HHS Public Access
Author manuscript
Assist Technol. Author manuscript; available in PMC 2017 July 01.

Published in final edited form as:
Assist Technol. 2016 ; 28(2): 105–114. doi:10.1080/10400435.2015.1095810.

A Novel Mobile-Cloud System for Capturing and Analyzing Wheelchair Maneuvering Data: A Pilot Study

Jicheng Fu, PhD[1,*], **Maria Jones, PT, PhD**[2], **Tao Liu, PhD**[1], **Wei Hao, PhD**[3], **Yuqing Yan, PhD**[4], **Gang Qian, PhD**[1], and **Yih-Kuen Jan, PT, PhD**[5]

[1]Department of Computer Science, University of Central Oklahoma, Edmond, OK, USA

[2]Department of Rehabilitation Sciences, University of Oklahoma Health Sciences Center, Oklahoma City, OK, USA

[3]Department of Computer Science, Northern Kentucky University, Highland Heights, KY, USA

[4]School of Informatics, Guangdong University of Foreign Studies, Guangzhou, China

[5]Departments of Kinesiology and Community Health, Computational Science and Engineering, University of Illinois at Urbana-Champaign, Champaign, IL, USA

Abstract

The purpose of this pilot study was to provide a new approach for capturing and analyzing wheelchair maneuvering data, which are critical for evaluating wheelchair users' activity levels. We proposed a mobile-cloud (MC) system, which incorporated the emerging mobile and cloud computing technologies. The MC system employed smartphone sensors to collect wheelchair maneuvering data and transmit them to the cloud for storage and analysis. A K-Nearest-Neighbor (KNN) machine-learning algorithm was developed to mitigate the impact of sensor noise and recognize wheelchair maneuvering patterns. We conducted 30 trials in an indoor setting, where each trial contained 10 bouts (i.e., periods of continuous wheelchair movement). We also verified our approach in a different building. Different from existing approaches that require sensors to be attached to wheelchairs' wheels, we placed the smartphone into a smartphone holder attached to the wheelchair. Experimental results illustrate that our approach correctly identified all 300 bouts. Compared to existing approaches, our approach was easier to use while achieving similar accuracy in analyzing the accumulated movement time and maximum period of continuous movement ($p >$ 0.8). Overall, the MC system provided a feasible way to ease the data collection process, and generated accurate analysis results for evaluating activity levels.

Keywords

Activity Level; Android; Cloud Computing; Google App Engine; Mobile Computing; Wheelchair Maneuver

*Correspondence and reprint requests to Dr. Jicheng Fu, Department of, Computer Science, University of Central Oklahoma, 100 North University Drive, Edmond, OK 73034, USA; jfu@uco.edu; telephone: 1-405-974-5704.

Declaration of interest
The authors report no conflicts of interest.

1. Introduction

Wheelchair maneuvering data are fundamental for studying wheelchair users' activity level (Tolerico et al., 2007), which is an important indicator of their quality of life and health status (Harris et al., 2010; Sonenblum et al., 2008). In addition, the characteristics obtained from wheelchair maneuvering data are critical for studying safety concerns, as wheelchair-related accidents can occur and some may lead to serious injuries (McClure et al., 2009; H. Wang et al., 2009).

One major research direction in capturing and analyzing wheelchair maneuvering data is to simplify the installation of data loggers and improve the efficiency of data collection. For example, in order to measure wheelchair maneuvering activities, a conventional approach is to attach magnets to the wheels and mount a reed switch to the wheelchair frame to record the number of times the magnets passing the reed switch (Cooper et al., 2002). Recently, the inertial sensors, such as accelerometers or gyroscopes, have been used to collect wheelchair maneuvering data (Coulter et al., 2011; Hiremath et al., 2013; Sonenblum et al., 2012). The use of inertial sensors simplifies the installation as it eliminates the need of mounting multiple magnets and reed switches to the wheelchair.

Although existing research has achieved great advancement, there are areas for improvement. For example, existing approaches require placing data loggers on the wheels of a wheelchair to collect wheelchair maneuvering data (Cooper et al., 2002; Cooper et al., 2008; Coulter et al., 2011; Harris et al., 2010; Hiremath et al., 2013; Sonenblum et al., 2012; Sonenblum et al., 2008; Tolerico et al., 2007). Since it is impractical to request wheelchair users to install the data loggers to the wheel, research personnel are needed to set up the experiments. The cost associated with research personnel and transportation (between the research lab and participants' homes) can be high. In addition, commercial inertial sensors may not fit in the wheels of pediatric wheelchairs due to the limited space of the wheels. Furthermore, except for Hiremath et al. (Hiremath et al., 2013), who reported using a gyroscope that could communicate with a smartphone through Bluetooth for data collection, the majority of the data loggers do not support real-time data transmission, making it difficult to provide timely feedbacks on wheelchair maneuvering characteristics (Hiremath et al., 2013).

To enrich existing research and address the aforementioned challenges, we propose a mobile and cloud computing-based (MC) system to capture, transmit, store, and analyze wheelchair maneuvering data. Specifically, the MC system has a mobile computing-based subsystem that employs smartphones to capture and transmit wheelchair maneuvering data, utilizing their ubiquitous nature, ever-increasing processing power, and rich set of sensors, such as accelerometers and gyroscopes. On the other hand, the MC system has a cloud computing-based subsystem that processes, stores, and analyzes wheelchair maneuvering data. This subsystem employs the cutting-edge cloud computing paradigm. The combination of mobile and cloud computing yields a distributed and integrated system, in which the mobile subsystem controls the sensors, collects wheelchair maneuvering data, and transmits it to the cloud, while the cloud subsystem handles the subsequent data management and analysis.

A main challenge associated with the MC system is that sensors in a smartphone are very sensitive to noise. We combined two approaches to mitigate the effect of noise on the collected wheelchair maneuvering data. First, we improved an existing noise reduction algorithm to preprocess noisy sensor data. Then, we analyzed the preprocessed data by employing a machine-learning algorithm, which could tolerate noise by recognizing the patterns of wheelchair maneuvers.

We performed experiments of the MC system using a power wheelchair in indoor settings. As a baseline, we also attached an ActiGraph GT3X accelerometer (ActiGraph, 2011) to each side of the wheels of the wheelchair. Hence, we could compare the analysis results obtained from our MC system and from the approach of existing research (Coulter et al., 2011; Sonenblum et al., 2012), which used data loggers attached to the wheels of the wheelchairs. Particularly, we analyzed the accumulated movement time, maximum period of continuous movement, and number of bouts (i.e., periods of continuous wheelchair movement) (Sonenblum et al., 2008), which are critical measures for evaluating a wheelchair user's activity level (Cooper et al., 2008; Harris et al., 2010; Sonenblum et al., 2008). Experimental results demonstrated that the MC system was capable of capturing wheelchair maneuvering data and automatically transmitting data to the cloud. Furthermore, the combination of the improved noise reduction algorithm and the machine-learning technique enabled the MC system to provide satisfactory analysis results, which were comparable to the existing research approach.

2. Methods

The proposed mobile-cloud (MC) system consists of two major subsystems, namely, the mobile computing-based data collection subsystem and the cloud computing-based data management and analysis subsystem, as illustrated in Figure 2. The mobile subsystem utilizes a smartphone to record wheelchair maneuvering data and periodically transmits data to the cloud subsystem. The cloud subsystem is responsible for processing, storing, and analyzing wheelchair maneuvering data. The analyzed results, such as the accumulated movement time, maximum continued movement period, number of starts and stops, etc., can then be made available to researchers or healthcare providers through the Internet.

