Design and Testing of Lightweight Inexpensive Motion-Capture Devices with Application to Clinical Gait Analysis

Eric Wade
Interaction Lab, Computer Science Department
University Southern California
Los Angeles, CA, USA
ericwade@usc.edu

Maja J Matarić
Professor of Computer Science and
Neuroscience
University of Southern California
Los Angeles, CA, USA
mataric@usc.edu

***Abstract*— The advent of more portable and affordable sensing devices has facilitated the study of rehabilitation robotics. Critical to the further development of therapies and interventions are low-cost, easy-to-use devices that can be applied in clinical and home care settings. In this paper, we present a low-cost motion capture system that relies on the open-source Player/Stage software development environment, and can be used in conjunction with a socially assistive robotic agent (or a computer interface) for various types of motor task rehabilitation training. We describe the hardware and software development for the device, and the activity recognition algorithm we developed to capture the relevant motion data. We present the overall framework in which this system can be adapted to other motor task-based rehabilitation regimens. Finally, we present initial experimental data in the domain of gait rehabilitation, in which we use the system to estimate cadence, walking speed, and stride length.**

I. INTRODUCTION

Wearable sensors have proven useful in several fields of study. Exploration of this technology has led to developments in wearable health monitoring systems [10], human computer and human robot interactions [8], and in service and assistive robotics [5]. Systems are typically composed of one or more sensors, a microcontroller, a wireless module, and a local PC. In healthcare applications, these components are used to acquire and store physiological data, either for future analysis and processing, or to generate real-time feedback.

Unfortunately, implementation of these systems can be challenging due to the wide range of application areas and the variety of software and hardware components available. As a result, many of today's systems suffer from one or more drawbacks that limit their usefulness to the target populations. For instance, some systems rely on proprietary or commercial software specific to a single device, rendering augmentation or modification of the software difficult, if not illegal. Other systems are limited by their cost – the prohibitive expense of many commercial systems makes them unrealistic for large-scale clinical trials or for in-home (or in-clinic) use by the target populations. Further limitations include the mechanical design of components. For instance, many stroke patients with limited mobility cannot outfit themselves with a skin-tight body suit, or subject themselves to heavy wearable sensing devices or computers. As wearable sensor networks are becoming more pervasive, further advancement in this field require systems that are inexpensive, simple in design, and easily adapted to multiple applications.

Toward that goal, we present a wearable motion sensing system designed to be lightweight, minimally invasive, low-cost, and easy to use and adapt to a variety of applications. We then examine the specific application of the system for use in a robot-based gait rehabilitation framework. Novel aspects of our system include: 1) the design of the motion capture sensing devices, 2) the software used to provide sensor data to the robotic assistant, and 3) the activity recognition algorithm used to process user motion data. Our goal is to utilize motion data in real-time to allow the socially assistive robotic (SAR) agent to assess and give feedback to the user as part of a real-time interaction. SAR systems assist through coaching, monitoring, and motivating, but without physical contact [5]. We present initial experimental results obtained from testing our system in the gait training context and then describe how our socially assistive rehabilitation robotics framework can be applied to other user populations. Though we describe a specific target application for the system, we discuss generally the hardware and software components, so as to demonstrate the dynamic nature and the ease with which our system can be adapted to a variety of applications.

II. BACKGROUND: CLINICAL GAIT ANALYSIS

Clinical gait analysis (CGA) is defined, according to Davis [3], as the systematic measurement, description, and assessment of quantities used to characterize human locomotion. CGA is the process by which abnormalities are assessed and the resulting assessment is then used to determine therapies or interventions to help the individual to relearn or improve the ability to walk. This is necessary for many populations, including those who have suffered from stroke and those living with Parkinson's Disease. In both cases, a degenerative neurological condition causes a variety of motor and non-motor deficits. The resulting symptoms can include

This work was supported by USC Women in Science and Engineering Program and the NSF CNS-0709296 grant for "CRI: IAD - Computing Research Infrastructure for Human-Robot Interaction and Socially Assistive Robotics" and NSF IIS-0713697 grant "HRI :Personalized Assistive Huma-Robot Interaction: Validation in Socially Assistive Robotics for Post-Stroke Rehabilitation"

problems such as tremor, slow movement, and rigidity. In the current treatment model for these individuals, regular visits are scheduled with a physical therapist or occupational therapist. Visits can occur as little as once a month – in the interim, the individual follows a rehabilitation regimen in the home setting. Thus, the health professional receives limited information regarding patient progress and the patient receives little steady, sustained coaching.

