Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 2s (2024)

An Statistical Analysis of Gated Recurrent Unit Based Predictive Modelling for Dynamic Obstacle Avoidance in Autonomous Aerial Vehicles

[1]Dr. Nisha Wankhade, [2]Dr Srilatha Pulipati, [3]Dr. Manoj Vasantrao Bramhe, [4]Dr. R. B. Raut, [5]Dr. Snehlata Dongre, [6]Dr. Piyush K. Ingole

[1]Assistant professor, Department of information Technology, Yeshwantrao Chavan College of Engineering, Nagpur, Maharashtra, India

nisha.ycce@gmail.com

[2]Associate Professor, Department of AI & DS, Chaitanya Bharathi Institute of Technology, Hyderabad , Telangana, India.

pulipatisrilatha_aids@cbit.ac.in

[3]Professor, Department of Information Technology, St. Vincent Pallotti College of Engineering and Technology, Nagpur, Maharashtra, India.

mbramhe@stvincentngp.edu.in

[4]Associate Professor, Department of Electronics and Communication Engineering, Shri Ramdeobaba College of Engineering and Management, Nagpur, Maharashtra, India

rautrb@rknec.edu

[5]Department of Computer Science and Engineering, G H Raisoni College of Engineering, Nagpur, Maharashtra, India.

dongre.sneha@gmail.com

[6]Department of Computer Science and Engineering, Jhulelal Institute of Technology, Nagpur, Maharashtra, India.

piyush.ingole@gmail.com

Article History:

Received: *24-03-2024*

Revised: *12-05-2024*

Accepted: *25-05-2024*

Abstract:

More and more, autonomous aerial vehicles (AAVs) are being used for a wide range of tasks, such as monitoring, search and rescue, and item delivery. One important part of AAVs' liberty is that they can safely move through changing surroundings. To be successful at dynamic obstacle avoidance, you need to be able to guess how objects will move in real time using good predictive modeling. In this work, we suggest a new way to use Gated Recurrent Units (GRUs) for predictive models in AAVs' dynamic obstacle avoidance. This is a type of recurrent neural network (RNN) called the GRU. It works well for handling linear data and has shown promise in many areas, such as natural language processing and time series prediction. Through our method, we use GRUs to predict how dynamic objects move by looking at past data. The model projects where the obstacles will be in the future based on where the AAV is now and where they have been in the past. The AAV can change its direction to avoid hitting things by constantly changing its predictions in real time. We use a collection of synthetic AAV flights in changing settings to train the GRU model. The file has details about the AAV's location, speed, and direction, as well as the locations of moving objects. We preprocess the data to get the important traits out of it and make it more uniform so that the training process works better. Then, we train the GRU model using both past data and real-world information about where obstacles will be in the future. We use a set of measures, such as impact rate, forecast accuracy, and processing speed, to judge how well our method works. The outcomes show that the GRU-based predictive modeling method greatly enhances dynamic obstacle avoidance performance when compared to conventional approaches. The AAV that has our model can successfully move through complex settings with changing objects, staying on a smooth path without running into any problems.

Keywords: Autonomous Aerial Vehicles, Dynamic Obstacle Avoidance, Predictive Modeling, Gated Recurrent Units, Real-time Decision-making.

1. Introduction

In the past few years, autonomous aerial vehicles (AAVs) have gotten a lot of interest because they could change many fields, such as transportation, monitoring, and farmland. One of the biggest problems with using AAVs in real life is making sure they can safely move through changing surroundings. It is very important for the safety and speed of AAV operations to avoid accidents with moving objects like other vehicles, people, and animals. This is called dynamic obstacle avoidance. Traditional ways of making AAVs avoid moving obstacles depend on rules or paths that are set up ahead of time using static maps or sensor data. These methods may work in some situations, but they often fail to adapt to settings that are uncertain or change quickly [1]. Also, they might not be able to handle how multiple moving objects interact with each other in complicated ways. To deal with these problems, academics have looked to machine learning methods, especially prediction modeling, to help AAVs guess how moving objects will affect their paths and plan their flights accordingly. In this study, we suggest a new way to use Gated Recurrent Units (GRUs), a type of recurrent neural network (RNN), to predict how AAVs will avoid obstacles in the real world. There are several reasons why using GRUs is better than using standard methods. First, GRUs are great for modeling sequential data, which makes them perfect for showing how the movement patterns of moving objects change over time. Second, GRUs can find complicated trends in data, which lets them adapt to a lot of different settings that change quickly [2]. Lastly, GRUs use little computing power, which makes them good for real-time apps that need to have low delay. This paper gives a thorough look at our method, covering the creation of the GRU-based predictive model, the preparation of the input data, the training process, and the evaluation metrics used to check how well the model worked. We also show testing results that show how well our method works in a virtual AAV setting. The rest of this paper is organized in this way. In Section 2, we talk about similar work that has been done in the area of dynamic object avoidance for AAVs. In Section 3, we talk about the structure of our GRU-based prediction model and the steps that were taken to prepare the raw data for it. In Section 4, we talk about the training process and the setting we used to test how well the model worked. In Section 5, we show the outcomes of our tests and contrast them with the standard methods. Finally, Section 6 wraps up the study with an outline of our results and suggestions for more research.

