Using machine learning to detect pedestrian locomotion from sensor-based data

Courtney Anne Ngo
De La Salle University
Manila, Philippines
courtney_ngo@dlsu.ph

Solomon See
De La Salle University
Manila, Philippines
solomon.see@gmail.com

ABSTRACT

The integration of low cost microelectromagnetic (MEM) sensors into smart phones have made inertial navigation systems (INS) possible for ubiquitous use. Many research studies developed algorithms to detect a user's steps, and to calculate a user's stride to know the position displacement of the user. Subsequent research have already integrated the phone's heading to map out the user's movement across a physical area. These research, however, have not taken into account negative pedestrian locomotion, wherein the user is moving but is not exhibiting any position displacement.

Current INSs are not suited to handle negative pedestrian locomotion movements, and this leads them to consider false steps as real steps. As the INS's modules depend heavily on the outputs of the other modules, a cascading error would most likely occur.

This research aims to solve this problem by collecting positive and negative pedestrian locomotion with data from phone-embedded sensors positioned in the research subject's front pocket. Using these data, a model will be built to classify negative pedestrian locomotion from positive ones, and to eventually improve the INS's accuracy overall.

Categories and Subject Descriptors

H.4 [**Location-based Services**]: [Information Systems, Information Systems Applications, Spatial-temporal Systems]

General Terms

Machine Learning, Inertial Navigation Systems, Sensors

1. INTRODUCTION

Indoor navigation systems determine where a device has traversed inside a building. These navigation systems can be employed in applications to help users find a specific location in closed places like conference centers and office buildings. Unlike outdoor navigation systems like the Global Positioning System (GPS), indoor navigation systems can not use satellite signals as heavy attenuation takes place when the signals make their way through physical obstacles.

To solve this, researchers have experimented with Wi-fi signals [2, 1, 17, 18, 3], vision [6], ultra-wide bands [14], cellular-based signals [11], magnetometers [5], and combinations of these [4]. All of these research are dependent on environment variables such as Wi-fi routers and markers, and some require data collection prior to system use. This would mean that a significant change in the environment or the variables would affect the performance of these navigation systems.

INSs, on the other hand, uses data from inertial sensors such as gyroscopes and accelerometers to determine the path a device has travelled. Smart phones currently already have these sensors as micro-electrical-mechanical systems (MEMS) devices, making it possible for INSs to be applied in smart devices and possibly for ubiquitous use. Compared to other navigational systems, INSs are independent of its environment, requiring less cost that otherwise would have incurred with the need of access points. This also implies less environment set-up as access points do not need to be installed for the navigation system to operate. Considering that it is a cheaper and simpler alternative, INS appears to be a more attractive approach to building navigation systems.

Challenges

Using INSs in real-world situations, however, is limited because its MEMS devices are susceptible to noise and gradual drifts that cause cascading errors. Because of this, most existing INSs integrate regular checking with access points with known positions such as satellites and Wi-fi routers to calculate the position of the mobile unit to compensate for these inaccuracies [9].

Another problem, which this study intends to address, is correctly classifying irregular movements. In this research, positive pedestrian locomotion is defined as movements that include moving from one physical position to another on foot. Examples of these are walking, jogging, running, and climbing up and down the stairs. False pedestrian locomotions are movements that do not require moving from a position, such as standing. There are, however, some false pedestrian locomotion movements that can simulate movement from position, and these presents a problem to some existing INSs. These movements include walking-in-place, jogging-in-place, and running-in-place. It is important future INSs can correctly disregard false pedestrian locomotion movements to avoid cascading errors as the modules depend on each other as displayed in Figure 1. Similarly, it cannot be expected that users would not exhibit any form of negative pedestrian locomotion movements in real-world applications. An INS that considers in these negative movements will better suit mobile applications that plan to map user paths in an area.

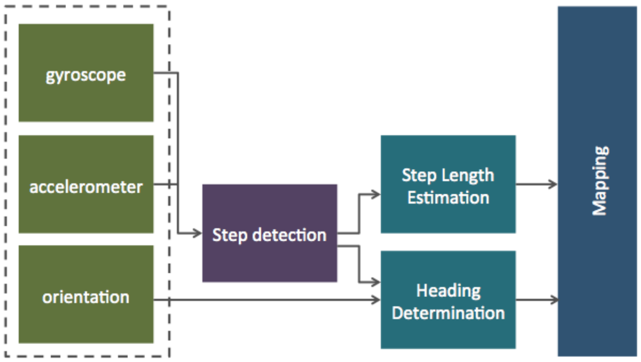

Figure 1: Conventional system flow of inertial navigation systems.

