

A Machine Learning-Based GNSS Performance Prediction For Urban Air Mobility Using Environment Recognition

Dr.Dilip Motwani¹, Dr Vidya Chitre², Dr Varsha Bhosale³, Dr. Swapnil Sonawane⁴, Dr.Amit Nerurkar⁵, Dr.Deepali Vora⁶

^{1,2,3,4,5}Professor, Vidyalankar institute of Technology, Wadala, Mumbai, Maharashtra, India

⁶Professor, Symbiosis Institute of Technology, Pune, Maharashtra, India,

dilip.motwani@vit.edu.in¹, vidya.chitre@vit.edu.in², varsha.bhosale@vit.edu.in³, swapnil.sonawane@vit.edu.in⁴, amit.nerurkar@vit.edu.in⁵, deepali.vora@sitpune.edu.in⁶

Abstract: Urban Air Mobility (UAM) has gotten a lot of attention because it could change the way people get around in cities with lots of people. But getting UAM cars to navigate reliably is still hard, especially since Global Navigation Satellite Systems (GNSS) aren't perfect in these situations. GNSS performance is often hampered by infrastructure in cities, signal blockages, and interference. This paper suggests a new way to predict GNSS success in UAM systems using machine learning and environment recognition methods together. The suggested model takes into account a lot of external factors that affect the quality of GNSS signals, like the number of buildings, plants, temperature, and other barriers. Based on these external factors, machine learning methods like regression models and support vector machines are used to look at and guess how well GNSS will work. GNSS systems built into UAM cars and weather sensors are used to collect data for the method. Both GNSS and weather data are preprocessed with feature extraction methods. This data is then put into machine learning models. Key measures like GNSS accuracy and forecast accuracy in different urban settings are used to judge the performance of the model. A case study shows how the model can be used in a real-life UAM situation, showing how well it improves the accuracy of GNSS performance predictions. The results show that adding environmental detection makes GNSS performance much more reliable in complex urban settings. This study helps UAM systems get better by giving a strong answer to one of the main problems these cars have: making sure they can navigate consistently and correctly. The study also talks about areas that could be studied in the future, such as adding more outdoor factors and better machine learning methods for predicting GNSS in real time.

Keywords: Urban Air Mobility, GNSS Performance, Machine Learning, Environment Recognition, Navigation Prediction

INTRODUCTION

Urban Air Mobility (UAM) could change the way humans get round cities by way of bringing in flying cars that can fly in towns, imparting quick and long-lasting approaches to get around. Unmanned aerial vehicles (UAVs), electric vertical takeoff and landing (eVTOL) planes and other superior airborne transportation structures are getting more popular. UAM claims to ease visitors, reduce down on carbon emissions, and make it simpler for people to get round in crowded cities. However, certainly one of the most important troubles with the use of UAM technologies is ensuring they can locate their manner round reliably, particularly in towns where global Navigation satellite systems (GNSS) work much less nicely. GNSS generation has been very vital in imparting correct area and timing offerings for lots styles of transportation structures, which includes those utilized in aeroplanes, ships, and automobiles on land. in terms of UAM, GNSS is very crucial for steering for vehicles that fly themselves or with little help from humans. However the crowded and complex city placing makes it more difficult for GNSS to paintings nicely in a number of ways. Satellite TV for pc indicators may be blocked or interfered with by way of such things as tall buildings, small streets, a whole lot of infrastructure, and weather that modifications fast. This will lead to less accuracy, sign loss, and much less dependable placement. GNSS signals can emerge as unstable or absent in those varieties of places, which can make UAM sports less safe and more inefficient [1]. On the way to solve this trouble, we need to create a system which could guess how well GNSS will work in cities. This may allow UAVs to modify to changing conditions right away. A vital part of that is surroundings identification, this means that locating and knowing the things inside the environment that exchange signal receiving. Figure 1 show how environmental popularity techniques and statistics can be used to are expecting GNSS overall performance for city air mobility.

