A Robust *GNSS* LOS/NLOS Signal Classifier

ROI YOZEVITCH, BOAZ BEN MOSHE, and AYAL WEISSMAN
Ariel University, Israel

Received July 2015; Revised May 2016

ABSTRACT: *GNSS signal classification to* LOS *and* NLOS *signals is of great value for conventional ranging-based and shadow matching algorithms. The most common attribute for performing this classification is the signal strength. Alas, such classification is often insufficient, in particular, in urban environments. In this paper, we present a novel approach for* LOS/NLOS *classification utilizing supervised machine learning algorithms. Provided with a sufficiently large labeled training set, the proposed approach is able to predict with high certainty (>85 percent) the satellites' visibility status in dense urban regions. This achievement was possible due to the vast raw measurements supplied for the algorithm and using sophisticated feature-selection techniques. Copyright © 2017 Institute of Navigation.*

INTRODUCTION

"Life has a rude way of intruding on theory"

Anonymous

The notion of Shadow Matching (*SM*) as an algorithm for *GNSS* accuracy improvement in dense urban canyons was first introduced by [1], followed by [2]. In essence, *SM* utilizes geometric constraints (obtained by 3*D* maps) in order to reduce the searching space within the receiver's predefined Region Of Interest (*ROI*). The main paradigm behind the *SM* algorithm is that *GNSS* signals can be (under some assumptions) treated as sun-rays that are blocked in the presence of obstacles such as buildings. An unblocked *GNSS* signal holds a direct geometric line of sight to the receiver and therefore can be classified as a Line-Of-Sight (*LOS*) satellite while a shaded *GNSS* signal has no direct line of sight, and hence, can be classified as a Non-Line-Of-Sight (*NLOS*) satellite. In theory, *SM* algorithms utilize this *LOS*/*NLOS* geometric distinction along with 3*D* data of the terrain to eliminate implausible areas within the *ROI*; i.e., areas where the *GNSS* receiver reports *LOS* to a specific satellite which is known to be blocked by a building and vice-versa. This method can reduce the searching space to a fraction of its actual size and along with other algorithms (e.g., Bayesian filters) can considerably enhance the *GNSS* receiver accuracy in those regions. Alas, one cannot escape from inherent inaccuracies in both the 3*D* data and the *LOS*/*NLOS* classification. Therefore, current *SM* algorithms utilize probabilistic scoring according to how well the predicted and observed *LOS*/*NLOS* classifications match [3, 4]. For a detailed Shadow Matching discussion the reader is referred to [1, 3–6].

Put in different words, given a 3*D* map of the terrain and buildings and given an **exact** method for *LOS*/*NLOS* classification to each navigation satellite, the *GNSS* accuracy can be improved significantly by implementing *SM* techniques. The denser the region, the greater the improvement. A correct *LOS*/*NLOS* classification is the heart of every *SM* algorithm, and is therefore, mandatory. In addition, other *GNSS* related algorithms, such as mapping algorithms or even conventional ranging-based positioning, can benefit tremendously from such an accurate *LOS*/*NLOS* classification method [7–10]. Assuming *NLOS* measurements are rejected or down-weighted, a more accurate position can be computed. To the best of our knowledge, a robust *LOS*/*NLOS* signal classification algorithm does not exist, thus, *SM* algorithms have not yet reached their full potential. In this work, we present a robust framework for the *LOS*/*NLOS* classification problem based on data mining techniques, in particular supervised machine learning algorithms. The term "*supervised*" implies that we can *a priori* know the signal visibility state and based on this information, extract rules for *LOS*/*NLOS* classification (elaborated in Supervised and Unsupervised Learning Algorithms Section).

The strength of the proposed method relies upon two key elements:

- The ability to exploit the vast raw data from the receiver, data which goes almost unnoticed.

NAVIGATION: *Journal of The Institute of Navigation*
Vol. 63, No. 4, Winter 2016
Printed in the U.S.A.

- The ability to create a supervised environment which allows the algorithm to learn the features of the terrain.

We believe that by combining those two elements new insights and classification rules can emerge. These rules are the building blocks of the robust classifier.

Paper Structure

The rest of this paper is organized as follows. The next section is dedicated to the related work already conducted on this problem. Since each *SM* algorithm must perform *LOS*/*NLOS* classification, several methods are mentioned. A LOS/NLOS Classifier Framework Section discusses the vast number of raw *GNSS* measurements we have applied in the classification process. Supervised and Unsupervised Learning Algorithms Section is devoted to the main learning algorithm (decision trees). The *LOS*/*NLOS* *a priori* labeling is also discussed there. In the Results Section, we present both the experiments we have conducted and their results. In Conclusion and Future Work Section, we conclude.

RELATED WORK

Since correct *LOS*/*NLOS* classification is of the utmost importance to every shadow matching algorithm, previous works have already faced this challenge.

The demand for *LOS*/*NLOS* classification is not restricted to SM algorithms solely and some conventional ranging-based *GNSS* positioning algorithms utilized a 3*D* city model for *LOS*/*NLOS* detection and *NLOS* corrections [9–11].

Since the carrier frequencies of *GNSS* are characterized by considerable signal attenuation in *NLOS* conditions, the most intuitive candidate for improving *LOS*/*NLOS* classification is the carrier-to-noise power ratio (also referred to as *C/No*). Moreover, [6] have shown that under sterile conditions (a single wall), the *C/No* value can serve as an excellent classifier.

One major flaw in this approach is its lack of symmetry. While a strong *GNSS* signal is very likely to indicate *LOS*, the opposite is not so clear. *LOS* signals can be detected with low *C/No* for various reasons (e.g., the antenna position, momentary blocks, etc.) and *NLOS* signals can be detected with high signal strength. In an attempt to compensate for that asymmetry, previous works [7] have modeled the relation between the *C/No* (signal strength) and the *LOS-NLOS* likelihood according to Rician/log-Normal distributions. This is demonstrated in Figure 1. A standard commercial Code Phase receiver with an external antenna is assumed for that figure.

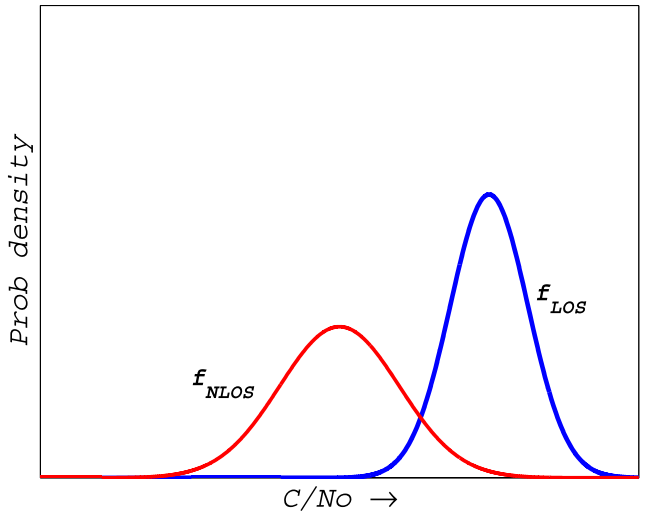

Fig. 1–A LOS/NLOS likelihood probabilistic model of a satellite's signal strength. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

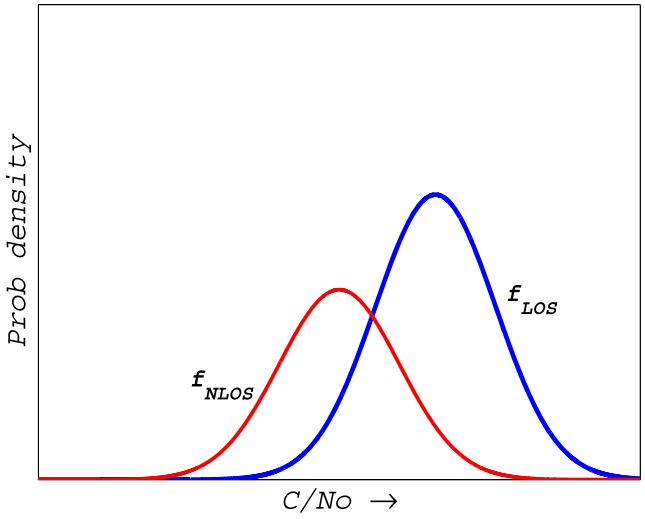

Fig. 2–A Smart-phone LOS/NLOS likelihood probabilistic model. The NLOS and LOS distributions overlap a lot more than shown in Figure 1. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

However, empirical research has shown that when using smart phones, the *LOS*/*NLOS* distributions overlap a lot more [4]. Figure 2 shows a smart phone *LOS*/*NLOS* likelihood probabilistic model of a satellite's *C/No*.

