# UAV Detection System with Multiple Acoustic Nodes Using Machine Learning Models

Bowon Yang
*Computer and Information Technology*
*Purdue University*
West Lafayette, United States
yang1270@purdue.edu

Eric T. Matson
*Computer and Information Technology*
*Purdue University*
West Lafayette, United States
ematson@purdue.edu

Anthony H. Smith
*Computer and Information Technology*
*Purdue University*
West Lafayette, United States
ahsmith@purdue.edu

J. Eric Dietz
*Computer and Information Technology*
*Purdue University*
West Lafayette, United States
jedietz@purdue.edu

John C. Gallagher
*Computer Science and Engineering*
*Wright State University*
Dayton, United States
john.gallagher@wright.edu

***Abstract*—Class 1 unmanned aerial vehicles (UAVs), known as drones, have become popular and accessible, which makes them tools for malicious purposes. As a result, there is an increasing demand for an effective defense system that can detect UAVs. In this paper, a UAV detection system with multiple acoustic nodes using machine learning models is proposed along with an empirically optimized configuration of the nodes for deployment. Features including Mel-frequency cepstral coefficients (MFCC) and short-time Fourier transform (STFT) were used for training. Support vector machines (SVM) and convolutional neural networks (CNN) were trained with the data collected in person. Experiments were done to evaluate models' ability to find the path of the UAV that was flying. Sensing nodes were placed in four different configurations and the best of test set was chosen which maximizes the detection range without blind spots. STFT-SVM model showed the best performance and a semi-circle formation with 75 meters distance between a node and the protected area is found to be the optimized configuration.**

***Index Terms*—Counter unmanned aerial system, UAV detection, machine learning, audio detection, drone security**

## I. Introduction

Unmanned Aerial Vehicles (UAVs), known as drones, have become increasingly popular over the last two years. Around 1.3 million UAVs were sold in 2015 for recreational purposes and 3 million in 2017, more than doubling the number [1]. Although most of the UAVs are used for recreation, they pose a threat to security because they can be used for malicious purposes like drug smuggling [2] and terrorist attacks [3]. As a result, academia and industry have started to build systems to detect UAVs and take measures, which are called Counter Unmanned Aerial Systems (CUAS). An effective detection phase is crucial to perform the counter measures successfully. In this paper, an affordable UAV detection system with multiple acoustic nodes using machine learning models is proposed. The system can be easily deployed in public and scaled by adding more, very affordable, nodes.

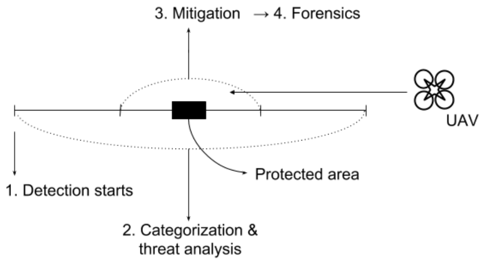

3. Mitigation → 4. Forensics
UAV
Protected area
1. Detection starts
2. Categorization & threat analysis

Fig. 1: The overview of the CUAS by Purdue University

## II. Related Work

### A. Counter Unmanned Aerial Systems (CUAS)

Typical missions of a CUAS can be illustrated by the sequence of four tasks as shown in figure 1. The sequence starts with initial detection of a UAV and then leads to the analysis step, where the system categorizes the type of the UAV, and analyze the threat using visual analysis or sound signatures. If it is identified as a threat, the UAV is mitigated using counter measures, depending on what kind of strategies the system is using. At the last forensics step, the UAV caught by the counter measures is analyzed and further investigation is performed.

Researchers at Purdue University realized a fully-autonomous CUAS that intercepts an unauthorized UAV [4]. The system detects, tracks, and takes down class 1 UAVs using a hunter UAV attached with a net. A radar-based station starts detecting and tracking the target UAV when it gets into the restricted zone. The hunter UAV keeps updated with the position of the target UAV and approaches the target UAV with the attached net to catch it. Similarly, a vision-based CUAS with a hunter UAV with a net was introduced in [5]. The success rate of tracking the target UAV is 87.23% and that of interceptions with a net attached is 81.3% on average.

