In: 2022 IEEE/AIAA 41st Digital Avionics Systems Conference (DASC), Portsmouth, Virginia, 18-22 September 2022
DOI: 10.1109/DASC55683.2022.9925850

Multipath Detection from GNSS Observables Using Gated Recurrent Unit

Patrick Geragersian
School of Aerospace, Transport, and Manufacturing
Cranfield University
Bedford, United Kingdom
Email: Patrick.Geragersian@Cranfield.ac.uk

Ivan Petrunin
School of Aerospace, Transport, and Manufacturing
Cranfield University
Bedford, United Kingdom
Email: I.Petrunin@Cranfield.ac.uk

Weisi Guo
School of Aerospace, Transport, and Manufacturing
Cranfield University
Bedford, United Kingdom
Email: Weisi.Guo@Cranfield.ac.uk

Raphael Grech
Spirent Communications Plc
Crawley, United Kingdom
Email: Raphael.Grech@Spirent.com

Abstract*— One of the most used Position, Navigation, and Timing (PNT) technology of the 21st century is Global Navigation Satellite Systems (GNSS). GNSS signals are affected by urban canyons that limit Line-Of-Sight (LOS) and increase position ambiguity. Smart cities are expected to adopt autonomous Unmanned Aerial Vehicles (UAV) operations for critical missions such as the transportation of organs that are time-sensitive. Therefore, techniques to mitigate Non-Line-Of-Sight (NLOS) interference are required for improved positioning accuracy. This paper proposes a Gated Recurrent Unit-based (GRU) multipath detection algorithm that uses pseudorange, ephemerides, Doppler shift, Carrier-To-Noise Ratio (C/N0), and elevation data from each satellite to determine whether multipath is present. Signals from the satellite classified as multipath are then flagged and ignored for Position, Velocity, and Timing (PVT) calculations until they are deemed as LOS. The classification algorithm is developed and tested on Spirent GSS7000 to generate GNSS Radio Frequency (RF). OKTAL-SE Sim3D is used to simulate urban canyon environments in which signals propagate from the satellite to the receiver. RF signals are then transmitted to a Ublox F9P GNSS receiver that can receive GPS and GLONASS signals which are processed to output PVT information. The data collected is used to train the GRU to classify received signals as no multipath or multipath. From performance evaluation, GRU outperforms decision tree, K-Nearest Neighbor (KNN) classifiers, and Support Vector Machines (SVM). Furthermore, comparing GRU with SVM, a 50% increase in accuracy is observed with a 95% error of 0.85 m for GRU compared to 1.78 m for SVM.

Keywords—Multipath, GRU, GNSS, Machine Learning

INTRODUCTION

Global Navigation Satellite System (GNSS) is one of the most used Position, Navigation, and Timing (PNT) technology of the 21st century. Since the launch of the Global Positioning System (GPS) in 1983, GNSS has been used in a variety of applications and devices such as navigation for vehicles both on the ground and in the air. [1] [2] Currently, six GNSS systems exist. All these systems work with similar principles to provide positioning information to the receivers. Focusing on GPS, for the navigation message to travel from the satellite to the receiver, a carrier wave is used, and the messages are modulated onto those carrier waves. These waves have a frequency of 1575.42MHz (L1) and 1227.60MHz (L2). Three satellites are required as a minimum to provide positioning information for a receiver [3].

Various GNSS measurement and error sources exist during the process of generating the signal to receiving, decoding, and calculating the position of a receiver. These include satellite ephemeris errors, clock errors (both at transmission and the receiver), ionospheric/tropospheric delay, multipath, spoofing, spamming, and receiver noise to mention a few [4]. The majority of these errors can be modeled to estimate their impact on the pseudorange and emitted. However, one of the biggest problems in urban canyons is GNSS navigation systems affected by multipath [5].

