Analysis of raw GNSS measurements derived navigation solutions from mobile devices with inertial sensors

Dong-Kyeong Lee, Matthias Petit, Damian Miralles, Sherman Lo, Dennis Akos

Analysis of raw GNSS measurements derived navigation solutions from mobile devices with inertial sensors

Dong-Kyeong Lee, *University of Colorado Boulder*
Matthias Petit, *École Nationale de l'Aviation Civile*
Damian Miralles, *University of Colorado Boulder*
Sherman Lo, *Stanford University*
Dennis Akos, *University of Colorado Boulder, Stanford University*

BIOGRAPHY (IES)

Dong-Kyeong Lee is a PhD student at the University of Colorado Boulder. He received is BSc and MSc at Korea Advanced Institute of Science and Technology. His interests lie in GNSS receivers, UAVs, and GNSS spoofing.

Matthias Petit is an engineering student at ENAC, the French Civil Aviation University in Toulouse, France. He is majoring in Aerospace Telecommunications and he came to the University of Colorado Boulder to participate in this effort as he intends to specialize further into GNSS during his next school year.

Damian Miralles is a graduate student in the Department of Aerospace Engineering Sciences at the University of Colorado Boulder. He received a B.S. in Electrical and Computer Engineering from the Polytechnic University of Puerto Rico. His research interests are in GNSS receiver technologies, SDR and digital signal processing.

Sherman Lo is a senior research engineer at the Stanford GPS Laboratory. He received his Ph.D. in Aeronautics and Astronautics from Stanford University in 2002. He has and continues to work on navigation robustness and safety, often supporting the FAA. He has conducted research on Loran, alternative navigation, SBAS, ARAIM, GNSS for railways and automobile. He also works on spoof and interference mitigation for navigation. He has published over 100 research papers and articles. He was awarded the ION Early Achievement Award.

Dennis M. Akos completed the Ph.D. degree in Electrical Engineering at Ohio University within the Avionics Engineering Center. He has since served as a faculty member with Luleå Technical University, Sweden, and then as a researcher with the GPS Laboratory at Stanford University. Currently, he is a faculty member with the Aerospace Engineering Sciences Department at the University of Colorado, Boulder and maintains visiting appointments at Stanford University and affiliation with Luleå Technical University.

ABSTRACT

Since the introduction of raw Global Navigation Satellite Systems (GNSS) measurements for Android devices in 2016, the applicability of GNSS chips on Android has increased significantly. Prior to the raw measurements availability, the device users were only able to access GNSS National Marine Electronics Association (NMEA) data which included limited rudimentary GNSS related information such as position, carrier-to-noise-density ratio (C/No), speed over ground, and heading. The introduction of raw measurements has allowed the users to directly compute additional parameters such as three-dimensional velocity, which would allow computation of higher accuracy GNSS navigation solutions including altitude rate, heading velocity, and acceleration, compared to NMEA-based solutions. One significant application for higher detail GNSS information is GNSS spoofing detection. GNSS spoofing refers to the inaccurate computation of GNSS navigation solutions due to intentional or unintentional artificial false satellite signals. There are already several methods to detect spoofing, such as the analysis of C/No and automatic gain control (AGC), or navigation solution anomaly detection using inertial sensors. The method that will be explored in this paper will be a navigation domain analysis, but instead of using NMEA messages, raw GNSS measurements will be used to derive the navigation solutions. Using the raw carrier phase measurements, three-dimensional velocity, acceleration, and heading will be derived, and compared to onboard inertial navigation sensors such as magnetometer, accelerometer, and barometer, which are immune to GNSS spoofing due to their measurements being unaffected by GNSS measurements. If the comparison metrics exceed a predesignated threshold, potential spoofing alert is triggered. The novelty of this study lies in looking into the feasibility of using a smartphone to detect potential GNSS spoofing using raw GNSS measurements-derived navigation solutions, and onboard inertial sensors. In order to trigger potential spoofing, the noise of the inertial measurements from both the GNSS chipset, and the inertial sensors will be assessed under various dynamic environments, in order to derive dynamics-dependent anomaly detection threshold. Afterwards, the statistical validity of the thresholds will be assessed using two drive tests: one to determine the proposed thresholds, and one to test the effectiveness of those proposed thresholds.

1. INTRODUCTION

1.1 GNSS receivers on Android devices

In the recent decade, there have been several additions to the Android devices concerning GNSS functionality. Prior to 2016, the only major source of GNSS-related information for Android device users was the NMEA message coming from the GNSS Application-specific integrated circuit (ASIC) chips provided by the GNSS chipset manufacturers. The NMEA message was able to provide the user with time, location, speed over ground, and heading. Although the available measurements were of high accuracy due to the chip manufacturers competing to provide high quality navigation measurements to mobile phone producers, the users had limited accuracy three-dimensional inertial measurements due to the navigation solutions except the horizontal speed being in the position domain. This limitation was due to possible factors including the chip development companies, operating systems (OS), or original equipment manufacturer (OEM) not providing the information.

In 2016, Android 7 introduced raw GNSS measurements for Android devices, and this availability stimulated numerous research opportunities concerning mobile devices. Although not all devices provided the entire suite of information, the raw measurements were capable of providing additional GNSS information such as the code and carrier phase measurements, doppler, and C/No. These measurements not only provided more information about the GNSS signals tracked by the Android devices, but also allowed the computation of other navigation outputs such as three-dimensional velocity. Following the preliminary availability of raw GNSS measurements, in 2017, Android 8 introduced AGC measurements in its suite of raw GNSS measurements. This further increased the utility of the applications of raw GNSS measurements, such as GNSS spoofing detection using a combination of C/No and AGC [1]. Furthermore in 2018, Android 9 introduced the option to turn off duty cycling for the GNSS chipsets. Duty cycling is the act of turning off GNSS chipsets momentarily to save device power. By disabling this, the receiver is not forced to go through temporary shutdowns, increasing the quality of raw GNSS measurements, especially the carrier phase measurements [2].

These advancements in Android devices have allowed the users to access raw GNSS measurements from GNSS ASIC chipsets far beyond the information available in NMEA messages. These raw measurements allow for better understanding of the characteristics and status of the individually tracked GNSS satellites, higher accuracy computable three-dimensional velocities, and greater potential in GNSS spoofing detection.

1.2 Susceptibility of Android to GNSS spoofing

GNSS spoofing is the act of providing artificial false GNSS signals to the GNSS receivers. Under nominal conditions, the GNSS receivers inside the Android devices receive true signals from GNSS satellites, and convert the signals into navigation solutions, which are in turn passed on to mobile applications. However, under spoofing scenarios, false signals are picked up by the receivers inside the mobile devices, which results in the users acquiring incorrect navigation solutions. These spoofing events can be user intentional, or unintentional, and come in the form of both software and hardware interferences.

For the software case, the user directly provides a false position to the OS, bypassing the GNSS ASIC completely. On the other hand, for the hardware case, the user creates a hardware spoofer consisting of a customized software-defined radio (SDR) and a transmitter to fool the GNSS ASIC with false GNSS signals. In the case of a transmitter setup, according to Goodin [3], the cost can be as low as $225, and the size can be as small as a pen for the length and width dimensions.

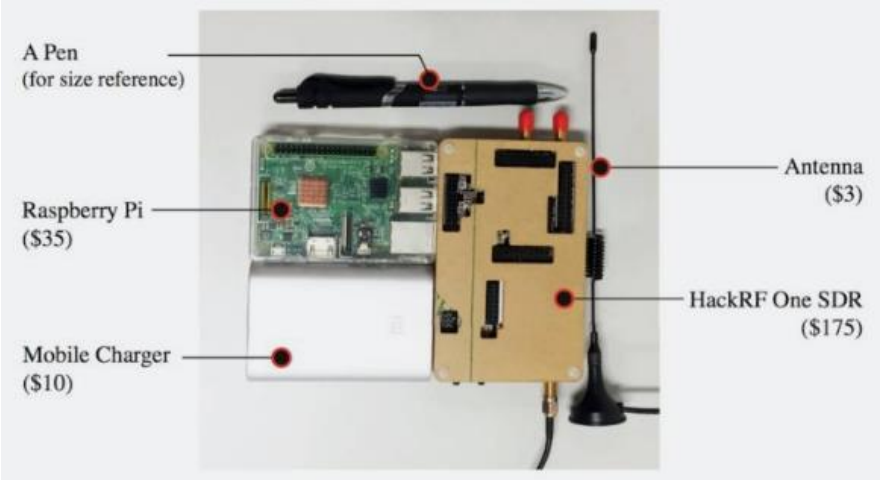

Figure 1. Hardware configuration and cost of a potential GNSS spoofer [3].

An example of software spoofing is for misleading the application Pokémon Go. By simply installing additional applications to the device, the user is able to fool the Pokemon Go application into thinking that the device is physically moving. For the hardware spoofing scenario, there have been several cases where individuals demonstrated the possibility of spoofing Android mobile devices. For example, in Blackhat Europe, Wang [4] was successful in fooling the Android device into thinking that it was located in Japan, when it was actually located in China.

Figure 2. Demonstration of GNSS spoofing for Android devices. Original location was in China, but the spoofed location reported by the Android device is in Japan [4].

Although this was a demonstration of a concept, under wrong hands and with malicious intent, the setup can become a troublesome GNSS spoofer for the unsuspecting public. Furthermore, although unintentional, a potential spoofing setup can cause trouble to the public, as was the in ION GNSS+ 2017. On September 28 2017, a GNSS simulator was leaking false signals as the unused ports were improperly sealed with plastic caps [5]. Due to this mistake, attendees in the conference area reported GNSS spoofing on their mobile devices.

