a prone position on the robot, leg motion was captured using a camera and marker system. When the children produced kicking motions that mimicked crawling, the robot provided forward/backward assistive movements. The children could also control the left/right turning movement of the robot with a joystick. They show that infant subjects increase their driving time, path length, and path complexity over four days in the study. Similarly, Schoepflin et al. (2010) proposed a similar robot assistant that was also controlled using a joystick for turning and markers on the legs for forward motion. Children (typically developing: 36 months; atypically developing: 49 months) were supported standing upright on a Pioneer 3-DX robot and asked to navigate mazes while avoiding obstacles. Schoepflin et al. concluded that the device was intuitive to control for children and that they could purposefully drive it through the maze.

This thesis is part of a larger study on the use of the Self-Initiated Prone Progression Crawler (SIPPC; Figure 1.1), which has been developed and employed across several previous studies (Kolobe et al., 2007; Kolobe and Pidcoe, 2007). This robotic assistant supports the infant in a configuration similar to crawling. The central wheels can be driven pro-actively, but can also be back-driven by the infant's own actions. A pressure plate located between the infant and the platform is used to determine the

weight distribution over time along the forward/backward and left/right axes. This plate can also be used to determine if the infant is attempting to push off of the SIPPC.

The SIPPC can support two different modes of assistance to the infant that is learning to crawl. First, when the infant causes a rotation of the wheels by pushing with the hands or feet, the SIPPC amplifies the movement. In this *powered assist mode*, a small forward motion of both wheels caused by pushing with both feet, for example, results in a powered movement in the same direction. Second, a shift in her weight distribution can be used to trigger a short motion of the SIPPC in the corresponding direction (*pressure assist mode*). Previous results using the SIPPC with typically-developing infants have been positive. The powered assist mode has been shown to reduce the time needed for typically developing infants to learn to competently drive the SIPPC by one month (Catalino et al., 2012). As previously mentioned, Kolobe et al. have found that infants with CP do drive the SIPPC with less amplitude than typically developing infants (Kolobe et al., 2007; Kolobe and Pidcoe, 2007), which suggests that the same gains from practicing with powered assist would not be realized in infants with CP.

Kolobe et al. hypothesize that infants with CP are more capable of making spontaneouss movement than of motions that will actually cause movement of the body or SIPPC (Kolobe et al., 2007; Kolobe and Pidcoe, 2007). This is because it is feasible for infants with CP to produce unloaded gestures, even with the reduced motor strength and lower coordination that often accompanies CP. Because spontaneous movement is more likely to be successfully accomplished, a system that rewards these gestures with assisted movement should be likely to motivate continued use.

It is possible that rewarding spontaneous movements could provide more encouragement than previous SIPPC approaches (Kolobe et al., 2007; Kolobe and Pidcoe, 2007; Catalino et al., 2012) which require actual movement forces to be generated by the infant. In order to improve on previous results with the SIPPC, these authors recommend additional powered assistance in the form of a sensor suit. I propose the use of IMU's to capture crawling-like actions with the trunk and limbs, in order to respond with real-time powered assistance.

There are a variety of approaches to sensing crawling-like limb movements (Chen et al., 2010). A common method for obtaining limb and body position information is to use a camera with actively controlled infrared markers, such as with Vicon™ motion capture systems (Glardon et al., 2004; Herda et al., 2000). Vision based systems suffer from occlusion, which prevents their use in situations where parts of the body may be hidden. Another method, used in Polhemus™ tracking sensors, is to obtain sensor pose using a self-constructed magnetic hemisphere. This approach can be very accurate, but, in most cases, has a limited range and is subject to distortions in the local magnetic field by ferrous metals or electrical current moving through motors. Alternatively, inertial-based sensors can be used to recognize the activities in which a subject is engaged.

Full Inertial Measurement Units (IMUs), which contain a gyroscope and/or magnetometer, in addition to an accelerometer, are able to provide accurate 3D orientation and rotation velocity, as well as linear acceleration within the local coordinate frame. For example, Shi et al. (2009) used Hidden Markov Models and Support Vector Machines to classify motion types given input from a set of IMUs.

The advantage of an inertial-based approach is that the sensors are entirely body-worn. The need for external components is minimized, which simplifies the process of quickly deploying such a system for data collection. However, for infants, the sensors and supporting hardware must be small, low mass and easy to mount quicklyh. Here, I propose an IMU-based kinematics suit that addresses many of these issues. With the suit, IMUs are mounted to the back and shoulders, and on multiple points on the limbs. Coupled with a kinematic model of the infant, these data enable us to reconstruct the configuration of the infant's trunk and limbs in real-time. This facilitates the recognition of gestures for the online triggering of assistive movements. In addition to triggering assistance, recognized gestures provide a quantitative means of describing the infant's behavior during the development of crawling skill.

I also propose an approach to gesture recognition using template matching against a canonical menu of actions. The proposed approach must be capable of real-time recognition of actions from the canonical menu. These actions, such as "push left with the left arm" or "push forwards with both arms", vary in complexity and possible usefulness. I will prioritize the assistance triggering process so that the more complex actions are considered first.

This thesis continues with a summary of related and background work in Chapter 2. A description of the kinematics suit is given in Chapter 3. Chapter 4 describes the suit calibration process and kinematic reconstruction. The canonical menu of actions, how the suit is used to recognize actions in real time and how the suit is used to control the SIPPC, are detailed in Chapter 5. Experimental results are presented

in Chapter 6, including both validation experiments done in the lab and infant experiments performed in the field. Details on parts of this work have been submitted for publication (Southerland et al., 2012).

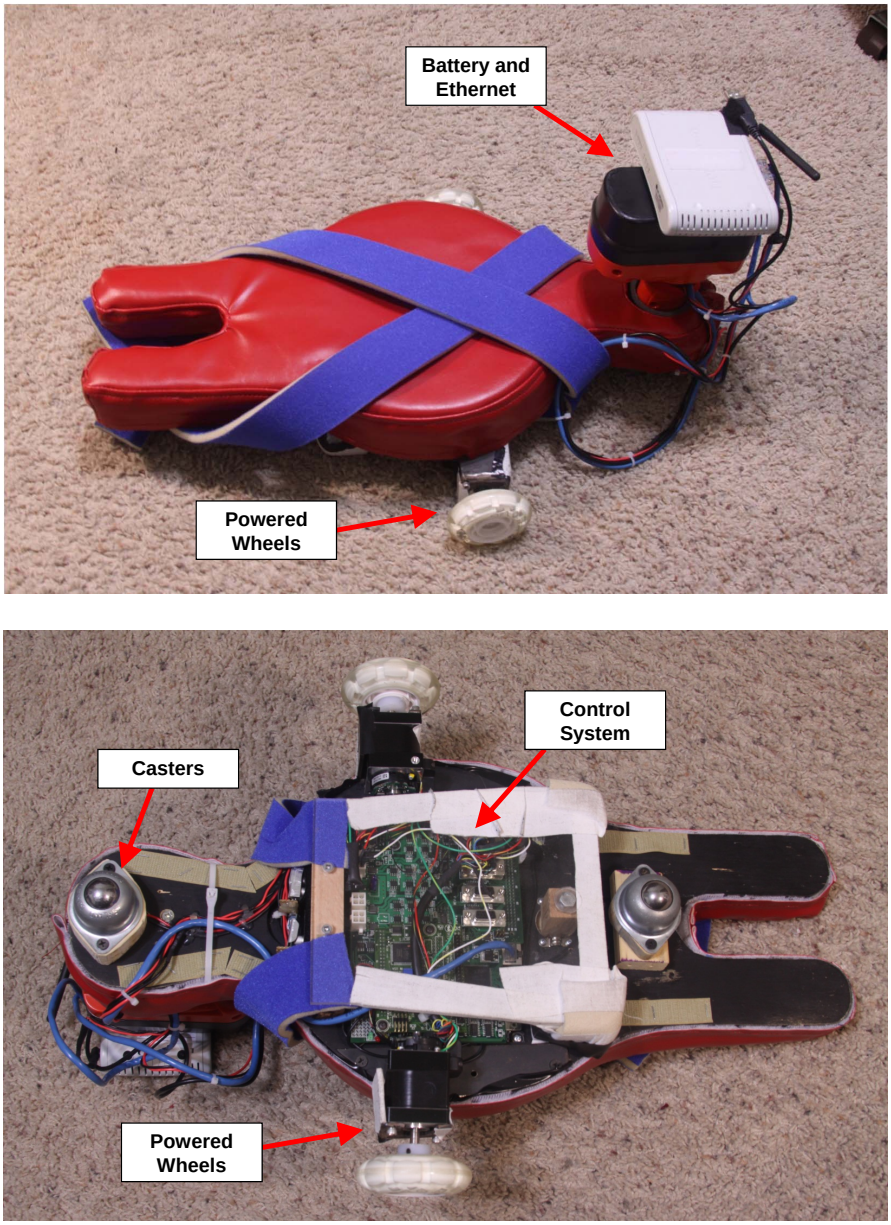

Figure 1.1: *SIPPC: top and bottom views. The U-shaped feature supports the infant's head while allowing them to breath easily. Also, this design allows some visual access to objects, as well as to the reachable parts of the floor. The powered central wheels are located near the infant's center of mass. Casters mounted toward the front and back of the robot prevent tipping. A Wi-Fi connection enables remote, real-time data logging and high-level control by a nearby laptop computer.*

Chapter 2

Background

2.1 Cerebral Palsy

Cerebral Palsy is a mobility-limiting disorder estimated to affect three to ten out of every 1,000 infants, based on age at birth (Anderson et al., 2003). Improvements in the survival rate of low birth weight infants have caused an increase in the prevalence of CP (Bhushan et al., 1993). While CP is a non-progressive disorder, it can have lasting side effects on cognitive and social development, as well as possibly delaying functional independence (Campos et al., 2000). Treatment is complicated, because early detection of CP is limited and the frequency of symptoms can vary (Hadders-Algra, 2001). Diagnosis can even take up to two years in some cases (Bhushan et al., 1993).

Infants learn to locomote by first generating spontaneous limb movements while in a prone or supine position. Early spontaneous limb movements cause the infant's body to move, which are rewarding and drive the infant to continue exploring limb movements. Through practicing these limb movements, infants learn to crawl. There

are differences in coordination and magnitude between the movements seen in typically developing infants and those with CP. These differences can make exploring early whole-body movement difficult, but are useful for diagnosing CP (Hadders-Algra, 2001). If early spontaneous limb movements are unsuccessful, the infant may not be enticed to further explore those movements. As a result, onset of crawling may be delayed, or the infant may never crawl. Delayed or prevented onset of crawling may also have side effects on cognitive development, as exploration of the infant's environment may be limited and less frequent.

2.2 Early Intervention

Nilhom (1996) reviews the field of early intervention with Down Syndrome (DS) and analyzes recent trends. The bulk of Nilhom's review is a re-interpretation of an older review by Gibson and Harris (1988) on the results of available papers at the time. The reviewed group of papers covers studies of the effects of early intervention on five different developmental domains in infants with DS. Of the 21 studies originally reviewed by Gibson, nine involve the gross motor domain. The studies typically involved "behavior modification" training given to the mother to practice at home with the infant. Of the nine studies, Gibson and Harris find that the subjects receiving early intervention in six show "little or no improvement". Upon re-evaluation however, Nilholm summarizes:

> *of six studies taken by Gibson and Harris (1988) as indication of "little or no improvement," two seem to involve quite large gains by the*

treatment-group, two involve gains, although insignificant, one does not have a bearing on the issue at stake and one is a clear indication of no improvement. Even though the pattern is not as clear as to the hearing/speech domain, it does seem that the bulk of evidence suggests positive effects for gross motor training (Nilholm, 1996).

Although many of these results in the gross motor domain seem qualitative, Nilholm finds some amount of encouragement for intervening early.

Mahoney et al. (2001) study the effects that early intervention has on infants with CP and DS. They studied 27 infants with DS and 23 with CP, who received either Neurodevelopmental Treatment (NDT) or Developmental Skills (DevS) treatment strategies. NDT was designed to help infants with CP move more naturally by manually inhibiting abnormal muscle tone, which can dominate normal reflexes such as righting and equilibrium. NDT is also used to assist infants with other impairments, such as in infants with DS. DevS is a different treatment model that involves practicing exercises in order to meet motor skill milestones associated with a modeled motor development trajectory. The authors explain that infants with DS are characterized as having a muscle response delay, whereas infants with CP experience interference due to noisy activation of primitive reflexes. As a result, they recognize that different intervention strategies might be required. Infants with DS need experience to learn skills faster, where infants with CP could benefit, instead, from inhibited problematic muscle tone. Mahoney et al. separate the subjects into “low intensity” and “high intensity” groups based on frequency of intervention. The authors show that all groups

improve in motor development over one year of the study. Also, the low intensity group shows motor development gains of approximately five months, whereas the high intensity group has made gains of about seven months. The difference between the high and low intensity groups is significantly different. However, Mahoney et al. feel that the difference could be explained by an increase in the parental involvement level for the high intensity group, and do not have a "no-treatment" control group to determine which was the case. Mahoney et al. do compare characteristics of groups receiving DevS and NDT, but not their overall success.

Ulrich et al. (2011) show that early intervention using a treadmill can reduce the time of onset of walking in infants with DS. The authors worked with 30 infants with DS, randomly split into a traditional therapy (control) group and a traditional therapy plus treadmill training (experimental) group. The thirty infant subjects began the study at approximately ten months of age. The control group began assisted walking at 240 days old and independent walking at 401 days, on average. The authors find that the onset of walking in the experimental group decreased by 73.8 days with assistance and by 101 days independently, relative to the control group. Their results are an example of early intervention being successful in infants who receive frequent home-based training.

2.3 Assisted Motion Devices

Schoepflin (2010) place children in a seat on a Pioneer 3-DX robot, and allow them to drive through a maze. Markers are attached to the children's legs, which were

allowed to remain free. Turning assistance is triggered using a joystick and forward assistance is triggered by kicking, where forward velocity assistance is proportional to the speed of the feet. Schoepflin's subjects include five typically developing children that are approximately three years old, and one 49-month old child diagnosed with spastic CP. The typically developing children are given five trials to navigate the maze, while the infant with CP is given ten trials. Schoepflin hypothesizes that children will show increased coordination and skill as they gained experience with the task, which will consequently decrease trial completion times. Schoepflin finds that the device was intuitive for the children to drive, but does not show significant improvements as trial count is increased. Low correlations could be a result of the wide variation in the subjects' cooperation and enthusiasm during the study.

Chen et al. (2010) use a Pioneer 3-DX, but support children in a prone position as opposed to upright. Two typically developing children were chosen for the study, ages 11 and 17 months old. Kicking motions are recognized using a marker system and camera, which is mounted high on the robot and is pointed down at the infant. The recognized kicking motions trigger a forward assistance with speed proportional to the speed of the infant's legs. The authors use a single camera to detect the position and orientation of both markers, which are placed near each ankle. The children can also turn the robot using a joystick. Chen et al. test if children can tolerate the assistive device for longer and make better use of it as the children gain experience with the device. The results show that both infant subjects increase their driving time, path length, and path complexity over four days in the study. The authors also observe that one subject showed signs of purposeful driving (towards a caregiver)

after three days of training. Chen et al. suggest that camera marker systems would be more successful in future work because they don't require the fine motor skills that joysticks do.

2.4 Self-Initiated Prone Progression Crawler

Kolobe and Pidcoe (2007) developed the Self-Initiated Prone Progression Crawler (SIPPC). Kolobe et al. use the SIPPC in order to promote prone locomotion in infants with or at risk of developing CP (Kolobe and Pidcoe, 2007; Kolobe et al., 2007). To determine the feasibility of the SIPPC, they conducted the first study with ten infants between 4-22 months of age. The study included four typically developing subjects, three with CP, and three with DS. The infants were encouraged to move on the SIPPC for three trials, twice a week for eight weeks. Kolobe et al. recorded balance information, wheel velocity, odometry, and videotape movements for each session. Their results show that typically developing infants learn to use the SIPPC independently (with no active assistance) by six to seven months of age, while the infants with CP or DS do not. SIPPC movements made by infants with CP or DS are characterized by smaller amplitudes, as measured by wheel encoders. Decreased movement amplitudes in infants with CP and DS suggest that they require more sensitive triggering. Movement speed and direction data also indicate decreased coordination. These results show that the SIPPC is a useful tool for measuring infant movement strategies and skill development.