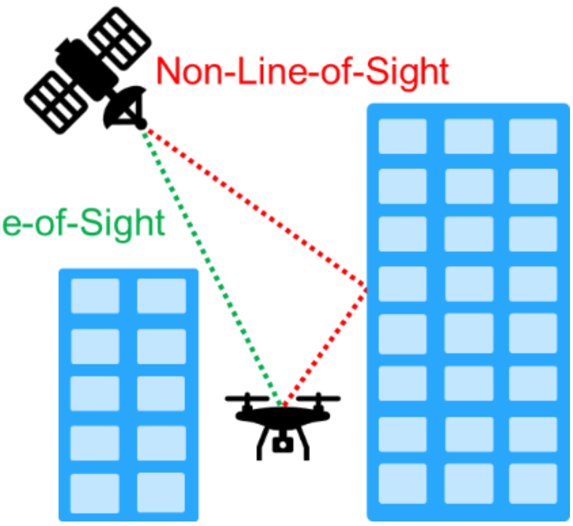


Figure 1. Line-of-Sight and Non-Line-Of-Sight illustration in Urban Canyons

Multipath is a term used to describe both the issue related to Non-Line-of-Sight (NLOS) signals and multipath interference where two signals of the same satellite are received at the same time. One of them is the direct Line-Of-Sight (LOS) path to the receiver whilst the other one has taken an indirect route [6]. This is shown in Figure 1. Multipath causes pseudorange measurement errors that lead to inaccurate positioning information provided to the navigation system which results in navigation errors. This is especially true for autonomous vehicles which rely on GNSS to transit through dense populations. Therefore, there is a need to mitigate and reduce this error wherever possible [7].

Existing techniques vary in their approach to this issue which can be categorized as antenna and receiver design, weight model, signal processing, image processing, consistency checking, mapping-aided, and statistical approaches [6]. In this paper, we will be focusing on GNSS modules available to consumers and therefore limit the depth of this paper to methods that can be applied after signal processing. One of the most common techniques used to check on GNSS integrity is Receiver Autonomous Integrity Monitoring (RAIM). RAIM supplies integrity monitoring for GNSS Fault Detection (FD). It requires the pseudorange measurements of a minimum of five satellites for FD or a minimum of six satellites for Fault Detection and Exclusion (FDE). The range and position errors modeled for RAIM are carried out in an open sky environment. Therefore, the performance is degraded in urban canyons with the majority of pseudorange measurements either not being available or experiencing multipath effects [8].

Another existing method used includes 3D environment model mapping used to estimate the possible magnitude of pseudorange errors or to help provide alternative navigation routes to autonomous systems. This method requires a 3D map of the environment to examine the areas, depending on the time of day, for the number of available satellites with LOS access to the receiver of the vehicle. However, these 3D environments may be outdated when compared with the real world. Additionally, this may not aid in detecting NLOS signals when large vehicles are present on the ground, blocking the direct path to the receiver. Lastly, alternative navigation routes may not always be possible as the destination might be in an area where most satellites are not in LOS [9].

Recently, statistical approaches are being researched as a possible solution in FDE of multipath pseudorange measurements. Paper [10] uses a k-means cluster approach for multipath detection. The algorithm uses a set of features such as carrier phase, pseudorange, and carrier-to-noise ratio to categorize into two groups. These groups are multipath and no multipath. From simulations, it has been shown to provide improvements over RAIM when multiple pseudorange measurements are affected by multipath. However, the author notes that more analysis is required on the selection of parameters that affect the performance of the suggested algorithm. Furthermore, the carrier-to-noise ratio for detection of LOS signals may change depending on the location of the receiver and the elevation of the satellite.

Another recent statistical approach from the author [11] uses Support Vector Machines (SVM) to classify the pseudorange measurements into three distinct categories, clean, multipath, and NLOS. The classification is carried out using four features including received signal strength, change of rate of Received Signal Strength (RSS), pseudorange residue, and pseudorange rate. The experiment results show a classification accuracy of 75%. An advantage of using SVM is the capability to handle non-linear datasets. The disadvantage is those past error dependencies are not considered. Moreover, large datasets slow down SVM solutions.

Author [12] proposes multipath detection by continuously monitoring the Carrier-To-Noise Ratio (C/N0) of incoming signals to detect NLOS. Using elevation to determine the C/N0 threshold, the author's algorithm was able to judge whether the signal experienced Multipath. Furthermore, to overcome the issues of the signal fluctuating between the threshold, continuous time-series C/N0 were investigated to put a time requirement for the signal to be above the threshold before it can be used again for positioning. The test results show a decrease in the 90th percentile of horizontal positioning error from 37.8 m to 4.31 m. The tests were conducted in multiple different environments with different urban landscapes. Horizontal Root Mean Squared Errors (RMSE) also showed a decrease in positioning errors from 12.47 m to 2.23 m. However, the tests conducted in this paper are all based on static antennas. Therefore, the thresholds are not re-evaluated and may not be suitable for moving vehicles. Furthermore, whilst it does use a continuous-time series to provide a minimum time required for the C/N0 to be above the threshold, it does not consider past error dependencies and therefore a minimum time that is set for all scenarios might not be suitable everywhere.

