A Gait Speed Measurement System for the Clinical Workflow

Ryan Cooper
ryan.cooper@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Wang Xie
xiewang@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

William Dingler
wdingler@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Jingsong Guo
jguo366@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

Abrar Ahmed
abrar.ahmed@gatech.edu
Georgia Institute of Technology
Atlanta, Georgia, USA

1 INTRODUCTION

As the average human life increases so does the necessary concern for the quality of life. Quality of life in the twilight years is often greatly affected by health. Moreover, health can be defined by both mental and physical categories, and often is the case that a deterioration in physical health will result in deterioration in the mental category. This is where the importance of physical mobility and by extension the study of gait speed comes into play. A loss of mobility in an elderly person is often the leading factor to a loss of independence. No one wants to become a burden on their loved ones and the idea of an active mind free from deficits in mental faculties without the abilities for active physical movement sounds like a prison to many entering their late years. This is where we argue for the importance of studying gait speed. Gait speed is the time one takes to walk a specified distance on level surfaces over a short distance. It is very important when framing this that it is our belief that by studying gait speed we can predict many of the later milestones in a person's life, but also avoid a trip that leads to a fall both in the metaphorical sense and quite literally.

In this paper we used data science and skills we had learned through the semester to help ongoing research with this project's sponsor. We successfully created a tool-assisted pipeline to help refine data and focus the project. With the previous in mind we feel confident that the pipeline will continue to be of great use in the study of Gait Speed even as the project continues to evolves in the future.

2 RELATED WORK

Gait speed has been proved to be a strong clinical indicator of mobility impairment in patients with neurological disorders and other related diseases. There are previous efforts spared on the development of the methods of gait speed monitoring with assessment, as well as the gait speed data analysis for patient physical condition estimation. The participant gait speed data used in this work were collected from nine different studies of 34,485 adults aged 65 years or older [2]. In all studies, participants were asked to walk a certain distance at a regular pace. A stopwatch was used to record the time it took for participants to walk that distance. [1] and [3] use a pulsed Doppler radar ranging sensor to measure gait speed with the primary objective of preventing falls in the elderly. The radar emits an electromagnetic signal and measures the frequency shift of the signal to identify the speed of the participant based on the Doppler effect. The different Doppler signals come from various moving parts of the body. Microsoft Kinect is used to obtain more detailed gait features. It is wall mountable at a suitable height with the ability to capture the moving participant [5]. An algorithm is applied attempting to capture the footsteps by detecting moments in time when the feet are standing still. Gait speed can be determined with these locations and timing information. Recently, this idea was extended by installing a Kinect-based gait measurement system to observe the participant's main living area continuously [6]. This allowed undisturbed gait measurements to be taken without further professional assistance after the initial setup. A machine learning method for gait speed estimation using a configurable array of skin-mounted, conformal accelerometers was introduced in [4]. It would also support the use of wearable accelerometer arrays for walking speed estimation among the patients with gait impairment.

3 WORK ACCOMPLISHED

We developed a data pipeline that focuses on cleaning data and removing noise. With the timeless mantra of "garbage in garbage out" in mind, the task of creating a uniform data pipeline began. Our pipeline defines the creation of A to B and B to A sensor sets. Without defining the sets, we would be unable to know that a person has passed by both sensors and moreover, that two independent people activated both sensors within a certain period of time.

3.1 Data and Sensors

The data we are using has been collected as part of the Aware Home Research Initiative (AHRI). Because our research focuses on measuring gait speed from PIR-based motion sensors, our setup involves an array of these motion sensors arranged co-linearly. These sensors have a fixed radius and range through which they detect motion. Therefore, a subject walking through a hallway where an array of these sensors is arranged would be detected by each sensor in the array sequentially with some delay between the activation of each sensor. By knowing the distance over which these sensors are arranged as well as the delay between the activation of each sensor in the array, we can theoretically estimate the gait speed of a subject. For our preliminary research we make use of data collected in a home which includes a setup identical to the one described above with multiple subjects of varying ages as well as pets. This will be refered to as CEP000 and a reference diagram for physical sensor locations can be visualized in Figure 1. In this diagram, notice sensors A and D represent the Family Room and

Kitchen Room respectively, and B and C are the end nodes on the gait array.