Using a SAR agent, we can increase the frequency of evaluation sessions and thus provide ongoing, sustained coaching. The robot or computer system can report richer data regarding the user's progress, allowing the health professional to update the exercise regimen as necessary. The wearable motion sensors also provides a tool through which the health professional can obtain a glimpse of the user's activities and capabilities during normal daily life. Monitoring this activity is essential to providing objective evaluations of the effectiveness of therapies. In the long term, we hope that our system can develop into a framework for home-based assessment and evaluation; at this stage we are validating the approach.

The design of technology-assisted therapies depends on accurate evaluation of the user's capabilities. CGA can be used to create metrics for the evaluation of gait-specific parameters; chief among these are cadence, walking speed, and stride length. It was shown by [13] that inertial measurement units (IMUs) consisting of accelerometers and rate gyros can be used to accurately estimate values for these characteristics. Marker-based vision systems are often used in gait experiments to obtain spatial data (e.g., pose, leg position) with pressure plates or shoe pads used to obtain temporal data (e.g., heel strike, toe-off, swing vs. stance). However, with IMUs, these data can be obtained simultaneously from a single device (see Section V). Our IMU design, and the design of the other components of our rehabilitation framework, are described in Section III.

III. COMPONENTS OF THE SAR REHABILITATION FRAMEWORK

The goal of SAR systems is to utilize robots to assist certain populations (in particular, those with special needs) with daily life. These systems are attractive because they require no physical interaction with the user; rather, through gestures, speech, and other "social" means, they provide encouragement, coaching, and training. In this section, we describe the hardware and software development for our rehabilitative robotic framework. The hardware components include the motion "suit", which consists of the inertial measurement units (IMUs) and the central computer, and the SAR agent. A schematic representation of the system is presented in Figure 1. Sensors inside the IMUs obtain user motion data; these data are then tranmitted through the I^{2}C data bus to the central computer. The use of a standard I^{2}C bus allows for the use of a large variety of devices. The central computer then wirelessly transmits the data from the IMUs to a computer on the SAR agent. The design of these components is discussed in greater detail next.

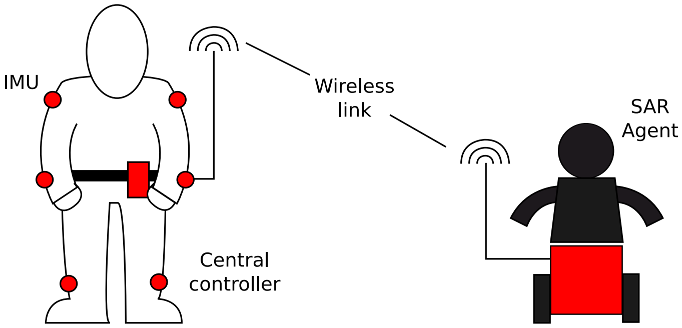

Fig. 1. Rehabilitative robotic system framework consisting of a wearable motion capture suit and a SAR agent. The user wears the IMUs and the central controller, which communicates motion data wirelessly to the SAR agent.

A. Inertial measurement units

Recently, integrated IMUs such as the Analog Devices© ADIS16365, have become available off-the-shelf, but they can be expensive, heavy, and bulky. Because we are developing wearable devices, we aimed for smaller, more user-friendly designs. Given the low cost of discrete components, we determined that we could design a smaller, more lightweight, and inexpensive IMU ourselves. Thus, we would determine the design for the printed circuit boards (PCBs), the sensor shell, and the fixturing mechanism simultaneously. By doing so, we were able to design the housing, and continually optimize the design for small size and weight, ease of calibration of the sensors, low-cost, and easy fixturing.

The IMUs rely on inertia-based MEMS sensors to obtain motion information from the user. Each IMU contains a 3-axis accelerometer, three single-axis rate gyros, and one single- and one dual-axis magnetometer. This design is a modified version of that presented in our previous work [8]. The accelerometer (Memsic© MXR9150) is used to obtain accleration (and subsequently, position (x, y, z)), the rate gyros (Analog Devices© ADXRS300) are used to obtain angular rate-of-change (and subsequently, orientation $(\theta_x, \theta_y, \theta_z)$), and the magnetometers (Honeywell© HMC1051 and HMC1052) are used to obtain magnetic North. The magnetometer data can then be used to determine the direction acceleration due to the Earth's gravitational field; this is subsequently subtracted from the accelerometer signal during data processing.