Given all the above research, this paper proposes a method of detecting and excluding GNSS signals from multiple constellations that are affected by multipath using Gated Recurrent Units (GRU) that classifies those signals as either LOS or affected by multipath. The proposed feature set used consists of pseudorange, ephemeris data such as elevation and satellite position, Doppler shift, and C/N0.

Therefore, the algorithm aims to improve GNSS positioning and velocity estimate by detecting and removing multipath signals. To achieve this aim, the features mentioned above are used as the input of the neural network. The data is then processed using various weights and biases to provide a link between the input features and the output decision. A GRU implementation provides the ability for the neural network to consider past information before providing an output.

In section three, the theory and structure behind the GRU are discussed and explained with an overview of the proposed system architecture. In section four, the testing methodology and analysis are described. Section five is a discussion and evaluation of the test results.

GATED RECURRENT UNITS

RNN, GRU & LSTM comparison

Recurrent Neural Networks (RNN) are a class of artificial neural networks that are utilized for sequence prediction challenges. Derived from feedforward neural networks by David Rumelhart [13], it uses the previous state output as the current input and therefore determines a relationship between the output and the input. These neural networks have been used primarily in handwriting and speech recognition in the past with newer development in sensor fusion. The RNN algorithm is shown in equations 1 and 2 [14]:

$$h_t = \sigma_h(W_{hh}h_{t-1} + W_{xh}X_t + b_h), \tag{1}$$

$$y_t = \sigma_y\text{-}W_y h_t + b_y, \tag{2}$$

where h_t is the hidden layer vector, y_t is the output vector, X_t is the input vector, W_{hh}, W_{xh}, W_y are the parameter matrices, b_h, b_y are the bias terms, and σ_h, σ_y are the activation functions. RNNs take the information from the previous state h_{t-1} and multiply it with a weight matrix. The same is done for the new input X_t and is then combined with the previous state to create the new hidden state h_t. Because of the way RNN is structured, it allows for information in the past to be linked with the information at the current timestep.

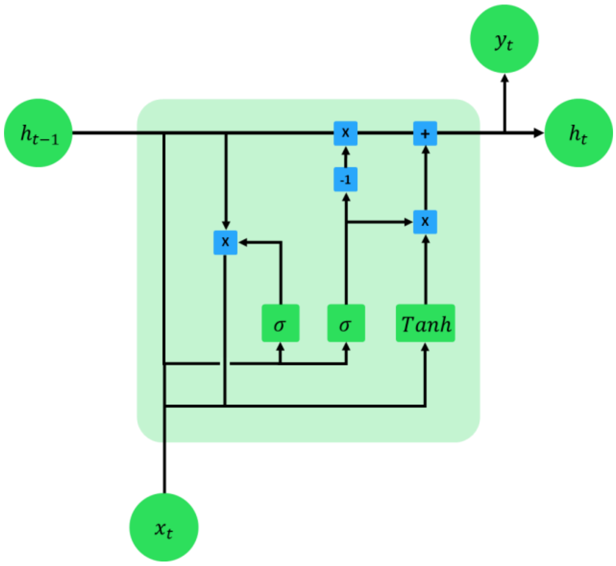


Figure 2. Inside a Gated Recurrent Unit

The activation functions (Tanh) are used to assess the sum weights of the input and decide which information is needed. However, RNNs suffer from vanishing and exploding gradients that may only provide a short-term memory for the past information being recorded. It also means that some information can be propagated even though the information is not useful. Two types that are derived from RNNs (GRU and LSTMs) solve these issues.

LSTMs and GRUs use gates to determine whether the input information should be kept or should be removed. The GRU architecture is shown in Figure 2. In a GRU, two gates deal with this. An update gate is used to help the neural network determine how much information from the previous time step needs to be passed to the next timestep. The formula for this is presented below [15]:

$$z_t = \sigma(W^z X_t + U^z h_{t-1}), \tag{3}$$

The reset gate is used to determine how much of the past information should be forgotten to improve the performance of the GRU. The formula is presented below:

$$r_t = \sigma(W^r X_t + U^r h_{t-1}), \tag{4}$$

These gates aid in removing the issues related to exploding and vanishing gradients. However, the additional gates add computational complexity to the neural network which may slow down the training process. LSTMs on the other hand use three gates to solve these issues. These are the input gate, output gate and a forget gate. Because LSTMs use three gates, they are more computationally expensive than GRUs but can provide more accurate information for training the model.

