

Received October 12, 2016, accepted November 4, 2016, date of publication November 18, 2016, date of current version March 8, 2017.

Digital Object Identifier 10.1109/ACCESS.2016.2631000

# Orientation and Displacement Detection for Smartphone Device Based IMUs

ABHIJIT SUPREM<sup>1</sup>, VISHAL DEEP<sup>2</sup>, AND TAREK ELARABI<sup>3</sup>

<sup>1</sup>Columbia University, New York City, NY 10027 USA

<sup>2</sup>California State University at Fresno, Fresno, CA 93740 USA

<sup>3</sup>Penn State Behrend, Erie, PA 16510 USA

Corresponding author: T. Elarabi (tkelarabi@gmail.com)

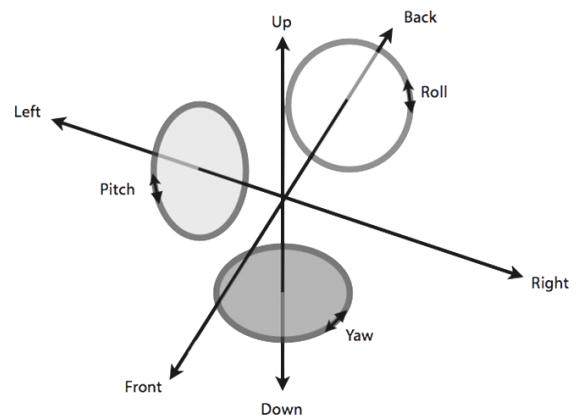
**ABSTRACT** Commonly, navigation uses global positioning system/network (GPS) signals to determine location. In addition, GPS data can be collected over time to determine the path taken. In spite of the ubiquity of GPS signals on the earth's surface, there are certain locations where GPS signals are not available, such as inside buildings or tunnels. Therefore, to determine accurate positioning in areas where the GPS signals are unavailable, an inertial measurement unit (IMU) can be used in conjunction with the GPS data. Modern IMUs are small enough to be contained in Microelectromechanical systems (MEMS) chips, including smartphone devices, such as iPhones. This paper studies the integration between the GPS signal and the collected data from smartphones' MEMS sensors. It also investigates the possibility of using the integrated GPS/MEMS information to estimate route when the GPS signal is missing. We propose estimating the missing GPS signal by sensor integration of smartphone data. This paper would enhance the GPS navigation to determine exact positions even in the case of signal failures. Modern IMUs are expensive, and this paper shows that GPS/IMU integration can be accomplished with off-the-shelf navigational components. This paper proposes a novel technique for estimating the missing GPS route by integrating data from the sensors available in modern smartphones.

**INDEX TERMS** Global positioning system, signal processing, navigation, inertial navigation, IMU integration, pedestrian tracking, access denied.

## I. INTRODUCTION

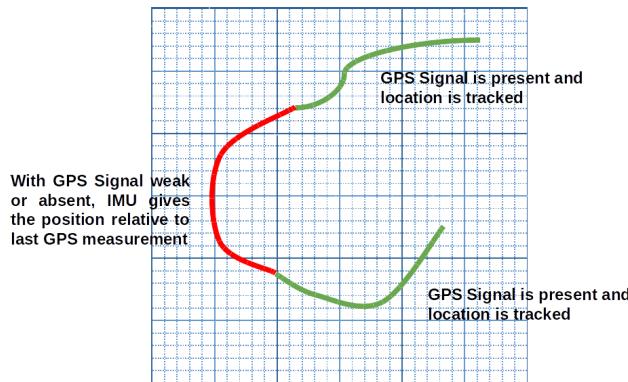
The Global Positioning System/Network (GPS) is used for navigation under a variety of approaches such as path tracking, location mapping, and position detection. However, GPS signals are not ubiquitous. The satellites cannot be reached from locations on the earth's surface with significant interference, such as within buildings, tunnels, or other locations with any physical barriers between the receiver device and the sky [1]. As such, a direct line-of-sight is required for proper position detection and tracking [2], [3].

It can be possible to fill in gaps between signal measurements from GPS satellites in such *access-denied* locations by utilizing some form of inertial measurement unit (IMU). Therefore, to determine accurate positioning and facilitate tracking in access-denied areas, an IMU can be used in conjunction with GPS data for orientation and dead reckoning [4]. The basic IMU contains an accelerometer and a gyroscope, Figure 1 [5]. Most modern smart phones contain a wide array of sensors in addition to the accelerometer and gyroscope, such as magnetometer, orientation sensor, barometer, and temperature sensors.



**FIGURE 1.** Accelerometer and gyroscope sense linear and angular acceleration.

This research project aims to integrate GPS data with the information collected from a smartphone's IMU sensors. It is possible to use data collected from the smartphone sensors to



**FIGURE 2.** Signal integration from IMU data for GPS tracking.

estimate the missing GPS signals [6], as shown in Figure 2. The embedded accelerometers, gyroscopes, and magnetometers can estimate the missing GPS signal. This research work would enhance navigation and tracking in a variety of applications where MEMS-based IMUs can be used, such as in context of Internet-of-Things (IoT) to facilitate device-to-device position tracking, or for pedestrian-tracking in access-denied areas [7]–[9].

## II. BACKGROUND

We have used smartphone-based MEMS sensors for the IMU measurements. Note that these MEMS sensors are similar to off-the-shelf integrated IMUs with respect to accuracy and drift. So our approach can work for both smartphone-based IMUs and low-cost off-the-shelf integrated chip IMUs.

### A. ACCELEROMETERS

An accelerometer is a common MEMS device. In the IMU, it is used to determine translational acceleration. In addition, it can be used to determine the yaw and pitch as well [10], to correct the gyroscope data, which tends to drift. Note that this acceleration needs to be corrected for the gravitational influence. The smartphone employs a 3-axis accelerometer. The accelerometer detects a force acting in the opposite direction to the acceleration vector as determined by the net acceleration. The raw data from the smartphone can be converted to readable data through calibration. The calibration involves determining any drift in the acceleration measurements in the positive and negative directions for each axis. The drifts and any potential deviations from Earth-gravity (since the acceleration measurements are normalized, the device measured gravity with a unit value) are then corrected.

The smartphone device we use comes equipped with a triaxial accelerometer sensor - the SMB380 from Bosch Sensortec (datasheet at [11]). Relevant details about the accelerometer are summarized in Table 1. We note that the hardware filter in the sensor is not enough to remove the ambient and temperature noise. As such, a digital butterworth low-pass filter on the backend was implemented to remove high frequency noise.

**TABLE 1.** Relevant accelerometer specifications.

Parameter	Min	Typ	Max	Unit
Acceleration range	-4		4	g
Data resolution			10	Bits
Refresh rate	2700	3000	3300	Hz
Hardware filter		1500		Hz
Noise ( $n_{rms}$ )		0.0005		$g/\sqrt{Hz}$

### B. GYROSCOPE

A gyroscope measures the rate of angular velocity. In the IMU, the gyroscope is used to determine the orientation. Since the gyroscope has a high rate of error, its data drifts over time. The drift can be corrected by using a complementary filter [12]. We have noticed that it is also possible to correct sensor drifts by comparing the orientation obtained from the gyroscope with the gravity vector from the accelerometer.

