

SmartSpaghetti: Accurate and Robust Tracking of Human’s location

Mostafa Uddin, Tamer Nadeem, Kurt Maly and Ajay Gupta
Department of Computer Science, Old Dominion University
{muddin,nadeem,maly,ajay}@cs.odu.edu

Abstract—An important tool of the *Lean* management is the “Spaghetti Diagram”, which helps to establish the optimum layout for a department or ward based on observations of the distances traveled by patients, staff and/or products (e.g., x-ray machines). The spaghetti diagram is usually created manually in which the movements of the staff member or patient are visually observed, which suffers from several challenges. Therefore, In our previous work, we reported the development of *SmartSpaghetti* system, which generates the Spaghetti Diagram in an automated and non-intrusive way by tracking a human’s location using the smartphone’s inertia sensors. In this paper, we address the challenges of the *SmartSpaghetti* system to track the human’s location using the distance and the direction change estimation in an accurate and robust way. Finally, we evaluate our distance and direction change estimation technique of the *SmartSpaghetti* system.

I. INTRODUCTION

Now-a-days many leading healthcare institutions are following a management approach called *Lean* in order to improve the quality and the efficiency of the healthcare system. *Lean* is a quality improvement philosophy based on the Toyota Production System that processes to maximize customer value while minimizing the waste. Recent studies [1], [2] show that healthcare organizations are adopting *Lean* management to improve their process and outcomes, reduce costs, and increase satisfaction among patients, providers, and staff. An important tool of *Lean* management is the “Spaghetti Diagram”, which help to establish the optimum layout of a department or ward based on observations of the distances traveled by patients, staff and/or products (e.g., x-ray machines).

Currently, the spaghetti diagram is usually created manually in which the movements of the staff member or patient are visually observed, and then, are manually drawn as lines on the layout diagram of the area under concern (figure 1a). Clearly, this traditional way is very inefficient and tedious. We proposed *SmartSpaghetti* system in our previous work [3] where we leverages the inertia sensors (i.e accelerometer, gyroscope, magnetic sensor, etc.) on smartphones carried by physicians and staff members to detect and track their movements and map these movements to flow paths in order to generate the Spaghetti Diagram in an automated, low cost, low overhead, and transparent way. More specifically, the system estimates the changes in user’s direction as well as user’s traveled distances. As a continuation of our work, in this paper we improvise the *SmartSpaghetti* system to use smartphone’s sensors to track the human’s location in an accurate and robust way.

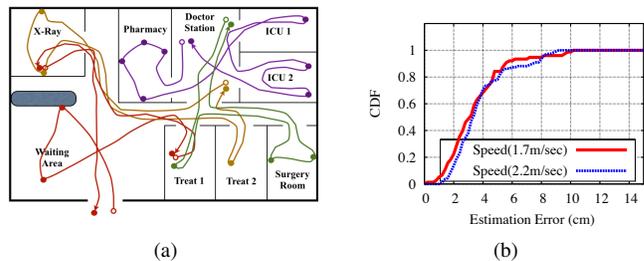


Fig. 1: (a) Example of Spaghetti Diagram. (b) Stride length estimation error at two different speeds.

In our previous work [3], we mentioned several challenges in regards of distance and direction change estimation. For example, In distance estimation it is challenging to convert the number of strides to the actual distance traveled by the user. Typically the stride length varies from user to user based on their height and weight. More importantly, even for the same user stride length can vary with the speed of walking.

Detecting the users direction changes, regardless of the orientation and the position of the smartphone (i.e. shirt pocket, pant pocket, in bag, in user’s hand, attach to belt etc.), is a challenging task. Given the inaccuracy of compass/magnetic field sensor in smartphone in indoor environments due to the surrounding ferromagnetic devices similar to the ones at hospital environment, we develop in this paper a technique to find the relative rotation/turn of the user with respect to the user’s movement direction rather than using absolute compass reading.

We summarize our contribution of this paper as follow:

- Development of a stride detection algorithm to efficiently detect the users stride using inertial sensors. Given different users have different strides and speeds, the proposed algorithm is adaptable to detect and estimate different stride lengths corresponding to different users and different speeds.
- Development of a robust direction change detection algorithm that infers a user’s relative direction changes by fusing multiple sensors of the smartphone.
- Evaluate the developed distance and direction change estimation technique.