Proposed Architecture

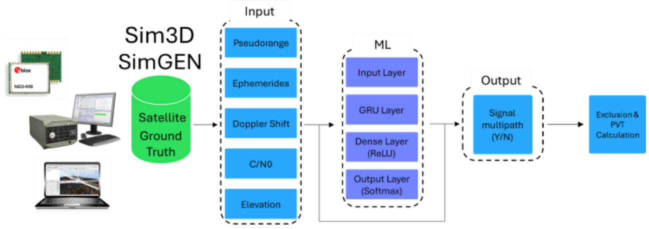


Figure 3. High-level architecture overview

This paper proposes a Gated Recurrent Unit (GRU) classification algorithm that employs a multi-constellation GNSS system using pseudorange, ephemerides, Doppler shift, C/N0, and elevation data from each satellite to determine whether multipath is present. The proposed architecture is shown in Figure 3. Using a GRU aids in finding nonlinear relationships and provides a method for utilizing past error dependencies for predicting multipath signals. The features mentioned above are used as the input to the neural network. The data is then processed using various weights, biases, and non-linear functions to provide a link between the input features and the output decision. The inputs were chosen based on current literature systems that were investigated as part of the state-of-the-art review. C/N0 was effectively used in [10] to classify NLOS. Elevation-based C/N0 was also reviewed in [12] as an influential aspect in deciding whether multipath was present. Furthermore, Doppler shift and pseudorange were also used in the paper [11] to improve the classification performance in SVM. Each satellite information that is processed by the receiver each second is classified. If the signal is classified as multipath, it is excluded from PVT calculations. To calculate PVT, a minimum of 4 satellites are required with the pseudorange and satellite positions information provided to the MATLAB receiverposition function to determine receiver position. When excluding a multipath signal, the satellite is flagged for a period of one second. Therefore, for one second, that satellite's pseudorange measurement and position are not used to calculate the receiver position.

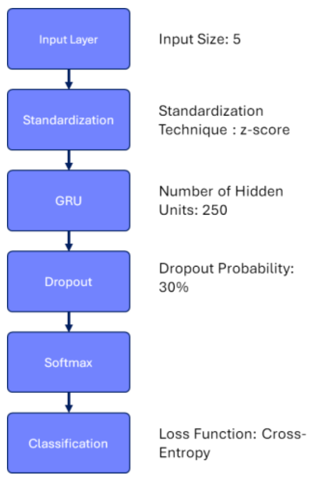


Figure 4. Machine Learning Architecture with optimized parameters

The GRU architecture is shown in Figure 4. Hyperparameter optimization using random search from the

Keras library was applied to optimize learning parameters, number of fully connected layers, activation function, regularization strength, and if standardization will affect convergence speed.

METHODOLOGY

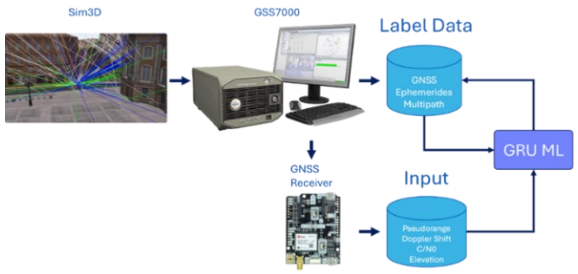


Figure 5. Data flow from simulation and GNSS Receiver to GRU ML for training

To train the Neural Network (NN), multipath data generated by OKTAL-SE Sim3D is used with a 3D environment that resembles Toulouse, France. A drone at 15m height with a rectangular trajectory is used to generate GNSS multipath data. The path used is shown in Figure 6. This information is then provided to the Spirent GSS7000 simulator which generates the radio frequency (RF) signals that are then transmitted to the Ublox F9P GNSS receiver. Both GPS and GLONASS signals are then captured and the C/N0, elevation data, Doppler shift, and pseudorange measurements are stored for processing later. The GSS7000 acquires the raw GNSS signal in a CSV format. This includes information such as the ephemerides and whether the GNSS signal is experiencing multipath. A representation of the data flow is shown in Figure 5. The update rate for both is set at 1 Hz. This data is then pre-processed before being used for training on the NN. The input and output data are split into 80% training data and 20% validation data. The five features used as the input are the pseudorange, ephemerides that consist of the satellite position, Doppler shift, C/N0, and elevation. The two outputs are whether the GNSS signal is experiencing multipath or not. This output is compared to the captured simulator data to determine if the NN was able to predict this correctly. After one thousand iterations, the NN is then tested using the testing data that was not seen before by the architecture.