The main objective of this research is to solve this problem by creating an additional module in an INS whose role is to classify whether a user is making a positive or negative pedestrian locomotion movement. In the proposed solution, false pedestrian locomotion movements will be properly detected, thus false steps would be avoided. This will consequently affect the estimated path length of the user and is hypothesized to improve the outputs of the INS.

Section 2 discusses previous research that have tackled the problem before, and Section 3 explains this research's solution to the problem. The resulting performance of the model and its effect on the the INS in general are shown and analysed in Section 4, and the conclusion and recommendations of the research are written in Section 5.

2. REVIEW OF RELATED LITERATURE

2.1 Pedestrian Locomotion Heuristics

Currently, there are no studies that have a separate module to classify positive from negative pedestrian locomotion, but there are some which have integrated similar measures in their step detection algorithms. In some studies [7, 8], additional heuristics were implemented to prevent allowing false positive steps. These heuristics are hard-coded based on each study's preliminary data. As it is, more heuristics will need to be added to allow more movements.

Although the following research did not take into consideration a wider range of movements compared to this study, their heuristics were able to prevent certain negative pedestrian locomotion movements as positive.

2.1.1 Lag Parameter

In a study conducted by Lee and Mase [7], a lag parameter was added in their step detection algorithm. With the lag, the system can supposedly check if the step taken is not a step but another body movement. It involves getting the z-axis of the accelerometer which is indicative of upward movements of the leg [7]. The lag parameter is as follows:

$$lag = min_{j=0...N}(\sum_{n=0}^{N} z(n)z(n-j)) \qquad (1)$$

where
lag is the lag parameter
N is the window length
$z(n)$ is the z-axis value of the accelerometer at time i

The lag must be greater than a threshold to pass the heuristic. As can be seen in the equation, the study assumed that other body movements would have less activity in the accelerometer's z-axis, and that walking would induce peaks in the z-axis. However, walking-in-place would also express a high activity in the z-axis even though it is truly a false pedestrian locomotion movement.

2.1.2 Dynamic Time Warping

Another study [8] further used dynamic time warping (DTW) as an added filter to detect false steps. Aside from (1) checking if peaks and valleys pass a certain threshold, (2) peaks and valleys must also not be too short, or (3) too long (maximum of 1 second). Acceleration's peak and valley's magnitudes are also considered, where (4) the magnitude must be within a minimum of 0.2g, and a maximum of 2.0g.

With DTW, two more heuristics were formed as shown in Figure 2. A fifth heuristic uses DTW to calculate the similarity of steps taken with the right leg, and similarity of steps taken with the left leg. In this condition, the similarity of the last step taken with the left/right foot and the current step taken with the left/right foot must be greater than a threshold. If the result is negative, a sixth heuristic compares the current left step with the next left step. If these two signal's similarity passes the threshold, the current left step would be considered a step. With this method, their step detection algorithm can tell the difference between a step taken while walking and a step taken while walking-in-place given that the two steps are taken after the other and the false step is just a momentary gap from a series of true pedestrian locomotion movements. However, their system can still possibly fail if the user continues to perform a false pedestrian locomotion movement.

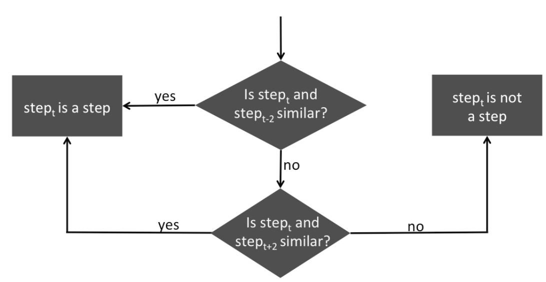

Figure 2: A fifth and sixth heuristic uses DTW to measure the similarity of to signals.

After adding the DTW heuristic, the research recorded a drop in false positives (incorrectly processed false steps) from 29 to 14.

In the study, false negatives are more important than false positives. False positives can be further checked with the step detection algorithm. Even if a false step was considered a step in the pedestrian locomotion model, there is still the possibility that the false step would be detected as false by the step detection algorithm. The false negatives increased from 0.4 to 0.5. But as stated in the study, the benefits outweighed the disadvantages.

