

Drone Detection & Classification using Machine Learning

Mr. Shaik Asif Basha¹, Dr. D. Jagadeesan², Mr. V Maruthi Prasad³

¹Student, IV-CSE, Dept. of CSE

Madanapalle Institute of Technology & Science, Madanapalle, AP, India,

²Asst. Professor, Dept. of CSE

Madanapalle Institute of Technology & Science, Madanapalle, AP, India,

³Asst. Professor, Dept. of CST

Madanapalle Institute of Technology & Science, Madanapalle, AP, India

ABSTRACT

Due to the increase in the development of economic and recreational of drones, moreover because the associated risk to airspace safety, this study proposal has emerged in recent years. This analysis plan has emerged within the previous couple of years because of the speedy development of economic and recreational drones and also the associated risk to airspace safety. A comprehensive review of current literature on drone detection and classification victimization machine learning with completely different modalities, during this analysis plan has emerged within the previous couple of years because of the speedy development of economic and recreational drones and also the associated risk to airspace safety. Drone technology has been used in practically every aspect of daily life, including food delivery, traffic control, emergency response, and surveillance. Drone detection and classification are carried out in this study using machine learning and image processing approaches. The research's major focus is on conducting surveillance in high-risk areas and in locations where manned surveillance is impossible. In such situations, an armed aerial vehicle enters the scene and solves the problem. Radar, optical, auditory, and radio-frequency sensor devices are among the technologies addressed. The overall conclusion of this study is that machine learning-based drone categorization appears to be promising, with a number of effective individual contributions. However, the majority of the research is experimental, therefore it's difficult to compare the findings of different articles. For the challenge of drone detection and classification, there is still a lack of a common requirement-driven specification as well as reference datasets to aid in the evaluation of different solutions.

Keywords: Drone Detection, Machine Learning, Airspace safety, Radar, Acoustic, Airspace vehicle, surveillance, Motion Detection, Image Processing

1. INTRODUCTION

Every day we come across many drones captured images and videos, now a days drones are used in many ways such as shooting in occasions, delivering the food using automated drone and many more usages are being done. Then why can't a drone is used for the purpose of security and surveillance. Here the problem has raised. Then,

A good system for detecting Drones must be able to cover a broad area while yet distinguishing themselves from other things. One approach to doing this can be to use a combination of wide and slim field of read cameras. Another strategy is to use a bunch of high-resolution cameras. there's no thanks to have a fisheye infrared detector or Associate in Nursing array of such sensors. in this thesis because there is only one with a fixed field of view. To obtain the required volume coverage, the IR-sensor will be installed on a moving platform. At different periods in time, this platform can be assigned objects or search on its own.

This is AN era of advancement and technology. attributable to this technology and advancement, the crime and attacks of criminals have conjointly inflated. to cut back and shield the voters, correct security and police work area unit required. several organizations try to develop advanced machines and techniques. one among those is that the ariel vehicle technique.

In this technique, the employment of drones is completed for the bar and protection of the voters of a selected place. this method is employed to look at and conduct police work at the border of a selected place.

This project is titled "Drone detection and classification" as a result of during this analysis the usage of the drone is completed for the detection of another drone, and therefore the algorithmic rule during this drone is employed to find and classify the item that is passing from it.

1.1 Objective

The purpose of the paper is to observe, classify the drone from different objects that are moving. Our Machine learning model itself is predicated on the motion of the objects and mapping of the objects with the dataset and provides the ultimate output. The machine learning algorithmic rule is developed for mapping and creating the correct representation of the moving object is finished. the foremost vital objective of the project is securing and police work of the border areas.

1.2 Limitations

Motion detection of the drone and capturing of the movement of the drone is possible and it can be done with the machine learning algorithm, Since the working on drone the main limitation of the project is the climatic condition and the battery backup.