Due to increasing events of spoofing, it is important for the users to detect potential GNSS spoofing for the safety and integrity of GNSS reliant systems. There are already several solutions to mitigate or detect these problematic GNSS spoofing. However, most solutions require expensive equipment and software, which are not readily available on most smartphones. Most smartphone devices do not have the luxury of having high quality spoofing detection algorithms, and are limited to low quality antennae which are not designed for spoofing mitigation nor detection. Therefore, in order to overcome this issue, an inertial sensor-based GNSS spoofing detection methodology for Android devices is proposed and explored in this study.

1.3 GNSS-independent sensor-based spoofing detection methodology

The core idea of this methodology was initially proposed by Lo [6], [7] in his paper on the use of GNSS-independent sensors to detect potential spoofing in GNSS receivers. There have also been some additional works such as the study by Borio [8], where he attempts to compare the magnitude and the horizontal components of the accelerometer and magnetometer to the GNSS-derived measurements. For this paper, we look at not only inertial sensors, but also other GNSS-independent sensors such as the magnetometer and barometer, but for the simplification of terminology, we will refer to all these GNSS-independent sensors as inertial sensors. The methodology compares the navigation solutions obtained from raw GNSS measurements to the measurements provided by inertial sensors to detect potential spoofing. Under nominal conditions, the navigation solutions from the sensors would only have discrepancies due to sensor inaccuracies. However, under spoofing scenarios, the measurements would deviate from each other, as GNSS receivers are affected by spoofing, while the inertial sensors are independent of GNSS measurements. Therefore, if the measurements deviate above the thresholds established by the sensors' inaccuracies, potential spoofing is flagged.

The advantages of using inertial sensors for GNSS spoof detection include robustness, accuracy, integrity, and low implementation costs. They are robust as their measurements are independent from GNSS signals. Also, for short time windows, the effects of drift are not significant, and the sensor measurement accuracies and integrity are high. In addition, the cost for implementation is low, because the individual inertial sensors are cheaper than expensive spoofing detection suites, and several inertial sensors are already embedded inside mobile devices. For the study, three inertial sensors and their capability in GNSS spoofing detection are observed: magnetometer, accelerometer, and barometer. In the case of a magnetometer and an accelerometer, under nominal conditions, they are able to accurately determine the dynamic motions experienced by the device with respect to global East North Up (ENU) coordinates. In addition to the two sensors, we also consider barometer, as it is considered to be an accurate altitude sensor for many airborne systems [9]. Other sensors that are not considered, but are future candidates include camera as a vision sensor, and gyroscope.

For the GNSS-derived navigation solutions, raw GNSS measurements are used instead of NMEA messages. This is because the NMEA messages provide position, speed, and heading, while the raw GNSS measurements allow us to directly compute three-dimensional velocities. The advantage in using the velocities directly is that in general, the carrier-derived velocities have higher accuracy (9.7mm/s or less) than GNSS positions (15m or less), and velocities are closer to the domain of comparison with respect to the inertial sensor measurements, which will be acceleration, heading rate, and altitude rate [10], [11]. The advantage of using higher accuracy GNSS-based measurements is that the thresholds established by the sensor inaccuracies will be tighter. If we can visually look at this phenomenon in Figure *3*. Let us assume that the measurements due to NMEA, carrier, and spoofing have normal distribution random noise. The thresholds are set for the NMEA and carrier measurements as the statistical three sigma values. When the statistics exceed the threshold, we flag potential spoofing. From the Figure *3*, we can see that the higher accuracy carrier threshold has lower spoofing misdetection probability than the NMEA threshold. Therefore, tighter thresholds allow us to decrease false detection and misdetection probability, and also detect more subtle spoofing occurrences where the navigation solution deviations are smaller.

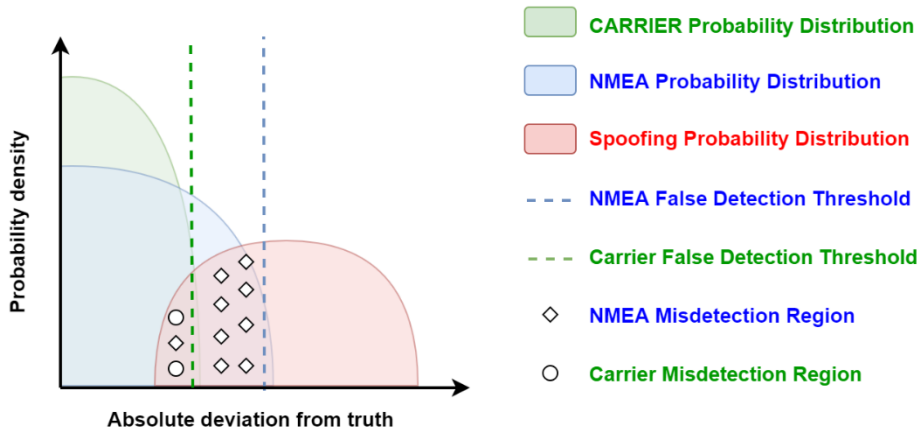

Figure 3. Spoofing false detection and misdetection probability statistical representation.

2. METHODOLOGY

2.1 Proposal

The inertial sensor-based GNSS spoofing detection methodology for Android devices proposed in this paper compares the raw GNSS measurements derived navigation solutions to the smartphone inertial sensor measurements. If the measurement discrepancies exceed the predetermined thresholds established by sensor inaccuracies, potential spoofing is flagged. The three inertial sensors that will be explored include magnetometer, accelerometer, and barometer. It is important to note that these sensors are not available in all mobile devices, but we investigate all of them to look into their potential in GNSS spoofing detection. In the following sections, we explore how these sensors can be used obtain heading rate, acceleration, and altitude rate for navigation solution anomaly detection, and also introduce a method to establish a suitable threshold for the proposed methodology. The overall outline of the spoofing detection methodology is shown in Figure *4*.

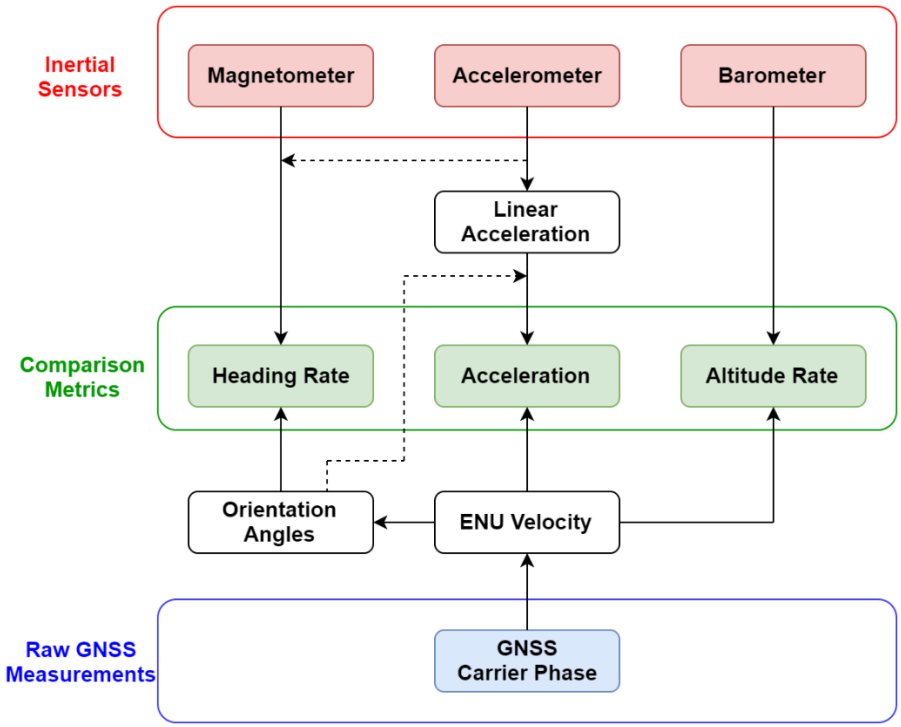

Figure 4. Flowchart overview of the proposed spoofing detection methodology. Inertial sensor measurements and raw GNSS measurements are processed to produce comparison metrics.

2.2 Magnetometer Inertial Measurement

The comparison metric we would like to obtain using the magnetometer and accelerometer is the heading rate. First of all, we need to derive the heading of the mobile device, and the technique outlined below is a common method employed by most Android devices to determine heading as outlined in the Android opensource codebase [12]. First, we use the magnetic field measurements from the magnetometer, $H = \left[H_x, H_y, H_z\right]^T$, and the body frame acceleration from the accelerometer, $G = \left[G_x, G_y, G_z\right]^T$, to determine the heading. This is similar to the heading derivation approach using the gravity and magnetic field vectors with the Wahba method outlined by Gebre [13], but we do not take into account the hard and soft iron vectors, and scale factor and misalignment errors. Let a us define rotation matrices and orientation angles as:

$$R = R_x(\phi) \cdot R_y(\theta) \cdot R_z(\psi) \tag{1}$$

where R is the rotation matrix of the mobile device from ENU frame to the local body frame, R_x, R_y, R_z are the single rotation matrices through the x, y, z axis, and ϕ, θ, ψ are the rotation angles roll, pitch, yaw respectively. It is important to note that the yaw angle here corresponds to the heading angle. Next, we assume that the dominant source of acceleration is gravity, and that the linear acceleration is negligible. Therefore, as shown in Equation 2, measured acceleration is gravity vector rotated through the rotation matrix *R*:

$$G = R \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{2}$$

where *g* is the acceleration due to gravity. For the magnetometer, as in Equation 3, we assume that the measured magnetic field is the rotation of the magnetic field strength vector:

$$H = R \cdot B \cdot \begin{bmatrix} cos(\delta) \\ 0 \\ sin(\delta) \end{bmatrix} \tag{3}$$

where *B* is the geomagnetic field strength, and δ is the angle of inclination of the geomagnetic field. By undoing the x and y frame rotation matrices for Equation 2, we obtain the expression shown in Equation 4:

$$R_y(-\theta) \cdot R_x(-\phi) \cdot G = R_z(\psi) \cdot \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{4}$$

By solving the equation, we can estimate the roll of the device as shown in Equation 5:

$$\phi = atan\left(G_y / G_z\right) \tag{5}$$

Using the obtained roll, we can compute the pitch of the device as shown in Equation 6:

$$\theta = atan\left(\frac{-G_x}{-G_y \cdot sin\phi + G_z \cdot cos\phi}\right) \tag{6}$$

Finally, using both the obtained roll and pitch, we can determine the yaw of the device as shown in Equation 7:

$$\psi = atan\left(\frac{H_z sin\phi - H_y cos\phi}{H_x cos\theta + H_y sin\theta sin\phi + H_z sin\theta cos\phi}\right) \tag{7}$$

However, there is one limitation to the heading angle. The mobile devices are prone to heading biases. This may be due to the sensor calibration level, accuracy, incorrect bias compensations, scale factor and misalignment errors, or unaccounted soft and hard iron errors due to the layout of the electronics inside the mobile devices, or nearby electromagnetic sources such as cars and machinery. For example, from Figure *5*, we can see that while the true heading is 0 degrees north, the Huawei

P10 calculates a heading that is approximately 21 degrees off, while Xiaomi Mi8 is approximately 103 degrees off. Although in Figure *5*, the mobile devices are in close proximity of each other, even at distances over 1m apart, the measurement offsets were consistent.

