A SMARTPHONE-BASED GAIT DATA COLLECTION SYSTEM FOR THE PREDICTION OF FALLS IN ELDERLY ADULTS

Matthew Martinez[1,2], **Phillip L. De Leon**[1]

[1]**New Mexico State University**
Klipsch School of Elec. & Comp. Eng.
Las Cruces, NM 88003-8001

[2]**Sandia National Laboratories**
Albuquerque, NM 87185

ABSTRACT

Falls prevention efforts for older adults have become increasingly important and are now a significant research effort. As part of the prevention effort, analysis of gait has become increasingly important. Data is typically collected in a laboratory setting using 3-D motion capture, which can be time consuming, invasive and requires expensive and specialized equipment as well as trained operators. Inertial sensors, which are smaller and more cost effective, have been shown to be useful in falls research. Smartphones now contain Micro Electro-Mechanical (MEM) Inertial Measurement Units (IMUs), which make them a compelling platform for gait data acquisition. This paper reports the development of an iOS app for collecting accelerometer data and an offline machine learning system to classify a subject, based on this data, as faller or non-faller based on their history of falls. The system uses the accelerometer data captured on the smartphone, extracts discriminating features, and then classifies the subject based on the feature vector. Through simulation, our preliminary and limited study suggests this system has an accuracy as high as 85%. Such a system could be used to monitor an at-risk person's gait in order to predict an increased risk of falling.

Keywords: Falls risk, gait, accelerometer, machine learning, smartphone app.

1 INTRODUCTION

Falls prevention efforts for older adults have become increasingly important and are now a significant health research effort. Unintentional falls are the leading cause of injury to those over 65 years of age and have significant societal and financial impacts [1]. The development of assessment tools has focused on both external and age-related risk factors. External risk factors include home safety and medication risks, whereas age-related risk factors include muscle weakness, falls history, and gait deficits. Data is typically collected in a laboratory setting using motion capture systems, force platforms, and foot pressure sensors. These sensors provide a set of spatio-temporal gait features including walking speed and cadence, joint angles, ground reaction forces, and moments. The resulting data sets have high dimensionality since many features are 4-D (x-, y-, z- vs. t) and furthermore, complex interdependent relationships may exist among the features. Nevertheless, classifiers and predictors have been developed for falls research based on this data [2].

Unfortunately, data collection in the laboratory environment can be time consuming, invasive and requires specialized equipment and trained personnel, making laboratory measurement impractical for real-time gait monitoring and analysis. On the other hand, inertial sensors such as gyroscopes and accelerometers, which are smaller and more cost effective, have been shown to provide gait data useful in falls research [3,4]. Smartphones now contain Micro Electro-Mechanical (MEM) Inertial Measurement Units (IMUs) which make them a compelling platform for gait analysis. Using a smartphone, the IMU data can be collected, stored on the device, and analyzed in real-time or offline. As mobile processing capabilities continue to evolve, smartphones can be a cost effective, real-time gait analysis system for falls prevention.

A survey of signal processing techniques and features, extracted from raw accelerometer data can be found in [5]. In this article, the authors reviewed a variety of time- and frequency-domain features as well as symbolic features extracted from the data in order to classify among running, walking, and jumping states. Some of the features are pointed out as also having been used for detection and classification of a falls.

Another more recent article surveyed forty studies that used inertial sensors, i.e. accelerometers and gyroscopes to evaluate falls risk in elderly persons [2]. The survey focussed on sensor placement, features or variables extracted from the raw data, classifier used to assess the risk, and faller/non-faller classification accuracy. In the articles surveyed, the inertial sensors were most commonly placed on the lower back which approximates the center of mass [2]. For purposes of collecting gait data, placement of a smartphone on the lower back may not be practical and placement at the hip is perhaps more common. As discussed in the article, placement at the hip has greater potential as a long-term sensor location than does the lower back. Also in the articles surveyed, 130 variables were utilized in the assessment. Of these variables, those related to postural instability and gait consistency over stride had high falls risk classification accuracy. Of the variables related to postural instability those such as mediolateral and anteroposterior postural sway length and velocity were found to be good discriminators [2]. Of the variables related to gait consistency, gait speed and periodicity were also found to be good discriminators [2]. Finally, neural networks and naive Bayes classifiers were among those classifiers found to be most accurate.

