Journal of Intelligent & Robotic Systems (2024) 110:84
https://doi.org/10.1007/s10846-024-02115-1

REGULAR PAPER

Towards Reliable Identification and Tracking of Drones Within a Swarm

Nisha Kumari[1] · Kevin Lee[1] · Jan Carlo Barca[1] · Chathurika Ranaweera[1]

Received: 19 May 2023 / Accepted: 10 May 2024 / Published online: 5 June 2024

Abstract
Drone swarms consist of multiple drones that can achieve tasks that individual drones can not, such as search and recovery or surveillance over a large area. A swarm's internal structure typically consists of multiple drones operating autonomously. Reliable detection and tracking of swarms and individual drones allow a greater understanding of the behaviour and movement of a swarm. Increased understanding of drone behaviour allows better coordination, collision avoidance, and performance monitoring of individual drones in the swarm. The research presented in this paper proposes a deep learning-based approach for reliable detection and tracking of individual drones within a swarm using stereo-vision cameras in real time. The motivation behind this research is in the need to gain a deeper understanding of swarm dynamics, enabling improved coordination, collision avoidance, and performance monitoring of individual drones within a swarm. The proposed solution provides a precise tracking system and considers the highly dense and dynamic behaviour of drones. The approach is evaluated in both sparse and dense networks in a variety of configurations. The accuracy and efficiency of the proposed solution have been analysed by implementing a series of comparative experiments that demonstrate reasonable accuracy in detecting and tracking drones within a swarm.

Keywords Swarm robotics · Drone tracking · Computer vision · Multi object tracking · Vision based detection · Real-Time surveillance · Machine learning · Object tracking

1 Introduction

Swarms are made up of large or dense groups of nodes with common goals that communicate locally [1]. Swarm robotics (SR) is a special sub-discipline of cooperative robotics in which swarm intelligence methods are used. SR is an alternative method for coordinating systems with several robots that use a large number of autonomous nodes [2]. Drone's interactions with one another and with their surroundings lead to the emergence of collective behaviour [3]. The main idea behind this domain research strategy is to build a large number of tiny low-budget robots that are expected to do the same job as a single complex robot or a small group of complex robots. The aim of multiple drones is to significantly increase group efficiency and make decisions about prospective outcomes collectively.

Future generations of society will be profoundly impacted by SR [4]. Swarms are expected to play an important role [5, 6] in the future, where they will be confronted with a variety of disruptions including adversarial attacks and equipment malfunctions [7]. The proposed research aims to establish the first step towards understanding the interactions between swarms of drones by investigating a novel drone tracking system. Designing tracking systems faces a major hurdle in detecting and tracing drones that possess high dynamic characteristics and distinguishing their behaviour amidst a considerable drone population [8, 9]. The proposed method aims to address these issues by concentrating on the creation of an identification and tracking system to detect the interconnection of the swarm. A high-level overview of the proposed

Kevin Lee, Jan Carlo Barca and Chathurika Ranaweera contributed equally to this work

✉ Nisha Kumari
kumarinis@deakin.edu.au

Kevin Lee
kevin.lee@deakin.edu.au

Jan Carlo Barca
jan.barca@deakin.edu.au

Chathurika Ranaweera
chathu.ranaweera@deakin.edu.au

[1] School of Information Technology, Deakin University, Burwood, Melbourne, Victoria 3125, Australia

method is depicted in Fig. 1, in which a stereo camera is used to capture images of drones in the swarm, and then the images are fed into the detection and tracking algorithm to track drones.

The importance of tracking swarms is best demonstrated by the importance of trying to disrupt them. Once the swarm is dispersed, it would be possible to use the information gathered to identify the network structure and communication links among them. To avoid this, an approach known as confusion can be used. Highly dynamic swarm behaviour which is often referred to as the confusion effect, exhibits certain dynamic features, such as high speed, uniform movement, large numbers, high densities and strong uniformity in appearance [10, 11]. The confusion effect is often considered as a worst-case scenario and is one of the difficult tasks for tracking and is not explored in terms of target tracking. When identifying swarm behaviour, drones that move in a crisscross movement and have a strong uniform appearance are particularly difficult because of density issues and overlapping of drones [12]. This worst-case scenario misleads the detection of swarm nodes, and the proposed research analyses and identifies each swarm individually over time [13] to address this scenario.

The increasing use of drones in various contexts has raised critical concerns. Drones, when operating in swarms, present unique challenges that demand specialized tracking methods. The potential risks associated with unauthorized drone use in public spaces, sensitive regions, and vital infrastructure underscore the need for robust and accurate tracking solutions. The absence of such solutions may compromise privacy, security, and public safety.

The motivation for this research is the need to create sophisticated tracking techniques that can effectively handle the difficulties presented by a swarm of drones. The increasing number of drones, together with their capacity to function as a group, necessitates advanced tracking technologies. We are driven to explore novel solutions to address the essential need to limit the risks connected with unregulated drone activities.

The research presented in this paper proposes a deep learning-based autonomous drone detection, identification and tracking system. Moreover, a lightweight model of the YOLOv6 (You only look once) deep learning method, which has lately gained popularity due to its resilience and accuracy, is used to detect and identify drones using stereo-vision cameras [14]and a Kalman filter is used for tracking the identified drones. This research builds upon the YOLOv6 framework as an object detection framework and adds Non-maximum Suppression (NMS) and a Kalman Filter to enhance its applicability to drone swarm tracking. The following are the most significant findings of the paper:

1. The implementation of an innovative deep learning-based approach, incorporating YOLOv6 for detection and Kalman filtering for tracking, to achieve reliable detection, identification, and real-time tracking of drone swarms through the utilization of stereo-vision cameras.
2. The establishment of a robust framework utilizing the approach to effectively identify drones and establish correspondences between detected drones in real-time scenarios, contributes to improved situational awareness and tracking precision.
3. Comprehensive experimental analyses conducted under diverse conditions, including variations in swarm density, the number of drones, camera distances, and other relevant parameters. The outcomes consistently demonstrated satisfactory results for detecting and tracking drone swarms, highlighting the adaptability and reliability of the proposed algorithm across a spectrum of operational scenarios.

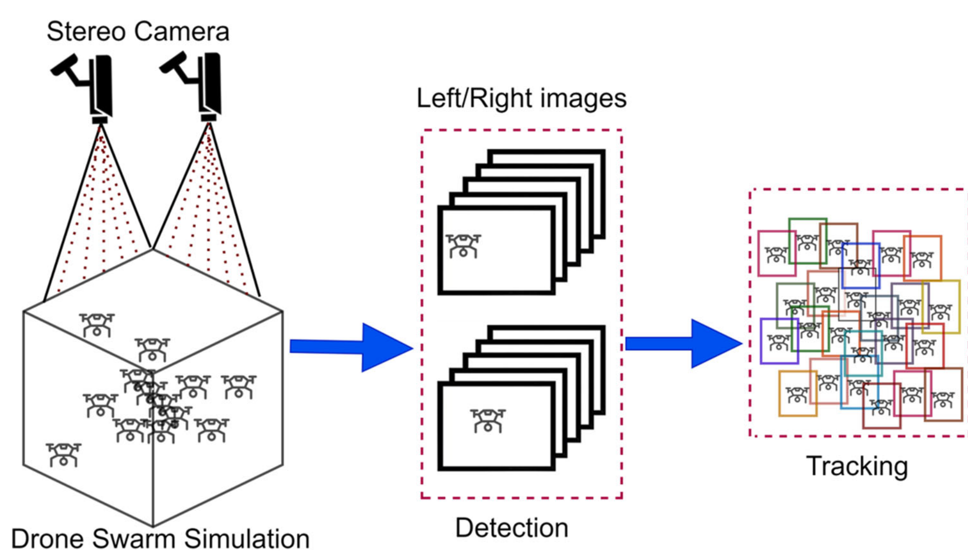


Fig. 1 Drone Swarm Tracking

The remainder of the paper is organised as follows. Section 2 present the discussion on swarms with their respective application and details the existing literature on drone tracking. Section 3 describes the proposed framework for the detection and tracking of drone swarms. Section 4 presents the experimental evaluation of the drones in a simulation environment. Finally, section 5 summarises the proposed research, and section 6 provides the concluding remarks and future work.

2 Background

The term swarm is commonly used to describe a collection of autonomous agents that collaborate to achieve a common goal or desirable behaviour [15–17]. Drones have rapidly grown in popularity in recent years, both on the battlefield and in everyday life. While modernised drones tend to receive the majority of attention, a significant transformation in warfare is currently taking place. This transformation involves the emergence of drone swarms, which consist of intelligent drones that operate as a cohesive unit and respond to the battlefield in a synchronised manner, maintaining a constant speed [13].

Military from all over the world is interested in swarm drone technology because of its special advantages such as reducing the amount of time soldiers must spend in training, increasing the durability of military weapons and enhancing operational effectiveness. Their interest has increased significantly over the past several years. The technology has the potential to completely change the way future wars are handled and can even be applied to search and rescue operations. For instance, the US has been engaged in research to create mini drones that can be launched from aeroplanes and used for reconnaissance missions. According to sources, the US [18, 19] is also investigating drones that can communicate with one another and collaborate to eliminate opponents.

2.1 Drone Swarm

Drone swarms equipped with AI is used in a variety of ways for different applications, including, but not limited to, surveillance, search and rescue [20], intelligent security [21], health care [22], autonomous surveillance of buildings [23], monitoring of forest areas [24], explore unknown zones [25], and flying robots [26]. Swarm technology is rapidly developing and its use is increasing as the market expands at an increasing rate. The drone swarm can automatically recognize and attack the enemies weapons. It's all because of how many drones may be in the air at once and are treated as though there were only one. Among the main benefits of this technology is that they would be able to occupy a wider area.

In drone swarm technology, drones make decisions based on the information they get from each other and have the potential to completely change the way wars are fought. Swarms will be useful in many aspects of regional and border security and they may also be used to look for enemy submarines in the water. To locate and destroy enemy surface-to-air missiles and other air defenses, drones might be dispersed across huge areas. Drone swarms are an exciting new development in technology but realising their maximum capabilities will need research and development in a selected few areas: swarm density, swarm characteristics [27].

Every day, innovative uses and features are developed, and one of the major challenges is keeping up with the swarm of drones. As there are now more drones in the air, there is a greater chance of complications and accidents. Nevertheless, the precise figure varies with each individual job. Many of these characteristics were mentioned in the preceding paragraph. Drones can move around independently, find each other, predict how they will move and stay out of each others way. Assuring the accomplishment of a shared goal by cooperative efforts [28].

The process of detecting and tracking can be accomplished through the employment of singular sensor technology, like radio frequency (RF) detection and spoofing, RADAR, optical devices (thermal and RGB cameras), or audio sensors, as evidenced by [29–31]. Alternatively, detection and tracking can be achieved by integrating multiple sensor technologies. RF signal analyses, which seek to record transmissions between the drone and the ground controller, are the most widely adopted method for drone surveillance [32]. The primary difficulty with this strategy is that the drone might be flown without any sort of ground control, either by following a predetermined flight route or completely autonomously. Drones may also be detected via acoustics, with microphone arrays [33, 34]. Drones try to classify the sounds of their rotors but they fail miserably when it comes to accuracy and range. The longest distance that audio-assisted devices can go is around 200 $m - 250\ m$. The technique also has the drawback of being impractical in busy and noisy places like airports.

In the context of encountering multiple unmanned aerial vehicles (UAVs), conventional countermeasures exhibit a reduction in efficacy. Consequently, scholars have shifted their attention towards developing systems that involve the pursuit of multiple UAVs [35]. High detection accuracy and a huge effective range make video-based recognition a powerful tool for detecting drones [36, 37]. The supervised learning method used in this study is a strong computer vision tool and it is used to do detection on real-time data.

Previous work typically relied on cameras for control and localisation in these hazardous areas when GPS availability could not be guaranteed. This makes it hard to distinguish one drone from another, as it requires linking a drone's unique physical identifier with its unique visual identifier (such as an object tracker output). LEDs [38] or coloured marker [39] have been used in previous attempts to solve this challenge by providing visually distinct information for identification. These techniques however can make deployment more complicated, are fragile to environmental variations, can't scale well with distance and may imply better equipment [40].

2.2 Multi Object Tracking

In computer vision, Multiple Object Tracking (MOT) plays a significant role which is [41] mainly divided by an input video to detect various targets, securing their identification and return their different orientation. The artifacts may be tracked for example, swarms of drones performing target search [42, 43], recognising swarm formations [44, 45] post-disaster [46], swarm of drones in military operations (such as Iraq and Syria) [47], street pedestrians [48, 49], road sports automobiles on the court etc). While various methods have been suggested to resolve this issue because of different factors such as sudden changes in appearance and extreme object occlusions, it still lacking in further experiments. The attacks mentioned above are likely to happen more often as the number of drone attacks increases each year. There are only a few ways to stop drones from engaging in activities that are both harmful and unlawful. There are devices on the market that may be purchased from these companies such as Dedrone, Drone Shield, and Orelia; however, because of the high price of these goods, it is generally agreed that they cannot be used for personal applications on a small scale in residential areas [50].

A study [12] on MOT investigated the use of high numbers, uniform movement and strong uniformity in appearance. They proposed a technique that uses both the reliable monitoring efficiency of recurrent neural networks and the fast testing of the memory formation to estimate the succeeding measure of the swarm. Swarming species have adapted habits that deliberately confuse prospective predators and the process is called the confusion effect however this study could not track and solve the density issue of swarms. Researchers present a method for swarm robots to simultaneously investigate many targets using a grouping approach based on the Particle Swarm Optimisation (PSO) algorithm [51]. A simulation platform is used to show how the search process works and to test the approach. The suggested technique has a high flexibility and success rate while searching for numerous targets (flocking behaviour of birds) according to the results. However, the study did not consider the confusing swarm characteristics in terms of target tracking. Some studies [52, 53] discuss tracking as an individual part and as a result of a program for pattern detection or video monitoring, object tracking is used. A study [54] investigated visual image monitoring strategies, like visual tracking appearance templates. Some research review and implement general visual tracking benchmarks [55] and on particular object tracking [56].

The literature presents how group density affects the confusion effect but the findings are conflicting with some information leaning in one regard [57] and some in the opposite [58]. Despite these challenges, it has been proposed that apparent rises in group densities with predators may assist in enlarging the confusion effect [59]. There are multiple research initiatives to study the MOT issue. However, different studies show research on factors that amplify the effects include high speed, uniform movement, large numbers, high densities, crisscross movement and strong uniformity in appearance [10, 11] but there has not been any comprehensive study that addressed all these factors related to MOT. The proposed approach will track each drone of a swarm over time towards a goal that can help to get reliable identification and track each drone in a swarm in real-time.

The increasing prevalence of drone swarm applications has underscored the urgency for robust tracking methodologies, a facet that has garnered attention in recent literature. Several studies have explored the challenges associated with drone swarm tracking, focusing on aspects such as coordination, collision avoidance, and overall system performance. However, a comprehensive examination reveals a notable gap in the existing research landscape, particularly in addressing the highly dense and dynamic behaviours exhibited by drone swarms.