Catalino et al. (2012) quantified limb movement patterns that typically developing infants use as they became proficient with the SIPPC. Ten infants between the ages of four and seven months participated in a study for up to twelve weeks each. The resulting data includes 500 vignettes that were coded using the Movement Observation Coding System (MOCS) to assess the infants' activity over the study period. MOCS (Rule, 2010) is comprised of a set of measures of motor skill development in infants. These measures include separate categories of posture, limb movement exploration and progression, and mastery of propelling the SIPPC. MOCS is also used to measure emotional responses of infants during trials on the SIPPC. The authors' results show a transition from widely varied movements at the early stages of development to more repeated movements. They also notice differences between limb movements used to propel the SIPPC and those used to locomote on the ground. These differences seem to be due to the support provided by the SIPPC against gravity and the height at which infants are supported. However, they find that many infants were capable of exploring the local environment with the SIPPC prior to unassisted crawling. Also, typically developing infants with powered assistance from the SIPPC learn to independently control the SIPPC one month earlier than those using the SIPPC in a passive state.

2.5 Motion Sensing

Chen et al. (2010) discuss a number of different approaches to sensing crawling-like movements. They first consider using computer mice, which could be used to

detect sweep movements from the arms on a flat surface. Computer mice are low cost and easy to interface to robotics software, but have a limited range of motion and have a tendency to be considered as toys by the infants. The authors also consider touch screens, which are easy to interface to, as well, and can encourage interaction through the use of color. However, touch screens can be expensive, potentially unintuitive to infants, and don't require much exercise to use. Chen et al. note that stereo cameras can be used to track head orientation to control robot orientation and leg activity to trigger forward/backward assistance. The authors also note that linear encoders, which directly sense position, make position, velocity, and acceleration of the hands easy to obtain. However, the authors point out that linear encoders limit limb motion. Accelerometers require integration to infer velocity information and are inaccurate due to drift. Strain gauges under the belly are expensive, but as an advantage, detecting a shift in weight should be feasible even if an infant does not possess fine motor skills. Finally, the authors mention that electromyography (EMG) can be used to detect bicep and leg muscle contractions in order to drive and orient the robot. Chen et al. note that an EMG approach is desirable because limb activity can be rewarded. It is possible that if more natural limb movements are rewarded, the likelihood for significant crawling onset gains made through early intervention could be greater. Unfortunately, EMG is difficult to use because of high classification errors associated with the current generation of sensors. Also, the required hardware and software interface layer is a burden to use with mobile crawling assistants.

Conover (2003) presents a new approach to measuring infant motion with limb-mounted accelerometers. The Gestalt Perception Principles (GPPs) are used in order to estimate how likely an infant is to develop CP. The GPP process involves assessment by a physician of motions made by an infant in their first two years following birth. The physicians quantify the amount of abruptness and smoothness of actions, as recorded on film. Conover provides a quantitative means of estimating the likelihood of developing CP, by obtaining measurements of common infant actions. Conover uses accelerometers to quantify General Movements (GMs), which are gross, spontaneous, whole body movements made by infants early in development. The hope is also that GMs will make comparison between different studies possible because GMs don't depend on qualitative assessment and the inaccuracy of multiple human reviewers using Gestalt Perception Principles. Conover used two-dimensional accelerometers placed on each wrist and ankle, for a total of four sensors. The sensors are placed in an enclosure measuring $2.2 \times 2.4 \times 0.6$ inches and are wired to data logging hardware being monitored by LabView.™ He employs two approaches at quantifying GMs given limited data. Analyzing the frequency at which GMs are utilized does not seem to yield useful information for predicting CP development. However, correlations between limb acceleration magnitudes successfully point out periods of coordination between limbs. In his future work section, he recommends developing smaller wireless sensors.

Dillon (2005) replaces accelerometers with a new wireless version with a circular footprint diameter of 1.25 inches and less than one inch thickness. He tests the new accelerometers on five infants and finds that the sensors do not have consistent

output frequencies, which makes analysis difficult. Also, the chosen sensors measure accelerations in a range of ± 25g, which limits the resolution available in the ± 3g range, causing signal-to-noise problems. Although these hardware limitations prevent Dillon from making many advancements in GM analysis, he concludes that the physical sensor configuration seems to work well and does not restrict the infants' movements, based on observing GMs.

Accelerometers are also used for biometric authentication (Gafurov and Snekkenes, 2009). The authors focus three approaches to gait sensing: video sensors, floor sensors and wearable sensors. Of these three, they choose to use wearable accelerometers, because of expense and portability. They attempt to identify individuals given time series of accelerations sensed at the feet, hips, trouser pockets, and arms. They find periods of motion and remove all data corresponding to standing still. Once the data corresponding to activity is found, they perform feature extraction in both the time and frequency domains. In the time domain, they normalize based on a walking cycle consisting of two steps and use sensor trajectories as feature vectors. In the frequency domain they use Fourier analysis to compute amplitudes in certain frequency ranges for constructing feature vectors. The final step in the algorithm is to match templates by using an Euclidean distance metric between the test feature vectors and the set of prototypes. The authors asked 21-100 subjects (depending on sensor location) to walk "normally" while recording acceleration from one sensor. The resulting identification rates are high, varying between 71.7% (arm) and 86.3% (pocket). This study shows that acceleration data alone are enough to distinguish between different individuals performing the same basic activity.

Lai et al. (2011) use six IMUs in order to detect human posture, with homecare service in mind. They use a 3D Adaptive human Motion Recognition (AMR) algorithm to reconstruct skeletal configuration given sensors mounted on the wrists, legs, neck, and waist. By using a Body Correction Algorithm (BCA), which is similar to a Neural Network, they are capable of calibrating the lengths of an individuals limbs. In order to perform this calibration, they make use of the correlation between limb length and measured joint angular accelerations. BCA takes limb accelerations and joint angular accelerations as inputs, limb positions as outputs, and trains using a method similar to backpropagation. The authors next use AMR in order to classify four motion states: standing, sitting, running, and walking. Their experimental group consists of twelve subjects, between the ages of 18 and 57. Their experiment involves matching joint angle estimates to templates within some tolerance for each of the four motion states. The resulting aggregate accuracies using AMR and BCA are 95.72% versus 90.3% using AMR alone. Using multiple sensors also results in higher classification accuracies relative to using a single sensor.

2.6 Action Recognition

Shi et al. describe a device that can automatically detect falling accidents for the elderly (Shi et al., 2009). They use small, wireless IMUs containing MEMS accelerometers and gyros in order to classify whether a person was standing, walking, running, climbing stairs, or falling. The three dimensional acceleration and angular velocity data were filtered in order to remove noise. The authors bin their data and

use a Fast Fourier Transform (FFT), which results in 60 features. These features corresponded to the first ten coefficients obtained by evaluating a FFT on each of the six vectors of acceleration and orientation data. The authors then classify the motion category type given five Hidden Markov Models (HMMs). Their HMMs estimate an action type probability at a particular time, given the recent history of sensed features. In order to solve the multi-class classification problem, they assign each test point to the class with the highest HMM probability. The resulting accuracies are between 90% and 100% for each action.

Chapter 3

Kinematics Suit

The design goals for the proposed kinematics suit include: accurately estimating the trunk and limb configuration of the infant in real-time, protecting the sensors, minimizing resistance to the infant's motion, and making it easy to place the suit on an infant so that data collection may begin very quickly in a session. To isolate the infant from the electronics, the suit is designed to be worn on the back with all of the electronics contained inside (Figure 3.1). The suit is secured using elastic straps with hook and loop fasteners. The straps also help to keep the rectangular IMU sensor boards from rotating or slipping during data collection.

The suit is tethered to the SIPPC for power and access to a Wi-Fi hub (Figure 3.2). Due to its size and power requirements, I have chosen the UM6 IMU from CH Robotics. This sensor module contains a 3-axis accelerometer, a 3-axis gyroscope, and a 3-axis magnetometer. Figure 3.3 shows how the laptop, SIPPC, mBed ARM7 development board, and UM6 sensors interact.

The UM6 IMUs use an on-board Extended Kalman Filter (EKF) to filter the raw data from the magnetometers, accelerometers, and rate gyroscopes, as well as for

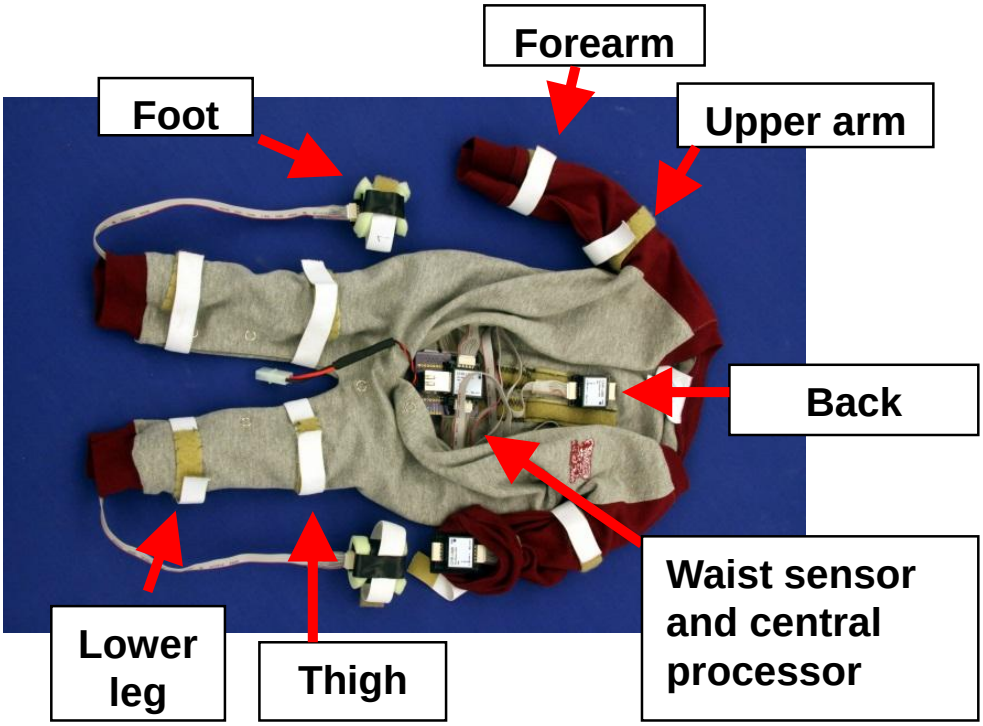

Figure 3.1: *The sensing suit with sensor locations labeled. One base sensor and eleven additional sensors are used. The suit draws power from the SIPPC, and the ARM7 processor transmits data via an Ethernet cable.*

estimating the sensor's 3D orientation. The suit allows easy placement of the sensors on the arms, legs, feet, back and waist. A *mBed* development board containing an ARM7 Cortex™-M3 processor gathers the sensor information from these sensors at 50Hz over a dual Serial Peripheral Interface (SPI) bus. With the combined information, one can infer the orientation of each sensor in a coordinate frame that is rooted at the waist.

There are two steps to calibrating the UM6 IMUs in the suit. First, the IMUs themselves must be calibrated. During this offline process, the biases of each of the 3D sensors are modeled, including variations due to temperature. The 3D orientation of the sensor is referenced in a right-handed, North-East-Down (NED) coordinate

Figure 3.2: *A typically developing infant wearing the suit while supported by the SIPPC. The infant's legs are suspended, which is typical for new subjects with smaller bodies.*

frame. Second, an online calibration step is necessary at experiment time to account for the placement of the sensors on the infant. This process is described in Section 4.1.

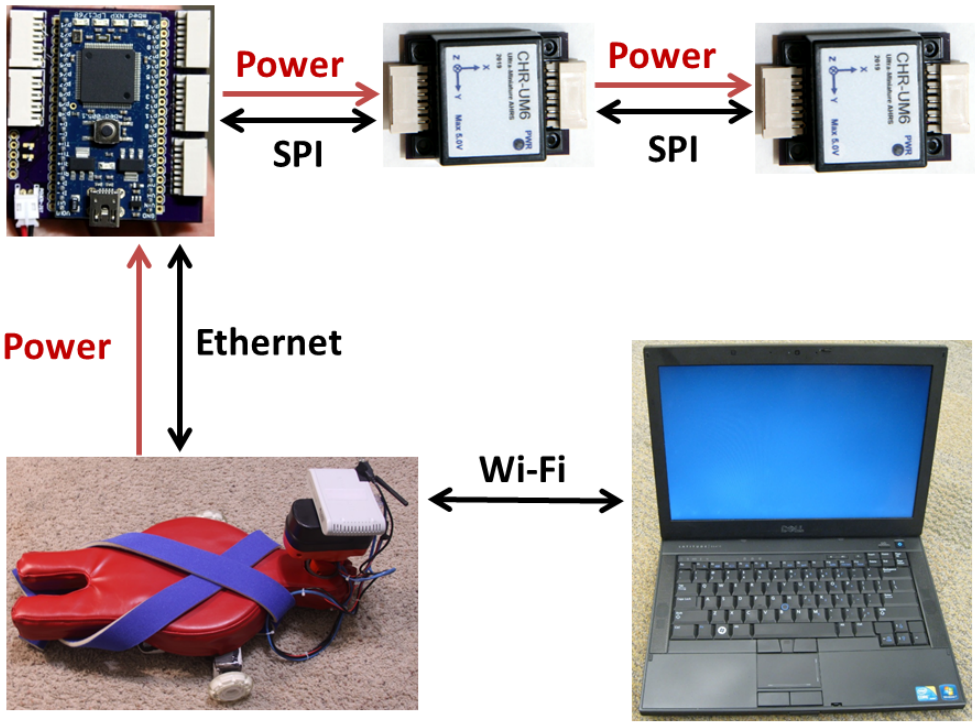

Figure 3.3: *The robot and data collection system. The Kinematics Suit powers the twelve UM6 sensors and communicates to them via SPI. The suit itself is powered by the SIPPC. The SIPPC and Suit data both travel over ethernet to a wireless adapter and then over Wi-Fi to a laptop. The UM6 sensors can be daisy chained together up to a maximum of three per chain, because there are three separate slave-select lines per cable.*

Chapter 4

Kinematics

4.1 Kinematic Model

The current configuration of the infant's trunk and limbs is estimated by combining a kinematic model of the infant with the IMU-based estimates of the orientation of the sensors mounted at key points of the infant's body. Figure 4.1 outlines this model, where sensor locations are shown as rectangles on the trunk and the right arm. Joints are represented as circles, and are assumed to allow arbitrary three degrees of freedom (DOF) rotations. However, the translational relationship from one frame in the kinematic tree to the next is assumed to be constant. The key coordinate frames are as follows:

- $G =$ the global coordinate frame to which the orientation of each of the IMUs is referenced,

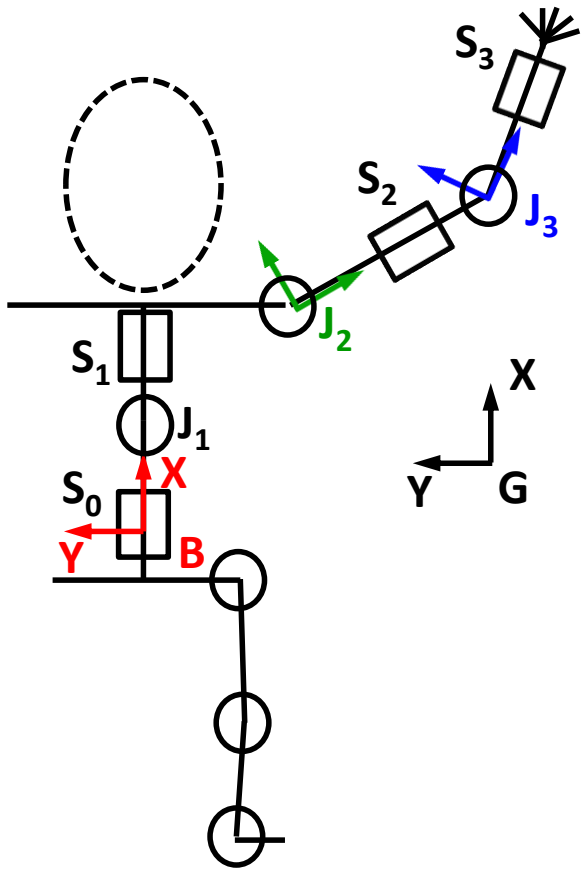

Figure 4.1: *Partial infant skeletal model and kinematic reconstruction for an arm, using two sensors on the arm and one each on the shoulders (between the shoulder blades) and on the waist. An arbitrary global coordinate frame is also shown, which is defined during the offline EKF reference calibration process.*

- B = the base coordinate frame of the SIPPC, which I define to be the coordinate frame of the sensor mounted on the waist. The position and orientation of all other coordinate frames are computed relative to this base frame,
- S_k = the coordinate frame attached to sensor k, and
- J_k = the coordinate frame of joint k after rotation.

Not shown in the model are the sensors mounted to the thigh, lower leg, and foot of the right leg, or sensors mounted on the left limbs.

Once calibrated, the kinematic model defines a partial set of position and orientation constraints between joints and the sensor units. The IMU-based orientation estimates supply the remaining constraints, allowing us to infer the position of any point on the kinematic model relative to the base frame.