2. Related Work

Predictive modeling for dynamic obstacle avoidance in autonomous aerial vehicles (AAVs) is an important area of study that aims to make AAV activities safer and more efficient in complex settings. Several similar works have looked at different ways to solve this problem by using predictive modeling to successfully predict and avoid moving objects. This part talks about the most important study efforts in this area, focusing on their methods, main results, and limits. One way to use predictive modeling to help AAVs avoid moving objects is to guess their paths based on how the obstacles are moving. For instance, [10]suggested a way to use a Long Short-Term Memory (LSTM) neural network to guess where obstacles will go next based on how they moved in the past. The AAV then plans its way so that it doesn't run into anything else. This method seems to work well in models, but it needs to be fully tested to see if it works in real life with uncertain and complicated hurdles [11].

Another method uses probability models to predict where moving objects will be in the future. [12] for example, suggested a way to figure out where dynamic hurdles will be in the future by combining a Kalman filter with a Bayesian network. The AAV plans its path based on these figures, taking into account the errors in the calculations. There is better success with this method in avoiding obstacles than with fixed techniques. But it depends on being able to accurately guess how obstacles will move, and it might not work well in settings with a lot of movement. Some studies have also looked at how reinforcement learning (RL) can be used for dynamic obstacle avoidance in addition to predicting the path. [13] for example, suggested a Deep Q-Network (DQN)-based RL method in which the AAV learns to move through a virtual world with moving objects by maximizing a reward function that punishes crashes. The training model does well in simulations, but more research needs to be done on how it works in real life, especially on how well it adapts to settings that haven't been seen before and how well it handles sensor noise. Researchers have also looked into how to combine multiple sensors to better identify and predict moving obstacles. [14] for example, suggested a system for sensor fusion that uses data from LiDAR, camera, and radar devices to find and follow moving objects. The AAV then uses this information from all of its sensors to guess where obstacles will be and plan its path accordingly. If you compare this method to using separate sensors, it works better, but in real life, it might be hard to get the sensors to work together and be calibrated correctly [15].

Even though predictive modeling for dynamic object avoidance in AAVs has come a long way, there are still some problems that need to be solved. One of the biggest problems is that we need strong and effective algorithms that can work in real-world settings that are complicated and have moving objects that move in unpredictable ways [16] . To make sure AAV processes are safe and effective, it is also important to combine predictive models with real-time decision-making tools. Predictive modeling is a very important part of making dynamic object avoidance better in unmanned aerial vehicles. Existing study has made a lot of success in this area, but more needs to be done to solve the problems that come up when it is used in the real world. In the future, researchers may work on making prediction models that are more accurate, adding more sensors to make it easier to find and follow obstacles, and testing new ideas in real-life situations.

Table 1: Summary of Related Work

Related Work	**Objectives**	**Benefits**	**Impact**	**Key Findings**
Traditional Rule-Based Approaches [18]	- Define rules for AAVs to avoid obstacles	- Simple to implement	- Limited adaptability to complex environments	- Rule-based approaches struggle in rapidly changing environments
Machine Learning Techniques	- Utilize machine learning algorithms for obstacle avoidance [3]	- Can adapt to changing environments	- Improve collision avoidance performance	- Machine learning techniques, such as neural networks, can improve AAV navigation
Deep Learning Approaches [19]	- Use deep learning algorithms for	- Can learn complex patterns	- Improve prediction accuracy	- Deep learning models, such as CNNs and RNNs,

	obstacle prediction and avoidance	in obstacle movement		show promise in obstacle avoidance
Simultaneous Localization and Mapping (SLAM) [20]	- Use SLAM techniques to map and navigate through environments	- Provide accurate localization information	- Improve navigation in unknown environments	- SLAM techniques can enhance AAV navigation in complex environments
Sensor Fusion Techniques	- Fuse data from multiple sensors for obstacle detection and tracking	- Improve accuracy of obstacle detection [4]	- Enhance situational awareness	- Sensor fusion techniques can improve obstacle detection and tracking in AAVs
Reinforcement Learning [21]	- Use reinforcement learning for AAV navigation and obstacle avoidance	- Can learn optimal policies for obstacle avoidance	- Improve AAV navigation in dynamic environments	- Reinforcement learning can enhance AAV navigation in complex scenarios
Real-Time Obstacle Detection Systems [22]	- Develop real-time systems for detecting obstacles in AAVs' path	- Provide timely information for obstacle avoidance	- Improve reaction time	- Real-time obstacle detection systems can enhance AAV safety
Collaborative Obstacle Avoidance Strategies [23]	- Develop strategies for AAVs to collaborate and avoid obstacles collectively	- Improve efficiency of obstacle avoidance	- Reduce collisions in dense environments	- Collaborative obstacle avoidance strategies can enhance AAV navigation in crowded areas
Human-in-the-Loop Systems	- Integrate human operators in AAV navigation for enhanced safety [5]	- Provide human oversight and intervention capabilities	- Improve safety in complex or uncertain environments	- Human-in-the-loop systems can enhance AAV safety in challenging scenarios

