# A Method for Intruder UAV Pilot Localization based on Neural Networks using Intercepting Drone Constellations

**Markus Schatten, Neven Vrček**
University of Zagreb
Faculty of Organization and Informatics
Pavlinska 2, 42000 Varaždin, Croatia
markus.schatten, neven.vrcek@foi.unizg.hr

***Abstract.*** *In this paper we introduce a method for the localization of intruder drone pilot using an intercepting 3-drone constellation. Using datasets collected during field experiments we have analyzed various machine learning (ML) methods including linear regression, decision tree, random forest, gradient boosting, support vector, and neural networks in order to implement a predictive model that will be able to predict the distance of each intercepting drone to the intruders drone pilot based on various parameters, most prominently Received Signal Strength Indicator (RSSI). Using this predicted distance and Global Positioning System (GPS) coordinates of the intercepting drones we are able to trilaterate the coordinates of the drone pilot. The analysis has shown that the neural network model gives the best results with an mean squared error of 0.12185.*

**Keywords.** pilot localization, drone constellations, intruder interception, neural network, unmanned aerial vehicles

## 1 Introduction

Unmanned Aerial Vehicles (UAVs), commonly known as drones, have seen a significant increase in popularity and usage in recent years (Idrissi et al., 2022). They are employed in a wide range of applications, from aerial photography and videography to delivery services and agricultural monitoring (Schatten, 2015). However, the proliferation of drones has also raised safety and security concerns. One of the most pressing issues is the unauthorized or malicious use of drones in sensitive areas such as airports, military bases, and public spaces. The ability to locate the pilot of an intruding drone is crucial in mitigating these risks and responding effectively to potential threats.

In this context, this paper presents a novel method for intruder drone pilot localization using an intercepting 3-drone constellation. Our approach leverages machine learning techniques to predict the distance of each intercepting drone to the intruder's drone pilot based on various parameters, most prominently Received Signal Strength Indicator (RSSI). By combining these predicted distances with the GPS coordinates of the intercepting drones, we are able to trilaterate the coordinates of the drone pilot.

This work is part of the ORKAN project - Unmanned Aerial Vehicle policy ecosystem[1], a comprehensive initiative aimed at developing advanced methods and technologies for drone detection, tracking, and neutralization. Our previous work (Magdalenić et al., 2022; Tomaš and Posarić, 2022) has made significant contributions to this area. In (Tomaš and Posarić, 2022), we proposed a novel approach to WiFi access point localization in an urban multi-sensor environment. This approach significantly influenced our current research allowing us to localize access points on UAVs. In (Magdalenić et al., 2022), we further improved this approach by integrating a a specially designed hardware device to allow for capturing (intruder) drone signals by using a drone constellation.

In our analysis, we compared several machine learning methods, including linear regression, decision tree, random forest, gradient boosting, support vector, and neural networks (multilayer perceptron). Our results show that the neural network model provides the best performance, with a mean squared error of 0.12185. This finding underscores the potential of machine learning, and neural networks in particular, in enhancing the capabilities of drone interception systems and improving safety and security in drone-affected areas.

The remainder of this paper is organized as follows. Section 2 provides a detailed description of the proposed method. Section 3 presents the results of our analysis and comparison of different machine learning models. Finally, Section 4 concludes the paper and outlines directions for future work.

## 2 Related Work

Apart from our research, to the best of our knowledge, there have been very few studies which try to develop methods for drone pilot localization. For example,

[1] https://orkan.foi.hr

(Zhen, 2019) propose a direction of arrival (DOA) estimation algorithm that uses support vector regression and integrates DOAs acquired with array sensors on a reconnaissance aircraft to estimate the location of illegal unmanned aerial vehicle operators. The algorithm performs well in circumstances where signals impinge on the sensor array with a small-angle interval, as well as in conditions of small samples and low signal-to-noise ratio.

On the other hand, (Mashhadi et al., 2020) presents a method for autonomously locating drone operators by tracking the drone's path in the sky using a deep neural network. The model, trained on a realistic simulation environment, achieved 73% accuracy in predicting the operator's location, offering a promising alternative to traditional methods that use RF sensors, which have certain limitations.