2. LITERATURE SURVEY

Unmanned Air Vehicles (UAVs) (commonly referred to as drones) create variety of considerations for airspace safety which will injury individuals and property. despite gaining widespread attention in a variety of civic and commercial uses. While such threats can range from pilot inexperience to premeditated attacks in terms of the assailants' aims and intelligence, they all have the potential to cause significant disruption. Drone sightings are becoming more common: in the last few months of 2019, for example, many airports in the United States, the United Kingdom, Ireland, and the United Arab Emirates have faced severe operational disruptions as a result of drone sightings [1]. A sensor or sensor system is required to detect flying drones automatically. When it involves the drone detection drawback, as verified within the fusion of information from several sensors, i.e., using many sensors in conjunction to induce a lot of correct findings than noninheritable from single sensors whereas correcting for his or her individual shortcomings, is sensible. [7]

2.1 Disadvantages

The proposed approach as well as the automated drone detection system are described. First, at a high level, then in further depth, both in terms of Hardware components, as well as how they are connected to the main computing resource and the software that is used. [3] Because matlab is the major development environment, all programs are called scripts, and a thread is referred to as a worker in the thesis material. [4] [5] Using a series of high-resolution cameras, as demonstrated in [7], is another option. There is no way to have a wide-angle infrared sensor because this thesis only has one infrared sensor with a set field of view. The IR-sensor, or a series of similar sensors, is used to achieve the requisite volume coverage. will be installed on a moving platform. When the platform's sensors aren't busy identifying and classifying stuff, it can be assigned objects or go on its own hunt. [6]

2.2 Advantages

Frequency modulation at a low cost In contrast to optical detection, continuous wave radars are resistant to fog, cloud, and dust, and are less susceptible to noise than acoustic detection. [6], It operates in low visibility conditions since it doesn't require a LOS. Depending on the microphone arrays used, the cost is low. [5] Low-cost due to the use of cameras and optical sensors, as well as the reuse of current surveillance cameras. RF sensors at a low cost. Long detection range, no LOS required. [7]

3. METHODOLOGY

The system uses the principal electro-optical sensors for detection and classification, a thermal infrared camera and a video camera. The technology makes ADS-B data available in order to keep track of cooperative aircraft¹ in the airspace. When a drone or a helicopter enters the system's area of operation, audio data is used to determine their presence using their different noises. The system's computations are performed on an ordinary laptop. This is also where the user sees the detection results. The following is the system architecture, which includes the main layout of the graphical user interface (GUI):

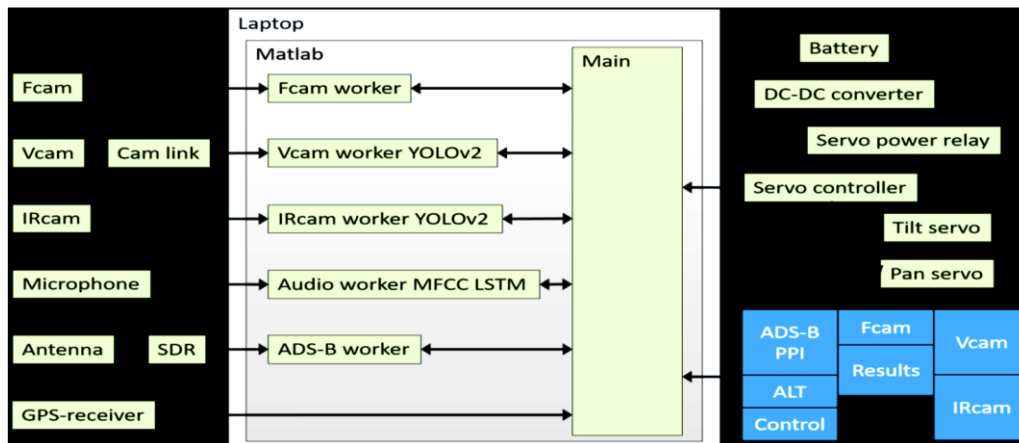


Fig 3.1 System Architecture

The IRcam is positioned beside the Vcam on a pan/tilt platform using data from a fisheye lens camera (Fcam) with a limited field of view that covers 180 horizontally and 90 vertically. If the Fcam detects and monitors nothing, the platform can be programmed to scan the skies in the area. the system in two distinct search patterns. The key components of the system are depicted in Figure 3.2



Fig 3.2 System Setup



Fig 3.3 System Setup

The drone detection system's key components. The microphone is on the lower left, and the fisheye lens camera is above it. The IR and video cameras area unit mounted on the central pan and tilt platform. A stand is found for the servo controller and power relay boards is found behind the pan servo, within the metal mounting channel. apart from the laptop computer, all hardware elements area unit mounted on a typical surveyor's rack. to provide a stable foundation for the system. This strategy also makes it easy to deploy the system outside, as shown in the diagram. The system has to be easily portable to and from any location given the nature of the technology. As a result, a transport solution was designed, in which the system could be disassembled into a few major components and placed in a transport box.