From a geometric perspective solely, a *GNSS* signal has either *LOS* or *NLOS* (you either see the satellite or not). In reality, however, *NLOS* signals can be further classified into two sub-categories: blocked or reflected signals. Although one can assume that brick-reflected signals are characterized by lower *C/No*, strong specularly reflected signals, especially from reflective material buildings, can be received with high *C/No*. Concrete examples of this phenomena are depicted in Figure 3. Moreover, a border-line *NLOS* signal (up to 3 degrees below the block) is received as *LOS* due to diffraction [12, 13].

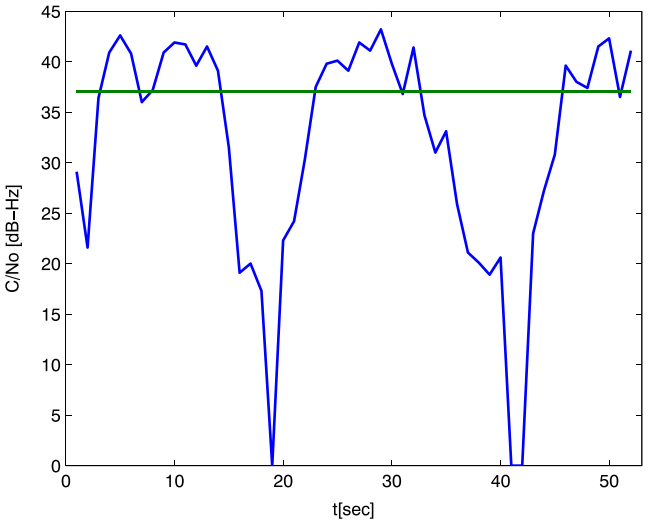

Fig. 3–Smartphone C/No measurement over time. The recording was done while walking in front of a building. Empirical threshold for LOS/NLOS classification. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

LOS/*NLOS* classification techniques can be divided into methods which require additional hardware and methods which don't.

Utilizing additional hardware can help detect multipath signals. Since most *LOS* signals have RHCP (right hand circular polarization), most multipath signals have LHCP (left hand circular polarization). The case of multipath signal via two reflections is rare since it holds a very low *C/No*. Thus, [14] have suggested to use a pair of antennas - one sensitive to RHCP and one sensitive to LHCP. Since the desired goal is to improve the accuracy of commercial *GNSS* devices, we restrict ourselves to methods which don't require additional software. If additional hardware is permitted, one can benefit from a sky-pointing camera as suggested by [15, 16]. In fact, this is the best theoretical approach to *LOS*/*NLOS* classification since a geometric *LOS* is what the algorithm seeks for. The use of panoramic camera to detect skyline requires an *a priori* data of the terrain's skylines. We have used a camera in this paper in a very similar way. However, the purpose was to create a supervised training set - *GNSS* measurements along with their geometric *LOS*/*NLOS* labeling.

It is worth mentioning that several works have been conducted along this very idea of *directly* seeking for *LOS* in the skylines. The most famous one is [16]. in that paper, the researchers captured city skylines using a fish-eye camera and tried to estimate the receiver (camera) position given a large enough database of those skylines. A more recent work using the same fish-eye camera was conducted to differentiate between *LOS* and *NLOS* satellites [10].

Beside the *C/No*, the satellite elevation also serves as a probabilistic *LOS* indicator where the lower the elevation angle is, the more probable the signal is *NLOS*. This concept is old and even for conventional *GNSS* positioning, measurements are weighed according to the *C/No* and the elevation. Combined together, this data can serve as an input for a scoring technique which reports a *LOS* likelihood. This approach produces better results than a binary *LOS*/*NLOS* classification [4]. Recent Shadow Matching efforts have adopted the probabilistic *LOS*/*NLOS* classification model and focus on the algorithm layer. This information was used later in our algorithm.

Figure 3 shows a recorded *C/No* value of a satellite with an elevation of 64 deg over time while walking against a shaded wall using our receiver (see Figure 15). The graph shows a clear distinction between *LOS*/*NLOS* states. Both the appearance and disappearance of the satellite are very rapid.

Our Contribution

As stated above, one cannot utilize the shadow matching algorithm without seriously considering the *LOS*/*NLOS* classification challenge. Moreover, even conventional range-based position algorithms can benefit tremendously from such classification. Therefore, in this paper we restrict ourselves solely to that challenge. The novelty of this work is threefold: First, we have expanded the searching space beyond the obvious signal strength (elaborated in the next section). Second, utilizing data-mining and machine-learning techniques, a set of rules can be applied to almost any Shadow Matching scenario. Those rules, in turn, can help *SM* algorithm developers and encourage *GNSS* manufacturers to incorporate them in their modules, to produce a probabilistic *LOS*/*NLOS* classifier.

Finally and most importantly, as the opening quote states, real field experiments, especially in urban regions, can be very different from sterile conditions depicted in Figure 3 (single wall obstruction in relatively open sky), thus, many field experiments were conducted. The raw data presented here comes from genuine *GNSS* recordings.

A *LOS/NLOS* CLASSIFIER FRAMEWORK

This section is devoted to the robust classifier general structure, both hardware and conceptual. We believe that such a classifier is viable assuming one can acquire a wide set of *GNSS* raw measurements. By raw, the authors mean ranging and *C/No* measurements. In a nutshell, the classifier's input is the set of *GNSS* raw measurements while its output is a *LOS*/*NLOS* probability function for each tracked satellite. The upper part of Figure 4 shows a simplified block diagram of a *GNSS* receiver. Any *GNSS* application (e.g., Waze [17]) has access to the user position/velocity vectors via the standard protocols such as the National Marine Electronics

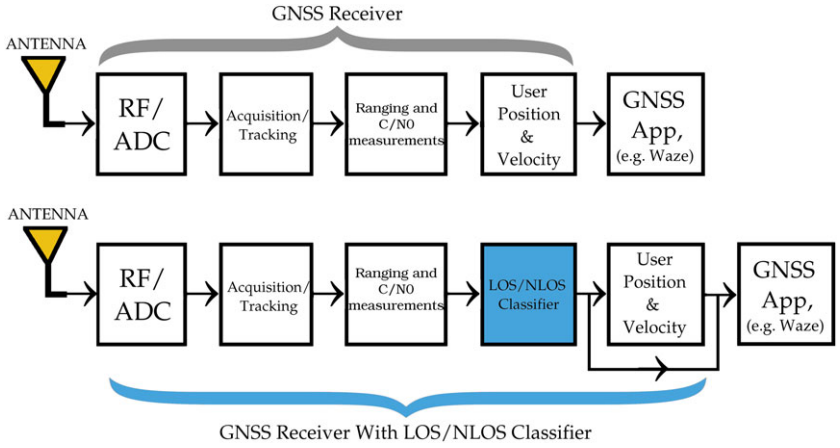

Fig. 4–A LOS/NLOS likelihood probabilistic model of satellite's signal strength. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