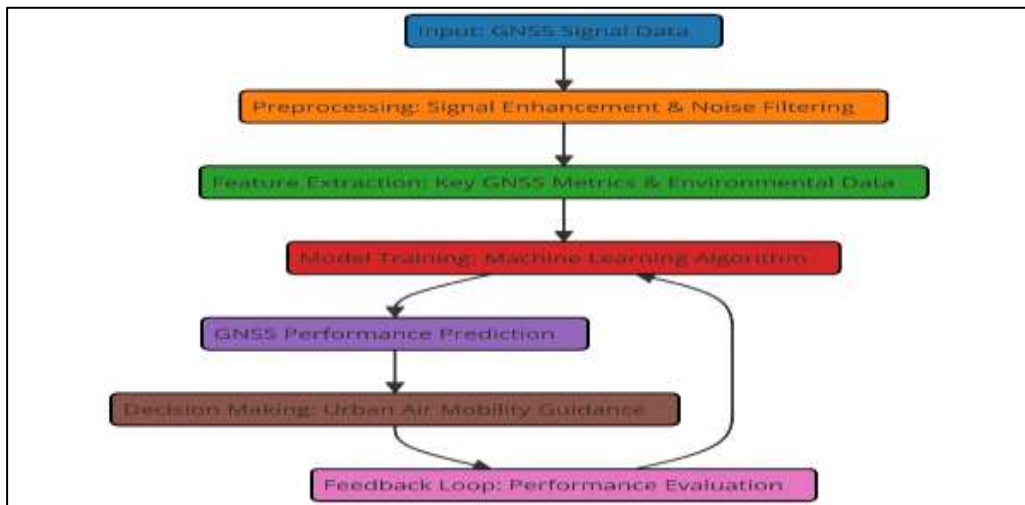


Figure 1: GNSS Performance Prediction for Urban Air Mobility Using Environment Recognition

Environmental elements like building height, city population, greenery, and climate conditions will have a massive effect on how properly GNSS works. Machine learning (ML) fashions can look at huge datasets and find complex trends. They can be an amazing manner to guess how properly GNSS will work based on data from the surroundings. This paper indicates the use of device studying to guess how nicely USA automobiles will do with GNSS by combining strategies for recognizing their environment. The counseled model can bet the first-rate of GNSS signals in diverse town settings through collecting and reading statistics on each the signals themselves and the features of the environment [2]. Adding device getting to know to the system lets it analyze from beyond records, alter to changing conditions within the international, and predict success in actual time. Getting GNSS to paintings better in city airspaces and making UAM structures safer and more effective are the principle goals of this observe. The rest of the paper is organized like this: In Section II, we look at important material that talks about GNSS performance, UAM uses, and forecast models based on machine learning.

RELATED WORK

More and more study is being done on how to combine Global Navigation Satellite Systems (GNSS) with Urban Air Mobility (UAM) technologies because people want safe, self-driving transportation options in cities. A lot of study has been done on how to make GNSS work better, especially in cities that are complicated and always changing, where signal loss from objects and multipath effects is common [3]. Line-of-sight contact with satellites is needed for traditional GNSS devices. However, tall buildings, small streets, and weather conditions can make this very difficult. Many research on these problems have produced recommendations for additional or new technology that could improve GNSS performance in urban areas. To better understand where something is, they have investigated mixed systems using GNSS, ocular odometry, inertial navigation systems (INS), and other sensor-based techniques [4]. These approaches attempt to cover for lost or distorted GNSS signals, but often they need additional hardware, which increases system complexity and expense. Machine learning (ML) models have evolved over the last several years into a fascinating approach to identify GNSS signal issues in localities and address them.

Regression models, support vector machines (SVM), and deep learning approaches—types of machine learning—have been employed in many research to project how effectively GNSS would function depending on surroundings. To estimate errors in signal power and placement, models have been created that resemble cities, with regard to their form and architectural details like building count and height. Various studies have attempted to estimate the real-time accuracy of GNSS communications by use of air and weather considerations [5]. Using convolutional neural networks (CNNs), one may manage outside input for GNSS performance prediction. In this area, this is a quite significant workpiece. These models may be taught by the data how to spot trends in location and time. This helps them to forecast more precisely and adjust to changes in their surroundings. Table 1 summarises the main results, ideas, and efforts of the urban air flow research [6]. Real-time data from the car's surroundings guides reinforcement learning in changing its course. This is an other approach to improve GNSS system performance.