Recent works [60, 61] provide valuable insights into the complexities of drone swarm dynamics. However, the specific challenges associated with dense and dynamic swarm configurations have yet to be adequately addressed. This is where our proposed approach aims to make a distinct contribution. By utilizing deep learning-based techniques, as demonstrated by this research intends to fill this crucial gap in the literature.

This work aligns with and advances the field by introducing a novel methodology for real-time detection and tracking of individual drones within dense swarms. This approach not only enhances the understanding of swarm behaviour but also facilitates improved coordination, collision avoidance, and performance monitoring. The significance of this contribution becomes apparent when considering the limitations of existing methods, as highlighted by [62, 63].

In conclusion, the literature review establishes a foundation for this study, emphasizing the need for a specialized approach in the context of dense drone swarms. By addressing this gap, this work not only contributes to the existing body of knowledge but also provides a practical solution to the challenges posed by dense and dynamic drone swarm scenarios.

3 A Framework for Swarm Drone Tracking

Among the various challenges, MOT is the biggest challenge in the computer vision domain. Recently several benchmarks have been presented in various MOT methodologies to improve MOT research. Building single to multi-object tracking is more difficult since there are typically many targets in a single frame rather than one [55, 64]. Due to the increased mobility in recent years, drone use has increased significantly [65]. In computer vision, the MOT is among the most essential aspects for analyzing video and has a wide range of uses, including surveillance cameras and robotic systems [66]. The ability to detect drones is a crucial requirement for establishing a seamless environment that accommodates all drone operators. This includes both air traffic control and surface security (to prevent drones from crashing in midair or injuring people or damaging property). In the subsequent sections, a detailed description of a comprehensive drone swarm tracking pipeline is provided, including the process for calculating three-dimensional coordinates.

3.1 Drone Swarm Tracking Pipeline

This research focuses on the challenge of finding and following fixed or moving targets in chaotic situations without knowing their position or the arrangement of impediments. The drone swarm's core objective is to take decisions using the knowledge that they exchange with one another. Due to their connection to a shared communications link, drones would be able to carry out a variety of functions. Various tasks in national security domains, such as those intended to overwhelm adversary sensors with targets.

Algorithm 1 Drones Swarm Tracking Pipeline

```
Input: Stereo Images
Output: 3D Coordinates
Detect the Image ← True
while True do
    CNN Prediction of b_boxes // bounding boxes
    if b_boxes == Multiple then
        Object Disambiguation (NMS) // Non Maxima Suppression
        Compute Filtered b_box
    end if
    if b_boxes ← True then
        Apply Sort Tracker ← 3D Coordinates
    else if Tracking Lost ← True then
        Apply Kalman Filter for Prediction
        Tracking Success Gets ← Missing Coordinates
    else
        Tracking Lost ← False
    end if
end while
```

Drones are used to carry out associated searching and tracking operations. The locations of the targets, which may be dynamic in this scenario, would shift over time and be monitored using the stereo-vision camera as shown in Fig. 2. It may be a reasonably low economical method of fending off an opponent swarm attack. Combining the YOLOv6 detection method with NMS and the Kalman filter tracking method from computer vision and robotics yields a framework for detecting and tracking swarms of drones. These additions enhance the performance of the tracking framework for drone swarms.

1. **YOLOv6 Detection**: YOLOv6 excels in detecting and tracking objects with speed and precision, making it particularly well-suited for our task of reliable detection and tracking of individual drones within a swarm. The architecture of YOLOv6 is characterized by a deep neural network with a focus on dividing the input image into a grid and predicting bounding boxes and class probabilities directly. This approach aligns with our goal of achieving real-time performance and accurate identification of drones in dynamic swarm scenarios. By utilizing the capabilities of YOLOv6, our methodology benefits from the model's ability to handle dense and complex scenes efficiently.
   YOLOv6 can be trained on a dataset of drone images and videos then it can identify drones in real-time video with speed and accuracy. YOLOv6 is a cutting-edge object detection algorithm that quickly and accurately locates various items within a single picture or video frame. It can detect drone swarms by recognising individual drones as distinct objects in the captured video or image. For object detection in real-time applications, the YOLOv6 deep learning technique is widely employed. However, sometimes it generates redundant tracking data due to its tendency to yield multiple bounding boxes for a single object. To solve this issue a technique called non-maxima suppression is used.
   This study uses the newest YOLOv6 algorithm, which is specifically designed to identify different types of drones. There are four main steps in the recognition process. First, the input data is carefully prepared to make sure it works with the suggested network. After that, the model goes through a training process to learn how to recognize the given types of drones. This creates a weight file that will be used for testing later. The trained model is then put through tests to see how well it works and how well it can recognize different kinds of drones. Lastly, the suggested deep learning network is carefully tested using standard methods, which gives a numerical value to its precision and usefulness. These steps, taken in order, show the whole process we used in our study. This study focused on developing, training, testing, and evaluating the YOLO-based network for versatile object detection.

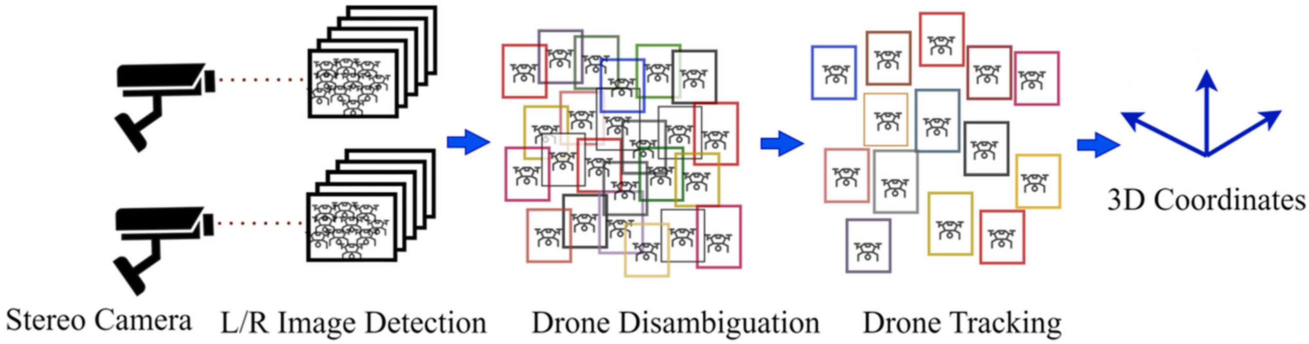


Fig. 2 Concept Diagram for Reliable Swarm Detection

2. **Non-maxima Suppression**: Non-Maxima Suppression (NMS) is a method that can be utilised to deal with this problem. In object detection, NMS is used to eliminate duplicate detections and keep only the best detection. NMS's fundamental principle is to compare all detections and find the ones that are most likely to be identical. In this research NMS is employed to refine the detection results, eliminating redundant bounding boxes and enhancing the accuracy of localization. This integration is particularly beneficial in swarm scenarios where multiple drones are detected in dense conditions.
3. **Kalman Filter**: A popular example of Bayesian filter techniques is known as Kalman filter [67]. It is a method that more accurately guesses unidentified parameters from a set of data that has been collected over time but includes noise and other irregularities. In 1960, it was first suggested by R.E. Kalman [68] and has since evolved into a widely used method for an optimal estimate. In the areas of orbit computation, military surveillance and navigational the Kalman filter has found widespread usage due to its advantages of real-time, quick, efficient and strong anti-interference. Also, it has a significant impact in the areas of digital image processing as well as other study areas like machine learning [69].
   The combination of computer vision methodologies, specifically the utilisation of YOLOv6 detection, in conjunction with Kalman filtering, presents a robust and effective approach for achieving precise and real-time tracking. The integration of the Kalman Filter stands out as a key modification, providing a robust mechanism for tracking individual drones over time. The Kalman Filter, by modelling the dynamic movements of drones, contributes to better trajectory predictions and enhanced tracking precision, particularly in scenarios marked by dynamic swarm behaviour. The Kalman filter is comprised of two fundamental stages: prediction, which involves the propagation of the system state, and update, which entails the incorporation of measurements. The prediction step entails utilising the state transition model to forecast the future state estimate by extrapolating the current state estimate. The update step integrates the YOLOv6 detection outcomes to enhance the accuracy of the state estimate and covariance matrix. We started by determining the object's state vector using the $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ coordinates in the camera coordination frame, which represents the relative positions and velocities from the camera. In addition, the fact that the dynamic state of the target exhibited a consistent velocity, an assumption is made that all states, measurements, and noises followed a Gaussian distribution. As a result, the Kalman filter is used to describe the drone swarm's dynamic system. This one filter keeps predicting the track of all drones [70].
   Q and R are covariance matrices, representing process and measurement noise, and are used to model and reflect the uncertainty in the state transition model and observations. The utilisation of matrices is important in the process of adapting the behaviour of the Kalman filter to align with the specific attributes of both the system and the sensor. To get the best results and make sure the tracking is robust, it's essential to fine-tune the parameters Q and R. This method provides a robust solution for practical situations by making use of the state transition model, observations, and noise matrices.

NMS algorithm is applied to each frame to get rid of any drone detections that happen more than once. Next, tracking drones in real-time and predicting their future location is possible with Kalman filtering. The state of a system can be estimated using noisy observations and the mathematical technique of Kalman filtering. Kalman filtering is useful for estimating the position, velocity and acceleration of a drone using noisy sensor measurements obtained either from the drone itself or from other sensors.

The Kalman filter in conjunction with NMS allows for precise drone location tracking with few false positives. Each drone's position and speed are estimated by the Kalman filter, which then predicts the drone's next location. In general, drone swarm technology can benefit from the integration of YOLOv6 detection, NMS and Kalman filter tracking due to the reduction of false detections and the enrichment of individual drone tracking.

This research indeed considers the complex dynamics of drone swarms, and we acknowledge the need for a more detailed explanation of the specific adaptations. Here are additional insights into how this approach captures swarm behaviour patterns beyond simple density and number changes:

This study used stereo cameras and the triangulation method used for 3D to understand how drones move together. These cameras and math help the model to see patterns in how drones arrange themselves and how they move over time, e.g., figuring out formations like flocking and interactions. To handle variations in drone sizes within the swarm, dynamic object scaling is employed, ensuring the model effectively detects and tracks drones of different sizes. Swarm behaviours such as flocking patterns and inter-drone distances are integrated, providing the model with a basis for understanding and predicting behaviour beyond density changes. The model can dynamically adjust the parameters based on observed swarm behaviour, contributing to improved accuracy by allowing the model to adapt to evolving dynamics and making it better at predicting their movements. Feature extraction is applied for anomaly detection, identifying changes from expected patterns in swarm behaviour to enhance the model's ability to capture swarm behaviour beyond simple density and number changes.

3.2 Estimating 3D Coordinates of Drones from Stereo Images:

Depth ($\mathcal{Z}$) perception arises from the Disparity ($\mathcal{D}$) of a given three-dimensional point in the left and right retinal images. Information from two images is used to find the depth of any point in the given image. When the depth or distance is found to every point, depth maps can be created with colours representing the distance away from the camera. Depth maps and stereo-vision cameras can be used in many real-life applications to estimate the distance to different objects. In this scenario, the depth map is used to know the depth of every drone in the frames.

$$\mathcal{X} = \frac{(\mathcal{X}_L - \mathcal{C}_X) * \mathcal{B}}{\mathcal{D}} \tag{1}$$

$$\mathcal{Y} = \frac{(\mathcal{Y}_L - \mathcal{C}_Y) * \mathcal{B}}{\mathcal{D}} \tag{2}$$

$$\mathcal{Z} = \frac{\mathcal{F}_L * \mathcal{B}}{\mathcal{D}} \tag{3}$$

$$\mathcal{D} = abs(\mathcal{X}_{\mathcal{L}} - \mathcal{X}_{\mathcal{R}})$$

Disparity ($\mathcal{D}$): The term disparity is used to describe the visual shift that occurs between two cameras due to the perspective projection of the same 3D point.

When using a stereo camera for positioning 3D coordinates, the following parameters are required:

- **Focal length ($\mathcal{F}_{\mathcal{L}}$) :** Camera's focal length in (*pixel*)
- **Baseline ($\mathcal{B}$) :** Distance between the two cameras in (*meter*)
- **Image size (resolution) :** height * width in (*pixel*)
- The point we locate is $(\mathcal{X}_{\mathcal{L}}, \mathcal{Y}_{\mathcal{L}})$, $(\mathcal{X}_{\mathcal{R}}, \mathcal{Y}_{\mathcal{R}})$ (unit: *pixel*)

The coordinates of the ideal camera midpoint are $\mathcal{C}_{\mathcal{X}}$ = width/2, $\mathcal{C}_{\mathcal{Y}}$ = height/2, respectively. From the above parameters, we can get the following calculation $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ position. Binocular disparity is a concept in computer vision that pertains to the variation in coordinates of corresponding features between two stereo images. Formulas used to get $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ coordinates are defined in the Eqs. 1, 2, and 3 respectively.

Within the domain of computer vision, the term disparity plays an essential role in the process of finding the relative location of objects that are perceived by a stereo camera system. The absolute difference between the horizontal pixel coordinates of the corresponding points in the left and right pictures ($\mathcal{X}_{\mathcal{L}}$ and $\mathcal{X}_{\mathcal{R}}$) is referred to as disparity, and it is used to quantify the pixel level separation across views. By applying this concept, the calculation of the three-dimensional (3D) cartesian coordinates ($\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$) of an object point can be achieved by taking into account the baseline B, the focal length FL of the camera, and the distances $\mathcal{C}_{\mathcal{X}}$ and $\mathcal{C}_{\mathcal{Y}}$ from the optical centres to the point. The calculation of the $\mathcal{X}$-coordinate involves multiplying the normalised disparity by the baseline and then dividing the result by the depth D. Likewise, the $\mathcal{Y}$-coordinate is derived from an analogous equation. The estimation of the $\mathcal{Z}$-coordinate, which represents the depth, is obtained by taking the reciprocal of the disparity normalised by the focal length. The calculations represent the fundamental principles of the disparity-based approach for estimating the three-dimensional spatial characteristics of a point ($\mathcal{X}_{\mathcal{L}}$, $\mathcal{Y}_{\mathcal{L}}$) or ($\mathcal{X}_{\mathcal{R}}$, $\mathcal{Y}_{\mathcal{R}}$) in relation to the intrinsic properties of the camera system, including the focal length, baseline, and image resolution. This formulation enhances the comprehension of the geometric principles involved in transforming pixel-level disparities into real-world 3D coordinates within the domain of computer vision research [71].

In this study, the main approach taken to track the drone swarm is that the information is taken from two images using stereo cameras as can be seen in flowchart Fig. 3 to find the depth of any point in the image. When the depth or distance to every point is found depth maps can be created the depth map is used to know the depth of each drone in the frames.