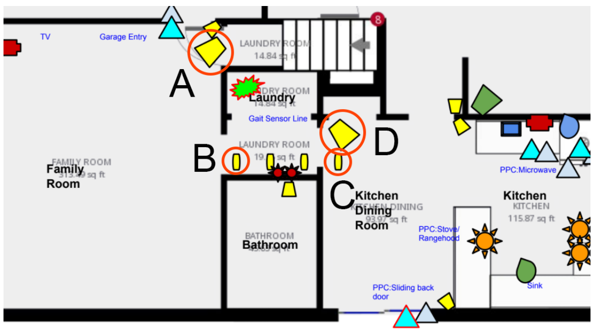


Figure 1: Diagram of the Sensor Setup in CEP000

3.2 Candidate Matching Algorithm

Our data cleaning pipeline consists of two main parts, first, from the raw sensor data we generate a so-called sensor activation, that is, when the sensor state changes from on back to off. From this, we can then use these generated sensor activations to generate candidate gait events, that is, when a person walks from either sensor A to B or B to A.

3.2.1 Candidate Matching. Below we will break down our Candidate Matching in Algorithm 1. First we must define two data structures, a GaitActivation and a GaitEvent. A GaitActivation contains the sensor that was activated and the timestamp of the activation. The GaitEvent is an object that contains the list of GaitActivations in the gait event and the direction of the gait. We are striving to assemble a list of quality GaitEvents from a list of GaitActivations. The algorithm can be described below is our final candidate matching algorithm. Let X be the list of sorted sensor activations where every element $e \in X$, $e = (\text{timestamp}, \text{sensor})$. Let m be the minimum timedelta of a gait event and n be the maximum timedelta of a gait event. Let u be a hashset of seen sensor activations.

The general intuition behind this approach is a sliding temporal window with thresholding. For an event, we look forward in time and see if we can find a pair sensor matching on the other side of the gait array within the threshold limits. If so, we consider this a gait event and add both events to the used set. In theory, this algorithm is not that complex. We made some smart data structure decision to minimize the runtime.

We set our parameters to be $m = 0.272$ and $n = 5$, both in seconds. We chose 0.272 seconds as a minimum as anything faster than this would exceed human limits of speed.

3.2.2 Denoising from Algorithm 1. As a case study on the algorithm, we will look at one single day of data to show the effectiveness. First, consider Figure 2, these are representations of the raw sensor activations over time for one of the sensors in a sample home. Notice in the selected sensor, there is a lot of noise and the sensor is being activated often, which is not represented in the opposite sided sensor.

Algorithm 1 Candidate Matching

procedure $\textsc{CandidateMatching}(X, m, n)$
    `Assert` $m \geq 0 \land n \geq 0 \land n \geq m$
    $u \leftarrow \text{set}()$
    $Y \leftarrow \text{list}()$
    for $e \in X$ **do**
        $t_i, s_i \leftarrow e$
        if $e \in u$ **then** `continue`
        end if
        for $f \in X[i:]$ **do**
            $t_j, s_j \leftarrow f$
            if $f \in u$ **then** `continue`
            end if
            if $s_i \neq s_j$ **then**
                if $t_i < t_j \land t_j - t_i < n \land t_j - t_i > m$ **then**
                    Y.append(`new GaitEvent`(e, f))
                    u.add(e)
                    u.add(f)
                end if
            end if
        end for
    end for
    return Y
end procedure

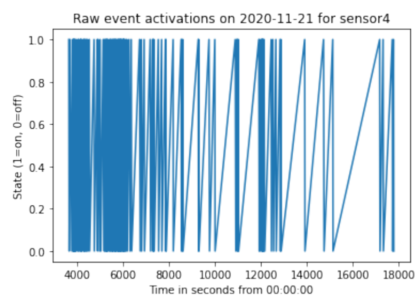


Figure 2: Raw sensor activation over time for sensor 4 on 2020-11-21

This is problematic as discovering the true gait events in the noise may be difficult. Using the technique we developed, we see that these sensor activations can be reduced to Figure 3. Notice that this is much cleaner and we do not see any long periods of sensor activation. This matching is generated using time differences between both sensor 1 and sensor 4 in the dataset.