Figure 6. Ground Trajectory (yellow line) used for generating GNSS data

To evaluate the performance of the proposed architecture, a comparison with existing techniques is carried out. Decision tree, K-Nearest Neighbor (KNN) Classifiers, and Support Vector Machines (SVM) are used to compare against the proposed GRU neural network. Confusion matrices, mean, standard deviation, 95% horizontal absolute error, and path scatters are compared.

RESULTS

Figure 7 shows the feature importance scores sorted using the Maximum Relevance – Minimum Redundancy (MRMR) algorithm. The highest score given to a feature was C/N0. From research, this was to be expected as most multipath mitigation approaches are based on C/N0. The second-ranked one was the elevation. From paper [12], C/N0 based elevation approach was used to predict multipath. Therefore, this also aligns with expectations. Pseudorange, Doppler shift, and ephemerides were ranked at the bottom but are still useful features for determining signals affected by multipath.

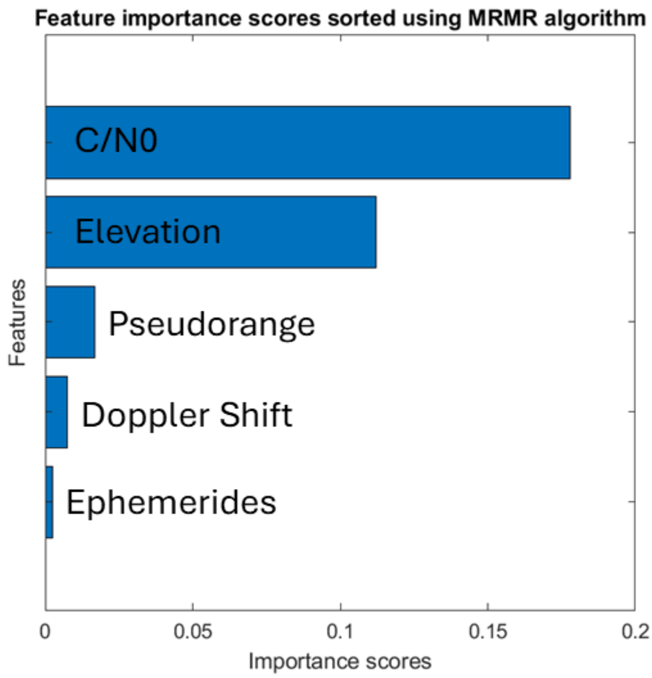
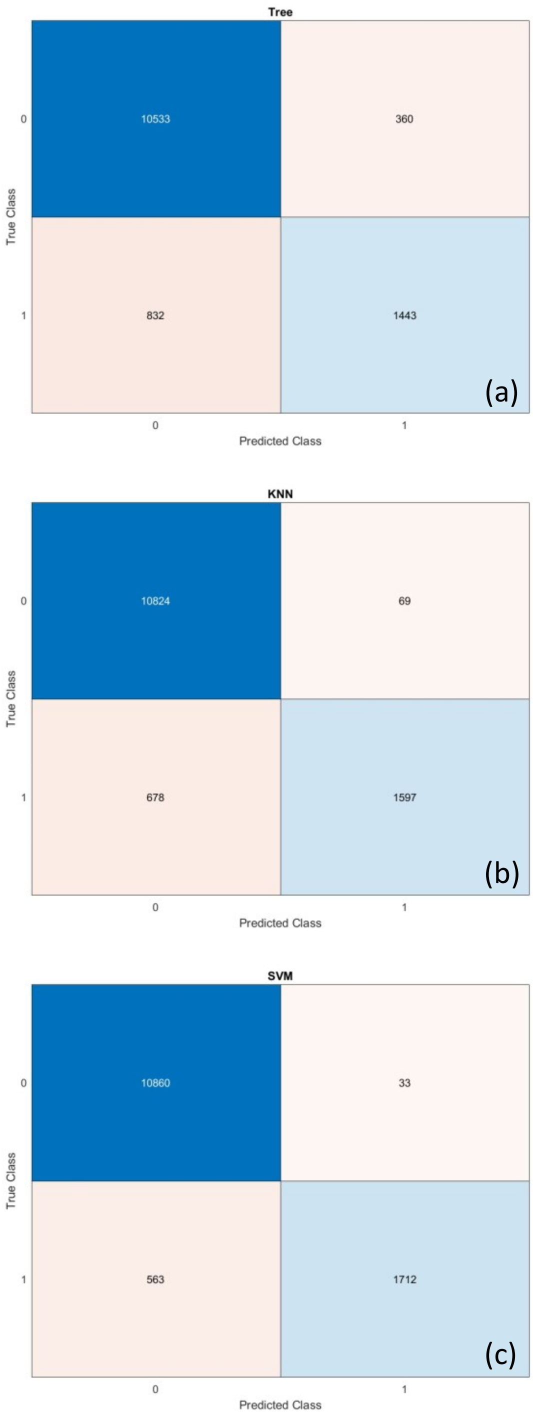