The power consumption of the system components has been considered in the same way that it has been in the any embedded system's design taken into account in order to avoid overloading the laptop's built-in ports and the USB hub. A Ruideng AT34 tester was used to measure the current drawn by all components for this purpose.

A FLIR Breach PTQ-136 thermal infrared camera with a Boson 320x256 pixels detector was employed. The IR-field cameras of vision There are a total of 24 longitudinally and 19 laterally. Figure 3.4 shows a image from the IRcam video stream.

FLIR Breach's Boson sensor, for example, has a better resolution than the FLIR Lepton sensor, which has an 80x60 pixel resolution. Up to a distance of at least meters, the scientists were able to distinguish three different drone kinds. However, detection in that investigation was performed manually by a human watching the live video broadcast, rather than using a trained embedded and intelligent system, as in this thesis.

The IRcam can output two types of files: a clean 320x256 pixel file (Y16 with 16-bit gray scale image) and an interpolated 640x512 pixel image in I420 format (12 bits per pixel). The colour palette of the interpolation image format can be modified, and there are a variety of other image processing choices. The raw format is employed in the system to avoid having extra text information placed on the interpolated image Because the image processing procedures are implemented in MATLAB rather than Python, this option gives you more control over them.

The output of the IRcam is distributed to the portable computer through a USB-C port at sixty frames per second (FPS). The IRcam is additionally hopped-up by the USB association.

A Sony HDR-CX405 video camera was accustomed capture the scenes within the spectrum. as a result of the Vcam's output is AN HDMI signal, AN Elgato Cam Link 4K frame disagreeable person is employed to supply a 1280x720 video stream in YUY2-format (16 bits per pixel) at fifty frames per second to the portable computer. Because the Vcam's zoom lens can be adjusted, the field of view can be made to be broader or narrower than the IRcam' s. The Vcam is a video camera. is set to have about the same field of view as the IRcam.

To monitor a larger area of the system's surroundings, an ELP 8-megapixel 180 fisheye lens camera is used. A USB port produces a 1024x768 Mjpg video stream at 30 frames per second.

A small cardioid directional microphone, the Boya BY-MM1, is also connected to the laptop to identify the sound of the drone.

A RADAR module is not included in the final design, as indicated in the system architecture specification. However, because one was available, it was determined that it should be included and detailed in this section. The RADAR module's performance is described, and its absence is justified by the short practical detection range.



Figure 3.4 IR and Video camera mounting

The system's computing component is run this was done using HP laptop. The GPU is an Nvidia MX150, and the CPU is an Intel i7-9850H. The laptop is connected to sensors and the servo controller via in built ports and an additional USB port.

There are two sections to the program utilized in the thesis. To begin, there is the software that is currently running in the system when it is delivered. A set of support tools is also available for tasks like creating data sets and training the setup.

When the drone detector is turned on, the software consists of a main script and five "workers," which are parallel threads made possible by the MATLAB parallel computing tools. Messages are transported in between the main program and the employees using data queues. Using a background detector and a multi-object Kalman filter tracker, the Fcam worker communicates the azimuth and elevation angles to the main program after establishing the position of the best-tracked target1. The main software can then use the servo controller to drive the pan/tilt platform servos, allowing the IR- and video cameras to analyze the moving object further.

The audio transmitter transmits the information about the classes and confidence to the main part. One is made up of a history track, while the other is made up of current tracks because of this the display depicts and the variations in altitude of the targets clearly.