We consider the device as 'facing up' when the accelerometer measures only Earth-gravity in the device-frame's local orientation on the Z-axis. At this time, the orientation with respect to rotation about the local X- and Y- axes is known to be zero. So, the measured orientation can be compensated for any integration errors.

**TABLE 2.** Relevant gyroscope specifications.

Parameter	Min	Typ	Max	Unit
Measurement range	-100		100	degree per second
Data resolution			8	Bits
Refresh rate		9090		Hz
Hardware filter				None
Noise ( $n_{rms}$ )		0.006		$dps/\sqrt{Hz}$

A L2G2IS gyroscope from STMicroelectronics is used in the study [13]. Relevant specifications are presented in Table 2. The refresh rate for the gyroscope is nearly 3 times higher than the rate for the accelerometer; further, the absence of a hardware filter entails higher noise to signal ratio. We also note that the gyroscope traditionally operates at much higher sampling rate than the accelerometer; hence its noise power is significantly higher than the accelerometer.

### C. MAGNETOMETER

The magnetometer is a sensor that detects the strength of magnetic fields. It is commonly used to determine heading by using the Earth's magnetic field. As such, the magnetometer can be used as a digital compass to get NSEW headings [14]. The magnetometer present in the smartphone functions properly when the device is in a planar position.

## III. ORIENTATION DETECTION

The implementation of GPS/IMU integration was undertaken by converting the smartphone to an IMU device. This is accomplished by integrating together data from the smartphone's accelerometer, gyroscope, and magnetometer. The implementation method is described as follows.

### A. ACCELERATION MEASUREMENT

The smartphone's accelerometer gives the device's acceleration in X, Y, and Z coordinates. This acceleration can be integrated over time to determine the device's velocity. The velocity can be integrated to determine the device's change in position. Since the accelerometer measures gravity as acceleration, the effects of gravity must be taken into account. When taken with an initial position from the GPS coordinates, the displacement data from the accelerometer can be used to determine accurate position via dead reckoning. However, the accelerometer's raw data is given in local device coordinates; thus, if the device's orientation is changed, the axes orientations also change. So, if the device is turned on its side, the Z-axis no longer points upwards; instead, it is also rotated. While the acceleration vectors can be used to determine the roll and pitch angles, these values may not be appropriate for real-time calculations. Instead, the roll and pitch angles calculated by the acceleration can be used to correct the gyroscope measurements. A complementary filter is used for this approach to drift correction.

### B. GYROSCOPE AND MAGNETOMETER FOR ROTATION MEASUREMENT

So, the acceleration vectors can be reoriented with the gyroscope, which gives the rate of angular rotation. This can be integrated to yield the angular displacement and, given an initial, known orientation, the device's angular position can be calculated at all times. Once the orientation is determined, the acceleration axes can be properly reoriented to correspond to a predetermined orientation so as to maintain constancy with the data, i.e. the virtual acceleration orientation can be normalized to follow an earth-relative orientation, rather than a device-relative orientation. However, the smartphone's gyroscope can give erroneous data when jolted. Errors can accumulate extremely quickly, unless a Kalman filter, or in our approach, a complementary filter, is implemented. Furthermore, the gyroscope is unstable and lower angular velocities do not register on the sensor. Thus, if the orientation changes slowly over time, it would be impossible to verify.

Another solution is to use only the magnetometer. Since the magnetometer's readings are based on the earth's magnetic field, the raw data is considerably more accurate. The magnetometer can be used as a digital compass to determine the device's orientation around the Z-axis. In fact, this leads to a more appropriate earth-relative orientation system using geodetic format, where North corresponds to the X-axis and East corresponds to the Y-axis. A constraint is introduced in this system while the device can turn on the Z-axis, it cannot turn on the other two axes, i.e. it must remain level at all times, as the gyroscope is not extensively used and the magnetometer stops functioning properly when the device is tilted along the X- or Y-axis.

Our approach uses the accelerometer in conjunction with the magnetometer to determine the device's earth-relative position. The gyroscope is then used to correct

the Earth-frame measurements and to reorient the net vectors.

### IV. EARTH FRAME ACCELERATIONS - WITHOUT GYROSCOPE

The smartphone's raw data consists of uncorrected readings from the three-axis accelerometer, three-axis magnetometer, and three-axis gyroscope. The accelerometer data must be corrected and calibrated for fluctuations that may occur due to occasional bumps or shakes. These fluctuations do not include the jerks from footsteps, which significantly affect the readings [15]. This is accomplished by taking some data with the device facing in the positive and negative X, Y, and Z axes and correcting any drifts with calibration equations, shown in Equation 1, where  $m$  represents the slope of any drift,  $t$  represents the time step and  $offset$  represents the DC offset. The uncorrected signal is translated by the correction factor for the drift.

$$Acc_{corrected} = Acc_{uncorrected} - (m \cdot t + offset) \quad (1)$$

#### A. DRIFT CORRECTION

Initialization drift, or turn-on bias for the sensors is corrected as per Equation 1. During the calibration step, the drift slope is determined. This is obtained by placing the device level on each axis and measuring drift. The waveforms are passed through a high pass filter and a regression analysis is performed to determine drift order. For linear drift, Equation 1 suffices; the slope  $m$  is the negative inverse of the drift slope - this negates the drift offset. Drift with higher order is rare and was not observed during any calibration. Figure 3 shows the stages of calibration for drift correction of the accelerometer. Corrected accelerometer readings for one sample walk are shown in Figure 4. Gyroscope drift is covered in Section V-A.