TABLE 1: SUMMARY OF RELATED WORK

Method	Key Finding	Approach	Challenges
GNSS-INS Hybrid Systems	Improved positioning accuracy in urban environments.	Combines GNSS with Inertial Navigation Systems (INS).	Sensor fusion complexity and additional hardware costs.
Environmental Feature Analysis [7]	Environmental factors like building density affect GNSS accuracy.	Uses environmental data to predict GNSS signal strength.	Difficulty in real-time data processing and integration.
Support Vector Machines (SVM)	SVM models can effectively predict GNSS performance based on environment.	Supervised learning model for prediction using GNSS and environmental data.	Requires large labeled datasets for training.
Deep Learning Models (CNNs) [8]	Deep learning models outperform traditional methods in complex environments.	Uses Convolutional Neural Networks (CNNs) for spatial pattern recognition.	High computational cost and need for large datasets.
Reinforcement Learning (RL)	RL can optimize GNSS signal prediction by adjusting flight paths.	Applies RL algorithms to adjust GNSS performance dynamically.	Training RL models can be resource-intensive.
Regression Models	Simple models can predict GNSS accuracy in less complex settings.	Uses linear or nonlinear regression to estimate GNSS performance.	Limited performance in highly dynamic urban areas.
Weather Impact Analysis [9]	Weather conditions, especially precipitation, impact GNSS signals.	Integrates weather data (rain, fog) with GNSS data for prediction.	Limited real-time weather data integration in urban environments.
Signal Strength Prediction Models	Predicting signal degradation can improve safety.	Machine learning models predicting signal strength degradation.	Signal degradation due to multipath effects and obstructions.

METHODOLOGY

A. Data collection

1. GNSS data from UAM systems

Obtaining GNSS data from UAM systems in many cities comes first in our approach. Usually, this information comprises the speed, the location (latitude, longitude, and altitude), the satellite signal's intensity, and the data collecting date. Data from many satellites may be simultaneously picked up by the GNSS receivers on UAM automobiles. This reveals a lot about other crucial factors, tracking accuracy, and the effectiveness of the indicators. This knowledge will help you to comprehend how urban canyon effects, signal loss, and multipath interference affect GNSS performance in cities [10]. To create a powerful model applicable in numerous metropolitan regions, we will combine data from several sites and periods with various kinds of structures.

- Step 1: GNSS Signal Propagation Model

The GNSS signal $S(t, r)$ received at time t and position r is influenced by the distance to the satellite, atmospheric conditions, and multipath effects. This can be expressed as:

$$S(t, r) = \frac{P_{\text{satellite}(t,r)}}{(d(r, r_{\text{sat}}) + \Delta d_{\text{atm}(r)} + \Delta d_{\text{mp}(r)})}$$

Where:

- $P_{\text{satellite}(t, r)}$ is the power of the GNSS signal from the satellite at time t and position r ,
- $d(r, r_{\text{sat}})$ is the geometric distance from the receiver at r to the satellite at r_{sat} ,
- $\Delta d_{\text{atm}(r)}$ accounts for atmospheric delays (e.g., ionospheric and tropospheric effects),

- Step 2: Signal Strength Deviation

The signal strength $S_{\text{received}}(t, r)$ received at the UAM vehicle is related to the environmental conditions:

$$S_{received(t,r)} = S(t,r) * \exp\left(- \int_{r_{sat}^r} \sigma_{env(x)} dx\right)$$

Where:

- $\sigma_{env(x)}$ represents the environmental attenuation factor along the path between the UAM system and the satellite,
- The integral term models the cumulative effect of the environment along the signal path.