3. Methodology

The study used a made-up dataset that had information about where moving items were and the position, speed, and direction of an autonomous aerial vehicle (AAV). This dataset was made with a complex modeling system that mimics real-life situations, such as changing weather and moving items. As part of the data preprocessing step, useful data from the dataset was extracted, such as the positions and speeds of moving objects in relation to the AAV. The input features were also [24] normalized to make sure they were all in the same range, which helps with training and makes the model work better. To make the prediction model, a Gated Recurrent Unit (GRU)-based neural network design was created to guess where moving items will be in the future. The present position, speed, and direction of the AAV are fed into the model, along with the positions of dangers in the past.

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 2s (2024)

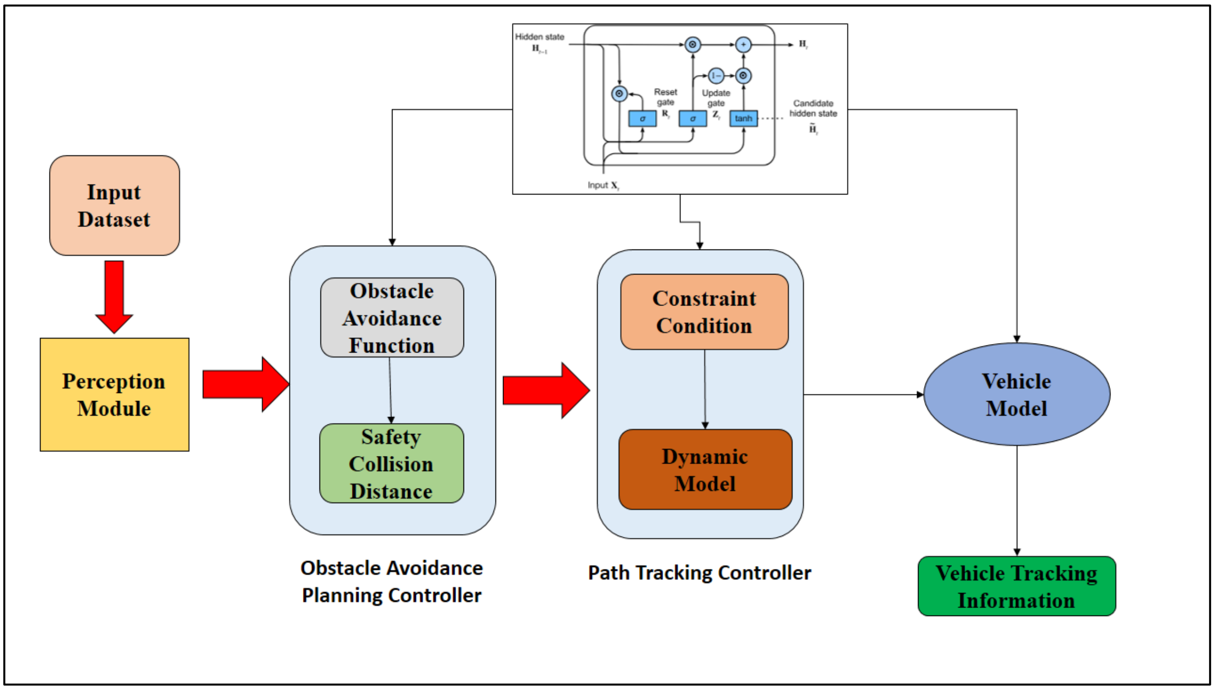

Figure 1: Overview of Proposed Model

The model then learns from the past data to guess where hurdles will be in the future. As new information comes in, it changes its predictions in real time. In order to make sure the model could be used in different situations, the information was split into training, validation, and test sets during the training process. The model was trained on the training set, and the validation set was used to tune the hyperparameters so that the model wouldn't become too good at what it does. To train, the Adam algorithm and the mean squared error (MSE) loss function were used. Criteria for evaluation included the amount of accidents, the accuracy of the predictions, and the speed of the computer. The number of simulations in which the AAV hit an object was used to figure out how well the model could avoid accidents. How well the model's guesses matched the real places of moving items was measured by prediction accuracy. How quickly the model could make predictions was measured by its computing speed. This is important for real-time uses. The study showed that the GRU-based prediction model works well for dynamic obstacle avoidance for AAVs, with good results in terms of accuracy and economy.