Basak and Scheers, 2018 discusses the development of a passive radio system that can detect and localize mini remotely piloted aircraft systems (RPAS) and their operators in three dimensions. The system uses Goodness-of-Fit (GoF) based spectrum sensing for signal detection and the MUSIC algorithm for direction of arrival (DoA) estimation, offering a low-cost, highly sensitive solution capable of detecting multiple drones and operators simultaneously across a wide frequency range.

Our approach to drone pilot localization involves the use of a three-drone constellation that follows an intruder drone and collects signal strength and GPS data. Using this approach we are able to achieve a more accurate result in pilot localization based on field experiment data using machine learning models.

# 3 Methodology

As already mentioned six machine learning methods have been tested for performance. Bellow is a short overview of each of the methods.

## 3.1 Linear Regression

Linear regression is a foundational statistical method used to model the relationship between a dependent variable $y$ and one or more independent variables $X$. The primary goal is to find the best fit straight line that accurately predict the output values within a range. The linear equation is given by:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

Where $\beta_0$ is the intercept, $\beta_1, \beta_2, \ldots, \beta_p$ are the coefficients of the independent variables, and $\epsilon$ is the error term. The coefficients are estimated using methods like the least squares.

## 3.2 Decision Tree

A decision tree is a flowchart-like structure where each internal node represents a feature (or attribute), each branch represents a decision rule, and each leaf node represents an outcome. The primary goal of using a decision tree is to create a training model that can predict the class or value of the target variable by learning simple decision rules inferred from prior data.

The decision to split at each node is based on some criterion, such as maximizing the information gain or minimizing the Gini impurity. The tree is constructed in a top-down recursive manner.

## 3.3 Random Forest

Random Forest is an ensemble learning method that combines multiple decision trees to produce a more accurate and stable prediction. It's essentially a collection of decision trees trained with subsets of the dataset. The main principle behind the ensemble model is that a group of weak learners come together to form a strong learner. In the case of random forests, the model creates an "ensemble" of decision trees from a random subset of the data and aggregates them to produce a final result.

## 3.4 Gradient Boosting

Gradient Boosting is an iterative ensemble method that adjusts for the errors of the previous models in the ensemble. It builds the model in a stage-wise fashion, and it generalizes them by allowing optimization of an arbitrary differentiable loss function. At each stage, the model introduces a new tree that corrects the errors of the previously built ensemble. This is done by fitting the new tree to the negative gradient of the loss function, effectively moving closer to the optimal model.

## 3.5 Support Vector Machines

Support Vector Machines (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. It operates by finding the hyperplane that best divides a dataset into classes. The best hyperplane is the one that represents the largest separation, or margin, between the two classes. The equation of the hyperplane is given by:

$$\mathbf{w} \cdot \mathbf{x} + b = 0$$

Where $\mathbf{w}$ is the normal vector to the hyperplane and $b$ is the bias. The objective of the SVM algorithm is to find the optimal values for $\mathbf{w}$ and $b$.

## 3.6 Neural Networks - Multilayer Perceptron

Neural Networks, specifically the Multilayer Perceptron (MLP), consist of layers of interconnected nodes