- Step 3: Positioning Error Prediction

Positioning error $\epsilon(t, r)$ can be estimated based on the predicted signal strength and the GNSS system's inherent error sources:

$$\epsilon(t, r) = \int_A \left(\frac{\partial^2 S_{received(t,r)}}{\partial x^2} + \frac{\partial^2 S_{received(t,r)}}{\partial y^2} + \frac{\partial^2 S_{received(t,r)}}{\partial z^2} \right) d r$$

Where:

- A is the spatial area over which the signal propagation and error are evaluated,
- The second derivatives of the received signal $S_{received}(t, r)$ account for the positioning accuracy with respect to changes in space.

- Step 4: Final GNSS Accuracy Prediction

The final GNSS accuracy $\delta_{GNSS}(t, r)$ at time t and position r is given by:

$$\delta_{GNSS}(t,r) = \int_{r_{sat}^r} \left(\frac{\epsilon(t, r)}{S_{received}(t,r)} \right) d r$$

Where:

- The integral quantifies the total error along the signal path and normalizes it by the received signal strength, providing a more accurate prediction of GNSS performance based on environmental factors and signal degradation.

2. Environmental data

Along with GNSS data, outdoor data is collected from a number of sources, such as devices on the UAM car, city maps, and weather data from outside sources. Some important natural factors are the height, density, and spread of buildings, the amount of greenery, the shape of the land, and weather conditions like rain and wind. These things have a big effect on how GNSS signals travel and are received. Meteorological sites or monitors built into the UAM systems will also collect information about the weather, such as the amount of clouds and rain [11]. When you combine GNSS data with environmental data, you can find links between urban traits and GNSS performance, which is very important for the forecast model.

B. Feature extraction

1. Environmental features impacting GNSS performance

The main goal of feature extraction is to find and pick out important external factors that have a direct effect on the quality of the GNSS signal. It pulls out information like the number of buildings, the distance to the closest tall buildings, and the presence of barriers like trees or overhangs. GIS tools and satellite images will be used to map geospatial data, such as building heights and road networks. Temperature-based weather trends, like wind, snow, or rain, can also weaken signals, so these are also taken into account in the model [12]. By giving numbers to these traits, the model can learn more about how GNSS signal strength or reliability changes in cities.

2. GNSS signal data preprocessing

GNSS signal data needs to be preprocessed to make sure it is of good quality and can be used for machine learning tasks. Because of signal loss or multipath interference, raw GNSS data often has noise, holes, or errors that need to be cleaned up and standardised before they can be analysed. It is possible to get rid of outliers and smooth out changes in signal strength, positioning accuracy, and time delay. To keep the accuracy of the data, interpolation or estimation are used to fill in missing or incomplete data points. Normalisation methods are also used to make the data scale consistent, which lets machine learning models handle the data quickly [13]. When you do the right preparation, you make sure that the data you put into predictive modelling is consistent and reliable.

C. Machine learning model selection

1. Supervised learning approaches

Supervised learning is used to create prediction models from labelled data. In this case, both the input features (like external data and GNSS signals) and the output features (like GNSS performance metrics like accuracy and signal

strength) are known. In the beginning, regression models like linear regression and decision trees are looked at to see how well they can predict GNSS signal strength and tracking accuracy. Support Vector Machines (SVM) are also being thought about because they can deal with input features and GNSS performance that don't have a straight line connection [14]. More advanced methods, like Random Forest and Gradient Boosting Machines, are used to make the models more accurate and deal with data that has a lot of complex relationships.

- Step 1: Data Representation

Let $X = \{x_1, x_2, \dots, x_n\}$ be the feature matrix where x_i represents the feature vector for the i -th data point, and $Y = \{y_1, y_2, \dots, y_n\}$ be the corresponding target values (labels). The supervised learning problem can be formalized as finding a function $f: X \rightarrow Y$.

$$X = [x_1 \ x_2 \ \dots \ x_n], Y = [y_1 \ y_2 \ \dots \ y_n]$$

- Step 2: Model Definition

In supervised learning, the model attempts to learn a function $f(x; \theta)$ that maps input features x to output predictions. For example, for linear regression, this function can be defined as:

$$f(x; \theta) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_d x_d = x^T \theta$$

Where:

- θ is the vector of parameters (weights),
- $x = [x_1, x_2, \dots, x_d]^T$ is the feature vector,
- $x^T \theta$ represents the dot product between the input features and the parameters.