II. BACKGROUND & RELATED WORK

In our previous work of *SmartSpaghetti* [3], we have used the state-of-art Dynamic Time Wrapping(DTW) algorithm to detect the user’s stride. Then we multiply the number of detected strides with the Human’s average stride length

to estimate the distance traveled by the user. Unlike our previous work, in this paper, we develop an adaptive DTW algorithm that measures the frequency of a stride. Then we build the well know personalized stride length model that map the frequency of strides to the stride length. Thus we estimate the distance traveled by the user more accurately. In direction estimation, we use the orientation sensor (from Android API) in our previous work. Note that, In our previous work, users have to hold the phone in their direction of walking, which is unlikely in our scenario. In addition, we found that the orientation sensor reading is highly noisy due to the surrounding ferromagnetic devices. Therefore, in this paper, we use the sensor fusion technique to fuse the accelerometer and the gyroscope sensors to measure the direction of the user’s walking regardless of how he/she carries his/her phone.

Recent related works [4], [5] utilize the existing inertial sensors in off-the-shelf smartphones to track the user’s indoor location, but they depend heavily on the knowledge of the environment. For example, the proposed system in [4] highly depends on the layout of the building, which is not practical to build the Spaghetti Diagram. The UnLoc system [5] requires to know enough landmarks of the environment in order to calibrate the estimated location using inertial sensors. However, besides the overhead of the initial training phase to build the database of landmarks, the UnLoc performance deteriorates when there are not enough distinguishable landmarks.

III. DISTANCE ESTIMATION

In distance estimation we estimate the distance at each stride. First, In order to detect the stride, we apply an adaptable stride detection algorithm. Second, we utilize a personal stride model to infer the user’s stride length. Thus we have following two actions, i) *Stride Detector*, and ii) *Personalized Stride Model*.

Stride Detector: In stride detection we use the commonly used accelerometer and the gyroscope sensors of the smartphone to detect and track user’s stride. Figure 3a, shows the details of our adaptive stride detection algorithm. Given that the stride detection is activated when a user begins moving, we use a change in gyroscope sensor reading above certain threshold as an indication of movement and initiate the stride detection algorithm to start capturing the accelerometer data. In our implementation, we set this threshold value to 0.3. Given that stride length is proportional to the walking speed [7], m_{max} represents the maximum length of a person stride in terms of samples size and is defined as follows, $m_{max} = \frac{s_{max}}{v_{max}} \times f_a$, where s_{max} is the maximum length of a person stride, v_{max} is the maximum walking speed of person, and f_a is the collection frequency of accelerometer samples from the smartphone. Similarly, m_{min} , which is the minimum length of a person stride in terms of samples size, is defined using s_{min} and v_{min} values. We used the $s_{max}, v_{max}, s_{min}$, and v_{min} values defined in [7] in our stride detector algorithm. Since we used $f_a = 50$ samples/sec in our implementation, the corresponding m_{max} and m_{min} values we used are 65 and 25 respectively.