Figure 7. Feature importance scores are sorted using the Maximum Relevance – Minimum Redundancy (MRMR) algorithm (C/N0, elevation, pseudorange, Doppler shift, ephemerides)

Figure 8 shows the confusion matrices after processing the testing data from decision tree, KNN, SVM, and GRU. The decision tree, shown in 8 a), has the worst performance in comparison to the other techniques with a false negative of 832 and a false positive of 360. This is because decision trees are not able to use past information to influence future decisions. They are also overly sensitive to small data changes which would influence the outcome of the output. Furthermore, training decision trees is not the fastest way to train based on the feature inputs and outputs. KNN, shown in 8 b), has a reduced number of errors in comparison to decision tree with a false negative of 678 and a reduced false positive of 69. The main advantages of KNN are the fast learning time. However, with larger datasets or large dimensionality, this makes it difficult for KNN to be able to predict the outputs. SVM, shown in 8 c), further reduces the errors to a false positive of 563 and a false positive of 33. SVM works well when the margin of separation is clear between classes and can provide non-linear relationships.

However, SVM is not suitable for larger datasets which may be required for future training and may not work well with noisy data which is the case with GNSS signals as noises can come from multiple sources from the environment and electrical equipment. Figure 8 d) shows the observation from the GRU. GRU shows a 50% reduction of false negatives in comparison to SVM. GRU can use past information to aid in predicting outputs. However, the training time required for GRU is the largest compared to the other three techniques.



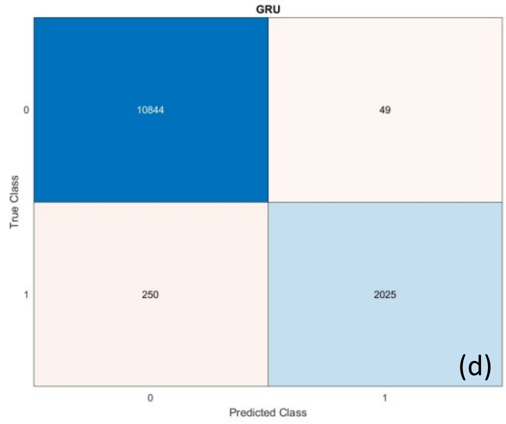
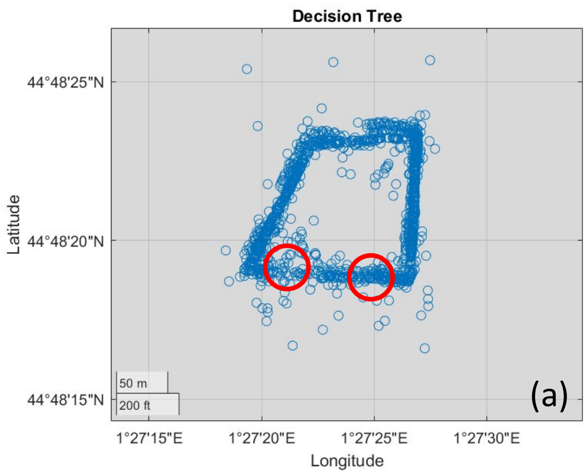


Figure 8. Confusion Matrix for (a) decision tree (b) KNN (c) SVM (d) GRU

Figure 9 shows a Geo-Scatter comparison between decision tree, KNN, SVM, and GRU. SVM has a greater variance in the position estimate than GRU. GRU has more consistency as it can identify the majority of the NLOS signals compared with SVM. Since four satellites are required for 3D positioning, the GRU path plot has some patches (shown in red circles) as a majority of the signals are NLOS. Furthermore, the same area where there are patches present in the GRU path plot, also has the biggest scatter of position estimates for the SVM. This shows that multipath is having a large effect on the accuracy which may lead to navigation issues. This is especially important for autonomous vehicles that might heavily rely on GNSS information as their primary navigation source. Decision trees and KNN also exhibit issues with accuracy as there is a large variance in the estimate positions. Decision tree performs the worst in providing accurate position information as was also the case when comparing the observations. This is shown by the large scatter seen when compared to other techniques. Large scatter in position estimates indicate reduced accuracy. All three classification algorithms (decision tree, KNN, and SVM) struggle in areas where a majority of GNSS signals are experiencing multipath. GRU, on the other hand, is more capable of determining non-linear relationships between the features and using previous state information to provide greater accuracy. This is seen in paper [1], where GRU provided better correlation in cases where the relationship between input features and the outputs was highly non-linear.