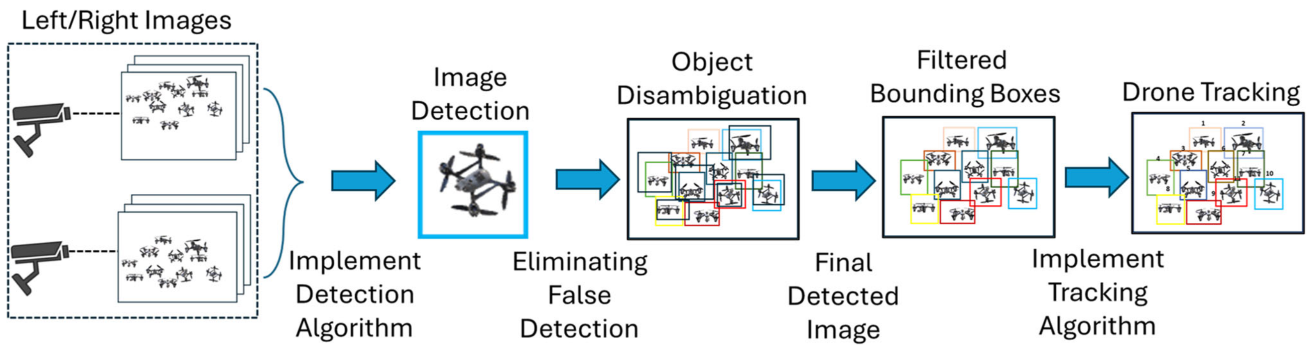


Fig. 3 Flowchart for Reliable Swarm Detection

Once the frames are detected using bounding boxes the NMS method is applied to extract or filter the correct bounding boxes from multiple bounding boxes in a single frame. After getting the correct predicted bounding box tracking is done by using the Kalman filter to predict the new locations of each drone and finally, 3D coordinates are calculated.

4 Evaluation

The experiments are designed to provide new information about the drone swarm. The Section 4 consists of implementation details and experimental setup to detect and track the position of each drone over time. In this paper, a computer-based method is presented for training and assessing networks. The configuration of the drone detection system is shown in Table 1.

In this study, experiments are conducted in the 3D Gazebo simulator, while robots use a Robot Operating System (ROS) that serves as the interface. Integrating both of these develops a powerful robot simulator and it is implemented in Python. It is a multi-robot simulator that is both efficient (good performance with many robots up to several thousand) and flexible (the possibility for the user to add new features such as a new robot or new sensors) at the same time. Gazebo is also used for inertia, gravity, illumination and physical engine etc. A lot of other 3D structures like robots can easily be created on a computer with the help of Gazebo. Originally it was designed to evaluate algorithms for robots. The presented study focuses on exploring the potential applications of the research in the context of drone swarm motion.

Table 1 Experiment Platform Configuration

Experiment Platform Configuration		
Drone Detection/	GPU	4GB NVIDIA T500
Tracking	CPU	Intel Core i7-1165G7 Quad Core
Technique	System	64-bit

4.1 Implementation

In this section, the swarm tracking challenge is described in Fig. 4 in the form of a flowchart which is the task of achieving a synchronized control scheme for a group of drones that allows them to accomplish and manage a certain desired behaviour. The focus is on the problem of sustaining a specified geometrical formation. To ascertain the aforementioned swarm position tracking CVBridge is used which is a ROS library that allows linking ROS and OpenCV in the same environment. The OpenCV library, which is open-source image processing software, is utilized to do separate processing on each and every frame that is captured by the stereo camera. This library is used to convert ROS images into OpenCV images using stereo-vision cameras. Stereo vision creates the original three-dimensional sequences by combining two stereo images having different angles.

Data holds a pivotal role in training models, and an inadequate dataset may lead to underfitting. To address this, we gathered 8000 images from simulations, capturing drone images at various angles, initially unlabeled. Utilizing the LabelImg tool, we labelled these images in YOLO format, denoted as object class-ID, X_{center}, Y_{center}, width and height, after opening an image with a drone and creating a rectangular box around it. The labelled data was then saved in a text file. This labelling process is instrumental for bounding box assignment and subsequent training. The dataset was subsequently split into training and testing subsets as can be seen in Fig. 5. We used the YOLOv6 model with drone images as input. With YOLO, a CNN network can do detection and classification in one pass, as opposed to the sequential procedures used by previous models. Convolutional layers play a vital role in feature extraction from images by addressing spatial redundancy through weight sharing. Through this process of redundancy reduction, the network gains a compressed yet rich representation of the image content. As the depth of the convolutional layers increases, the network becomes more adept at extracting accurate semantic features, contributing to heightened precision in feature representation. In the classi-

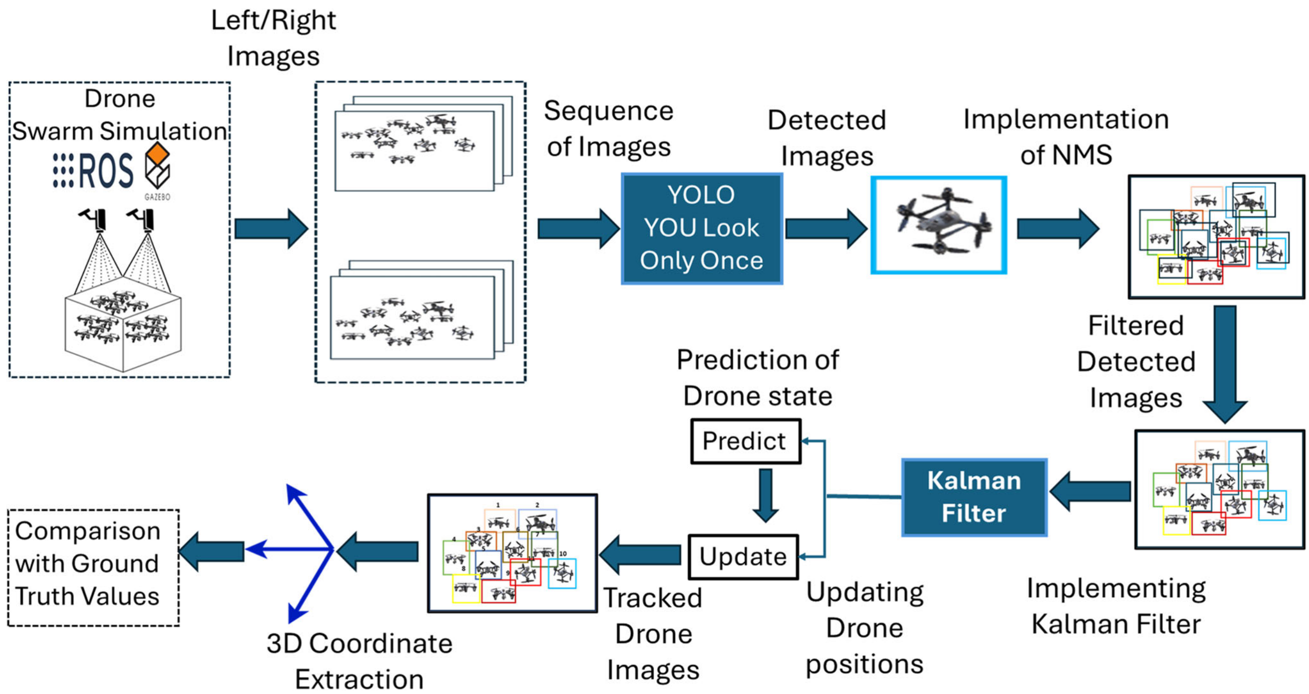


Fig. 4 Experimental Setup

fication and regression step the classification in YOLO refers to the task of assigning a specific class label to each detected object within an image and regression in YOLO involves predicting the coordinates of the bounding box that tightly encloses the detected object. The model, functioning on a grid system, divides each drone image into a grid of fixed dimensions. Grid cells are considered potential drone detectors when the drone's centre falls within a cell. Each grid cell predicts bounding boxes and confidence scores, reflecting the presence of a drone in the box and the model's prediction accuracy. Notably, the YOLO algorithm predicts multiple boxes per grid cell during the training phase, with only one bounding box predictor assigned for an object based on the highest Intersection over Union (IOU) with the ground truth. This meticulous process ensures effective training and accurate predictions for drone detection. Figure 6 depicts the decrease in loss with respect to raining epochs.

To accomplish the desired goal the camera is placed 3 *m* away from the centre of the cube. Using the COCO pre-trained system as initialisation parameters and the YOLOv6 framework as the basis for their identification and categorization of deep learning techniques. Input images are 800 x 800 pixels in size, and the training cycle consists of 80 epochs. The model is developed on an NVIDIA T500 GPU having 4GB of RAM and a batch size of 48, where the deep learning techniques are conducted. The algorithm is stochastic gradient descent (SGD), and its parameters are as follows: weight decay 5 x 10^{-4}, momentum = 0.9. Using a 1 epoch

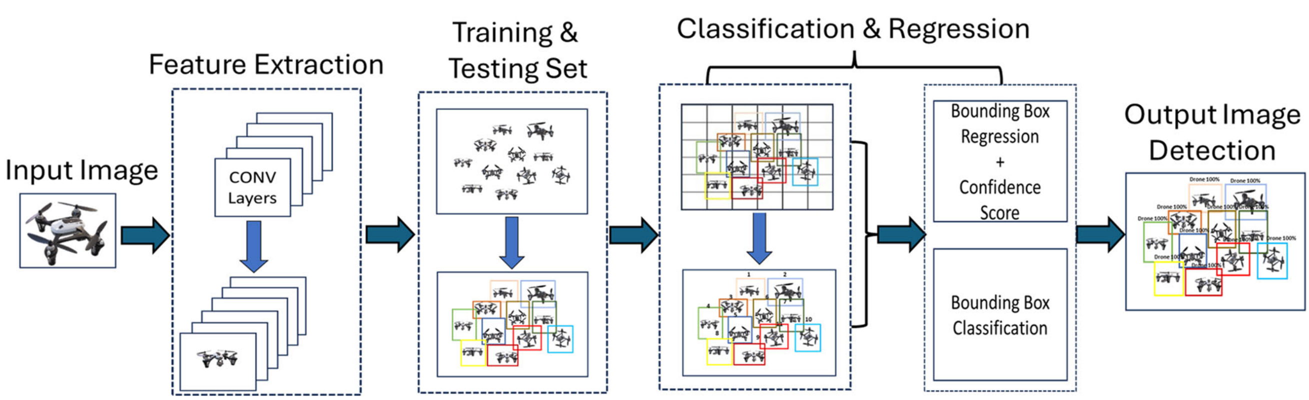


Fig. 5 Training Approach Using YOLOv6

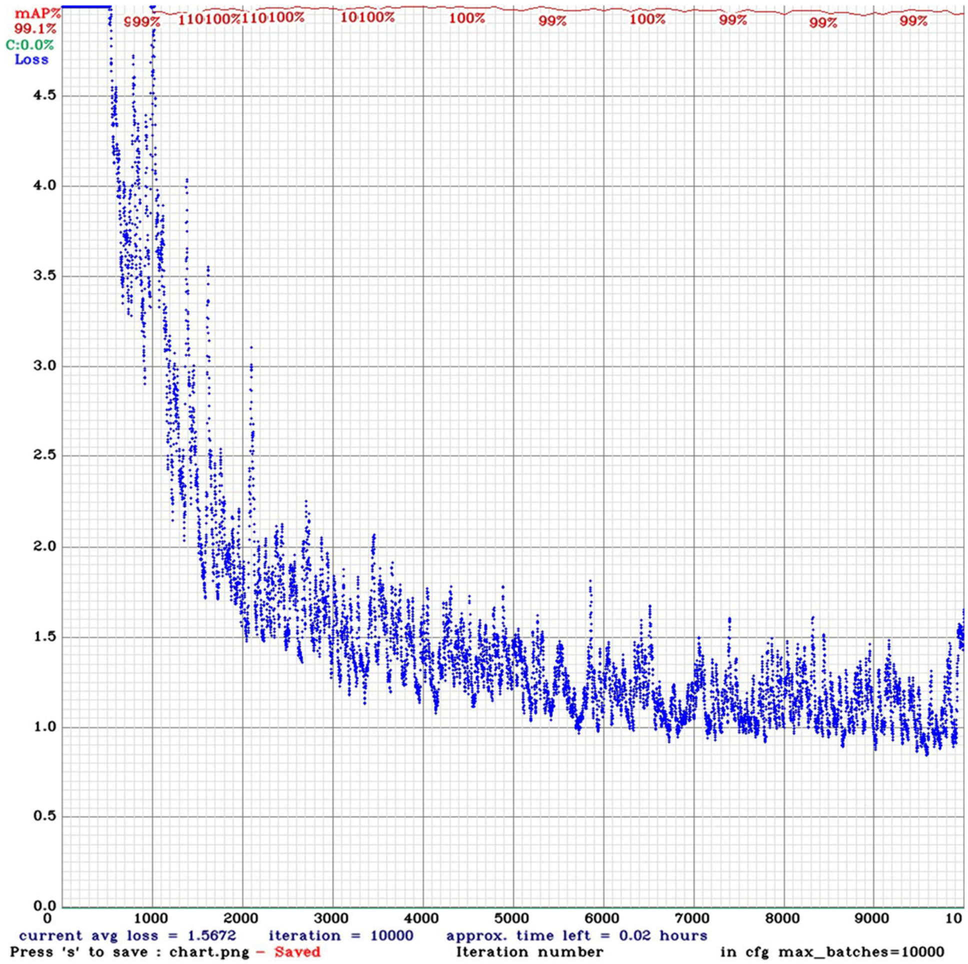


Fig. 6 Screenshot of Training for YOLO

warm-up and a cosine processing schedule, the starting learning rate is 10^{-3}. The entire duration of training is around 2 to 3 days. The system employs RGB stereo cameras positioned on top of the swarm to monitor and record the drone swarm's behaviour. A Linux desktop or embedded device with an NVIDIA GPU is employed as the major processing unit. Both RGB cameras have the same design, specifications, and performance levels as high-end industrial models. The cameras can provide a resolution of 800 x 800 pixels at a frame rate of about 25 frames per second. We are interested in identifying drones from far away, even if they are very small. The method uses the depth image's 3D information to do this. The detection technique is broken down into its component parts and depicted in Figs. 7 and 8. A detailed explanation is given in experiment 1 and experiment 2. To attain tracking accuracy for hidden drones the Kalman filter is used to predict their location and track overlapped and hidden drones over time.