978-1-5386-9245-5/19/$31.00 ©2019 IEEE
DOI 10.1109/IRC.2019.00103

### B. UAV Detection Using Radar

Radar is an object detection system that sends radio waves and detects the echo returned from the target [6]. Traditionally, radar has been mainly used for air defense systems to track aircrafts or missiles. These radar-based detection systems are optimized for bigger objects and highly expensive [7]. Detection of class 1 UAVs is challenging because they have very small radar cross sections (RCS) as well as lower speed and lower altitude [8]. However, research shows the possibility of a low-cost UAV detection and tracking system by using a synthetic aperture radar (SAR) on top of an unmanned ground vehicle [9]. The system uses frequency modulated continuous wave (FMCW) and operates at a center frequency of 2.4274 GHz with a bandwidth of 260 MHz. The range of detection is up to 50 meters.

### C. UAV Detection Using LiDAR

Lidar (Light Detection And Ranging) sensors transmit laser beams and analyze the beams reflected back from the target [10]. Lidar is frequently used in underwater detection [10] and forestry monitoring [11]. Although it has higher resolution and accuracy, the cost is high and the results are vulnerable to weather conditions, which makes it unpractical for UAV detection [5]. However, a group of researchers implemented a UAV detection system using an affordable lidar sensor mounted on a UAV [12]. The system was deployed indoors and could detect a UAV as well as the velocity and speed.

### D. UAV Detection Using Computer Vision

UAVs can be detected using object detection technologies in computer vision. A. Rozantsev [13] presents research on a UAV detection system using a camera on board a UAV. The motion information is extracted to detect flying objects by motion compensation. Each frame is processed to decide if the object is a target object using Convolutional Neural Networks. The average precision of their system is 0.732 according to the author.

Using computer vision to detect UAVs is available with inexpensive optical sensors compared to radar or lidar and inherently enables accurate tracking. However, this can be challenging when the weather condition is bad or there is a lot of noise. Also, it can consume a lot of resources depending on the algorithm, which makes it hard to get detection results immediately with limited resources.

### E. UAV Detection Using Acoustic Sensors

Using acoustic sensors is another inexpensive and easily deployable way to detect UAVs. Audio signal processing can be more economic than vision-based methods because it requires less computing resources than image processing in terms of data size, and the sensors are less expensive than cameras. Also, a wider range of detection is possible, meaning that UAVs can be detected from greater distances and the sound signals can be recognized from any angle.

A paper published in 2008 [14] demonstrates the possibility of an inexpensive UAV detection system using acoustic nodes. The system makes use of a microphone array and beamforming, which allows for determination of position and direction of the target UAV. The array consists of 24 microphones aligned in a straight line with the sensors 10 inches apart from each other. Although this implementation still requires initial calibration to get the position of the array, this shows an inexpensive way to detect UAVs using well-known and simple techniques.

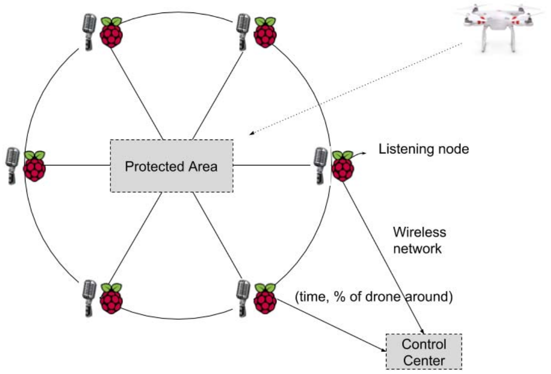


Fig. 2: Overview of the system hardware

Recently, as machine learning has been demostrated effective in the audio signals domain, researchers used different learning algorithms to develop detection modules. An implementation from [15] detects UAVs in real-time using deep learning models, including recurrent neural networks and convolutional neural networks, as well as a Gaussian mixture model as a baseline algorithm. The study focuses on event sound detection in a noisy urban area using binary classification with Mel-frequency Cepstrum Coefficients (MFCC) as a feature.

## III. Methodology

### A. System Overview

The proposed acoustic UAV detection system consists of multiple listening nodes and a control center as in figure 2. A listening node, or a node, is a computing unit that can detect the UAV sound. Nodes are placed on a circle surrounding the protected area. The control center interacts with the nodes, receiving detection results through wireless network.

Each node runs a process to keep listening to the ambient sound using a microphone and a sound card installed on a node. At the same time, another process runs the detection module per frame using a machine learning model implanted in the node. Then, the node transmits the detection results to the control center so that the user can monitor current status. An access point is established near the center and the nodes send data to the control center over 802.11. The wireless communication is facilitated by wide range network adapters. For data transmission, WebSocket [16] is used to allow users to monitor the detection results with web browsers.