3. PROPOSED SOLUTION

This research proposes to create a separate module in the standard INS framework that will focus on classifying a movement as either false or true pedestrian locomotion movement. As shown in Figure 3, the new module would operate first before the step detection module. If the module identifies a window of movement as false pedestrian locomotion, the succeeding modules would not process that window. If it does detect the window as true, the succeeding modules would operate normally. This would imply that the INS can perform more efficiently should the new module classify well. On the other hand, a cascading error can transpire instead.

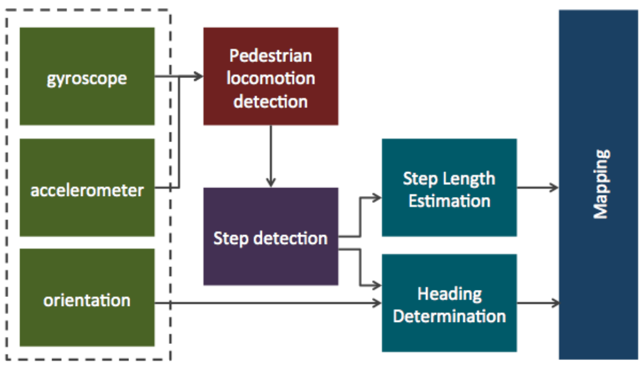

Figure 3: Conventional system flow of inertial navigation systems with the model this research aims to create.

3.1 Inertial Navigation System

A simple INS would be created to compare the performance of a conventional INS against an INS with the pedestrian locomotion detection module. The modules are discussed below along with the algorithms and heuristics used in each.

3.1.1 Step Detection Module

The step detection module would detect steps from accelerometer signals once the pedestrian locomotion model determines that the user is performing a positive pedestrian locomotion movement.

The accelerometer signals would be scoured for a value greater than threshold α. In order to discard false peaks, a second threshold β is introduced. Threshold β is the minimum time gap between two steps. Before a step is identified, the time gap between the said step and the previous step must be greater than threshold β.

Both thresholds are determined after collecting user data.

3.1.2 Stride Length Estimation Module

The stride length estimation module would start calculating for the step length once the Step Detection Module has determined the user made a step. A linear model would be created as studies have shown before that a linear relationship exists between stride length and step frequency [8]. This module would update the step frequency along with the Step Detection Module. A linear model would be generated after collecting data.

3.1.3 Heading Determination Module

This module would work side-by-side with the Stride Length Estimation Module after the Step Detection Module determines a step has been taken. It is responsible of approximating the direction the user is heading. In this research, the orientation y-axis data would be used to determine the heading. The values can range between 0° to 359°.

3.1.4 Mapping Module

The mapping module outputs a series of points indicating a user's traversal across a space. It would receive inputs from the stride length estimator module and heading determination module, and would have knowledge of the coordinates of the previous point. The coordinates of the initial point would be set to (0,0).

The new point would be calculated as:

$$x_{cur} = l * cos(a) + x_{prev} \tag{2}$$

$$y_{cur} = l * sin(a) + y_{prev} \tag{3}$$

Where x_{cur} is the x-coordinate of the current point, y_{cur} is the y-coordinate of the current point, x_{prev} is the x-coordinate of the previous point, y_{prev} is the y-coordinate of the previous point, l is the stride length, a is the heading.

The final predicted path would be relative to the user's initial position, and would be superimposed manually over the actual route of the user. A map would not be used in this module.

3.2 Pedestrian Locomotion Model

As the main component of the pedestrian locomotion module, the pedestrian locomotion model is a classifier that identify movements as either positive or negative pedestrian locomotion movements. A discussion of how the model was created is written below.

3.2.1 Data Collection

In this research, 20 subjects will participate by performing 12 movements for data collection. Each subject should be at the age range of 19 to 49 years old, as a stable gait has been found across that age range [15]. On a similar note, the subjects should also be able-bodied. Every subject will perform each of the 12 movements for 5 minutes each. The 12 movements are composed of 3 true pedestrian locomotion movements : (1) walking, (2) climbing up stairs, and (3) climbing down stairs; and 9 false pedestrian locomotion movements : (4) standing, (5) walking-in-place, (6) rocking on heel and and ball of the feet, (7) bending, (8) twisting, (9) turning, (10) swinging the legs, (11) sitting, and (12) random. The last movement, random, is to be used to test the robustness of the model in terms of classifying unlisted movements in future research.