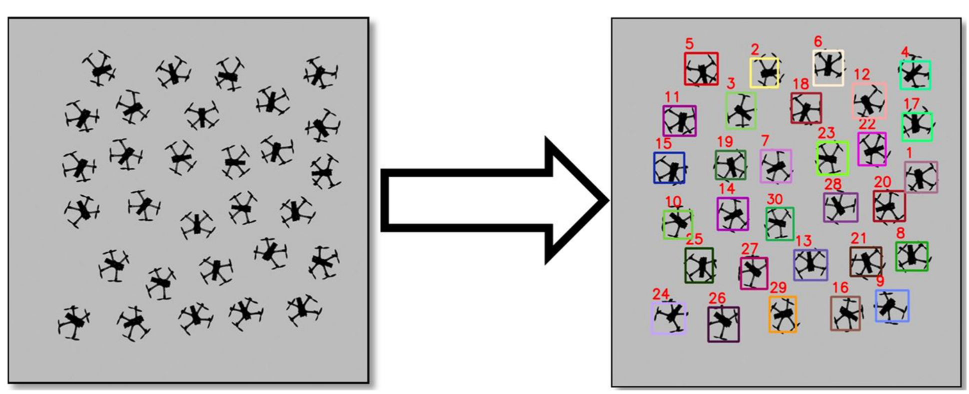


Fig. 7 Before and After Detection

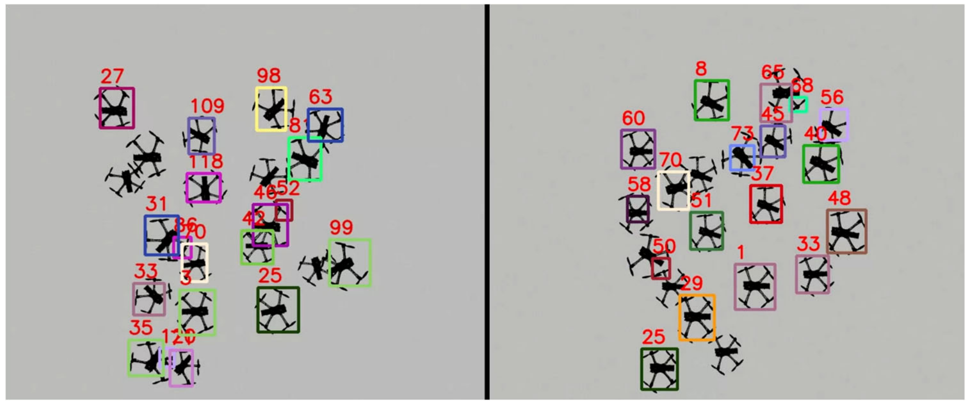


Fig. 8 Before and After NMS

4.2 Experiment Setup

In this research, we are focused on modelling a drone swarm operating within a Gazebo environment. The primary goal of this simulation is to analyze the performance and behaviour of the drone swarm under various conditions. To achieve this, we have carefully selected specific parameters, including drone size and simulation size, and have defined a camera distance range.

The drone size of 30 cm closely resembles dimensions seen in commercial drones used in various applications, allowing us to model realistic movements without overwhelming computational resources. By opting for this size, we can effectively model the movements and interactions of drones within our simulation without introducing unnecessary complexity or computational overhead. It allows us to maintain a reasonable level of realism while ensuring that the simulation remains manageable and resource-efficient.

The obtained 3D positions (X, Y, Z) derived from tracking accurately depict the location of the drone within a three-dimensional domain. The dimensions of the simulation environment are 200 cm along each axis which is represented as 200x200x200 cm. The choice of a 200cm simulation size in each axis is deliberate and serves the purpose of providing ample space for our drone swarm to operate. This size ensures that the virtual environment can comfortably accommodate multiple drones and any potential obstacles or structures that may be encountered during simulation. While it is important to reflect real-world conditions, maintaining a compact enough environment is essential for efficient simulation. This simulation size strikes a suitable balance between realism and computational feasibility, facilitating meaningful experimentation with swarm behaviours.

The distance between the two cameras which is the baseline is 1.3962 mm. Unless otherwise provided, the high detection score threshold τ_{high} is set to 0.6, the low detection score threshold τ_{low} is set to 0.1, and the trajectory initialisation score ϵ is set to 0.7. At the linear assignment phase, the matching will be discarded if the intersection over the union between the detection box and the tracklet box is less than 0.2. The data is stored for the missing tracks for 25 frames in case they reappear during that time.

The specified camera distance range of 2 meters to 9 meters is a key parameter that allows us to explore a wide range of spatial configurations and swarm behaviours within our simulation. This range encompasses distances that are both close and farther apart, mimicking scenarios where drones might operate in various real-world applications. By testing our drone swarm under these conditions, we gain insights into their adaptability and performance in different proximity scenarios, which has direct implications for tasks like surveillance, inspection, and coordination. Another reason for choosing this distance is that moving the camera farther away in the simulation does not capture drones or drones that are not visible for more than this distance in simulation however in the real world using high resolution camera can be implemented with more distance. It is worth noting that in our simulation, capturing drones or making them visible beyond this distance is restricted, reflecting real-world limitations, although in practical applications using high-resolution cameras, longer distances could be achieved.

Image resolution is a critical aspect of our simulation, especially in relation to camera distance. As the camera moves further away from the drones, maintaining image quality and object recognition becomes increasingly challenging. To address this, we have carefully selected an image resolution that aligns with the specified camera distance range. This means that we employ higher resolutions when the camera is positioned farther from the drones to ensure that details are adequately preserved in the captured images. This parameter choice is essential for accurately simulating the visual feedback and challenges encountered by drones at varying distances from their target objects or points of interest.

The experiments conducted different ways drones could be close together and arranged, mainly focusing on tracking individual drones within a swarm. This made us think about whether the network design used has special settings for this. The network architecture is made carefully to handle the specific challenges that come with tracking individual drones within a swarm. Some important things the system does include: it pays more attention to important parts of the information, helping it find and follow individual drones accurately, especially in dense swarm scenarios. It adjusts its learning speed depending on how challenging the information is, ensuring effective learning in various situations when tracking drones within a swarm. It uses a specific way of looking at the information to understand both how individual drones are placed and how they move, making the tracking more accurate. Also, the system can change how much it can do based on how many drones are there, making sure it works well in different situations of tracking individual drones within a swarm. These mechanism-driven optimizations play a vital role in the network's design, contributing to its robustness and effectiveness in addressing the challenges specific to drone swarm tracking, underscoring its suitability for real-world applications.

4.3 Experiment 1: Drone Detection

In experiment 1, the approach for detecting drone swarms is shown and identifying individual drones can be used to identify drones in real-time by finding matching pairs. In the first phase, the approach used is called contour detection and depth triangulation using a stereo camera, the experiment is conducted multiple times and a number of drones (2) are simulated with different positions and angles. Initially, a single frame of 2 drones was detected. Once the results were satisfying then increased the number of drones (30) with random positions as shown in Fig. 7. The procedure is repeated for 30 drones and tracked the single frame of all the 30 drones and took the Euclidean distance between actual and tracked coordinates. Overall by calculating the average distance, this experiment got high accuracy for each run. The overall average distance is 0.068 between actual and tracked coordinates which means data is consistent and very close to the mean.

4.4 Experiment 2: Drone Disambiguation

Experiment 2 demonstrates the details of tracking the correct bounding boxes of our drone swarm detection framework for drone identification. The aim of this experiment is to detect multiple drones with the correct bounding box. A computer vision technique called NMS is used. To enhance the performance and enable the detection of drones from various angles, the algorithms utilise predictive techniques to generate multiple bounding boxes of varying sizes and aspect ratios. NMS constitutes the final stage of object detection algorithms, serving the purpose of identifying the bounding box that is most suitable for the drone. Using this method, the less likely bounding boxes are suppressed in favour of the most optimal ones. The method for choosing the best bounding box with NMS is as follows. Figure 8 shows the output of using NMS before and after steps.

- Step 1: The selection of the box that has the highest objectiveness score.
- Step 2: Compare how this box overlaps with other boxes (intersection over union).
- Step 3: bounding boxes that have an intersection over union greater than $>50\%$ are eliminated.
- Step 4: Then, choose the next best score for the objectiveness score.
- Step 5: Finally, repeat steps 2-4.

4.5 Experiment 3: Framework Validation and Scalability

Experiment 3 demonstrates the details of tracking each drone over time. The aim of this experiment is to track multiple drones over time. For this experiment, a simulation of 30 moving drones and a computer vision algorithm is used for image detection. Despite the best efforts of the system, it may be possible to retrieve some of the missing frames using the tracking-based approach. The semantic structure of the deep learning-based YOLOv6 algorithm allows the system to achieve highly promising performance for a main detection method even with a small number of filters.

4.5.1 Tracking of 10 Drones:

Initially, the experiment was conducted to track the single frame of all 10 drones and took the Euclidean distance between actual and tracked coordinates as seen in Fig. 9. The experiment is conducted numerous times with random positions and by calculating the average distance, this experiment got high accuracy for each run by comparing it with ground truth values. The ground truth values are the random initial actual position and the movement of the drones in each iteration is considered as the *baseline* criteria for comparison. The overall average distance is 0.068 between actual and tracked coordinates additionally the $\sigma = 0.025$ which means data is consistent and very close to the mean.

The experiment was conducted 10 times for 10 drones at random positions (Fig. 10). Figure 11(a) shows the average error between the actual and the tracked location of drones at 10 iterations. The average error is very low at initial iterations indicating high accuracy but it fluctuates due to the random positions of drones used at each iteration. Furthermore, Fig. 11(b) depicts the standard deviation of the distance between the actual and tracked locations and it gives information on how the locations deviated from the mean of the distance between them. Overall, this experiment is conducted