The IMU sensor outputs are low-pass filtered (using analog filters) and sampled by a 10-bit ADC. An Atmel© ATmega324 microcontroller running in slave-transmitter mode reads the digitized sampled data, arranges them into individual packets, and transmits each packet (along with a timestamp) to the central computer using the I^{2}C bus interface.

Each IMU is composed of a plastic enclosure and four printed circuit boards (PCBs) we designed (see Figure 2). The lightweight enclosure has a concave shape in order to contour to the human body. The shell requires a single

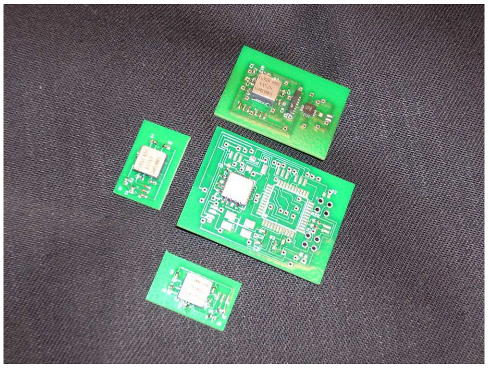

Fig. 2. Printed circuit boards contained in each IMU. The circuits handle all sensing, analog-to-digital conversion, and I^{2}C transmission.

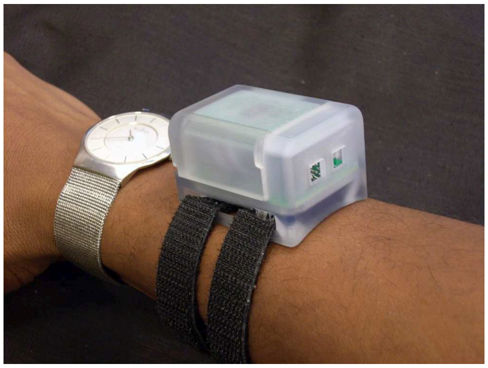

Fig. 4. Inertial measurement unit, including sensor shell and PCBs, shown on a wrist along with a watch for size comparison.

fastener to be securely closed. It also contains "slots" into which the PCBs fit (See Figure 3). The slots are arranged so that the PCBs (and the axes of the various sensors) are properly aligned. This design allows for simple construction and also ensures easy access to the boards for any required maintenance.

After an iterative design process, and multiple prototypes, we arrived at the final version of the IMU pictured in Figure 4. It has total weight of 45.3g, and measures 27mm×35mm×45mm.

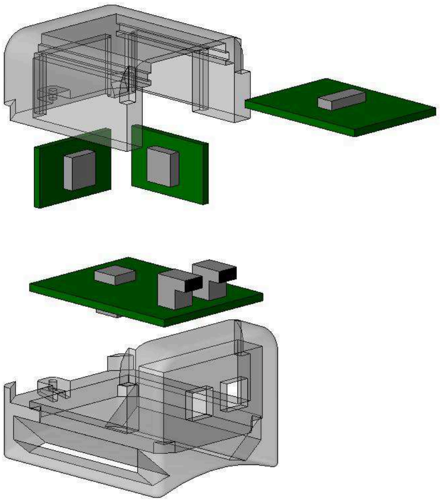

Fig. 3. Exploded view of the IMU. This solid-model rendering shows the PCBs and the sensor shell. The slots in the shell for the PCBs are also visible.

B. Central control unit

The central control unit of the motion suit contains a computer, a battery pack, a power converter, and other discrete components (see Figure 5). To ensure scalability for future applications, we opted to use a powerful computer instead of a microcontroller. We chose the Gumstix©Wifi-Stix pack, a small Linux computer (80mm×20mm) with a 400MHz processor with 64MB of RAM. Wireless communication is facilitated by a Netwifi-MicroSD wireless transmitter. The transmitter, which includes an antenna, is FCC-certified and can use the 802.11(b) and 802.11(g) transmission protocols. The small size and form factor allows for easy packaging of the computer for wearable applications. The Gumstix has a large developer community, and software for a number of interface devices (including the I^{2}C and SPI data transmission protocols) is readily available. An advantage of the Gumstix platform is that it lends itself to development for various robotics software suites, such as the Player/Stage development environment, which is further discussed in Section D.