All of the above parts also confirm the main script's command to run the detector/classifier or be idle. the number of frames per second presently being handled is additionally passed to the most script. any analysis into the various output categories or labels that the most code will receive from the employees finds that not all sensors will output all of the target categories employed in this analysis. Then, the audio employee contains a background category. If the received message's vehicle class field is missing, ADS-B can provide the output "No knowledge Class".

Table 3.1: Observations of different cameras and modules

IRcam	AIRPLANE	BIRD	DRONE	HELICOPTER	
Vcam	AIRPLANE	BIRD	DRONE	HELICOPTER	
Audio			DRONE	HELICOPTER	BACKGROUND
ADS-B	AIRPLANE		DRONE	HELICOPTER	NO DATA

4. WORKING ENVIRONMENT

The main script is the software's brain; it not only starts the five workers (threads) and establishes communication queues for them, but it also delivers commands to the servo controller and gets data from the GPS receiver. Following the start-up procedure, the script enters a loop that runs until the user closes the application through the user interface (GUI).

The most common actions performed on each iteration of the loop refreshing the user interface and reading user inputs. The main script also connects with the servo controller and the workers. at regular intervals. Ten times per second, servo positions and queues are polled. The system results, such as the This component also calculates the corresponding output labels and confidence using the most current worker outcomes. Furthermore, new commands are supplied to the servo controller at a rate of 5 Hz for execution. The ADS-B graphic is updated every two seconds. Having varied intervals for distinct jobs improves the efficiency of the script because, for example, because an aero plane transmits its position every second via ADS-B, updating the ADS-B charts too frequently would be a waste of processing resources.

The IRcam worker attaches to a queue it gets from of the parent script in the function definition when it starts up for the first time. The IRcam worker uses the worker's queue to build a queue for the main script, because the main script can only create a queue from the worker. This establishes bidirectional communication, allowing the worker to provide information about the detections as well as receive directives, such as when the detector ould operate and when it should be idle. When the IRcam worker function is started without a queue.

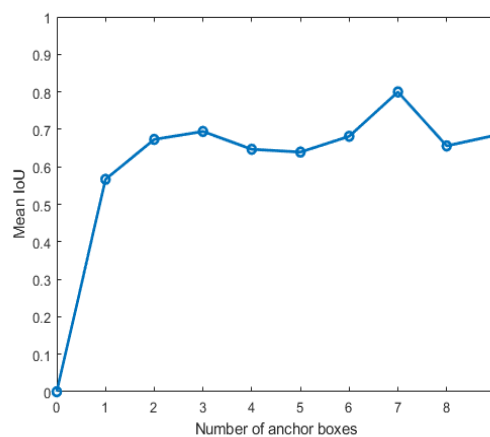


Figure 4.1 Graphical representation of Anchor Boxes

The trade-off to consider when deciding on the number of anchor boxes to Using a big IoU guarantees that the anchor boxes overlap effectively with the training data's bounding boxes. but utilizing more anchor boxes increases the computational cost and increases the risk of overfitting. Three anchor boxes are chosen after reviewing the plot, and their sizes are generated from the output of the estimate Anchor Boxes-function, with a width scaling ratio of 0.8 utilized to match the input layer's shrinkage from 320 to 256 pixels.

After the evaluation data has been selected and set aside, the detector is trained utilizing data from the available dataset. The IRcam YOLOv2 detector's training set contains 120 video clips, each lasting little more than 10 seconds and evenly dispersed over all class and distance bins, totaling 37428 annotated photos. Using the stochastic gradient descent with momentum (SGDM) optimizer, the detector is trained for five epochs¹ with an initial learning rate of 0.001.

The trainYOLOv2ObjectDetector-function does pre-process augmentation automatically because the training data is supplied as a table rather than utilizing the datastore-format. Reflection, scaling, and modifying brightness, color, saturation, and contrast are among the enhancements used.