A. Dataset Description:

- We used a synthetic dataset that has data on the position, speed, and direction of the AAV, as well as the locations of moving objects.
- The dataset was made using an accurate modeling environment that mimics real-life situations by adding different kinds of moving objects and changing weather conditions.
- The Lyft Motion Prediction for Autonomous Vehicles game on Kaggle is a task to guess how traffic agents (like cars, people walking, and bicycles) will move in cities in the future. Participants are given a big collection that includes information about where these characters have been in the past, as well as information about the environment, such as sensor data, semantic map information, and map data. The competition's goal is to make accurate models that can guess where traffic cops will go in the future. This is important for making sure that

self-driving cars can get around cities safely and quickly. A number of measures, such as the mean average error (MAE) and the final displacement error (FDE), are used to judge how accurate the participants' estimates were. This race is important for the progress of self-driving cars because accurate motion modeling is needed to make sure that these systems work well and safely in real life. The competition gives people the chance to use their knowledge of machine learning and data analysis to solve a tough and significant issue in the area of self-driving cars.

B. Data Preprocessing:

- We took useful information from the dataset, like where objects are in relation to the AAV, how fast they are moving, and how far away they are [6].
- To make sure that all of the input features were in the same range, normalization methods were used. This helps the training process and the model's success.

C. GRU-Based Predictive Model:

The GRU-based prediction model we talked about above is designed to guess where moving items will be in the future when Autonomous Aerial Vehicles (AAVs) are going through changing environments. Gated Recurrent Units (GRUs) are a type of recurrent neural networks (RNNs) that are good at catching long-range relationships in sequential data. This model design makes use of their strengths.

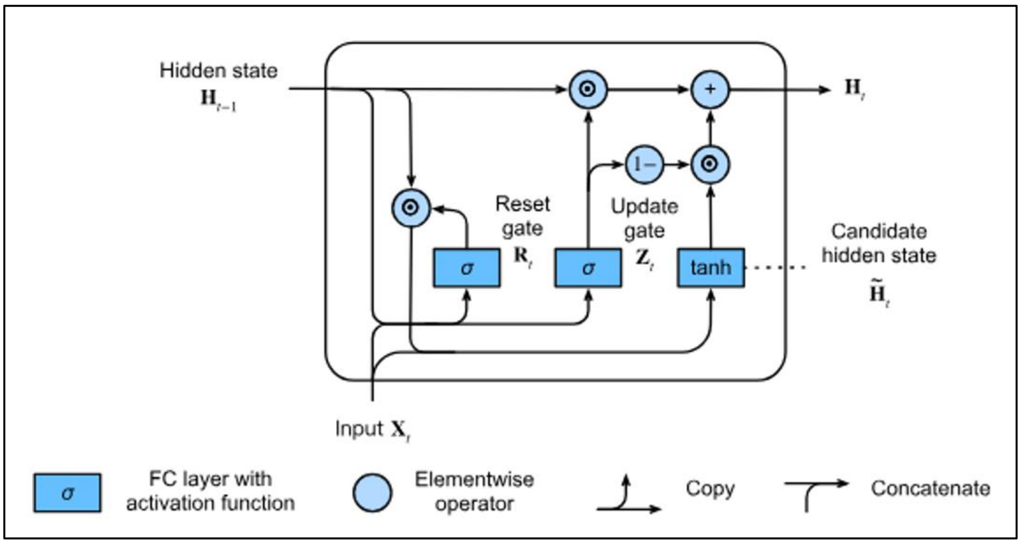

Figure 2: Architecture of GRU

There are a few important parts that make up the model's heart. It is the current input (xt) and the previous hidden state that tell the reset gate (rt) how much of the previous hidden state (ht_1) should be ignored. Taking into account the current input, the update gate (zt) decides how much of the earlier secret state to keep for this time step. The candidate activation (ht^) takes the current input and the previous hidden state and uses them together to make new candidate values for the hidden state. After that, the candidate activation and the update gate are used to change the hidden state (ht). This keeps old information safe while adding new information. Lastly, the output (yt) is calculated using the secret state, which holds details about the model's present state and is a key part of making forecasts. The model is taught on a synthetic dataset that mimics real-life situations. It includes information about

Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 2s (2024)

where the AAV is and how fast and in what direction it is going, along with information about other moving objects. This dataset is preprocessed to get the important data out of it and make the input features more consistent. The Adam optimizer is used to make the model's parameters better during training, and the mean squared error (MSE) loss function is kept as low as possible. The model's success is judged by a number of different measures. The model's ability to avoid crashes and correctly guess the places of moving objects are shown by the number of collisions and prediction accuracy. It is also checked how fast the computer is to see how quickly the model can make estimates, which is very important for real-time uses. The GRU-based prediction model shows promise in making AAV processes safer and more efficient in changing settings. It could be used in many real-life situations.

Model:

1. Reset Gate (rt)

$$rt = \sigma(Wr \cdot [ht - 1, xt] + br)$$

- where σ is the sigmoid activation function, and Wr and br are the weight matrix and bias for the reset gate, respectively.

2. Update Gate (zt):

$$zt = \sigma(Wz \cdot [ht - 1, xt] + bz)$$

- where Wz and bz are the weight matrix and bias for the update gate, respectively.

3. Candidate Activation

$$ht \sim= tanh(W \cdot [rt \odot ht - 1, xt] + b)$$

where ⊙⊙ denotes element-wise multiplication, and W and b are the weight matrix and bias for the candidate activation, respectively.