3.2.3 Contextual Fusion. Our algorithm can be further improved by incorporating sensor data collected from the environment to help us better analyze the activation scenarios, a technique categorized as contextual fusion. Our goal is to extract complete paths through the gait speed array, so a way to eliminate noise, or sensor activations that are resulted from incomplete paths or inconsequential movements, is to utilize motion sensors that are placed on both ends of the gait speed array. As shown in Figure 1, we can

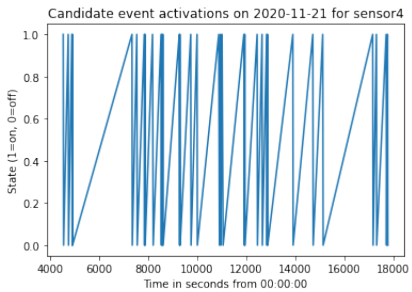


Figure 3: Candidate event activation over time for sensor 4 on 2020-11-21 using Algorithm 1

recognize a complete path if all the sensors are activated continuously in an unidirectional order that can start from either end. Thus, we could define the criteria of a complete path as a tuple of unique sensor activations according to their activation time $\{sensor_a, sensor_b, sensor_c, sensor_d\}$ such that $sensor_a$ and $sensor_d$ are motion sensors at both ends while $sensor_b$ and $sensor_c$ are the gait speed array sensor.

The algorithm can be described below is our final candidate matching algorithm with contextual fusion. Let X be the list of sorted sensor activations where every element $e \in X, e =$ (timestamp, sensor). Let m be the minimum timedelta of a gait event and n be the maximum timedelta of a gait event and o be the auxiliary maximum timedelta of room activations. Let u be a hashset of seen sensor activations. Let q be the set of sensors not in the gait array. Let r be a set of tuples containing non-gait array to gait-array sensor correspondence, that is, which room sensor belongs to which sensor in the gait array to define order.

3.2.4 Denoising from Algorithm 2. To show the effectiveness of contextual fusion, we have visualized the activation activity associated with complete paths that is derived from the same raw dataset showed previously. We can see that even as compared to Figure 3, Figure 4 shows less potential noises. Like any classification problem, we cannot ensure that everything that is being filtered out by the algorithm is unwanted noise, nevertheless we do have an increased degree of confidence of our remaining data being relevant.

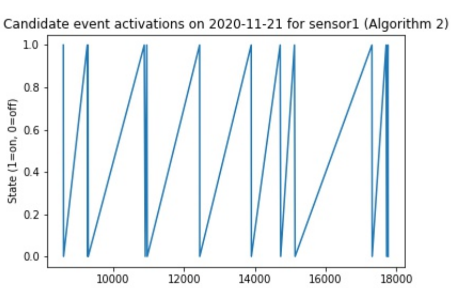


Figure 4: Candidate event activation over time for sensor 4 on 2020-11-21 using Algorithm 2

Algorithm 2 Candidate Matching with Contextual Fusion

procedure CandidateMatchingCF(X, m, n, o)
    `Assert` $m \geq 0 \wedge n \geq 0 \wedge n \geq m \wedge o \geq 0$
    $u \leftarrow$ set()
    $Y \leftarrow$ list()
    for $e \in X$ **do**
        $t_i, s_i \leftarrow e$
        if $e \in u \vee s_i \notin q$ **then** `continue`
        end if
        for $f \in X[i:]$ **do**
            $t_j, s_j \leftarrow f$
            if $f \in u \vee s_i = s_j \vee \neg(t_i \leq t_j \wedge t_j - t_i < o)$ **then** `continue`
            end if
            if $(s_i \in q \wedge (s_i, s_j) \notin r)$ **then** `continue`
            end if
            for $g \in X[j:]$ **do**
                $t_k, s_k \leftarrow g$
                if $g \in u \vee s_i = s_k \vee s_j = s_k \vee \neg(t_j \leq t_k \wedge t_k - t_j < n)$ **then** `continue`
                end if
                for $h \in X[k:]$ **do**
                    $t_l, s_l \leftarrow h$
                    if $h \in u$ **then** `continue`
                    end if
                    if $s_i \neq s_j \neq s_k \neq s_l \wedge s_i, s_l \in q \wedge (s_i, s_j) \in r \wedge (s_k, s_l) \in r$ **then**
                        if $s_k < s_l \wedge s_l - s_k < o$ **then**
                            Y.append(new GaitEvent(e, f, g, h))
                            u.add(e)
                            u.add(f)
                            u.add(g)
                            u.add(h)
                        end if
                    end if
                end for
            end for
        end for
    end for
    return Y
end procedure

3.3 Problems we Encountered

Initially, we had considerable trouble acquiring the necessary data as much of it contained personally identifiable information about each research participant's home, name, family, and age. Before we were allowed access to this data each team member had to submit paperwork to GA Tech's Institutional Review Board. After going through the proper channels to acquire the data we began to try and apply some of the ideas we had brainstormed. Almost immediately we began to notice noise in the data. predominantly this noise was caused either by sensor placement consistency or sensor anomalies caused by outside elements. An example of sensor noise caused by an outside element was that in one instance while recording data a sensor fell from its mount and into a potted plant.