### *B. Data Collection*

The dataset was collected in person and was used to train and test the models. For training data, the audio was recorded using the six listening nodes placed near each other. For testing data, the six nodes recorded audio data, placed on a circle or a half circle in certain configurations which will be mentioned in the section III-E. Data collection was conducted in a large grass area on sunny and windy days. The UAV was flown 0 to 10 meters above the node, and the distance between the UAV and each node was up to 20m. Only one type of UAVs was used in this research, which is AR Drone 2.0 [17] by Parrot.

The data for the negative class is raw audio data of the environment noise without UAVs flying around while the data in the positive class includes raw audio files that the UAV can be heard as well as background noise. The types of noise recorded in the training audio files include insect sounds, occasional human voices, airplanes and wind, which make up 14 minutes in total.

### *C. Feature Extraction*

Mel-frequency cepstral coefficients (MFCC) and short-time Fourier transform (STFT) are the two types of features used in this research. MFCC is a feature set that reflects human perception of sounds [18]. It is commonly used in audio classification fields, and it is successfully used with machine learning approaches [19].

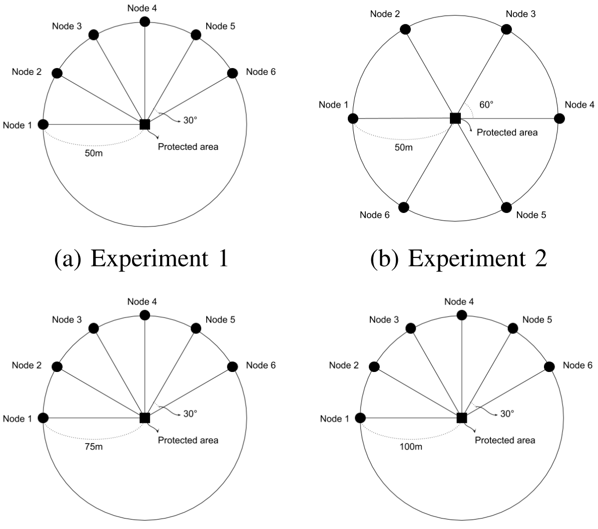
Fast Fourier transform (FFT) is to transform signals in the time domain into the frequency domain. STFT is a feature set in the time domain with magnitudes of each frequency band by stacking FFT of short signals. STFT is an intermediate feature compared to MFCC in the sense that MFCC compresses signals as it represents them with a set of coefficients. STFT contains more information as well as noise so MFCC has been used more commonly. Recently, with the advent of deep neural networks, STFT is revisited as deep learning models can handle more complicated data than traditional models like Gaussian mixture models [20].

### *D. Machine Learning Models*

A Support Vector Machine (SVM) is a supervised classifier that learns the boundaries among classes [21]. SVM is based on kernel functions that "map the input vectors into a very high-dimensional feature space through some nonlinear mapping chosen a priori" [22]. Radial basis function (RBF) kernel is used in this research for nonlinear classification. The input space is mapped into the feature space by nonlinear mapping $\Phi$. When $K(\mathbf{x}, \mathbf{y})$ is the RBF kernel on the input space $(x, y)$,

$$K(\mathbf{x}, \mathbf{y}) = exp(-\gamma||\mathbf{x} - \mathbf{y}||^2)), \tag{1}$$

where parameter gamma $\gamma$ is the free parameter, meaning more variance of the kernel function when the value is smaller. Another parameter C is taken into consideration, which is a parameter for the soft margin function. If C is big, there is a trade off with correctness of classification.



(a) Experiment 1

(b) Experiment 2

(c) Experiment 3

(d) Experiment 4

Fig. 3: Experimental configurations

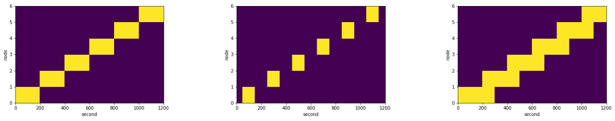
| Exp # | Radius(m) | Angle(°) | # Nodes |
|---|---|---|---|
| 1 | 50 | 30 | 6 |
| 2 | 50 | 60 | 6 |
| 3 | 75 | 30 | 6 |
| 4 | 100 | 30 | 6 |

TABLE I: Set of Experiments

Another machine learning model that is used in the system is convolutional neural networks (CNN). CNN has brought significant success in pattern recognition or classification in computer vision. Researchers also approach audio classification problem with CNN as audio signals can be represented in two dimensions like an image [23], [24]. CNN is composed of different layers including an input layer, convolutional layers and pooling layers, fully connected layers, and an output layer [25]. The parameters that should be configured to build a CNN include number of filters and their size for convolutional layers, pool size and stride size for pooling layers and output size for fully-connected layers. Also, an activation function should be selected as well as learning rate and dropout rate. Recitified linear units (ReLU) was used as the activation function throughout the network in this paper.