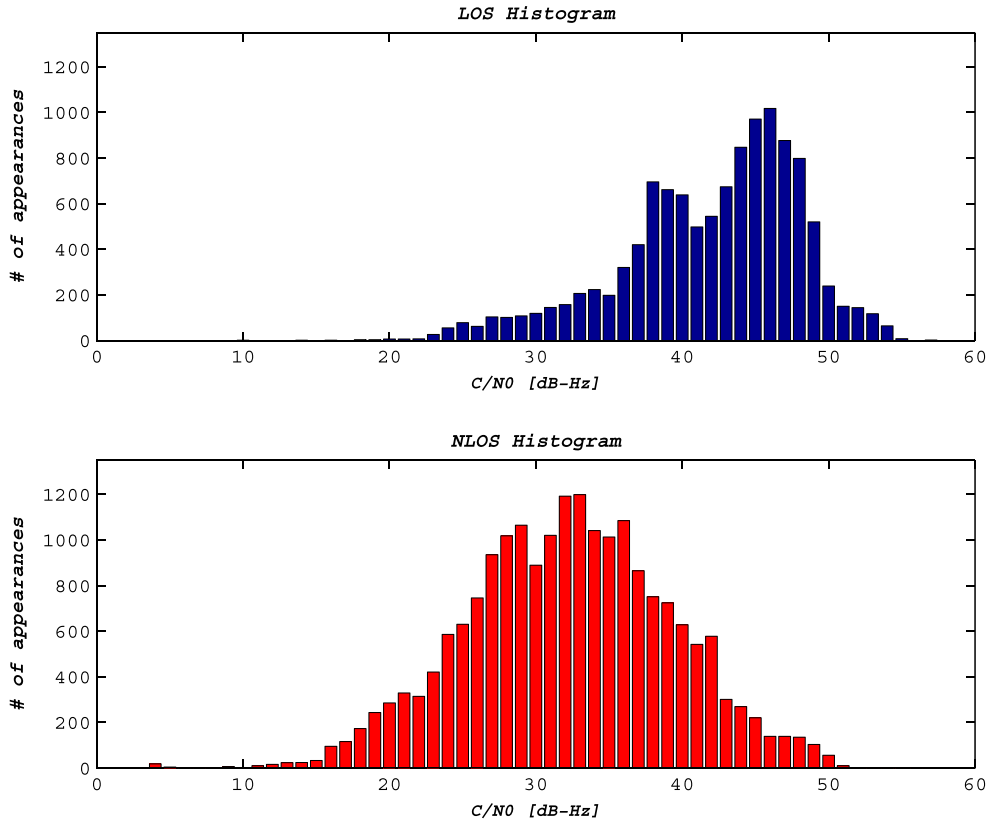

Fig. 5–LOS histogram (upper graph) and NLOS histogram (lower graph). The frequency is measured by number of instances. Over 31,000 samples during eight separate recording were used to construct this graph. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

Association (NMEA) protocol. The lower part of the figure clarifies where the suggested algorithm lies in the chain. The algorithm utilizes raw measurements for *LOS/NLOS* classification. This important information can improve the user position (similar to the Intelligent Urban Positioning algorithm where *NLOS* signals are eliminated from the position equations [12]). In addition, the classification can be reported *directly* to the navigation application (see Incorporating LOS data to the NMEA format Section).

It is important to bear in mind those measurements exist in any commercial C/A receiver. The standard NMEA 0183 protocol, however, reports only a fraction of those, especially position and velocity. The satellite's pseudorange and carrier frequency (hence the Doppler shift) cannot be obtained using this protocol. There are other protocols (e.g., One Socket Protocol Sirf binary by CSR Inc. and propriety messages by STMicroelectronics Inc.), which enable this information. Furthermore, no external sensor data such as an Inertial

Measurement Unit (IMU) is used. The data is obtained solely from the receiver itself.

Supervised and Unsupervised Learning Algorithms Section focuses on the output - the *LOS*/*NLOS* classification along with its confidence values. In this section, however, we shine the light on the input variables - what are the relevant *GNSS* raw data that hold insights regarding the satellite visibility state. Each piece of data (*C/No*, carrier frequency, etc.) is a feature in the classification algorithm. In the next section, we show that the *LOS*/*NLOS* classification gate cannot be opened using a single feature. It is the delicate fusion of those features that make classification viable. After inspecting all of the data one can obtain from a C/A receiver, we believe the following terms to be of great value.

Signal Strength (C/No)

A common opinion in the realm of data mining states that "Very simple classification rules perform well on most commonly used datasets" [18]. Unfortunately, this is not the case in *LOS*/*NLOS* classification. However, the best indicator is still the *C/No*. As mentioned in Related Work Section, while the theoretical signal *C/No* is an excellent *LOS* predictor under sterile conditions, real world urban scenarios draw a different picture. Real world data shows that both low *LOS* and high *NLOS* values exist (the latter is more common). Those phenomena are depicted in Figure 5. One can see this histogram as the empirical and similar version of Figure 1. The two confirm a high *C/No* measurement is a good *LOS* predictor, and low *C/No* measurement is much more ambiguous since the two graphs overlap each other. While *LOS* probability rapidly decreases with *C/No*, *LOS* satellites can be received even in the lower area ($\leq 35\,dB\text{-}Hz$). It is important to mention that during the entire recording (over 31,000 samples), most of the *C/No* values were in the mid-uncertain range ($> 25\,dB\text{-}Hz \& < 45\,dB\text{-}Hz$). The environment was a dense urban region with high and reflective buildings. The data was collected in four different locations within the region using *ORG*1208 *SIRF* receiver by *OriginGPS*.

Moreover, a careful examination of the *C/No* numbers shows yet another interesting phenomenon. Occasionally, short period signal degradation occurs in *LOS* signals. An example of this is seen in Figure 6. Each *GNSS* satellite holds a unique Pseudorandom Noise (PRN) number. For PRN 23, a $10\,dB\text{-}Hz$ degradation occurs in ≈ 10 s period. During the same time, another *LOS* satellite (PRN 4) remained above the threshold and another *NLOS* satellite stayed below (PRN 3). Several explanations can be offered (e.g., a momentary blocking or a measurement noise). However, the point is that this phenomenon exists and a simple signal-strength threshold is insufficient [4].

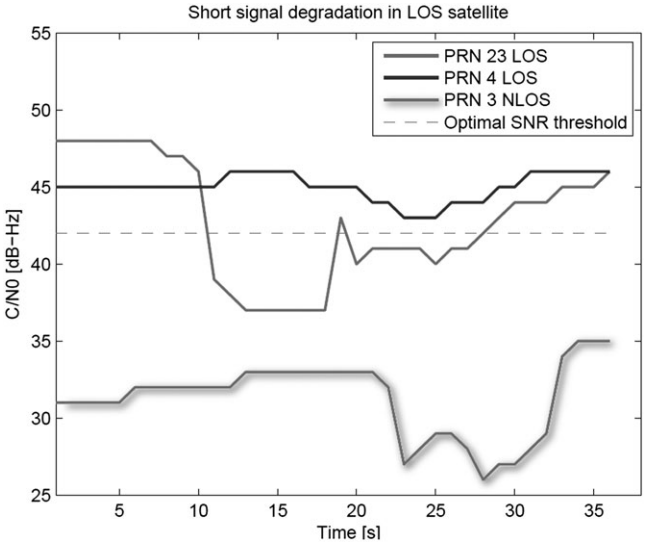

Fig. 6–A demonstration of the insufficiency of a binary C/No threshold. PRN 23 and 4 are LOS while PRN 3 is NLOS. In pink, an optimal C/No LOS/NLOS threshold of 42 *dB-Hz. In the course of* 10 *s PRN 23 decreases below the threshold.*

Pseudorange

To put it simply, the pseudorange is computed as the time delay from the signal being transmitted from the satellite to the detection of the signal in the receiver multiplied by the speed of light plus the clock synchronization error multiplied by the speed of light. The theoretical range is supposedly the distance between the receiver and the satellite. If the latter position is known from the ephemeris, the receiver position can be computed [19]. It is important to mention that this value increases/deceases in time since the satellite approaches the receiver from the horizon to the zenith and retrogresses from the zenith to the horizon. Once the receiver's computed position is known, the 'real' range between the receiver and the satellite can be computed. The difference between those two numbers, also known as pseudorange residuals is of great value for *LOS*/*NLOS* computation as stated in [12]. *NLOS* signals can (also) be detected via their multipath (reflections from other buildings). Since pseudorange is a time measurement, reflected signals travel a longer distance, and hence, they are larger. However, when direct and reflected signals are received, the measured pseudorange can be longer or shorter than the direct range. In any case, it contains more noise. In this paper's context, the pseudorange value itself of a specific satellite is of no interest to us. It is the pseudorange residuals we seek. Large residuals, at least in theory, imply *NLOS* signals. Alas, this is the case when only a small portion of the signals are *NLOS* [20]. Another