2.1 Mobile-based Data Collection

The past decade has seen rapid development of smartphones, which has opened a new era of mobile computing (Zheng & Ni, 2006). The ever-increasing power in hardware, software, and communication empowers smartphones to play a critical role in the proposed MC system. In particular, since smartphones have a rich set of sensors, such as accelerometers, gyroscopes, compasses, GPS, etc., it is feasible to use smartphones to directly capture wheelchair maneuvering data and simplify the setup of the data collection process. We developed a mobile subsystem that controlled the built-in accelerometer of a smartphone to capture wheelchair maneuvering data. As illustrated in Figure 3, the accelerometer can record wheelchair accelerations in three dimensions (i.e., X, Y, and Z).

One characteristic of smartphone sensors is that they are event-driven, i.e., sensor data can be accessed whenever the sensors experience a change. Hence, the data sampling frequencies of the built-in sensors are not fixed. Accordingly, the mobile subsystem records the time points at which the corresponding sensor reads the data instances. When transmitting recorded data to the cloud subsystem, the amount of data to transmit is configurable in our mobile subsystem. Specifically, the mobile subsystem temporarily stores the data instances in the smartphone's memory and, when the number of data instances reaches a preset limit Δ (e.g., 1,000), they are transmitted to the cloud subsystem. Hence, if we preset a small value of Δ (e.g., 10), the MC system will work in a real- /near-real-time mode; otherwise, it will work in a batch mode. In the case of no Internet connection, our mobile subsystem can still collect wheelchair maneuvering data and organize them into a local file (i.e., a CSV file), which can be exported into the cloud later.

2.2 Cloud-based Data Management and Analysis

In the proposed MC system, the cloud plays an important role in storing, processing, and analyzing wheelchair maneuvering data.

2.2.1 Data storage—We used Google App Engine (GAE) (Google, 2011) as the cloud computing platform in our MC system. GAE allows web applications to be developed directly in the cloud. The cloud subsystem we developed is essentially a web application in GAE that interacts with the mobile subsystem. It responds to data transmission requests sent by our mobile subsystem, accepts transmitted maneuvering data, and saves them to the Google Blobstore, which is a data service that can store large data objects (e.g., 10,000 wheelchair maneuvering data instances) in GAE. Note that wheelchair users do not need to create their own GAE accounts to participate because the interaction between our mobile subsystem and cloud subsystem is transparent to the users. In the meantime, the cost of data storage is also affordable in GAE. We present more discussions on the cost associated with our MC system in the Discussion Section.

2.2.2 Data analysis—While it is possible to conduct a variety of analyses based on wheelchair maneuvering data, in this pilot study, we focus on developing approaches for analyzing the accumulated maneuvering time, the maximum period of continuous movement, and the number of bouts, which are critical measures for studying a wheelchair user's activity level (Cooper et al., 2008; Harris et al., 2010; Sonenblum et al., 2008). The accumulated maneuvering time and the maximum period of continued movement can reveal how the wheelchair is used (Tolerico et al., 2007). Bouts of mobility refer to periods of continuous wheelchair movement (Sonenblum et al., 2012; Sonenblum et al., 2008). The number of bouts can provide insightful information regarding the activities performed in different locations (Sonenblum et al., 2008).

The key to perform the aforementioned analyses is the ability to distinguish whether the wheelchair is moving or stationary. This is by no means easy owing to the presence of noise. As shown in Figure 4 (a), even if the phone is stationary (during the first half of the data curve), its accelerometer still generates readings due to the gravity, rotation of the earth, and/or environmental noise (Sonenblum et al., 2012). In this pilot study, we integrated two approaches to mitigate the impact of noise in order to accurately determine a wheelchair's status. In the following discussions, we use Figure 4 as a running example to illustrate how these two approaches work to overcome noise and determine wheelchair activities.

(1) Data pre-processing: We improved the Common Average Reference (CAR) algorithm (Ng & Raveendran, 2007), which is widely used in EEG (Electroencephalography), by employing a threshold-based strategy. The improved CAR (ICAR) algorithm is specifically designed to process wheelchair maneuvering data. Particularly, by observing raw wheelchair maneuvering data captured in indoor settings with flat floors, we noticed that the values of noise fluctuated within a certain range. To counteract such noise, our ICAR algorithm takes three steps. First, for each dimension d, it averages the sensor readings of the immobile period to obtain the averaged value $\overline{\alpha_d}$. Second, ICAR deducts all data instances in dimension d by $\overline{\alpha_d}$ to shift the curve toward its actual position. Finally, based on data obtained in the second step, ICAR determines a threshold τ such that a data instance α is reset to 0 if $|\alpha| < |\tau|$. The value of τ is determined if it can reset 95% or more of the data instances to 0 for the immobile period. We chose the value of 95% so that the majority of the noise could be removed. As an intuitive example, Figure 4 (a) shows a series of segments of raw data. The first half represents data collected in a stationary period, which is drifted mostly above zero. After the ICAR algorithm (with $\overline{\alpha_d}$ determined to be 0.1) is applied, the entire data curve is shifted toward its actual position, as shown in Figure 4 (b).

To effectively calculate $\overline{\alpha_d}$, our mobile subsystem employs the text-to-speech technique to use audible messages to remind the wheelchair user to stay still for 5 seconds in the beginning of data collection. $\overline{\alpha_d}$ is then calculated based on sensor data of the first 5 seconds. While the calculated $\overline{\alpha_d}$ is used by ICAR to preprocess all raw data points, we found that, as the flatness of the floor varies slightly at different locations, the noise-reducing effectiveness of $\overline{\alpha_d}$ is not uniform in all preprocessed data segments. As a result, certain data segments in a stationary period may still be classified as moving. We observed that the background noise tends to produce accelerations of the same values in a short period of time. In our implementation, we utilized this pattern to identify and fine-tune certain data segments. In particular, sensor data is partitioned into a series of data segments, each of which contains 10 data items. If a segment contains at least 3 identical items, we apply ICAR to this data segment individually, i.e., (1) calculating the averaged acceleration for this data segment; (2) deducting each item by the averaged acceleration; and (3) resetting data items whose values are smaller than a threshold to 0.

Although ICAR can largely reduce noise, it cannot completely eliminate noise. Hence, we utilize the machine-learning technique over the ICAR processed data to recognize wheelchair maneuvering patterns.

(2) Data analysis with the machine-learning technique: We propose to use the machine-learning algorithm, k-nearest neighbors (KNN), to classify wheelchair maneuvers. KNN is a widely used classification algorithm due to its simplicity and effectiveness (Dhurandhar & Dobra, 2013; Shang et al., 2005). The use of KNN greatly enhances our MC system's noise tolerance by recognizing the patterns of wheelchair maneuvers. Specifically, we design the KNN algorithm to perform binary classification: it classifies a given data segment into one of the two classes, namely, moving or stationary. The classification decision is made based on the majority class of the data segment's K nearest neighbors. This approach fits in the application because we can adjust the parameter K to mitigate the impact of noise.