This paper reports the development of an iOS app for collecting accelerometer data while a subject is walking and an offline machine learning system to classify a subject, based on this data, as faller or non-faller given their history (or lack thereof) of falls. The system uses the accelerometer data captured on the smartphone to extract discriminating features related to gait and then classifies based on the feature vector. This paper is organized as follows. In Section 2, we describe the iOS app we developed in order to collect accelerometer and gyroscope data and in Section 3, we describe how the data was collected in the field. In Section 4, we discuss how the data was post-processed which includes time-alignment of the right- and left-hip data streams. In Section 5, we describe the features, based on the harmonic spectrum as proposed by Liu, *et. al.* [6], which are extracted from the data. In Section 6, we review three classifiers that were used in this study: Support Vector Machine (SVM), naive Bayes, and K Nearest Neighbors (KNN) and describe how these classifiers were trained and evaluated. In Section 7, we provide the results of the classifiers including the confusion matrix, sensitivity, specificity, and accuracy. Finally, in Section 9 we conclude the article.

2 GAITLOGGER iOS APP

Smartphones contain a variety of sensors used to measure device motion, orientation, and various environmental conditions. These sensors include accelerometer, gyroscope, magnetometer, proximity sensor, and barometer [7,8]. With their programmability, ubiquity, local storage capability, and Internet connectivity, smartphones make a convenient and low-cost device for unobtrusively collecting health-related data and in particular motion data. Most recently Apple Inc., with the release of the open source Framework ResearchKit, has embraced the use of mobile devices as a medical research tool [9].

For this study, two Apple iPhone 6's were placed on the right and left hip to collect inertial gait data from subjects. The iPhone 6 makes inertial sensor measurements using the InvenSense MP67B 6-axis accelerometer and gyroscope [10]. The sensor consists of three independent accelerometers for the x-, y-, and z-axis and measures linear changes in velocity along each axis. The 3-axis gryoscope measures the rotation rate for each of the three axes. Access to sensor data is provided via Apple's iOS Software Development Kit (SDK) and the Core Motion Framework [11]. Core Motion gives the developer access to both the raw and processed inertial and magnetometer measurements. The raw sensor data is processed using the InvenSense's onboard Digital Motion Processor, which is capable of performing 6-axis (accelerometer and gyroscope) and 9-axis (accelerometer, gyroscope, and magnetometer) sensor fusion [12]. The processed data provides the iPhones' attitude, unbiased rotation rate, total acceleration, and the user-generated acceleration, i.e. the gravitational component is removed.

For this work, a custom iOS app was developed for logging inertial gait data from each subject. Data is collected via a push method which allows the app to sample and log sensor 6-axis data at a rate of 100 Hz. The data is post-processed through the Core Motion Framework and saved to files which are stored locally on the iPhones. Each sample is logged with a time-stamp provided through Core Motion based off of the iPhone clock. The frame of reference is selectable in Core Motion and for this study, the frame of reference is selected such that the z-axis is in the vertical direction.

3 DATA COLLECTION

Inertial gait data was collected on May 11 and May 12, 2015 by Electronic Caregiver Company, also known as Sameday Security, Inc., in Las Cruces, NM. In total, 25 older adults participated in the data collection event. Prior to data collection, each subject was given a comprehensive falls risk screening [1], where the subjects were asked to self-report as having experienced a fall within the past year. There were a total of 11 subjects which self-identified as a faller and 14 subjects which self-identified as a non-faller.

(a) (b)

Figure 1: (a) A subject standing on the walkway wearing the custom vest with IR reflectors attached to the back, legs, and feet is also shown fro 3-D motion caption. The right iPhone 6 can be seen attached to the gait belt. (b) Another subject where the left iPhone 6 is shown attached to the gait belt.