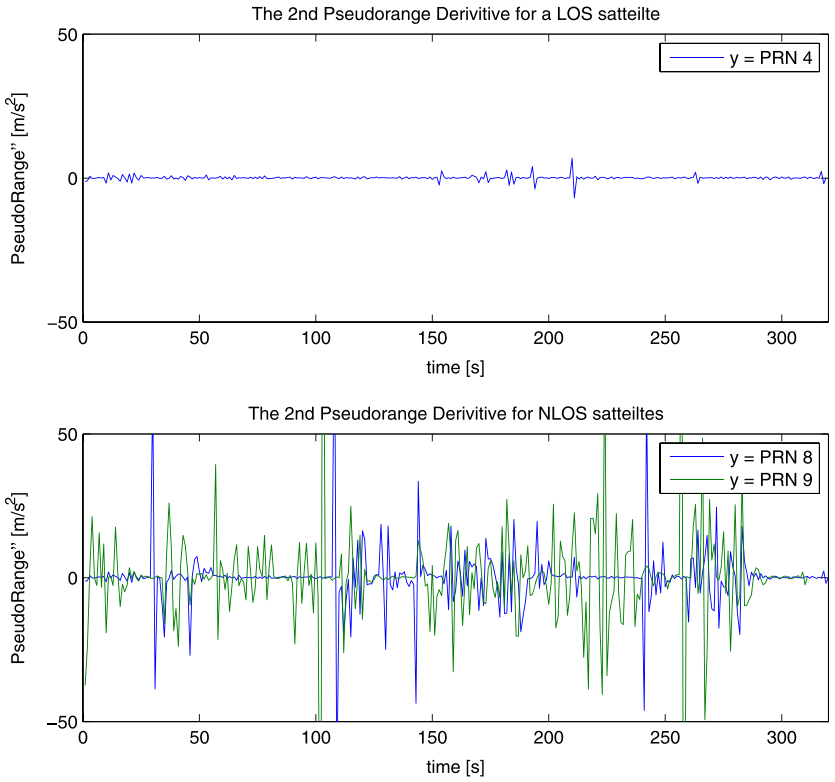

Fig. 7–The effect of noisy pseudorange measurements. The graphs demonstrate second pseudorange time derivative for both LOS and NLOS satellites. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

ramification of utilizing pseudorange residuals is that the receiver's position must be computed. Done with several satellites (some of which are *NLOS*), the reported position is biased. One can overcome this noise by examining the estimated pseudorange derivatives (both first and second). We define $PR_{SV_X}(t)$ as the computed pseudorange in time t for satellite x. Therefore, its first approximated derivative is:

$$PR_{SV_X}(t)' = \frac{PR_{SV_X}(t) - PR_{SV_X}(t-dt)}{dt} \quad (1)$$

Since our main goal is to differentiate between *LOS* and *NLOS* signals, the time of flight represented by the pseudorange seems to be more sensitive to *NLOS* (multipath-reflected) than the Doppler shift which is being used for approximating the relative speed between the receiver and the satellite. In low (or zero) velocities, the pseudorange second derivative seems to be a relatively reliable predictor of *LOS*/*NLOS* change while the Doppler shift and its derivative often seems immune to *LOS*/*NLOS* transitions. We define the second approximated derivative as:

$$PR_{SV_X}(t)'' = \frac{PR_{SV_X}(t)' - PR_{SV_X}(t-dt)'}{dt} \quad (2)$$

Figure 7 demonstrates a graph of the second derivative, both for *LOS* and *NLOS* satellites. One can see the *NLOS* values are far more noisy. As explained above, this is due to the inherent *NLOS* reflection effect. However, although the difference is easily seen over a long inspection time period ($\approx$ 30 seconds), a single sample inspection produces much less certainty.

Borderline Signals

As discussed above, the authors of [12] have pointed out that *NLOS* signals can be detected as *LOS* around the *LOS-NLOS* transition elevation. Such borderline signals can be omitted from the algorithm. The purpose of this elimination is to tackle the inherent 3*D* map inaccuracies since those inaccuracies mainly affect the borderline signals (*LOS-NLOS* misclassifications).

Satellite Elevation

While the satellite's elevation itself does not imply a *LOS*/*NLOS* signal (this is why it appears last), it can be useful in cases of ambiguity; there is a clear positive correlation between the elevation of a satellite and its probability to be considered *LOS*. A satellite in the zenith is by definition *LOS* while as the elevation decreases, the satellite is more likely to be blocked by buildings and other man-made obstacles. Again, the elevation value mainly aids the confidence computation.

SUPERVISED AND UNSUPERVISED LEARNING ALGORITHMS

This section gives a brief introduction to supervised and unsupervised machine learning algorithms, specifically Decision Trees ($C4.5$) and EM (Expectation Maximization). For a detailed discussion on this fascinating subject, we refer the reader to [21–23]. A machine learning algorithm is a framework to enable the computer to mathematically model the input data, thus solving problems such as classification, clustering, and regression. Machine learning is further divided into two main approaches - supervised and unsupervised learning. Unsupervised learning can be thought of as merely clustering, i.e., data separation to two (or more) semi-defined groups. Unsupervised learning techniques do not require labeled data, thus acquiring a training set is a relativity simple task. However, the downside of such techniques is that the learning phase cannot do more than merely attempt to divide the data into groups, without an *a priori* knowledge of what the groups are meant to represent. Due to this, feature selection is very important. In this paper, we have utilized the EM algorithm for unsupervised learning. Expectation Maximization (EM) algorithm is an iterative method for finding maximum likelihood or maximum *a posteriori* (MAP) estimates of parameters in statistical models. In our specific case, the statistical model is a Gaussian-Mixture model, in which we have N random variables (the *GNSS* measurements) corresponding to observations, each assumed to be distributed according to a mixture of two components, *LOS* and *NLOS*. EM is a standard algorithm in the realm of machine learning and easy to implement in practicality in any tool (e.g., Matlab, Weka, Python, etc.). Supervised learning, on the other hand, requires the input data to be labeled. Using these labels, supervised learning algorithms can create complex models which take into account many mathematical measures, such as information gain, cross entropy, etc., which in many cases can indicate the meaningful features. A common definition of supervised machine learning is: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E" [24]. In our context, the task T is *LOS*/*NLOS* signal classification. Supervised learning can be described as inferring an optimal function from labeled data [25]. Labeled data is defined as a feature vector and a classification (label). In our context, the labeled data would be the vector of a satellite's features (*C/No*, elevation, pseudorange residuals etc.) along with previous knowledge of its visibility state. A supervised learning algorithm assumes the correct classification is known and that information regarding the data vector properties in this state can be learned (hence the term supervised). One of the key ideas in this paper is that such *a priori* classification is both viable and effective.

A priori LOS/NLOS *Signal Labeling*

Each supervised learning process comprises a training phase and a testing phase [21]. The training phase teaches the algorithm by updating the relative weights (importance) of each feature so that the summation will correctly classify the occurrences. Each region must has its own training data. If the classifier is trained using data from a *US* or East Asian city with wide streets and high-rise glass-faced buildings, it may not work in a European city with narrow streets and medium-rise stone and brick buildings and vice versa. Moreover, the buildings' density within the same city demands its own training data. The training phase requires an *a priori LOS*/*NLOS* classification. In other words, a geometric method for calculating visibility states must be utilized here. This method cannot rely on the signal's features (such as the *C/No*). Facing this problem, several different orthogonal approaches were taken.

The first (and straightforward method) was to utilize data from $3D$ terrain maps. Google Earth holds a database of all buildings in every major city in the US and this data can be extrapolated to geometrically check visibility states from a given location (SpectraCom Inc. announced a feature in their *GNSS* simulator which based on this data creates *GNSS* signals). This approach, however, holds two inherent drawbacks. The first is the assumption that an exact and updated $3D$ building map of the terrain is present. Painful experience has shown us those maps are not always synchronized with reality (a new sky-scraper foundation was present). This is why the elimination of borderline signals is crucial. The second is that a correct *LOS*/*NLOS* classification is not viable if the receiver's location is not given. The latter drawback is easily overcome in the Diamond Exchange district in Ramat-Gan, Israel, since the ground is paved with a grid of large tiles, enabling an exact deduction of the recording spot. One can also use traditional *GIS* surveying techniques to determine to the receiver's exact location.