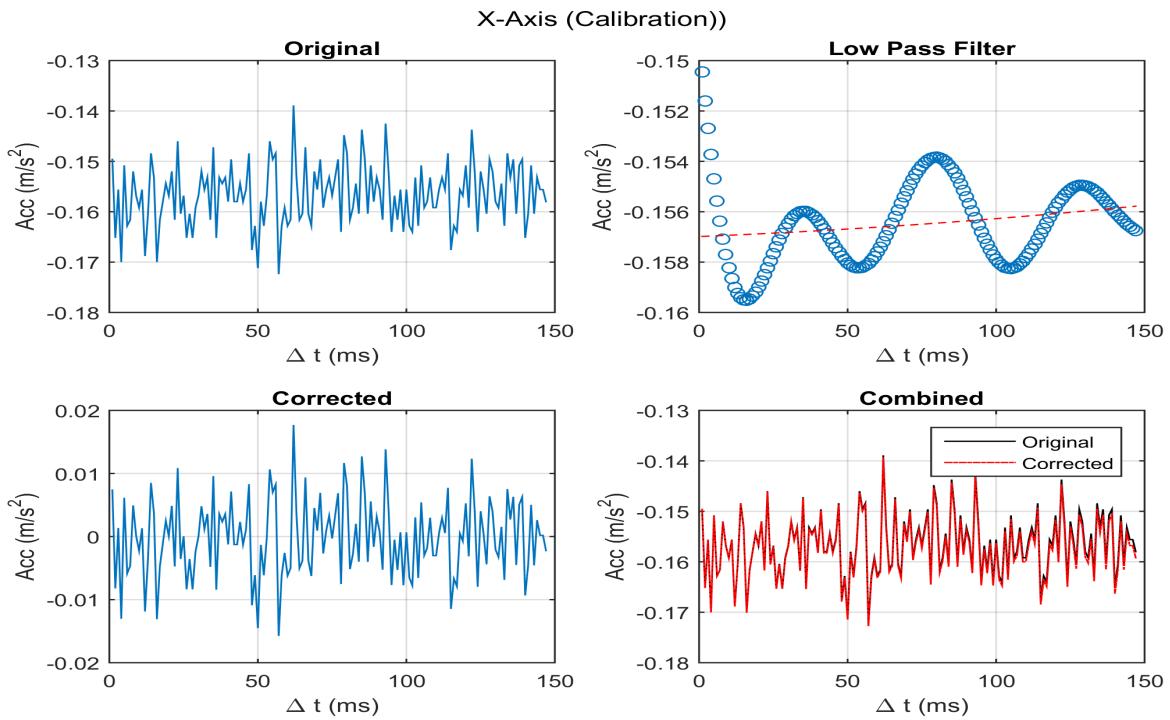
### B. ACCELEROMETER-MAGNETOMETER APPROACH FOR EARTH FRAME ACCELERATIONS

The device frame axes are assumed to be fixed. Since the device may rotate, the data from these axes must be corrected relative to device orientation in the world frame. So, after the data are calibrated with Equation 1, the vectors are normalized and unit vectors obtained. Since the background acceleration is always downwards towards Earth, we can assume the acceleration vector  $\mathbf{G}$  to be the gravity vector. Furthermore, the magnetometer vector  $\mathbf{B}$  always points North because the magnetometer measures the magnetic field relative to the North pole. As a result, the cross product of  $\mathbf{G}$  and  $\mathbf{B}$  yields the magnitude of the vector pointing East, or the  $\mathbf{E}$  vector.

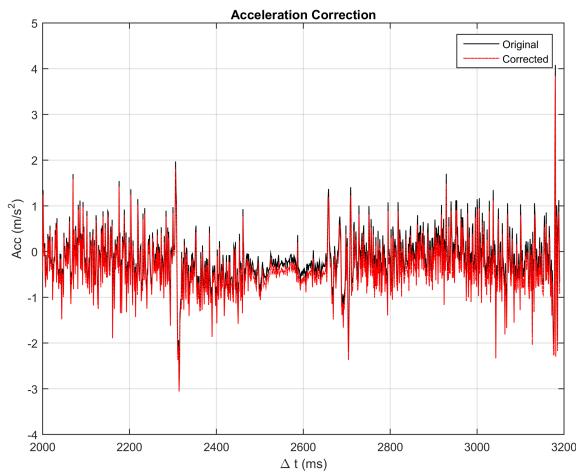
$$\mathbf{E} = u\mathbf{G} \times u\mathbf{B} \quad (2)$$

Note that although the magnetometer vector  $\mathbf{B}$  is considered the North vector, it cannot be used for calculation. Instead, the East vector  $\mathbf{E}$  crossed with  $\mathbf{G}$  yield the true normalized North vector.

$$\mathbf{N} = u\mathbf{E} \times u\mathbf{G} \quad (3)$$



**FIGURE 3.** Turn-on bias/drift correction.



**FIGURE 4.** Corrected accelerometer readings for sample walk.

These calculations can then be compared with the measurements from the reoriented acceleration vectors. We are currently studying an appropriate method for analyzing differences between these results for reducing errors in the acceleration measurements. Note that the results from this mentioned approach may not be correct when the device is on its side as the magnetometer suffers from significant heading errors.

### 1) MAGNETIC DECLINATION

Issues with magnetic declination towards the magnetic north pole are resolved trivially. The requirement for the end user

is to begin the procedure facing a pre-determined direction. We selected this to be True East. As such, our system knows initial orientation. So, the orientation detection presented in Section V is sufficient to accurately track device orientation. This software calibration of the magnetometer is necessary to ensure correctness of determined paths.

## V. ORIENTATION DETECTION AND COMPENSATION

We have shown a method to determine Earth-frame accelerations using only the accelerometer and the magnetometer.

### A. ROLL AND PITCH CALCULATION

The roll and pitch values allow correction of the gyroscope values. Note that the device must be initialized pointing to the Earth-relative X axis, i.e. the North pole, for orientation detection to function properly. The pitch and roll are defined in Equations 4 and 5, respectively.

$$\text{Pitch} = \arctan\left(\frac{G_y}{|G_{xz}|}\right) \quad (4)$$

$$\text{Roll} = \arctan\left(-\frac{G_x}{G_z}\right) \quad (5)$$

The roll and pitch are also obtained from the gyroscope calculations and then compared to these measurements. The complementary filter is used to correct the gyroscope readings.

The complementary filter is a weighted filter, similar to the Kalman filter. However, the weights are not updated; in effect, it is a static coefficient filter that is nevertheless useful in this scenario because of the fast computation and because

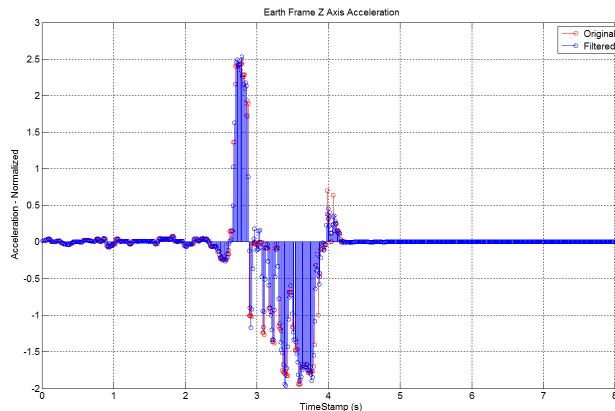
the error covariance matrix need not be calculated. Since the exact specifications of the smartphone sensors are not known, constructing the error covariance matrix is difficult.

### B. COMPLEMENTARY FILTER

The complementary filter combines accelerometer and gyroscope measurements to obtain the unified, gyroscope-drift free roll and pitch angles. We can consider a preliminary model for the filter:

$$\theta_{net} = W_g \cdot \theta_g + W_a \cdot \theta_a \quad (6)$$

where  $A_{net}$  is the true angular displacement,  $W_{g,a}$  are gyroscope and accelerometer weighting factors ( $W_a = 1 - W_g$ ) and  $\theta_{g,a}$  are the angular displacement obtained from the gyroscope and accelerometer, respectively.