4.2 Estimating Joint Orientation

The rotation matrix that describes the orientation of each joint, J_k, within the base coordinate frame is expressed as ${}^{B}_{J_k}R(t)$. This rotation is inferred from the orientation of sensor k in the base coordinate frame, ${}^{B}_{S_k}R(t)$, and knowledge of the relationship of the corresponding joint and sensor ${}^{S_k}_{J_k}R$. Specifically:

$$ {}^{B}_{J_k}R(t) \quad = \quad {}^{B}_{S_k}R(t) \quad {}^{S_k}_{J_k}R \ . \tag{4.1} $$

The orientation of sensor k in the base coordinate frame is estimated using the orientation of the sensors attached to k and the base frame in the global coordinate frame. Furthermore, I assume the possibility of a fixed bias between what is reported by sensor k and the true orientation of sensor k, specifically:

$$ {}^{B}_{S_k}R(t) \quad = \quad {}^{G}_{B}R^{T}(t) \quad {}^{G}_{\hat{S}_k}R(t) \quad {}^{\hat{S}_k}_{S_k}R \ , $$

where ${}^{\hat{S}_k}_{S_k}R$ is the rotational bias. I also assume, without lack of generality, that there is no bias with respect to base sensor. Equation 4.1 then becomes:

$$ {}^{B}_{J_k}R(t) \;=\; {}^{B}_{G}R(t) \;\; {}^{G}_{\hat{S}_k}R(t) \;\; {}^{\hat{S}_k}_{S_k}R \;\; {}^{S_k}_{J_k}R \;, \tag{4.2} $$

where the first two terms of the right-hand-side are derived from the IMUs, and the latter two terms are fixed rotations. In general, these latter two terms are not known *a priori*. However, their product can be estimated through a calibration process that is performed after the kinematics suit is attached to the infant.

In our field experiments, it is necessary to perform any calibration process quickly. Our approach is to sequentially place each limb in a known configuration, so that ${}^{B}_{J_k}R(t)$ is known. Given the current IMU sensor states, one can then solve Equation 4.2 for the product of the rightmost two matrices. Although I there is no way to factor the contribution of the bias from that of the transformation from sensor to joint, there is no need to do so to make subsequent use of Equation 4.2.

4.3 Kinematic Reconstruction

For this step, our goal is to estimate the positions of each of the joints and the limb endpoints within the base coordinate frame. Without loss of generality, let $k = 0$ be the base coordinate frame, $k = 1...n-1$ be the joints from the base along one limb, and $k = n$ be the coordinate frame of the corresponding limb endpoint (e.g., the wrist). Furthermore, let ${}^{k-1}P_k$ describe the position of the center of rotation of joint k within the coordinate frame of joint $k-1$.

For a single limb, the location of joint coordinate frame k is defined by the recursive relation:

$$ {}^{B}P_{J_k}(t) \quad = \quad {}^{B}P_{J_{k-1}}(t) + {}^{B}_{J_{k-1}}R(t) \quad {}^{J_{k-1}}P_{J_k} \quad , $$

where ${}^{B}P_{J_0} = [0\ 0\ 0]^T$.

It is possible to estimate each of the relative transformations from one coordinate frame to the next (${}^{J_{k-1}}P_{J_k}$) through a calibration process that minimizes the squared difference between IMU-sensed linear accelerations and accelerations estimated from the kinematic model. In the context of this work, these transformations are measured by hand. The auto-calibration approach is discussed as future work in Chapter 7.

Chapter 5

Behavior Recognition

5.1 Canonical Menu

Once kinematic reconstruction is complete, the next step is to recognize spatio-temporal actions in real-time in order to describe the high-level behavior of the infant and to provide corresponding assistive actions by the SIPPC. The proposed approach is to match specific limb trajectories to particular assistive actions. For example, a "sweep the left arm from right to left" is mapped to a right turn of the SIPPC. Similarly, a trajectory involving both arms, "push both arms back while on/near the ground" is mapped to a forward movement. Figure 5.1 illustrates a partial menu of *canonical actions* and their corresponding SIPPC response.

A particular canonical movement is recognized through a template matching process. Each template is described by a combination of conditions:

1. a region of space within which the limb movement must begin its motion,

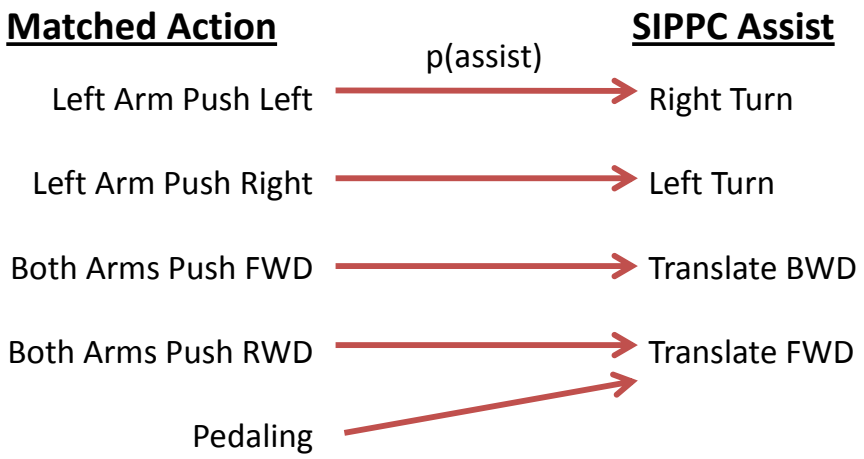

Figure 5.1: *Partial canonical movement menu. There exists a many-to-one mapping from canonical movements to assistive responses.*

2. a minimum relative displacement for the movement (described as a vector), and

3. a time period within which the movement must occur.

For example, the "sweep the left arm right" limb movement may be described by: any starting location, a displacement of $-0.05m$ along the Y dimension and completion within $0.5s$. Note that this template will accept a wide class of rightward movements. Because this movement mimics a push action to the right, the SIPPC will respond with a left turn. In particular, the infant need not be in contact with the ground in order to receive this response. Early in the development process, when movement is infrequent and uncoordinated, these general (easier) actions are rewarded with assistance. Later, as the infant is more active and productive on the SIPPC, only the more complex actions are rewarded.

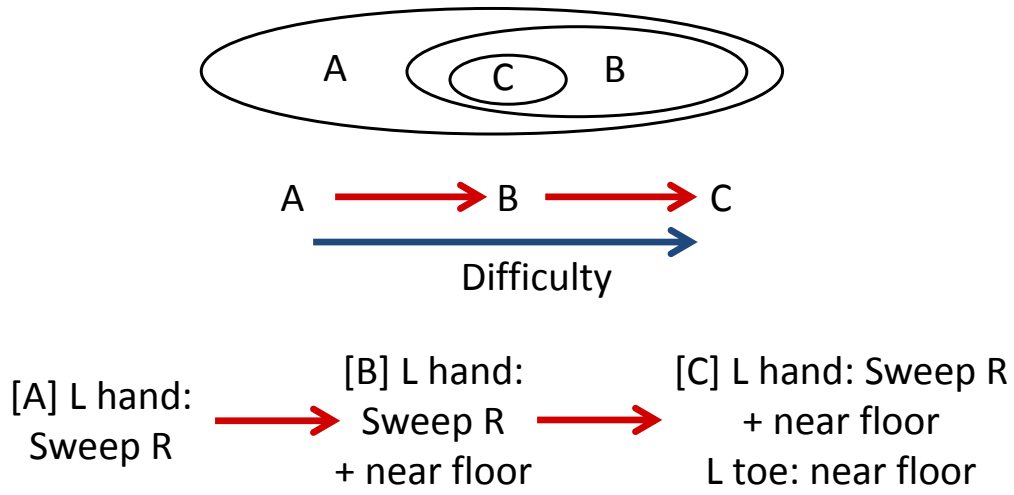

Figure 5.2: *Canonical Menu Action Hierarchy. A complex action can be expressed in a more simple form by dropping constraints.*

These actions of increasing complexity can be thought of as being related to one another in an abstract *action space* (Figure 5.2). Here, action *A* ("left arm sweep right") is the most general of the hierarchy and contains actions *B* (which adds "left hand near floor") and *C* (further adds "left toe near floor"). While action A can be easy to use to elicit a response from the SIPPC, it is action C that is actually closer to a crawling-like behavior. With typically developing infants, one expects to observe a progression from the "easier" actions to the more complicated ones. One long-term goal of this work is to lead an infant along this progression by differentially rewarding the actions in this hierarchy.

5.2 Template Matching

The canonical menu of hierarchical actions is represented as a partially ordered set. One would like to match complex movements before simpler movements. However, some actions in the canonical menu have similar difficulty or are equally complex. For example, bias between matching actions that make use of the right side of the body versus those that are made with the left should be avoided. Also, similar coordination is required for some arm versus leg actions. Actions of similar complexity, coordination, and expected usefulness should have even chances at triggering assistance (e.g. there should not be an "arms vs. legs" bias for the arms, or a "left vs. right limbs" bias). A hierarchical ordering is implemented through adding or dropping constraints, such as "near the floor." While making decisions about what gestures should trigger assistance, one can begin with the more complex actions. This ensures that more general actions such as "left arm push backwards" are not rewarded without considering the more coordinated "both arms push backwards" action.

A convenient approach for solving this problem is to consider the actions as a prioritized list of sets. By assigning priority to actions based on their difficulty, coordination, and expected utility, the actions are naturally arranged into groups with respect to hierarchy. The highest priority action that matches its criteria is selected. Priority ties are handled with random tie-braking, so that actions in the same difficulty class are considered with equal probability. This approach avoids biasing assists triggered by the left or right limbs. The partial action hierarchy used for this work is shown in Table 5.1. A complete list of actions and their priorities can be found in

Table C.1. If the three actions shown in Figure 5.2 were the only three being considered, one would first consider action C ($priority = 7$). If action C does not match then the next most general action (B in this case, with $priority = 5$) is considered. If action B does not match, action A ($priority = 2$) will be considered.

After receiving updated sensor orientations and limb positions, the first step to identifying gestures is to calculate displacements for points on limbs involved in actions found in the Canonical Menu (Section 5.1). Primitive boolean conditions are defined in order to represent whether or not these displacement measurements and other action criteria have been met. For example, one primitive condition could be defined as a displacement of $-0.05m$ along the Y dimension.

Because an individual primitive condition is often a component of multiple actions, each primitive condition is evaluated at each control cycle, as shown in Algorithm 1. The set of primitive conditions for a particular limb l are represented by $primitive\_conditions_l$. An action is *matched* if all of its component primitive conditions match, as shown in Algorithm 2. An array ($matched\_primitives$) is used for storing the results of each match. In order for an action to match a given template, every primitive condition specified by the template must be satisfied by the current state of the limbs. For example "left arm push backward with right foot near the ground" requires limb displacement as well as a limb position conditions to be true.

Once an action is found to match, meaning that all of the primitive conditions associated with that action are true, it generates an assistance with some probability p (Algorithm 3). A hysteresis period is used, so that no other action can occur for some amount of time following a match. In practice, I use a hysteresis period of

Algorithm 1 Primitive Condition Evaluation

```
function EVAL_CONDITIONS(limbs)
    for each limb l ∈ {limbs} do
        // Evaluate gesture primitives involving the current limb l
        for each c ∈ {primitive_conditions_l} do
            // Check a gesture direction, magnitude, or position criterion
            matched_primitives[c] ← criterion_met(c)
        end for
    end for
    return {matched_primitives}
end function
```

Algorithm 2 Template Matching

```
// Match templates from list of action sets, prioritized by difficulty (Table 5.1)
function MATCH_ACTIONS(matched_primitives)
    // Initialize matched_actions to all true
    for each a ∈ actions do
        matched_actions[a] = true
        // If any primitive conditions do not match, the action does not match
        for each c ∈ primitive_set(a) do
            matched_actions[a] ← matched_actions[a] &&
                                 matched_primitives[a]
        end for
    end for
    return {matched_actions}
end function
```

1.5 seconds and actions must satisfy the template conditions within 0.5 seconds. For example, if an action requires a limb displacement, the corresponding limb must achieve that displacement within 0.5 seconds.

It is currently possible for infants to degenerate into using only easy actions. One approach to addressing this problem is to reduce the probability of assistance corresponding to easy actions. Also, the amplitude is not different for easy vs hard variants

Algorithm 3 Gesture Based Assistance

```
while (1) do
    // Get New Sensor Orientations
    sensor_orientations ← update_sensor_buffer()
    // Update Forward Kinematics Model
    limbs ← update_infant_kinematics(sensor_orientations)
    // Evaluate Set of Boolean Primitive Conditions (Algorithm 1)
    matched_primitives ← eval_conditions(limbs)
    // Match action templates using a set of primitive conditions (Algorithm 2)
    matched_actions ← match_actions(matched_primitives)
    if (hysteresis is active) then return "no assist"
    else
        for each pr in priorities do
            actions ← {a | a ∈ actions(pr) && matched_actions[a]}
            if (actions ≠ ∅) then
                // activate hysteresis before returning an assistance
                hysteresis ← 1
                // pick a random action with the highest priority
                a ← pick_random(actions)
                if (with probability p(a)) then
                    return {assist with a}
                else
                    return {"no assist"}
                end if
            end if
        end for
    end if
end while
```

of actions that I have defined. It would be possible to have actions such as "sweep arm left" and "greater sweep arm left" to trigger different SIPPC displacements.

Action	Priority
feet near floor + hand near floor + arm sweep	7
sweep both arms in same direction, near floor	6
feet near floor + arm sweep	5
sweep both arms in same direction	4
pedaling with both feet	3
sweep with one arm	2
kick with one foot	1

Table 5.1: Action Hierarchies are represented by 7 different classes. The more complex actions are specified by multiple position and velocity constraints (Priority = 7). A complete list of actions and their priorities can be found in Table C.1.

Chapter 6

Results

6.1 Validation

There exist a number of sources of potential error in estimating the endpoint limb positions, including: errors in IMU orientation estimates, errors in the calibration process (e.g., not placing the limb in the proper calibration configuration), errors due to IMU filtering lags, and errors due to shift of the sensor from its calibrated position. The UM6 IMUs claim "accuracies" of two degrees in roll and pitch, and five degrees in yaw under static testing conditions, which can result in wrist position estimate errors of up to $3cm$. I examine the implications of these errors, and those of calibration and filtering using a physical model of the infant limb. The kinematics suit is mounted to a passive, five DOF arm. The reconstructed wrist positions are compared to those measured by a magnetic field-based localization sensor (Polhemus Patriot™).[1] This arm approximates the kinematics and scale of an infant's arm. The

[1]Within these controlled conditions, the Patriot exhibits positional repeatability on the order of $7 \times 10^{-6}m$

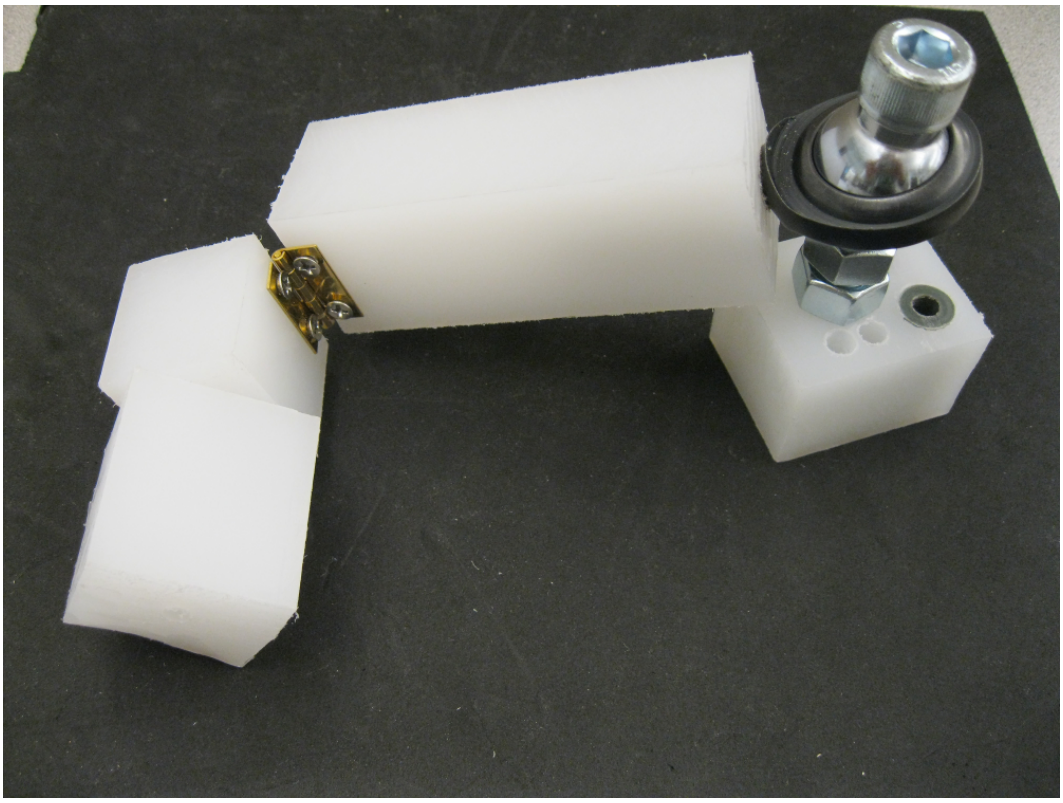

Figure 6.1: *Five DOF test bed. The arm is made out of low-density polyethylene (LDPE). There is a three DOF shoulder ball joint, one DOF elbow hinge joint, and one DOF modeling forearm twist.*

five DOFs consist of a ball joint for the shoulder, a hinge joint for the elbow, and a twist joint for the forearm.