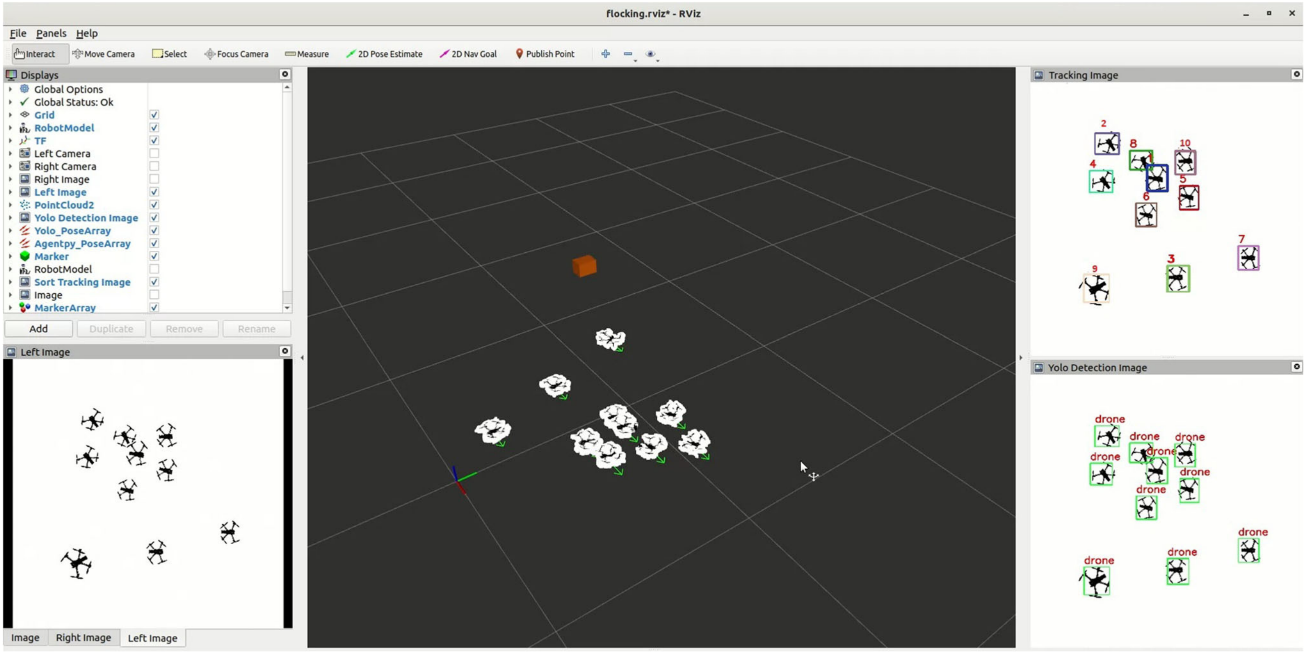


Fig. 9 Tracking of 10 Drones

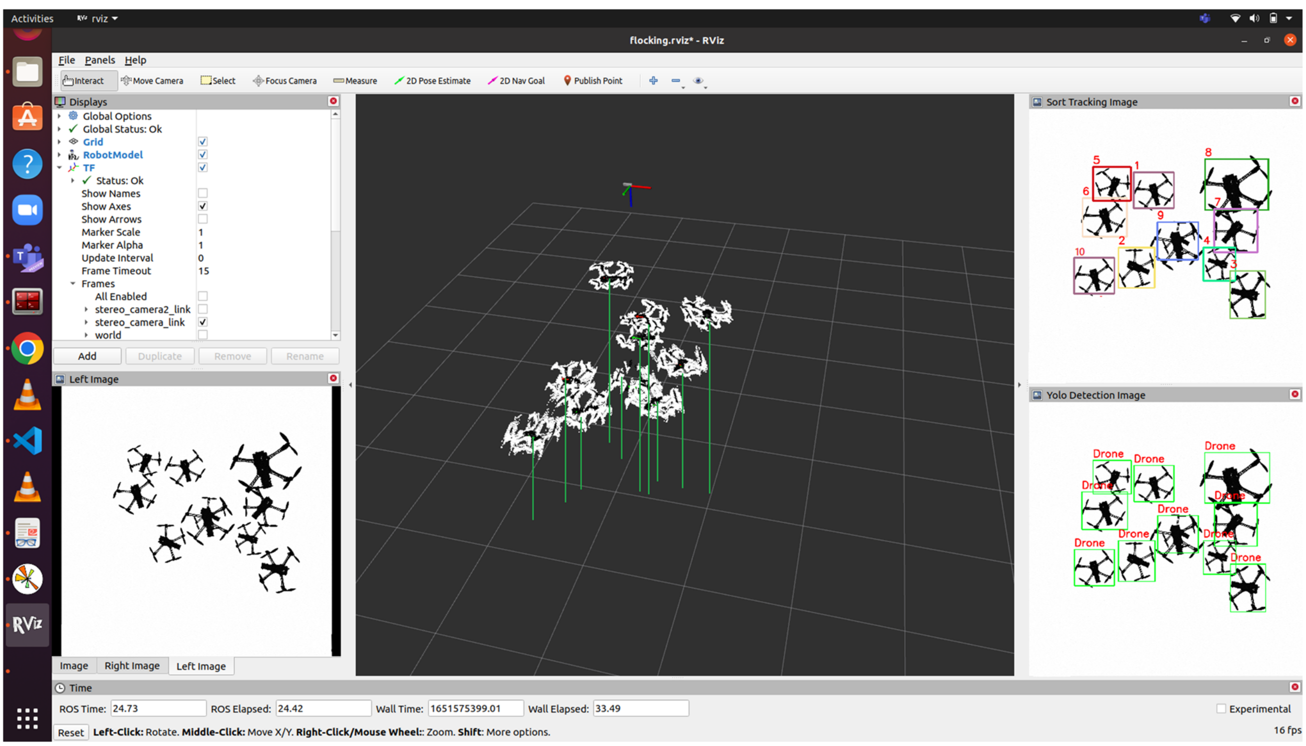
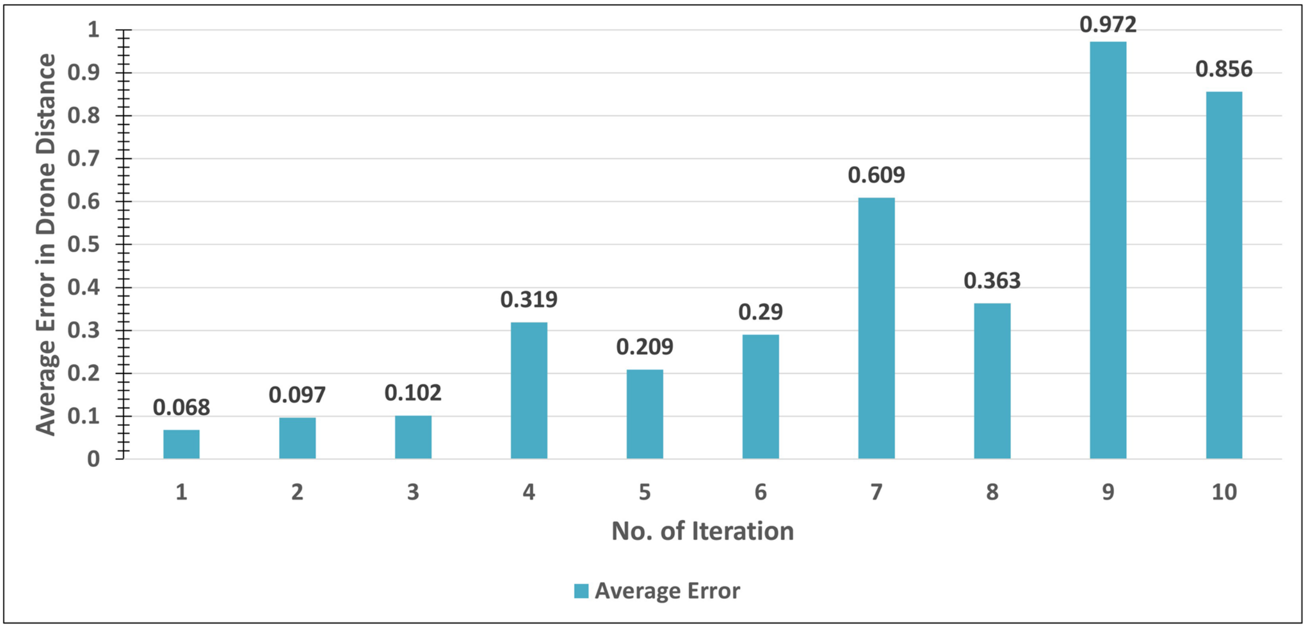
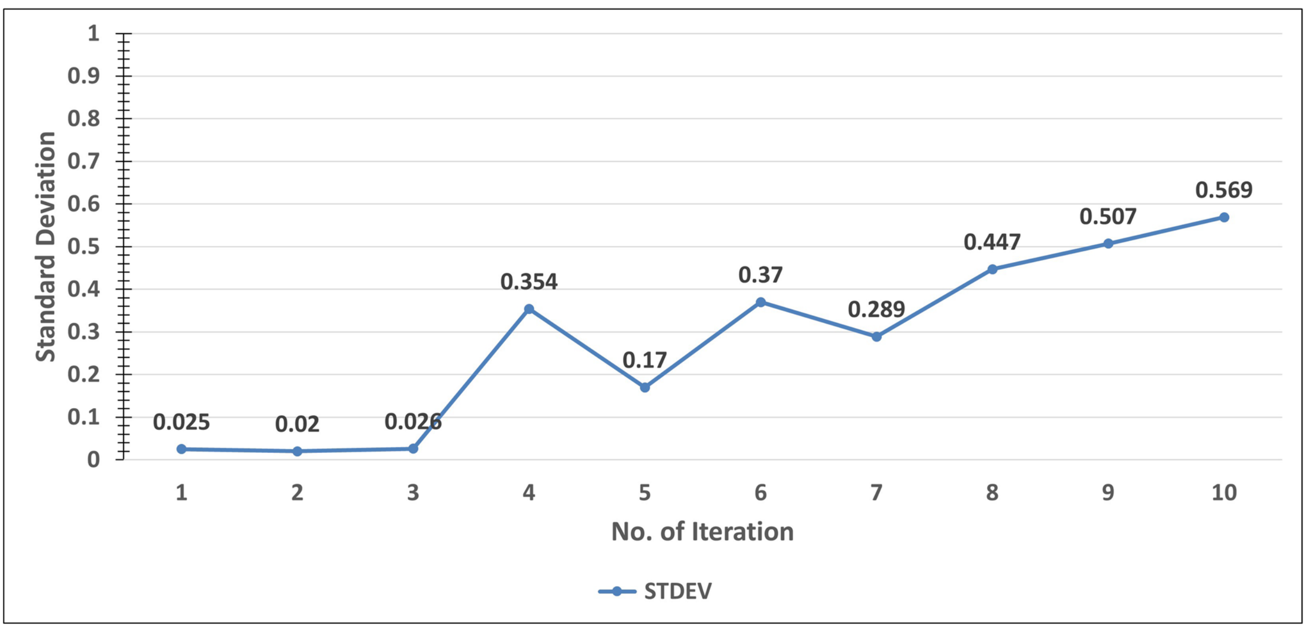


Fig. 10 10 Drones Flight Trajectory



(a) Accuracy Over time



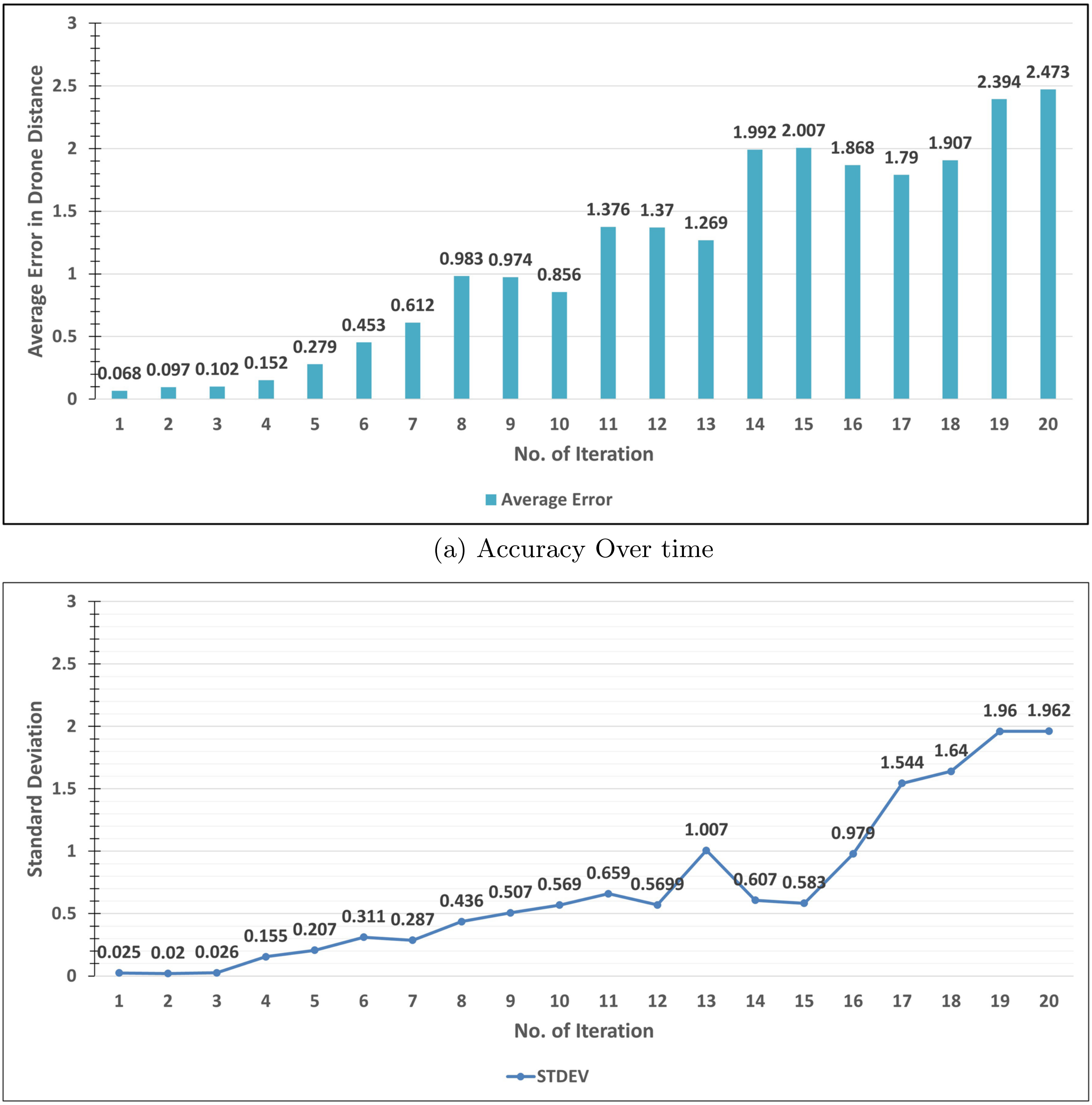
(b) Standard Deviation

Fig. 11 Output of 10 Drones

with random locations each time to analyze the accuracy of the algorithm.

The metrics presented in Figs. 11, 12, and 13 do indeed represent the average error and standard deviation in the estimation of drone distances. These metrics were calculated for all drones for 10 iterations within the respective swarm configurations.

In the initial iteration of our experiments, we introduced 10 drones, all of which were initially visible and evenly spaced from each other. However, it is important to note that the behaviour of these drones in our simulation is inherently dynamic and stochastic. As the simulation progresses, the swarm of drones starts moving, and their positions begin to change. Due to the inherent randomness and dynamic nature



(a) Accuracy Over time

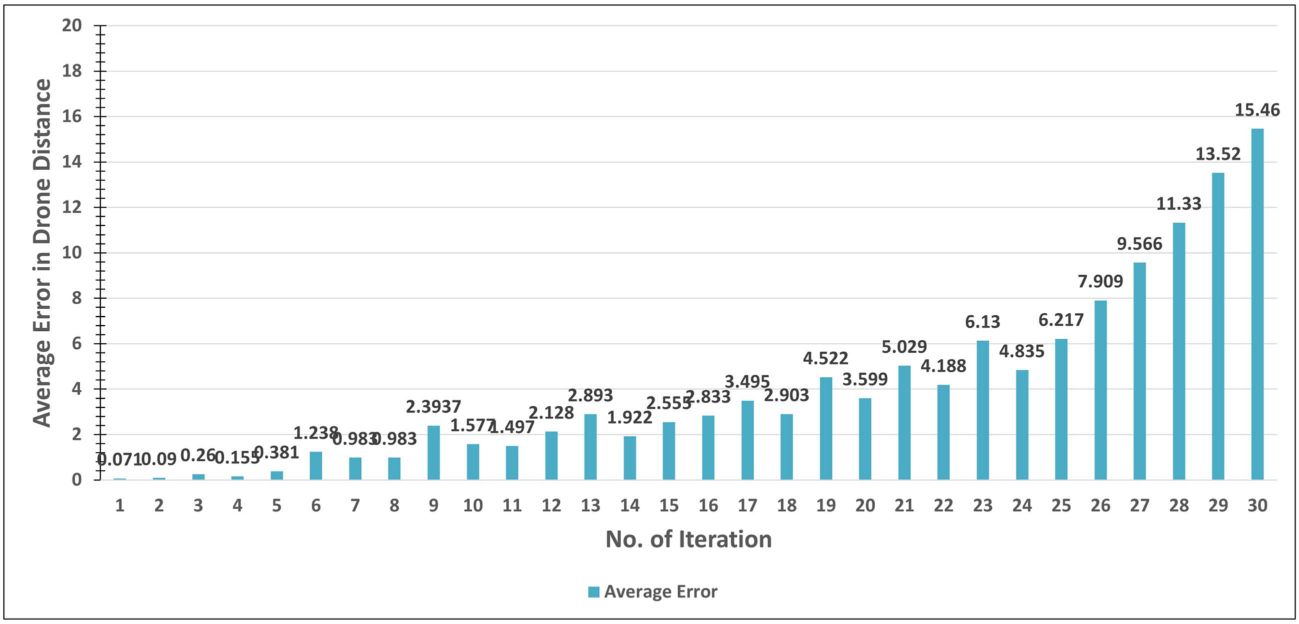
(b) Standard Deviation

Fig. 12 Output of 20 Drones

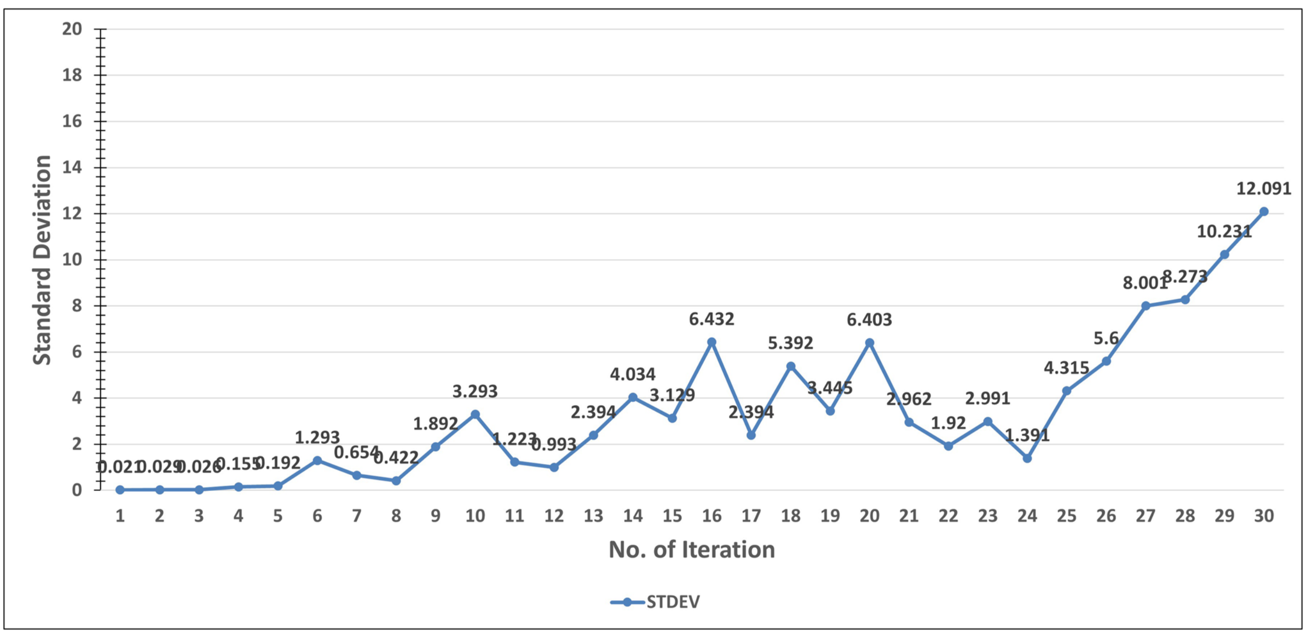
of the swarm behaviour, these initially equidistant drones gradually overlap each other as the simulation evolves.