Another approach for facing the data labeling challenge is by measuring the skyline contour, i.e., measuring both azimuth and elevation of each building corner. This was achieved by an in-house mobile application. A screen-shot from the application we have developed is shown in Figure 8. Given a calibrated compass, the azimuth of each building corner can be easily computed. The elevation is produced a from a fusion of the accelerometer measurement

Fig. 8–An in-house android application which allows us to associate the camera view with a 3D angular vector. This application is used to achieve the satellites' real visibility state. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

and pixel counting. When the device is stationary, its tilt can be extracted from the three-axis accelerometer sensor utilizing a gravity vector analysis [26, 27]. The Android API provides utility functions for that purpose. For a 13 *MegaPixel* camera, the resolution is more than 4 pixels, per milliradian. This method is superior to the skyline contour and the 3*D* data since both the *LOS*/*NLOS* geometric computation and a 3*D* map are unnecessary. The downside is the complexity of the processes (which is done manually).

LOS/NLOS *Classification Algorithm*

We start by introducing a conceptual algorithm. The *LOS* likelihood of satellite x can be defined as:

$$Prob(LOS_X) = \frac{\sum_{n \in N} Weight_n \times Prob(Feature_n)}{\sum_{n \in N} Weight_n} \tag{3}$$

N is the total number of *GNSS* features, $Weight_n$ is the *a priori* weight (importance) of the normalized $Prob(feature_n) \in [0, 1]$. In addition, we define a probabilistic mapping function from the actual measurement to its normalized *LOS* probability (e.g., $C/No = 37\,dB\text{-}Hz$ is mapped to 0.7 or 70 percent *LOS* probability). An example of $Prob(feature_n)$ for the signal's *C*/*No* is depicted in Figure 1. Both *NLOS* and *LOS* probabilities sum to one. Both the weights and the mapping functions are deduced empirically. Those values can be modified, depending on the environment's scenario (urban regions, forest, open-sky, etc.).

After inspecting several supervised machine learning algorithms (e.g., Support Vector Machine, Neural Network) we decided to process the data using a binary decision tree (C4.5). Such a tree classifies its instances via different nodes to two (or more) separate classes (*LOS*/*NLOS*). A major advantage of decision trees is the ease of understanding; one can rather easily extract a set of rules out of them, a crucial feature for a robust classifier. The more nodes and splits the tree has, the more complicated the algorithm. The tree depicted in Figure 9 is a binary treebased solely on the *C*/*No* value with a 37.5 $dB-Hz$ threshold. This tree produces an accuracy rate of 71 percent.

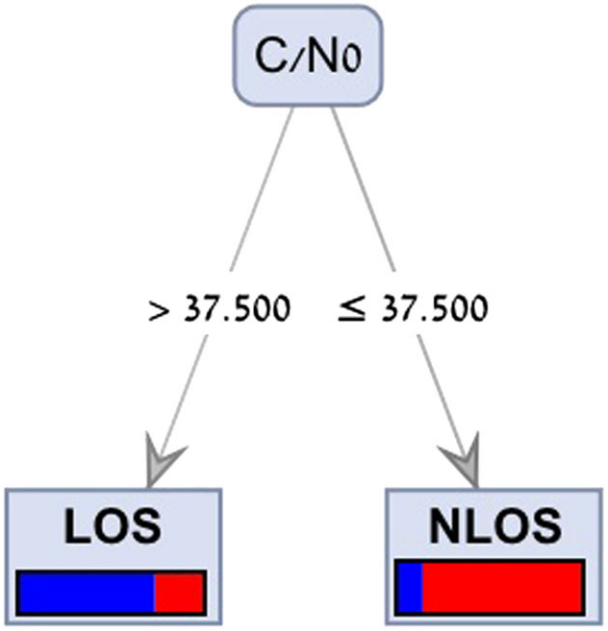

Fig. 9–A decision tree diagram based solely on C/No. Image was fabricated using RapidMiner Software [28]. *[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]*

The optimal (best *LOS*/*NLOS* predictor) tree is seen in Figure 10. The algorithm which creates the tree assures this, using information theory metrics such as cross-entropy and information gain. There are an infinite number of trees for each feature set, so under a certain set of parameters (e.g.,

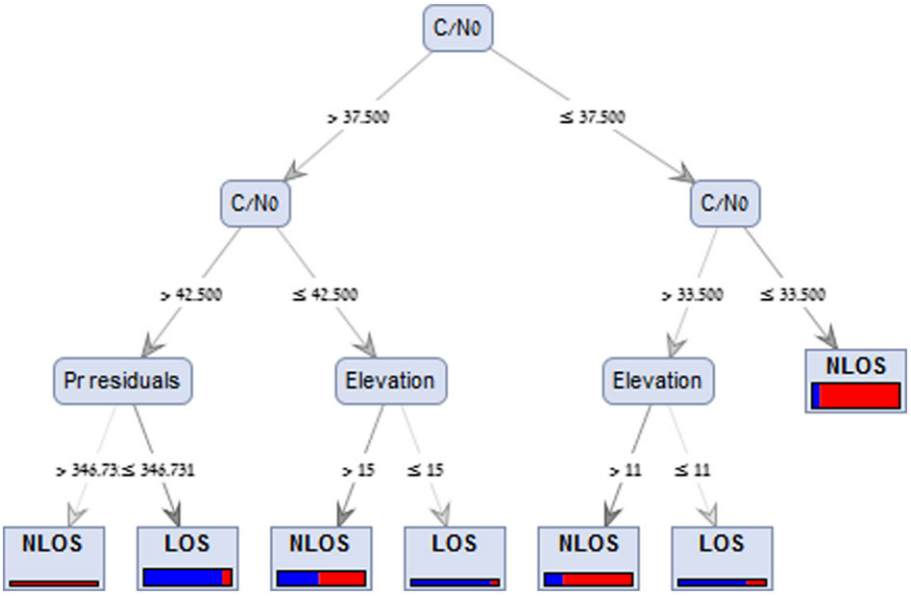

Fig. 10–A decision tree diagram of the algorithm. The image was fabricated using RapidMiner Software [28]. *[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]*

information gain), which are accepted in the computer science community, it is the optimal tree. The structure of the tree represents the information gain of each feature, i.e., the classification ability using that feature solely. One can see the *C/No* is still the most important feature, hence the first. However, the pseudorange residuals and the elevation also play a part in the model. The blue and red represent the ratio of *LOS* and *NLOS* samples (respectively) in the final node. A majority vote is applied to determine the node's *LOS*/*NLOS* state. The ratio between the bars in each node represents the node's confidence, where a 1 : 1 ratio means the lowest confidence. For example, in the binary tree depicted in Figure 9, the *NLOS* node has more confidence than the *LOS* node, i.e., there are more *NLOS* mis-classifications (NLOS signals with C/N_0 of 37.5 or more) than *LOS* mis-classifications (*LOS* signals with C/N_0 of 37.5 or less). As stated above, it is the fusion of those measurements that makes it viable. A detailed discussion on the various aspects of binary trees, including optimal methods for choosing the feature order, can be found in [23].

Over Fitting Avoidance/Generalizing the Algorithm

More often than not, classifiers perform suspiciously well on the training data (over 97 percent accuracy). This suspicion is justified and usually referred to as over-fitting the problem, a property one seeks to avoid. Several methods are known to face this problem. One can teach the algorithm, using only 90 percent of the data and use the remaining 10 percent for testing. Another approach is to use cross-validation [23]. The problem with *GNSS* raw data is that each recording is characterized by some sort of 'water mark' (e.g., the location bias). Implementing cross validation on data obtained solely from a single recording, does not simulate real life. Therefore, in order to generalize the classifier, three orthogonal approaches were taken. First, three separate measurements (different time and different location) were combined to create the learning/training set while a different unseen measurement was chosen to test the algorithm.

Second, we have restricted the decision tree depth. A depth 4 tree is, by its nature, more general than a depth 10 tree. Enforcing a small depth tree almost always produces more general results [23]. Third, a lower bound was put in place for the minimum instances in each final decision node and for the minimal number of instances for splitting the tree. Indeed, each final node contains more mis-classification. The classifier, however, is far more robust than a complex tree without minimum instance bound.