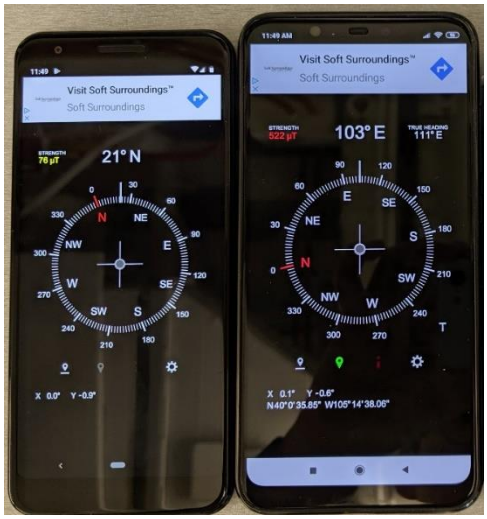

Figure 5. Huawei P10 (Left), Xiaomi Mi8 (Right) compass headings. The true heading is supposed to be 0 degrees north, but both phones have differing levels of heading biases.

Therefore, in order to overcome some of these errors, we take the time differential of the heading measurements to attain heading rate for the comparison metric. By looking at the heading rate, we remove the effects of the bias, and consider the sensor noise and temporal rate of change of measurements.

2.3 Accelerometer Inertial Measurement

The comparison metrics we would like to obtain using the accelerometer are the ENU accelerations, and their magnitude. This requires several steps, because as shown in Figure *6*, the measurements from the accelerometer are in the body frame, and the effects of gravity are present. Gravity is present due to the reason that the accelerometers measure force rather than the acceleration itself. Consequently, the sensor measures gravity and centrifugal force which are not measured by GNSS acceleration.

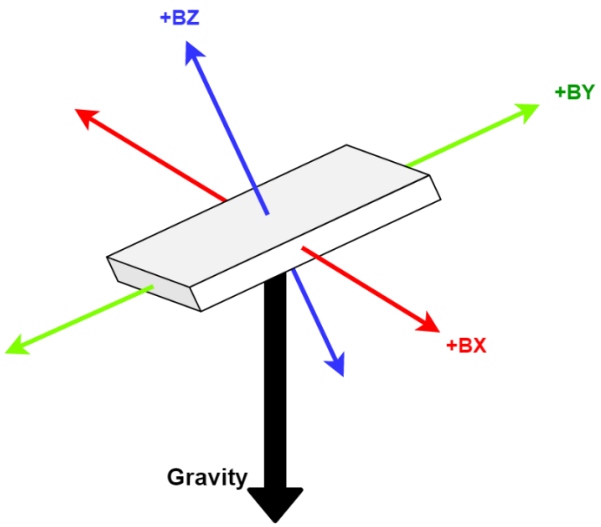

Figure 6. Acceleration from accelerometer has measurements in XYZ local body frame (BX, BY, BZ), and effects of gravity.

First, we remove the effects of gravity, as we would like to compare the linear acceleration from the accelerometer to the GNSS-derived measurements. Unlike the accelerometer, the GNSS receivers compute the acceleration by measuring the change in motion of the receiver. Therefore, gravity should not be present in the accelerometer comparison, and only the change in dynamics should be considered. There are several techniques available to remove the effects of gravity, but in our paper, we use a technique employed by Curran, where the second order Butterworth high-pass filter is used to remove the effects of gravity [14]. In essence, the high-pass filter assumes gravity to be a slowly changing term in acceleration and removes it. It is important to note that this technique has limitations, as the only residual measurements after this filter are high frequency accelerations. For a person or vehicle that may be steadily accelerating, the filter would effectively remove these dynamics. Therefore, further studies should be done on other techniques of gravity removal, such as orientation angle-based gravity removal. For the cutoff frequency, although Curran [14] proposed a preliminary value of 0.01Hz, in our case, we used a cutoff frequency of 0.12Hz as it seemed to be more effective in removing gravity. More explanation on how this frequency was determined is provided in the results section of the paper.

Secondly, we need to do a coordinate transformation from the local body frame to the ENU coordinate frame using orientation angles as shown in Figure 7. There are several possible sources of these orientation angles which include: magnetometer and accelerometer combination, GNSS NMEA messages, and raw GNSS carrier measurements. There are pros and cons to the different approaches. For the inertial sensors, at low dynamics, the accuracies of the angles are higher. This is because the Wahba method assumes the gravity and magnetometer vectors to be low noise and non colinear. On the other hand, for high device dynamics, the dynamics can potentially corrupt the gravity vector measurements. Also, there are latent issues with the sensors having biases as explained in the above section.

In the case of GNSS NMEA messages, although it is possible to determine the orientation angles by taking the time differential of the positions, the accuracy of the positions is a limiting factor for the accuracy of the angles. Furthermore, although we can use the heading angles from the NMEA messages, the angles are for the horizontal plane, which will be problematic for non-horizontal motion scenarios. For the raw GNSS carrier measurements, they have high accuracy orientation angles, but at low velocities, the angles are significantly affected by GNSS receiver noises. For example, if the receiver is stationary, the velocity noise is considered to be motion, and incorrect headings may be computed. In addition, one large assumption we are making is that the heading of the mobile device is in the direction of device motion. This assumption is valid for our experiment, because we fix the mobile phone to the car for testing, and the car does not slip while in movement. However, if there is slip, or if the person is moving in the direction not aligned with heading, there can be non-alignment issues even if the GNSS measurements and inertial sensors are error free.

In our case, we use the orientation angles from the raw GNSS carrier measurements, because although some dynamics are required for accurate measurements, when there are significant levels of motion, the angle accuracies are the highest for this source. How we determine these angles will be explained later in the raw GNSS measurements processing portion of the paper.

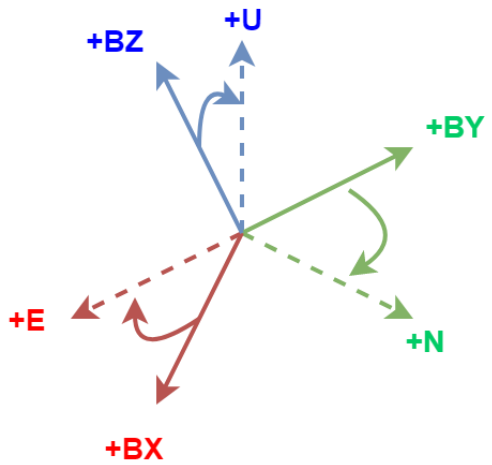

Figure 7. Illustration of coordinate transformation from local body frame XYZ to ENU.

After the gravity removal and coordinate transformation, we average the acceleration measurements over a period of 1 second. This is because the rate of GNSS measurements is 1Hz, while the rate of accelerometer is approximately 10Hz. The reason why we take the average instead of using other methods such as exponential averaging is because we wish to compare the inertial-derived metrics under the same conditions as the GNSS-derived metrics. GNSS measurements determine the acceleration by looking at how the carrier measurements have changed over a period of 1 second. Therefore, we would like to evenly average the acceleration measurements over the same time window to observe the total acceleration changes in the equivalent time frame. It is important that we do the averaging after the coordinate transformation for the ENU acceleration, because the frame orientation may not be consistent for the 1 second period.

For the comparison metric, we compare both the acceleration magnitude and the ENU frame axial accelerations. The advantage of comparing the magnitude is that we do not have errors introduced by the orientation angle-based rotation, which makes the metrics have higher accuracy. However, despite the accuracy tradeoff, the benefit of using individual ENU direction accelerations is that we can detect anomalies in any of the ENU axis even if the total acceleration magnitude is constant.

2.4 Barometer Inertial Measurement

The comparison metric we would like to obtain from the barometer is the altitude rate. The output of the sensor is initially atmospheric pressure. We can convert this pressure into an estimated altitude using the relationship shown in Equation 8.

$$h = h_b + \frac{T_b}{L_b}\left[\left(\frac{P}{P_b}\right)^{\frac{-RL_b}{g_0 M}} - 1\right] \quad (8)$$

where P_b is the static pressure, T_b is the standard pressure, L_b is the standard temperature lapse rate (-0.0065K/m), h is the height above sea level, h_b is the height at the bottom of the atmospheric layer, R is the universal gas constant (8.31432J/(molK)), g_0 is the gravitational acceleration constant (9.80665m/s^2), and M is the molar mass of Earth's air (0.0289644kg/mol).