The variations observed in the average error and standard deviation for the 10 drone swarm are a direct consequence of this dynamic behaviour. As drones move and overlap during the simulation, the distances between them change, leading to fluctuations in the localization estimates. This dynamic nature of the behaviour of the swarm, which results in overlapping drones, contributes to the significant variations seen in the localization accuracy.

When the experiments are extended to include 20 and 30 drones, the complexity of swarm dynamics increases, further accentuating the variations in localization estimates. The interactions between a larger number of drones within the swarm can lead to more intricate movement patterns and overlaps, amplifying the variations observed in the results.



(a) Accuracy Over time



(b) Standard Deviation

Fig. 13 Output of 30 Drones

In summary, the observed variations in the localization estimates are a direct reflection of the random and dynamic nature of drone swarm behaviour in our simulation. These variations should be seen as an intrinsic characteristic of swarm dynamics rather than a limitation of the proposed scheme. We believe that this understanding of swarm behaviour dynamics is crucial for designing effective swarm detection and tracking algorithms in real-world scenarios, where drone movements are inherently unpredictable and can lead to similar variations in localization accuracy.

4.5.2 Tracking of 20 Drones:

In this experiment, the number of drones is increased to 20 with random positions. The same procedure is repeated for 20 drones and tracked the single frame of all the 20 drones and took the Euclidean distance between actual and tracked coordinates. By comparing it with ground truth values and calculating the average error this experiment got high accuracy for each run. The overall average distance calculated is 0.097 drone distance and the standard $\sigma = 0.02$ which means data is quite consistent and is very close to the mean.

The experiment with 20 iterations for 20 drones at different positions is shown in Fig. 12(a). The bar graph gives the measured distance between the actual and tracked position and it increases significantly for every iteration as the swarm of drones gets denser at each time step which specifies the average accuracy over time. Moreover, Fig. 12(b) portrays the standard deviation of the experiment and it is evident that the standard deviation is low at the start and it increases rapidly with the increase in iterations. However, as can be seen in iterations 7, 12 and 14, there is a random behaviour of standard deviation, one possible reason for this behaviour is the uncontrolled random selection of drone position where the network is sparse and/or dense at different iterations.

These results show that our algorithm performs better at tracking drones in both sparse and dense networks when compared with the original ground truth values of drones, nonetheless, the accuracy of the algorithm decreases with the dense network and distance increases in iterations.

4.5.3 Tracking of 30 Drones:

This experiment is conducted by increasing the number of drones to 30. In this experiment, some drones overlap and are therefore not clearly visible to the camera. The overall average distance for 30 drones is 1.376 drone distance and the $\sigma = 0.659$ which means a higher number of drones leads to a denser swarm and makes it hard to track all the swarms with high accuracy. Overall the accuracy is higher when the drone count is 20 to 25 and decreases to some extent when the drone count is 30 to 35. However, the algorithm works quite well and is successful in detecting the maximum number of drones in a single frame.

The last experiment has 30 iterations for 30 drones at a random position. Figure 13(a) shows the accuracy (in terms of distance) for 30 drones at each iteration. when compared with the original coordinates it is evident that the distance between the actual and tracked position is a minimum of high accuracy at first however, the distance is gradually increasing accuracy decreasing with an increase in the number of iterations. The standard deviation of the distance between actual and tracked locations is shown in Fig. 13(b), and it is evident that our algorithm performs well at the start but the performance is affected when the network is denser.

With the denser network, it is difficult to track the precise locations and values standard deviation is entirely deviated from the mean and it is the limitation of our approach. Overall, in this experiment, it can be concluded that because of the density issue, the algorithm is not able to track the drones more accurately, however, the algorithm performs well for the sparse network.

Regarding the interpretation of Figs. 11, 12, and 13 and the question about whether the y-axis depicts the average error and standard deviation for all drones for each iteration or the average error and standard deviation for each drone across all iterations. This aspect has been discussed in detail in our previous responses, where we clarified that the results indeed represent the average error and standard deviation for all drones in each iteration.

To reiterate, the observed variations in these figures are primarily due to the inherent randomness and dynamic behaviour of the drone swarm during the simulation, which leads to changes in drone positions and overlaps. The provided results collectively contribute to our understanding of the algorithm's performance and adaptability under varying conditions.

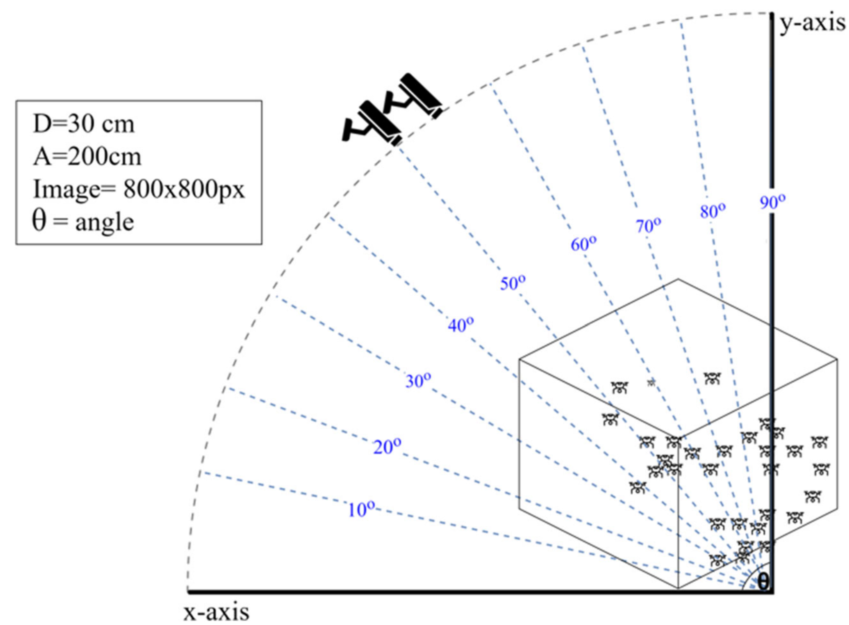
4.6 Experiment 4: Varying Camera Position

The aim of this experiment is to test varying camera positions to stress the performance of detection and tracking multiple drones over time under different conditions. The approach evaluates the accuracy of tracking by varying the position of the stereo camera at different angles. All experiments use an input image size of 800x800 pixels. This experiment demonstrates the details of tracking each drone over time by changing the angle of the camera at different positions of 0° to 90°. A YOLOv6 algorithm is employed to detect moving objects drones in real-time. Figure 14 shows the camera setup on different angles with visibility under 3 m at each angle.

In general, altering the viewpoint at which a swarm of drones is observed can have a significant impact on the amount of information and perspective that can be captured by the camera, even if the visibility distance between the camera and the swarm remains constant. Some of the views may include, **top-down view**, this perspective is useful for understanding the structure and behaviour of the drone as a whole. In addition, this viewpoint allows the observation of potential formations or configurations that the drones may form. Second, a **front/side view**, in which the camera is offset to one side of the swarm, can reveal interesting information about the lateral mobility of drones, it has the ability to give information on swarm's interactions with its surroundings (obstacles as well as drones).

This research considered the top-view scenario to observe the overall structure and behaviour of a swarm of drones. In this scenario, the experiment is conducted 10 times with random positions of drones. Furthermore, the trained model has the ability to identify drone swarm at varying angles ranging from 0° to 90° by improving detection results with predictions from the Kalman filter. To further validate the per-

Fig. 14 Camera Position with Varying Angles

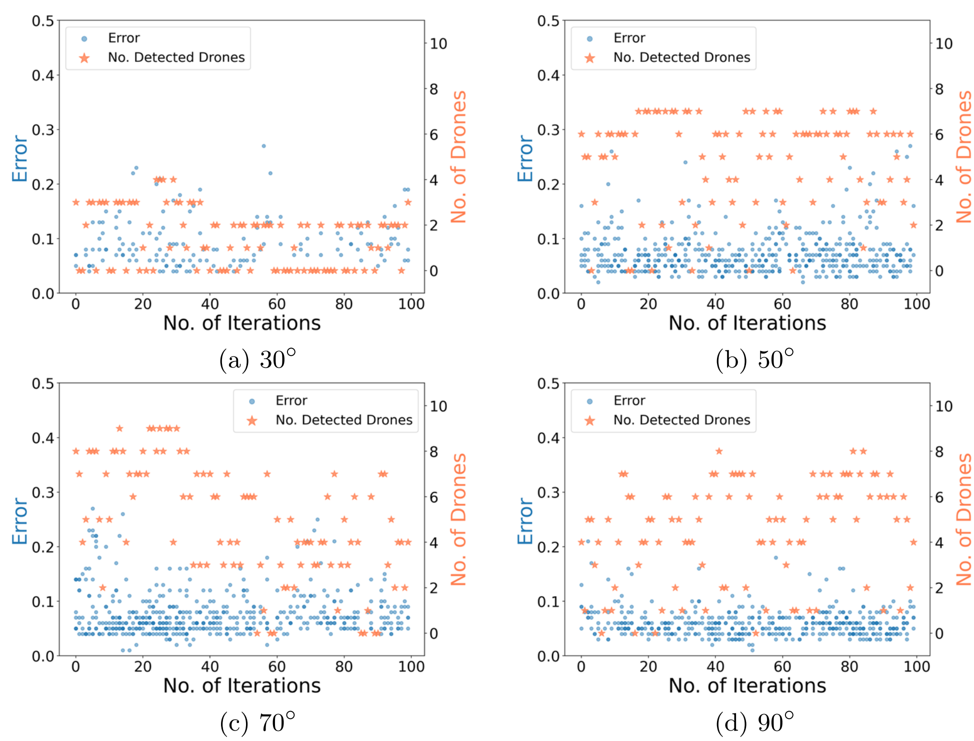


formance of drone detection, the presented framework was compared to the ground truth values obtained from the experiment. On the other hand, the trained model has achieved an 80% mAP (mean average precision) score at the intersection over the union threshold of 0.3.

In this experiment, the stereo camera angle is varied from 0° to 90°, and this experiment was repeated for 10 drones. Then, the tracking of various frames of drone locations takes place to calculate the Euclidean distance between actual and tracked locations of drones. Most of the drones are not visible to the camera from 0° to 20°. Figure 15, illustrates at 30°, a few drones are visible. The error at each angle is shown in graphs (Fig. 15) and it can be observed that our algorithm has a lower error at 90° almost all the drones are visible at this angle. The overall accuracy with varying angles can be seen in Fig. 16.

The presented line graph illustrates the number of drone detections across various angles. There are a total of ten

Fig. 15 Error Graphs for Varying Camera Angles



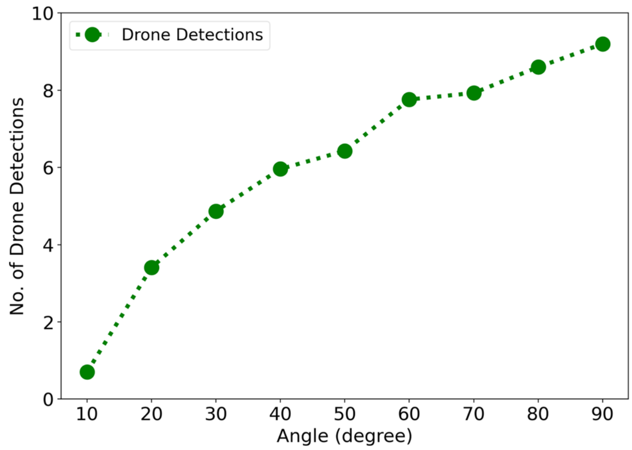


Fig. 16 Drone Detection at Varying Angles

drones and one hundred iterations. For each individual drone, we conducted 100 iterations for 10 drones with 1000 data points. After executing the simulation calculated the percentage of identified drones at each angle and plotted the resulting graph. This line facilitates the observation of the relationship between the number of drones and the variation in angle. The algorithm obtains higher accuracy when the camera angle is 90°. In Fig. 17 a box and whisker plot is presented, showcasing the distribution characteristics of errors. The plot consists of a rectangular box, whiskers, and individual data points (outliers) that fall outside the normal range of values. The x-axis represents the different angles of inclination, ranging from 30° to 90°. The y-axis indicates the magnitude of the errors, which provides insights into the variability and accuracy of measurements at each angle. It is interesting to note that the error at 30° inclination exhibits a relatively higher error value and a larger interquartile range compared to other angles. This suggests a tendency for measurements taken at 30° to have a higher central error value and greater variability. On the other hand, the error at 90° inclination displays a lower median value and a narrower interquartile range, indicating better accuracy and consistency in measurements. Experiments illustrate that the lack of detection is due to the highly mobile nature of drones, however, Kalman filter prediction has provided some useful information for tracking drones.

The seeming discrepancy between the results in Figs. 11, 15 and 18 arises from the different experimental setups and objectives underlying these figures. In Figs. 11, 12, and 13, we conducted a set of initial experiments with a fixed number of drones (10, 20, and 30, respectively) to evaluate the performance of the algorithm. However, it is important to note that in these experiments, the behaviour of the drones was inherently random.

This random behaviour meant that the drones started out in equidistant positions from one another, but as the simulation progressed, their movements became unpredictable. Consequently, we observed variations in localization accuracy among different iterations of these experiments. Figures 15 and 18 represent a distinct set of experiments where we deliberately manipulated various parameters, such as drone density, camera distance, and the number of drones, to assess the algorithm's adaptability and robustness under diverse conditions. In these scenarios, the randomness of drone behaviour was retained, reflecting real-world unpredictability.