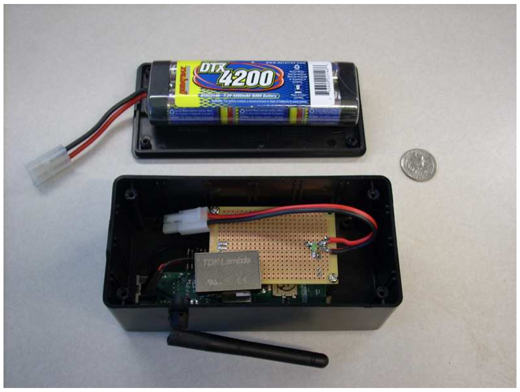

Fig. 5. The central controller (Gumstix, battery, DC-to-DC converter, and housing).

The current central control unit is powered by a 7.2V RC battery, and a DC/DC converter to provide the appropriate power level to the Gumstix pack. For ease of use, we included a power switch and a power indicator LED on the shell. The shell measures 50mm×75mm×150mm. There is a plastic clip on the back of the shell that allows it to be hung from a belt or waistband.

C. Socially assistive robotic agent

The SAR agent is actually two robots working in concert (see Figure 6). One is a Pioneer 2DX mobile robot base used for locomotion. Mounted on the base is a SICK LMS laser range finder for obstacle avoidance and speakers for verbal interaction with the user. The mobile base runs autonomously, reacting to proxemic and motion data from the user.

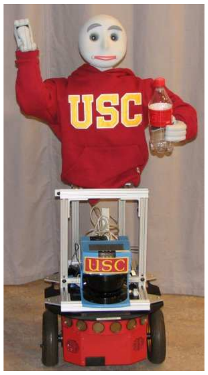

Fig. 6. SAR agent system consisting of a Pioneer mobile base and the Bandit III humanoid torso. The Bandit torso is capable of performing a variety of gestures, using its arms and its face.

The other robot is the humanoid torso Bandit III, custom designed and developed for our group's SAR experiments. The humanoid robot torso weighs 6.8kg, is 400mm high, 355mm wide (across the chest) and 127mm thick, with 355mm long arms. Each arm has 7 degrees-of-freedom (DOF), including the 1 DOF gripper. There are 16 DOF in the torso, including the 2 DOF in the neck/head. The head of the robot has 1 DOF expressive eyebrows, and 3 DOF expressive mouth. The humanoid has a fully actuated neck, an actuated mouth, actuated eyebrows, and Firewire cameras mounted in the eyes. It can communicate expressively using speech, arm movements and gestures, head movements, and facial expressions.

Both the mobile base and humanoid torso are controlled by an onboard quad-core Linux computer. The mobile base is a standard test-bed in mobile robotics, and has been extensively tested and evaluated by various users [4]. While the original version of the IMU was tested by stroke patients as described in [4], the version presented in this paper is revised, lighweight and lower cost. Our past work has validated the previous framework, consisting of a single IMU used with a mobile base (no humanoid) robot. In the newer work, we are using multiple IMUs to facilitate richer and more precise understanding of the user's motion and provide customized and detail feedback. Furthermore, we are using the humanoid robot to take advantage of the expressive interaction modalities (for demonstration of movements, praise, and motivation) that were not possible with the previous robot embodiment.

D. Software drivers – development for the Player/Stage environment

Because our system consists of a collection of sensing devices, robotic agents, and computers devices, it is convenient to use a shared network for communication. We use the Player/Stage software environment [6]. Player is a free software package – it provides a server/client architecture that can be used in a variety of robot and sensor system applications. In a typical implementation, the Player server runs on a host machine. Individual clients can then facilitate tasks such as sensor reading, or generating commands for actuators. Details of the Player architecture and how it can be applied to robotic applications and mobile sensing are available from numerous sources (e.g., Beetz [11]). Drivers are readily available for many platforms, including the Pioneer mobile robot. In our lab, we developed drivers for the humanoid robot and have also our own Gumstix driver (that we have made available through the Gumstix message boards). This driver can be used to issue command from the Gumstix terminal (e.g., polling devices connected to the I^2C bus) and to access Gumstix functions (e.g., transmitting arrays to the robot using the wireless connection). It follows the typical driver structure detailed in Beetz [11]. This driver must be compiled along with Player on a local machine; subsequently, the binary can be sent to the Gumstix hardware.