**FIGURE 5.** Filtered accelerometer readings.

Equations 4 and 5 yield roll and pitch angles from accelerometer readings. These are passed through a low-pass filter to remove high-frequency noise components (Figure 5). The low-frequency signal is used for long-term correction of the noisy gyroscope data. Conversely, the angular velocity measured from the gyroscope is integrated to obtain short-term angular displacement. This displacement ( $\theta_g = A_{net} + \omega \cdot dt$ ) is passed through a high pass filter to remove steady-state signals (the drift) and pass the gyroscope's true angular velocity signal. The two components - angle measurement from gyroscope and accelerometer - are summed to obtain true angular displacement. The gyroscope is weighed heavily for higher sampling frequency. The weighting factor is determined by the gyroscope drift rate - the time constant for the filter should be less than the drift threshold. Note that for digital filters,

$$\tau = \frac{W_g \cdot dt}{1 - W_g} \implies W_g = \frac{\tau}{\tau + dt} \quad (7)$$

As such, the constants will change based on sampling rate and drift threshold. Equation 8 shows the utilized complementary filter with averaged constants.

$$\theta_{net} = 0.977 \cdot (\theta_{net} + \theta_g \cdot dt) + 0.023 \cdot \theta_a \quad (8)$$

### C. DIRECT COSINE ROTATION MATRIX

We transform the corrected roll and pitch values to a direct cosine matrix (DCM) with Equation 9 (where  $C_\alpha$  represents  $\cos(\alpha)$  and  $S_\alpha$  represents  $\sin(\alpha)$  and where  $\alpha$  is the yaw,  $\beta$  is the pitch, and  $\gamma$  is the roll). The matrix multiplication of the acceleration vectors in the local orientation with the DCM yields the earth frame accelerations. In the earth frame, we use geodetic mapping, where the X-axis is mapped towards East, the Y-axis is mapped to North, and the Z-axis is mapped to altitude.

$$W = \begin{bmatrix} C_\alpha C_\beta & C_\alpha S_\beta S_\gamma - S_\alpha C_\gamma & C_\alpha S_\beta C_\gamma + S_\alpha S_\gamma \\ S_\alpha C_\beta & S_\alpha S_\beta S_\gamma + C_\alpha C_\gamma & S_\alpha S_\beta C_\gamma - C_\alpha S_\gamma \\ -S_\beta & C_\beta S_\gamma & C_\beta C_\gamma \end{bmatrix} \quad (9)$$

These geodetic mappings are compared with the results from the earlier earth frame analysis from the accelerometer and magnetometer sensors. While the earlier method is not appropriate when the device has rotations about the X- or Y- axes.

### VI. EARTH-FRAME ACCELERATION

We use the earth frame accelerations to determine net displacement. The algorithm is summarized in Figure 6.

Note that the unit-accelerations must be converted to real accelerations considering the Earth Gravitation Constant. In addition, the constant gravity must be subtracted to get the true acceleration.

#### A. Earth Frame Rotation Matrix

The Earth Frame Rotation Matrix (EFRM) is different from the Body Frame Rotation Matrix  $W$  presented in Equation 9. The EFRM, denoted by  $B_M$ , is the rotation of the device with respect to the Earth frame geodetic system. Note that for correct usage of the rotation matrices, it is necessary that the device's axes are initially oriented with the Earth Frame. In the presence of GPS data, this can be corrected without initial orientation. Currently, we lack access to sample GPS data, and so cannot determine initial orientation.

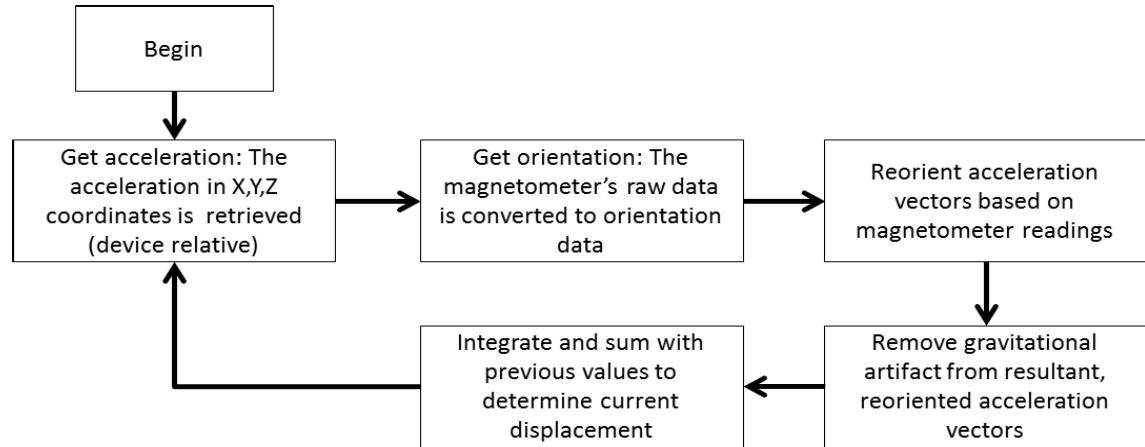
The rotation matrix  $B_M$  is determined as follows:

$$B_M = \begin{bmatrix} 0 & -G_z \Delta t & G_y \Delta t \\ G_z \Delta t & 0 & -G_x \Delta t \\ -G_y \Delta t & G_x \Delta t & 0 \end{bmatrix} \quad (10)$$

Note that a Body Frame Rotation Matrix determined from  $B_M$  can be more accurate than the matrix  $W$  determined from purely acceleration vectors. As such, a new rotation matrix  $G_M$  is determined as shown in Equation 11, where  $M_{unit}$  is the unit matrix (3 dimensions) and  $\sigma$  is the magnitude of the gyroscope's rotation vector ( $\sigma = \sqrt{G_x^2 + G_y^2 + G_z^2}$ ).

$$G_M = M_{unit} + B_M \cdot \frac{\sin(\sigma)}{\sigma} + B_M^2 \cdot \frac{1 - \cos(\sigma)}{\sigma^2} \quad (11)$$

The projections of the accelerations in the X, Y, and Z along the normalized geodetic frame is determined with the



**FIGURE 6.** Algorithm flowchart.

gyroscope-based rotation matrix. Note that the vectors  $\mathbf{E}$ ,  $\mathbf{N}$ , and  $\mathbf{G}$  denote the East (x-axis), North (y-axis), and height (z-axis) in the geodetic frame. The gravitational constant is  $9.8 \frac{m}{s^2}$ .

$$X_{acc} = G_{const}(X_{corr}E_x + Y_{corr}E_y + Z_{corr}E_z) \quad (12)$$

$$Y_{acc} = G_{const}(X_{corr}N_x + Y_{corr}N_y + Z_{corr}N_z) \quad (13)$$

$$Z_{acc} = G_{const}(X_{corr}G_x + Y_{corr}G_y + Z_{corr}G_z) \quad (14)$$

### B. EARTH FRAME VELOCITIES AND DISPLACEMENT DETERMINATION

The Earth frame velocities are determined by integration of the acceleration vectors. A second integration yields the Earth frame displacement along each axis:  $X_{displacement} = \int \int_{\Delta t} X_{acc}$ .