RESULTS

Four different locations and several pedestrian routes were recorded by the receivers. Each recording was conducted twice. This section presents both the stationary and dynamic/moving results. The challenge is the same: given a set of unlabeled raw *GNSS* data measurements, can the algorithm optimally classify those to *LOS* and *NLOS* and how reliable is the confidence factor. By optimal, one means minimum false positive, false negative errors.

The chosen location was the Ramat-Gan Diamond Exchange district, a famous Israeli location for *GNSS* testing. Several *GNSS* data recordings, both stationary and dynamic, were conducted in different places including a walking scenario. The 3*D*

building map of the region is well known, hence each satellite can be geometrically classified to either *LOS* or *NLOS*. Several *GNSS* receivers were utilized, all of them commercial. The lions share of the analysis was done with the data obtained from an ORG1208 SIRF receiver by OriginGPS. This is a commercial receiver for smartphones. The receiver enables relatively easy access (via its binary One Socket Protocol) to most of the raw data (including pseudorange, carrier phase etc.). The decision tree was then tested on an unseen data set (e.g., location D). The tree can be seen in Figure 10. Several observations can be deduced from this tree.

Stationary Results and Field Experiments

As stated throughout the paper, this work focuses on real world *GNSS* raw data recording in dense urban regions. We prefer the mess of reality over the neatness of theory.

Ramat-Gan Diamond Exchange district is characterized by straight floor lines which makes it easy to reconstruct the exact position using Google Earth (Figure 11). The duration of each recording was approx 15 min. A warm reset was performed before each such recording. The time interval between those recordings was 6 months. Over 24,000 instances were gathered. As stated above (Over Fitting avoidance/Generalizing the algorithm Section), one method to steer away from over-fitting algorithms is to use a completely different data set for training (constructing a decision tree) and for testing its performance. In our experiments, three locations constructed the training set (*A-B-C*, *A-B-D*, *B-C-D*, *A-C-D*) while the remaining location served as the testing set. In order to make the testing and training data sufficiently independent, they were gathered in different locations and on separate days. Assuming any probability above 50 percent is labeled as *LOS* and any probability below 50 percent is labeled as *NLOS*, the accuracy rate ranges between 80-90 percent. This is the most restricted assumption since it forces the algorithm to classify all of its instances. In reality, some of the measurements can be bypassed to get better results. One must remember that a simple *C/No* threshold produced an accuracy rate of approx 70-75 percent. Ours is neither a small win nor a colossal win. The Decision tree algorithm slightly outperforms simple *C/No*/ Gaussian classification. However, a well constructed decision tree produces other valuable information that cannot be obtained otherwise - a confidence value. This value can be thought of as the belief one has in the *LOS*/*NLOS* classification. The computation of this value is described in the next section.

Confidence Analysis

The figures reported in this section demand an additional clarification. What is the meaning of being 85 percent right? One can feel intuitively this is a good result and it certainty outperforms a binary *C/No* threshold. However, the correct classification rate alone can draw an incomplete picture (similar to an average without a standard deviation). A more comprehensive depiction of the error rates is discussed here. Another important figure is the

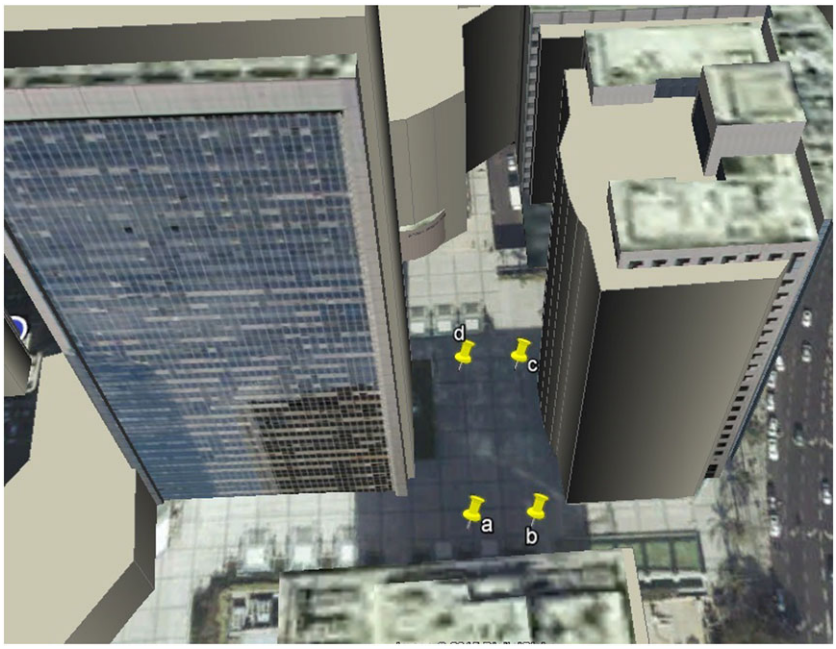

Fig. 11–The stationary experiment scenario. One can easily track the exact position with the aid of the floor lines. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

false-positive, false-negative errors. Assuming the algorithm reports *NLOS* for all the satellites, one gets a terrific *NLOS* error rate since all the *NLOS* signals are classified correctly. Alas, in that scenario, *none* of the *LOS* signals is classified correctly. Thus, this case does not support the supremacy of the suggested method. Therefore, we seek scenarios where the algorithm states *LOS* (or *NLOS*) and we want to test the correctness of that statement. Doing this is important from another perspective. Since the algorithm does not report just a binary result (*LOS* or *NLOS*) but also a confidence, one can decide not to decide. There are several methods to extract a confidence figure from a decision tree. The easiest is to inspect the final node.

The confidence value of a final decision node is defined as the number of correct node classifications divided by the number of instances in that node. For example, a *LOS*/*NLOS* confidence value of 100 pecent is represented in a whole blue/red bar, while a 50 percent confidence value is represented by an equally blue-red ratio bar. A more delicate approach would be to estimate the probability space within each final node and base the confidence value on that probability. Alas, such probability space (for each final node) is hard to obtain. Thus, a more practical approach is to detect low confidence final nodes and to perform additional clustering (such as EM) over them. In essence, the confidence value enables us to eliminate borderline results. Since the algorithm output can be served as an input for another location algorithm, the power of eliminating borderline results is crucial.

Another important observation (and Figure 5 clearly shows it) is that *LOS* detection is simpler than *NLOS* detection (given $C/No \geq 40$ *LOS* is much more likely), hence, any confidence analysis must contain false-positive/false-negative error analysis for both *LOS* and *NLOS*. Figure 12 demonstrates both *LOS* and *NLOS* error rates. A *LOS* error rate is the number of times that we detect *LOS* but the actual value was *NLOS*, divided by the number of times that we detect *LOS*. *NLOS* error rate is defined accordingly. The x-axis in the figure is the percentage from the number of samples, sorted by their likelihood. What can be learned from the graph? First, for the upper 20 percent likelihood samples, the *LOS*/*NLOS* error rates are 0 and 15 percent, respectively. Second, although *NLOS* measurements are more common, the numbers on the figures show the *relative* error in *LOS* and in *NLOS* measurements. Third, considering all data, the *LOS* accuracy rate is approximately 85-90 percent while the *NLOS* is approximately 65 percent. However, if we eliminate the lower 50 percent likelihood samples, the *LOS* accuracy rate rises to 95 percent and the *NLOS* to 80 percent.