The driver issues commands from the Gumstix, acting as a master-receiver, polling the microcontrollers in each IMU for its ADC data. It then transmits this information to the host machine using the Player server (over TCP-IP). Finally, a client on the host machine reads the sensor data, and publishes it for processing and storage. This is convenient, as all data (e.g., laser range finder data from the scanner mounted on the mobile base, IMU data, robot position) can be read simultaneously by the SAR agent's onboard computer, allowing us to use sensor fusion techniques to determine an appropriate response to the user's actions.

IV. DATA PROCESSING AND ACTIVITY RECOGNITION ALGORITHM

Performing CGA with a SAR agent requires an accurate and efficient activity recognition algorithm. Activity recognition, in general, consists of the acquisition of motion data, and a subsequent matching of the perceived motion to some model of motion trajectories. Typically, raw motion or pose data are obtained using motion sensors, the data are then segmented temporally based on a specific feature, (e.g., zero crossings), a database of motions and poses is generated *a priori* using motion models, these motion models are grouped using techniques such as *k*-nearest neighbors, and finally the segmented sequences are matched using similarity (and often, dynamic time warping) to the known motions stored in the database. In our algorithm, the final data processing step is to classify the gesture using a Hidden Markov Model (HMM) [1].

The above activity recognition procedure can, unfortunately, be extremely inefficient and computationally intensive, rendering real-time recognition difficult. To address this problem, researchers have begun to utilize geometric features in order to more efficiently process and recognize motion data [2][7][9]. Of particular interest is the work by Müller [9] who applied geometric techniques to perform query-based database searches for specific poses and gestures with application to animation and computer graphics. By considering qualitative, geometric relationships between various body parts, an additional pre-processing step can be added to the traditional recognition algorithm described above. Before segmenting temporally, we can use geometric relationships to identify logically corresponding events in a data sequence. This can be used to determine specific segments of the data that are of particular interest; subsequently, more computationally intensive content-retrieval techniques can be used only on these smaller segments. In this way, geometric features can be used to increase the speed and reliability of the content-based search, allowing for real-time user feedback from the SAR agent. Once relevant geometric features are determined, they are defined using geometric feature planes. For instance, a geometric feature such as "the left hand is in front of the body" can make use of a geometric plane defined by the two shoulders and the navel. We apply the techniques to activity recognition for specific motions. In doing so, we maintain all of the advantages of traditional content-based retrieval with the increased efficiency of a more generalized classification system. This geometric segmentation technique fits into our overall data processing framework as shown in the schematic in Figure 7.

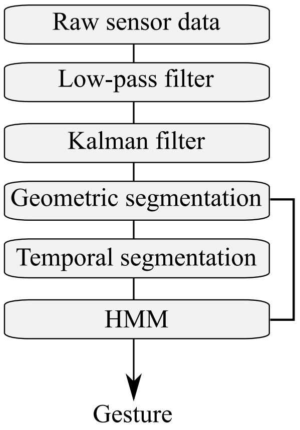

Fig. 7. Data processing algorithm used in our SAR rehabilitation framework. First, the raw sensor data are low-pass filtered. Next, Kalman filters are used to estimate bias, position, and orientation. The position and orientation information is geometrically segmented and used to determine when temporal segmentation is enacted. Finally, the geometrically and temporally segmented data are used to determine a gesture, using Hidden Markov Models.

In a typical CGA experiment, the user is expected to do a task (e.g., walk in a straight line for a specific amount of time). In most applications of the rehabilitation framework, the user will be guided by the SAR agent. Thus, in spite of variations in actual performance, we have a rough expectation for the observed motion. This expectation informs our choice of the relevant geometric relationships for a given task, and can be described mathematically by the geometric feature planes.

The implementation is best described using a simple example. Using the terminology of Müller, we can define a data stream as a sequence of poses P:

$$P \in \mathbb{R}^{3\times|J|}, P = \begin{bmatrix} x_1 & ... & x_j \\ y_1 & ... & y_j \\ z_1 & ... & z_j \end{bmatrix} \qquad (1)$$

where J is the number of joints in the kinematic model. We then define a Boolean function, $F : P \rightarrow \{0, 1\}$. This Boolean function is based on a specific geometric feature. In the instance of walking, we can think of a geometric feature describing the position of the feet with respect to the torso – in this case, a plane coincident with the coronal plane. We have $F \rightarrow \{0, 1\}$, with:

$$F_i = \begin{cases} 1 & \text{When foot}_i \text{ is in front of plane} \\ 0 & \text{When foot}_i \text{ is behind plane} \end{cases} \qquad (2)$$