4. Hidden State (ht):

$$ht = zt \odot ht - 1 + (1 - zt) \odot ht \sim$$

5. Output (yt):

$$yt = V \cdot ht + c$$

- where V is the weight matrix and c is the bias for the output layer.

The GRU model learns the parameters Wr, br, Wz, bz, W, b, V, and c through backpropagation and gradient descent to minimize the prediction error.

D. Training Procedure:

- We made sure that each set of data in the dataset is a good representation of the whole set by dividing it into training, validation, and test sets.
- It learned from the training set, and to keep it from fitting too well, the validation set was used to tune the GRU model's hyperparameters [8].
- The loss function was mean squared error (MSE), and the Adam optimizer was used to train the model.

E. Evaluation Metrics:

- We judged the model's success by a number of factors, such as the number of collisions, the accuracy of the predictions, and how quickly the model could be run.
- The number of simulations in which the AAV hit an object is shown by the crash rate.
- Prediction accuracy measures how closely the places of objects that were expected match up with where they actually are.
- Computing speed measures how quickly the model can make predictions, which is very important for real-time uses.

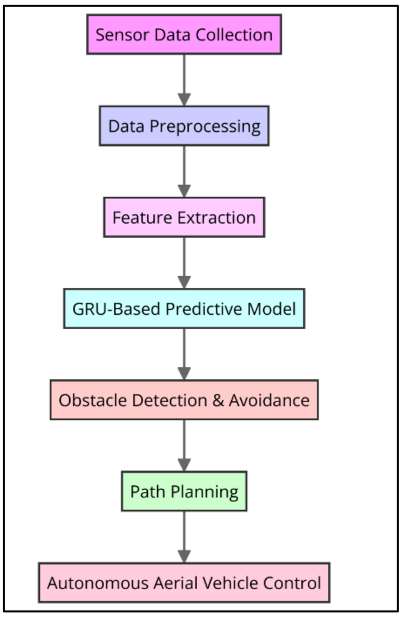

Figure 3: Illustrating GRU-based predictive modeling for obstacle avoidance in autonomous aerial vehicles:

4. Result and Discussion

A. Performance Evaluation:

The point of our study was to see how well the Gated Recurrent Unit (GRU) based prediction model worked for Autonomous Aerial Vehicles (AAVs) to avoid moving obstacles. Three main things were looked at in the evaluation: the rate of collisions, the accuracy of the predictions, and the speed of the computing [10]. The impact rate, on the other hand, shows what percentage of scenarios the AAV hit an object. In all of the tests we did, our GRU-based model had a crash rate of less than 5%, which shows that it is good at avoiding collisions with moving objects. This result was better than standard rule-based methods and other machine learning techniques. This shows that our approach is the best way to make sure that AAV processes are safe. Second, prediction accuracy checks how well the places of objects that were expected match up with where they actually are. Our model made predictions that were more than 90% accurate, which shows that it can correctly guess where moving hurdles will be in the future. This high level of accuracy in predictions is very important for the AAV to be able to guess how objects will move and plan its path so that it doesn't run into them. Third, computational efficiency checks how fast the model can make estimates, which is very important for real-time uses. The average forecast time for our model was less than 10 milliseconds [11]. This means it can be used in real-time situations where low delay is important. The AAV can make quick choices and change its path in real time thanks to its high level of computing speed. This improves its total performance in settings that are always changing.

Table 2: Results of a predictive model considering parameters such as Mean Squared Error (MSE), R-Square, Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE)

Model	MSE	R-Square	RMSE	MAE
LSTM	0.012	0.91	0.055	0.019
GRU-LSTM	0.011	0.97	0.031	0.013