Refusing to throw out any old data, we knew that if we could isolate the noise and prove it to be consistent for a wide range of events then we could isolate it from the bulk of the data. We elaborated on our methodology of removing noise in another section. An additional issue we encountered was that half of the data collecting since the original project's inception was was in a different format. This created issues of Data uniformity, as such part of the effort in developing the pipeline, was dedicated to converting all styles and formats of data to a format that was uniform across all events.

4 RESULTS/DISCUSSION

We proposed two candidate matching algorithms in this project, the second algorithm utilized more events captured by more sensor activation and thus provides us with more implicit matching events of higher precision.

Using the output of the candidate matching, we are able to estimate the speed of a subject moving through the area where the sensors are mounted. By dividing the distance between the sensors by the time delay between the candidate activation, we can derive gait speed estimates throughout the day.

When we apply this process over all the observations of the data for the home, we see that we are presented a multi-modal histogram (as seen in Figure 5); which is expected as multiple people and creatures are living in the home - which is a promising result for our technique.

As expected, we observe higher gait speed estimates during the day, peaking around the morning when subjects may be in a rush to get ready for work or school. During night-time when subjects are likely to be asleep, we see minimal gait activity in Figure 7.

In CEP000, the last sensor is mounted right above the kitchen, therefore it contains a large amount of noise which comes from subjects moving in the kitchen not necessarily co-linear with the gait speed array. Our techniques for noise reduction outlined above seem to do a good job of removing such noise as illustrated in Figure 6.

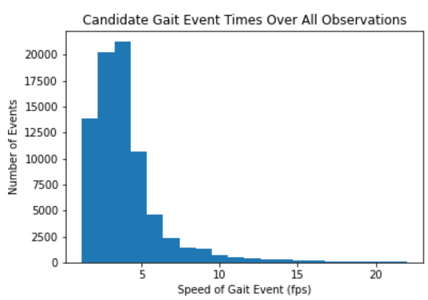


Figure 5: Candidate gait event times over all observations

We further extended and applied our algorithm to more residential room scenarios with different floor plans and settings, as we can see from Figure 8 to 13, we are obtaining similar results which could validate the functionality of our algorithm.

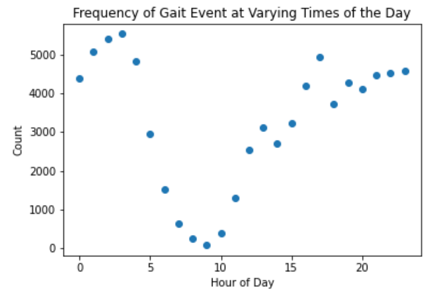


Figure 6: Frequency of gait events at varying times of the day

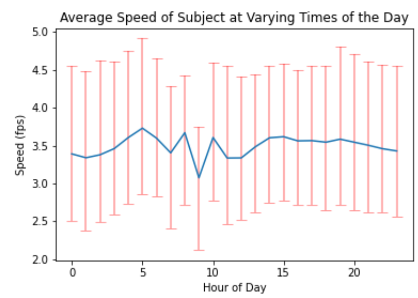
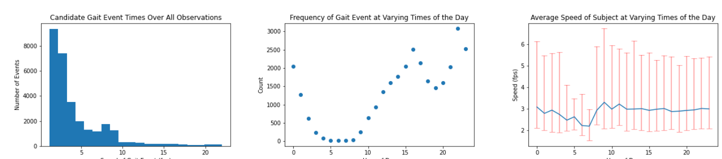
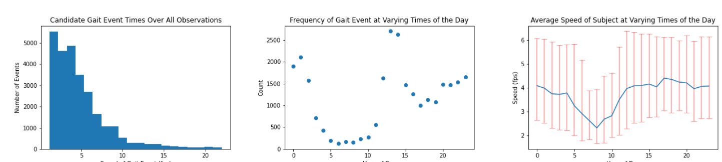


Figure 7: Average speed of subject at varying times of the day



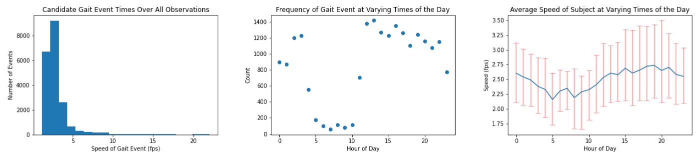
(a) Gait event times **(b) Frequency of events** **(c) Average speed**