According to this structure, a data sequence in which the user is walking produces:

$$\mathbf{F_{walk}alk} \equiv \begin{pmatrix} F_{left} \\ F_{right} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \cdots \qquad (3)$$

Thus, if we observe a sequence in which $\mathbf{F_{walk}alk}$ alternates between zero and one (and considering that we are only interested in gait characteristics), we can perform traditional retrieval techniques during a particular sequence of interest. For CGA, IMUs placed on the ankles can easily detect heel-strike. We can use this to determine right foot initial contact, or the beginning of right stance phase. We can then wait until the transition of $\mathbf{F_{walk}alk}$ from $(0; 1)$ to $(1; 0)$ – this will indicate when the left toe is passing the right foot, and can be used to analyze toe clearance in a subject. This is a simple example – the power of this technique lies in the use of multiple geometric features simultaneously to distinguish more complex motions.

In practice, the selection of the feature planes is an iterative process. In the gait example, we looked at a number of different geometric plane placements including the coronal and sagittal planes. In the end, we determined that the planar definition used in the example was most informative. In the future, we plan to apply learning techniques to properly determine the most informative geometric feature planes.

V. RESULTS – PILOT EXPERIMENT

To demonstrate the efficacy of our system, we present initial pilot data from a gait analysis experiment. As mentioned, much of CGA depends on accurate predictions of cadence, stride length, and walking speed. Our hypothesis was that, using only the motion data from two IMUs (one per ankle), we can apply our activity recognition algorithm to estimate these three parameters.

A. Experimental setup

To validate our hypothesis, we performed the following experiment. We placed markers on the ground for position measurements. We then set up a camera perpendicular to the markers, as depicted in Figure 8, to capture marker locations and the experimental subject. Finally, we outfitted the subject

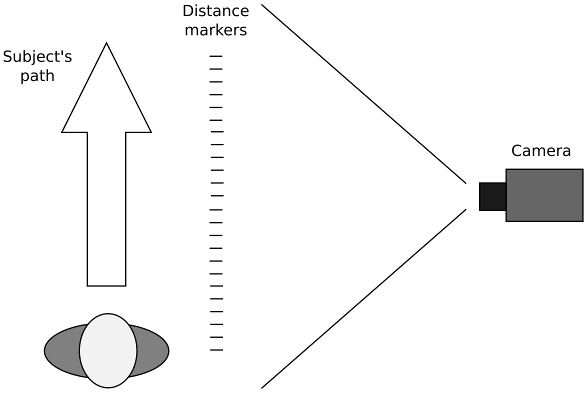

Fig. 8. Experimental setup for gait experiments, depicting the subject, the position markers, and the camera.

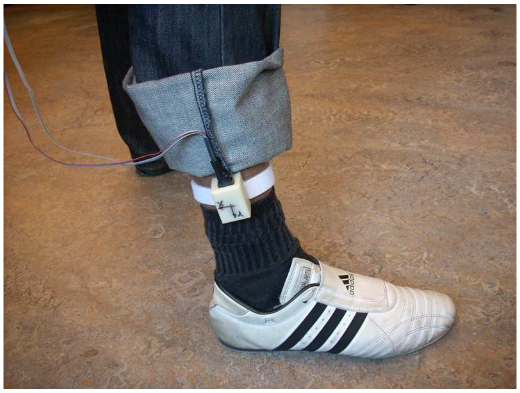

Fig. 9. IMU fixed to the ankle for gait analysis experiments.

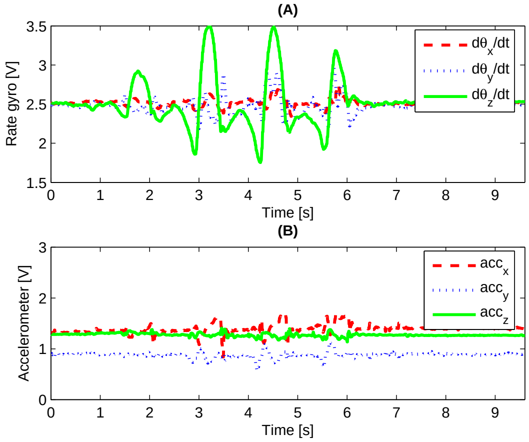

Fig. 10. Typical resuls from an ankle IMU. In this case, the IMU is placed on the right ankle.