Although this is a great way to estimate the altitude of the device, there are a lot of assumptions and errors that contribute to biases and inaccuracies in the altitude. For example, the height at the bottom of the atmospheric layer may lead to a bias, and other parameters will lead to scaling issues. This issue is visualized in the results section, where we show the altitude estimations from the mobile device barometer and NMEA messages, and NovAtel GNSS receiver. However, we notice that the trend for the change of altitude is almost identical between the sources despite the distinct biases in the absolute altitude measurements. Therefore, in order to remove these issues due to biases, we look at the altitude rate for the comparison metrics.

Also, although not considered for this study, we can use the pressure corrections from local hourly aeronautical code Q code (QNH) data to increase the accuracy of the barometer. QNH is the atmospheric pressure adjusted to the mean sea level, and has high local accuracy irrespective of surrounding atmospheric temperature.

2.5 Raw GNSS Measurements into Inertial Domain

In this section, we will explain the method to derive all the aforementioned comparison metrics from the raw GNSS measurements. For the derivation of the three classes of measurements (heading rate, acceleration, and altitude), we need the GNSS code and carrier measurements. The code measurements can be used to determine the user position and coordinate transformation, and the carrier measurements can be used to determine the relative velocities of the GNSS receiver with respect to its latest position solution.

Using the code and carrier measurements and the least square methods, we found the positions and velocities of the receivers. Further details on the process can be found in works by Misra [11], and Van Graas [10]. We need to note that the navigation solutions we computed are in the earth centered earth fixed (ECEF) domain. However, we wish to compute all the measurements in the ENU frame, as we had converted all inertial measurements into the ENU frame. Therefore, we carry out a coordinate transformation from the ECEF to the ENU domain, and the difference in these coordinate frames can be seen in Figure 8. For the ECEF frame, the z-axis extends through the true north, and the x-axis intersects the sphere of the

earth at the equator and prime meridian in Greenwich. On the other hand, ENU frame is a local geographical coordinate system where the E points in the local eastern axis, and N points in the northern axis. We use the position solutions to determine the rotation matrices, and apply them to the velocities to obtain ENU velocity.

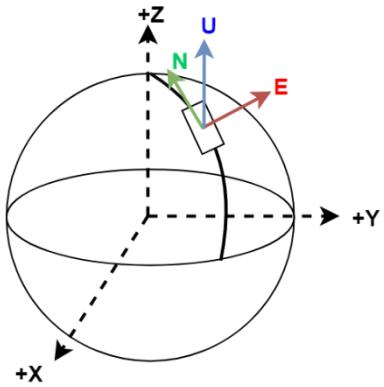

Figure 8. Illustration of the difference between ECEF and ENU coordinate frames.

From the ENU frame velocities, we can simply take the upwards component of the velocity to determine the altitude rate. For the heading rate, we take the angle between V_E and V_N, which are the north and east axis velocities respectively for the GNSS-derived heading, ψ_{GNSS}, and then take the time differential. To determine the other orientation angles, we take the arctangent of the other axial velocity vectors. For the acceleration, we simply take the time differential of the velocities.

$$\psi_{GNSS} = atan\left(\frac{V_E}{V_N}\right) \tag{9}$$

2.6 Decision Metrics

In order to quantify the discrepancies between the sensor measurements, we determined the decision metric between inertial-derived and GNSS-derived navigation solutions, $C(GNSS, INS)$, *by comparing* the mean values of the navigation solutions for a fixed time window as shown in Equation 10:

$$C(GNSS, INS) = |\langle GNSS\rangle_T - \langle INS\rangle_T| \tag{10}$$

where $\langle\cdot\rangle_T$ represents the mean of the GNSS-derived solution for the time window of T seconds, and $|\cdot|$ represents the magnitude attained by taking the normalized value of the difference. In our case, we used a time window of 5 seconds. The advantage of using long time windows is that the noises of the measurements are reduced. However, the disadvantage of long windows is that if the spoofing duration is significantly lower than the duration of the window, the spoofing detectability will decrease as the effects of spoofing will be averaged out.

For the spoofing detection threshold, we decided to determine a dynamics dependent threshold instead of a static threshold. This is because at different levels of dynamics, the accuracies of the measurement metrics varied for each of the navigation solutions. For heading rate, at lower velocities, the GNSS-derived heading rate accuracy diminished due to the receiver noise affecting the velocities enough to introduce large errors in the heading. Consequently, in our study, we set a limit of 1m/s for the minimum velocity at which heading rate metric is computed. This is a reasonable criterion, as the definition of heading is the direction of motion, and we are setting a minimum threshold for the definition of motion. Furthermore, as the average person's walking pace is 1.4m/s as mentioned by Barton [15], the heading rate comparison is possible for anyone walking with the mobile device, or within a moving vehicle moving faster than a human walking pace. For acceleration, the decision metrics increased at higher accelerations. This is because the gravity removal methodology assumes that the only source of acceleration is gravity, and that linear acceleration is insignificant. This assumption falls apart when the linear acceleration is non-zero, and is not in the direction of gravity.

In order to determine the dynamics dependent thresholds, we need to quantify the sensor inaccuracies. In order to do so, we compare the accuracy of the inertial sensors and the GNSS-derived measurements under different dynamics with respect to a truth reference. In our case, we used a high accuracy NovAtel GNSS receiver to serve as the truth reference. The reason why we use the NovAtel receiver is due to its high accuracy and integrity compared to the smartphone sensors. Further discussion on this is provided in the devices configuration section below.

By carrying out the truth comparisons, we noted the sensor accuracies under different dynamics, and derived the expected dynamics dependent sensor accuracy. In order to do so, we created different bins for different levels of dynamics, and took the mean standard deviation of each of the sensors from the truth reference. Afterwards, we added the expected error variances of the inertial sensor and GNSS carrier derived measurements to derive the total expected sensor error standard deviation. For the threshold, we took three times the standard deviation to achieve a false detection threshold statistic of 0.027%.

3. TESTING ENVIRONMENT

3.1 Devices Configuration

The mobile devices considered for this study are Xiaomi Mi8, and Huawei P10. Mi8 has a GNSS chip, and all four of the sensors we would like to study: GNSS chip (Broadcom BCM47755), magnetometer (AK0991x series), accelerometer (ICm20690 Qualcomm), and barometer (BMP285 Bosch). On the other hand, P10 lacks a barometer but has a GNSS chip (Broadcom BCM4774), magnetometer (AKM09911), and accelerometer (ICm20690 Invensense). For the truth reference, NovAtel SPAN-CPT Single Enclosure GNSS/INS Receiver was used. The receiver has high accuracy GNSS antenna and receiver, and Inertial Measurement Unit (IMU) which consists of high-quality accelerometer and gyroscope. The measurements are post-processed using tightly coupled Precise Point Positioning (PPP) solution with inertial measurements and local GNSS reference station data from online databases, through the NovAtel postprocessing software Waypoint. The NovAtel instrument is used to determine the truth heading rate, acceleration, and altitude rate.

Figure 9. Devices used for this study: Xiaomi Mi8, Huawei P10, NovAtel SPAN-CPT.

Table *1* has some expected noise statistics of each sensor. For the NovAtel, the noise was reported from the statistics report created after running the collected SPAN-CPT data through the NovAtel Waypoint software using one of the scenarios covered in this paper. For the accelerometer, and barometer, the statistics were taken from the manufacturer. For the magnetometer, a standard smartphone ecompass heading accuracy for a magnetometer with $0.5\mu F$ accuracy was taken [16]. For the GNSS chipset, the velocity accuracy is the value reported by Van Grass [10], but we need to take into consideration that the accuracy of the smartphone device GNSS chipsets may not be as high due to low C/No, low cost patch antenna, and frequent possible cycle slips for the carrier measurements despite disabling duty cycling. We can see that the accuracy of the NovAtel is higher than all of the inertial sensors, and due to the expected degraded performance of the smartphone GNSS chipset, the velocity from NovAtel is expected to be of higher accuracy and integrity.

Table 1. Expected noise of NovAtel GNSS receiver and smartphone sensors.

	NovAtel			Accelerometer	Magnetometer	Barometer	GNSS Carrier
	Heading	**Acceleration**	**Velocity**	**Acceleration**	**Heading**	**Altitude**	**Velocity**
Accuracy	0.047 deg	1.05 mm/s^2	30 mm/s	2.39 mm/s	3 deg	1 m	9.7 mm/s

For the testing setup, the NovAtel GNSS antenna was mounted on top of the test drive vehicle, while the NovAtel IMU, and both the phones were placed inside the car near on the dashboard. For the processing of the NovAtel data, the data was processed with the reference point at the IMU instead of the antenna.

3.2 Testing Scenario

In this paper, two drive tests with different routes were used near the University of Colorado Boulder. The first route, which was a training set, was used to collect data to derive thresholds for various dynamics. The second route, which we define as the validation set, was used to validate the proposed thresholds. The reason why the two selected routes were different was to ensure that the proposed thresholds were valid in different environments as long as the same levels of dynamics were present.