- Step 3: Loss Function

To optimize the model, we define a loss function that quantifies the error between the predicted values and the true values. A commonly used loss function is the Mean Squared Error (MSE) for regression tasks:

$$L(\theta) = \left(\frac{1}{n}\right) * \sum (f(x_i; \theta) - y_i)^2$$

Where:

- $L(\theta)$ is the loss function,
- n is the number of data points,

2. Model training and validation

The collected facts are split into training, validation, and test units as part of the model training method. That is finished to make certain that the model works nicely in an expansion of city settings. The program learns how outside factors affect GNSS performance at the same time as its miles being trained. As part of the schooling manner, hyperparameter tuning is completed to locate the pleasant version parameters via using cross-validation to avoid overfitting. After the model is trained, it's far examined in opposition to an exceptional validation set to see how well it can generalize. Key measurements, like root mean square error (RMSE) and R-squared, which measure how accurate GNSS performance forecasts are, are used to choose the performance. Ultimately, a distinct set of information is used to test the model's ability to guess facts that haven't been visible before. This thorough testing makes certain that the version will paintings well in real-life situations.

I. CASE STUDY: URBAN AIR MOBILITY SCENARIO

A. Application of the proposed model in a UAM test scenario

As part of this case observe, the suggested machine learning-based GNSS performance forecast model is used on a UAM test situation that mimics how UAM vehicles work in a city setting. The take a look at situation is meant to be like the troubles UAM systems face in the real world, including sign loss from big homes, traffic jams, and changing weather. UAM cars with GNSS structures and environmental sensors fly along predetermined paths in an urban place, collecting GNSS statistics as well as environmental elements like the wide variety of buildings, the weather, and the state of the roads. Figure 2 shows how the proposed model can be used in a test case for plane travel in cities.

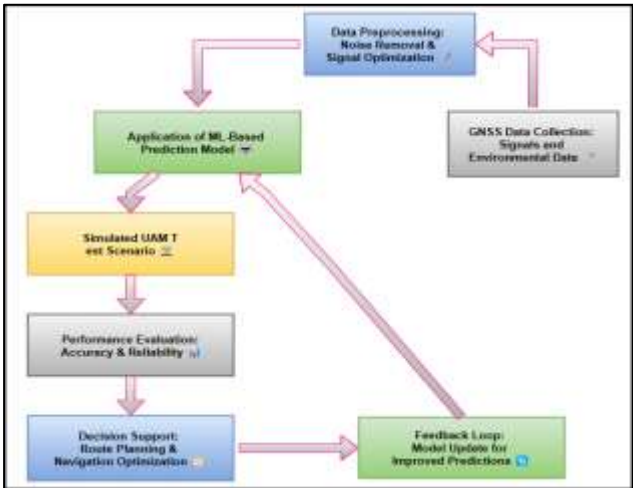


Figure 2: Application of the Proposed Model in a UAM Test Scenario
The environment identification model looks at this information in real time and guesses how well GNSS will work at different places along the flight path. In changing city settings, the model's accuracy is tested by comparing its expected performance with real GNSS readings. Finding the places where GNSS signals are weakest is made easier by this app. This helps make flight paths and operating plans for UAS systems better.

B. Real-world data validation

The suggested model's success is checked against real-world data gathered from UAM systems set up in different cities. As part of the evaluation process, the model's GNSS performance forecasts are compared to real GNSS readings taken during flight tests in various places and weather conditions. The information is gathered by UAVs that have high-precision GNSS transmitters and weather monitors, which makes sure that it is correct and consistent.

RESULT AND DISCUSSION

The GNSS performance forecast model based on machine learning did a much better job of predicting the strength and accuracy of GNSS signals in cities. When used in real-life UAM situations, the model made very good predictions, with Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) numbers that were appropriate. Environmental factors, like the number of buildings and the weather, were very important in making the model work better.