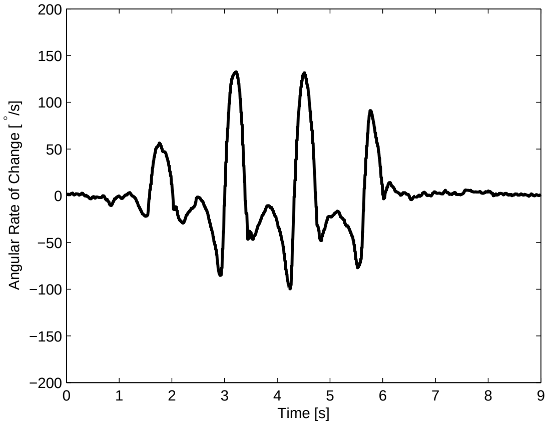

Fig. 11. z-direction rate gyro output for the right ankle. The three 'toe-offs' and 'heel-strikes' are indicated by the dips at 2.9s, 4.3s, and 5.6s, and at 3.6s, 4.9s, and 5.9s, respectively.

with two IMUs (one per ankle, as seen in Figure 9). The test subject then walked the length of the marked path.

Typical output results can be seen in Figure 10. Here, the *x*-, *y*- and *z*-direction rate gyro and accelerometer outputs are shown (in (A) and (B)), with respect to time, for the IMU placed on the right ankle. In this reference frame, the x-axis is normal to the sagittal plane, the y-axis is normal to the transverse plane, and the z-axis is normal to the coronal plane. (Note the presence of gravity in the y-direction acceleration in Figure 10.) When comparing these results to those obtained from other inertial sensors used for gait, we get similarly accurate results, but with the advantage of being able to place our IMU on the ankle, out of the way of any footwear [12]. In Figure 11, we place a focus on the z-direction angular rate-of-change from Figure 10; these data can be used to pinpoint heel-off and toe-strike (note that the vertical axis has been converted to $^o/s$). 'Toe-off' is indicated by the dips at 2.9s, 4.3s, and 5.6s, and 'heel-strike' by the peaks at 3.6s, 4.9s, and 5.9s.

B. Results

In Figure 12, we show data obtained from an IMU placed on the ankle. In this case, we use the geometric rule from our example in Section IV. Transitions occur at times $t = 11.45s, 13.12s$ and $14.87s$ (note that the plot results are for acceleration, not position). Thus, we obtain: $\mathbf{F_{walk}} = (0;1)_{t=9.87:11.45s}$, $\mathbf{F_{walk}} = (0;1)_{t=11.89:13.12s}$, and $\mathbf{F_{walk}} = (0;1)_{t=13.38:14.87s}$. In the time before $t \approx 9.8s$, the time between the aforementioned intervals, and the time after $t \approx 16.2s$, $\mathbf{F_{walk}} = (1;0)$ until sampling ends. Thus, we know that walking is occuring between 9.8 and $16.2s$.

Cadence can be ascertained from looking at the very distinct spikes corresponding to heel-off (e.g. at 9.87s for the right foot). Stride length can then be estimated by looking at the change in the x-axis position, obtained by subtracting the gravity component and twice integrating the data. Finally, walking speed can be ascertained by dividing the change in x-axis position by the time between subsequent instances of heel-off. Comparing the values shown here to indicators from hand-annotated video data, we obtain an 15% error in our cadence estimate, 8.6% error in our stride length estimate, and 9.3% error in our walking speed estimate (See Table 1).

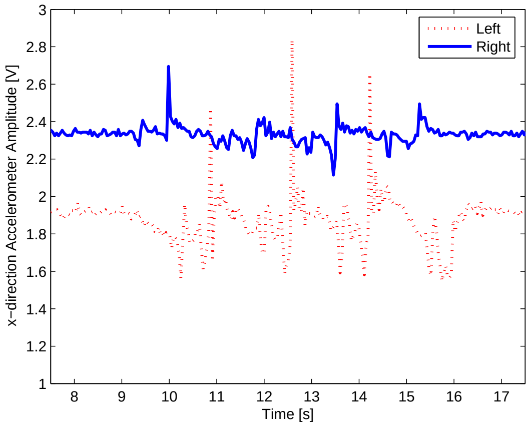

Fig. 12. Accelerometer data from IMUs placed on the ankles. For simplicity, only the *x*-direction (parallel with the ground) result is shown.