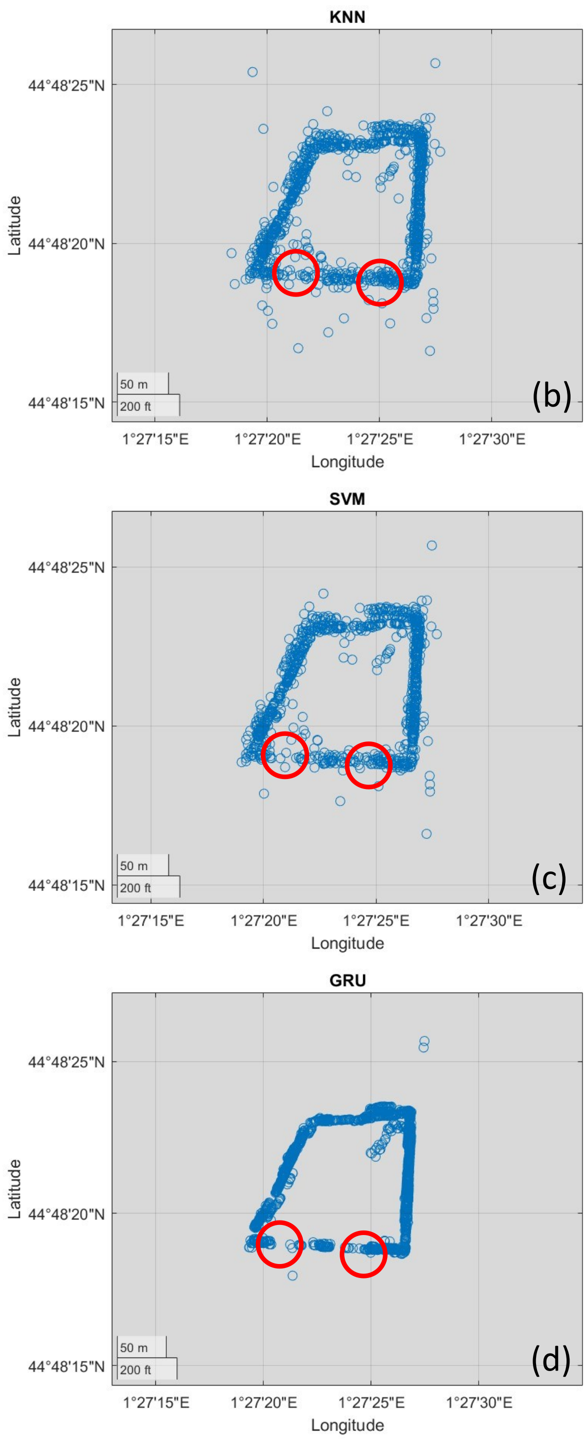


Figure 9. Geo-scatter plot of GNSS signals from decision tree, KNN, SVM, and GRU

Table 1 compares the mean, standard deviation, and 95% between the original output (Ublox F9P), tree, KNN, SVM, and GRU. SVM has a mean of 0.0134 m compared to 0.0111 m for GRU. This is to be expected as both architectures are mitigating some of the NLOS multipath GNSS signals. However, there is a larger difference between the standard deviation of position accuracy using SVM (circa 0.8 m) and GRU (circa 0.4 m). This behavior is similar to the evidence seen in Figure 9, where there is a large variance in position estimates. 95% positioning error for the GRU is more than 50% lower at 0.85 m compared to 1.78 m for SVM. This shows that GRU improves multipath classification accuracy and therefore reduces the position estimate errors. Furthermore, looking at decision tree and KNN, they perform the worst. This is to be expected as seen from the previous comparison.