### *E. Experiment Setup*

After the training phase, tests were done to evaluate the model and to determine the configuration of the nodes. Test data was collected by placing nodes on possible formations and recording the audio of the UAV flying around the nodes. The test data was put into trained models, generating a series of detection results. The results were plotted as a color map and the model was decided by the visibility of the path of the UAV on the color map. Then using the selected model, a formation is chosen that maximizes the detection area without



(a) (b) (c)

Fig. 4: Virtual results of six nodes that are plotted with dark and bright color each representing negative and positive results

blind spots. Circular configurations surrounding the protected area are proposed considering these parameters below:

- *Radius* is the distance between a node and the protected area that can affect the range of detection.
- *Angle* measures how far it is between nodes.
- *Height* is the vertical distance between the UAV and the ground where nodes are set up.

Four experiments were performed varying the radius and angle with the maximum height fixed as 3m as in table I. And the nodes were placed as in figure 3 accordingly. The way the tests are done is to set up nodes and fly the UAV following the rim of the circle and around the node. The trajectory was designed to find the configurations and to cover all angles of the sound. Audio data were collected with nodes in these configurations. Each audio file was fed into the detection models and the color maps were generated. If the models are valid, the color maps should have a noticeable pattern. When the patterns are visible, the best configuration can be chosen by comparing the patterns of each configuration.

For example, figure 4 shows virtual test results assuming that the quality of the model stays the same for every device with the clear boundary for detection. X axis is time and Y axis is the node number, and the bright area means positive results and the dark area means negative results. When the UAV flies following the nodes from node 1 to node 6, the bright areas should make a diagonal form as all three of the figures. From these color maps, optimized configurations can be found by the distance between each bright area. Figure 4 (a) is the case when the nodes are as apart as the detection range, (b) is when the nodes are farther than detection range, and (c) is when the nodes are placed closer than the range.

## IV. Results

### A. Training

For SVM, scikit-learn library is used for implementation. Modules such as `sklearn.svm` and `sklearn.model_selection.GridSearchCV` are primarily used. C and gamma are found through the grid search method, exhaustively searching for the best performing parameter pair over given parameter values as below.

- C: $C = 10^i$, where $i = 1, 2, ..., 14, 15$
- gamma: $\gamma = 10^i$, where $i = -15, -14, ..., -2, -1$

For each parameters, the model was cross-validated by the factor of three. The training process is repeated for MFCC and STFT. The training results are shown in table II.

| Feature | C | Gamma | F1-score |
|---|---|---|---|
| MFCC | $10^5$ | $10^{-10}$ | 0.779 |
| STFT | $10^{12}$ | $10^{-12}$ | 0.787 |

TABLE II: Results of training SVM

| Layer | Component | Parameter |
|---|---|---|
| Convolutional layer 1 | Kernel size<br>Initializer<br>Number of filters<br>Activation function | [13,13]<br>Xavier<br>16<br>ReLU |
| Pooling layer 1 | Pool size<br>Stride size<br>Padding | [3, 3]<br>2<br>'SAME' |
| Convolutional layer 2 | Kernel size<br>Initializer<br>Number of filters<br>Activation function<br>Padding | [3, 3]<br>Xavier<br>16<br>ReLU<br>'SAME' |
| Pooling layer 2 | Pool size<br>Stride size<br>Padding | [3, 3]<br>2<br>'SAME' |
| Dense layer 3 | Output size<br>Activation | 100<br>ReLU |
| Dropout layer 3 | Dropout rate | 0.5 |
| Dense layer 4 | Output size<br>Activation | 10<br>ReLU |
| Dropout layer 4 | Dropout rate | 0.5 |

TABLE III: Convolutional Neural Network Parameters for MFCC

Training a CNN is based on heuristics using single fold where the parameter values are selected manually because exhaustively searching for the best combination is not feasible. The considered factors are number of layers, size of filters, number of filters, learning rate, and dropout. Table III shows the final MFCC-CNN architecture with two convolutional layers and two dense layers. Considering that 16 coefficients were used for MFCC, the size of the filter, [13, 13], can be seen as too large. However, it gives higher accuracy than using a smaller filter. It came from the intuition that the patterns are not repeated along the axis of MFCC because the frequency range is unique to each sound. Training iteration is set to 1,000, and the process was stopped when the cost curve and validation curve started to become constant.