Data collection was performed using four different sensors, a pressure sensitive walkway, a system of 3-D motion capture cameras aligned along the perimeter of the walkway, Google glass, and two Apple iPhone 6's. Fig. 1(a) shows a picture of the data collection environment. During the data collection process the participant walks across the walkway, where planter force and pressure are measured, full-body movement is measured using infrared markers [see Figs. 1(a) and (b)] and the 3-D motion capture cameras. In addition to pressure and motion capture data, inertial sensor measurements are made using Google Glass and the Apple iPhones. Each iPhone is attached to the subject's left and right hip using a gait belt and a holster clip [see Fig. 1(b)]. The sensors on the walkway, 3-D motion capture system, and Google glass are synchronized to start data collection when the subject begins moving; the iPhones are not synchronized with the system. The 3-D motion capture and walkway data are to be used as ground-truth validation for the inertial sensor data in a possible future study. Logging of the gait data begins 10 s after start buttons on each app are touched and is roughly synchronized to the other data being captured. Gait data is logged for 30 s which is sufficient time for the subject to walk down the walkway, turn around, and walk back. Not every subject performed two passes across the walkway. Of the 25 subjects only 15 had two passes. Example plots of x-, y-, and z-axis accelerometer data are given in Fig. 2, where we also observe the periodic nature of the gait signal in all three axes. The first characteristic heel strike signature can be observed at approximately 0.6 s on the z-axis line, and the heel strike for the opposite foot occurs at approximately 1.3 s. Successive heel strikes appear in approximately 1 s intervals.

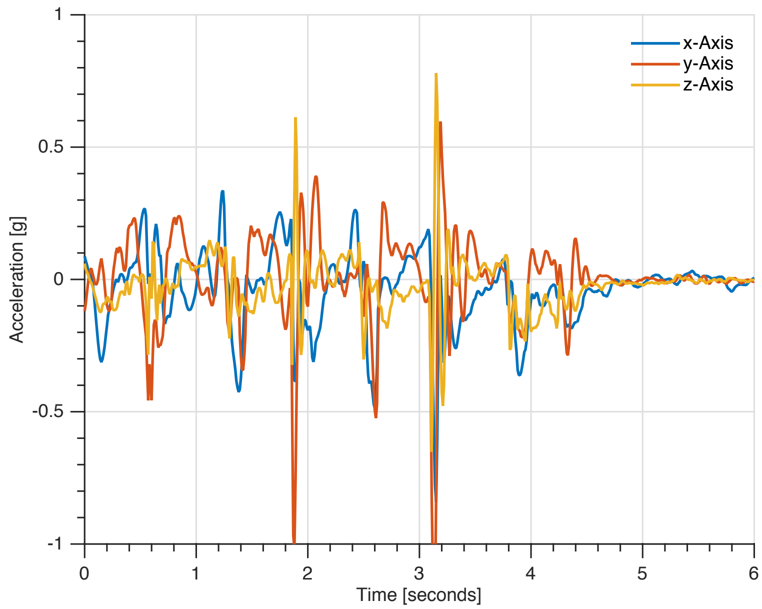

Figure 2: Plot of the raw acceleration signals from the iPhone NM GaitLogger app for all three axes. The periodic nature of the gait signals can be observed in all three axis, where the gait cycle has a period of approximately 1 s.

4 DATA POST-PROCESSING

The data from each iPhone is collected independently using separate instances of the NM Gait Logger app. The two iPhones are sampling data asynchronously at a rate of 100 Hz. Each sample is time-stamped using the iPhone's internal clock as the local time reference. Each clock has a precision of 10 μs and is synchronized to a global time reference via Apple's network time protocol (NTP) time server network. The worst case offset between the global time reference and the iPhone's local clock is 50 ms. However, the local clock offset is usually within 10 ms of the global time reference. In a worst case scenario, the samples can be at most ten sample periods apart. This difference is acceptable given the expected stride duration is within 1 to 2 s.