One difficulty is that the stationary and the linear constant speed maneuvers are theoretically indistinguishable because the accelerations for both maneuvers should be 0. Hence, KNN should not be directly applied to the raw data items collected by the accelerometer in the smartphone. In practice, after the ICAR is applied to the raw accelerations, we observed that a stationary maneuver contains less non-zero data items that also have smaller values than those in a linear constant speed maneuver. Hence, we take two steps to convert the raw data items into data vectors that can facilitate the accurate classification of maneuvers. First, we partition the raw data items into a series of data segments that have an equal length of 10. If the size of the last segment is less than 10, it will be discarded. This will not cause a loss of accuracy because a data segment of size 10 only corresponds to 0.59 ~ 0.71 second (i.e., the sampling rate is SENSOR_DELAY_UI (14 ~ 17 Hz)). Second, we convert every data segment of size 10 into a 7-tuple vector $\langle f_1, f_2, f_3, f_4, f_5, f_6, f_7\rangle$, namely, the number of positive accelerations (f_1), the sum of the positive accelerations (f_2), the number of positive accelerations greater than a threshold (e.g., > 0.3) (f_3), the number of zeros (f_4), the number of negative accelerations (f_5), the sum of the negative accelerations (f_6), and the number of negative accelerations less than a threshold (e.g., < −0.3) (f_7). Figure 4 (c) shows a data vector (0, 0, 0, 0, 10, −6.5599, 10), which is derived from the 11$^{\text{th}}$ data segment in the series. The first four items (f_1 to f_4) are 0s because all accelerations in this data segment are negative. The first "10" means that there are 10 negative accelerations. The sum of the negative accelerations is −6.5599, and the number of negative accelerations less than a threshold (i.e., −0.3 in this example) is 10.

To measure the distance between a sample vector $S_i = (f_1^{\,i}, f_2^{\,i}, f_3^{\,i}, f_4^{\,i}, f_5^{\,i}, f_6^{\,i}, f_7^{\,i})$ ($i = 1, 2, \ldots, m$, with m being the number of sample vectors) and a testing vector $T_j = (f_1^{\,j}, f_2^{\,j}, f_3^{\,j}, f_4^{\,j}, f_5^{\,j}, f_6^{\,j}, f_7^{\,j})$ ($j = 1, 2, \ldots, n$, with n being the number of testing vectors), we employ the Euclidean distance as follows:

$$\sqrt{\sum_{t=1}^{7}\left(f_t^i - f_t^j\right)^2}$$

where f_t^i is the factor t ($t = 1, \ldots, 7$) in a sample data vector S_i and f_t^j is the factor t in a testing data vector T_j.

For each class (moving or stationary), we prepared 36 sample data vectors. The choice of 36 is to ensure that these sample data can cover possible situations in its class. For example, the sample data for the moving class includes data vectors of acceleration, deceleration, and constant speed for both linear and turning maneuvers (i.e., 6 moving maneuvers in total). Then, we collected both typical and boundary samples for each of the moving maneuvers (e.g., linear acceleration, turning deceleration, etc.). In addition, we tested K = 1, 3, 5, and 7 for KNN. We chose odd numbers for the K values to avoid ties in voting for the majority because the classification is binary, i.e., stationary or moving. In addition, we considered 7 types of maneuvers including 6 moving maneuvers and 1 stationary maneuver. Hence, we could limit the K value to be no bigger than 7. Figure 4 (c) and (d) demonstrate how the KNN works using $K = 3$. For the 11th data vector in Figure 4 (c), its 3 nearest neighbors all belong to the moving class. Hence, it is classified as moving.

2.3 Protocol for Data Collection

To collect wheelchair maneuvering data, the smartphone will be attached to the smartphone holder as shown in Figure 1. The use of the smartphone holder provides a convenient way to use our mobile subsystem. The wheelchair user can manipulate the wheelchair as usual while being able to see the smartphone. The interface of the mobile subsystem contains an arrow with descriptions, showing the direction of the smartphone orientation. The X-axis of the smartphone is aligned with the wheelchair moving direction (see Figure 1 and Figure 3). The text-to-speech technique is also incorporated to guide the user to properly set up the system. Specifically, when the function of data collection is turned on, the mobile subsystem will use the audible message to remind the wheelchair user, i.e., “Please remain stationary for 5 seconds”. When five seconds have passed, it will tell the wheelchair user “Now, you are ready to maneuver the wheelchair”.

As a baseline, we also attached an ActiGraph GT3X accelerometer (ActiGraph, 2011) on each side of the wheels of the wheelchair. In this way, we could use the approach employed in existing research (i.e., attaching data loggers to the wheels) (Coulter et al., 2011; Sonenblum et al., 2012) to analyze GT3X data. We could then compare analysis results based on data collected by GT3X accelerometers with those collected by our MC system. Note that we chose not to apply noise filtering to the sensor data from the GT3X sensors in our experiments. Since the GT3X sensors were attached to the wheels of the wheelchair, their sensor data resembled a clear sinusoid as the wheels rotated. This strong data pattern made the GT3X sensors resistant to the impact of the background noise.

2.4 Experiments

We implemented the proposed MC system using Google’s platforms, i.e., Android + Google App Engine (GAE). The smartphone we used was a Samsung Galaxy SII (GT-I9100) with Android OS 4.1 Jelly Bean. The built-in sensor, i.e., accelerometer, was used to capture wheelchair maneuvering data. As discussed previously, the built-in sensor is event-driven so that its data sampling frequencies are not fixed. Android APIs offer four data sampling settings; each corresponds to a range of data sampling frequencies. For example, if a phone app needs to obtain sensor data as frequent as possible, it should choose the option of SENSOR_DELAY_FASTEST, with a sampling frequency falling within 96 ~ 100 Hz for the Samsung Galaxy SII smartphones. Other options in the descending order of sampling rates are SENSOR_DELAY_GAME, SENSOR_DELAY_UI, and SENSOR_DELAY_NORMAL (Zhao et al., 2010). In our experiments, we chose a slower option, i.e., SENSOR_DELAY_UI (14 ~ 17 Hz), because we found it adequate for our data analyses while consuming less battery power (Liu et al., 2015). During the data collection period, our mobile subsystem automatically transmitted data to the cloud whenever 10,000 data instances were recorded until the data collection was finished.

We used an Invacare® Formula CG power wheelchair in the study. The experiments were conducted inside an academic building on campus. Since our focus was on indoor settings, we used short bouts that lasted for 7 to 24 seconds, during which the wheelchair moved 5 to 26 meters. The wheelchair speed was set to indoor mode with the peak speed at about 1.2 meter/s. All data was collected by the same driver. In addition, we conducted another set of experiments to evaluate whether our approach could achieve satisfactory results with the same training dataset when driving inside a different building, i.e., evaluating the generalization capability of our approach. In these experiments, we set up a fixed route with waypoints indicating where the wheelchair should turn in a bout. Then, we used a timer to record the duration for each bout so that we could compare the accuracies achieved by the GT3X and KNN. In this study, all statistical tests were performed using Excel Data Analysis Tools at the significance level of 0.05.

3. Results

3.1 Analysis of Bouts

We followed the approach discussed in Section 2.2.2 to use the KNN to analyze the number of bouts. A bout is identified by determining continuous moving data segments that are sandwiched in between two consecutive stationary data segments. In our experiments, we conducted 30 trials, each containing 10 bouts. As illustrated in Table 1, except $K = 1$, our approach accurately determined all the bouts (for $K = 3$, 5, and 7). This result suggested that KNN correctly distinguished the stationary segments from the moving ones so that the correct number of bouts was obtained. This experiment demonstrated that data collected by the built-in sensor of a smartphone could lead to accurate analysis results.