In order to measure the kinematic reconstruction accuracy relative to the positions estimated by the Polhemus sensor, two calibration steps are necessary. The first is the standard calibration procedure, as required by Equation 4.2, in which relationships between sensed orientations and joint orientations are estimated. In the second calibration step, the base coordinate frame is aligned with that of the Polhemus sensor, and the relative transformations between subsequent coordinate frames (${}^{J_{k-1}}P_{J_k}$) are estimated. To solve for these parameters, I first collected a data set consisting of a

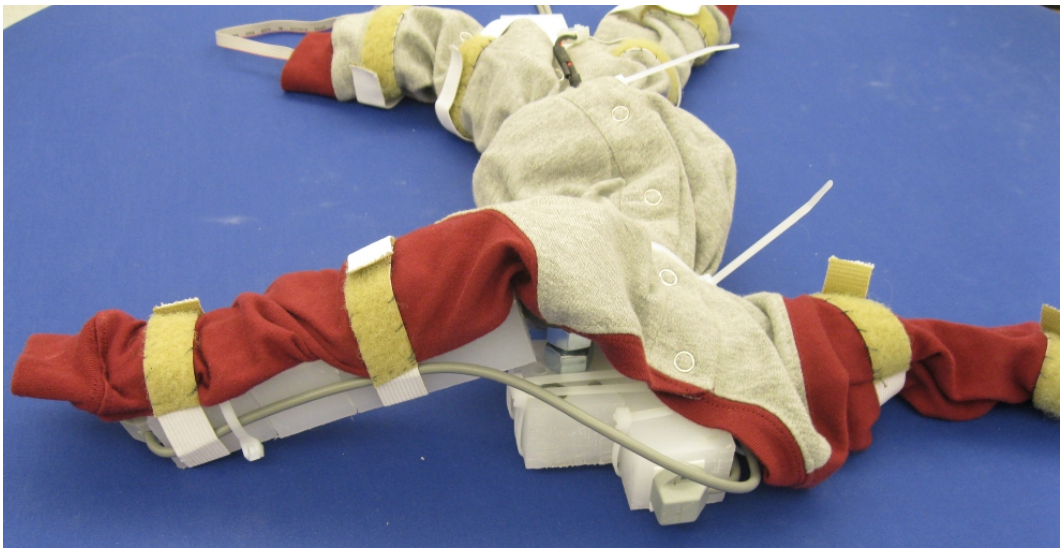

Figure 6.2: *Five DOF test bed with Kinematics Suit and two Polhemus sensors attached. Corresponding suit-based position estimates of the wrist are matched with Polhemus-based measurements as the 5-DOF arm workspace is sampled.*

time-series of matched Polhemus and IMU sensor states, while exploring the configuration space of the arm (1118 samples taken over 111.8 seconds). A line-search algorithm chooses a quaternion to represent the rotation between the patriot and suit, a three dimensional offset to align both origins, and a scaling factor for the length of the arm, so as to minimize the difference in the suit-based and Polhemus estimates of wrist position.

Following calibration, an additional 4473 matched samples (447.3 seconds) are collected as the arm was moved over the range of the workspace at a range of speeds. Figure 6.3 shows wrist positions as estimated by the kinematics suit and the Polhemus sensor over a 300 second period. The two estimates of position correspond well. However, a drift in the kinematic reconstruction can be seen at around 170 seconds. This departure lasts for approximately 30 seconds. However, the Kalman Filter then

"catches up" to the true state of the arm. These time-series are typical for the other axes, and for other data sets. Over the 447.3 second data set, average position estimate errors were $4.4 \pm 2.3cm$. Steady state inaccuracies of the shoulder IMU alone, as advertised, can account for approximately 0.03 cm of this error.

Because the Extended Kalman Filter is a causal filter, it necessarily induces some delay in the orientation estimates. A positive relationship between wrist speed and the kinematic-based position estimate error is expected. Figure 6.4 shows a weak, but significant correlation ($R = 0.17$, $p < 0.001$) between these variables. The increase in error as the wrist moves faster is minimal and is acceptable for this application.

Finally, I measure the calibration repeatability in order to assess the magnitude of errors associated with the manual placement of the infant's limbs into a target configuration. Over 30 independent manual placements of the limb, variation in wrist position, as measured by the Polhemus sensor, was $0.6 \pm 0.3cm$. This corresponds to a 1.3 ± 0.64 degree variation at both the shoulder and elbow joints.

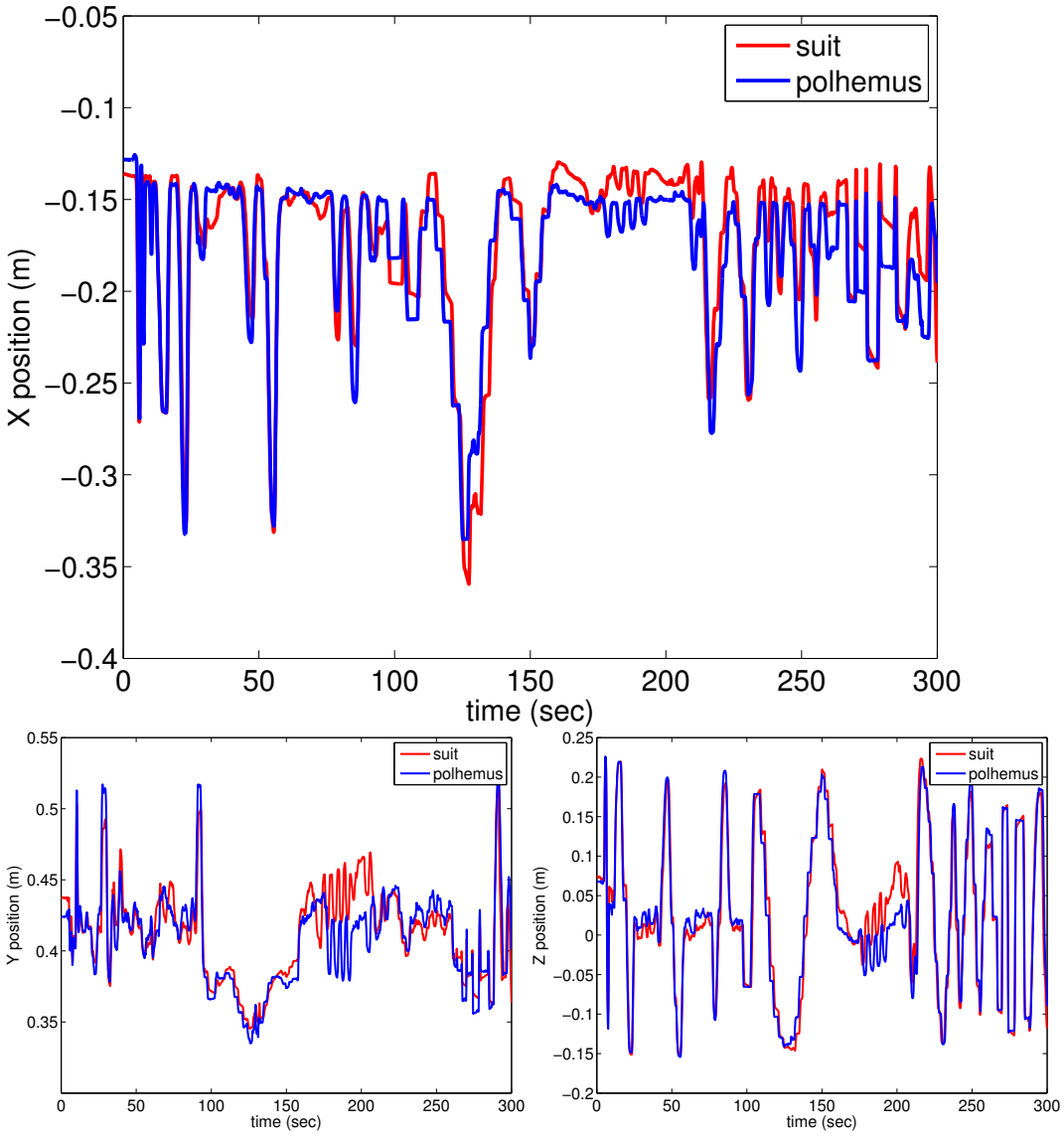

Figure 6.3: *Wrist Position after Patriot and Suit alignment, for the x,y and z axes. The reconstruction tracks the Patriot with little error. Sensor drift results in increased error in the kinematic reconstruction at around* 170 *seconds. Drift can be seen when reversing the direction of a sweep at high frequency (around* 170 *seconds), suggesting a similar error when the infant is shaking a limb.*

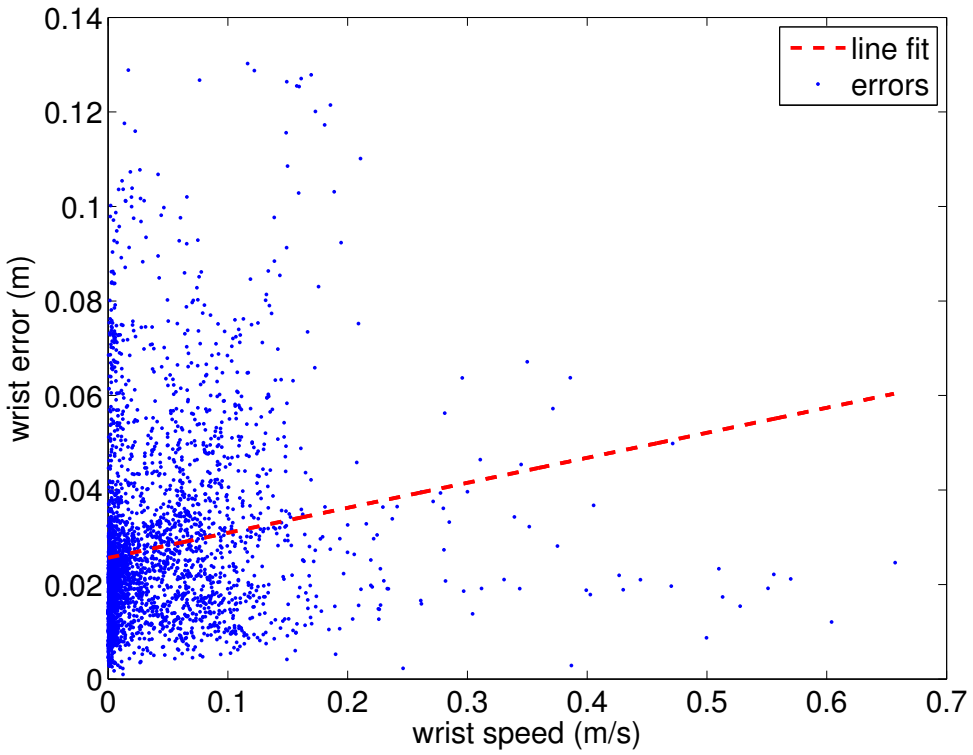

Figure 6.4: *Wrist error vs speed, showing weak correlation. Sensor orientation estimate errors increase with sensor velocity.*

6.2 Infant Experiments

This section reports on data obtained from four typically-developing infants, ages five to ten months, who participated in a larger study (OU-HSC IRB #15323). All subjects began the study at about five months of age. Twice per week, the subjects were fitted with the kinematics suit and placed on the SIPPC for up to fifteen minutes per session. During the session, the subjects were presented with a variety of toys in order to encourage attempts at producing goal-directed locomotion and reaching. Depending upon the day, the SIPPC operated in either a powered assist or a combination of powered and gesture assist modes. A subject's participation in the study was terminated after twelve weeks or when the subject reached a predetermined crawling activity criterion.

I hypothesize that the kinematics suit and SIPPC, together, are capable of measuring behavioral artifacts that parallel the development of crawling behavior in typically-developing children. In particular, I expect to be able to measure an increase in the infant's production of coordinated, multi-limb movements, and an increase in the infant's ability to locomote with the SIPPC.

Figure 6.5 shows a sequence of video frames with the corresponding kinematic reconstruction. Black lines correspond to the legs and feet; blue lines represent the upper back and arms. The red line connects the waist to the back, where the lower back is modeled as a ball joint. In this example, the infant has recently reached for a toy with the left arm (frame 1). The infant then performs a "push-up" on the SIPPC

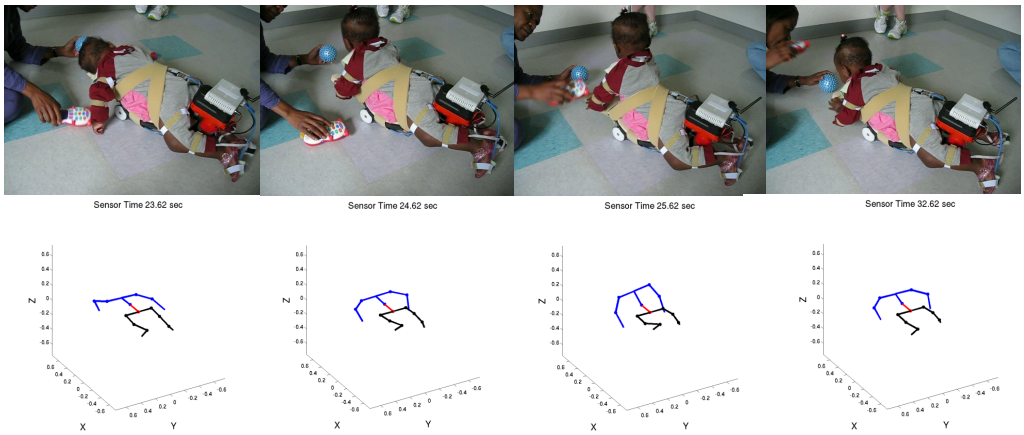

Figure 6.5: *Video frames (top) showing the infant and corresponding full-body kinematic reconstructions (bottom). The reconstruction shows our estimates of limb positions as well as orientation of the back for each frame. The infant can be seen performing a "push-up" on the SIPPC in the third frame, which can also be seen in the corresponding skeletal reconstruction figure.*

(frames 2 and 3), which is a common behavior for older subjects. The kinematic reconstruction also clearly captures the knee flexion in frame 3.

The SIPPC itself provides a means of measuring the activity of the infant. For example, Figure 6.7 summarizes these SIPPC-derived measures over a $300s$ time period for one subject during its last week in the study. For this particular session, the SIPPC provided powered assistance (responses to wheel movements caused by the infant). The linear distance along the path, over the $300s$, is well over $6m$ (top panel). The middle panel shows both the linear velocity and the times at which the SIPPC provided assistance, where assistance events are are indicated by black (pushing forwards/backwards) and red (turning left/right) bars. The bottom panel shows angular velocity. Many of the clusters of assistance events contribute substantially to progress along the path (the time period between 20 and 40 seconds is one such

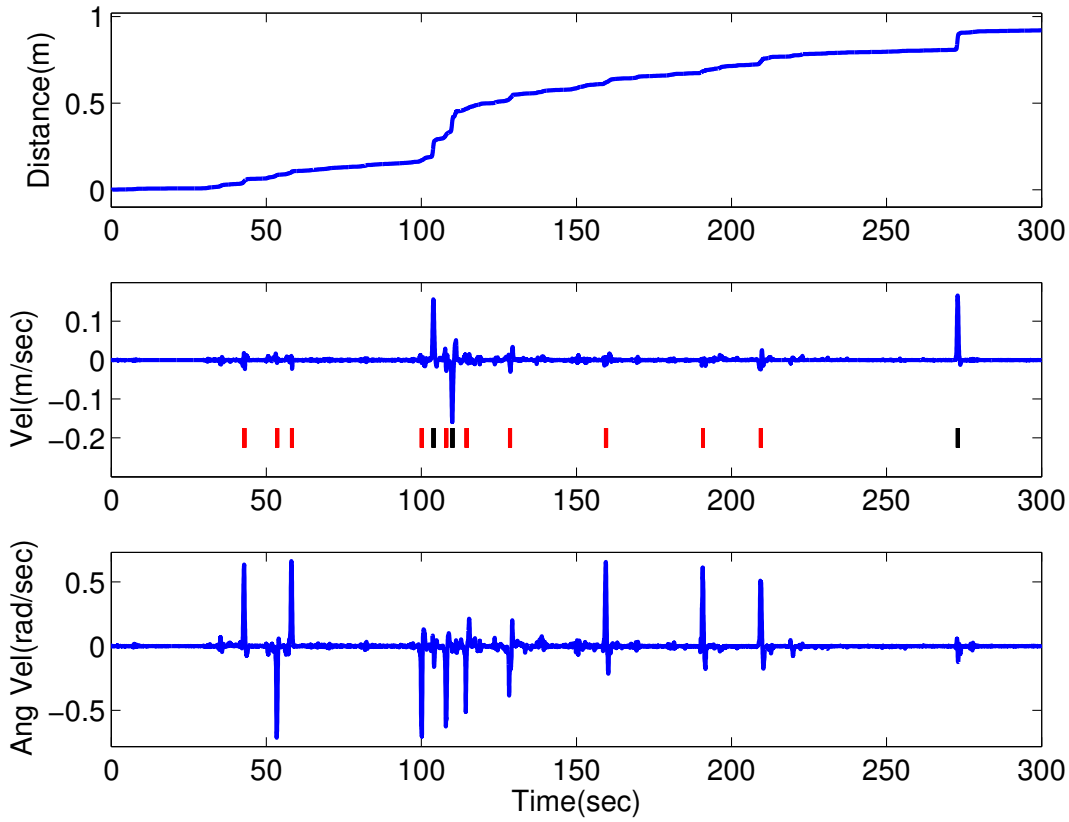

Figure 6.6: *SIPPC-based assessments of subject behavior (subject A, week 3). Top: total distance traveled as a function of time in the session; middle: linear velocity; and bottom: rotational velocity. The black bars indicate forward and backward powered assistance events. Similarly red bars indicate turning powered assistance events. The infant makes large translations around 100, 110, and 275 seconds. There are just over three times more rotation events than translation events. This figure is derived from Kolobe et al. (2012).*

case). These correspond to forward assistance and are shown as black bars. However, the SIPPC also produces assistance in turning (red bars). For example, there are several assistance events (marked in the time period of 130 to 140 seconds, but these manifest themselves in turning velocity peaks.