With a few exceptions, The Vcam worker looks a lot like the IRcam worker. The input image for the Vcam is 1280x720 pixels, and it is shrunk to 640x512 pixels following the obtain snapshot-function. The visible video image is subjected to only one image processing step. The input layer of the YOLOv2 detector measures 416x416x3. When compared to the IRcam worker's detector, the detector's larger size is immediately evident in the detector's FPS performance. The training period is longer than in the IR example due to the higher image size necessary to train the detector. On a system with an Nvidia GeForce RTX2070 8GB GPU, one epoch takes 2 h 25 min. The training set contains 37519 images, and the detector, like the IRcam worker's detector, is trained for five epochs. The fisheye lens camera was originally configured to look upwards, however this caused significant visual distortion in the area just above the initial point, which is the interesting target that appear.

The motion detector is less influenced by image distortion after rotating the camera forward, and since half of the field of view is not used otherwise, this is a feasible approach. The image was initially edited to eliminate the area hidden by the pan/tilt platform. The image below the horizon in front of the system, on the other hand, is no longer evaluated.

The Fcam worker, like the IRcam and Vcam workers, establishes queues and connects to the camera. The camera's input image is 1024x768 pixels, and the lower part of the image is reduced to 1024x384 pixels when the obtain snapshot-function is used.

The image is subsequently analyzed using the computer vision toolbox's Foreground Detector function. This use a Gaussian Mixture Models-based background subtraction technique (GMM). The moving elements are one, and the background is none. hence this function outputs a binary mask. The mask is then processed using the imopen-function, which accomplishes morphological erosion and dilation. The element of imopen function is to set it to 3x3 and to ensure that only extremely small items are removed.

4.1 Fcam Detection

The Fcam worker's output is the FPS-status, as well as the optimum track's elevation and azimuth angles, if one exists at the time. The Fcam has the most tuning parameters out of all the employees. The image processing procedures the multi-object Kalman filter tracker parameters, the front detector and blob analysis must all be chosen and tuned.

The audio worker captures acoustic data using the associated directional microphone and stores it in a one-second buffer (44100 samples) that is refreshed 20 times per second. The mfcc-function from the mfcc-library is used to classify sound in the buffer. the toolbox for audio the parameter Log Energy is set to Ignore based on empirical trails, and the retrieved features are then passed to the classifier.

The LSTM-classifier is made up A fully connected layer, a SoftMax layer, Associate in Nursing a classification layer comprise an input layer, 2 biface LSTM layers with a drop out layer in between, a completely connected layer, a SoftMax layer, and a classification layer. The classifier expands on this, but with the addition of inclusion of a third dropout layer between the bidirectional LSTM layers and an increase in the number of classes from two to three. Because each of the output classes In the audio database, there are 30 ten-second snippets; five clips from each lesson are kept aside for study.

The remaining audio clips are incorporated into the course. For every 10 sec's clip's final second is used for validation, while the rest is used for training. The classifier is being given first trained for 250 epochs, using a near-zero training set error and a marginally increasing validation loss.

4.2 Approximate frequency calculation of different objects

Because, unlike the YOLOv2-networks, the audio module is a classifier rather than a detector, and it is trained using a separate class of general background sounds. These were captured outside in the system's regular deployment environment, and include some clips of the servos moving the pan/tilt platform. The output, like those of the other workers, is a class label and a confidence score. As previously mentioned, not every aircraft will broadcast its vehicle category as part of the ADS-B squitter message. The implementation of ADS-B message decoding can be done in two ways. The first approach is to dump the data using Dump1090-software and then import it into Matlab, with the worker just sorting it according to the main script. Another option is to use the Communications toolbox in Matlab to create ADS-B decoding.

The difference between the two approaches is that Dump1090 requires fewer computational resources, but the ADS-B squitter's vehicle category field is not included in the output JSONformat1 message, even if it is filled.

The Matlab technique, on the other hand, outputs all available information about the aircraft while consuming more computational resources, causing the system to slow down overall performance.

Data was collected to identify the percentage of aircraft that sent out vehicle class information in order to find the best option A script was created as part of the support software to collect basic statistical data, and it was discovered that 328 of the 652 planes, or slightly more than half, sent out vehicle type information. The vehicle category for the rest of these planes was set to "No Data." Given that around half of all aircraft send out their vehicle category, one of the main pillars of this strategy is to detect and track other flying objects that could be mistaken for drones, despite the computing burden.