Similar to the MFCC-CNN model, the STFT-CNN model is built with two convolutional layers and two dense layers. The major difference is the filter shape of the first convolutional layer because the STFT has more values per frame. The filter size of the model is [3, 3] because for STFT model, rectangular-shaped filters did not converge during training as the values are more diverse.

### B. Testing

Test results for each model and experiment are shown in color maps from figure 6 (i) to (iv). Figure 5 shows the mapping between the color maps and the actual trajectory of the UAV flying from node 1 to node 6. The X-axis of the color map is minutes and the Y-axis is the node number. Dark areas represent negative detection results and bright areas represent positive results. Each red box indicates period when

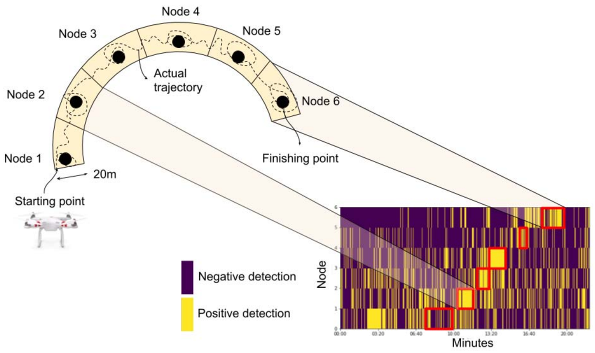


Fig. 5: Mapping between result color maps and trajectory

UAV sounds are significant, meaning the UAV is close to the corresponding node. For example, in figure 5, there are six boxes and two of them are mapped to the trajectory. When the detection is successful, the red box areas should match bright areas.

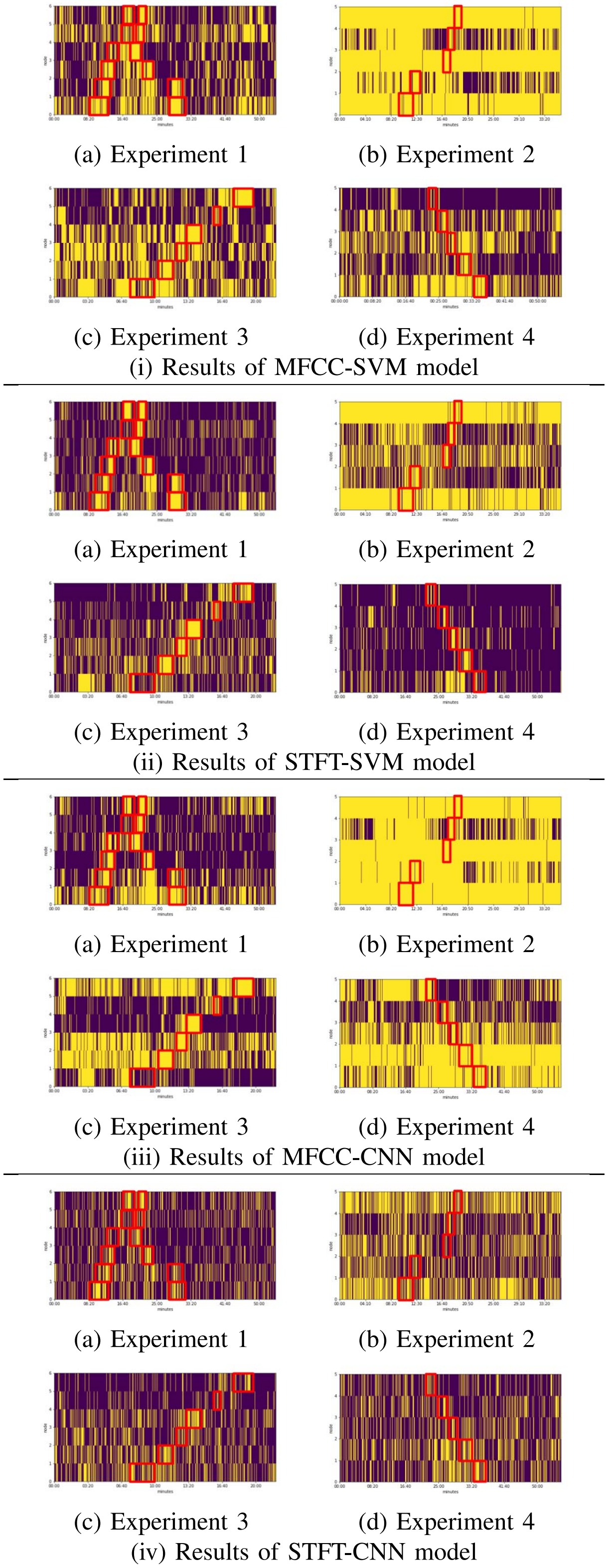
Figure 6 (i) shows the results of MFCC-SVM model. While the red box areas of experiment 1 match the path, it can hardly be seen for experiment 2 because there were constant plane sounds for experiment 2. The model is not able to distinguish UAV sounds and plane sounds clearly, leading to dominant false positive results. The paths are not matched from the results of MFCC-SVM for experiment 3 and 4. Compared to MFCC-SVM, STFT-SVM model from figure 6 (ii) returns better results with clearer paths.