A Samsung Galaxy S2 phone was used to collect data. For this purpose, a mobile application was created to collect accelerometer and gyroscope data at a rate of 100Hz. The phone was placed in the subjects's right-side pockets at the front. Placing the phone in the mid-section of the subject is strategic as it is the person's center of gravity, making it sensitive to movements made with the limbs. The position is also a typical location phones are placed in. The phone is limited to a specific orientation that faces the phone screen towards the thigh of the subject, and the top of the phone is pointed down.

3.2.2 Data Pre-processing

Windowing. The data entries would be grouped into windows of size 100. This window size is equivalent to a second worth of records, and will have an overlap of 50%.

Feature Modeling. After applying a low-pass filter to smoothen the raw data, three features were extracted from each of the sensors's axes: mean, standard deviation, and energy.

Mean The mean of a set of numbers is normally used to estimate where a sample point is "located" when the set is arranged in an ascending order, thus making it a measure of location[12]. Although calculating for the mean has its advantages, it is easily influenced by outliers [12].

$$Ave_{i,w} = \frac{1}{N}\sum_{i=0}^{N-1} sens_{i,w} \tag{4}$$

Where N is the window size, $Ave_{i,w}$ is the mean of the w-axis of the accelerometer/gyroscope over window i, $sens_{i,w}$ is the w-axis value of the accelerometer/gyroscope at time, and w can either be the x, y, or z-axis.

Using preliminary data collected in this research, the means of the unfiltered Euclidean norm of the accelerometer data were collected as shown in Figure 4. Each mean was gathered over a window of 200 samples, with a 50% overlap. The results show that there is a distinction between the means collected when a subject is walking as opposed to when a subject is standing and walking-in-place. Based on the figure, the signals tend to reach higher frequencies when performing positive pedestrian locomotion movements. As compared to the subsequent features, the mean is less distinguishing of positive and negative pedestrian locomotion movements.

Standard Deviation Getting the variance of a data set aids in understanding the position of a sample point relative to the set's mean[16]. The standard deviation is equivalent to the square root of the variance of a given data set [12]. It is usually used in advantage of its sensitivity to the variance of the upper and lower data samples [16].

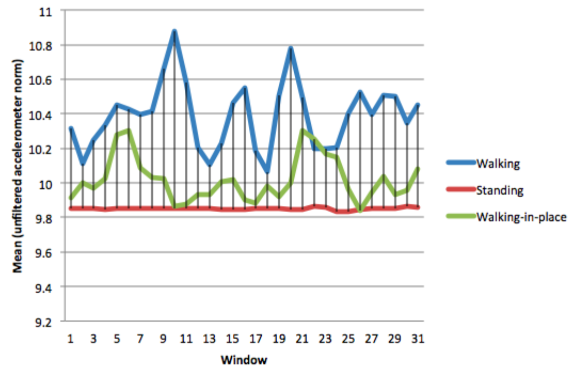

Figure 4: Graph showing that means collected while walking are distinct from the means collected when standing or walking-in-place.

$$Stddev_{i,w} = \sqrt{\frac{1}{N-1}\sum_{i=0}^{N-1}(sens_{i,w} - Ave_{i,w})^2} \tag{5}$$

Where N is the window size, $Stddev_{i,w}$ is the standard deviation of the w-axis over window i, $sens_{i,w}$ is the w-axis value of the accelerometer/gyroscope at time i, w can either be the x, y, or z-axis, and $Ave_{i,w}$ is the mean of the w-axis of the accelerometer/gyroscope over window i.

The standard deviations collected from the preliminary data shows a wide distinction between positive and negative pedestrian locomotion. Using the same data in gathering the means, the standard deviations of the unfiltered Euclidean norm of the accelerometer data were computed. As seen in Figure 5, the signals tend to fluctuate more when walking, as opposed to standing. The difference in walking and walking-in-place is explained by a higher y-axis activity when doing the former.