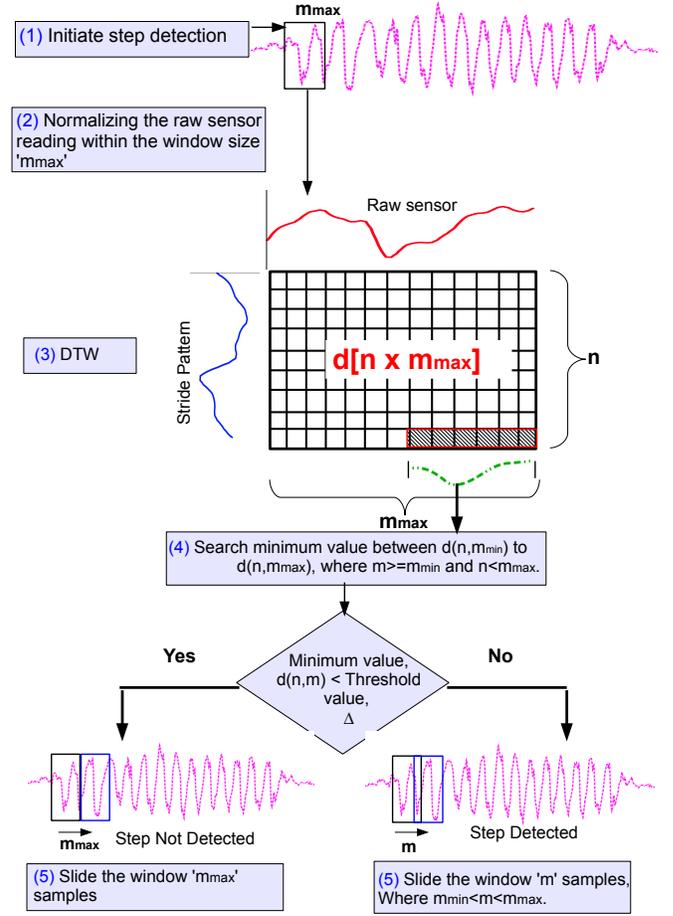


Fig. 2: Adaptable Stride Detection module in SpyLoc localization system.

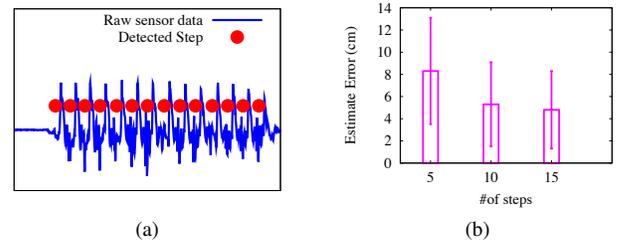


Fig. 3: (a) Stride detection in raw sensing data. (b) Accuracy of the stride model in estimating the average stride length.

After collecting m_{max} number of raw 3-axis accelerometer samples, we calculate the vector magnitude of each 3-axis accelerometer sample in order to make our algorithm independent of the user orientation. Similar to the other algorithms [6], [4], [5], we apply the IIR (Infinite Impulse Response) low pass filter to reduce the impact of the noise of the m_{max} samples. Then, we normalize the m_{max} samples before feeding it to the Dynamic Time Wrapping (DTW) algorithm [6]. The DTW algorithm compares the similarity between the predefined stride pattern (with size n samples where $n \leq m_{max}$) and the captured m_{max} samples to detect whether a stride exists within the m_{max} samples. Unlike correlation and threshold-based methods [4], [5],

DTW adaptively detect user strides regardless of the different lengths corresponding to different walking speeds. In our implementation, we used the predefined stride pattern of size $n = 45$ samples.

The DTW algorithm calculates $d[n \times m_{max}]$ matrix scores with positive values. The lower score of $d[i, j]$ indicates a better matching between predefined stride pattern of size i and the captured samples of size j . Unlike common use of the DTW algorithm [6], we search for a cell $d[n, m]$ with the minimum value between $d[n, m_{min}]$ and $d[n, m_{max}]$ cells. (the minimum value at the green dotted curve In the Figure 3a). If this minimum value is below a certain threshold Δ , then a stride length of m samples is detected. Otherwise, there is no stride detected within the captured m_{max} sample. By conducting several experiments, we set the threshold Δ to 0.4 in our implementation. If a stride is detected, then we shift the searching window for detecting the next stride by m samples. Otherwise we shift it by m_{max} samples. Figure 3a shows how accurately the strides detected by our scheme match the actual accelerometers samples corresponding to user strides in a walking experiment.

Personalized Stride Model: We use the commonly used following stride length model [6], [4] as our personalized stride model, $s = a \times f + b$, where s is the stride length, f is the frequency of strides, and a, b are the person-dependent constants. In order to define the personalized stride model, we have to calculate the constant parameters a, b for each user. In a controlled experiment we told the users to take few strides, then we measure the traveled distance to calculate their model parameters using line fitting algorithm.

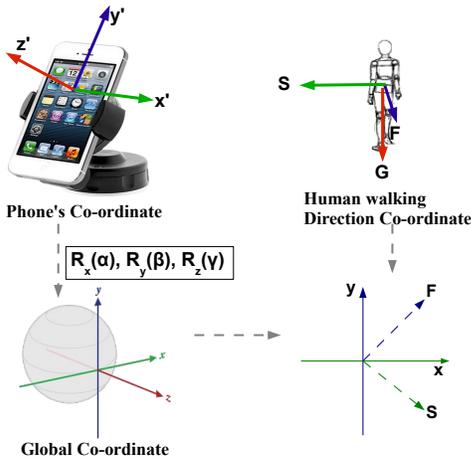


Fig. 4: Three different coordinate system and their relations.