TABLE I

COMPARISON OF MEAN, STANDARD DEVIATION, AND 95% ABSOLUTE HORIZONTAL ERROR BETWEEN UBLOX RECEIVER PVT OUTPUT, DECISION TREE, KNN, SVM, AND GRU

Architecture	***Mean (m)***	***Standard Deviation (m)***	***95% error***
F9P	0.1	1.2	2.5
Decision Tree	0.065	0.95	1.97
KNN	0.043	0.91	1.86
SVM	0.013	0.85	1.71
GRU	0.011	0.45	0.911

Conclusion and Future Work

The paper aimed to reduce the position error due to multipath GNSS signals that are present in urban canyons. This is important, especially for autonomous vehicles that require accurate navigation in these environments to transport goods or provide services safely and reliably. Existing methods that try to mitigate multipath errors can reduce these effects. However, existing methods have certain limitations or simply do not work in urban canyons. Such systems include RAIM which aims to detect and exclude multipath signals by checking for consistency. However, these systems assume most signals are LOS, which is not the case in urban canyons. Newer systems such as SVM-based multipath detection work well to reduce the effects of multipath by correctly classifying signals that are affected and excluding them. However, SVM has training limitations and cannot utilize past error dependencies to influence current state outputs. Therefore, a GRU-based GNSS multipath mitigation approach was considered to improve positioning estimates by improving multipath classification. Comparing the decision tree, KNN, and SVM with GRU, the latter shows the best performance in multipath identification. Furthermore, comparing SVM and GRU, a 50% reduction in position error is observed with a 95% absolute horizontal error of 0.85 m for GRU compared to 1.78 m for SVM. This shows that GRU can be used to reduce positioning estimate errors by improving multipath classification. This method will aid in enhancing the safety and reliability of autonomous vehicles in urban environments which are prone to these kinds of issues and where safety is a paramount concern. Future work will be focused on implementing this system with other sensors and error reduction systems to improve the overall position estimate. Furthermore, it is of interest to consider evaluating each multipath signal and using a combination of LOS and multipath signals to provide position information in scenarios where there are not enough LOS signals to determine the location.

References

[1] P. Geragersian, I. Petrunin, R. Grech and W. Guo, "An INS/GNSS fusion architecture in GNSS denied environment using gated recurrent unit," in *AIAA Scitech Forum 2022*, San Deigo, 2022.

[2] D. A. Grekner-Brzezinska, C. K. Toth, T. Moore, J. F. Raquet, M. M. Miller and A. Kealy, "Multisensor Navigation Systems: A Remedy for GNSS Vulnerabilities?," *Proceedings of the IEEE,* pp. 1339-1353, 2016.

[3] J. Leclère, R. Landry and C. Botteron, "Comparison of L1 and L5 Bands GNSS Signals Acquisition," *Sensors,* vol. 18, p. 9, 2018.

[4] U. Hugentobler, H. van der Marel and T. Springer, "Identification and mitigation of GNSS errors," in *Position Paper, IGS 2006 Workshop Proceedings*, Darmstadt, 2008.

[5] P. Zabalegui, G. D. Miguel, A. Perez, J. Mendizabal, J. Goya and I. Adin, "A Review of the Evolution of the Integrity Methods Applied in GNSS," *IEEE Access,* vol. 8, pp. 45813-45824, 2020.

[6] N. Zhu, J. Marais, D. Betaille and M. Berbineau, "GNSS Position Integrity in Urban Environments: A Review of Literature," *IEEE Transactions on Intelligent Transportation Systems,* vol. 19, no. 9, pp. 2762-2778, 2018.

[7] X. Wu, J. Zhou, G. Wang, X. Hu and Y. Cao, "Multipath error detection and correction for GEO/IGSO satellites," *Science China Physics, Mechanics and Astronomy,* vol. 55, pp. 1297-1306, 2012.

[8] O. K. Isik, J. Hong, I. Petrunin and A. Tsourdos, "Integrity Analysis for GPS-Based Navigation of UAVs in Urban Environment," *Robotics,* vol. 9, no. 3, p. 66, 2020.

[9] K. Nagai, T. Fasoro, M. Spenko, R. Henderson and B. Pervan, "Evaluating GNSS Navigation Availability in 3-D Mapped Urban Environments," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Portland, 2020.

[10] C. Savas and F. Dovis, "Multipath Detection based on K-means Clustering," in *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019)*, Miami, 2019.

[11] L.-T. Hsu, "GNSS multipath detection using a machine learning approach," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, 2018.

[12] N. Kubo, K. Kobayashi and R. Furukawa, "GNSS Multipath Detection Using Continuous Time-Series C/N0," *Sensors,* vol. 20, no. 14, p. 4059, 2020.

[13] D. Rumelhart, G. Hinton and R. Williams, "Learning representations by back-propogating errors," *Nature,* vol. 323, pp. 533-536, 1986.

[14] M. Jordan, Chapter 25 - Serial Order: A Parallel Distributed Processing Approach, North-Holland, 1997, pp. 471-495.

[15] S. Hochreiter and J. Schmidhube, "Long Short-Term Memory," *Nerual Computer,* vol. 9, no. 8, pp. 1735-1780, 1997.