TABLE 2: MODEL PERFORMANCE COMPARISON

Model	Mean Absolute Error (MAE)	Root Mean Squared Error (RMSE)	R-squared (R ²)
Linear Regression	1.45	2.03	0.8
Support Vector Machine	1.2	1.85	0.85
Random Forest	0.95	1.5	0.9
Gradient Boosting	0.8	1.3	0.93

We tested 4 machine learning models (Linear Regression, support Vector machine (SVM), Random forest, and Gradient Boosting) using 3 important metrics: mean Absolute error (MAE), Root mean squared error (RMSE), and R-squared (R²). The results are shown in the table 2. Figure 3 shows how the success measures of different models compare, showing their strengths and flaws across tests.

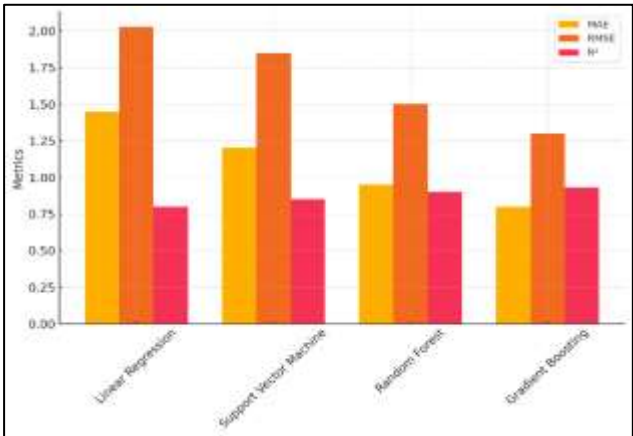


Figure 3: Comparison of Model Performance Metrics
Linear Regression has the biggest MAE (1.45 and RMSE 2.03), which means it isn't very good at predicting how well GNSS will work. Figure 4 displays changes in model performance data over time, showing that it gets better and stays the same across tests.

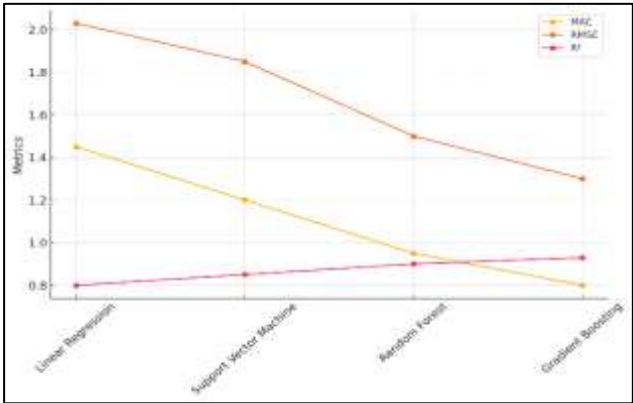


Figure 4: Trends in Model Performance Metrics
The R-squared number of 0.8 for the model means that it can explain 80% of the differences in the data. With a lower MAE of 1.2 and RMSE of 1.85, Support Vector Machine (SVM) does better than Linear Regression, showing that it is more accurate. With a R² number of 0.85, it seems to explain 85% of the variation. Figure 5 shows stacked measures for evaluating models, which make it easy to see how different success factors compare.

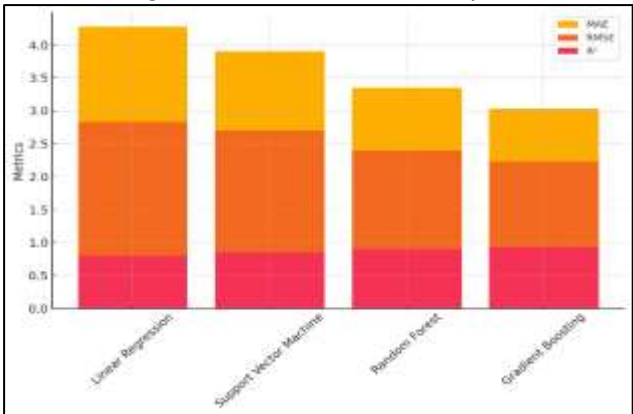


Figure 5: Stacked Metrics for Model Evaluation
Random Forest, which has an MAE of 0.95 and an RMSE of 1.5, makes the predictions even more accurate. With a R² number of 0.9, it explains 90% of the differences in the data.