Alignment of the two data streams was performed in two steps. First, the two data sets were time aligned to remove the offset in start times, which is a result of starting the logging process manually. The alignment was performed by truncating the initial samples from the file with the earliest time-stamp such that the offset between the initial time-stamp from each data set was minimized. In general, the offset between the two time-stamps was within 1 ms and the two data sets were aligned to account for the asynchronous clocks by resampling the data onto a global clock. The time support for the left iPhone was arbitrarily selected as the global clock and the data for the right iPhone was resampled on the global clock using MATLAB's `interp1` function to perform a B-spline interpolation. The input data points are the acceleration sample points and corresponding time-stamps for the right iPhone and the query points are the time-stamps for the left iPhone.

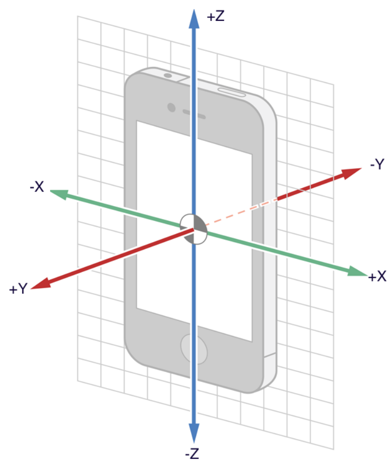

Figure 3: Local coordinate system for iPhone 6 inertial sensors [13]

Each iPhone performs measurements within its own local coordinate system (see Fig. 3) and shares the same z-axis, which represents the vertical direction. Positive z is the upward direction. The x- and y-axes for the right iPhone are a 180° rotation of the left iPhone. For the data from the second pass, the x- and y-axes for the iPhone on the left hip are a 180° rotation of the left iPhone from the first pass. To maintain a common reference plane, data is post-processed such that each iPhone is referenced to the coordinate system of the first pass for the left iPhone, which is arbitrarily selected as the global coordinate system. For the first pass right iPhone and the second pass left iPhone, the coordinate transform is achieved by by changing the numerical sign on the x and y data.

Each signal was segmented into two passes using the turn as the demarcation point in order to increase the number of faller and non-faller examples for training. The end of the first pass occurred before the turn when the subject stopped walking and the start of s pass occurred after the turn before the subject started walking again. There were 15 subjects which provided data with two passes, resulting in 40 total examples of the faller and non-faller classes. Of these 40 examples, there were 24 examples of the non-faller class, $\mathcal{C}_0$ and 16 examples of the faller class, $\mathcal{C}_1$.

5 FEATURE EXTRACTION

In [2], the authors examined discriminating features proposed in prior research in classification of fallers and non-fallers from inertial sensor measurements. The features found to be the most discriminating were those computed from the harmonic spectrum of the acceleration data [6]. Features 11–13 and 19–33, from Table I in [6] were implemented in this work. In addition, the fundamental frequency, f_1 for each axis is also used as a feature. Thus 21 total features were extracted for each iPhone and are concatenated to form a 42-D feature vector.

$$\mathbf{x} \quad = \quad [f_1^x, f_1^y, f_1^z, F_{11}, F_{12}, F_{13}, F_{19}, \ldots, F_{33}]^T \tag{1}$$

where F_k denotes the kth feature in [6].

The fundamental frequency for each axis is computed as the inverse of the time between the first two largest peaks in the auto-correlation function. A Hamming window is then applied to the acceleration data for each axis and the harmonic spectrum is computed using the fast Fourier transform (FFT). The harmonic spectrum is computed such that the frequency bins are integer multiples of the fundamental frequency.

The first subset of features is based on the ratio of a harmonic (fundamental, second, third, and fourth) to the sum of the first six harmonics. As an example, feature number 20 in [6] is the second harmonic ratio for the x-axis acceleration and is given by

$$F_{20} = \frac{|X(f_2)|}{\sum_{k=1}^{6} |X(f_k)|} \tag{2}$$

where $|X(f_k)|$ is the kth harmonic magnitude for the x-axis acceleration. In this first subset, the ratios of the first four harmonics for all three axes are computed which yields twelve features.