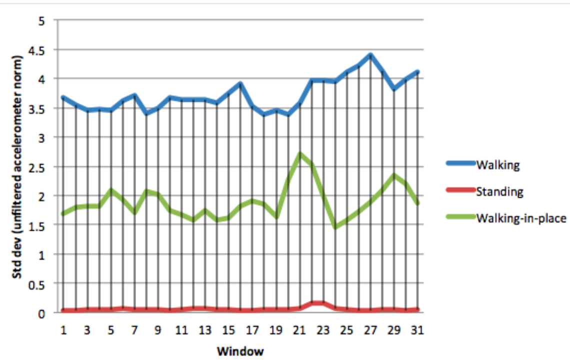

Figure 5: Graph showing that the standard deviations collected while walking are distinct from the means collected when standing or walking-in-place.

Energy The energy of a signal is usually interpreted as the strength of the signal. Given this, a signal's energy can be measured by calculating the area under the curve [10]. To treat the negative samples that accompany digital signals, the square of the signal is computer over a temporal window.

$$Energy_{i,w} = \sum_{i=0}^{N-1} sens_{i,w}^2 \quad (6)$$

Where N is the window size, $Energy_{i,w}$ is the standard deviation of the w-axis over window i, $sens_{i,w}$ is the w-axis value of the accelerometer/gyroscope at time i, and w can either be the x, y, or z-axis.

The formula used to determine the signal energy is derived from the formula used to identify energy in Physics.[13]

A preliminary test was also done to gauge the efficacy of using energy as a feature. Based on the same data set, the energies of the unfiltered Euclidean norm of the accelerometer data were calculating using the same window size. A distinction between walking and standing or walking-in-place can be clearly seen in Figure 6, similar to the results displayed by the previous two features. It is evident in this experiment that positive pedestrian locomotion movements contain more energy as opposed to its negative counterpart.

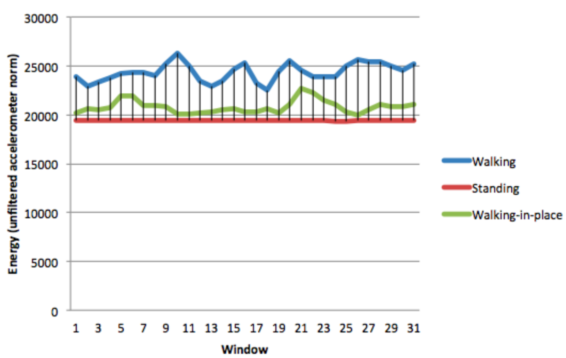

Figure 6: Graph showing that energies collected while walking are distinct from the means collected when standing or walking-in-place.

Model Generation. A C4.5 model and a support vector machines (SVM) model would be generated using WEKA's J48 and sequential minimal optimisation (SMO) algorithms. The model would be used in the pedestrian locomotion detection module, and would determine if the person is performing a positive or negative pedestrian locomotion movement.

4. RESULTS AND DISCUSSION

4.1 Model Evaluation

Using WEKA's 10-fold cross validation, initial evaluations of the models for the new module were made.

With the C4.5 algorithm, a 10-fold cross validation revealed the model performs well with an accuracy of 95.6%, and with an equally high recall and precision. The generated tree showed that the standard deviation of the data from the accelerometer's y-axis at the root node of the tree.

On the other hand, the linear kernel SVM model achieved a 93.3% accuracy, and a similar percentage for both recall and precision.

4.2 INS Evaluation

Two kinds of tests were conducted for the INS evaluation: a straight route, and a multi-movement route. Three subjects participated in the experiments for both tests. The performances of a conventional INS and an INS with the pedestrian locomotion detection module is shown below.

4.2.1 Straight route test

The straight route is composed of a 5-meter walk. Since this test is composed purely of positive pedestrian locomotion movements, the INS with the prediction module should correctly identify all movements as positive. If not, the conventional INS will be at a clear advantage.

The actual total number of steps and total path length carried out by the subjects are shown in Table 1. The same table also shows the estimated number of steps and path length the conventional INS computed. Because the movements are purely positive, it is correct to consider all detected steps. The simple step detection algorithm used in this research can explain the incorrect detected number of steps for subjects A and B. Similarly, a simple stride length estimation algorithm was also used for this study, which is the reason why the estimated path length is not too accurate.

Subject	Actual		Conventional INS	
	Total steps	Total length	Total steps	Total length
A	7	5m	7	4.633m
B	8	5m	7	4.622m
C	7.5	5m	6	3.956m

Table 1: Actual total number of steps and total path length compared to those calculated by a conventional INS from users performing the straight route test.