$$X_{displacement} = \int \int_{\Delta t} X_{acc} \quad (15)$$

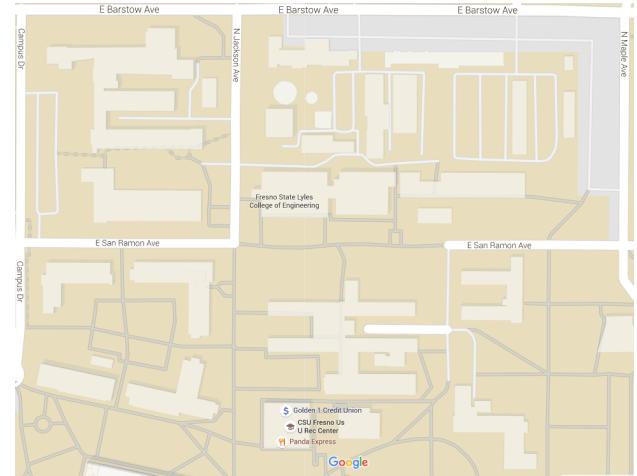
### C. DISPLACEMENT VERIFICATION

It is necessary to verify the net displacement. We accomplish this by considering the net Z-axis accelerations and observing the changes in acceleration caused by the steps. The accelerations can be classified into step-profiles and displacement calculated by summing the steps taken. Note that each step covers similar distances, and as such, net displacement is trivial. In addition, the orientation is determined from the algorithm presented in Section V. Consequently, the continuous tracking can be verified.

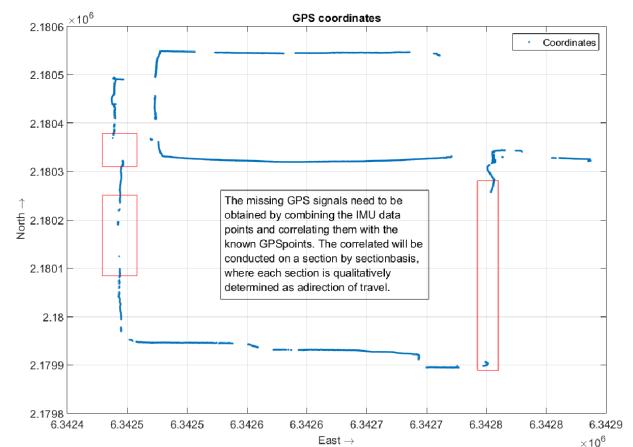
## VII. RESULTS

### A. DATA COLLECTION

A sample walk was taken with a GPS receiver and a smartphone-based IMU device. The path taken is shown in Figure 9, where the GPS data points are shown independently and superimposed on a map of the California State University, Fresno campus. Note that where the path goes through

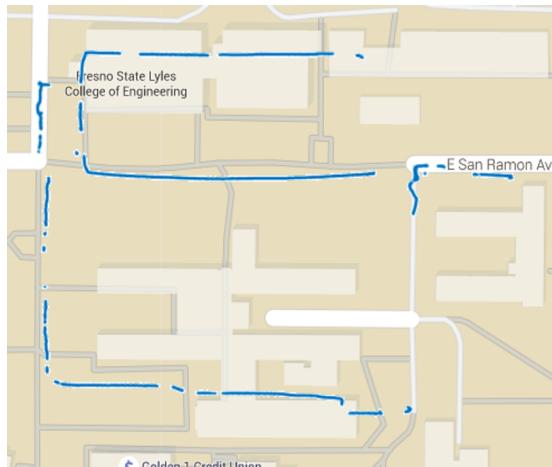


**FIGURE 7.** Walk location map.



**FIGURE 8.** Data collected from GPS unit.

buildings or in areas surrounded by partial or complete canopy cover, the GPS signal is weak or negligible and data points are missing.



**FIGURE 9.** GPS data superimposed on map.

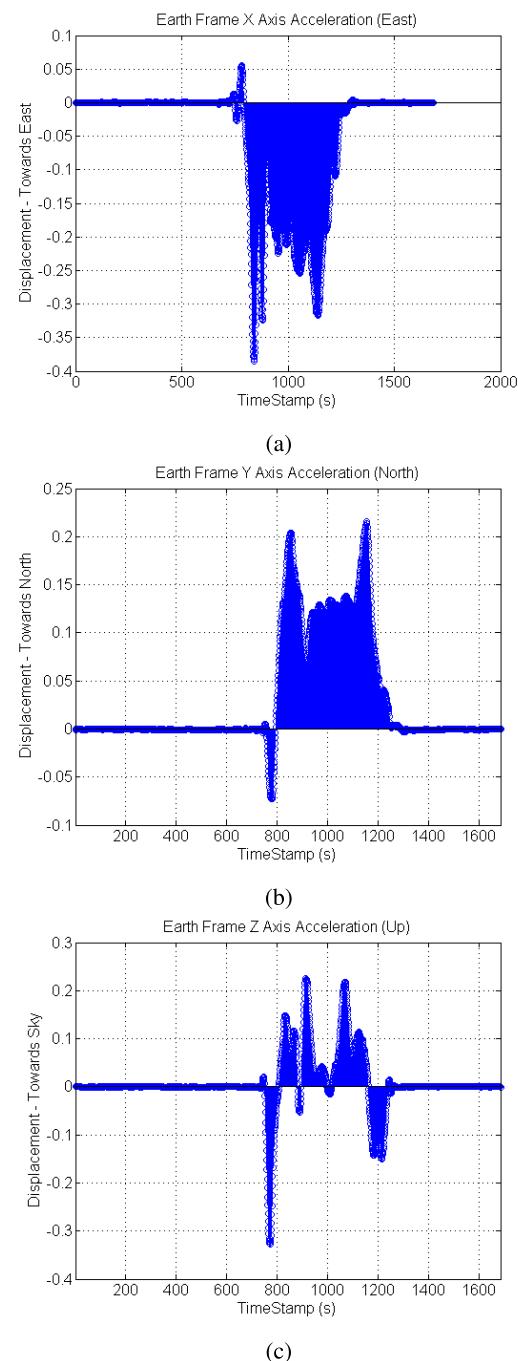
Results from a data sample are presented. The sample contains the individual accelerations and the magnetometer data. The gyroscope data given here is used to determine the rotation matrices. The algorithms and equations as outlined are applied to this data. The accelerations in the North (X axis), East (Y axis) and Z axes are determined. The resultant values are passed through a Savitzky-Golay FIR smoothing filter to remove noisy elements. The graphs of each component of the acceleration are given in Figures 10a, 10b, and 10c.

We can further note the orientation tracking in Figure 11. The planar orientation detection in Figure 11a, is obtained by combining the Euler angles from Figures 11b, 11c, 11d.

It is possible to see the movement of the device over time in each of the graphs. In the X axis acceleration, it is clear that the device moves towards the North between 2 and 3 seconds. The data at 4 seconds shows positive and negative accelerations. This indicates a bump and a reverse acceleration. The integration of this acceleration will actually yield zero average velocity. The Savitzky-Golay filter has reduced the error at 4 seconds to a more appropriate value as well. In the Y axis acceleration, the device moves East quickly before stopping between 2 and 3 seconds. The bump in the X axis acceleration is also present in the Y axis acceleration. Finally, the Z axis acceleration indicates the device moved up from approximately 2.6 seconds to 3 seconds and subsequently moved down in small step increments between 3 and 4 seconds.