The kinematic reconstruction of limb endpoint positions gives us the ability to measure the activity level of individual limbs. For example, Figure 6.8 shows the

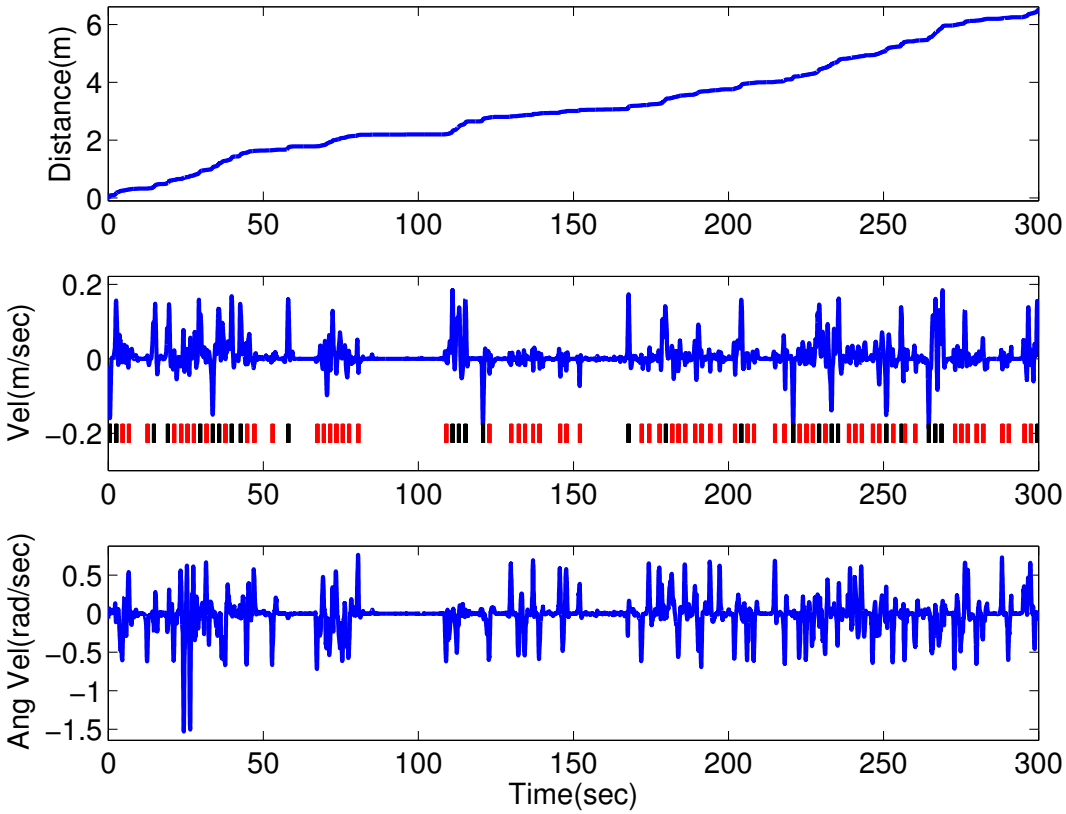

Figure 6.7: *SIPPC-based assessments of subject behavior (subject A, week 12). Top: total distance traveled as a function of time in the session; middle: linear velocity; and bottom: rotational velocity. The black bars indicate forward and backward powered assistance events. Similarly red bars indicate turning powered assistance events. The infant is much more active than in week 3 (Figure 6.6). Also, the infant has achieved a much greater path length. This figure is derived from Kolobe et al. (2012).*

total path length traversed by each limb endpoint (in the base coordinate frame) over the course of a $300s$ session, during which the SIPPC provided both powered and gesture-based assistance. In this particular session, the menu of canonical actions includes sweeps of the left and right hands in either the left or right direction. During this period of time, the right hand is most active. In general, it is not uncommon for the infants to prop their upper body up with one hand, while using the other hand to navigate and to reach. For example, subject C is shown during week 2 using their

right hand more (Figure 6.9) in one dataset, and later their left hand is used much more in the same day (Figure 6.10). Subject C is shown to increase the use of their left hand throughout the day, as their right hand is used approximately the same amount at both times.

Figure 6.8 shows assistance events that have been triggered by either the action of the left (*L-arm assist*) or right arm (*R-arm*). Each of these categories of actions can lead to either a left (purple) or right (red) turn of the SIPPC, depending on the direction of sweep by the arm. The highest density of left arm triggered assists occurs around $40s$. This time period also corresponds to the time of highest left arm activity, as indicated by the derivative of the pink curve at this time. Likewise, many of the most active periods of the right arm correspond to clusters of right arm triggered assists.

Another perspective of the recognized gestures is shown in Figure 6.11. Each curve shows the hand and feet trajectories in the base coordinate frame during the $500ms$ leading up to a right arm triggered assist. The approximate location of the SIPPC is indicated using dotted lines. The four clusters of curves, shown around the SIPPC, represent the trajectories of each hand and foot during the matched actions (e.g., the upper left cluster corresponds to the left arm). The skewing of the cluster locations is indicative that the infant is reaching off the front-right corner of the SIPPC. The trajectories are colored consistently with Figure 6.11, where red indicates a right turn assist and purple indicates a left turn assist. The state of each limb endpoint at the time of the assist event is indicated with a circle. Right turns are generated by pushing the right hand to the left, which are shown by a red curve starting with a dot

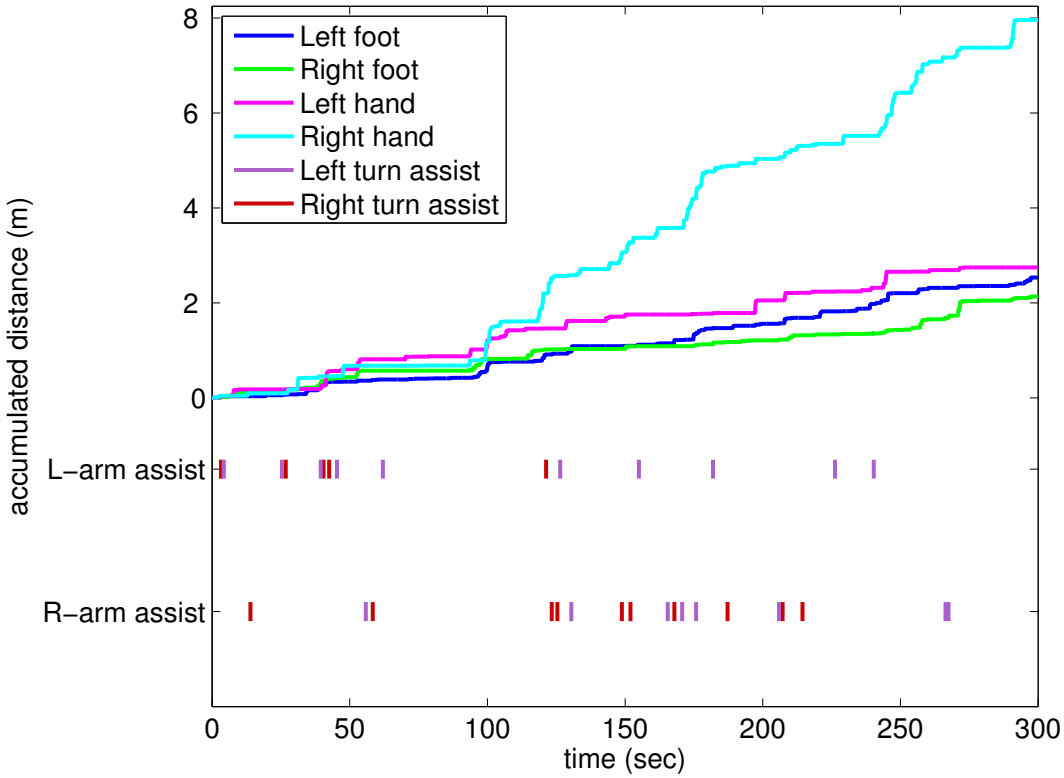

Figure 6.8: *Path length traversed by each limb endpoint (in the base coordinate frame) during real-time assistance (subject B, week 11). The right hand is moving much farther, as shown by the cyan line. The bars indicate left or right arm triggered assistance, and a turn to either the left (red) or right (purple).*

and moving left. Similarly the same hand is pushed to the right in order to turn left, which are shown with a purple curve moving from left to right. The left hand and both feet move in inconsistent ways prior to these events, indicating that the action of these limbs (in this case) is not well correlated with that of the right hand. The opposite behavior is shown in Figure 6.12, where both feet can be seen to be moved in coordination as the infant turns left (purple).

Not only do the SIPPC and kinematics suit allow us to examine the behavior on a session-by-session basis, but they also allow us to explore changes in behavior with

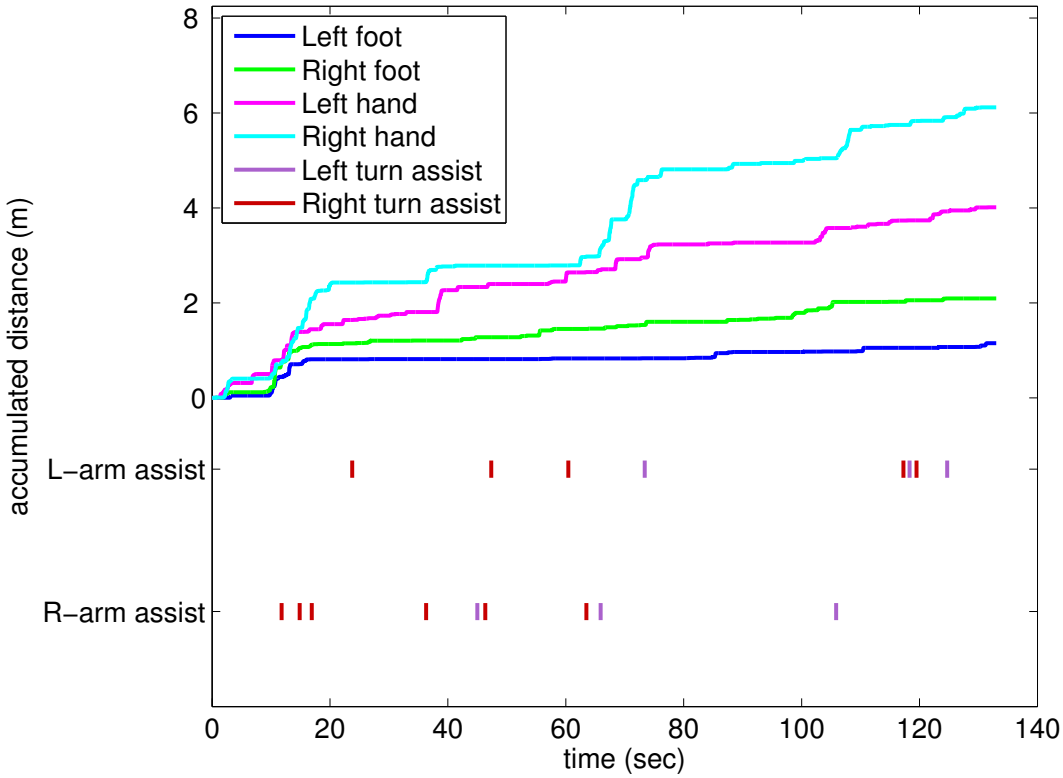

Figure 6.9: *Path length traversed by each limb endpoint (in the base coordinate frame) during real-time assistance (subject C, week 2). The right hand is farther, as shown by the cyan line. This is an example of how arm use patterns can change rapidly, as this figure corresponds to the same infant and day as in Figure 6.10. The bars indicate left or right arm triggered assistance, and a turn to either the left (red) or right (purple).*

development. Because the different subjects participated in the study for different periods of time, the data from the subjects are aligned on the time at which each subject met the study completion criteria, where week zero corresponds to each subject's last week in the study. Also, in order to account for session-to-session variation in the overall level of activity of each infant, only the most active session for each subject for each week was selected. Each fifteen minute session is split into three data sets, spanning five minutes each. One data set, with the most activity out of the three for the chosen day, is selected so that five continuous minutes can be analyzed from each

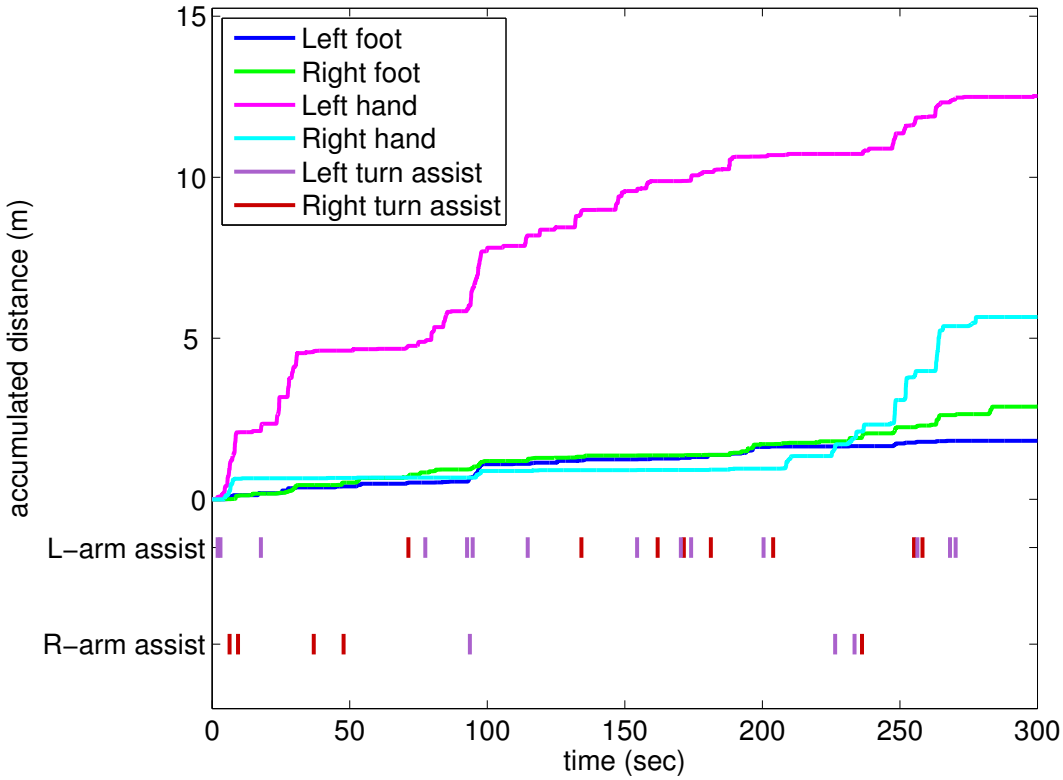

Figure 6.10: *Path length traversed by each limb endpoint (in the base coordinate frame) during real-time assistance (subject C, week 2). The left hand is moving much farther, as shown by the magenta line. This in contrast to the right hand being preferred earlier in the day (Figure 6.9). The bars indicate left or right arm triggered assistance, and a turn to either the left (red) or right (purple).*

week. For example, Figure 6.13 shows the average linear speed of the SIPPC as a function of week within the study. Each point shows the mean speed and standard deviation. Note that weeks 10 and 11 only contain a single sample. Average speed lumped over weeks 0-2 is significantly higher than weeks 3-5 and 6-8, according to a two-sample t-test with Bonferroni correction ($p < .01$).