3.2 Analysis of Accumulated Maneuvering Time

After a bout is recognized, the time spent within the bout can also be determined since all the collected data items were time-stamped (see Section 2.1). By summing up the time spent on bouts, we could analyze the accumulated maneuvering time. Since $K = 1$ may not determine the correct number of bouts, we used $K = 3$, 5, and 7 in this experiment. As shown in Table 2, we considered the average time, standard deviation (SD), standard error (SE), and the p-value measured by using the one-way ANOVA. The experimental results of the accumulated maneuvering time obtained by our MC system were very close to those calculated using data from the GT3X accelerometers in each trial. The differences between the use of GT3X and our approach were statistically insignificant (the p-values varied from 0.751 to 0.868). When K was set to 3, our approach achieved the most similar result measured by the SD, SE, and p-value.

3.3 Analysis of Maximum Continued Maneuvering Time

The maximum continued maneuvering time is the time of the bout that has the longest maneuvering duration. Table 3 illustrates the analysis results for the maximum continued maneuvering time. We considered the average time, standard deviation (SD), standard error (SE), and the p-value measured by using the one-way ANOVA. Once again, our approach achieved similar analysis results to those based on GT3X. The differences were statistically insignificant (the p-values varied from 0.801 to 0.863).

3.4 Analysis of Bouts in a Different Building

In this experiment, we used the same training dataset for KNN to classify data collected in a different building. As shown in Table 4, our approach achieved satisfactory accuracy in calculating the wheelchair maneuvering time. The difference between the KNN and the timer was insignificant (the *p*-values varied from 0.18 to 0.32 measured by using the one-way ANOVA). When *K* was set to 3, the KNN achieved the best result measured by the *p*_value. We could also see that the results obtained by using the GT3X sensor were slightly better than those by KNN. However, the difference was not statistically significant (the *p*-values varied from 0.54 to 0.81).

4. Discussion

Mobile cloud computing is an emerging paradigm, which has attracted significant research efforts, such as augmented execution, elastic applications, and migration optimization (Han & Gani, 2012). In this pilot study, we demonstrated that our MC system is promising in advancing research on capturing and analyzing wheelchair maneuvering data, which can either be used independently or complement existing approaches. First, our MC system substantially simplifies the installation of data loggers and makes possible the timely feedback on wheelchair maneuvering characteristics. Specifically, the MC system does not need external data loggers. Either the wheelchair user or a family member can set up the sensor/data logger, since no specialized skills are required as it only takes a single step to complete, i.e., attaching a smartphone to the smartphone holder as shown in Figure 1. The use of the smartphone itself to collect wheelchair maneuvering data is more feasible owing to the smartphone's rich set of sensors and communication capabilities (WIFI, 3G/4G networks, Bluetooth, etc.). The MC system can transmit recorded data to the cloud subsystem in either a real- /near-real-time mode or a batch mode by configuring the preset limit on the number of data instances to transmit. The real- /near-real-time mode can provide fast feedbacks on wheelchair maneuvers, but it may consume a large amount of battery power because WIFI or 3G/4G network connection is power-consuming (Crk et al., 2009). The batch mode, on the other hand, has the advantage of preventing smartphones from frequent network connections, which saves battery power consumption of the phone.

Second, our MC system largely simplifies data management. Current research typically uses data loggers to continuously collect wheelchair maneuvering data for one or two weeks. The large amount of data generated by data loggers makes data management very challenging. For example, if a data logger is set to collect data at a frequency of 15 Hz, it will generate about 9 million data instances in one week. Manually downloading and managing 9 million data instances from data loggers may be cumbersome. Collecting and managing data from multiple wheelchair users can be even more difficult. In comparison, our MC system can automate data transmission between the mobile and cloud subsystems and hence save much effort on data management.

Third, the cost associated with the MC system compares favorably with those of existing approaches. Up to May 2013, the majority of adults (56%) in the U.S. own a smartphone (Smith, 2013). We can reasonably assume that a large number of wheelchair users can use the MC system with their own smartphones. Even if the researchers need to provide smartphones, the cost still compares favorably with that of the commercial data loggers. For example, a smartphone of $200 is sufficient for the MC system. In comparison, the prices of accelerometers commonly used in research vary from $225 (ActiGraph, 2011) to more than $600 (PalTechnologies, 2013). The cost will grow even higher if multiple data loggers are used (e.g., accelerometers and gyroscopes), let alone the cost of software used for managing the data loggers. For the cloud subsystem, the cost for data storage is also affordable. For example, GAE offers 5GB of free storage space for a regular user account and the price for additional storage is $0.13/GB for one month (Google, 2011). If the MC system stores the 9 million data instances (as discussed in the previous example) in GAE, the cost will be less than $0.13 for each wheelchair user because a single data instance in our study occupies only 0.1 KB and the 9 million data instances only need a space of 0.9 GB. In terms of the communication cost, statistics show that the majority of households (61%) in the U.S. own a WIFI network (Business-Wire, 2012). Hence, our mobile subsystem will most likely be able to use the wheelchair user's private WIFI network to communicate with the cloud subsystem. In case when a WIFI network is unavailable, the researchers will still have two options: (1) reimburse the expense on the Internet data usage (the cell phone carriers usually charge $10/GB or less (AT&T, 2013)), or (2) use the local storage option of the MC system to store sensor data locally in the smartphone (data can be manually uploaded into the cloud later).

One challenge we faced in the development of the MC system was to deal with the noise generated by the sensors, especially the accelerometer, in the smartphone. As discussed earlier, even when the wheelchair was stationary, the accelerometer still recorded small acceleration values in the range of −0.2 to −0.5 m/s^2. As a result, it is even difficult to determine whether a wheelchair is stationary or moving. The Kalman filter (KF) is a widely used algorithm for handling noisy inputs and generating statistically precise estimates of the underlying system state (Welch & Bishop, 1995). For example, the KF was applied to shoe-mounted inertial sensors (i.e., accelerometer and gyroscope) to correct the velocity error (Foxlin, 2005). Also, due to its light computation nature, the KF was widely used in indoor localization. Particularly, the KF was used to "fuse" multiple techniques, such as the WiFi-based, pseudo-odometry measurements, and/or pedestrian dead reckoning techniques, to achieve better localization accuracy (Chen et al., 2015; J. Wang et al., 2015). We have studied the KF in processing the accelerometer data. The KNN was then applied to the KF-processed data for classification. As shown in row 1 of Table 5, the classification accuracy of KNN was low if the KF was applied directly to the raw sensor data. This is because the KF could not remove the background noise, which shifted the entire data curve away from its actual position. If the KF was applied to data with background noise removed, the classification accuracy was improved for $K = 1$ and 3 as shown in Row 2 of Table 5. Comparing Table 1 and Table 5, we can see that the performance of KNN dropped when it was applied to the KF-processed data. The reason is that the KF tended to shrink the data curves in height and widen the curve in length (i.e., move a data item to a later time point). However, the proposed 7-tuple scheme needs to count the number of positive/negative accelerations and add up their values. The changes in data curves by the KF made it difficult for KNN to capture the right patterns. In comparison, the ICAR algorithm does not alter the characteristics of wheelchair maneuvering data. Table 1 shows that our approach could accurately recognize all the bouts. Table 2 and Table 3 illustrate that our approach achieved satisfactory results in analyzing the accumulated maneuvering time and the maximum continued maneuvering time, which were comparable to the approach employed by the existing research. We also verified our approach in a different building by using the same set of training data. As shown in Table 4, our approach could generalize to achieve satisfactory accuracy. Hence, these experimental results demonstrate that the proposed MC system is feasible in collecting wheelchair maneuvering data and generating accurate analysis results.