Figure 13 shows *LOS*/*NLOS* error rates generated by the EM algorithm. The reason we prefer this

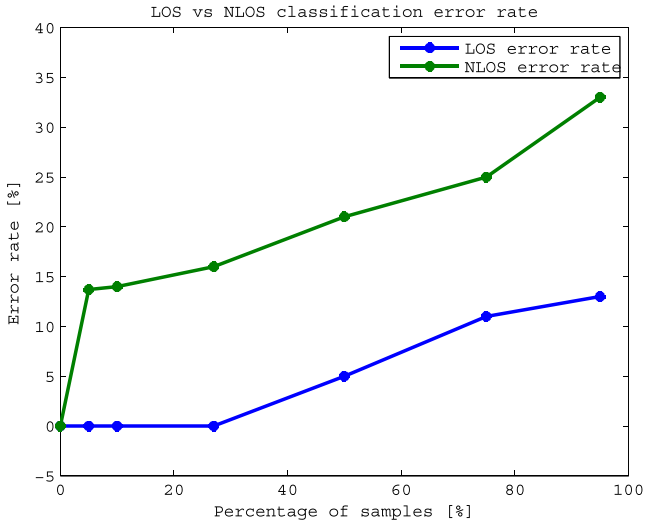

Fig. 12–Classification error rates using a decision. Note that by taking the records with the top 50 percent likelihood score and discarding the rest, we can achieve very low error rates- 5 percent for LOS and 17 percent for NLOS. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

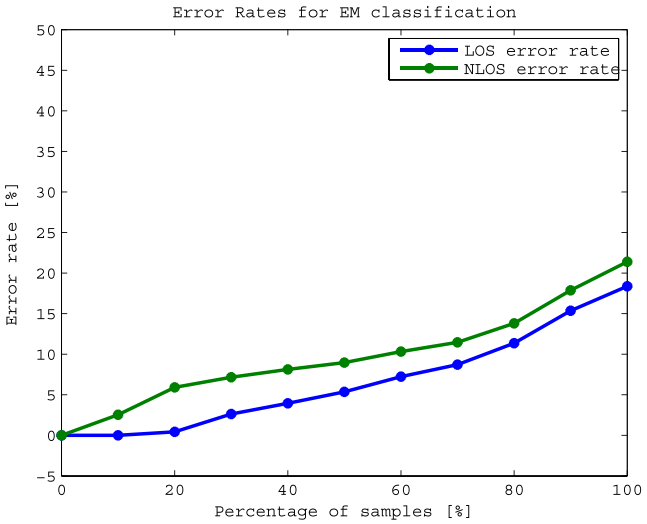

Fig. 13–Classification error rates using EM, with only the C/No. Note the rapid increase in NLOS error rates. This is far worse than the performance of the decision tree (see Figure 12). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

algorithm over a *C/No* threshold is that EM produces the probability function, hence, the confidence of each sample.

Table 1 shows the average classifier output figures for the supervised decision tree (C4.5), the unsupervised EM, and a simple *C/No* threshold. One can observe that the suggested methods outperform the binary threshold approach.

Dynamic Results

In addition to stationary scenarios, a pedestrian scenario was recorded. Both scenarios are depicted in Figure 14. The trajectory is shown in Figure 14 as

Table 1 — Classifiers' statistics

Actual State	Decision Tree		Expectation Maximization		Simple C/No Threshold	
	Positive (LOS)	Negative (NLOS)	Positive(LOS)	Negative (NLOS)	Positive (LOS)	Negative (NLOS)
LOS	77.6%	22.4%	81%	19%	55%	45%
NLOS	12.8%	87.2%	22%	78%	13.8%	86.2%

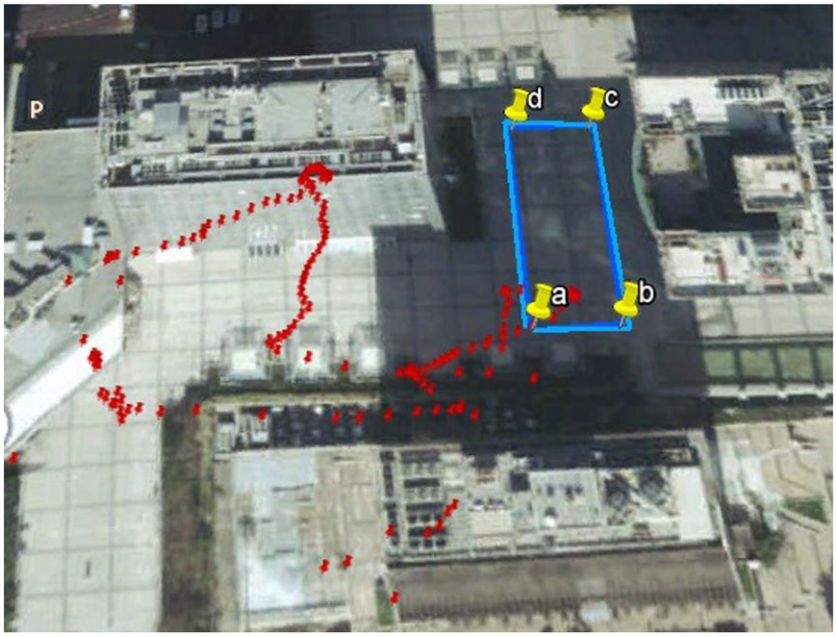

Fig. 14–The pedestrian experiment scenario. The double A-B-C-D square. Route as computed and reported by the GNSS receiver. No correlation exists between the two. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

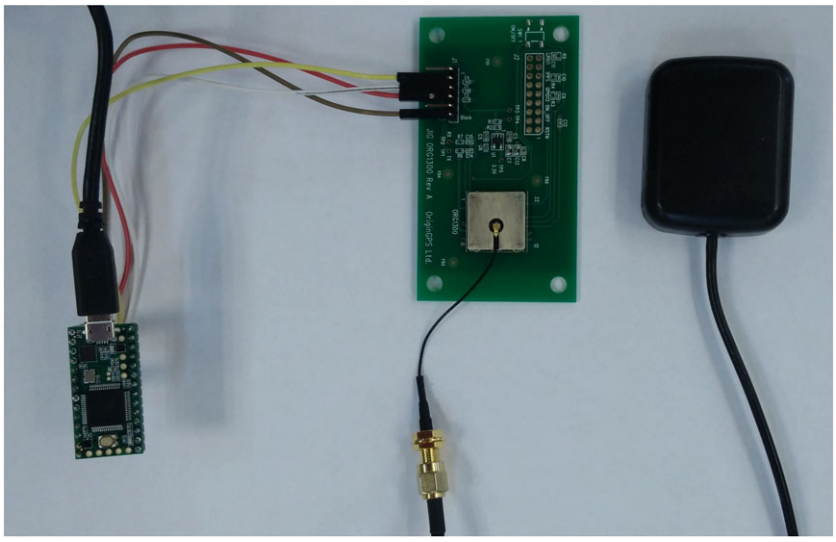

Fig. 15–A prototype of a LOS/NLOS classifier system. One can see an EVK by OriginGPS attached to a Teensy micro Controller. The system reports in its NMEA output an additional LOS sentence. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com and www.ion.org]

a double *A-B-C-D* square. The red tacks in this figure represent the *GNSS* receiver's *reported* positions. Moreover, the position of some of the tacks cannot be seen in the picture (100 *m* error and above). Almost no correlation exists between the conducted route and the reported one.

IMPLEMENTATION

Incorporating LOS *Data in the NMEA Format*

As explained in the introduction, a robust approach for addressing the *LOS/NLOS* classification problem is of great value for several applica-

tions such as accuracy improvement and mapping in urban regions and RAIM. The previous section proved that such (statistical) classification is viable, hence, any *GNSS* processor can compute a *LOS* probability for each tracked satellite. Since the computation itself is both transparent and irrelevant to the user who only seeks the final classification (and confidence) result, reporting that information via the serial protocol (e.g., NMEA) is of great value. In order to demonstrate this concept, we connected the *GNSS* module via serial communication to a micro Controller (e.g., Arduino based Teensy [29]). The micro-controller receives the *GNSS* raw data, calculates a *LOS* probability for each satellite, and produces an additional NMEA sentence containing the classification. The raw data itself can be obtained either from a proprietary NMEA message (e.g., TESEO II by STM) or another interface protocol (e.g., Sirf [30]). A confidence value of 100 indicates maximum *LOS* certainty, 0 confidence value indicates maximum *NLOS* certainty, and a value of 50 represents a borderline point. An optional syntax can be

$$\$GPLOS, \langle Number\_of\_Sats\rangle, \langle PRN_1\rangle, \langle Confidence\rangle, \langle PRN_2\rangle, \langle Confidence\rangle, \langle PRN_3\rangle, \langle Confidence\rangle ... \quad (4)$$

Here, we use *GPLOS* for both *GPS* and *GLONASS* satellites. Such a syntax produces the following output:

$$\$GPLOS, 6, 4, 89, 12, 90, 13, 50, 70, 15, 72, 12, 78, 95 \quad (5)$$

In the above example, six satellites are classified. PRN 4 is *NLOS* with probability of 89 percent; PRN 12 is *LOS* with 90 percent probability; PRN 13 is *LOS* with 78 percent probability, etc..