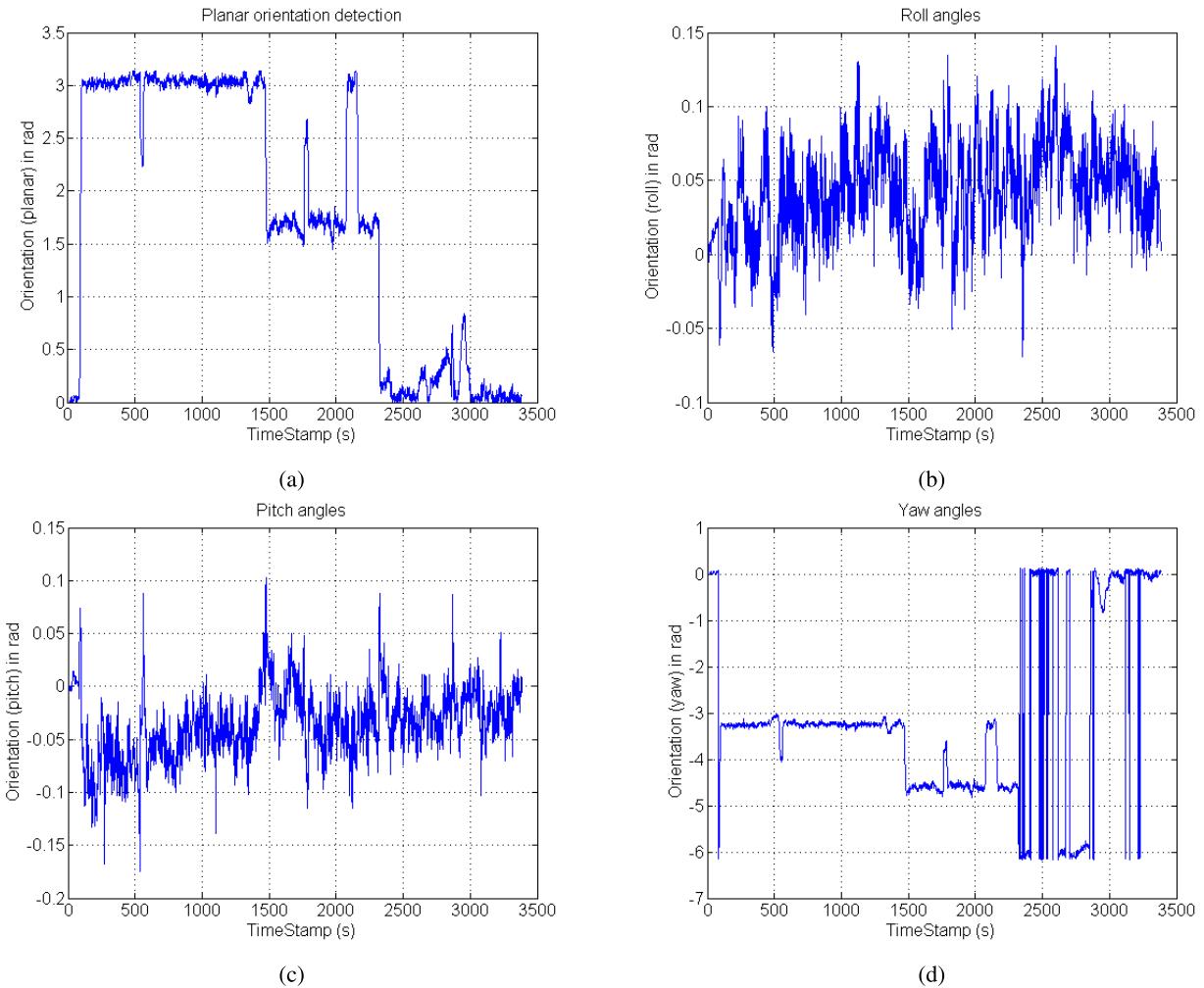
## B. IMU DATA

The step detect filter is used to extract distance traveled from the IMU's step-based accelerometer jolts (Figure 12). A threshold of 0.1g is chosen to identify legal steps from accidental jolts. A fuzzy peak-detect filter identifies the steps and combines the number with a stride-length estimator (in this case, we consider each stride to be 0.577m from empirical analysis).

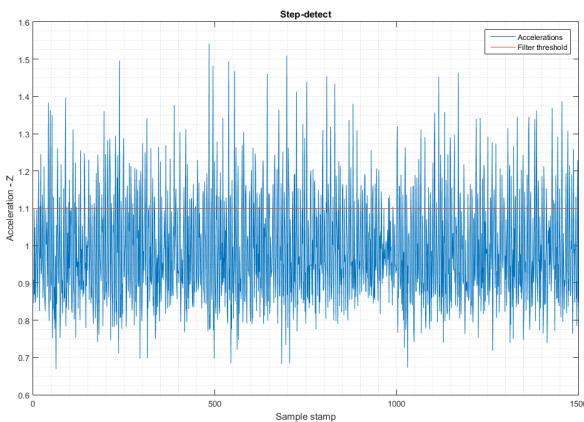


**FIGURE 10.** Earth-frame accelerations for three axes. (a) X Axis Net Acceleration. (b) Y Axis Net Acceleration. (C) Z Axis Net Acceleration.

The movement and rotations are combined to get the IMU's displacement (shown in Figure 13a). There are some clear deviations from the GPS data - the IMU and GPS data. The IMU data does not have any gaps because the sensors are, as mentioned, local. As such, the gaps in the GPS data can be corrected and estimated using the IMU data. This is accomplished with a MATLAB-based GUI developed for this purpose. The data is processed by MATLAB and a finalized, GPS/IMU integrated map is generated.



**FIGURE 11.** Earth-frame accelerations for three axes. (a) Planar orientation detection. (b) Roll angles. (c) Pitch angles. (d) Yaw angles.



**FIGURE 12.** Step-detect and fuzzy filter threshold.

The MATLAB based GUI as shown in Figure 14 consists of windows of original GPS and IMU values path. There is also a corrected or orientationally fixed IMU values path. There is one bigger window called GPS/IMU Integration, which

shows how the missing GPS values are fixed by the values taken from IMU. This GUI is good intuitive tool to see the whole process of GPS and IMU integration.