Another challenge faced by the MC system is the limited battery life of a smartphone. We address this problem with two options. First, a wheelchair can be equipped with a mounting and charging kit, which can provide power supply while firmly holding the smartphone (Broaden-Horizons, 2015). Second, a portable battery charger could also provide a financially affordable solution. A portable charger has a small size (similar to or smaller than a smartphone) and is easy to carry. It will enable the wheelchair user to use our MC system while charging the smartphone.

Besides inertial sensors, researchers have attempted to use other means to capture wheelchair maneuvering data. Bitsch et al (Bitsch Link et al., 2012) tried to estimate wheelchair speed by analyzing video streams captured by a smartphone's camera. This approach may not be practical because taking videos consumes a significant amount of battery power. Also, this approach is not as balanced as our MC system since all processing and analyses are conducted solely by the smartphone. Our MC system is not subject to these potential problems due to its distributed workload between the mobile and cloud subsystems.

Nowadays, smartphones have been increasingly used to monitor a person's health status. For example, all mainstream mobile operating systems, such as iOS, Android, and Windows, have built-in apps for monitoring a person's activity by counting the number of steps. This is achieved through step detection and step length estimation (Link et al., 2011; Microsoft-Research). Unfortunately, these smartphone apps are designed for healthy people. The dynamics of a power wheelchair do not possess the strong characteristics of steps. For example, the acceleration and deceleration periods of a wheelchair are usually short (less than 3 seconds). The subtle changes in maneuvers as well as the noise in maneuvering data compound the difficulty in determining maneuvering types. Therefore, our approach has enriched the research by expanding the scope of using smartphones as mobile sensors to wheelchair users.

Other related work includes QMedic (QMedic, 2014) and the research projects in the mHealth research group ("mHealth Research Group", 2012). QMedic targets at seniors who are at risk of falling or other dangers at home. Through a wearable bracelet that wirelessly communicates with a base station installed in the home, the system can communicate with QMedic representatives through the landline phone outlet if an emergency occurs. The projects at mHealth employ mobile computing techniques to deal with a wide range of health topics, such as weight management for young adults, African-American children's sleep, active transportation in urban areas, etc. Comparing with the existing systems, our MC system is unique in several aspects, including the research goal, targeted users, methods, and equipment.

In summary, we propose a new approach to conveniently collect, store, and analyze wheelchair maneuvering data. Our approach will enrich research for collecting wheelchair maneuvering data and analyzing wheelchair users' activity levels.

4.1 Study Limitations

In this pilot study, we conducted experiments with a Samsung Galaxy SII smartphone, which was attached in a smartphone holder. The use of smartphone holder avoids affecting the manipulation of the wheelchair while allowing the wheelchair user to see the smartphone. The arm of the holder is sturdy and bendy. The smartphone may vibrate slightly while the wheelchair is moving. In the next step, we will investigate how the rigidness of mounts will impact the analysis accuracy by testing mounts of different rigidness. The rigidness may also yield different data patterns when a wheelchair user stops to type on the phone. We will comprehensively consider screen touch events and employ additional sensors, such as the proximity sensor, to identify stationary maneuvers with noise, such as bumping into the phone, typing on the phone, etc. In addition, we will study other ways to hold the smartphone, such as putting the smartphone into the wheelchair user's shirt/pant pocket.

In this study, the sampling frequency of data collection was fixed to SENSOR_DELAY_UI. In a separate study (Liu et al., 2015), we analyzed the effect of different settings of the sampling frequency. Our preliminary study shows that higher sampling rates achieved better analysis accuracy, but consumed more battery power. Subsequent work is needed to study the impact when different brands of smartphones are used. In addition, the experiments were conducted in indoor settings, where the floor was flat without up and down variations, such as ramps. As a next step, we will conduct experiments on different floor surfaces and in more complex indoor settings. We will also study how to further balance the data storage and processing between the mobile and cloud subsystems to improve performance and reduce power consumption of smartphones.

5. Conclusion

We proposed a mobile and cloud computing-based (MC) system for capturing and analyzing wheelchair maneuvering data in this study. We developed a mobile-computing subsystem that controls the smartphone sensors, collects sensor data, and periodically transmits recorded data to the cloud. We also developed a cloud-computing subsystem that controls data storage and analysis. To determine wheelchair maneuvering states, we improved the Common Average Reference algorithm to reduce noise in raw accelerometer data. We then employed the machine-learning algorithm, KNN, to further mitigate the effect of noise by recognizing wheelchair maneuvering patterns. Experimental results demonstrated that our MC system could simplify data collection by allowing the wheelchair user or his/her family members to easily set up the system. The MC system can also improve data collection efficiency by automating data transmissions from the mobile subsystem to the cloud subsystem. Furthermore, the MC system produced accurate results for analyzing the number of bouts, the accumulated maneuvering time, and the maximum continued maneuvering time. Since mobile computing and cloud computing are two dynamic areas undergoing rapid development, as more and more sensors are being added to smartphones and the cloud is becoming more efficient and convenient to use, we see great potential in the proposed MC system for depicting a timely and comprehensive picture of a wheelchair user's daily activities.

Acknowledgments

This work was supported by the National Institute of General Medical Sciences of the National Institutes of Health through Grant Number 8P20GM103447 and the Oklahoma Center for the Advancement of Science and Technology (OCAST HR12-036).

References

ActiGraph. 2011. from http://www.theactigraph.com/

Armbrust, Michael; Fox, Armando; Griffith, Rean; Joseph, Anthony D.; Katz, Randy; Konwinski, Andy; Zaharia, Matei. A view of cloud computing. Commun. ACM. 2010; 53(4):50–58. doi: 10.1145/1721654.1721672.

AT&T. Data Plan. 2013. from http://www.att.com/shop/wireless/plans/dataplans.html

Bitsch Link, JA.; Gerdsmeier, F.; Smith, P.; Wehrle, K. Indoor navigation on wheels (and on foot) using smartphones; Paper presented at the Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on; 2012, 13–15; 2012 Nov.

Broaden-Horizons. Power2Go Dual USB Phone & Tablet 3A Rapid Charger. 2015. from http://www.broadenedhorizons.com/phone-mounting-charging-kit-custom

Business-Wire. Strategy Analytics: A Quarter of Households Worldwide Now Have Wireless Home Networks. 2012. from http://www.businesswire.com/news/home/20120404006331/en/Strategy-Analytics-Quarter-Households-Worldwide-Wireless-Home

Chen Z, Zou H, Jiang H, Zhu Q, Soh YC, Xie L. Fusion of WiFi, smartphone sensors and landmarks using the Kalman filter for indoor localization. Sensors (Basel). 2015; 15(1):715–732. [PubMed: 25569750]

Cooper RA, Thorman T, Cooper R, Dvorznak MJ, Fitzgerald SG, Ammer W, Boninger ML. Driving characteristics of electric-powered wheelchair users: how far, fast, and often do people drive? Arch Phys Med Rehabil. 2002; 83(2):250–255. [PubMed: 11833031]

Cooper RA, Tolerico M, Kaminski BA, Spaeth D, Ding D, Cooper R. Quantifying wheelchair activity of children: a pilot study. Am J Phys Med Rehabil. 2008; 87(12):977–983. [PubMed: 19033759]

Coulter EH, Dall PM, Rochester L, Hasler JP, Granat MH. Development and validation of a physical activity monitor for use on a wheelchair. Spinal Cord. 2011; 49(3):445–450. [PubMed: 20856261]

Crk I, Albinali F, Gniady C, Hartman J. Understanding energy consumption of sensor enabled applications on mobile phones. Conf Proc IEEE Eng Med Biol Soc. 2009; 2009:6885–6888. [PubMed: 19964448]

Dhurandhar, Amit; Dobra, Alin. Probabilistic characterization of nearest neighbor classifier. International Journal of Machine Learning and Cybernetics. 2013; 4(4):259–272.