The second subset of features is based on the ratio of the sum of the first six harmonics to the sum of the remaining harmonics. As an example, feature number 12 in [6] is the harmonic ratio for the y-axis acceleration and given by

$$F_{12} = \frac{\sum_{k=1}^{6} |Y(f_k)|}{\sum_{k=7}^{20} |Y(f_k)|} \tag{3}$$

where $|Y(f_k)|$ is the kth harmonic magnitude for y-axis data. This ratio is computed for the three axes and yields three features.

The third subset of features is based on the ratio of the sum of the even harmonics to the sum of the odd harmonics. As an example, feature number 33 in [6] is the even-to-odd harmonic ratio for the z-axis acceleration and given by

$$F_{33} = \frac{\sum_{k=1}^{10} |Z(f_{2k})|}{\sum_{k=1}^{9} |Z(f_{2k+1})|} \tag{4}$$

where $|Z(f_k)|$ is the kth harmonic magnitude for z-axis data. This ratio is computed for the three axes and yields three features. The ratio of even to odd harmonic magnitudes of acceleration data has been found to be a discriminating feature for faller and non-faller classes in [2, 14]. This ratio reflects the proportion of acceleration that is in phase with the subject's stride frequency, with even harmonics correlating with in-phase components and odd harmonics correlating with out-of-phase components [2].

6 CLASSIFIERS

6.1 Support Vector Machine

For the SVM classifier, a separating hyperplane or set of hyperplanes are constructed in a high dimensional space from the training data using a kernel function. The best hyperplane is that which maximizes the distance between itself and the closest data points [15]. For this analysis a linear kernel was selected and a box constraint of 0.1 was used.

6.2 Naive Bayes

The distributional parameters for multivariate, normal distributions were estimated via maximum likelihood (ML) estimation for the faller and non-faller classes from the training data [15]. For the given test feature vector, $\mathbf{x}$ the likelihood of each class is computed and a ML decision is made. Thus, assuming equal priors, the decision is given by [15]

$$y = \arg\max_{k=0,1} p(\mathbf{x}|C_k) \tag{5}$$

where the distribution for class C_k is given by

$$p(\mathbf{x}|C_k) = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{6}$$

and $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are the mean vector and covariance matrix for class C_k.

6.3 K Nearest Neighbors

For the K nearest neighbors classifier, the Euclidean distances from the test feature vector to the training feature vectors are computed. The class decision is made by determining which class is most frequent among the K nearest training points [15]. For this work, $K = 2$.

6.4 Feature Selection and Classifier Training

The most discriminating feature set was found by removing the same feature from each iPhone's 21-D feature vector. The classifier was than trained with the resulting 40-D feature vector and the feature set with highest accuracy was selected as the most discriminating.

An exhaustive leave-one-out cross-validation was performed using the feature vectors extracted from the 32 data files. In this validation, we train each classifier using 31 feature vectors from the two classes and then classify the remaining feature vector and determine whether the decision is correct or incorrect based on the class label [15]. This process is repeated through the data set and the results are averaged in order to arrive at an accuracy measure for the classifier.

7 RESULTS

Classifier results for the prediction of faller vs. non-faller based on a 40-D feature vector extracted from accelerometer data captured on the iPhones are shown in Tables 1-2. Table 1 gives the confusion matrix for the SVM classifier and Table 2 provides the sensitivities, specificities, and accuracies for each of the classifiers. The SVM had the highest accuracy at 85%, followed by the KNN classifier at 75%. The Naive Bayes classifier performed the worst with an accuracy of 67.5%. The most discriminating feature set for the SVM had feature number 11 in [6] removed, where as the most discriminating feature set for the KNN classifier had feature number 12 in [6] removed. The Naive Bayes classifier had the highest accuracy when feature number 28 in [6] was removed from the feature set.

Table 1: Confusion matrix for the support vector machine classifier.