The results of the INSs with a C4.5 and an SVM model is shown in Table 2 and Table 3. Using the conventional INS as a benchmark, it can be said that both INSs performed well in the straight route test. Both models were able to properly predict all movements to be true.

Subject	Total steps	Total length	Accuracy	Recall
A	7	4.633	1	1
B	7	4.622	1	1
C	6	3.956	1	1

Table 2: Performance statistics of an INS with a C4.5 model on the straight route test.

4.2.2 Multi-movement route

Subject	Total steps	Total length	Accuracy	Recall
A	7	4.633	1	1
B	7	4.622	1	1
C	6	3.956	1	1

Table 3: Performance statistics of an INS with a SVM model on the straight route test.

The multi-movement routine takes 35 seconds: 5 seconds of walking, 5 seconds of walking-in-place, 5 seconds of walking, 5 seconds of bending, 5 seconds of walking, 5 seconds of standing, and 5 seconds of walking. Mixing positive and negative pedestrian locomotion movements in one routine can really test the accuracy, recall, and sensitivity of both the C4.5 and SVM models. The effect of the pedestrian locomotion detection module on the step detection and stride length estimation modules can also be observed in this experiment.

The results of the conventional INS can be seen in Table 4. As expected, the estimated number of steps and path length are grossly remote from the actual numbers. This is on account of the false steps that were detected in the three 5-second blocks of negative pedestrian locomotion movement. Because the conventional INS would just consider all detected steps in the step detection module, the error cascaded to the stride length estimation module as well. In this test, the conventional INS had an accuracy of 57.14%, a recall of 100%, and a specificity of 0%.

Subject	Actual		Conventional INS	
	Total steps	Total length	Total steps	Total length
A	28	20m	41	27.086m
B	32	20m	42	27.724m
C	32	20m	47	30.865m

Table 4: Actual total number of steps and total path length compared to those calculated by a conventional INS from users performing the multi-movement route test.

The INS with the C4.5 model performed with a better accuracy than the conventional INS, as shown in Table 5. And, although the recall declined lower than that of the benchmark, the over-all effect on the estimated number of steps and path length is considerably more advantageous.

Subject	Total steps	Total length	Accuracy	Recall	Sensitivity
A	27	17.715	0.7429	0.7	0.8
B	38	25.011	0.7568	0.9	0.67
C	32	20.868	0.8857	0.85	0.93

Table 5: Performance statistics of an INS with a C4.5 model on the multi-movement route test.

Not only did the INS with the SVM model surpassed the conventional INS, it also outperformed the INS with the C4.5 model. This resulted in a generally improved outcome in path length estimates. The number of steps detected is still of equal error rate with the C4.5 model, with the only difference being the latter is more open in including in detected steps. The results of the INS with the SVM model is presented in Table 6.

Subject	Total steps	Total length	Accuracy	Recall	Sensitivity
A	29	19.009	0.8857	0.85	0.93
B	33	21.682	0.8378	0.9	0.867
C	38	24.822	0.8571	0.95	0.73

Table 6: Performance statistics of an INS with a SVM model on the multi-movement route test.

In this multi-movement test, the effect of the models towards the succeeding modules are more apparent. The C4.5 model managed to avoid 12 false steps on average, while the SVM model avoided 12.67 steps on average. The step detection module, in turn, affected the stride length estimation, thus reducing the path length estimate to a closer approximation.

It must be noted that a high recall rate is important in this research. Though a higher recall increases the possibility of false positives, a better step detection module can remedy the problem.

5. CONCLUSION

This work shows that the C4.5 and SVM pedestrian locomotion models achieved a high accuracy and recall in their 10-fold cross validation tests. The straight route and multi-movement tests also proved that adding a pedestrian locomotion detection module significantly improved the output of the succeeding modules in the INS, especially for routes with both positive and negative pedestrian locomotion movement. The proposed INS had more false positives, but as aforementioned, false positives can be solved with better step detection algorithms. It will be important to take effort in decreasing the false negatives, but given the INS evaluation, it is clear that the additional module's benefits outweighed the disadvantages. The study also demonstrated that instead of using a complex step detection algorithm, adding a module that analyses the pedestrian locomotion movement can also refine the outcome of the INS in general.