What is crucial to understand is that the seemingly contradictory results highlight the ability of the algorithm to adapt to varying conditions. In scenarios with higher drone density, larger view angles, or other parameter changes,

Fig. 17 Error at Varying Angles

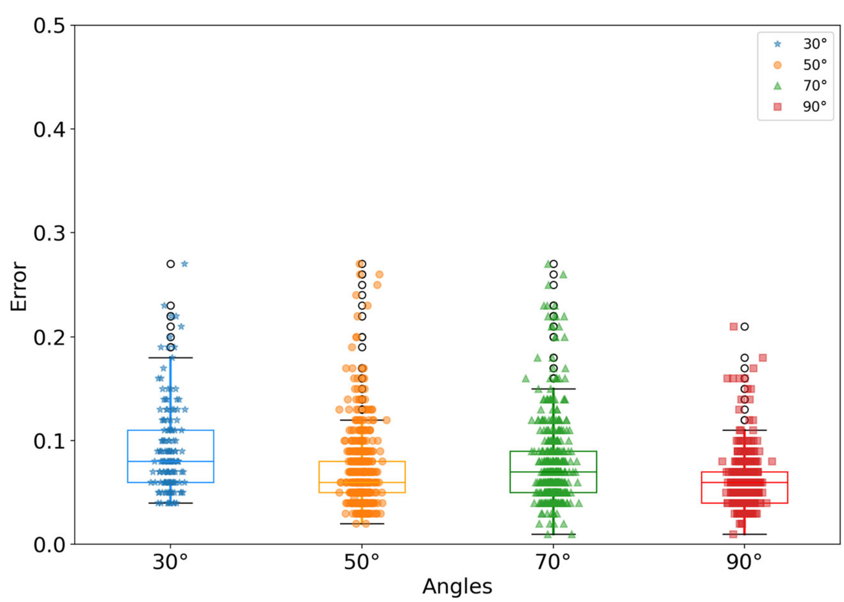
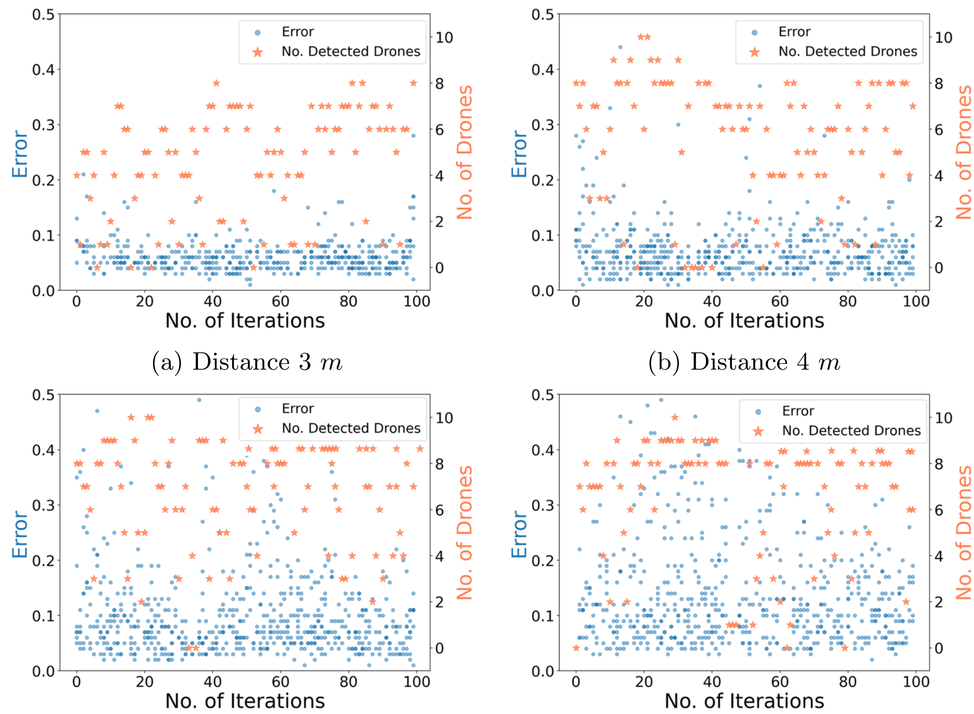


Fig. 18 Error Graphs for Varying Observer Distance



(a) Distance 3 m

(b) Distance 4 m

(c) Distance 5 m

(d) Distance 6 m

the algorithm demonstrated improved detection and tracking accuracy. This can be attributed to the capacity of the algorithm to handle complex swarm dynamics, even when drones exhibit random behaviours. The variations in performance under different conditions, as highlighted in Figs. 15 and 18 emphasize the potential of the algorithm to excel under varying scenarios. This adaptability is a valuable asset when dealing with real-world drone swarms, which are characterized by diverse and often unpredictable behaviours.

In summary, the variations in results between Figs. 11, 15 and 18 stem from the difference in experimental objectives. While the initial experiments showed variations due to the inherent randomness in drone behaviour, the subsequent experiments intentionally introduced variations in parameters to assess the algorithm's adaptability. These diverse scenarios and their corresponding results collectively contribute to a comprehensive analysis, allowing us to develop a more robust solution for drone swarm detection and tracking in real-world applications.

4.7 Experiment 5: Varying Observer Distance

The aim of this experiment is to evaluate the performance of the proposed framework by varying camera distance. This approach evaluates the accuracy of tracking by varying the position of the stereo camera for 10 drones to observe the drone swarms individual behaviour. The swarm can significantly change depth perception and clarity when deciding how far away to keep the camera few points should be considered.

When the camera is at $2\ m - 3\ m$, distance from the swarm, it can capture detailed images of the individual drones and their flight patterns. This can be helpful when evaluating drones one at a time or when researching swarm behaviour as a whole. Furthermore, when the camera is at $3.5\ m - 4.5\ m$, the distance from the swarm can observe the activity of the drone as well as any patterns or formations they may be forming. This can be helpful for investigating the swarm as a whole. Additionally, at a distance of $5\ m - 6\ m$, the camera will only observe the swarm as a whole and not be capable of identifying individual drones. In particular, the appropriate camera distance can change based on what exactly is being observed. To better understand the drone's individual behaviour and where they are in real-time within the swarm, a camera distance of $3\ m$ is recommended for the current simulation scenario, however, for real-world applications, a distance of a few kilometres is recommended with high-resolution cameras.

In this experiment, the stereo camera distance is varied from $2\ m - 6\ m$, and this experiment was repeated for 10 drones. Then, the tracking of various frames of drone locations takes place to calculate the Euclidean distance between the actual and tracked location of drones. The results achieved are quite encouraging, especially considering the aim was to identify a swarm of drones at varying camera distances. By looking at different graphs shown in Fig. 18 at $2\ m$ only a

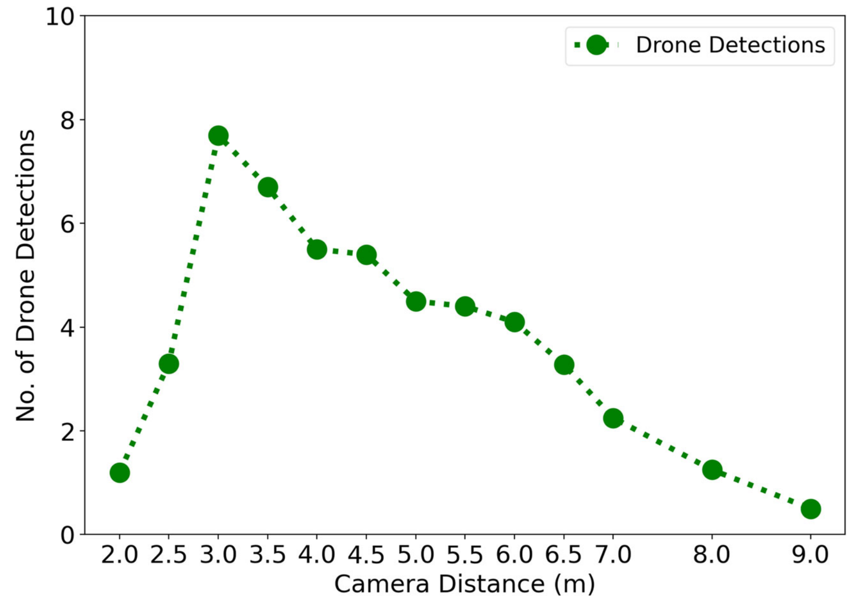


Fig. 19 Drone Detection at Varying Observer Distance

few of the drones are visible. The error at each camera distance is shown in graphs and it is apparent that our algorithm has promising results or higher accuracy at 3 m where most of the time all drones are visible. The overall accuracy with varying camera distances can be seen in Fig. 19.

The presented box and whisker chart in Fig. 20 offers an insightful representation of the variation in errors observed in a dataset collected at different camera distances. These distances range from $3\ m - 6\ m$. The horizontal axis (x-axis) on the chart represents the camera distances in meters. Specifically, it ranges from $3\ m - 6\ m$. The vertical axis (y-axis) quantifies the magnitude of errors, which measures the extent of deviation from the expected values. These plots comprise a rectangular box that symbolizes the interquartile range (IQR), which encapsulates the middle >50% of the error values. The lower and upper edges of the box signify the lower quartile (25th percentile) and upper quartile (75th percentile), respectively. The width of the box provides insights into the dispersion of errors for each camera distance. The whiskers visualize the range of error values. Data points falling beyond the whiskers are identified as outliers and are indicative of instances with exceptionally high or low error

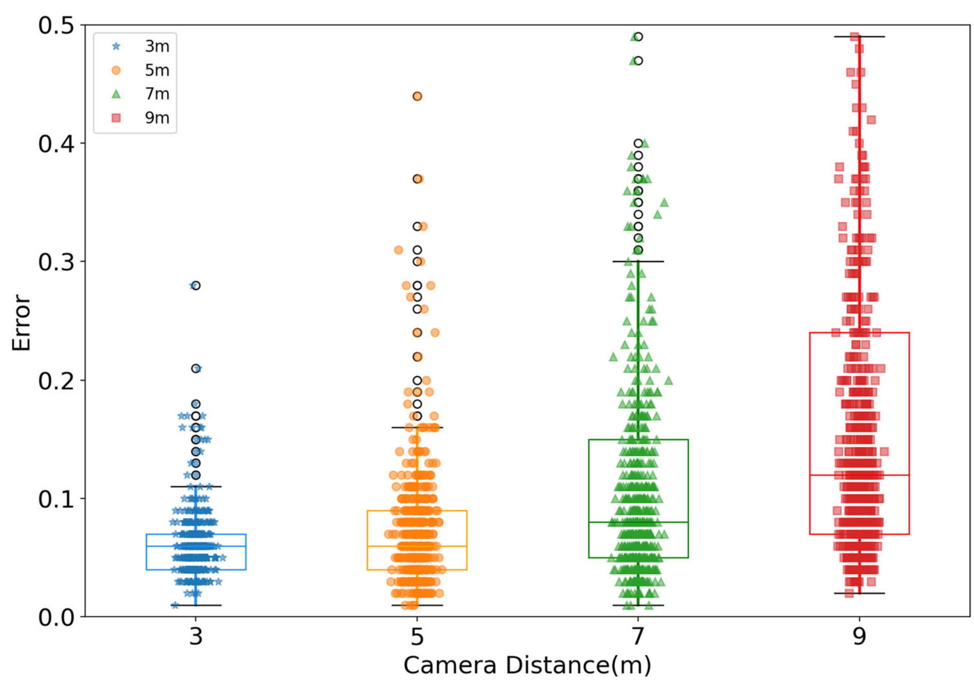


Fig. 20 Error at Varying Observer Distance

values. These outliers signify extreme deviations from the norm and offer insights into exceptional conditions during data collection.

4.8 Experiment 6: Varying Number of Drones

The aim of this experiment is to evaluate the performance of the proposed framework by varying the number of drones keeping the same camera distance. Changes in the number of drones in a swarm can affect the detail and perspective visible to the observer, even when the viewing distance is constant. Some factors to think about when adjusting the number of drone count are as follows:

If the swarm is small, the camera may get a far better view of the individual drones and their motions from the same camera distance. Possible applications include researching the behaviour of small swarms or tracking individual drones. Observers may be able to see the swarms general form and motion, as well as any patterns or formations the drones are forming if the swarm comprises a moderate number of drones. This could be helpful in understanding the social dynamics of a population of moderate size. If there are a large number of drones in the swarm, the observer may only see the swarm as a whole, with no way to tell them apart, even if the camera's distance is kept constant. This could be helpful for understanding the size and route of massive groups. It is worth noting that the optimal swarm size for different observations can differ greatly in the number of drones used. To better examine individual drone behaviour, for instance, a smaller swarm may be ideal. A swarm of medium or large size may be preferable for research on its collective behaviour.

In this experiment, the number of drones varied from 5 drones to 25 drones. Then, the tracking of various frames of all the drone locations takes place to calculate the Euclidean distance between the actual and tracked location of drones. The results achieved are quite encouraging, especially considering the aim was to identify a swarm of drones in varying numbers of drones. By looking at different graphs shown in Fig. 21 at varying numbers of drones the error can be seen in graphs and it is evident that our algorithm has promising results or higher accuracy at 10 drones graph where most of the time all drones are visible. The overall accuracy at increasing no. of drones can be seen in Fig. 22.

The box and whisker chart in Fig. 23 illustrates the fluctuating errors as the number of drones within an environment progressively increases, encompassing scenarios from 10 to 25 drones. A notable trend emerges as the number of drones increases from 10 to 25. At 10 drones, the error exhibits a lower median value and a narrower interquartile range, indicating higher accuracy and consistency in detection. Conversely, with 25 drones, the error displays a higher median value and wider interquartile range, reflecting decreased accuracy and heightened variability. This trend underscores the challenge of drone detection as the number of drones increases, primarily due to overlapping drones that limit visibility.

Regarding Fig. 22 it is important to note that this experiment involved varying the number of drones from 10 to 25,

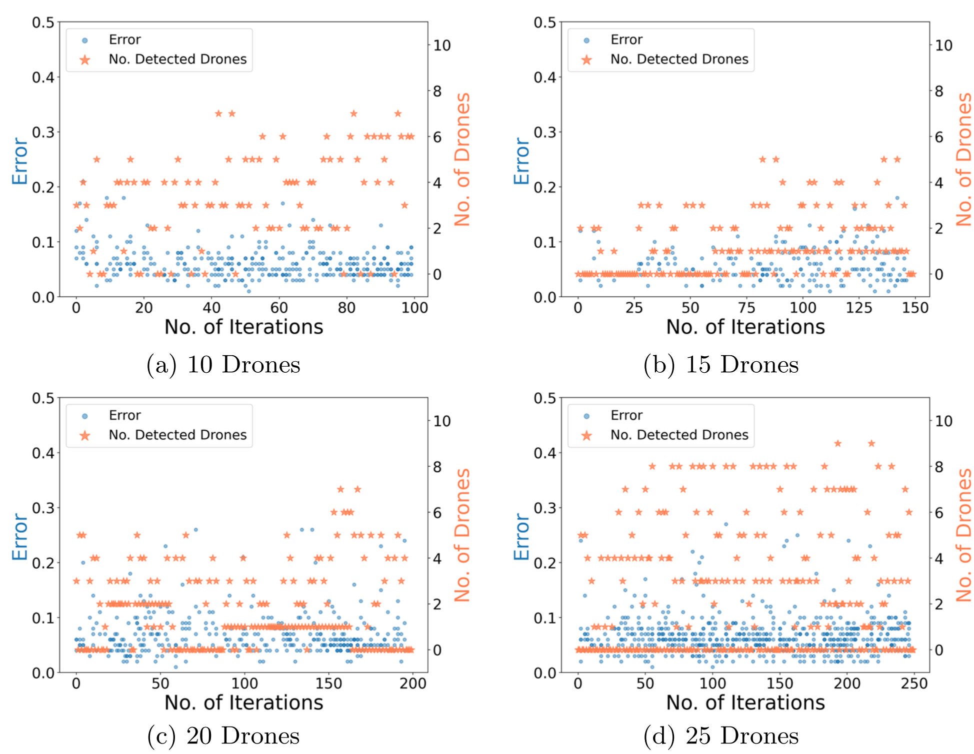


Fig. 21 Error Graphs for Varying no. of Drones

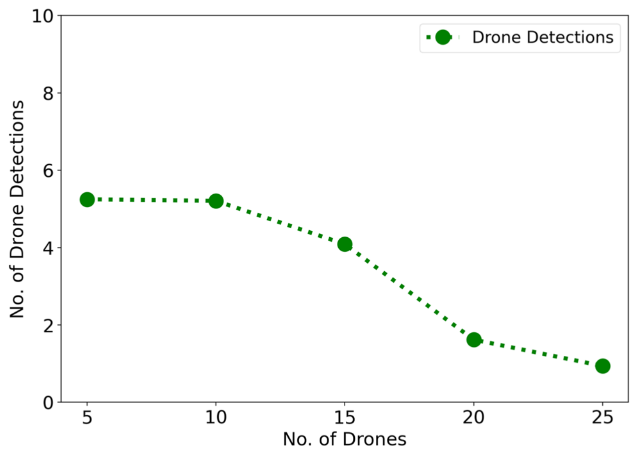


Fig. 22 Increasing no. of Drones

which had a direct impact on drone density within the swarm. In scenarios with low swarm density, where a higher number of drones were present within a fixed simulation space, the drones became densely overlapped and consequently were not visible to the camera. This led to a significant drop in detection accuracy, with less than 20% accuracy observed for a 25 drone swarm.