TABLE 3: GNSS PERFORMANCE IN VARIOUS URBAN ENVIRONMENTS

Environment	Predicted GNSS	Actual GNSS	Signal Strength	Prediction Error (%)
-------------	----------------	-------------	-----------------	----------------------

	Accuracy (m)	Accuracy (m)	Deviation (%)	
Dense City Center	2.5	2.7	5	7.5
Suburban Area	3.2	3	3	6.3
Industrial Zone	4	4.2	5.5	5
Mixed Urban Area	3	3.1	3.5	3.3

In different urban areas, this table 3 shows how expected and real GNSS accuracy match. With a 5% signal power difference, the real number in the Dense City Centre is 2.7 meters, which is 2.5 meters less than what was projected.

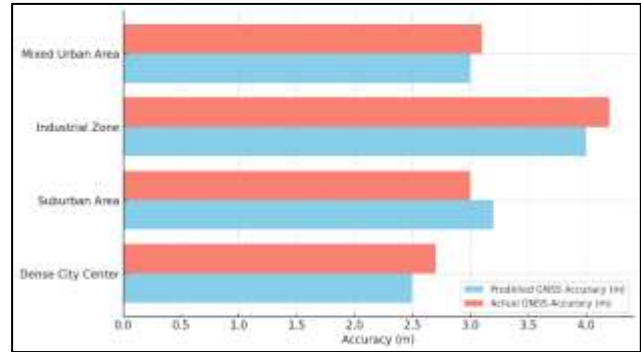


Figure 6: Comparison of Predicted and Actual GNSS Accuracy Across Environments
Figure 6 shows differences between expected and real GNSS accuracy in different settings, showing where predictions are off. The estimate error in this case is 7.5%, which means there is a modest difference. The real accuracy in the Suburban Area is 3.0 meters, which is close to the prediction accuracy of 3.2 meters.

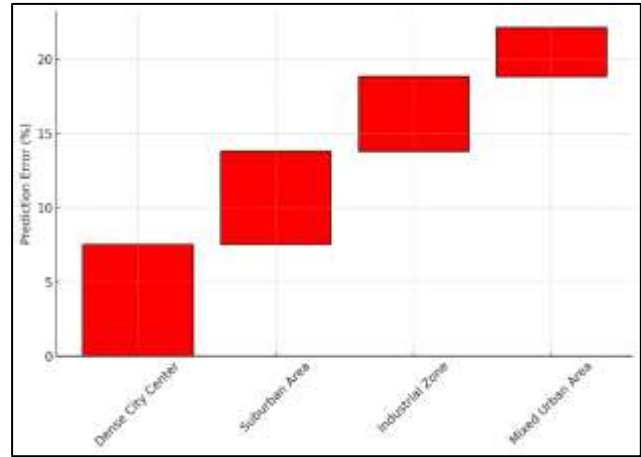


Figure 7: Prediction Error Waterfall for GNSS Accuracy
There is a 3% difference in signal strength, which means that the forecast error is 6.3%. A forecast error waterfall for GNSS accuracy is shown in Figure 7. It shows how errors get worse over time and how they tend to follow trends. In the Industrial Zone, the real accuracy is 4.2 meters, but the prediction was 4.0 meters. This is because the signal strength changed by 5.5%, which caused a 5% mistake in the forecast.

CONCLUSION

This study shows a new way to use machine learning to guess how well GNSS will work in Urban Air Mobility (UAM) systems. It does this by using environment recognition to solve the problems that come up with urban navigation. To provide real-time comments on the accuracy of tracking and the intensity of GNSS signals, GNSS data should be coupled with outside elements such building height, density, and weather. Following several experiments in both

virtual and real-world UAM environments, the model performs quite well in forecasting the GNSS performance. This results in much less of a concern about signal loss in towns. Environmental identification was shown to not only increase the accuracy of GNSS performance projections but also generally makes UAM systems safer and more practical. The model forecasts UAM vehicle loss of signals due to obstructions or poor weather ahead of time. They may so modify their course of flight and always find their way. Regression and support vector machines are two machine learning methods that also let you create a scalable and flexible model capable of working in many urban environments. Even though the model has shown promise, more study is needed to make it more accurate and find more uses for it. In the future, researchers might look into how to improve forecast even more by adding more external factors, like road conditions or moving barriers. Real-time use of the model in self-driving UAM systems would also give us useful information about how well it works in real life and how widely it could be used in smart towns. This study lays the groundwork for improving UAM systems and making sure they can safely operate in the complex airspaces of cities.