Although MFCC is widely used for audio classification tasks, it can be bad for UAV detection because both wind and UAV have stronger amplitude on lower frequency bands. MFCC contains more dense information of sounds as it represents sounds with several coefficients, while STFT is relatively an intermediate feature. For that reason, UAV can be better distinguished when STFT is used.

MFCC-CNN in figure 6 (iii) shows a clear path with matching bright areas particularly for experiment 1. In experiment 2 and 4, meaningful patterns are difficult to find. STFT-CNN in figure 6 is less obvious compared to MFCC-CNN. The patterns are noticeable but they are very light with a lot of false negatives.

Whereas the SVM model shows noticeable patterns with STFT, the results of CNN are ambiguous. CNN has more fluctuations so it is hard to find a solid dark or bright area and the patterns are more random. Considering that the task is only binary classification, and that a lot of resources are needed for training CNN, SVM is more appropriate for this task, and the parameters can be easily found through a grid search.

The best configuration uses STFT-SVM for the model. The results of STFT-SVM are shown in figure 7 with synchronized to allow comparison.Considering the distance between bright areas, experiment 3 from figure 7 (b) shows the best results. Experiment 1 has a higher portion of overlapped areas



(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

(i) Results of MFCC-SVM model

(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

(ii) Results of STFT-SVM model

(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

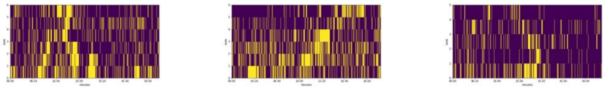
(iii) Results of MFCC-CNN model

(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

(iv) Results of STFT-CNN model

Fig. 6: Results of the four models



(a) Experiment 1 (b) Experiment 3 (c) Experiment 4

Fig. 7: Results of experiment 1, 3 and 4 by STFT-SVM model

compared to experiment 3 and 4. Experiment 4 shows that bright areas are far apart while bright areas in experiment 3 are closely located. It is obvious that the configuration of experiment 3 has higher chance of detecting UAVs without blind areas when a UAV flies between nodes.

Using these color maps, not only the path but also the speed of the UAV can be estimated from the length of the bright area and the distance between nodes. It can give additional insight on analyzing threats by providing data on how long the UAV stayed around a node.

## V. Conclusion

### A. Limitation

There are limitations regarding modeling. First, the signals had a lot of noise and were not normalized. Normalization of the audio signals is needed and noise should also be reduced. As shown in the previous chapter, the detection results were different although the same sounds were recorded because microphones and sound cards have different qualities. There are known ways to normalize signals including scaling, which is a technique to divide the signals by the maximum value of the total audio file. However, this cannot be applied for the real-time environment because the maximum value keeps changing as new signals are fed.

Second, the quality of the test data is not consistent with the training data because of equipment issues. Below are the factors that can affect the results:

- The UAV was tethered to a string. When the UAV became out of range because of wind, the string was pulled and the resistance could have affected the audio signals.
- After several experiments, the UAV was not able to fly. The pilot held the UAV and walked along the nodes instead of flying it.

Third, the models are not strong enough leading that they cannot differentiate the UAV from noise when the noise is dominant.