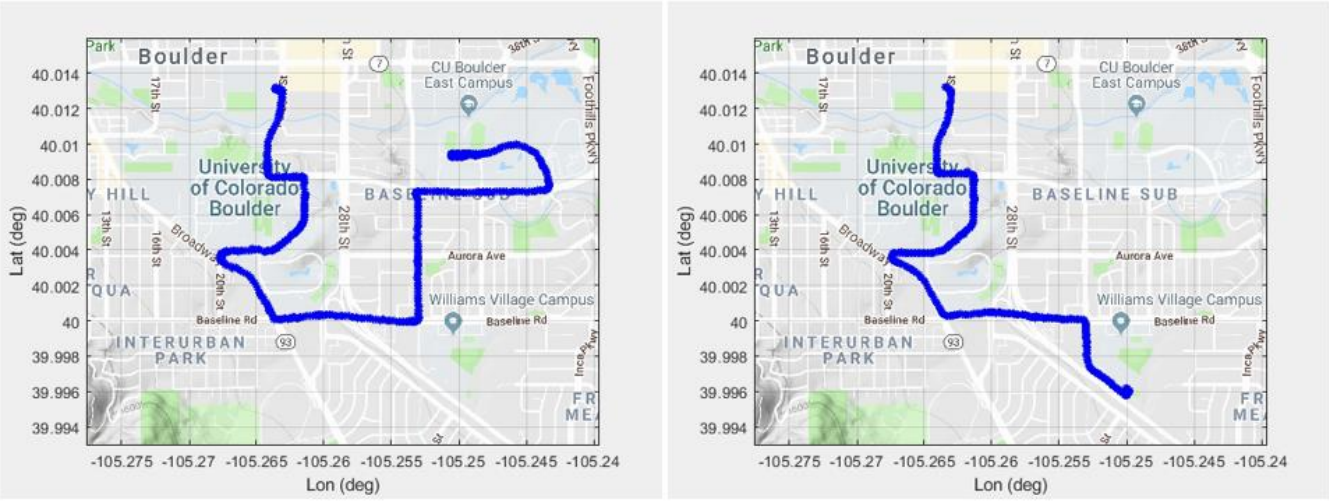

Figure 10. The two drive tests carried out for this experiment. The left trajectory was used as the training set, and the right trajectory was used as a validation set.

4. EXPERIMENTAL RESULTS

4.1 Magnetometer

Figure 11 shows the computed heading rates for the magnetometer, GNSS carrier, and NovAtel truth. We can see that when the NovAtel truth is the heading rate of carrier is closer to the truth than magnetometer at most times. However, at times, there are jumps in the carrier measurements, which may have been due to the carrier cycle slips.

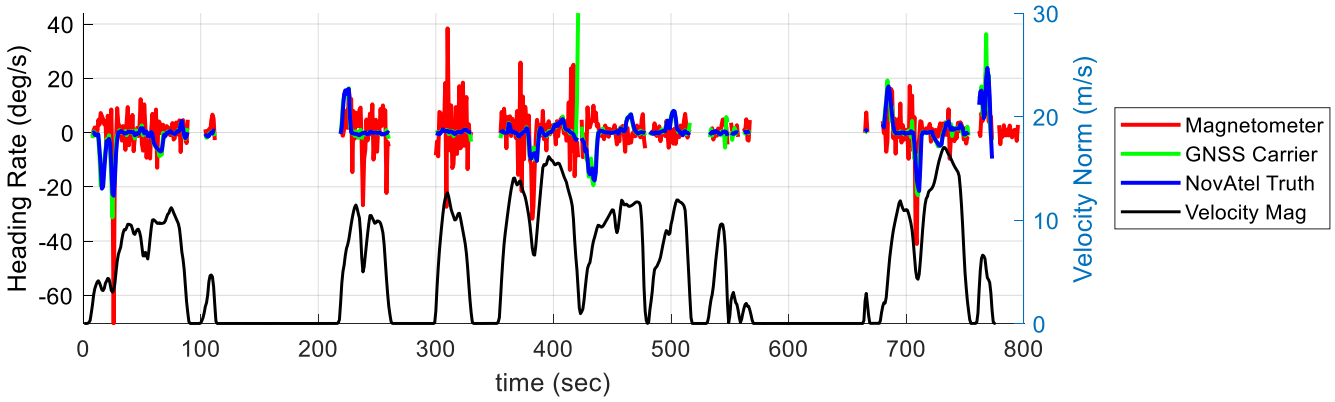

Figure 11. Heading rate computation and comparison between the magnetometer, GNSS carrier, and NovAtel. No heading rate is computed for velocities lower than 1m/s.

From Figure 12, we can observe the accuracy of the heading rate with respect to the velocity magnitude. At lower velocities, the accuracy of the heading rate is lower than at higher velocities. Therefore, more conservative thresholds are required at low velocities.

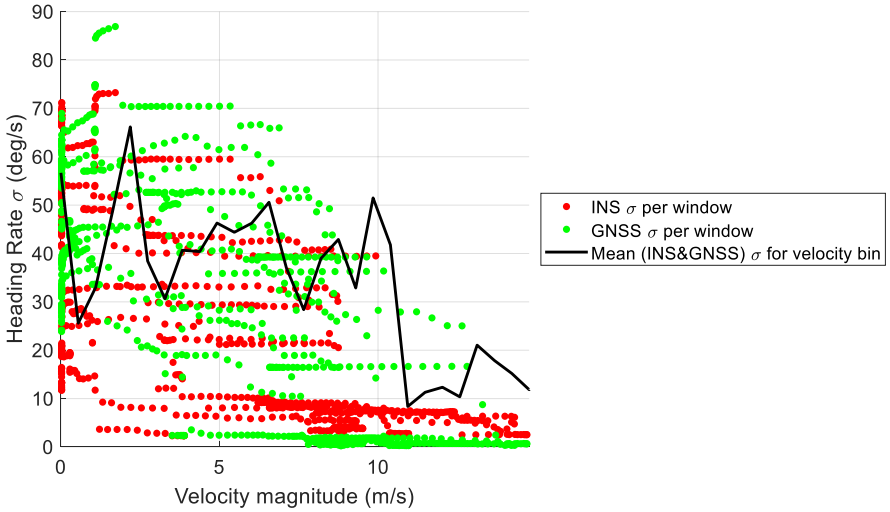

Figure 12. Heading rate standard deviation with respect to the magnitude of the velocity for data collection scenario. At higher velocities, the agreement of the heading rate between INS and GNSS increases.

After the preliminary thresholds were established from the data collection scenario, they were tested using the validation scenario. For the false detection rate, for the data collection scenario, it was maximum of 0.3%, and for the validation scenario, it was 0% for both mobile devices. From statistics and what we can see from Figure 13, we can see that the thresholds were too conservative. Therefore, tighter thresholds are possible for both Mi8 and P10.

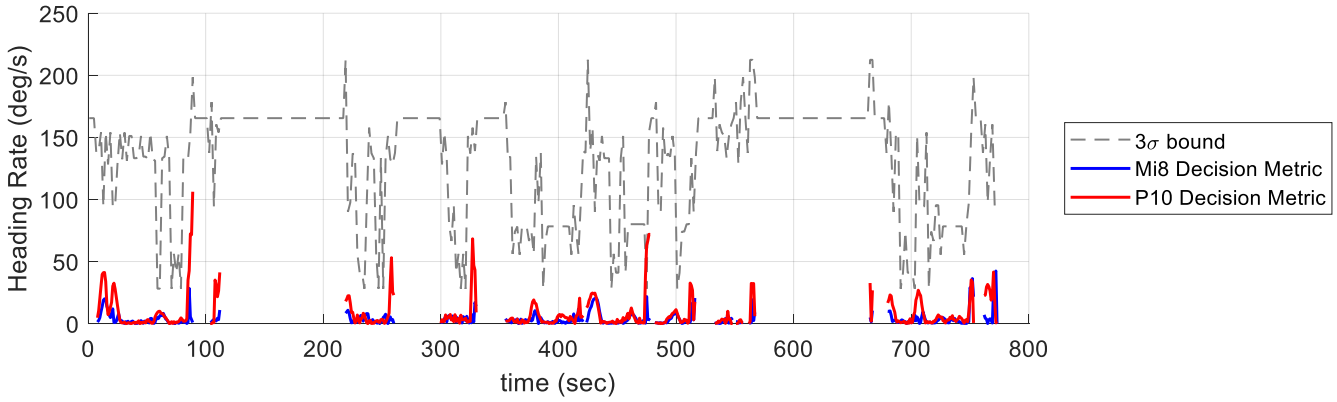

Figure 13. Heading rate decision metrics for Mi8 and P10 using the validation scenario. 100% of the decision metrics for both Mi8 and P10 are within the threshold.

4.2 Accelerometer

For the gravity removal, we tested various cutoff frequencies and compared the resulting linear acceleration with the NovAtel truth linear acceleration. In order to compare the performance of the frequencies, we took the differences in the acceleration measurements, and took the root mean square (RMS) of all the differences over the entire test period. Afterwards, we determined the frequency at which the RMS difference was minimized. Figure 14 shows the comparison of different cutoff frequencies and their respective RMS values. From the results, 0.12Hz was the minimizing frequency, so was selected as the frequency for the gravity removal.

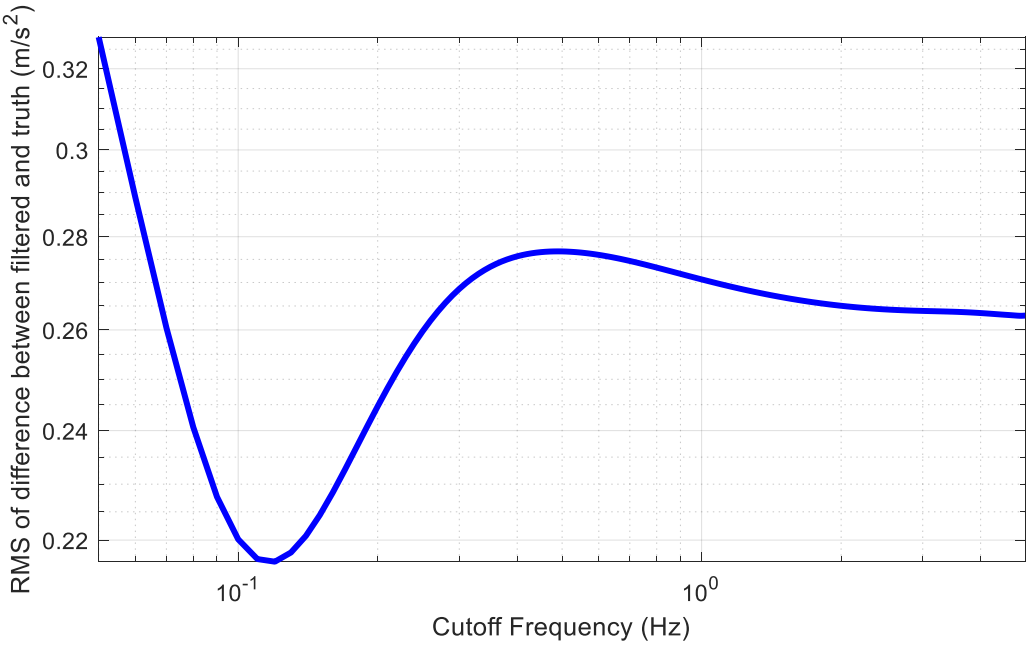

Figure 14. Cutoff frequency versus RMS matching between filtered linear acceleration and NovAtel linear acceleration. Cutoff frequency of 0.12Hz minimizes the RMS of the difference between the filtered and truth accelerations.