The assembled system is depicted in Figure 15. We strongly believe this added data to the NMEA protocol can be very beneficial to many *GNSS* application developers.

CONCLUSION AND FUTURE WORK

In this paper, we have introduced a robust approach for the *LOS*/*NLOS* signal classification challenge, mainly for dense urban scenarios. We have shown the proposed algorithm outperforms more conventional methods (mainly simple *C/No* threshold). The robustness was demonstrated by using different stationary and dynamic *GNSS* recordings. For future work, we plan the following. First, incorporate filtering analysis within the classification processes (i.e., the reported classification from the previous second.) Second, add real-time learning ability for the algorithm, thus, enable it to adapt itself to various locations and scenarios. Third, construct a reliable *LOS*/*NLOS* hardware classifier with learning abilities and finally, implement an *SM* accuracy improving algorithm based on an advanced particle filter which uses the suggested *LOS*/*NLOS* classification framework presented here.

Appendix - Releasing the experiments' database

It is very common in the realm of machine learning to publish the data utilized for the paper itself. Furthermore, papers in those disciplines often refer to well known free data-bases (image, bio, finance, etc). Finding a raw *GNSS* measurements database, however, is rare. Therefore, in accordance with the 'expanding the knowledge' academic approach, the database (*GNSS* measurements) along with the *LOS*/*NLOS a priori* classification was uploaded to the Kinematic and Computational Geometry lab website [31]. The files cover all of the stationary points (A-B-C-D), along with the ABCD route. The reader is encouraged to inspect the data for either recreational or pure academic purposes.

REFERENCES

1. Groves, P., "Shadow Matching: A New GNSS Positioning Technique for Urban Canyons," *Journal of Navigation*, Vol. 64, No. 03, 2011, pp. 417–430.
2. Ben-Moshe, B., Elkin, E., Levi, H., and Weissman, A., "Improving Accuracy of GNSS Devices in Urban Canyons," *CCCG*, Toronto, 2011.
3. Yozevitch, R., and Ben-Moshe, B, "A Robust Shadow Matching Algorithm for GNSS Positioning," *NAVIGATION*, Vol. 62, No. 2, Summer 2015, pp. 95–109.
4. Wang, L., Groves, P. D., and Ziebart, M. K., "Smartphone Shadow Matching for Better Cross-Street GNSS Positioning in Urban Environments," *Journal of Navigation*, Vol. 68, No. 3, 2015, pp. 411–433.
5. Wang, L., Groves, P. D., and Ziebart, MK, "GNSS Shadow Matching: Improving Urban Positioning Accuracy Using a 3D City Model with Optimized Visibility Scoring Scheme," *NAVIGATION*, Vol. 60, No. 3, Fall 2013, pp. 195–207.
6. Yozevitch, R., Ben-Moshe, B., and Levi, H, "Breaking the 1 Meter Accuracy Bound in Commercial GNSS Devices," *27th Convention of the Electrical and Electronics Engineers in Israel (IEEEI)*, Eilat, Israel, 2012, pp. 1–5.
7. Irish, A. T., Isaacs, J. T., Quitin, F., Hespanha, J. P., and Madhow, U., "Belief Propagation Based Localization and Mapping Using Sparsely Sampled GNSS SNR Measurements," *IEEE International Conference on Robotics and Automation (ICRA),*Hong Kong, 2014, pp. 1977–1982.
8. Weissman, A., Ben-Moshe, B., Levi, H., and Yozevitch, R., "2.5D Mapping Using GNSS Signal Analysis," *10th Workshop on Positioning Navigation and Communication (WPNC), IEEE*, Drezden, Germany, 2013, pp. 1–6.
9. Obst, M., Bauer, S., and Wanielik, G., "Urban Multipath Detection and Mitigation with Dynamic 3D Maps for Reliable Land Vehicle Localization," *Proceedings of IEEE/ION PLANS 2012*, Myrtle Beach, SC, April 2012, pp. 685–691.

10. Peyraud, S., Bétaille, D., Renault, S., Ortiz, M., Mougel, F., Meizel, D., and Peyret, F., "About Non-Line-of-Sight Satellite Detection and Exclusion in a 3D Map-Aided Localization Algorithm," *Sensors*, Vol. 13, No. 1, 2013, pp. 829–847.
11. Hsu, L.-T., Gu, Y., and Kamijo, S., "3D Building Model-Based Pedestrian Positioning Method Using GPS/GLONASS/QZSS and its Reliability Calculation," *GPS Solutions*, Vol. 20, 2015, pp. 1–16.
12. Groves, P. D., Jiang, Z., Wang, L., and Ziebart, M.K., "Intelligent Urban Positioning using Multi-Constellation GNSS with 3D Mapping and NLOS Signal Detection," *Proceedings of the 25th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2012)*, Nashville, TN, September 2012, pp. 458–472.
13. Bradbury, J., Ziebart, M., Cross, P., Boulton, P., and Read, A., "Code Multipath Modeling in the Urban Environment Using Large Virtual Reality City Models: Determining the Local Environment," *Journal of Navigation*, Vol. 60, No. 01, 2007, pp. 95–105.
14. Jiang, Z., and Groves, P. D., "NLOS GPS Signal Detection Using a Dual-Polarization Antenna," *GPS Solutions*, Vol. 18, No. 1, 2014, pp. 15–26.
15. Meguro, J.-I., Murata, T., Takiguchi, J.-I., Amano, Y., and Hashizume, T., "GPS Multipath Mitigation for Urban Area Using Omnidirectional Infrared Camera," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 10, No. 1, 2009, pp. 22–30.
16. Ramalingam, S., Bouaziz, S., Sturm, P., and Brand, M., "Geolocalization Using Skylines from Omni-Images," *IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops)*, Kyoto, Japan, 2009, pp. 23–30.
17. Waze Inc. http://www.Waze.com [Accessed on April 12, 2016].
18. Holte, R. C., "Very Simple Classification Rules Perform Well on Most Commonly Used Datasets," *Machine Learning*, Vol. 11, No. 1, 1993, pp. 63–90.
19. Kaplan, E., and Hegarty, C., *Understanding GPS: Principles and Applications,* Norwood, MA: Artech House Publishers, 2006.
20. Groves, P. D., and Jiang, Z., "Height aiding, C/No Weighting and Consistency Checking for GNSS NLOS and Multipath Mitigation in Urban Areas," *Journal of Navigation*, Vol. 66, No. 05, 2013, pp. 653–669.
21. Bishop, C. M., *et al., Pattern Recognition and Machine Learning,* Vol. 1, New York: Springer, 2006.
22. Cherkassky, V., and Mulier, F. M., *Learning from Data: Concepts, Theory, and Methods,* Hoboken, NJ: John Wiley & Sons, 2007.
23. Witten, I. H., and Frank, E., *Data Mining: Practical Machine Learning Tools and Techniques,* San Fransisco, California: Morgan Kaufmann, 2005.
24. Mitchell, T. M., *Machine Learning,* Vol. 45, Burr ridge, IL: McGraw Hill, 1997.
25. Mohri, M., Rostamizadeh, A., and Talwalkar, A., *Foundations of Machine Learning,* London: MIT Press, 2012.
26. Tian, X.-F., Lu, Q.-Y., and Xiong, C., "Design of Tilt-Sensor Based on Accelerometer [j]," *Chinese Journal of Sensors and Actuators*, Vol. 2, 2006, p. 021.
27. Pedley, M., "Tilt Sensing Using a Three-Axis Accelerometer," *Freescale Semiconductor Application Note*, Vol. 1, 2013, pp. 2012–2013.
28. Akthar, F., and Hahne, C., "Rapidminer 5 Operator Reference," *Rapid-I GmbH*, 2012. https://rapidminer.com/wp-content/uploads/2014/10/RapidMiner-5-Operator-Reference.pdf [Accessed on April 12, 2016].
29. Stoffregen, P. J., and Coon, R. C., "Teensy USB Development Board," Code Library, 2014.
30. House, C., and Park, C. B., "Sirfstariv One Socket Protocol Interface Control Document," 2009.
31. KCG Lab, raw measurements. online, Available: https://reli1536.wix.com/au-kcg-lab/ [Accessed on April 12, 2016].