		Predicted		
		Faller	Non-Faller	Total
Actual	Faller	12	4	16
	Non-Faller	3	21	24
	Total	15	25	40

Table 2: Sensitivity, specificity, and accuracy of the support vector machine, naive Bayes, and K-nearest neighbor classifier

	Sensitivity	Specificity	Accuracy
SVM	**81.3%**	**87.5%**	**85%**
NB	62.5%	70.8%	67.5%
KNN	62.5%	83.3%	75%

8 ACKOWLEDGEMENT

Electronic Caregiver$^{\text{TM}}$, Sameday Security, Inc., provided hardware and data for this study.

9 CONCLUSION

In this paper we have reported on the development of a system to classify a person as a faller or non-faller, based on their gait pattern. This system includes a custom app which logs acceleration and gyroscope data from an iPhone placed on the subject's hip. Offline, we extract features based on the harmonics of the x-, y-, and z-axis acceleration data as well as the signal vector magnitude and use these to classify the subject as a faller or non-faller. We evaluated three classification methods using support vector machine, naive Bayes, and K nearest neighbors. The SVM classifier had the best performance of the three classifiers with sensitivity, specificity, and accuracy of 81.3%, 87.5% and 85%, respectively. Such a system could be used to monitor an at-risk person's gait in order to predict an increased risk of falling.

References

[1] J. M. Fabre, R. Ellis, M. Kosma, and R. H. Wood, "Falls risk factors and a compendium of falls risk screening instruments," *J. Geriatric Physical Therapy*, vol. 33, no. 4, pp. 184–197, 2010.

[2] J. Howcroft, J. Kofman, and E. D. Lemaire, "Review of fall risk assessment in geriatric populations using inertial sensors," *J. Neuroengineering Rehabil.*, vol. 10, no. 91, 2013.

[3] D. Giansanti, "Investigation of fall-risk using a wearable device with accelerometers and rate gyroscopes," *Physio. Meas.*, vol. 27, pp. 1081–1090, 2006.

[4] A. H. Khandoker, D. T. H. Lai, R. K.Begg, and M. Palaniswami, "Wavelet-based feature extraction for support vector machines for screening balance impairments in the elderly," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 4, pp. 587–597, Dec. 2007.

[5] D. Figo, P. C. Diniz, D. R. Ferreira, and J. M. P. Cardoso, "Preprocessing techniques for context recognition from accelerometer data," *Pers Ubiquit Comput*, vol. 14, pp. 645–662, 2010.

[6] Y. Liu, S. J. Redmond, N. Wang, F. Blumenkron, M. R. Narayanan, and N. H. Lovell, "Spectral analysis of accelerometry signals from a directed-routine for falls-risk estimation," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 8, pp. 2308–2315, Aug. 2011.

[7] Apple Inc. (2014) iPhone 6 Technology. [Online]. Available: https://www.apple.com/iphone-6/technology/

[8] Google Inc. (2015) Sensor Overview. [Online]. Available: http://developer.android.com/guide/topics/sensors/

[9] Apple Inc. (2015) ResearchKit Technical Overview Version 1.0. [Online]. Available: https://developer.apple.com/researchkit/researchkit-technical-overview.pdf

[10] iFixit. (2014) iPhone 6 Teardown. [Online]. Available: https://www.ifixit.com/Teardown/iPhone+6+Teardown/29213

[11] Apple Inc. (2015) CoreMotion Framework Reference. [Online]. Available: https://developer.apple.com/library/ios/documentation/CoreMotion/Reference/CoreMotion_Reference/

[12] "Mpu-6000 and mpu-6050 product specification," InvenSense, San Jose, California.

[13] Apple Inc. (2015) Event Handling Guide for iOS. [Online]. Available: https://developer.apple.com/library/prerelease/ios/documentation/EventHandling/

[14] H. J. Yack and R. C.Berger, "Dynamic stability in the elderly: identifying a possible measure," *J. Gerontology: Med. Sci.*, vol. 48, p. M225?M230, 1993.

[15] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.