## VIII. INTEGRATION

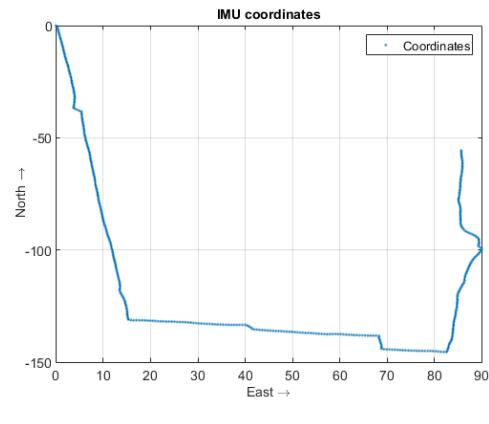
The algorithm for integrated using a MATLAB-based GUI (Figure 14) is given in Algorithm 1. Lines 3-6 obtain individual segments of the path. Segments are separated by  $> 35^\circ$  planar rotations of the IMU device, as this may cause non-trivial error accumulation due to some gyroscope drift. The segmentation resets the gyroscope and accelerometer profiles to pre-calibrated settings. In the absence of GPS signal due to loss of Line-of-Sight, the IMU continues measurements. We have noted segmentation can sharply reduce error buildup. Note Figures 15a and 15b. Segmentations are performed at the 'corners'. Consequently, error buildup is minimized and IMU orientations are not 'warped' due to drift error in the gyroscope. Line 7 orients IMU segments to GPS segments. In the case of significant missing content (Figure 16), segmentation and reorientation using the minimal

**Algorithm 1** GPS/IMU Integration

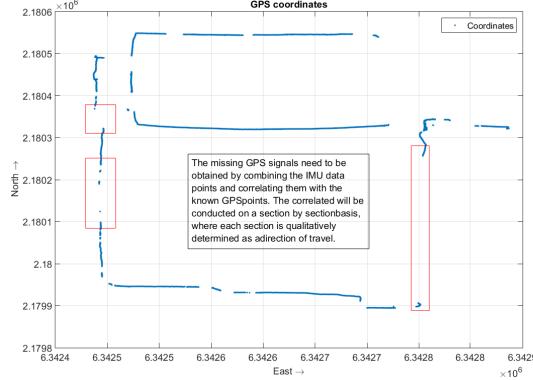
```

1: procedure GPS/IMU–Integration
2:   for each segment  $i \in PATH$  do
3:     GET GPS initial point, if available
4:     GET IMU segment range
5:     GET GPS heading (GPS template)
6:     GET IMU heading (IMU template)
7:     Orient IMU:  $[IMU_{new}] = [Rot] * [IMU_{old}]$ 
8:     Correct IMU magnitude (estimate if scale error)
9:     Translate corrected data points
10:    end for
11: end procedure

```



(a) IMU Data Collected



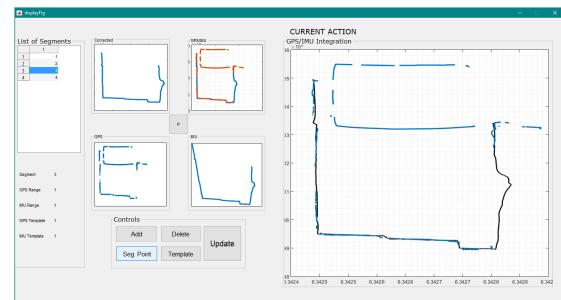
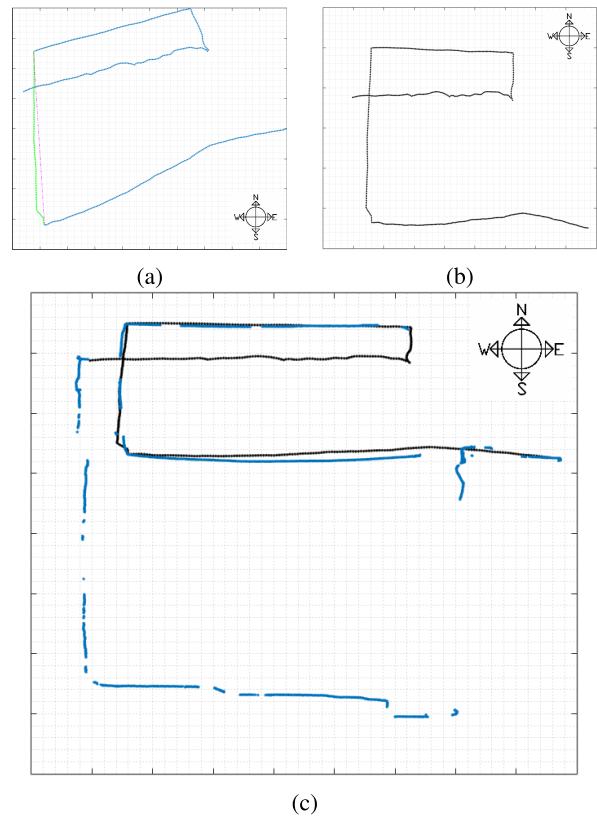
(b) GPS Data (reproduced from Figure 8)

**FIGURE 13.** IMU and corresponding GPS data - note the deviations from correct data. (a) IMU Data Collected. (b) GPS Data (reproduced from Figure 8).

GPS coordinates is sufficient to yield a complete map. Line 8 scales IMU measurements to coincide with GPS measurements. Once again, segmentation at larger turns preserves corners to be used in accurate magnitude estimation for scaling. It is a trivial matter to translate the IMU coordinates to the appropriate GPS locations by coinciding initial point(s).

**A. ORIENTATION AND MAGNITUDE CORRECTION**

For orientation correction, a 2D rotation matrix is obtained using the difference angle between the IMU and GPS

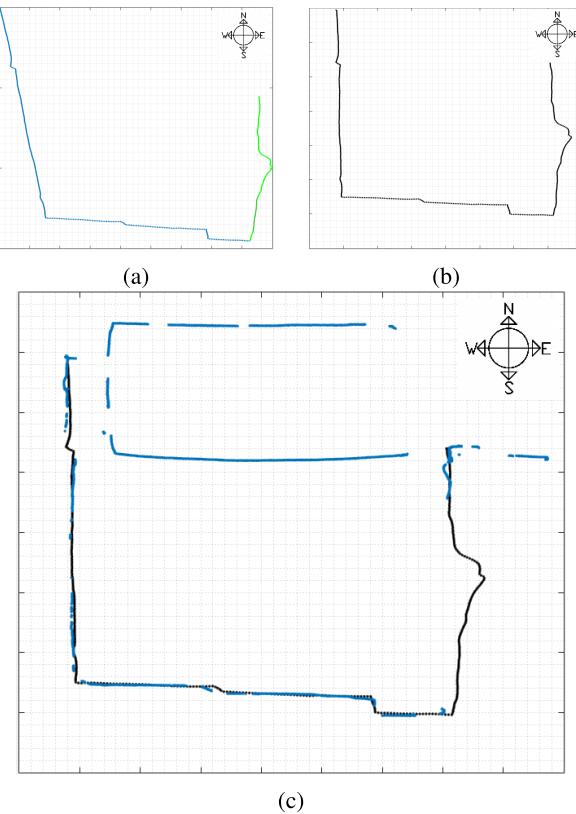
**FIGURE 14.** MATLAB GUI for GPS/IMU Integration.**FIGURE 15.** IMU Part I - integrated with GPS. (a) IMU Data. (b) Corrected IMU data. (c) Integrated data.

headings and a corrected IMU heading is calculated:

$$\mathbf{R}_{IMU} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (16)$$

A template direction is created at the beginning of each segment in the IMU and GPS data to obtain headings. The rotation matrix represents the difference in planar rotation for the IMU template heading and GPS template heading. The matrix is applied to the IMU segment to correct heading towards any GPS measurements. This also corrects any region in the segment without GPS counterpart as the entire segment is corrected from a smaller template region.