Foxlin E. Pedestrian Tracking with shoe-mounted inertial sensors. IEEE Comput Graph Appl. 2005; 25(6):38–46. [PubMed: 16315476]

Google. Google App Engine. 2011. from http://code.google.com/appengine/

Han, Qi; Gani, A. Research on mobile cloud computing: Review, trend and perspectives; Paper presented at the Digital Information and Communication Technology and it's Applications (DICTAP), 2012 Second International Conference on; 2012, 16–18; May. 2012

Harris F, Sprigle S, Sonenblum SE, Maurer CL. The participation and activity measurement system: an example application among people who use wheeled mobility devices. Disabil Rehabil Assist Technol. 2010; 5(1):48–57. [PubMed: 19941440]

Hiremath SV, Ding D, Cooper RA. Development and evaluation of a gyroscope-based wheel rotation monitor for manual wheelchair users. J Spinal Cord Med. 2013; 36(4):347–356. [PubMed: 23820150]

Link, JAB.; Smith, P.; Viol, N.; Wehrle, K. FootPath: Accurate map-based indoor navigation using smartphones; Paper presented at the Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on; 2011, 21–23; 2011 Sep.

Liu, Tao; Chen, Chuanwei; King, Melicent; Qian, Gang; Fu, Jicheng. Balancing Power Consumption and Data Analysis Accuracy through Adjusting Sampling Rates: Seeking for the Optimal Configuration of Inertial Sensors for Power Wheelchair Users; Los Angeles, California, USA. Paper presented at the 17th International Conference on Human-Computer Interaction; 2015.

McClure, Laura A.; Boninger, Michael L.; Oyster, Michelle L.; Williams, Steve; Houlihan, Bethlyn; Lieberman, Jesse A.; Cooper, Rory A. Wheelchair Repairs, Breakdown, and Adverse Consequences for People With Traumatic Spinal Cord Injury. Arch Phys Med Rehabil. 2009; 90(12):2034–2038. [PubMed: 19969165]

mHealth Research Group. 2014, from http://mhealth.ccs.neu.edu/

Microsoft-Research. Mobile Indoor Localization. 2013. from http://research.microsoft.com/en-us/projects/indoorloc/

Ng, SiewCheok; Raveendran, P. Comparison of different Montages on to EEG classification. In: Ibrahim, F.; Osman, N.; Usman, J.; Kadri, N., editors. 3rd Kuala Lumpur International Conference on Biomedical Engineering 2006. Vol. 15. Springer Berlin Heidelberg; 2007. p. 365-368.

PalTechnologies. 2013. from http://providingtheevidence.com/PAL/products/

QMedic. QMedic 2014. 2014. from http://www.qmedichealth.com/

Shang, Wenqian; Huang, Houkuan; Zhu, Haibin; Lin, Yongmin; Wang, Zhihai; Qu, Youli. An Improved kNN Algorithm – Fuzzy kNN. In: Hao, Y.; Liu, J.; Wang, Y.; Y.-Cheung, m; Yin, H.; Jiao, L.; Ma, J.; Y.-Jiao, C., editors. Computational Intelligence and Security. Vol. 3801. Springer Berlin Heidelberg; 2005. p. 741-746.

Smith, A. Smartphone Ownership – 2013 Update. 2013. from http://www.pewinternet.org/~/media/Files/Reports/2013/PIP_Smartphone_adoption_2013.pdf

Sonenblum SE, Sprigle S, Caspall J, Lopez R. Validation of an accelerometer-based method to measure the use of manual wheelchairs. Med Eng Phys. 2012; 34(6):781–786. [PubMed: 22698978]

Sonenblum SE, Sprigle S, Harris FH, Maurer CL. Characterization of power wheelchair use in the home and community. Arch Phys Med Rehabil. 2008; 89(3):486–491. [PubMed: 18295627]

Tolerico ML, Ding D, Cooper RA, Spaeth DM, Fitzgerald SG, Cooper R, Boninger ML. Assessing mobility characteristics and activity levels of manual wheelchair users. J Rehabil Res Dev. 2007; 44(4):561–571. [PubMed: 18247253]

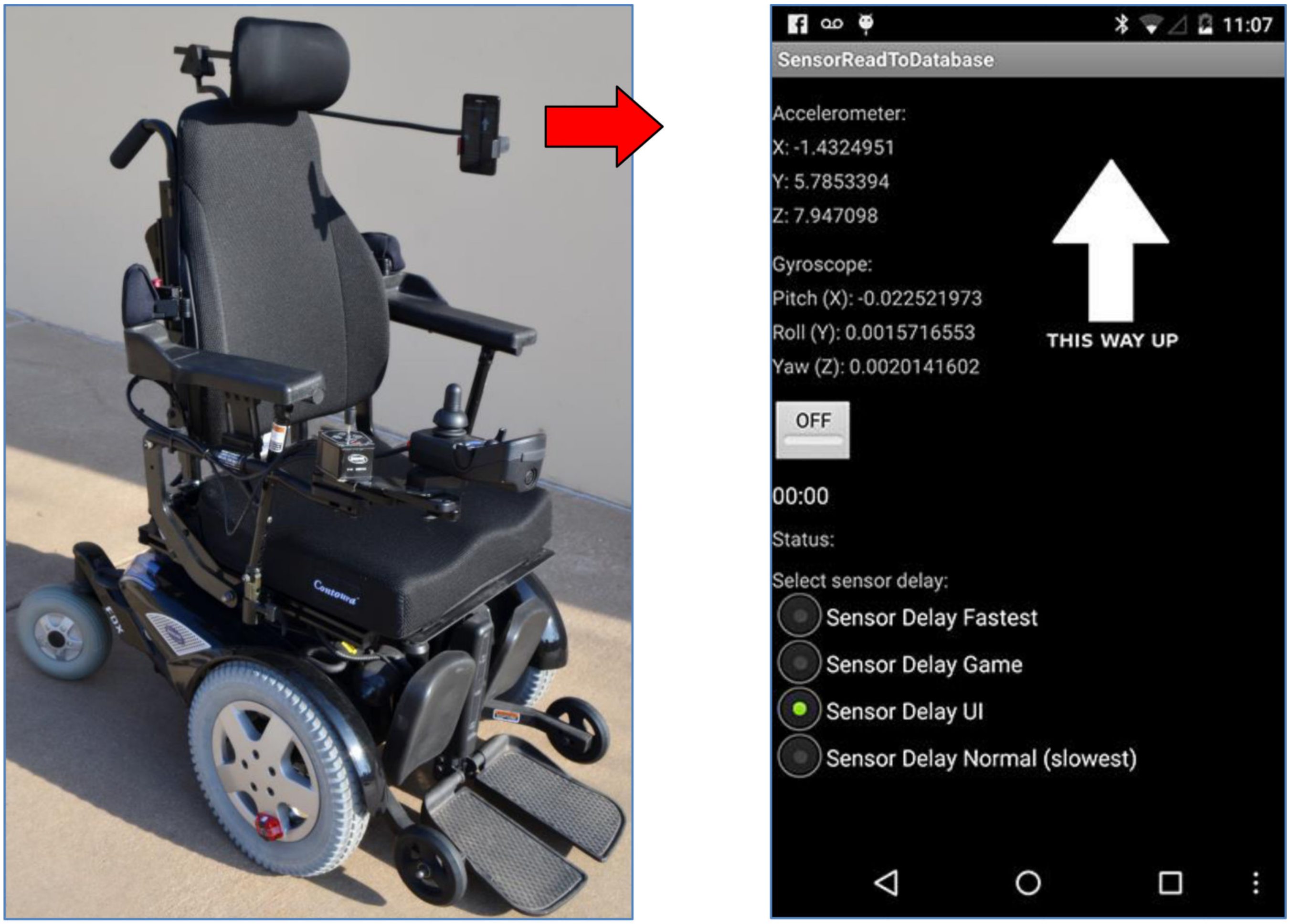
Wang H, Salatin B, Grindle GG, Ding D, Cooper RA. Real-time model based electrical powered wheelchair control. Med Eng Phys. 2009; 31(10):1244–1254. [PubMed: 19733494]