Figure 15 verifies that the proposed filter is able to successfully remove gravity from the raw acceleration. Compared to the raw measurements which has an offset due to gravity, the filtered acceleration no longer has a bias.

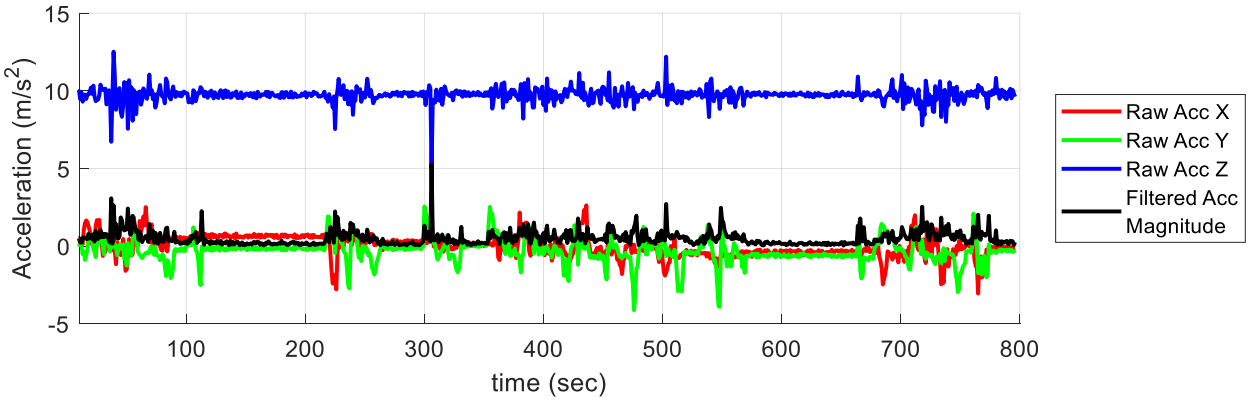

Figure 15. Proof of concept of gravity removal using high-pass filter. The raw acceleration has effects of gravity, while the filtered acceleration magnitude shows that the gravity is removed.

However, we need to keep in mind that this cutoff frequency is not the perfect value. As we can see from Figure 16, when the filtered acceleration is deviating from 9.81m/s^2, which indicates that there are dynamics acting on the mobile device, the difference between the filtered and truth linear accelerations increase as well. This is because the high-pass filter removes low frequency measurements, but if the direction of gravity is changing, the gravity is not removed properly. Also, if there are slowly varying acceleration components, they are removed by the filter as well, leading to decreased sensor accuracy.

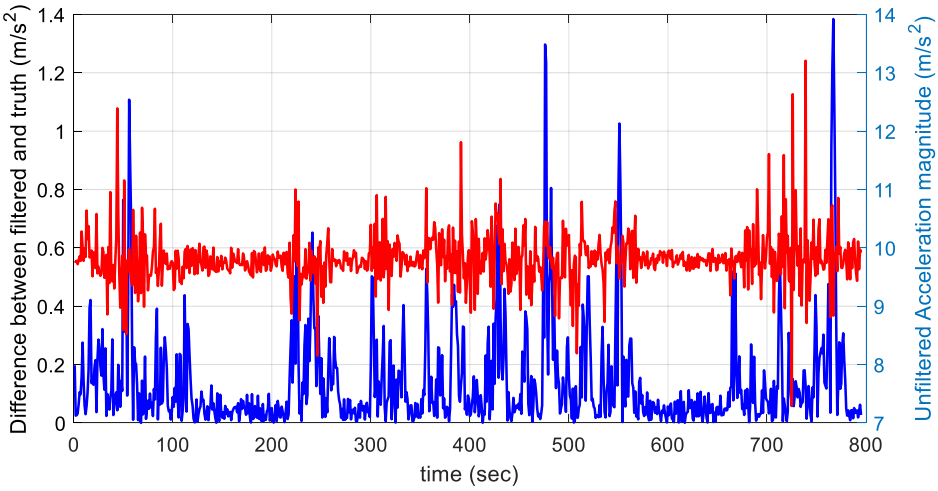

Figure 16. Performance of gravity removal under various levels of dynamics. When there are higher deviations of acceleration from the gravity magnitude of 9.81m/s^2, the effectiveness of the filter decreases.

After the gravity removal, we use the orientation angles to rotate the local body frame to the ENU coordinate frame. In order to prove that the processes have been done correctly, we verified the algorithms using NovAtel raw acceleration, and ENU acceleration output. We removed the gravity from the raw acceleration, then used the carrier-derived orientation angles to rotate the accelerations from the body frame to the ENU frame. From Figure 17, we can see that the rotated accelerations do indeed match the reported ENU acceleration.

After the data processing, we looked at the accuracy of the ENU accelerations with respect to the truth. In Figure 18, we can see that at higher dynamics, the acceleration accuracies decreased. Also, we can see that similar to the ENU acceleration, the acceleration magnitude accuracy decreased at higher dynamics. However, as the acceleration magnitude is not affected by error due to rotation by orientation angles, the overall accuracy was higher.

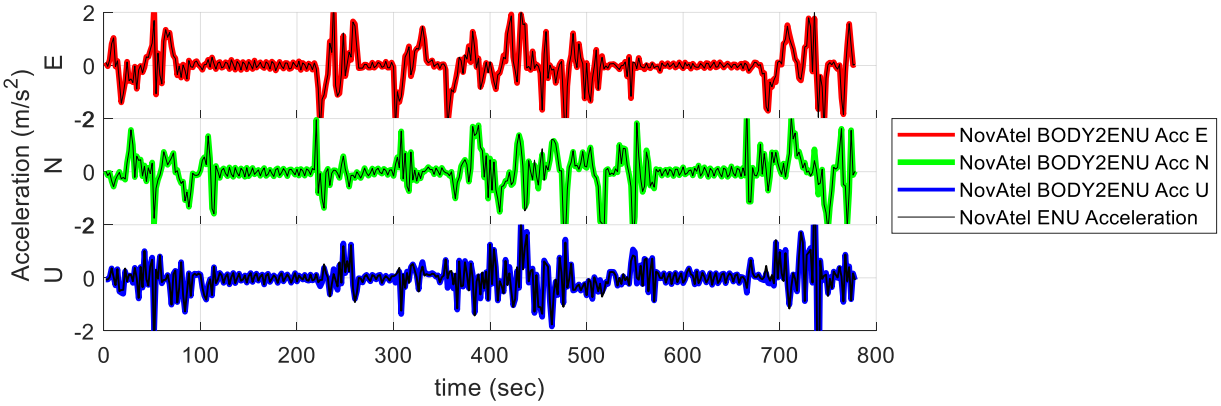

Figure 17. Proof of concept of rotating from body frame to ENU frame using filtered and rotated raw NovAtel acceleration data (BODY2ENU Acc) and reported ENU acceleration data. The filtered and rotated measurements have agreement.