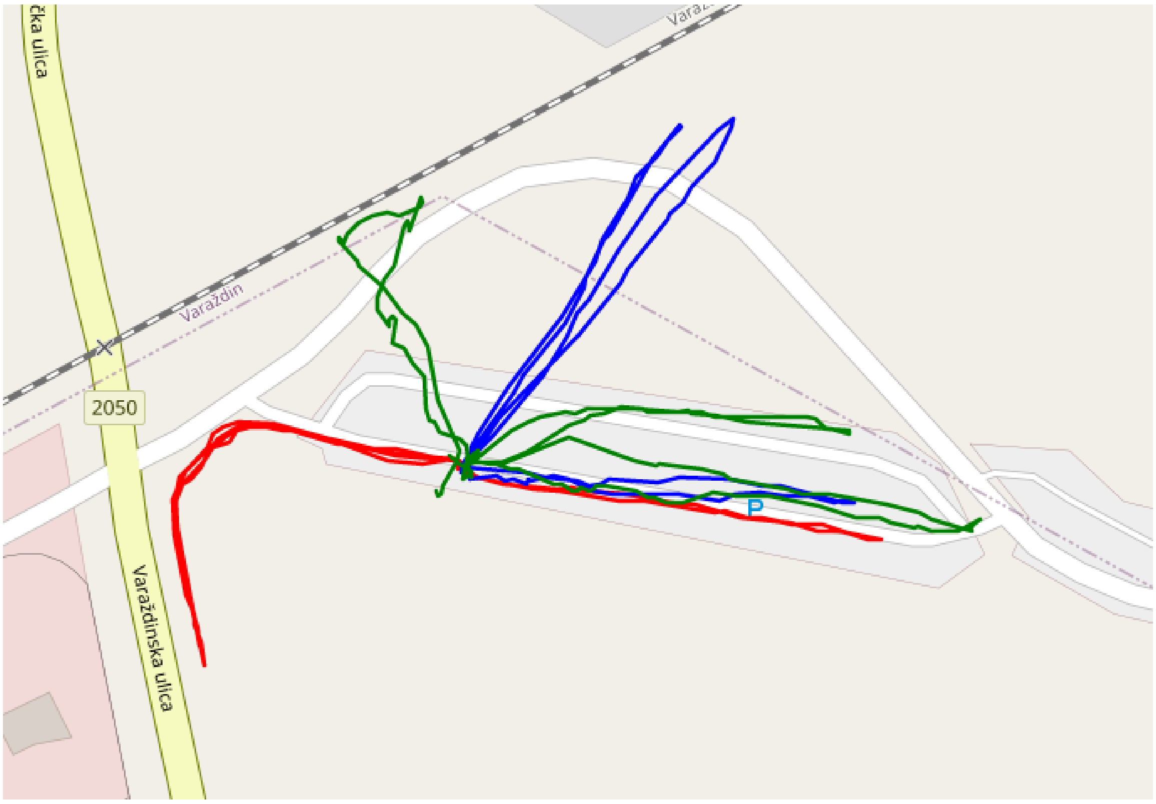



**Figure 1:** Visualization of the GPS data collected during one of field experiments (each color representing one drone of the 3-drone constellation)

or "neurons". Each connection between neurons has a weight, and each neuron has an activation function. The primary goal of a neural network is to receive inputs, process them in hidden layers using weights that are adjusted during learning, and produce an output. The output $y$ of a neuron is given by:

$$y = f\left(\sum_i w_i x_i + b\right)$$

Where $f$ is the activation function, $w_i$ are the weights, $x_i$ are the inputs, and $b$ is the bias. The network "learns" by adjusting the weights to minimize the error in its predictions.

# 4 Experiments

The 3-drone constellation is designed such that each drone in the constellation is equipped with a specially designed hardware device that allows for capturing the intruder drone's signals.

The localization process can be divided into two main steps: distance prediction and trilateration.

## 4.1 Distance Prediction

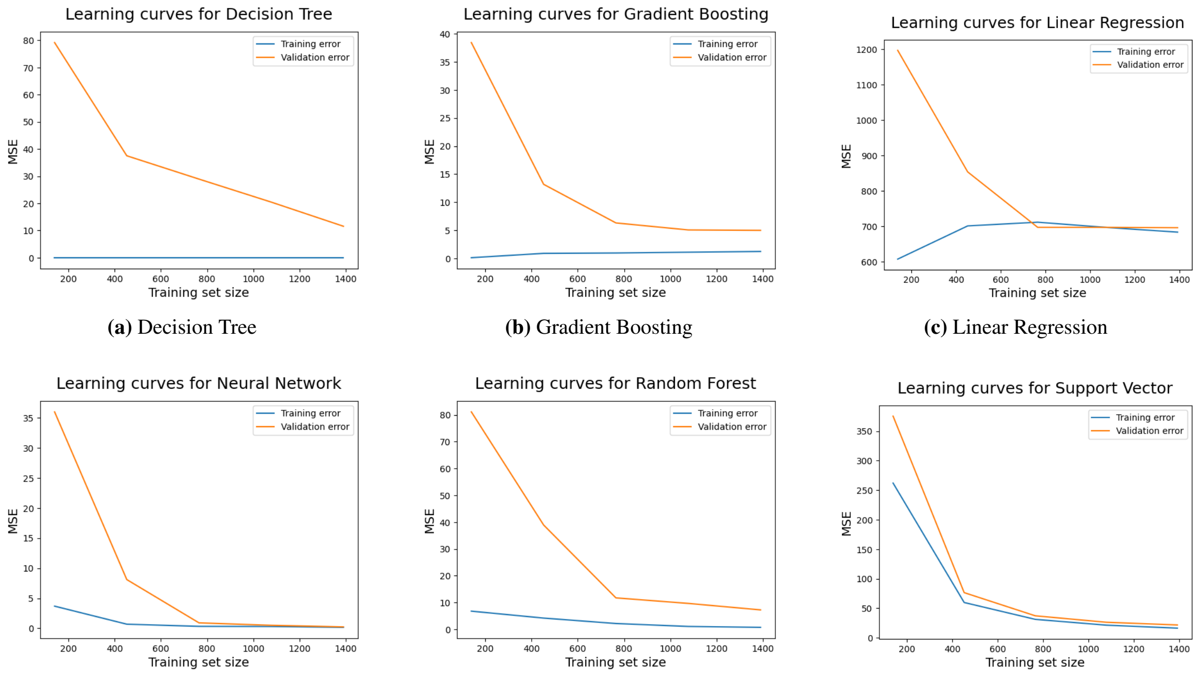
The first step in the localization process is to predict the distance between the pilot and each drone in the constellation. For this purpose, we use a machine learning model that takes as input the collected signal strength and GPS data and outputs a predicted distance. In our case, we found that a neural network model provides the best performance.