Avg. Characteristics	**True Value**	**Estimate**	**Error**
Cadence [Hz]	1.3	1.1	15%
Step length [m]	.35	.32	8.6%
Walking speed [m/s]	.54	.49	9.3%

While these results are somewhat limited, they are encouraging in that we are able to find characteristics relevant to CGA. We have only validated the results with a single, healthy subject. However, this pilot is useful in showing that our devices can in fact be used to obtain useful gait information. Going forward, these results will inform the design and testing of the devices with populations suffering from known gait deficits.

VI. CONCLUSIONS AND FUTURE WORK

We have presented the design of a low-cost, easy-to-use wearable inertial measurement unit (IMU)-based motion sensor system. The devices used for the fabrication of the IMU and the central computing unit are inexpensive, and use an open, easily modified software architecture that can be applied to a number of different applications. They are economical, yet capable of acquiring the relevant data for motion recognition. We validated the sensor design on a single subject, with the specific goal of acquiring gait information. We then validated our activity recognition algorithm using the trial data.

Future work will include more rigorous validation of our activity recognition algorithm with application to CGA and othor motor task recovery interventions.

In addition to more complete clinical trials in the domain of CGA, we expect the motion capture devices and the activity recognition algorithm to be useful for a number of populations and applications, including upper body gesture recognition for stroke rehabilitation and therapies and other uses of SAR systems.

REFERENCES

[1] E. Alpaydin. *Introduction to Machine Learning*. MIT Press, Cambridge, MA, 2004.

[2] S. Carlsson. Combinatorial geometry for shape representation and indexing. *Object Representation in Computer Vision*, pages 53 – 78, 1996.

[3] R. B. Davis. Clinical gait analysis. *IEEE Engineering in Medicine and Biology Magazine*, pages 35 – 40, September 1998.

[4] J. Eriksson, M. J. Matarić, and C. Winstein. Hands-off assistive robotics for post-stroke arm rehabilitation. In *Proc. IEEE International Conference on Rehabilitation Robotics (ICORR'05)*, pages 21–24, Chicago, Il, USA, June 2005.

[5] D. J. Feil-Seifer and M. J. Matarić. Defining socially assistive robotics. In *International Conference on Rehabilitation Robotics*, pages 465–468, Chicago, IL, Jun 2005.

[6] B. Gerkey, R. Vaughan, and A. Howard. The player/stage project: Tools for multi-robot distributed sensor systems. In *Proceedings of the International Conference on Advanced Robotics*, pages 317–323, Coimbra, Portugal, Jul 2003.

[7] L. Kovar and M. Gleicher. Automated extraction and parameterization of motions in large data sets. *ACM Transactions on Graphics*, 23(3):559 – 568.

[8] N. Miller, O. C. Jenkins, M. Kallmann, E. Drumwright, and M. J. Matarić. Motion capture from inertial sensing for untethered humanoid teleoperation. In *Proc. of the IEEE-RAS International Conference on Humanoids Robotics (Humanoids'04)*, Santa Monica, CA, USA, Nov. 2004.

[9] M. Mullüer, T. Röder, and M. Clausen. Efficient Content-Based Retrieval of Motion Capture Data. *ACM Trans. on Graph.*, 24:677–685, 2005.

[10] E. Post, M. Reynolds, M. Gray, J. Paradiso, and N. Gershenfeld. Intrabody buses for data and power. *IEEE Symposium on Wearable Computers*, pages 52 – 55, October 1997.

[11] R. B. Rusu, A. Maldonado, M. Beetz, and B. Gerkey. Extending player/stage/gazebo towards cognitive robots acting in ubiquitous sensor-equipped environments. *Proc. of the ICRA Workshop on Network Robot Systems*, April 2007.

[12] A. Sabatini, M. Chiara, S. Scapellato, and F. Cavallo. Assessment of walking features from foot inertial sensing. *IEEE Transactions on Biomedical Engineering*, 52(3):486 – 494, March 2005.

[13] A. Salarian, H. Russmann, F. J. G. Vingerhoets, C. Dehollain, Y. Blanc, P. Burkhard, and K. Aminian. Gait assessment in parkinson's disease: Toward an ambulatory system for long-term monitoring. *IEEE Trans. on Biomedical Engineering*, 51(8):1434 – 1443, August 2004.