The module, however, can significantly impede the performance of the INS if it incorrectly classifies positive pedestrian locomotion movement as negative. Since the modules of an INS depend heavily on each other, the prediction module may also contribute in the error. Another weakness is the module's insistence to label a movement as negative or positive. As the window spans a second, it is possible that a user shifted from a positive to a negative pedestrian locomotion movement in that duration. The model, unfortunately, cannot handle this and is forced to choose between positive or negative.

Future research can work with to create a better model that will include more data subjects for the training data. Since the data was only collected from people with a stable gait, later research can delve into unstable gaits. They can also look into considering different positions for the phone, as in this study only one position was examined. Additional tests that can further test the capabilities and weaknesses of the solution will also be needed.

Furthermore, the model can go further by classifying what kind of movement the user is performing. Instead of just ”positive”, it will say ”walking down the stairs”. With this, the INS can be expanded to different applications too.

6. ACKNOWLEDGEMENT

This research is made possible with funding from the Philippine government. The researchers would like to thank the Department of Science and Technology's Engineering Research and Development for Technology (DOST-ERDT) program for its incalculable support for this study.

7. REFERENCES

[1] P. Bahl and V. N. Padmanabhan. Enhancements to the radar user location and tracking system. 2000.

[2] P. Bahl and V. N. Padmanabhan. Radar: An in-building rf-based user location and tracking system. In *INFOCOM*, pages 775–784, 2000.

[3] R. Battiti, T. L. Nhat, and A. Villani. Location-aware computing: A neural network model for determining location in wireless lans. Technical report, 2002.

[4] M. Brunato and R. Battiti. Statistical learning theory for location fingerprinting in wireless lans. *Computer Networks*, 47(6):825–845, 2005.

[5] J. Chung, M. Donahoe, C. Schmandt, I.-J. Kim, P. Razavai, and M. Wiseman. Indoor location sensing using geo-magnetism. In A. K. Agrawala, M. D. Corner, and D. Wetherall, editors, *MobiSys*, pages 141–154. ACM, 2011.

[6] N. Karlsson, E. D. Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. E. Munich. The vSLAM algorithm for robust localization and mapping. In *2005 IEEE International Conf. on Robotics and Automation, ICRA 2005*, 2005.

[7] S.-W. Lee and K. Mase. Recognition of walking behaviors for pedestrian navigation. In *Proceedings of the 2001 IEEE International Conference on Control Applications*, UbiComp '12, pages 1152–1155, 2001.

[8] F. Li, C. Zhao, G. Ding, J. Gong, C. Liu, and F. Zhao. A reliable and accurate indoor localization method using phone inertial sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 421–430, New York, NY, USA, 2012. ACM.

[9] J. D. Martin, J. Krosche, and S. Boll. Dynamic gps-position correction for mobile pedestrian navigation and orientation. In *Proceedings of the 3rd Workshop on Positioning, Navigation and Communication*.

[10] A. B. Melissa Selik, Richard Baranuik. Signal energy vs signal power. Online, August 2004.

[11] V. Otsason, A. Varshavsky, A. LaMarca, and E. de Lara. Accurate gsm indoor localization. 2005?

[12] S. L. M. K. Y. Ronald E. Walpole, Raymond H. Myers. *Scientists and Engineers: Guide to Probability and Statistics*, chapter 1. Pearson Education South Asia Pte. Ltd., 2010.

[13] S. Tan. *Linear Systems*, chapter 8. University of Auckland, 2008.

[14] A. Teuber and B. Eissfeller. A two-stage fuzzy logic approach for wireless lan indoor positioning. In *IEEE/ION Position Location Navigat. Symp*, volume 4, pages 730–738, April 2006.

[15] T. N. Thanh, Y. Makihara, H. Nagahara, Y. Mukaigawa, and Y. Yagi. Performance evaluation of gait recognition using the largest inertial sensor-based gait database. In *ICB*, pages 360–366, 2012.

[16] R. E. Walpole. *Introduction to statistics*, chapter 2. Pearson Education South Asia Pte Ltd., 2009.

[17] M. Youssef, A. Agrawala, and A. U. Shankar. WLAN location determination via clustering and probability distributions. In *Pervasive Computing and Communications (PerCom).Proceedings of the First IEEE International Conference*, pages 143–150, 2003.

[18] M. Youssef and A. K. Agrawala. Handling samples correlation in the horus system. In *INFOCOM*, 2004.