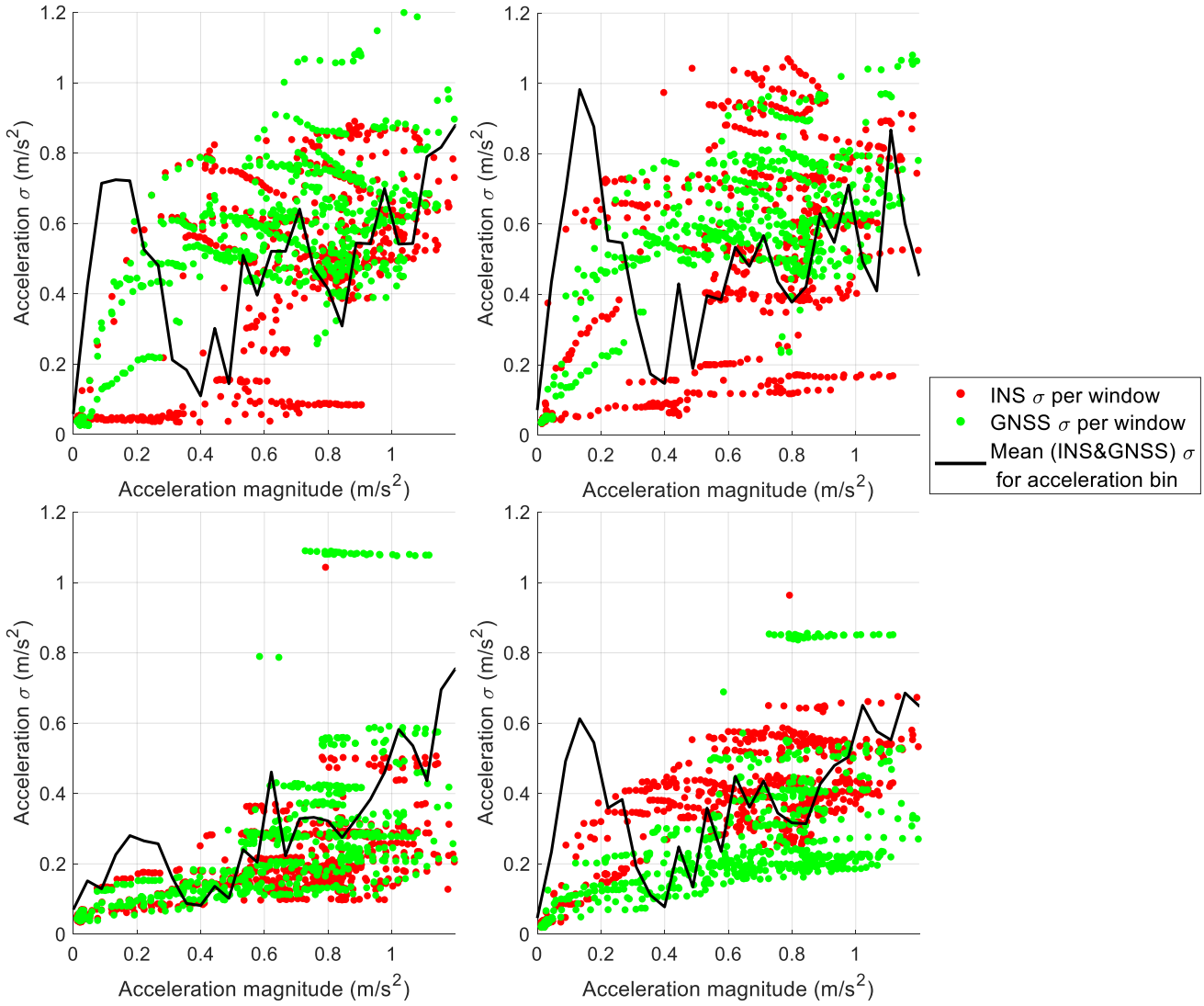

Figure 18. East(top left), North(top right), Up(bottom left), magnitude (bottom right) acceleration standard deviation per acceleration magnitude bin. At higher dynamics, the agreement between INS and GNSS decreases.

Finally, the results for the testing of the proposed thresholds is shown in Figure 19. For the ENU acceleration, the maximum false detection rate was 0.5% when considering both mobile devices and all axis for the data collection scenario. For the validation scenario, the maximum ENU false detection rate was 4.92%. In the case of acceleration magnitude, the maximum false detection rate was 0.4% for data collection scenario, and 4.92% for the validation scenario as well. These indicate that the thresholds were too tight, and should be loosened to account for greater decision metrics.

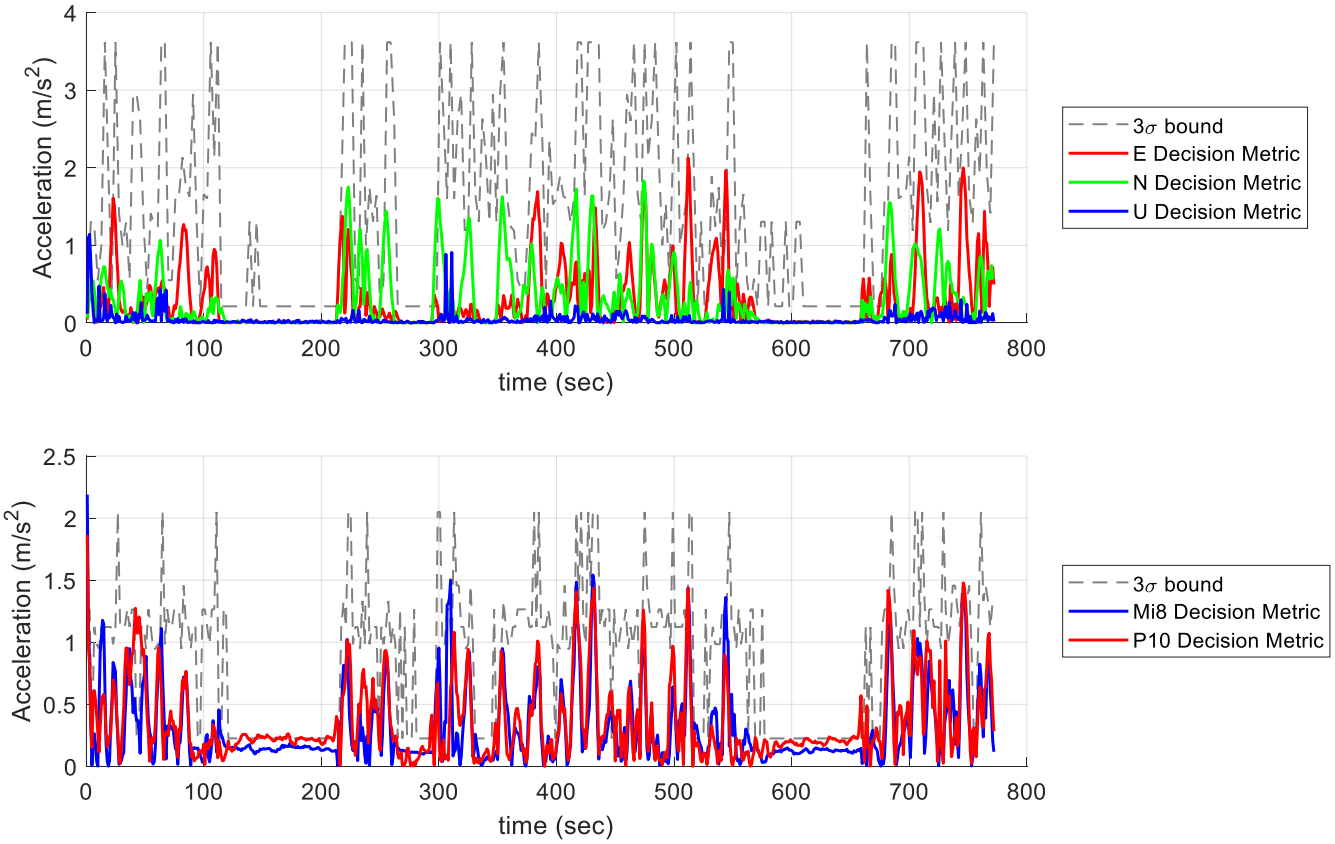

Figure 19. Acceleration decision metrics for Mi8 and P10 using the validation scenario. The top figure shows the ENU decision metrics for Mi8, and the bottom figure shows the magnitude decision metrics for both mobile devices. For both the magnitude and ENU acceleration comparison, maximum of 95.08% of the decision matrices were below the threshold.

4.3 Barometer

As mentioned in the methodology section, when we compute the altitudes from different sources, there are clear biases present as seen in Figure 20. Also, it is noticeable that although there are distinct biases, there is a similar trend for the change of altitude between different sources.

When we compute the altitude rates for the barometer and raw GNSS carrier, we can see that the barometer readings are noisier with respect to the NovAtel truth measurements than raw GNSS carrier-based readings as shown in Figure 21. If we compute the difference from the NovAtel measurements, the barometer has a standard deviation of 0.4m, and the carrier has a standard deviation of 0.2m. This is expected as the noise level of the barometer is greater than the GNSS chipset and NovAtel navigation solution suite.

When we look at the accuracy of the altitude rates in Figure 22, we can see that as expected, there is no significant correlation between the dynamics and the altitude rate accuracy. However, there is a little increase maybe due to scalability problems, or data syncing issues. This is because compared to low dynamic variations, at high dynamic variations, the data mismatching will introduce greater amounts of errors in the comparison.

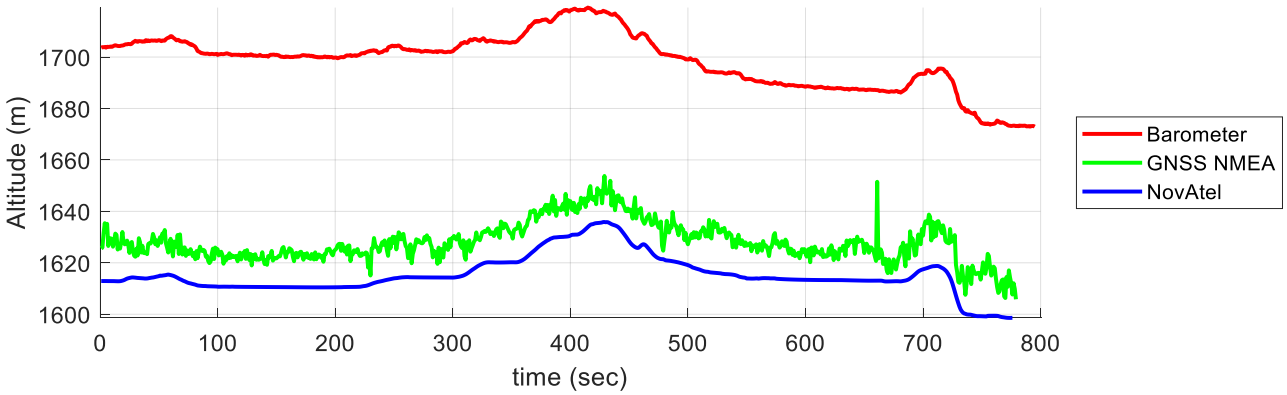

Figure 20. Altitude estimation from different sources: mobile device barometer, GNSS NMEA messages, and NovAtel receiver. There is a bias offset between each of the altitude sources.