### B. Future Work

First, research can be done on normalization or noise reduction methods for real-time signals. Second, multi-class model can be trained with more audio data rather than binary classification. Separate classes for wind, airplanes and birds can be defined for clearer distinctions. Different types of machine learning models can also be used to detect multiple classes at the same time.

Additionally, machine learning and rule-based methods can be combined to find which node is closer to the UAV. It is hard to find where the UAV is when a louder UAV approaches and multiple nodes detect it. Utilizing the fact that each sound has its own frequency range, the system can monitor if the UAV is approaching closer by calculating the magnitude of specific frequency range.

## References

[1] "Drone Industry Analysis: Market trends & growth forecasts - business insider."

[2] "Prison drones drug smuggling gang jailed,"

[3] J. Warrick, "Use of weaponized drones by ISIS spurs terrorism fears,"

[4] J. M. Goppert, A. R. Wagoner, D. K. Schrader, S. Ghose, Y. Kim, S. Park, M. Gomez, E. T. Matson, and M. J. Hopmeier, "Realization of an autonomous, air-to-air counter unmanned aerial system (CUAS)," in *2017 First IEEE International Conference on Robotic Computing (IRC)*, pp. 235–240.

[5] A. R. Wagoner, D. K. Schrader, and E. T. Matson, "Towards a vision-based targeting system for counter unmanned aerial systems (CUAS)," in *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, pp. 237–242.

[6] Skolnik, *Introduction to Radar Systems*. Tata McGraw Hill. Google-Books-ID: zlZom9QkjCkC.

[7] M. Benyamin and G. H. Goldman, "Acoustic detection and tracking of a class i UAS with a small tetrahedral microphone array."

[8] . Gven, O. Ozdemir, Y. Yapici, H. Mehrpouyan, and D. Matolak, "Detection, localization, and tracking of unauthorized UAS and jammers," in *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*, pp. 1–10.

[9] S. Park, Y. Kim, E. T. Matson, and A. H. Smith, "Accessible synthetic aperture radar system for autonomous vehicle sensing," in *2016 IEEE Sensors Applications Symposium (SAS)*, pp. 1–6.

[10] V. Mitra, C.-J. Wang, and S. Banerjee, "Lidar detection of underwater objects using a neuro-SVM-based architecture," vol. 17, no. 3, pp. 717–731.

[11] L. Wallace, A. Lucieer, and C. S. Watson, "Evaluating tree detection and segmentation routines on very high resolution UAV LiDAR data," vol. 52, no. 12, pp. 7619–7628.

[12] N. Jeong, H. H., and M. E. T., "Evaluation of low-cost lidar sensor for application in indoor uav navigation," in *2018 IEEE SAS*, p. ?

[13] A. Rozantsev, "Vision-based detection of aircrafts and uavs," p. 116, 2017.

[14] E. E. Case, A. M. Zelnio, and B. D. Rigling, "Low-cost acoustic array for small UAV detection and tracking," in *2008 IEEE National Aerospace and Electronics Conference*, pp. 110–113.

[15] S. Jeon, J. W. Shin, Y. J. Lee, W. H. Kim, Y. Kwon, and H. Y. Yang, "Empirical study of drone sound detection in real-life environment with deep neural networks," in *2017 25th European Signal Processing Conference (EUSIPCO)*, pp. 1858–1862.

[16] I. Fette and A. Melnikov, "The WebSocket protocol,"

[17] "Parrot AR.drone 2.0 elite edition."

[18] V. Tiwari, "MFCC and its applications in speaker recognition,"

[19] R. Serizel, V. Bisot, S. Essid, and G. Richard, "Acoustic features for environmental sound analysis," in *Computational Analysis of Sound Scenes and Events*, pp. 71–101, Springer, Cham.

[20] L. Deng, G. Hinton, and B. Kingsbury, "New types of deep neural network learning for speech recognition and related applications: an overview," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 8599–8603.

[21] G. Guo and S. Z. Li, "Content-based audio classification and retrieval by support vector machines," vol. 14, no. 1, pp. 209–215.

[22] V. N. Vapnik, "An overview of statistical learning theory," vol. 10, no. 5, pp. 988–999.

[23] H. Lee, Y. Largman, P. Pham, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Proceedings of the 22Nd International Conference on Neural Information Processing Systems*, NIPS'09, pp. 1096–1104, Curran Associates Inc.

[24] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6.

[25] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," vol. 521, no. 7553, pp. 436–444.