Another metric that is particularly important is how often the infant elicits different actions from the canonical menu. Because the gesture-based assistance was not engaged for all sessions in this study, this question can only be addressed through a post-hoc analysis. Figure 6.14 shows left vs right "easy" arm action counts for

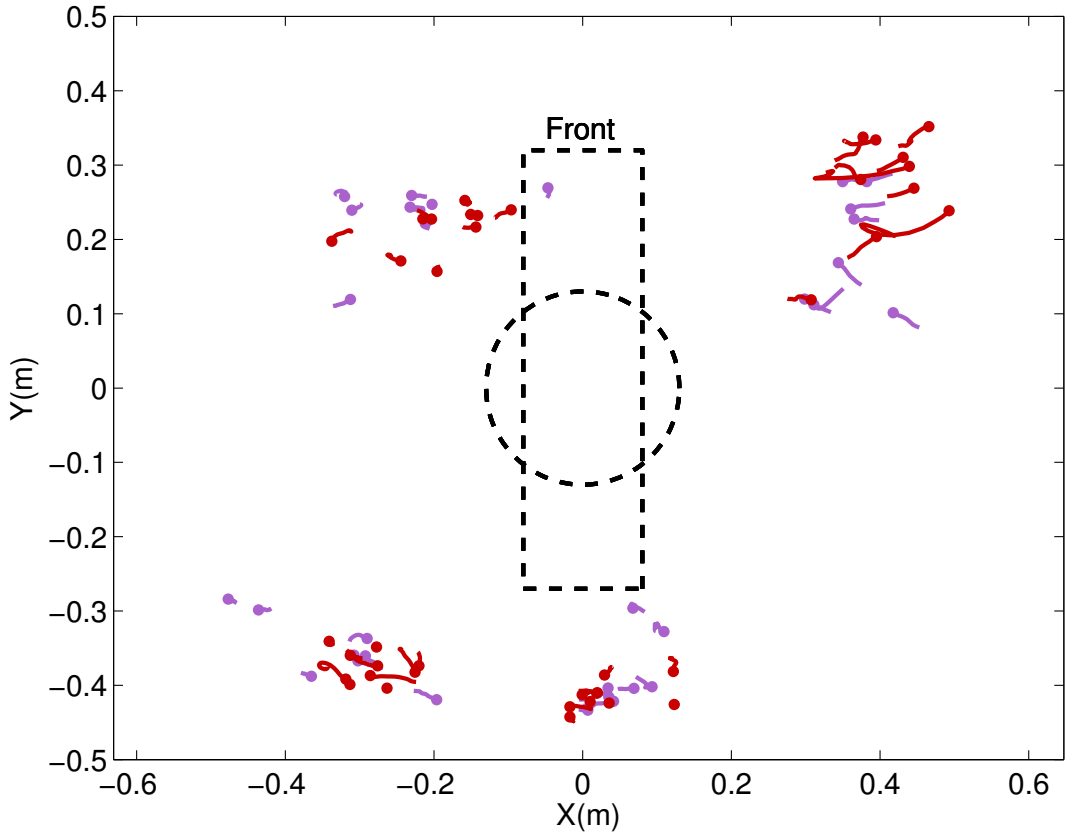

Figure 6.11: *Positions of the hands and feet relative to the SIPPC during periods of time leading up to a right arm triggered assistance event (subject B, week 11). The circles indicate position of the limb at the time of the event. The approximate footprint of the SIPPC is shown with dotted lines and the four clusters of red and purple points correspond to positions of each limb. Right turns are generated by a push to the left (red curves). Similarly, left turns are generated by a push to the right (purple curves).*

subject B as a function of week until completion. Care was taken so that no limb movement was counted as multiple actions for this analysis. Over the ten weeks of available data, this infant did not show a strong bias towards using either their left or right arms. However, the balance varied from week to week.

Similarly, Figure 6.15 shows the frequency of recognition of two different classes of gestures, for all of the subjects over the duration of the study. Here, the actions

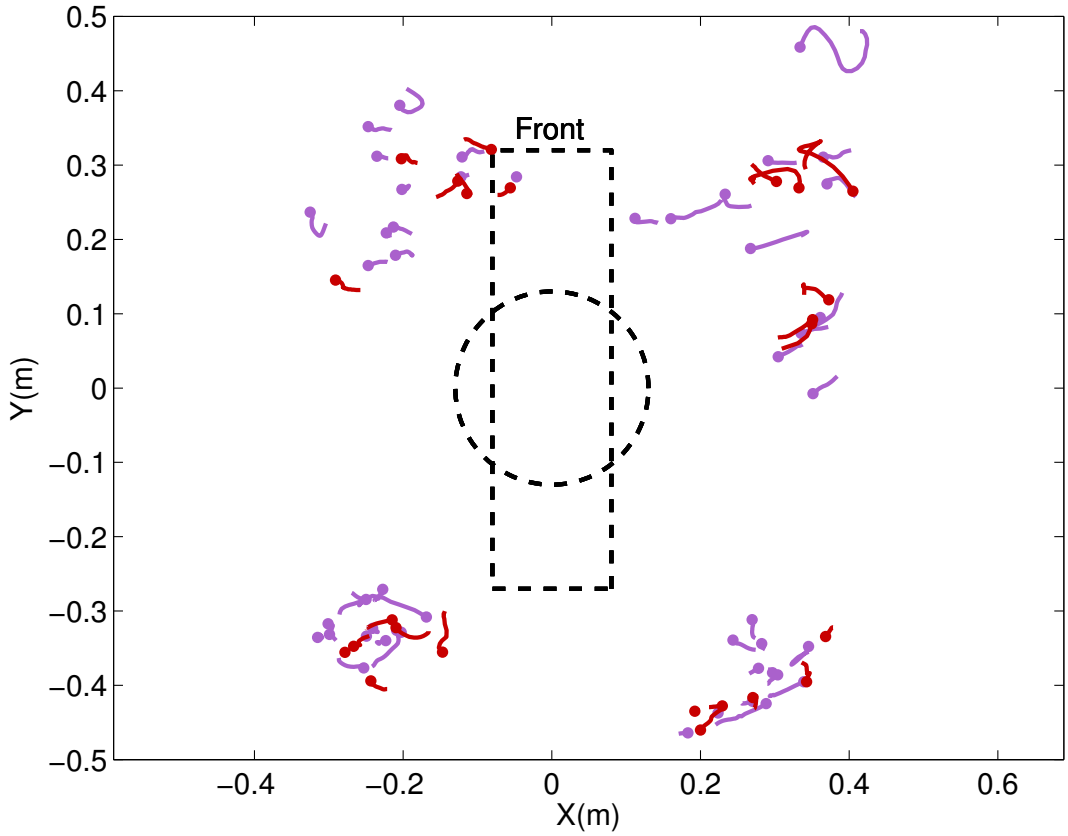

Figure 6.12: *Positions of the hands and feet relative to the SIPPC during periods of time leading up to a right arm triggered assistance event (subject C, week 2).*

in the menu were partitioned into sets of "easy" and "difficult" actions based on how many criteria were used for specifying the movement templates. Difficult actions required limbs to be near the floor, coordination between multiple limbs, or both. In general, the frequency of easy actions was higher than that of hard actions. The frequency of easy actions in weeks 6-8 is not significantly different than in weeks 3-5 (t-test, with Bonferroni correction, $p < 0.73$), but is significantly different from weeks 0-2 ($p < 0.02$). Also, the frequency of hard actions in weeks 6-8 is is significantly different than weeks 0-2 ($p < 0.001$). The infants are using the more complex actions with higher frequency near the end of the study.

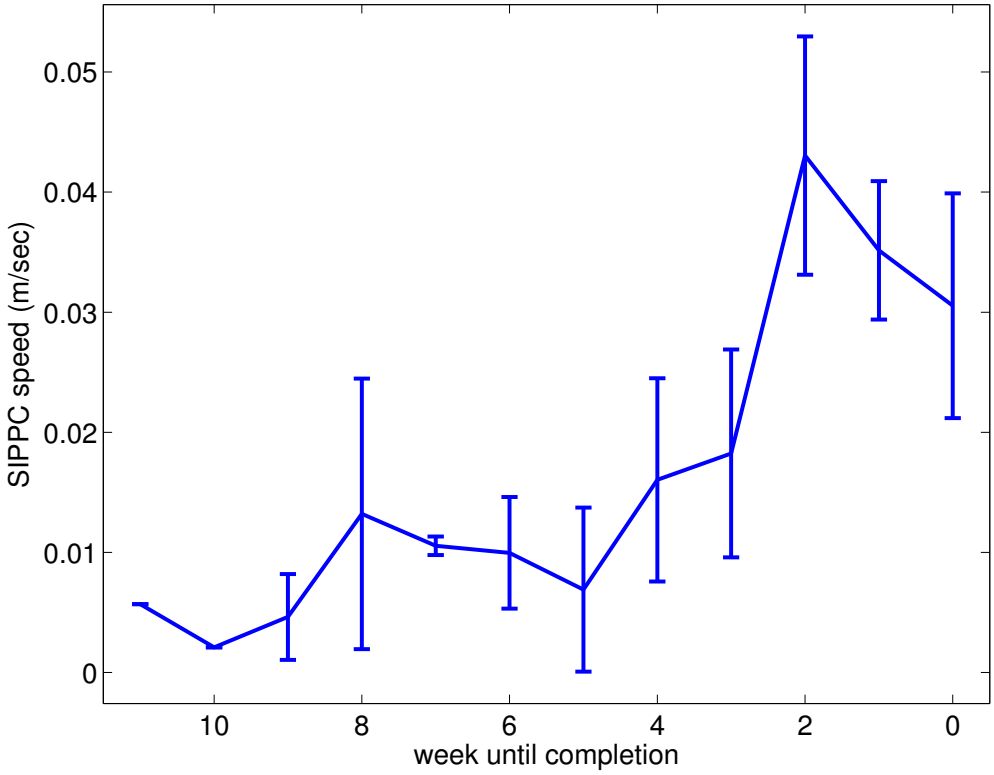

Figure 6.13: *Mean and standard deviation of linear speed of the SIPPC as a function of study week. Each point includes data from up to four infants. Because the different subjects participated in the study for different lengths of time, these data are aligned by their last week in the study.*

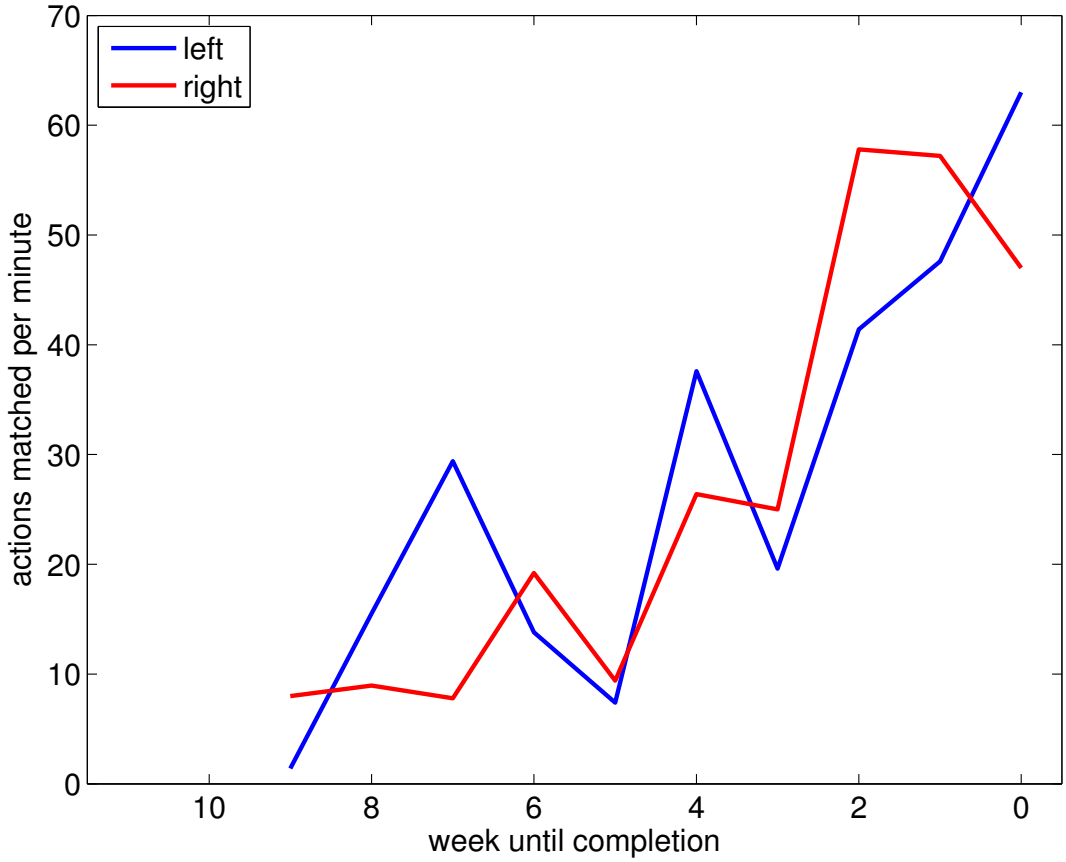

Figure 6.14: *Left vs right arm easy action counts for subject B. The infant is not strongly biased towards using either arm*

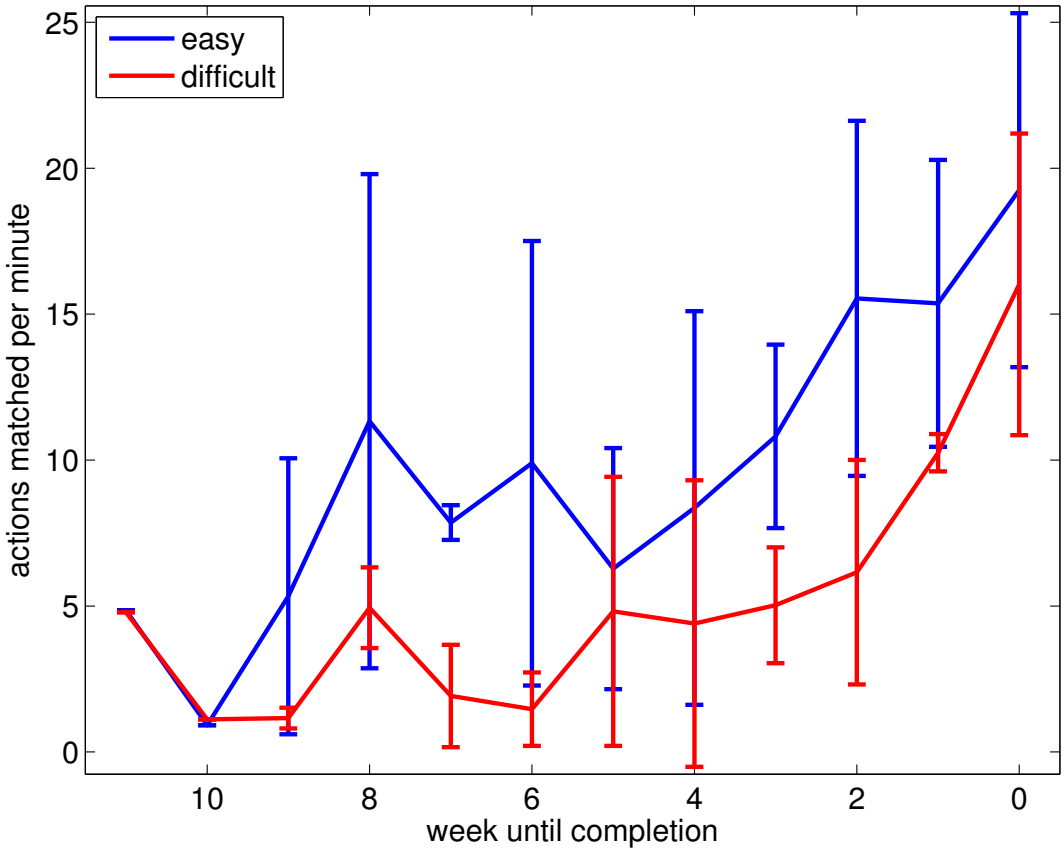

Figure 6.15: *Mean and standard deviation of recognized action frequencies as a function of study week. The actions in the menu have been partition into easy and hard action.*

Chapter 7

Conclusion

In this thesis, I propose an inertial measurement unit kinematics suit that enables the sensing of an infant's trunk and limb configuration in real-time. I also propose a method for recognizing classes of limb gestures. Finally, I show that recognized gestures can be used to trigger assistive movements of the Self-Initiated Prone Progression Crawler (SIPPC) robot in real-time.