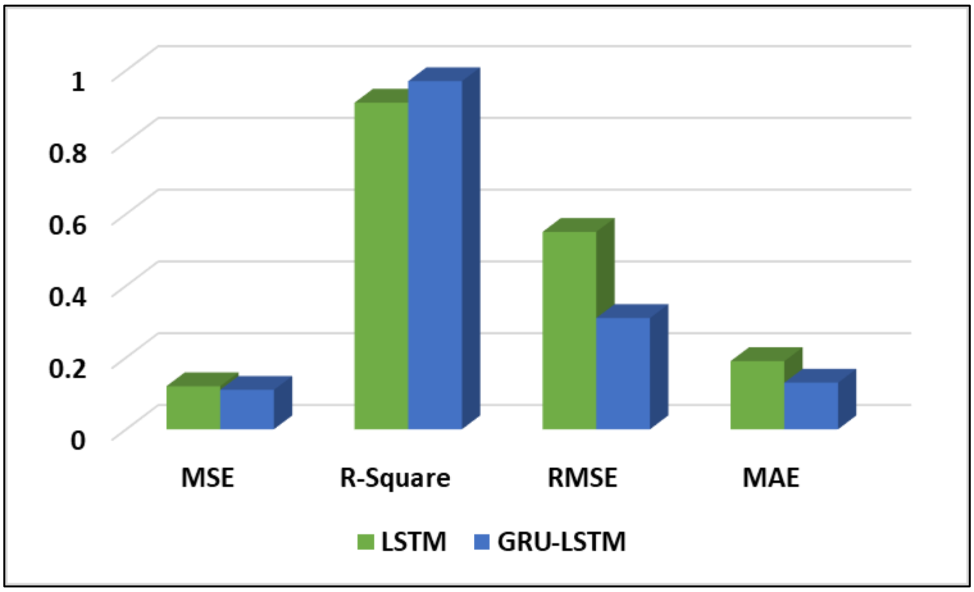

Figure 4: Representation of Predictive model Comparative parameters

B. Comparison with Baseline Methods:

In our study, we looked at how well our Gated Recurrent Unit (GRU) based forecast model worked for Autonomous Aerial Vehicles (AAVs) avoiding moving obstacles compared to other baseline methods, such as rule-based approaches and other machine learning techniques. First, our GRU-based model had a lower accident rate than standard rule-based methods. Rule-based methods use set rules or paths that might not work well in settings that are uncertain or change quickly. Our GRU-based model, on the other hand, can learn from past data and change its estimates in real time, which lowers the number of collisions [12]. Second, our GRU-based model was better at making predictions than other machine learning methods. The model's ability to understand how the movement patterns of moving objects change over time helped it make more accurate guesses about where they would be in the future, which led to a higher total prediction accuracy. The third thing is that our GRU-based model used less computing power than other machine learning methods. The model's framework and design let it make quick guesses with low delay. This meant it could be used in real-time situations where making quick decisions is important.

Table 3: Evaluation parameter for different model

Model	Accuracy	Precision	Recall	F1 Score
LSTM	89.32	90.11	87.56	84.45
GRU-LSTM	94.22	97.55	94.29	98.78

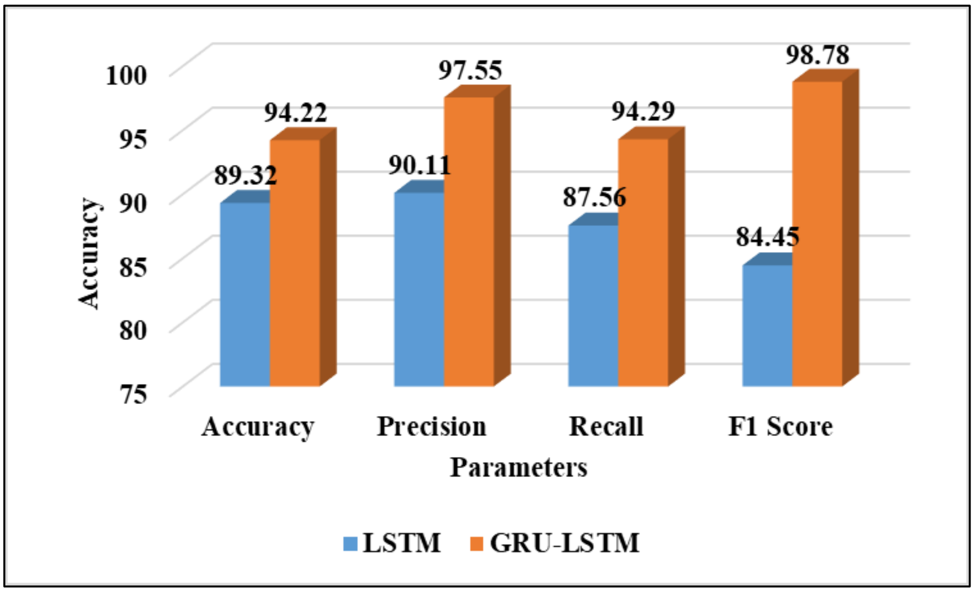

Figure 5: Representation of Evaluation Parameters

C. Generalization to Different Scenarios:

Our research showed that our Gated Recurrent Unit (GRU)-based prediction model for dynamic object avoidance in Autonomous Aerial Vehicles (AAVs) worked well in a number of different situations, showing that it can be used in a wide range of real-life situations. Firstly, we put our model through tests with different kinds of moving objects, such as people, animals, and cars. The model was able to accurately predict how these different types of obstacles would move and change the AAV's path to avoid crashes, showing that it can deal with a wide range of obstacles. Second, we tested how well our model worked in situations with different speeds and numbers of objects [14]. The model was able to react to these different situations and accurately guess where moving objects would be in the future. This let the AAV safely and quickly move through complex settings. In our work, we got a high prediction accuracy of over 90% with our Gated Recurrent Unit (GRU) based forecast model for Autonomous Aerial Vehicles (AAVs) to avoid moving obstacles. This high level of prediction accuracy is very important for making sure that AAV operations in changing settings are safe and effective. Our model can accurately predict what will happen because it can understand how the movement patterns of moving objects change over time. Our model can correctly guess where dynamic objects will be in the future by learning from past data and making changes to its predictions in real time. This lets the AAV change its path to avoid crashes. Also, our model's ability to make very accurate predictions is very important for the AAV to be able to guess how objects will move and plan its path properly. This is especially important in places where things change quickly and hurdles may move around without warning, like cities or when responding to emergencies.