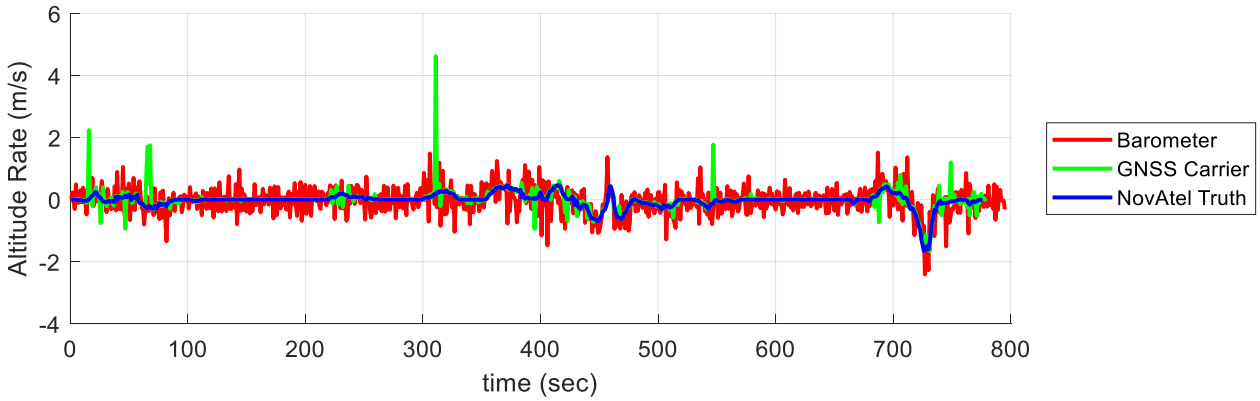

Figure 21. Altitude rate computation from barometer, GNSS carrier, and NovAtel GNSS receiver. Barometer has greater noise ($\sigma = 0.4$m) than the GNSS carrier ($\sigma = 0.2$m) with respect to the NovAtel truth.

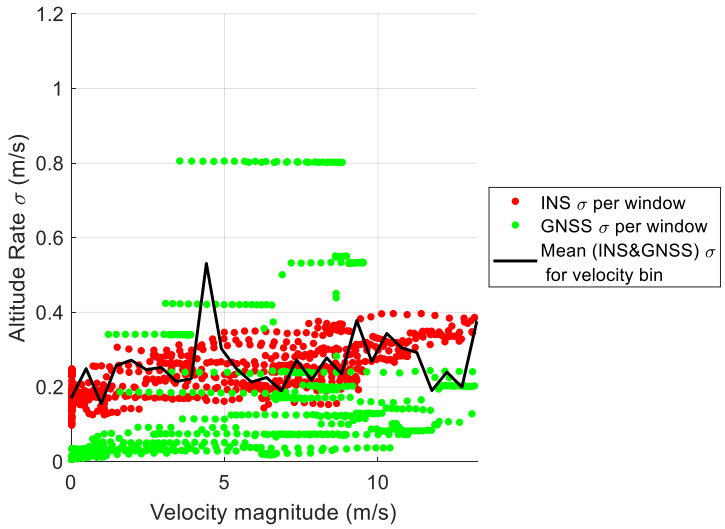

Figure 22. Altitude rate standard deviation per velocity magnitude bin. At higher dynamics, there is a slight gradual increase in the disagreement.

Finally, when we compare the false detectability statistics, the collection scenario had 0.4%, and the validation scenario had 0.9%. This indicates that the thresholds were sufficient, but if we look at Figure *23*, at low dynamics, we could establish tighter thresholds for better spoofing detection.

As mentioned in the methodology, local correction from QNH data may increase the accuracy of the barometer measurements, and hence decrease the proposed threshold for better spoofing detection.

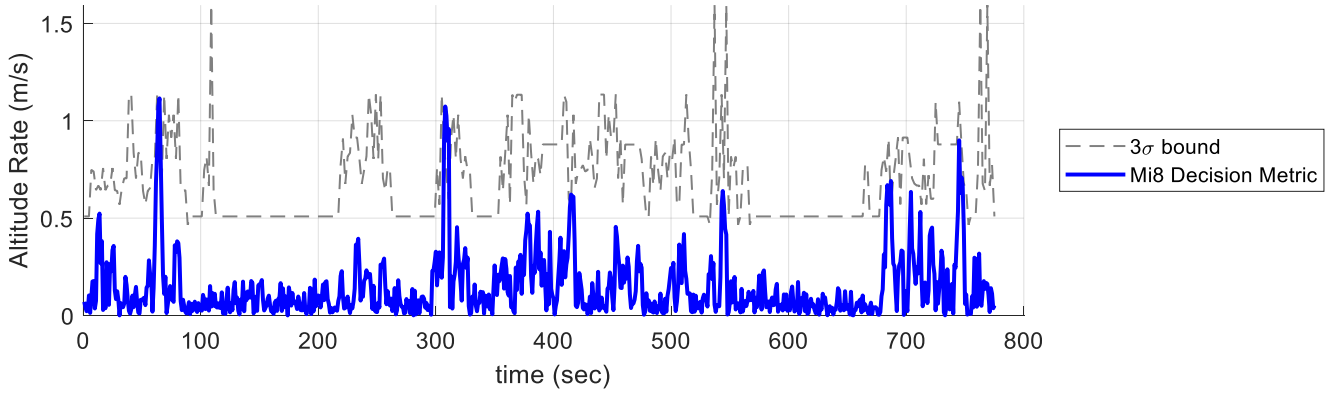

Figure 23. Altitude rate decision metrics and threshold for the validation scenario. 99.1% of the decision metrics are within the threshold.

4. CONCLUSION

In this paper, the feasibility of using Android smartphone inertial sensors and raw GNSS measurements to detect potential GNSS spoofing was explored. In order to investigate the feasibility, comparison metrics derived from different inertial sensors and raw GNSS measurements were assess using a proposed decision metric. Furthermore, preliminary dynamics dependent thresholds were established for different measurements, and tested under two independent scenarios.

Although the feasibility has been assessed, there are several limitations identified in this paper. First, the proposed thresholds will be different for various mobile devices. This is because different devices may not have the same quality and type of inertial sensors, contributing to differences in expected sensor accuracies. Furthermore, depending on the type of GNSS chipset available on the mobile devices, the accuracy of the raw GNSS measurements-derived navigation solutions may vary as well. Second, the syncing of the inertial sensors with the GNSS carrier derived measurements is still a difficulty, and although we have proposed a methodology for this in this paper, other techniques should be explored.

Some future works would be testing the thresholds under actual spoofing scenarios to test misdetection statistics, and improving the accuracy of the navigation solutions in the form of Kalman filters, and measurements smoothing. The misdetection statistics would define the limiting factors of increasing the threshold, and the improvement of measurements for better detection statistics may apply to both the GNSS-derived measurements and inertial sensors.

ACKNOWLEDGMENTS

This material is based upon work partially supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE 1144083

REFERENCES

1. Miralles, D., Levigne, N., Akos, D. M., Blanch, J., Lo, S., "Android Raw GNSS Measurements as the New Anti-Spoofing and Anti-Jamming Solution," *Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018), Miami, Florida, September 2018*, pp. 334-344.
2. Robustelli, U., Baiocchi, V., Pugliano, G., "G. Assessment of Dual Frequency GNSS Observations from a Xiaomi Mi 8 Android Smartphone and Positioning Performance Analysis," *Electronics,* Vol. 8, No. 1, 2019, pp. 91.
3. Goodin, D., "A $225 GPS Spoofer Can Send Sat-Nav-Guided Vehicles into Oncoming Traffic," *Ars Technica*, 2018, arstechnica.com.
4. Wang K., Chen S., Pan A., "Time and Position Spoofing with Open Source Projects," *Black Hat Europe*, Vol. 148, 2015.
5. Scott, L. "Spoofing Incident Report: An Illustration of Cascading Security Failure." *Inside GNSS*, 9 October 2017, insidegnss.com.
6. Lo, S., Chen, Y. H., Akos, D. M., Miralles, D., Cotts, B., "Tests of Crowdsourced Smartphones Measurements to Detect GNSS Spoofing and Other Disruptions," *Proceedings of the 2019 International Technical Meeting of The Institute of Navigation*, Reston, Virginia, January 2019, pp. 373-388.
7. Lo, S., Chen, Y. H., Reid, T., Perkins, A., Walter, T., Enge, P., "The Benefits of Low Cost Accelerometers for GNSS Anti-Spoofing," *Proceedings of the ION 2017 Pacific PNT Meeting*, Honolulu, Hawaii, May 2017, pp. 775-796.
8. Borio, D., "An Experimental Evaluation of GNSS/INS Verification Strategies for Vehicular Applications," *IEEE Intelligent Transportation Systems Magazine*, Pending publication
9. Nelson, R. C., "Flight Stability and Automatic Control," 2nd ed., *New York: McGraw-Hill*, 1989.
10. Van Graas, F., Soloviev, A., "Precise Velocity Estimation Using a Stand-Alone GPS Receiver," *Navigation*, Vol. 51, Issue 4, 2004, pp.283-292.
11. Misra, P., Enge, P., "Global Positioning System: Signals, Measurements, and Performance," *Ganga-Jamuna Press*, 2006.
12. "Position Sensors," *Android Developers*, developer.android.com/guide/topics/sensors/sensors_position.
13. Gebre-Egziabher, D., Elkaim, G. H., Powell, J. D., Parkinson, B. W., "A non-linear, two-step estimation algorithm for calibrating solid-state strapdown magnetometers," *8th International Conference on Navigation Systems*, St. Petersburg, Russia, 2001.
14. Curran, J. T., Broumendan, A., "On the use of low-cost IMUs for GNSS spoofing detection in vehicular applications," *Proceedings of the International Technical Symposium on Navigation and Timing*, Toulouse, France, November 2017, pp. 1-8.
15. Barton, H., Grant, M., Guise, R., "Shaping neighbourhoods: A guide for health, sustainability and vitality," *New York: Spon Press*, 2003.
16. Ozyagcilar, T., "Layout Recommendations for PCBs Using a Magnetometer Sensor," *Freescale Semiconductor*, cache.freescale.com/files/sensors/doc/app_note/AN4247.pdf.