The experiments with infants show that both the SIPPC and the kinematics suit can be used to quantify the activities of the infant within individual sessions, and over the course of many weeks. In particular, the findings show that over an eight to twelve week period leading up to the onset of unassisted crawling, changes in behavior can be measured with both the SIPPC and the kinematics suit. Through the gestural recognition that is enabled by the kinematics suit, the findings show that overall action frequency has increased, including complex, coordinated movements. Practicing these difficult movements may help infants transition to unassisted crawling.

Previous studies with the SIPPC have suggested that the use of this robot by typically developing infants, while using powered assist mode, can lead to a reduction in the onset time for independent SIPPC control (Catalino et al., 2012). The study reported here does not address this question. Nevertheless, it is our hope in future work to show that gesture-based assistance can reduce the onset time of crawling in typically developing infants, as well as infants with CP.

This work can be extended in many ways. First, a semi-supervised learning technique could be employed to construct the canonical menu of actions automatically from observations. These observations could be derived from typically developing infants during the development of crawling behavior. An automatically learned menu of actions could be more general, as well as more precisely tuned to common movements made by infants. Specifically, such an approach would cluster limb and trunk movements that lead to the passive movements of the SIPPC. These clusters would then be generalized to construct the hierarchy of canonical actions.

Second, canonical movements can be differentially rewarded depending on the prior successes of the infant. As "easy" canonical movements become commonplace, the probability of responding with an assistive SIPPC movement will be reduced, thus encouraging the infant to produce "harder" canonical movements. Through this process, the hope is to lead infants along developmental trajectories that will set them up for success when they attempt unassisted crawling.

Third, all components of the kinematic model can be automatically calibrated. Specifically, automatic calibration of the relative transformations from one coordinate frame to the next (${}^{J_{k-1}}P_{J_k}$) can be performed, as described in Chapter 4. Automatic calibration will be accomplished by minimizing discrepancies between sensed linear accelerations and accelerations estimated from the kinematic model.

Bibliography

Anderson, P., Doyle, L. W., and the Victorian Infant Collaborative Study Group (2003). Neurobehavioral outcomes of school-age children born extremely low birth weight or very preterm in the 1990s. *JAMA*, 289(24):3264–3272.

Bhushan, V., Paneth, N., and Kiely, J. L. (1993). Impact of improved survival of very low birth weight infants on recent secular trends in the prevalence of cerebral palsy. *Pediatrics*, 91(6):1094–1100.

Campos, J. J., Anderson, D. I., Barbu-Roth, M. A., Hubbard, E. M., Hertenstein, M. J., and Witherington, D. (2000). Travel broadens the mind. *Infancy*, 1(2):149–219.

Catalino, T. (2011). Development of infant prone progression using an assistive device. Master's thesis, University of Oklahoma Health Sciences Center.

Catalino, T., Kolobe, T., McEwen, I., and Fagg, A. H. (2012). Development of prone locomotion in infants using an assistive device. In *Proceedings of the Combined Sections Meeting Conference of the American Association of Physical Therapy (poster abstract)*, Chicago, IL.

Chen, X., Lang, S., Dolph, S., Ragonesi, C. B., Galloway, J. C., and Agrawal, S. K. (2010). Design of a novel mobility interface for infants on a mobile robot by kicking. *Journal of Medical Devices*, 4(3):1–5.

Conover, M. S. (2003). Using accelerometers to quantify infant general movements as a tool for assessing motility to assist in making a diagnosis of cerebral palsy. Master's thesis, Virginia Tech.

Dillon, T. E. (2005). Feasibility of assessing an infants general movements using wireless accelerometers for early diagnosis of neurological dysfunction. Master's thesis, Virginia Tech.

Gafurov, D. and Snekkenes, E. (2009). Gait recognition using wearable motion recording sensors. *EURASIP Journal on Advances in Signal Processing*, 2009:1–16.

Gibson, D. and Harris, A. (1988). Aggregated early intervention effects for down's syndrome persons: patterning and longevity of benefits. *Journal of Intellectual Disability Research*, 32(1):1–17.

Glardon, P., Boulic, R., and Thalmann, D. (2004). PCA-based walking engine using motion capture data. In *In the Proceedings of the Computer Graphics International*, pages 292–298.

Hadders-Algra, M. (2001). Evaluation of motor function in young infants by means of the assessment of general movements: A review. *Pediatric Physical Therapy*, 13(1):27–36.

Herda, L., Fua, P., Plänkers, R., Boulic, R., and Thalmann, D. (2000). Skeleton-based motion capture for robust reconstruction of human motion. *Computer Animation(CA)*, pages 77–83.

Kolobe, T. and Pidcoe, P. (2007). Movement strategies used by infants with cerebral palsy and down syndrome during self-initiated prone progression. In *Proceedings of the World Confederation of Physical Therapy (poster)*, Vancouver, Canada.

Kolobe, T. H. A., Pidcoe, P., McEwen, I., Pollard, V., and Truesdell, C. (2007). Self-initiated prone progression in infants at risk for cerebral palsy. *Pediatric Physical Therapy*, 19(1):93–94.

Lai, Y.-x., Shu, L., Vasilakos, A., Rodrigues, J. J. P. C., Lai, C.-f., and Huang, Y.-m. (2011). 3d adaptive reconstruction of human motion from multi-sensors. *System*, pages 531–536.

Mahoney, G., Robinson, C., and Fewell, R. R. (2009). The effects of early motor intervention on children with down syndrome or cerebral palsy: A field-based study. *Journal of Developmental & Behavioral Pediatrics*, 22:153–162.

Nilholm, C. (1996). Early intervention with children with down syndrome past and future issues. *Down Syndrome Research and Practice*, (4):51–58.

Olney, S. J. and Wright, M. J. (2006). Cerebral palsy. In Campbell, S. K., Vander Linden, D. W., and Palisano, R. J., editors, *Physical Therapy for Children*, pages 625–664. Elsevier, Inc., St Louis, MO.

Rule, B. A. (2010). Motor strategies implemented by infants using the self-initiated prone progression crawler. Master's thesis. Unpublished.

Schoepflin, Z. (2010). Pediatric, bio-driven, mobile-assistive devices and their effectiveness in purposeful driving for typically-and atypically-developing toddlers. Master's thesis, University of Delaware.

Shi, G., Zou, Y., Jin, Y., Cui, X., and Li, W. (2009). Towards HMM based human motion recognition using mems inertial sensors. In *the Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1762–1766.

Southerland, J., Fagg, A. H., Miller, D. P., and Kolobe, T. (2012). Activity recognition and crawling assistance for infants with cerebral palsy. *IEEE Robots & Automation Magazine, Special Issue on Assistive Robotics*. submitted.

Ulrich, D. A., Ulrich, B. D., Angulo-Kinzler, R., and Yun, J. (2001). Treadmill training of infants with down syndrome: Evidence-based developmental outcomes. *Pediatrics*, 108(5):1–7.

Appendix A

Schematics & Board Layouts

Custom printed circuit boards (PCB's) were created for this study. These include a breakout board for the UM6 orientation sensor, which allows for SPI daisy chaining. Also, a carrier board for the mBed was designed to allow for five SPI daisy chains and an Ethernet connection. Schematics and board layouts produced using Cadsoft Eagle CAD are provided below, as well as photographs.

A.1 UM6 Breakout Board

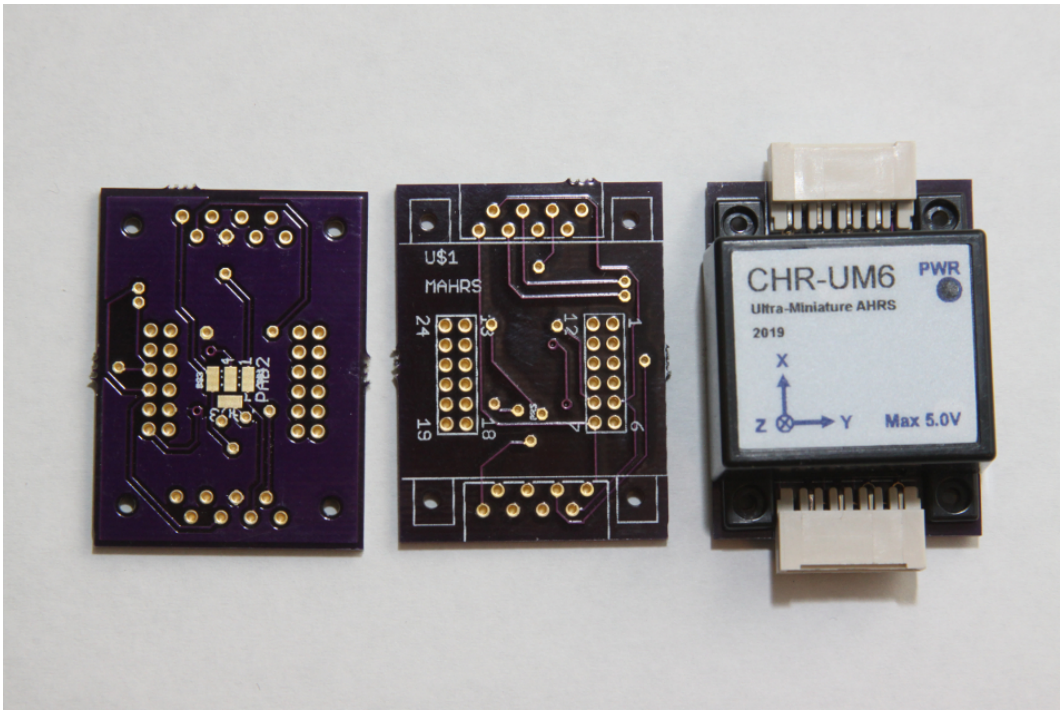

Figure A.1: *UM6 SPI-breakout board.*

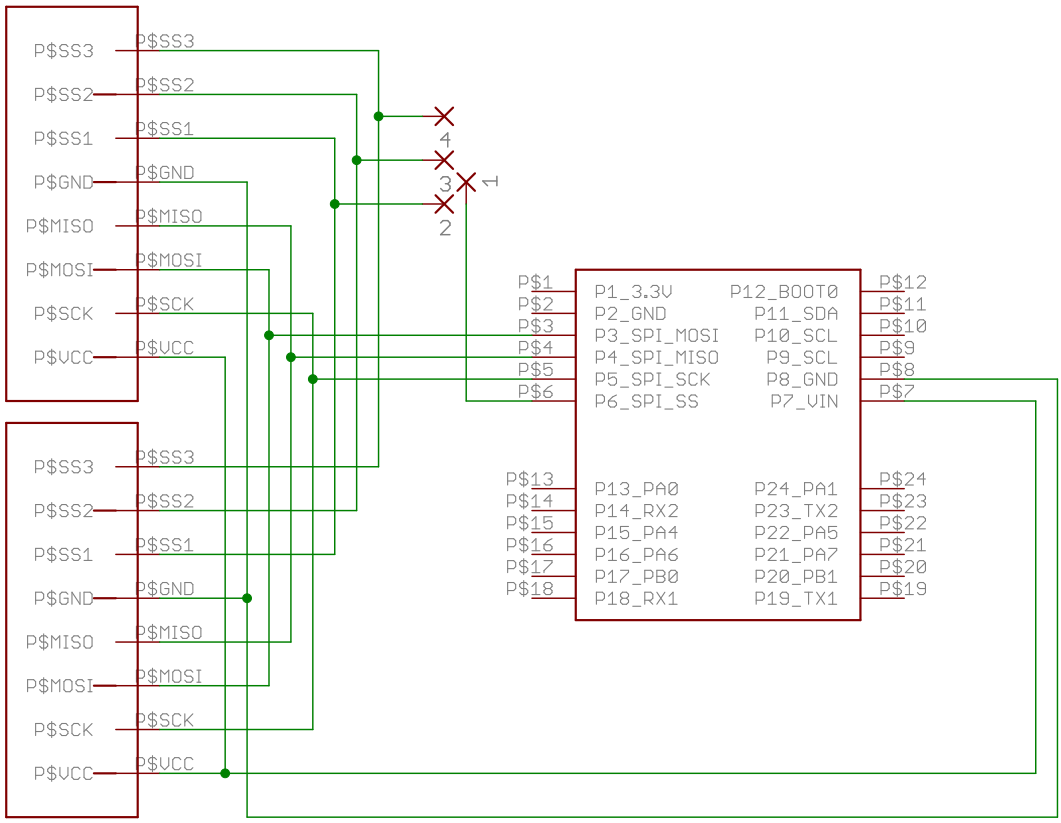

Figure A.2: *UM6 SPI-breakout board schematic. There are two connectors, allowing for the sensors to be daisy-chained. Each eight-wire cable carries power, ground, three SPI data signals, and three SPI slave-select signals.*

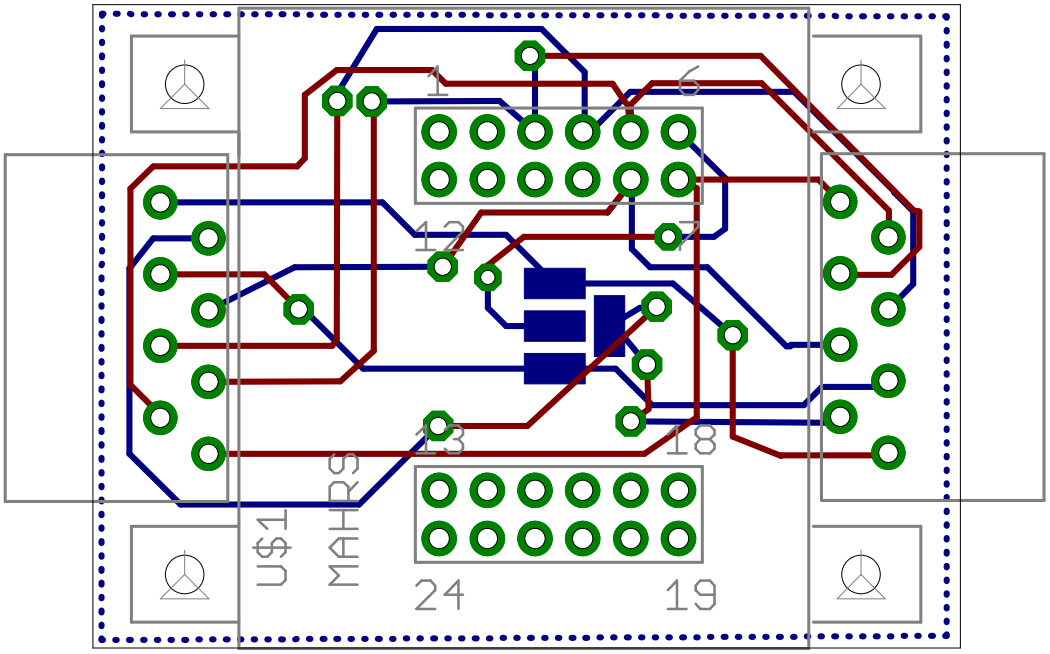

Figure A.3: *UM6 SPI-breakout board layout.*

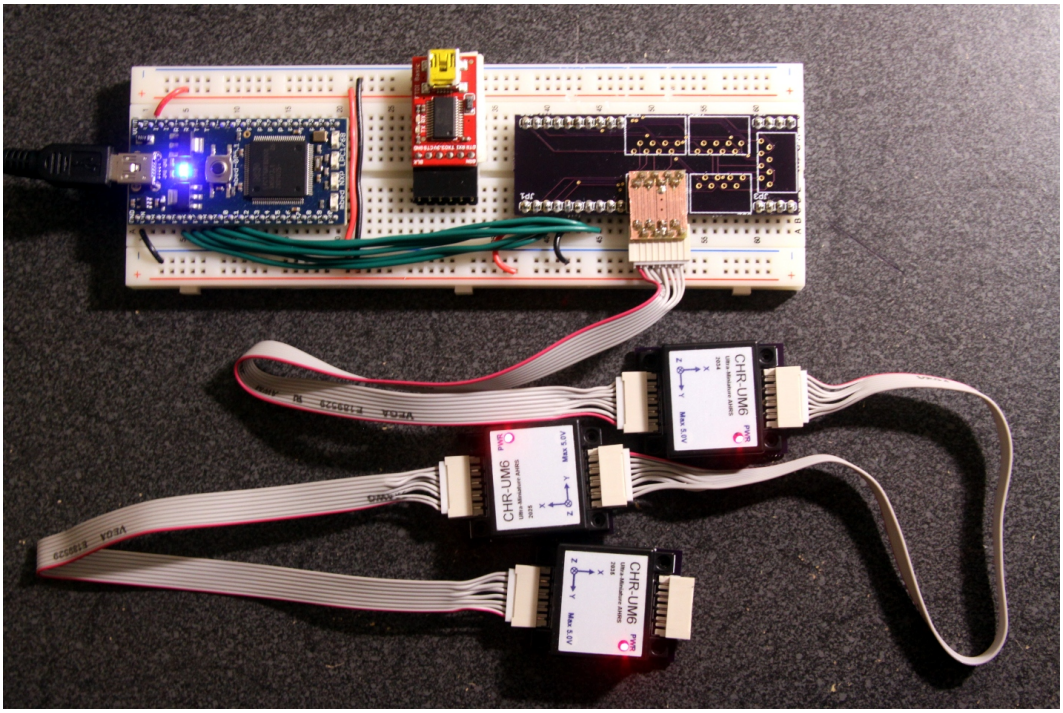

Figure A.4: *Early prototype showing how three UM6s are daisy chained via SPI to a mBed(blue board). The FTDI serial-usb conversion board (red) was not being used at the time this picture was taken and is no longer needed. This picture also shows a custom SPI hub (purple) allowing for five SPI daisy chains of three sensors each to be connected, which was worked into the current mbed carrier board (See Figures A.5, A.6, and A.7)*