Wang J, Hu A, Liu C, Li X. A floor-map-aided WiFi/pseudo-odometry integration algorithm for an indoor positioning system. Sensors (Basel). 2015; 15(4):7096–7124. [PubMed: 25811224]

Welch, G.; Bishop, G. An Introduction to the Kalman Filter. 1995. 2012, from http://www.cs.unc.edu/~welch/kalman/kalmanIntro.html

Zhao, Zhongtang; Chen, Yiqiang; Liu, Junfa; Liu, Mingjie. Cross-mobile ELM based Activity Recognition. IJEI: International Journal of Engineering and Industries. 2010; 1(1):30–40.

Zheng, Pei; Ni, Lionel. Smart Phone and Next Generation Mobile Computing. Elsevier: Morgan Kaufmann ; Elsevier Science [distributor]; 2006.



(a) Experimental Configuration

(b) Screenshot of the mobile subsystem

Figure 1.
The Mobile Subsystem of the MC System

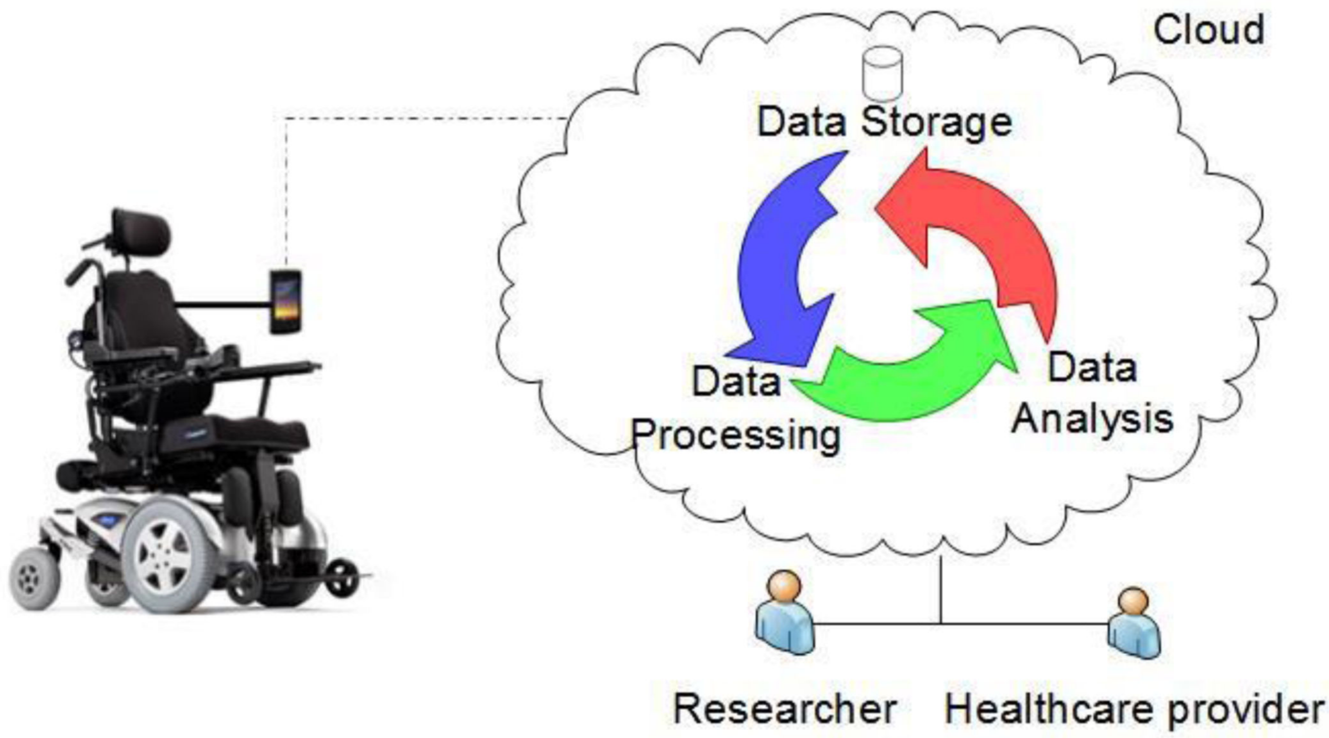


Figure 2.
The Architecture of the MC System

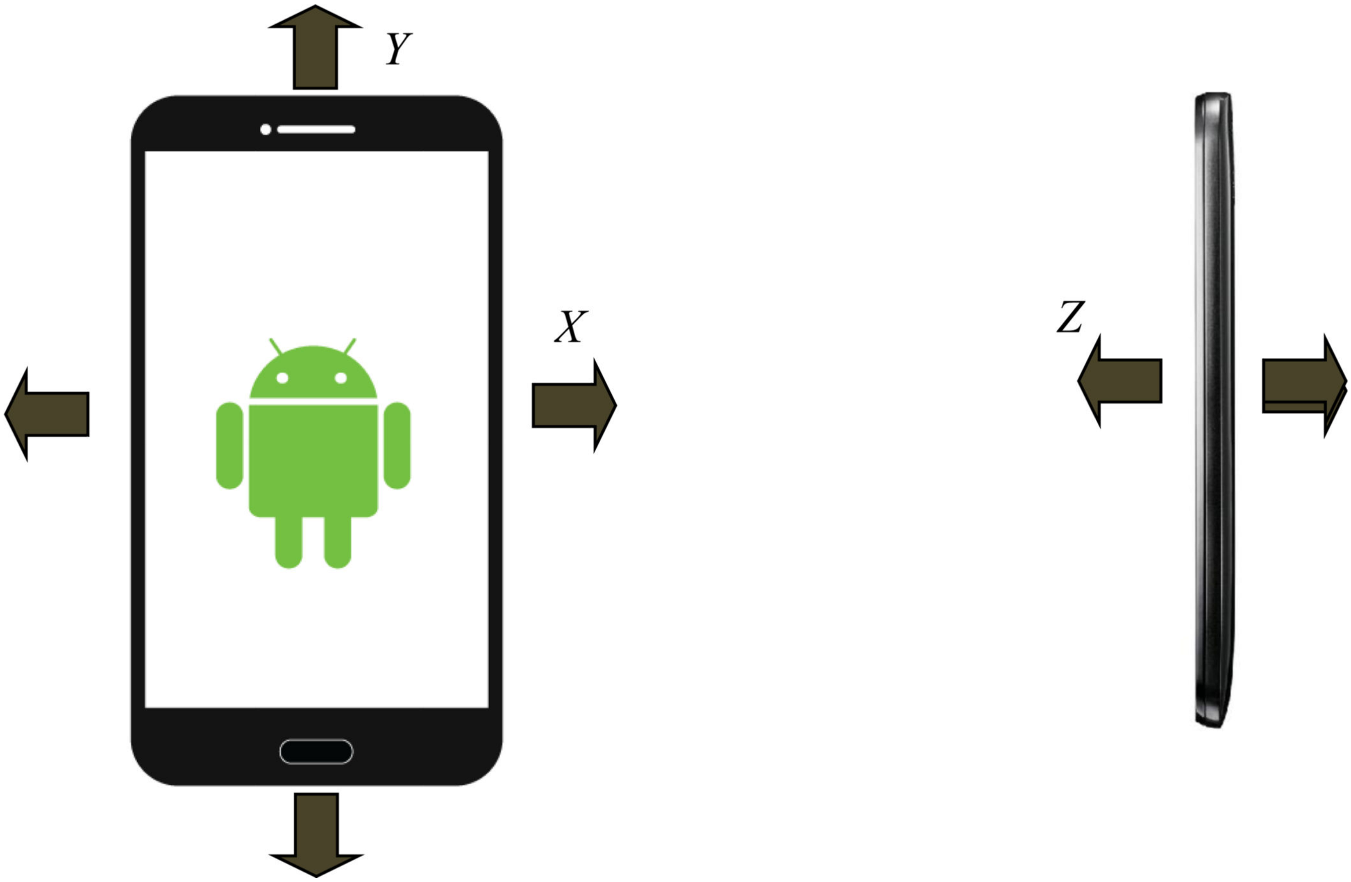


Figure 3.
Three Dimensions of a Smartphone

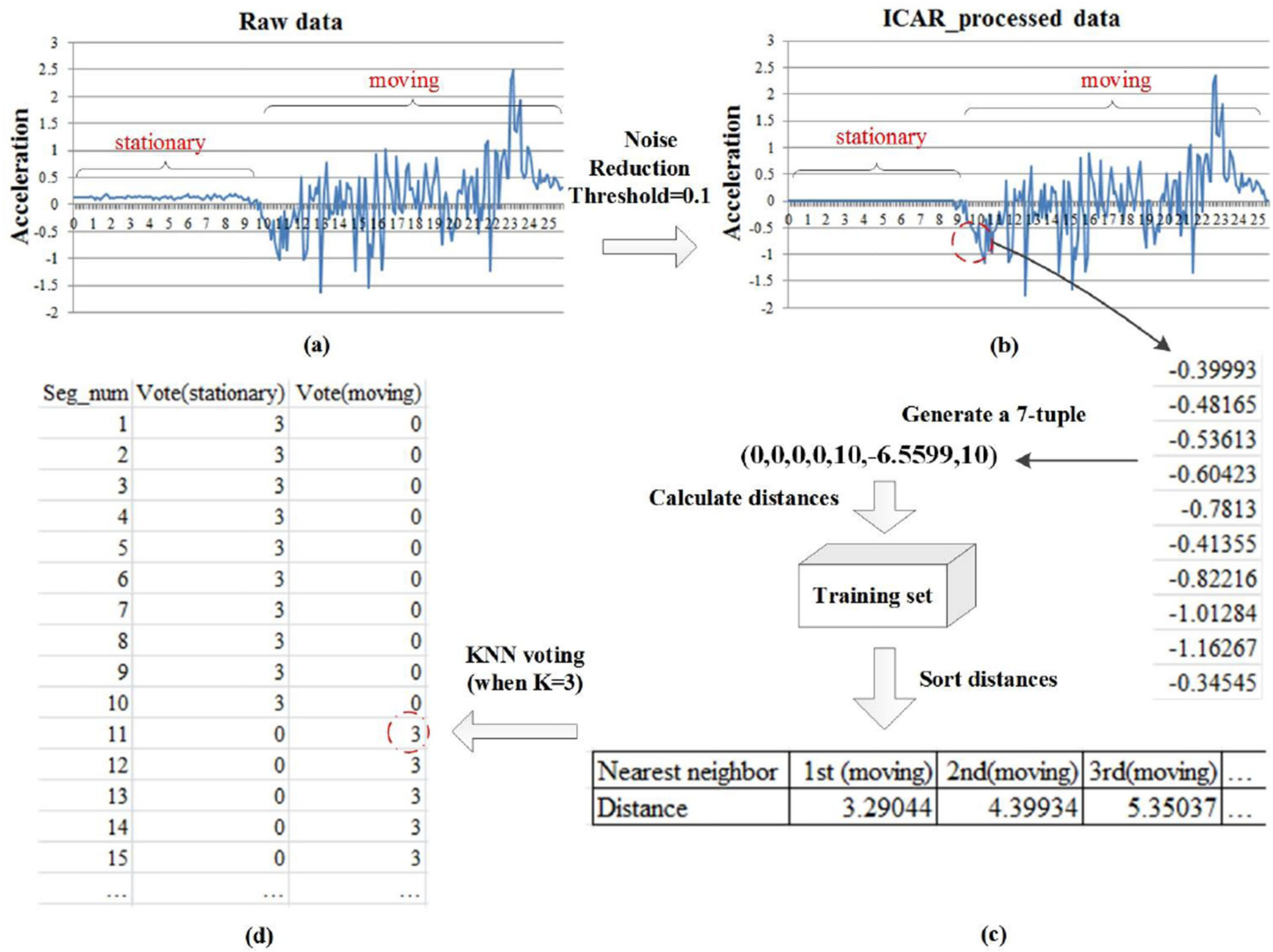


Figure 4.

A Running Example for Data Processing

Author Manuscript

Author Manuscript

Author Manuscript

Table 1

Analysis of Bouts for 30 Trials