Evaluation: In figure 3b, we plot the estimated stride length error by building the stride model from 5, 10, and 15 number of strides. Increasing the number of strides to build the model reduces the overall error of estimating the stride's length. In figure 1b, we also evaluate our adaptive stride length estimation technique for two different speeds. The speed 1.7 m/sec and the 2.2 m/sec is the normal and the fastest walking speed of a human being respectively. In both speed, we found almost similar distribution of estimation

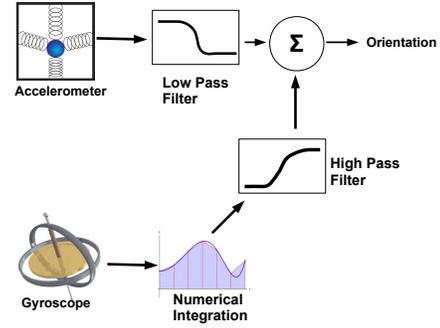


Fig. 5: Sensor Fusion block diagram.

error. More specifically, about 90% of the estimation error is less than 6cm for both speed.

IV. DIRECTION ESTIMATION

In a realistic environment, the orientation of the smartphone is independent with respect to the user's direction of movement. Therefore, It is a challenging task to determine the user's direction changes using the smartphone's sensors reading. In order to address this challenge, we consider three different coordinate systems, and their relation to each other. Figure 4 shows these three coordinate systems, First is the phone's coordinate system which is shown in the figure 4. Second, the human's walking coordinate system which represents the human's movement direction (figure 4). This human's walking coordinate system represents the forward direction, side, and gravity. The third coordinate system is the global coordinate system which represent the north pole, the east and the gravity of the earth. The global coordinate system is a fixed coordinate system, while the other two coordinate systems are not fixed. For example, the phone's coordinate system can vary for different orientation of the phone. The human's coordinate system also changes with the direction of human's movement. Thus our idea is to map the phone's and human's walking coordinate systems to the global coordinate system. In phone's coordinate system, we need to determine the three rotation or orientation ($\alpha_x, \beta_y, \gamma_z$) around the three axes to transform the phone's coordinate system to the global coordinate system.

In determining the phone's orientation, we just use the accelerometer and the gyroscope sensors. We avoid the magnetic field/compass sensors due to its high sensitivity to the surrounding magnetic devices. Figure 5 shows the block diagram of the sensor fusion technique that we have used to determine orientation of the phone with respect to the global coordinate system.

In human's walking coordinate system, the gravity (G) axis is same as the $-z$ axis in global coordinate system. Moreover, the other two axis the forward direction (F) and the side (S) are in x, y plane of the global coordinate system. However, the orientation of the F and S might not be same with respect to y and x axes (figure 4). Note that, the linear accelerometer reading from the smartphone is in respect to the phone's coordinate system. Therefore we use the estimated orientation $\alpha_x, \beta_y, \gamma_z$ to transform

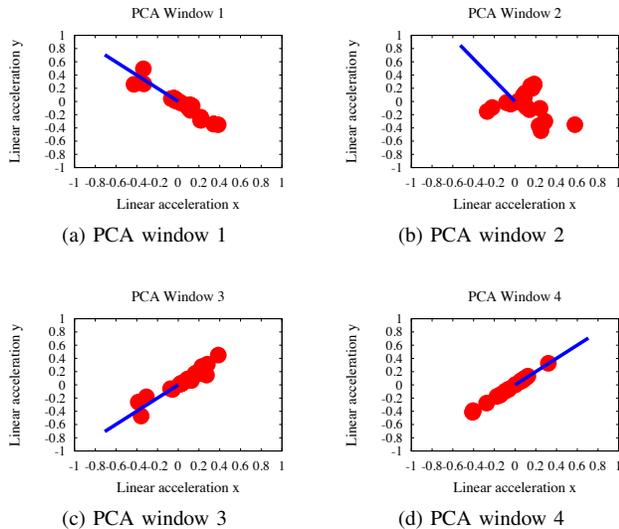


Fig. 6: Sequence of non-overlapping PCA window. The blue line represents the PCA axis.