A.2 mBed Carrier Board

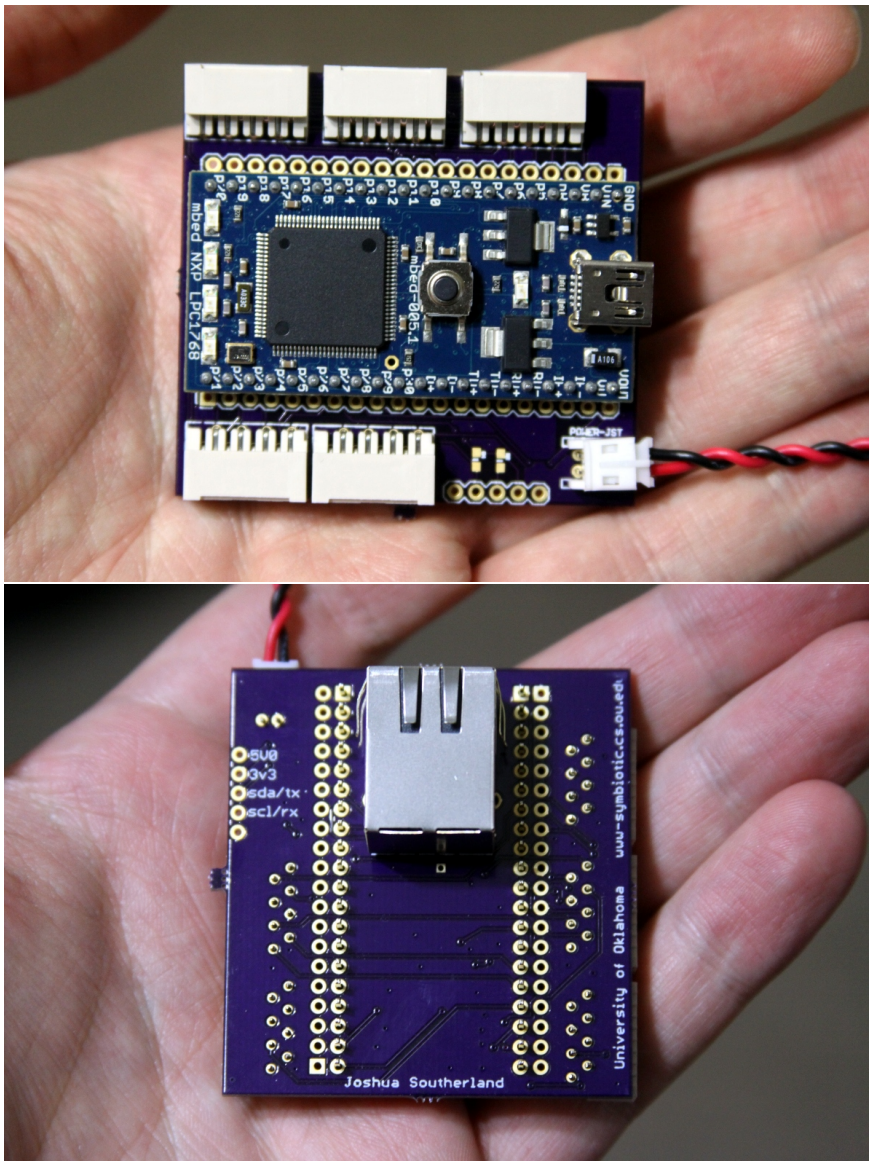

Figure A.5: *Pictures of the mBed carrier board. These pictures were taking while test fitting the components, so they are not shown soldered into place.*

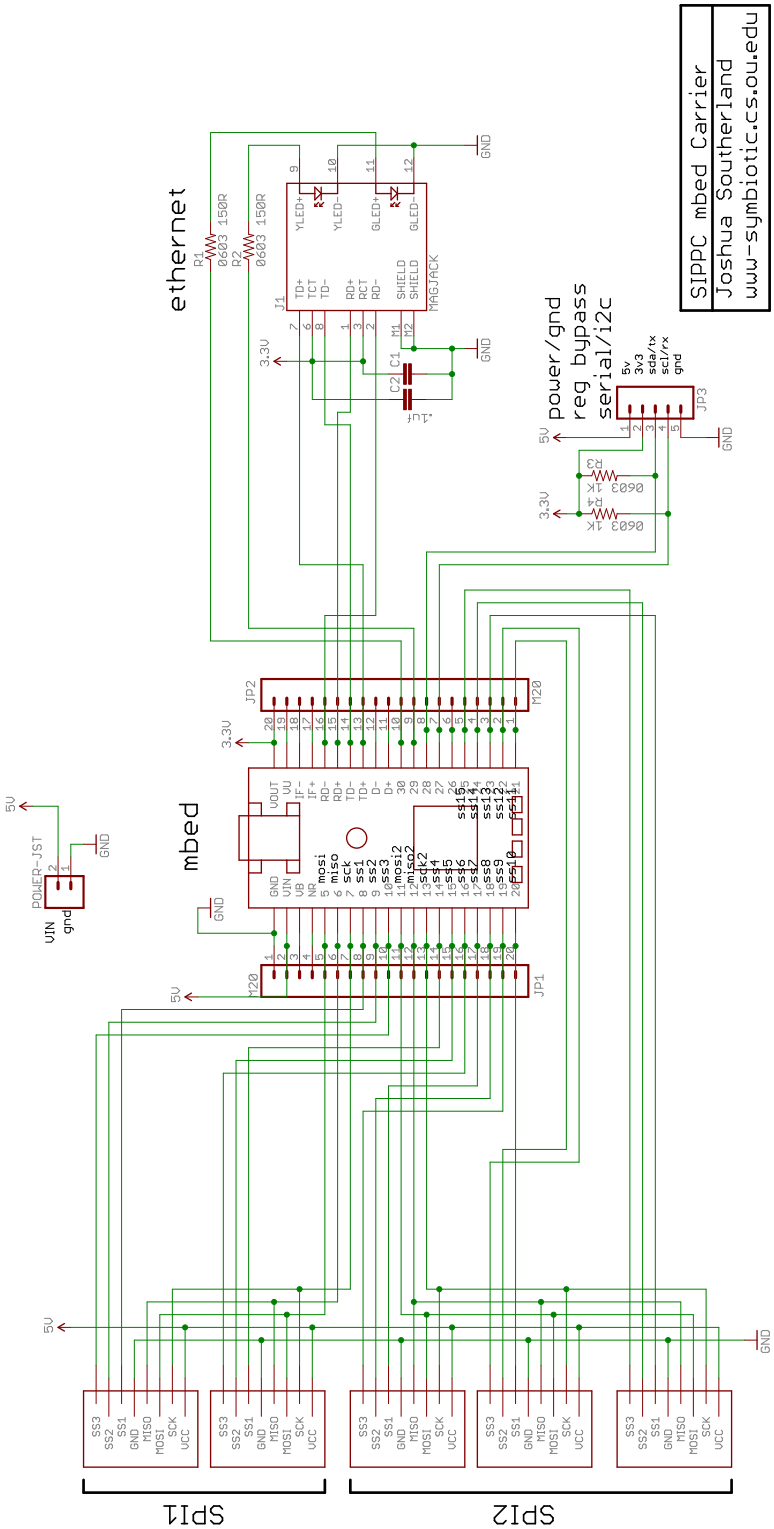

Figure A.6: *Schematic for the mBed carrier board.*

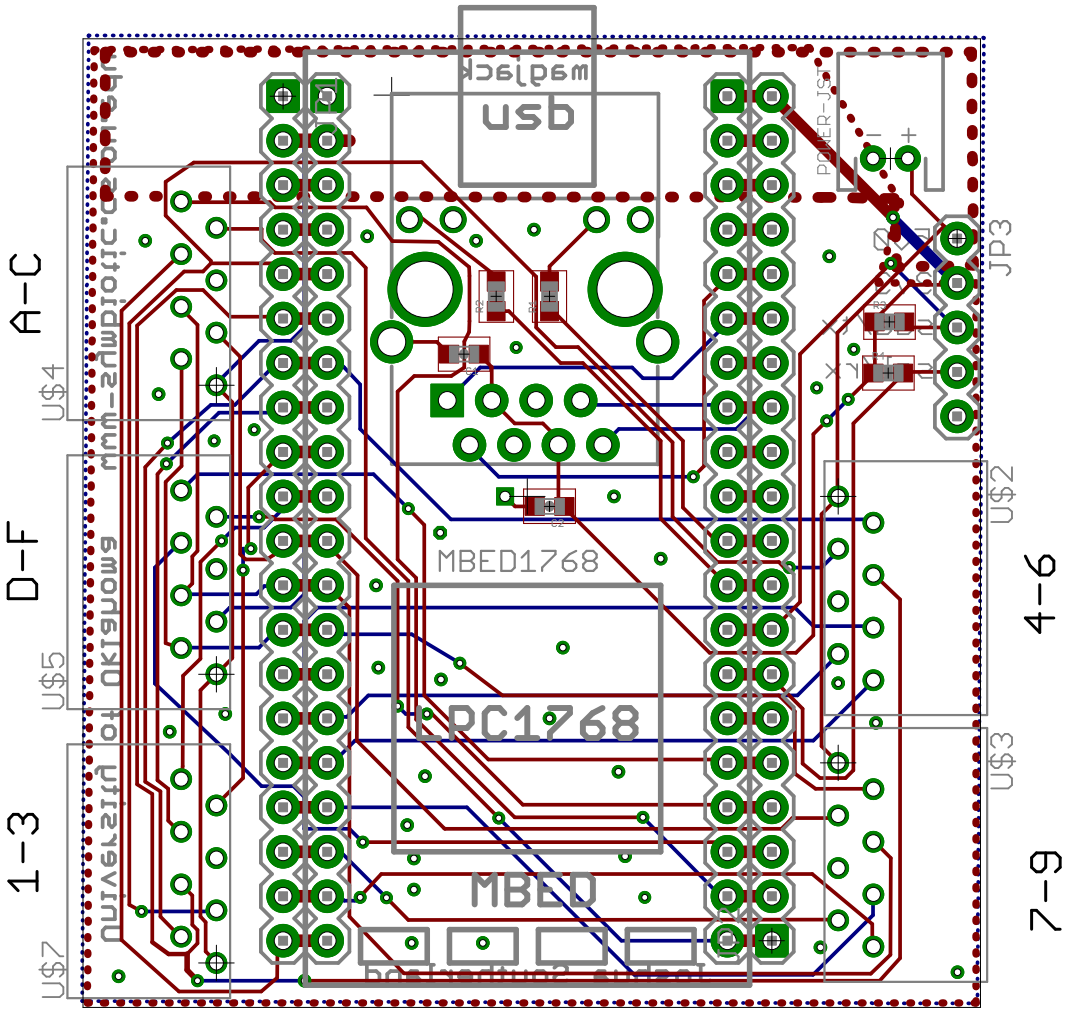

Figure A.7: *Layout for the mBed carrier board. This board connects five chains of sensors using two SPI ports on the mBed's ARM7 processor. Also, a Magjack is included to allow for communication via ethernet.*

Appendix B

Data Collection Process

B.1 Using the Graphical User Interface

The data collection process has evolved over the study, from using a simple command line interface in the beginning, to a more friendly graphical user interface (GUI) shown in Figure B.1. The steps for collecting data with the GUI are as follows:

1. Replace battery with a fresh one
2. Connect the suit power and ethernet cable
3. Turn the SIPPC on
4. Turn on the laptop program / gui
5. Press connect in the gui
6. Calibrate the suit in the gui
7. Press “start log” and a “recording” message should show up
8. **data collection**
9. Press “stop log”
10. Press “disconnect”
11. Remove the suit from the infant
12. Turn the laptop off and pack everything up

Note: I typically stop data collection after five minutes and started recording again, so that 15 minutes of data are split into three separate files.

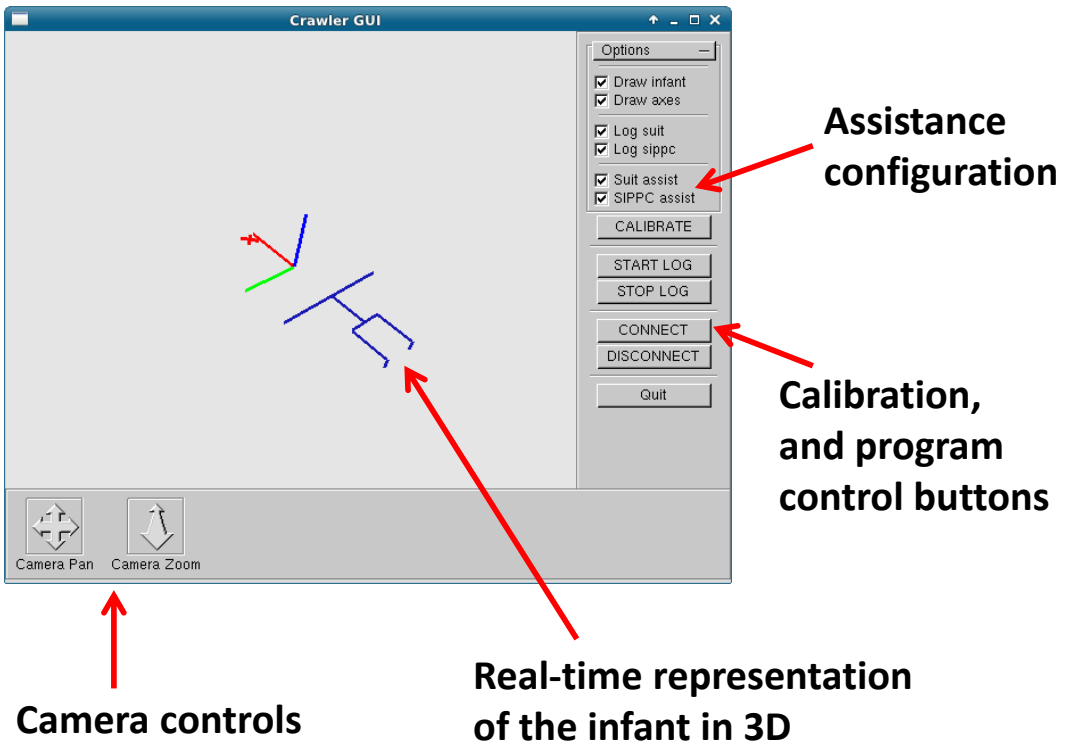

Figure B.1: *Graphical user interface used during data collection. The reconstructed skeletal configuration for the infant is shown in real-time and in 3D. Data collection can be started and stopped using the GUI. Also, SIPPC assistance can be configured as triggered by the SIPPC encoders, suit, both, or neither.*

Appendix C

Canonical Menu of Actions

C.1 Hierarchical Actions

The Canonical Menu contains a set of partially ordered actions. I choose to assist more complicated, coordinated, and potentially useful actions first. I rank action priorities by difficulty and use a priorized list of sets to simplify the process of deciding which action to consider first. The following table lists each action considered, as well as its corresponding priority level:

Action	Priority
feet near floor + left hand near floor + left hand sweep forward	7
feet near floor + left hand near floor + left hand sweep backward	7
feet near floor + right hand near floor + left hand sweep forward	7
feet near floor + right hand near floor + left hand sweep backward	7
sweep both arms forward, near floor	6
sweep both arms backward, near floor	6
feet near floor + left arm sweep forward	5
feet near floor + left arm sweep backward	5
feet near floor + right arm sweep forward	5
feet near floor + right arm sweep backward	5
sweep both arms forward	4
sweep both arms backward	4
pedaling with both feet	3
sweep with left arm forward	2
sweep with left arm backward	2
sweep with right arm forward	2
sweep with right arm backward	2
sweep with left arm to the left	2
sweep with left arm to the right	2
sweep with right arm to the left	2
sweep with right arm to the right	2
kick with left foot	1
kick with right foot	1

Table C.1: Action Hierarchy is implemented with a prioritized list of sets, where actions with the highest priority are considered for triggering assistance first. The first actions considered are specified by multiple position and velocity constraints (Priority = 7).