	***K*= 1**	***K*= 3**	***K*= 5**	***K*= 7**
Actual No. of Bouts	300	300	300	300
Calculated No. of Bouts	301	300	300	300
Accuracy	99.67%	100%	100%	100%

Table 2

Analysis of Accumulated Maneuvering Time (Second)

	GT3X	*K* = 3	*K* = 5	*K* = 7
Average	115.767	115.325	115.302	114.915
Standard Deviation	-	0.452	0.465	0.487
Standard Error	-	0.083	0.085	0.089
*P*_value	-	0.868	0.862	0.751

Table 3

Analysis of Maximum Continued Maneuvering Time (Second)

	GT3X	***K* = 3**	***K* = 5**	***K* = 7**
Average	15.759	15.585	15.608	15.539
Standard Deviation	-	0.770	0.751	0.717
Standard Error	-	0.141	0.137	0.131
*P*_value	-	0.843	0.863	0.801

Table 4

Experimental Results in a Different Building

	Timer	**GT3X**	***K* = 3**	***K* = 5**	***K* = 7**
Average	12.55	12.31	12.23	12.21	12.12
Standard Deviation	--	0.31	0.45	0.46	0.49
Standard Error	--	0.06	0.08	0.08	0.09
*p*_value (Timer)	--	0.39	0.32	0.29	0.18
*P*_value (GT3X)	0.39	--	0.81	0.75	0.54

Table 5

Bout Classification Accuracy with KF

	K **= 1**	***K*** **= 3**	***K*** **= 5**	***K*** **= 7**
KF on Raw Data	64.30%	61.30%	59.70%	61%
KF without Base Noise	92.30%	78.70%	55.70%	46.30%