the linear acceleration reading of the smartphone to global coordinate system. Now, if we plot the linear acceleration reading in the x, y plane of the global coordinate system then the highest variation of changes of the projected linear accelerometer sensor reading will indicate us the direction of the human's walking movement, which is basically the forward direction F -axis. We apply the *Principal Component Analysis (PCA)* [8] on the transformed samples to find out the direction of the F axis in the x, y plane. Note that in the implementation we use 25 samples as our PCA window where 33.36 is our sampling rate per second. Figure 6 shows four sequential PCA window where the user took a 90 degree turn. The dots in the plots are the transformed linear acceleration samples in the x, y plane of the global coordinates. The straight line in the plot represents the PCA axis which is the user's walking direction. In figure 6, the PCA window 2 to 3 shows the transition of the user's 90 degree turn.

Evaluation & Discussion: In figure 7a, we evaluate the stability of our direction estimation technique while user is walking straight. In straight walking, the *Cumulative Distribution Function (CDF)* [9] plot shows that 98% value of the estimated direction change is less than 10 degrees. In figure 7b, we evaluate our direction estimation technique where we took a sequence of direction changes (x axis), and the y axis shows our estimated direction changes. The solid line in figure 7b represents the ideal situation and the dotted line represents the estimation. Note that, we use the right and the left turn as the positive and the negative direction change respectively. In this evaluation experiment, we place the phone in the pant pocket while walking.

We observe, the linear acceleration samples in a PCA window provides information about the walking speed of the user. Over the experiments, the range of sample's value in a PCA window spread out more as the user starts to walk faster. For example, the PCA window 2 in figure 6, the linear

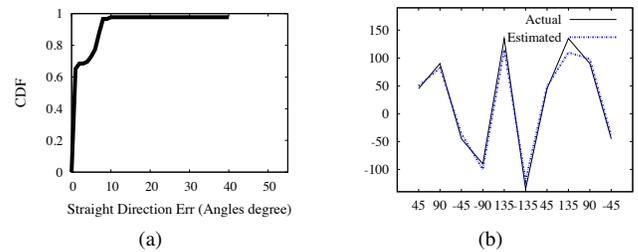


Fig. 7: (a) CDF of Direction error while user is walking straight. (b) Evaluating different degree of relative direction changes.

accelerometer samples are less spread out compared to other PCA windows. Note that, In the PCA window 2, the user initiates the turning event and before the turning the user slows down. Moreover, before the turning we observe that the sample points are more randomly scattered, where in other PCA windows the sampled points show a certain pattern of direction.

V. CONCLUSION

In this paper, we develop two important techniques for the *SmartSpaghetti* system; distance and direction change estimation. The distance estimation technique is based on the mechanism of detecting strides and estimating the stride's length. In this paper, we develop the adaptive stride and stride's length estimation technique regardless of the users walking speed. Moreover, in the direction estimation technique we relate the three coordinate systems, thus we can estimate the direction changes regardless of what the orientation of the user's phone is. We also apply the sensor fusion technique, where we use both the accelerometer and the gyroscope sensors. Therefore it reduces the impact of the surrounding noises from the ferromagnetic devices.

REFERENCES

- [1] John S. Toussaint, and Leonard L. Berry, *The Promise of Lean in Health Care*, Mayo Clinic Proceedings, Volume 88, Issue 1, Pages 74-82, January 2013
- [2] James P. Womack et al., *Going Lean in Health Care. IHI Innovation Series white paper*. Cambridge, MA: Institute for Healthcare Improvement; 2005.
- [3] Mostafa Uddin, Ajay Gupta, Kurt Maly, Tamer Nadeem, Sandip Godambe, and Arno Zariitsky, *SmartSpaghetti: Use of Smart Devices to Solve Health Care Problems*. International Workshop on Biomedical and Health Informatics, 2013
- [4] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization,". In Proceedings of the 18th annual international conference on Mobile computing and networking, 2012.
- [5] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: unsupervised indoor localization." In MobiSys 2012.
- [6] 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.
- [7] S. H. Collins and A. D. kuo, "Two independent contributions to step variability during over-ground human walking." Journal of Applied Physiology, 2011.
- [8] K. Pearson, "On Lines and Planes of Closest Fit to Systems of Points in Space". Philosophical Magazine 2 (11): 559-572, 1901.
- [9] D. Zwillinger, S. Kokoska, "Standard Probability and Statistics Tables and Formulae". CRC Press. p. 49, 2010.