All of the vehicle categories that are subclasses of the aero plane target label are grouped together. "Light," "Medium," "Heavy," "High Vortex," "Very Heavy," and "High-Performance Highspeed" are all grades. Helicopter is the name of the class "Rotorcraft." There is also a "UAV" category designation, which is interesting. Although the label is translated into drone, this is also implemented in the ADS-B worker.

One might wonder if such aircraft exist, and if so, whether they fall under the UAV vehicle category. Take a look at the Flightradar24 service for an example. One such drone may be seen flying near Gothenburg City Airport, one of the locations where the dataset for this thesis was collected.

4.3 Flight Radar24 software

If the vehicle category message is received, the classification's confidence level is set to 1, and if it isn't, the label is changed to aero plane, which is the most prevalent aircraft type, with a confidence level of 0.75. any of the other sensors could affect the final classification. The support software comprises of a series of scripts for creating training data sets, configuring the YOLOv2 network, and training it. There are other scripts that import a network that has previously been trained and do additional training on it. The following are some of the duties that the support software routines handle:

- ADS-B statistics are gathered.
- Transformations and recordings of audio and video
- The training datasets are made up of the data that is available.
- Estimating the number and size of anchor boxes needed to set up the detectors

CONCLUSION

The rising use of drones, as well as the resulting safety and security concerns, underline Drone detection technologies that are both efficient and dependable are in high demand. This thesis looks at the possibilities for creating and building a multimodal drone detection system employing cutting-edge machine learning techniques and sensor fusion. Individual sensor performance is evaluated using F1-score and average accuracy mean (mAP).

Machine learning algorithms can analyze the data from thermal imaging, making them ideal for drone identification, according to the study. With an F1-score of 0.7601, the infrared detector performs similar to a regular camera sensor, that

has an F1-score of 0.7849. The F1-score of the audio classifier is 0.9323. Aside from that, in comparison to previous studies on the use of an infrared sensor, this thesis increases the amount of target classes used in detectors. The thesis also includes a first-of-its-kind analysis of detection performance. employing a sensor-to-target distance as a function of The Johnson-based Detect, Recognize, and Identify (DRI) criteria were used to create distance bin division.

When compared to any of the sensors working alone, the suggested sensor fusion enhances detection and classification accuracy. It also highlights the value of sensor fusion in terms of reducing false detection.

REFERENCES

1. J. Sanfridsson et al. Drone delivery of an automated external defibrillator – a mixed method simulation study of bystander experience. *Scandinavian Journal of Trauma, Resuscitation and Emergency Medicine*, 27, 2019
2. S. Samaras et al. Deep learning on multi sensor data for counter UAV applications a systematic review. *Sensors*, 19(4837), 2019
3. E. Unlu et al. Deep learning-based strategies for the detection and tracking of drones using several cameras. *IPSI Transactions on Computer Vision and Applications* volume, 11, 2019.
5. H. Liu et al. A drone detection with aircraft classification based on a camera array. *IOP Conference Series: Materials Science and Engineering*, 322(052005), 2018.
6. P. Andradi et al. Night-time detection of UAVs using thermal infrared camera. *Transportation Research Procedia*, 28:183–190, 2017.
7. M. Mostafa et al. Radar and visual odometry integrated system aided navigation for uavs in gnss denied environment. *Sensors*, 18(9), 2018.
8. Schumann et al. Deep cross-domain flying object classification for robust uav detection. *14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 68, 2017.
9. RFbeam Microwave GmbH. Datasheet of K-MD2. https://www.rfbeam.ch/files/products/21/downloads/Datasheet_K-MD2.pdf, November 2019
10. MathWorks. <https://se.mathworks.com/help/vision/examples/create-yolo-v2-object-detection-network.html>, October 2019.
11. MathWorks. <https://se.mathworks.com/help/audio/examples/classify-gender-using-long-short-term-memory-networks.html>, October 2019
12. M. Robb. Github page. <https://github.com/MalcolmRobb/dump1090>, December 2019.
13. Flightradar24. <https://www.flightradar24.com>, April 2020
14. Ever drone AB. <https://www.everdrone.com/>, April 2020.