The neural network model is trained on a dataset collected during field experiments (see figure 1), with the signal strength and GPS data as the input features and the actual distance as the target variable.

The dataset consists of signal strength and GPS data collected by a three-drone constellation following an intruder drone (a total of 2174 datapoints). The dataset includes the following features: time, number of satellites, longitude, latitude, altitude, Horizontal Dilution of Precision (HDOP), maximum RSSI, average RSSI, and distance. The target variable is the distance between each drone and the pilot.

The machine learning models were implemented and trained using Python in Google Colab, an online platform that allows for the execution of Python code in the cloud. The Python libraries used in the implementation include pandas for data manipulation, numpy for numerical computations, and scikit-learn for machine learning.

In order to determine the best performing machine learning model, we have trained six ML models. For each machine learning model, the dataset was first split into a training set and a test set using a 80-20 split. The models were then trained on the training set and evaluated on the test set.




**(a)** Decision Tree **(b)** Gradient Boosting **(c)** Linear Regression

**(d)** Neural Network **(e)** Random Forest **(f)** Support Vector

**Figure 2:** Training curve visualizations of the machine learning models.

| Model | Best Parameters | MSE |
|---|---|---|
| Linear Regression | N/A | 634.05166 |
| Decision Tree | {'max_depth': 20, 'min_samples_split': 2} | 6.93033 |
| Random Forest | {'max_depth': None, 'min_samples_split': 2, 'n_estimators': 200} | 2.11822 |
| Gradient Boosting | {'learning_rate': 0.1, 'max_depth': 3, 'n_estimators': 200} | 3.23053 |
| Support Vector | {'C': 10.0, 'kernel': 'rbf'} | 11.13768 |
| Neural Network | {'activation': 'relu', 'hidden_layers': (50, 50), 'learning_rate_init': 0.01} | 0.12185 |

**Table 1:** Model performance comparison with best parameters

The linear regression model was implemented using the LinearRegression class in scikit-learn. The decision tree model was implemented using the DecisionTreeRegressor class, and the random forest model was implemented using the RandomForestRegressor class. The gradient boosting model was implemented using the GradientBoostingRegressor class, and the support vector model was implemented using the SVR class. The neural network model was implemented using the MLPRegressor class.

The hyperparameters of each model were tuned using grid search with cross-validation. The performance of each model was evaluated using the mean squared error (MSE) between the predicted distances and the actual distances. The individual training curves of each model are shown on figure 2.

## 4.2 Trilateration

The second step in the localization process is to estimate the GPS coordinates of the pilot based on the predicted distances. This is done using a trilateration approach, which is a method used to determine the location of a point by measuring distances to it from known points at fixed distances apart.