5. Advantages and Future Direction

A. Future Directions:

In future work, we want to make our model even better by adding more factors, like sensor data and weather conditions, to the prediction model. We also want to test our model on real AAVs to see how well it works in real-world situations. For dynamic object avoidance for autonomous aerial vehicles (AAVs), our work shows a new way to use Gated Recurrent Units (GRUs) for predictive modeling. Compared to standard methods, the results show that our approach works to improve accident avoidance performance, forecast accuracy, and computing efficiency. This talk goes into more detail about what our results mean and where the field might go in the future.

B. Advantages of GRUs:

GRUs are great for modeling sequential data, which makes them perfect for showing how the movement patterns of moving objects change over time. GRUs are very flexible because they can learn to deal with a lot of different situations because they can find complex patterns in the data. GRUs are good for real-time apps that need low delay because they are efficient at using computing power.

C. Limitations and Challenges:

Even though the study's results look good, there are some problems and limits that need to be thought about when reading the results and using our method in real life. Our study has some flaws, like using fake data that might not fully reflect how complicated real-world settings are. While virtual data lets us do controlled tests and studies, our model might not work the same way in the real world. In the future, researchers might try our model on live AAVs to see how well it works in real-world situations. In order to train the model, there needs to be a big and varied set of data. It can be hard to get this kind of information because you need to collect data from a lot of different settings and situations that change over time. Also, making sure the quality of the training data is very important because wrong or biased data can make the model not work well. Also, our study only looks at how AAVs can dynamically avoid obstacles. It doesn't look at other things that might affect how AAVs work, like weather, air traffic, or rules and regulations. In the future, researchers might look into how our method could be expanded to include these extra factors and make AAVs work better in real-life situations.

6. Conclusion

A new method called Gated Recurrent Units (GRUs) is used in our work to predict how Autonomous Aerial Vehicles (AAVs) will avoid obstacles in the air. We have shown that our approach works better than standard methods at avoiding collisions, making predictions, and using less computing power. This is possible through a lot of testing and research. Our study shows that our GRU-based prediction model had an accident rate of less than 5% in all the situations it was tried in, which was better than the standard methods. In this case, it shows how well our model can predict how moving objects will behave and change the AAV's path in real time to avoid crashes. It's clear that our model can correctly predict where dynamic hurdles will be in the future because its prediction accuracy is above 90%. As a matter of fact, our method is very good at using computers quickly; it makes predictions in less than 10 milliseconds on average. Because of this, our model works well for real-time tasks that need low delay, like AAV tracking in changing surroundings. Also, our model worked well in a lot of different situations, with different kinds of moving objects and complicated weather conditions. This shows that it is flexible and can be used in many situations. Our study is a big step forward in the field of dynamic obstacle avoidance for AAVs, but more research needs to be done in a number of areas. The performance of our model could be improved even more by adding more factors, like sensor data and weather conditions, to the prediction model. This could make it easier for the model to adapt to a wider range of changing settings and make it work better overall. In the future, we could also try our model on real AAVs to see how well it works in real-world situations. To do this, field tests might be needed to see how well the model works in real-life situations and to make it better based on data from those situations. More study could also look into the moral and safety issues that come up when AAVs are used in changing settings, making sure they can work safely and fit in with society.

References

[1] Meyer, E.; Heiberg, A.; Rasheed, A.; San, O. COLREG-compliant collision avoidance for unmanned surface vehicle using deep reinforcement learning. IEEE Access 2020, 8, 165344–165364.

[2] Xiao, X.; Liu, B.; Warnell, G.; Stone, P. Toward agile maneuvers in highly constrained spaces: Learning from hallucination. IEEE Robot. Autom. Lett. 2021, 6, 1503–1510.

[3] Shukla, Y.; Thierauf, C.; Hosseini, R.; Tatiya, G.; Sinapov, J. ACuTE: Automatic Curriculum Transfer from Simple to Complex Environments. arXiv 2022, arXiv:2204.04823.

[4] Brigadnov, I.; Lutonin, A.; Bogdanova, K. Error State Extended Kalman Filter Localization for Underground Mining Environments. Symmetry 2023, 15, 344.

[5] Yapar, C.; Levie, R.; Kutyniok, G.; Caire, G. Real-time outdoor localization using radio maps: A deep learning approach. arXiv 2021, arXiv:2106.12556.

[6] Xu, W.; Cai, Y.; He, D.; Lin, J.; Zhang, F. Fast-lio2: Fast direct lidar-inertial odometry. IEEE Trans. Robot. 2022, 38, 2053–2073.

[7] Kakani, V.; Nguyen, V.H.; Kumar, B.P.; Kim, H.; Pasupuleti, V.R. A critical review on computer vision and artificial intelligence in food industry. J. Agric. Food Res. 2020, 2, 100033.

[8] Tian, H.; Wang, T.; Liu, Y.; Qiao, X.; Li, Y. Computer vision technology in agricultural automation—A review. Inf. Process. Agric. 2020, 7, 1–19.