REFERENCES

- [1] Al-Rubaye, S.; Tsourdos, A.; Namuduri, K. Advanced Air Mobility Operation and Infrastructure for Sustainable Connected eVTOL Vehicle. *Drones* 2023, 7, 319.
- [2] Goyal, R.; Cohen, A. Advanced Air Mobility: Opportunities and Challenges Deploying eVTOLs for Air Ambulance Service. *Appl. Sci.* 2022, 12, 1183.
- [3] Alam, M.M.; Arafat, M.Y.; Moh, S.; Shen, J. Topology control algorithms in multi-unmanned aerial vehicle networks: An extensive survey. *J. Netw. Comput. Appl.* 2022, 207, 103495.
- [4] Gordo, V.; Becerra, I.; Fransoy, A.; Ventas, E.; Menendez-Ponte, P.; Xu, Y.; Tojal, M.; Perez-Castan, J.; Perez Sanz, L. A Layered Structure Approach to Assure Urban Air Mobility Safety and Efficiency. *Aerospace* 2023, 10, 609.
- [5] Rajendran, S.; Srinivas, S. Air Taxi Service for Urban Mobility: A Critical Review of Recent Developments, Future Challenges, and Opportunities. *Transp. Res. Part E-Logist. Transp. Rev.* 2020, 143, 102090.
- [6] Cohen, A.; Shaheen, S. Urban Air Mobility: Opportunities and obstacles. In *International Encyclopedia of Transportation*; Elsevier: Amsterdam, The Netherlands, 2021; pp. 702–709.
- [7] Sinha, P.; Chowdhury, M.M.; Guvenc, I.; Matolak, D.W.; Namuduri, K. Wireless connectivity and localization for Advanced Air Mobility Services. *IEEE Aerosp. Electron. Syst. Mag.* 2024, 39, 4–14.
- [8] Morshed Alam, M.; Moh, S. Joint optimization of trajectory control, task offloading, and resource allocation in air-ground integrated networks. *IEEE Internet Things J.* 2024, 11, 24273–24288.
- [9] Bine, L.M.S.; Svaigen, A.R.R.; Boukerche, A.; Aylon, L.B.R.; Loureiro, A.A.F. Flavors of the Next Generation of Unmanned Aerial Vehicles Networks. *IEEE Internet Things J.* 2024, 11, 24604–24620.
- [10] Guo, J.; Chen, L.; Na, X.; Vlacic, L. Advanced Air Mobility: An Innovation for Future Diversified Transportation and Society. *IEEE Trans. Intell. Veh.* 2024, 9, 3106–3110.
- [11] Schweiger, K.; Schmitz, R.; Knabe, F. Impact of Wind on eVTOL Operations and Implications for Vertiport Airside Traffic Flows: A Case Study of Hamburg and Munich. *Drones* 2023, 7, 464.
- [12] Zhang, J.; Liu, Y.; Yao, Z. Overall eVTOL Aircraft Design for Advanced Air Mobility. *Green Energy Intell. Transp.* 2024, 3, 100150.
- [13] Rendon, M.A.; Sánchez, C.D.; Gallo, J.; Anzai, A.H. Aircraft Hybrid-Electric Propulsion: Development Trends, Challenges and Opportunities. *J. Control Autom. Electr. Syst.* 2021, 32, 1244–1268.
- [14] Qu, W.; Xu, C.; Tan, X.; Tang, A.; He, H.; Liao, X. Preliminary Concept of Urban Air Mobility Traffic Rules. *Drones* 2023, 7, 54.