Given the GPS coordinates $(\text{lat}_i, \text{lon}_i)$ and the predicted distance $d_i$ for each drone $i$ in the constellation, we first calculate the bearing $\theta_{ij}$ between each pair of drones $i$ and $j$ using the formula:

$$\theta_{ij} = \text{atan2}(\text{lon}_j - \text{lon}_i, \text{lat}_j - \text{lat}_i) \tag{1}$$

We then calculate the destination point $(\text{lat}_{ij}, \text{lon}_{ij})$ for each drone $i$ in the direction of each other drone $j$ at a distance $d_i$ using the formula:

$$(\text{lat}_{ij}, \text{lon}_{ij}) = \text{destination\_point}(\text{lat}_i, \text{lon}_i, \theta_{ij}, d_i) \tag{2}$$

Finally, we calculate the centroid of the destination points as the estimated GPS coordinates of the pilot:

$$(\text{lat}_{\text{pilot}}, \text{lon}_{\text{pilot}}) = \frac{1}{3}\sum_{i=1}^{3}(\text{lat}_{ij}, \text{lon}_{ij}) \tag{3}$$


**Figure 3:** Detected pilot location

This process is repeated for each time step, and the final estimate of the pilot's coordinates is obtained by taking the mean of all the estimates.

## 5 Results

The performance of the ML models varied significantly (see table 1). The neural network model achieved the best performance with an MSE of 0.121846, followed by the random forest model with an MSE of 2.118221. The gradient boosting model achieved an MSE of 3.230534, and the decision tree model achieved an MSE of 6.930330. The support vector model and the linear regression model achieved higher MSEs of 11.137681 and 634.051665, respectively.

These results indicate that the neural network model provides the most accurate predictions of the distance between each drone and the pilot. This model can therefore be used to effectively trilaterate the coordinates of the drone pilot. Figure 3 shows a visualization of the detected pilots coordinates based on the field experiment data.

## 6 Conclusion

In this paper, we presented a novel method for localizing the pilot of an intruding drone using an intercepting 3-drone constellation. The method leverages machine learning techniques to predict the distance of each intercepting drone to the intruder's drone pilot based on various parameters, most prominently RSSI. By combining these predicted distances with the GPS coordinates of the intercepting drones, we are able to trilaterate the coordinates of the drone pilot.

Our analysis showed that the neural network model provides the best performance, with a mean squared error of 1.994414. This finding underscores the potential of machine learning, and random forests in particular, in enhancing the capabilities of drone interception systems and improving safety and security in drone-affected areas.

However, it is important to note that the proposed method is limited to scenarios where the intruder drone is operated using WiFi. This is a constraint that could limit the applicability of the method in some contexts. Nevertheless, the method offers significant benefits in terms of its ability to accurately and efficiently locate the pilot of an intruding drone, which is crucial for mitigating risks and responding effectively to potential threats.

The proposed method has a wide range of potential applications, including in airports, military bases, and public spaces, where unauthorized or malicious use of drones is a major concern. It could also be used in other contexts where the ability to locate the operator of a drone is important, such as in search and rescue operations or in the monitoring of wildlife.

In conclusion, the proposed method represents a significant contribution to the field of drone pilot localization. It offers a practical and effective solution to a pressing problem, and has the potential to significantly enhance safety and security in a wide range of contexts.

## Acknowledgement

This work has been fully supported by the Croatian Science Foundation under the project IP-2019-04-4864.


## References


Basak, S., & Scheers, B. (2018). Passive radio system for real-time drone detection and doa estimation. *2018 International Conference on Military Communications and Information Systems (ICMCIS)*, 1–6.

Idrissi, M., Salami, M., & Annaz, F. (2022). A review of quadrotor unmanned aerial vehicles: Applications, architectural design and control algorithms. *Journal of Intelligent & Robotic Systems*, *104*(2), 22.

Magdalenić, I., Vrček, N., Tomaš, B., & Guðmundsson, K. S. (2022). Localization of drone operator using unmanned aerial vehicles: Preliminary research. *Central European Conference on Information and Intelligent Systems*, 335–339.

Mashhadi, E., Oren, Y., & Weiss, G. (2020). Can the operator of a drone be located by following the drone's path? *Cyber Security Cryptography and Machine Learning: Fourth International Symposium, CSCML 2020, Be'er Sheva, Israel, July 2–3, 2020, Proceedings 4*, 85–93.

Schatten, M. (2015). *Multi-agent based traffic control of autonomous unmanned aerial vehicles* (tech. rep.). Artificial Intelligence Laboratory, University of Zagreb.

Tomaš, B., & Posarić, L. (2022). Wifi access point localization in urban multi-sensor environment. *Central European Conference on Information and Intelligent Systems*, 443–448.

Zhen, J. (2019). Localization of unmanned aerial vehicle operators based on reconnaissance plane with multiple array sensors. *IEEE Access*, *7*, 105354–105362.