The IMU data's magnitude is corrected using the current GPS segment endpoints. The IMU endpoints are scaled to



**FIGURE 16.** IMU Part II - integrated with GPS. (a) IMU Data. (b) Corrected IMU data. (c) Integrated data.

coincide with GPS segment endpoints. If a segment's initial or endpoints are missing in the GPS data, the scale is estimated considering other IMU segments' correlation with the GPS data to ensure the current segment is placed appropriately in context with other segments (see Figure 16).

## B. INTEGRATION RESULTS

The path shown in Figure 8 is analyzed and the missing GPS data points are obtained. The IMU data was separated into two parts (Figures 15, 16). The results show that the IMU data can fill in the gaps in the missing GPS data collected by the hand-held receiver. In particular, the IMU gives the precise data points within the building in Figure 15, where the GPS data is non-existent and approximate path cannot be estimated using interpolation. The results obtained in Figures 15 and 16 can be combined to yield the complete data for the path taken. These results show that the IMU data can occupy the gaps in the missing GPS data collected by the hand-held receiver.

## IX. CHALLENGES AND FUTURE WORK

Our research team has noticed that the accelerometer in the smart phone is not useful in slow moving applications as it cannot detect the small accelerations. For more accurate pedestrian or device tracking, it is necessary to use off-the-shelf low-cost IMUs that nevertheless have more accuracy than smart phone accelerometers. Further, a SLAM approach

to position tracking may be useful to precisely locate certain landmarks during travel and ensure higher accuracy for GPS/IMU integration. These issues are to be investigated and to be addressed in future publications.

## X. CONCLUSION

This research presents an GPS/IMU integration for better positioning when the GPS signals are not available. An orientation detection and subsequent displacement detection algorithm for use in GPS/IMU integration for smaller devices in context of IoT applications or pedestrian tracking is also proposed. The orientation detection was accomplished by using IMU sensors viz. accelerometer, gyroscope, and magnetometer to determine device orientation relative to a fixed Earth geodetic frame.

It has been found that accelerometer in smart phones are not suitable for slow moving applications because it can't detect small variations. The results of this research show that a lowcost GPS/IMU integration for dead-reckoning in access-denied areas for device or pedestrian tracking is possible.

## ACKNOWLEDGMENT

The authors would like to thank Penn State University, for providing the materials necessary for this research.

## REFERENCES

- [1] L. Ojeda and J. Borenstein, "Personal dead-reckoning system for gps-denied environments," in *Proc. IEEE Int. Workshop Safety, Secur. Rescue Robot. (SSRR)*, Sep. 2007, pp. 1–6.
- [2] R. M. Rogers, "In-flight line-of-sight sensor boresighting using gps receiver data," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON)*, May 1994, pp. 197–203.
- [3] Y. Wang, J. Mangnus, D. Kostic, H. Nijmeijer, and S. T. H. Jansen, "Vehicle state estimation using gps/imu integration," in *Proc. IEEE SENSORS*, Oct. 2011, pp. 1815–1818.
- [4] Y. Jin, H.-S. Toh, W.-S. Soh, and W.-C. Wong, "A robust dead-reckoning pedestrian tracking system with low cost sensors," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2011, pp. 222–230.
- [5] D. E. Dilger. (2010) *Inside iPhone 4: Gyro Spins Apple Ahead in Gaming*. [Online]. Available: [http://appleinsider.com/articles/10/06/16/inside\\_iphone\\_4\\_gyro\\_spins\\_apple\\_ahead\\_in\\_gaming/page/2/](http://appleinsider.com/articles/10/06/16/inside_iphone_4_gyro_spins_apple_ahead_in_gaming/page/2/)
- [6] M. Ilyas, Y. Yang, Q. S. Qian, and R. Zhang, "Low-cost IMU/odometer/GPS integrated navigation aided with two antennae heading measurement for land vehicle application," in *Proc. 25th Chin. Control Decision Conf. (CCDC)*, May 2013, pp. 4521–4526.
- [7] S. Madakam, R. Ramaswamy, and S. Tripathi, "Internet of things (IoT): A literature review," *J. Comput. Commun.*, vol. 3, no. 5, pp. 164–173, May 2015.
- [8] S.-S. Wu and H.-Y. Wu, "The design of an intelligent pedometer using android," in *Proc. 2nd Int. Conf. Innov. Bio-Inspired Comput. Appl. (IBICA)*, Dec. 2011, pp. 313–315.
- [9] X. Jiang, D. Li, P. Xiao, S. Liu, and Z. Lu, "Lessons of IOT effects on backbone networks learnt from traffic characteristics," in *Proc. 8th Int. Conf. Wireless Commun., Netw. Mobile Comput. (WiCOM)*, Sep. 2012, pp. 1–4.
- [10] H.-W. Wang, C.-J. Qi, Q.-X. Wu, X.-N. Wang, and Z.-H. Xing, "Algorithm of roll, pitch and yaw angular velocity for three-axis gyroscope," in *Proc. Int. Conf. Environ. Sci. Inf. Appl. Technol. (ESIAT)*, vol. 2, Jul. 2010, pp. 395–398.
- [11] *Triaxial Accelerometer Sensor (SMB380)*, Bosch Sensortec GmbH, Kusterdingen, Germany, 2007.
- [12] W. T. Higgins, "A comparison of complementary and Kalman filtering," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 11, no. 3, pp. 321–325, May 1975.
- [13] *MEMS Gyroscope (L2G2IS)*, STMicroelectronics, Geneva, Switzerland, 2015.

- [14] Y. Chai, W. Feng, C. Jiang, and K. Liu, "Design and realization of digital compass with magnetometers," in *Proc. 7th World Congr. Intell. Control Autom. (WCICA)*, Jun. 2008, pp. 7672–7675.
- [15] Y. Shoji, T. Takasuka, and H. Yasukawa, "Personal identification using footstep detection," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst. (ISPACS)*, Nov. 2004, pp. 43–47.



**ABHIJIT SUPREM** received the B.S. degree (Hons.) in electrical engineering and minors in computer engineering and mathematics from California State University at Fresno, Fresno, USA. He is currently pursuing the master's degree in computer science with Columbia University. His recent research work is on sensor fusion for GPS and IMU-based systems. He has received the Best Student Paper Award at the World Congress of Engineering and Computer Sciences in 2013.



Medal 2016.

**VISHAL DEEP** received the bachelor's degree in electronics and communication and the master's degree in microelectronics and VLSI design from Kurukshetra University, India. He is currently pursuing the master's degree in computer engineering with California State University at Fresno, Fresno, USA. His research interests include image and video processing, VLSI design, and embedded systems. He received the Outstanding Graduate Student Award and nominated for Graduate Dean's



VLSI design, and embedded systems.

**TAREK ELARABI** received the Ph.D. degree in computer engineering from the Center for Advanced Computer Studies, University of Louisiana at Lafayette. He is currently a Faculty Member of Computer Engineering with the School of Engineering, Penn State Behrend. His expertise is in digital image and video processing with particular interest in video compression algorithms, heterogeneous video transcoding for real-time video applications and Internet of Things devices,

• • •