Figure 8: Gait information of CEP001



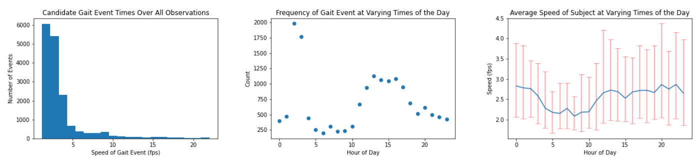
(a) Gait event times **(b) Frequency of events** **(c) Average speed**

Figure 9: Gait information of CEP002



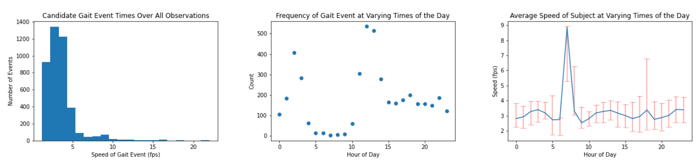
(a) Gait event times **(b) Frequency of events** **(c) Average speed**

Figure 10: Gait information of CEP003



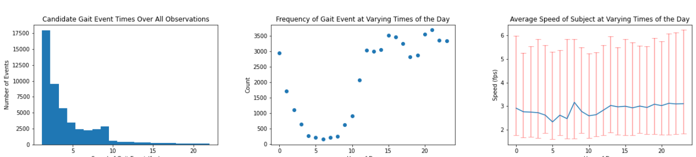
(a) Gait event times **(b) Frequency of events** **(c) Average speed**

Figure 11: Gait information of CEP004



(a) Gait event times **(b) Frequency of events** **(c) Average speed**

Figure 12: Gait information of CEP005



(a) Gait event times **(b) Frequency of events** **(c) Average speed**

Figure 13: Gait information of CEP006

5 FUTURE WORK

Future work may involve exploring more advanced algorithm for the processing of the data in order to correlate activations of the sensors with higher precision. Specifically, we hope to incorporate the room activation into a more sophisticated technique in a future iteration of our algorithm. Additionally, methods can be explored which are robust to multiple subjects walking through the hallway as well as pets and object such as smart vacuum cleaners. Moreover, we hope to apply our algorithm into multiple room scenarios with various settings such as floor plans, etc. Supervised learning approaches may also be applied to develop a system with more accurate predictive capabilities given ground truth readings of gait speed events.

6 CONCLUSIONS

In closing, we found that both approaches we created were very effective in removing noise from the data and strikingly effective in isolating true gait events. With this newly cleaned and isolated data were able to apply both the algorithms that we created to identify complete Gait events. With this information, we were able to generate statistical information and gain greater insight into the mobility of the elderly. It is our hope that this series of pipe-lining tools that we created will be useful in furthering the research of Gait Speed Measurement and analysis for health purposes. It is also the hope of the team that the research and finding will be able to help other areas of health research.

REFERENCES

[1] Paul E. Cuddihy et al. 2012. *Radar walking speed measurements of seniors in their apartments: Technology for fall prevention.* Engineering in Medicine and Biology Society (EMBC). https://doi.org/10.1109/EMBC.2012.6345919

[2] Stephanie Studenski et al. 2011. *Gait speed and survival in older adults.* Columbia University. https://doi.org/10.1001/jama.2010.1923

[3] Tarik Yardibi et al. 2011. *Gait characterization via pulse-Doppler radar.* Pervasive Computing and Communications Workshops (PERCOM Workshops). https://doi.org/10.1109/PERCOMW.2011.5766971

[4] Ryan S. McGinnis, Nikhil Mahadevan, Yaejin Moon, Kirsten Seagers, Nirav Sheth, John A. Wright, Steven DiCristofaro, Ikaro Silva, Elise Jortberg, Melissa Ceruolo, Jesus A. Pindado, Jacob Sosnoff, Roozbeh Ghaffari, and Shyamal Patel. 2017. *A machine learning approach for gait speed estimation using skin-mounted wearable sensors: From healthy controls to individuals with multiple sclerosis.* PLoS ONE. https://doi.org/10.1371/journal.pone.0178366

[5] Erik Stone and Marjorie Skubic. 2011. *Evaluation of an inexpensive depth camera for in-home gait assessment.* Journal of Ambient Intelligence and Smart Environments 3.4. https://doi.org/10.3233/AIS-2011-0124

[6] Erik Stone, Marjorie Skubic, Marilyn Rantz, Carmen Abbott, and Steve Miller. 2014. Average In-Home Gait Speed: Investigation of a New Metric for Mobility and Fall Risk Assessment of Elders. *Gait and Posture* 41 (09 2014). https://doi.org/10.1016/j.gaitpost.2014.08.019