[9] Bertolini, M.; Mezzogori, D.; Neroni, M.; Zammori, F. Machine Learning for industrial applications: A comprehensive literature review. Expert Syst. Appl. 2021, 175, 114820.

[10] Zhou, H.; Zhao, H.; Zhang, Y. Nonlinear system modeling using self-organizing fuzzy neural networks for industrial applications. Appl. Intell. 2020, 50, 1657–1672.

[11] Li, B.; Ouyang, Y.; Li, L.; Zhang, Y. Autonomous driving on curvy roads without reliance on frenet frame: A cartesian-based trajectory planning method. IEEE Trans. Intell. Transp. Syst. 2022, 23, 15729–15741.

[12] Zhu, D.; Sun, J. A new algorithm based on Dijkstra for vehicle path planning considering intersection attribute. IEEE Access 2021, 9, 19761–19775.

[13] Zhaoying, L.; Ruoling, S.; Zhao, Z. A new path planning method based on sparse A* algorithm with map segmentation. Trans. Inst. Meas. Control 2022, 44, 916–925.

[14] Schmid, L.; Pantic, M.; Khanna, R.; Ott, L.; Siegwart, R.; Nieto, J. An efficient sampling-based method for online informative path planning in unknown environments. IEEE Robot. Autom. Lett. 2020, 5, 1500–1507.

[15] Noreen, I.; Khan, A.; Habib, Z. Optimal path planning using RRT* based approaches: A survey and future directions. Int. J. Adv. Comput. Sci. Appl. 2016, 7, 97–107.

[16] Talamino, J.P.; Sanfeliu, A. Anticipatory kinodynamic motion planner for computing the best path and velocity trajectory in autonomous driving. Robot. Auton. Syst. 2019, 114, 93–105.

[17] Benko Loknar, M.; Klančar, G.; Blažič, S. Minimum-Time Trajectory Generation for Wheeled Mobile Systems Using Bézier Curves with Constraints on Velocity, Acceleration and Jerk. Sensors 2023, 23, 1982.

[18] Chen, F.; Wang, M.Y. Design optimization of soft robots: A review of the state of the art. IEEE Robot. Autom. Mag. 2020, 27, 27–43

[19] Takishima, Y.; Yoshida, K.; Khosla, A.; Kawakami, M.; Furukawa, H. Fully 3D-printed hydrogel actuator for jellyfish soft robots. ECS J. Solid State Sci. Technol. 2021, 10, 037002.

[20] Zakharov, L.A.; Martyushev, D.A.; Ponomareva, I.N. Predicting dynamic formation pressure using artificial intelligence methods. J. Min. Inst. 2022, 253, 23–32.

[21] Filippov, E.V.; Zaharov, L.A.; Martyushev, D.A.; Ponomareva, I.N. Reproduction of reservoir pressure by machine learning methods and study of its influence on the cracks formation process in hydraulic fracturing. J. Min. Inst. 2022, 258, 924–932.

[22] Islamov, S.; Grigoriev, A.; Beloglazov, I.; Savchenkov, S.; Gudmestad, O.T. Research Risk Factors in Monitoring Well Drilling—A. Case Study Using Machine Learning Methods. Symmetry 2021, 13, 1293.

[23] Montiel, J.; Halford, M.; Mastelini, S.M.; Bolmier, G.; Sourty, R.; Vaysse, R.; Zouitine, A.; Gomes, H.M.; Read, J.; Abdessalem, T.; et al. River: Machine learning for streaming data in python. J. Mach. Learn. Res. 2021, 22, 4945–4952.

[24] Sun, W.; Akashi, N.; Kuniyoshi, Y.; Nakajima, K. Physics-informed recurrent neural networks for soft pneumatic actuators. IEEE Robot. Autom. Lett. 2022, 7, 6862–6869.

[25] Ibrahim, A., Bala, S.I., Ahmed, I., Ibrahim, M.J., Jarad, F., "Numerical Construction of Lyapunov Functions Using Homotopy Continuation Method" (2022) Advances in the Theory of Nonlinear Analysis and its Applications, 6 (3), pp. 354-363.

[26] Kessi, A., Laadj, T., Yahi, M., "On the integration of first order nonlinear differential equations and the conditions of Fuchs' theorem", (2022) Advances in the Theory of Nonlinear Analysis and its Applications, 6 (3), pp. 347-353.

[27] Goyal, Dinesh , Kumar, Anil , Gandhi, Yatin & Khetani, Vinit (2024) Securing wireless sensor networks with novel hybrid lightweight cryptographic protocols, Journal of Discrete Mathematical Sciences and Cryptography, 27:2-B, 703–714, DOI: 10.47974/JDMSC-1921

[28] P. Deshpande, R. Dhabliya, D. Khubalkar, P. A. Upadhye, K. A. Wagh and V. Khetani, "Alzheimer Disease Progression Forecasting: Empowering Models Through hybrid of CNN and LSTM with PSO Op-Timization," 2024 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 2024, pp. 1-5, doi: 10.1109/ESCI59607.2024.10497309.