However, it is crucial to emphasize that this experiment was designed to investigate the performance of the algorithm under extreme conditions, where drones were intentionally made to overlap excessively. Such conditions are seldom encountered in practical drone swarm scenarios, where maintaining a certain minimum distance between drones is typically adhered to.

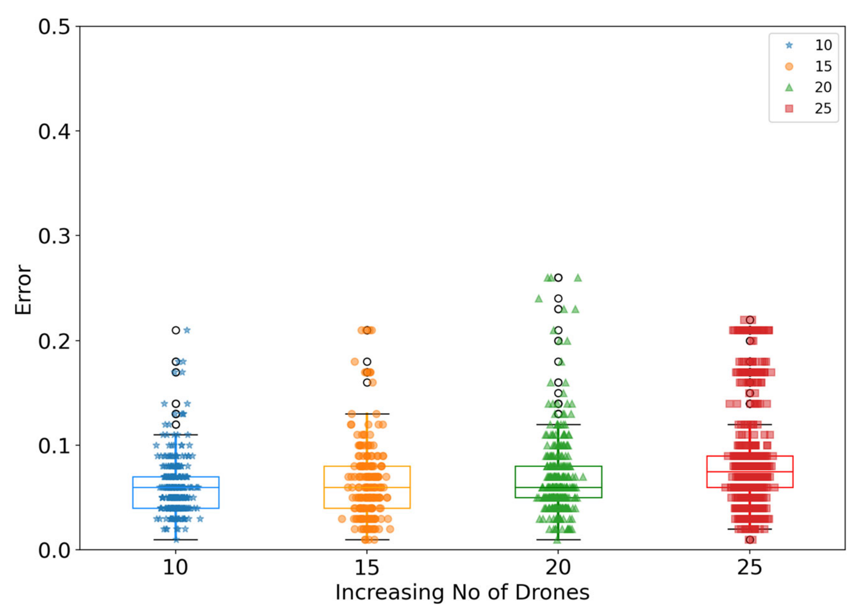
In all other experiments and scenarios in our research, we maintained a fixed number of drones, representing more typical operational conditions. Under these conditions, the algorithm consistently demonstrated good accuracy in detecting and tracking drones within the swarm.

Overall, Fig. 22 results offer insights into the behaviour of the algorithm when subjected to challenging and rare situations where drone density is exceptionally high due to an increased number of drones. These results do not reflect the algorithm overall performance under standard operational scenarios, where maintaining a practical drone density is the norm. Our research aims to provide a comprehensive understanding of the algorithm capabilities and limitations to develop a robust detection and tracking system for practical drone swarm applications.

4.9 Experiment 7: Varying Density of Drones

The aim of this experiment is to evaluate the performance of the proposed framework by varying density between drone swarms. The approach evaluates the accuracy of tracking by varying the density of drones keeping the same position of the stereo camera. To observe the drone swarm individual behaviour, the experiment was repeated for 10 drones in a dense environment. If the drones are far apart in the swarm, the observer may get a clear view of each one and its motions, regardless of how remote the camera is. This could be helpful for analysing the behaviour of sparse swarms or checking individual drones. With a medium density, an observer has a better chance of making out the swarm's general shape and motion, as well as any patterns or formations formed by the drones. This could be helpful in understanding how swarms of medium density interact with one another. When the drones in a swarm are packed in tightly, it can be impossible to tell them apart with the naked eye even if the cameras focal length is increased to its maximum. The density and

Fig. 23 Error at Increasing no. of Drones



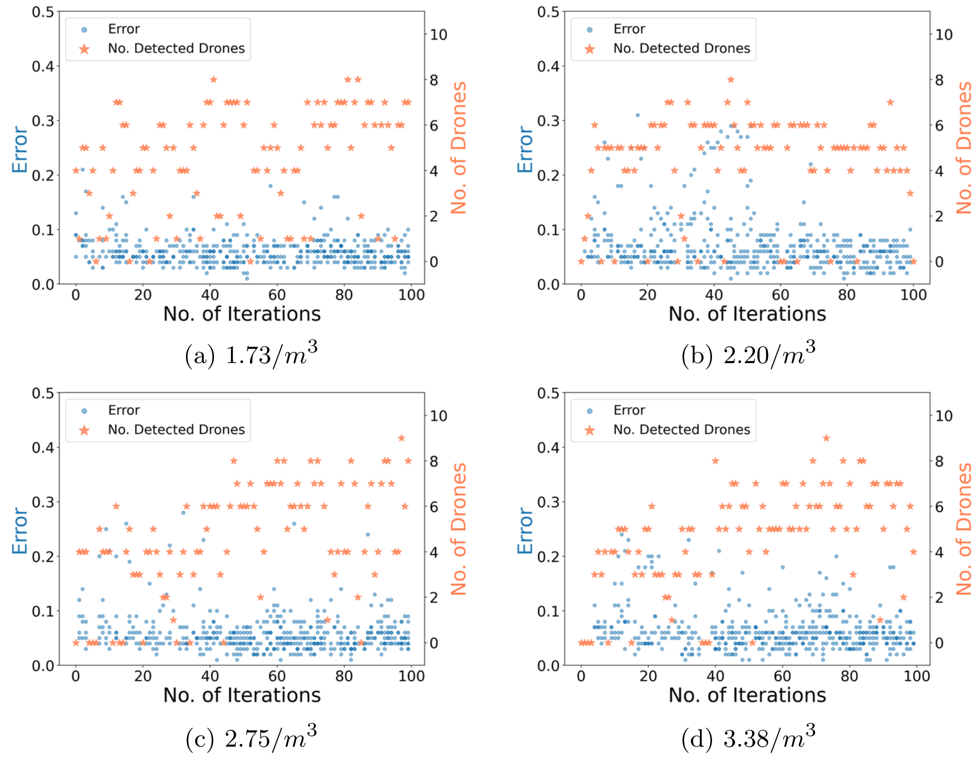


Fig. 24 Error Graphs for Varying Simulation Size (m^3)

motion of dense swarms could be better understood with this method. It is interesting to note that the optimal swarm density of drones can change with the goals of the observation. A low-density swarm may be preferable, for instance, if the purpose is to investigate drone behaviour at the individual level. A swarm of medium or high density may be preferable for research on its collective behaviour.

As in this experiment, the density of the swarm of drones varied from $1.33/m^3$ to $3.38/m^3$. This procedure repeated and tracked the multiple frames of all the drone locations. Then, the tracking of various frames of all the drone locations takes place to calculate the Euclidean distance between the actual and tracked location of drones. The results achieved are quite promising. By looking at different graphs shown in Fig. 24 at $1.73/m^3$ few of the drones are visible. Similarly, at each denser swarm graph, the error is shown and it is evident that our algorithm has promising results or higher accuracy at $3.38/m^3$ where most of the time all drones are visible. The overall accuracy with varying density can be seen in Fig. 25. The presented box and whisker chart in Fig. 26 offers a comprehensive visualization of errors observed as the simulation size for drones increases, with densities ranging from $1.73/m^3$ to $3.38/m^3$. The recorded errors are closely tied to the complexities of simulating drone operations, where lower errors indicate more precise modelling and higher errors are associated with challenges posed by increased drone density and overlapping. At $1.73/m^3$ errors exhibit a higher median value and a wider interquartile range, reflecting the complexities introduced by increased drone density and overlapping. Conversely, at$3.38/m^3$ the error displays a lower median value and a narrower interquartile range, indicating more accurate modelling and less variability.

4.10 Experiment 8: Comparison with Other Approaches

This experiment aims to compare the proposed approach with other state-of-the-art object-tracking methods. The experimental setup involves tracking a swarm of 10 drones using a stereo camera positioned at a 90° angle. This setup allows

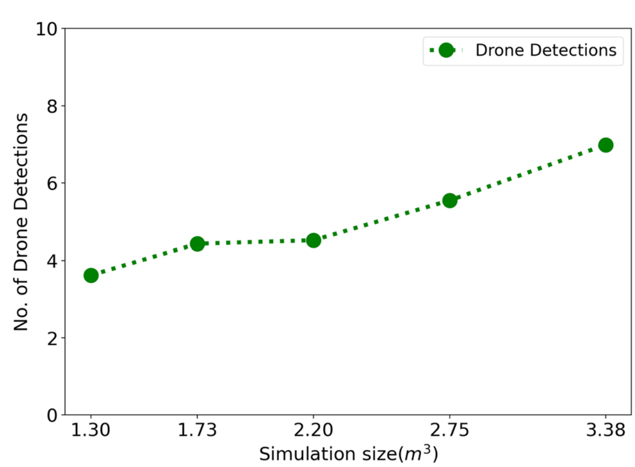


Fig. 25 Drone Detection with Varying Simulation Size

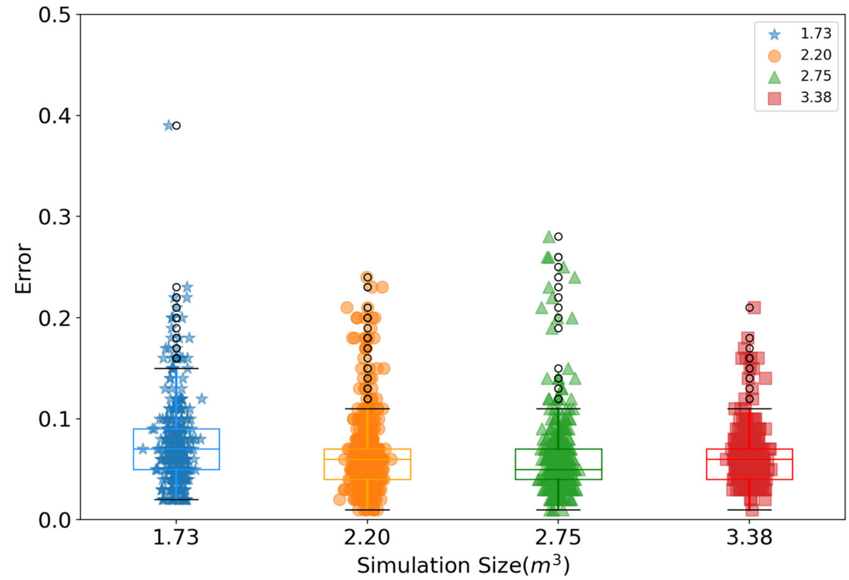


Fig. 26 Error Graph with Varying Simulation Size

us to use the triangulation method to extract images from both the left and right cameras to extract a depth map. This depth map serves as a foundational layer for generating point cloud data, representing the spatial distribution of the drones in three dimensions. The transformation from depth maps to point clouds is instrumental in our analysis, as it provides a detailed representation of the drone swarm's configuration in 3D space. With the point cloud data, we implemented various tracking algorithms, each designed to determine the precise position of each drone within the swarm. This configuration is chosen to evaluate the tracking accuracy of each method under identical conditions. The algorithms compared in this study are Deepsort [72], MOTDT (Multi-Object Tracking with Dual Matching Attentional Networks) [73], and Bytetrack [74]. The accuracy of each tracking algorithm is evaluated based on its ability to detect and track drones across 100 iterations. Each iteration represented a discrete time step in the drone's flight, with the position data generated to simulate real-world tracking scenarios. The performance metric used for comparison is the average tracking error, calculated as the deviation from the drone's actual positions. Under these conditions, the proposed algorithm consistently demonstrated good accuracy in detecting and tracking drones within the swarm.

By looking at different graphs shown in Fig. 27 faceted grid plot above visualizes the error trend across iterations for each tracking approach. Each subplot corresponds to one of the approaches, showing how the error changes over

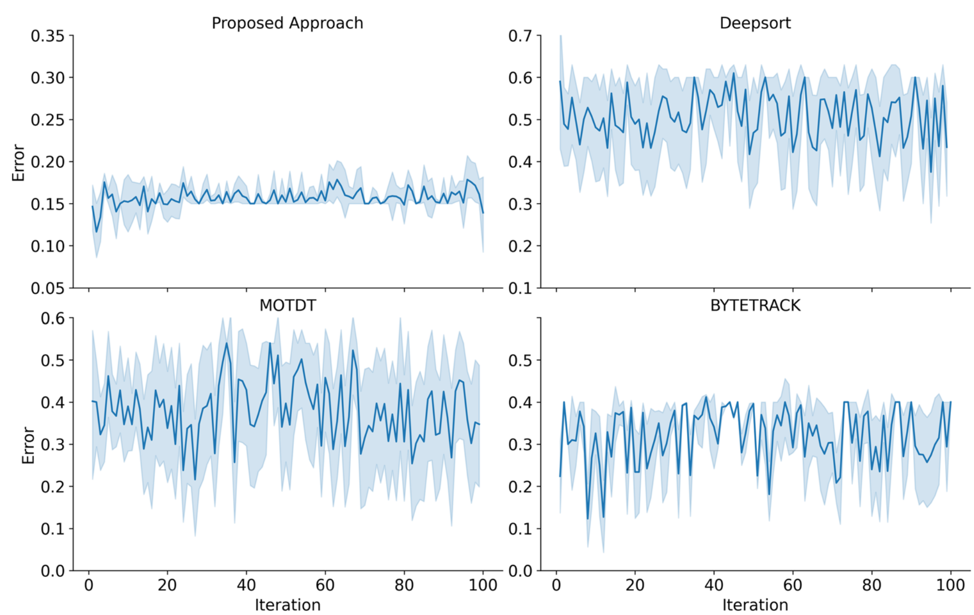


Fig. 27 Error Trends for Each Approach

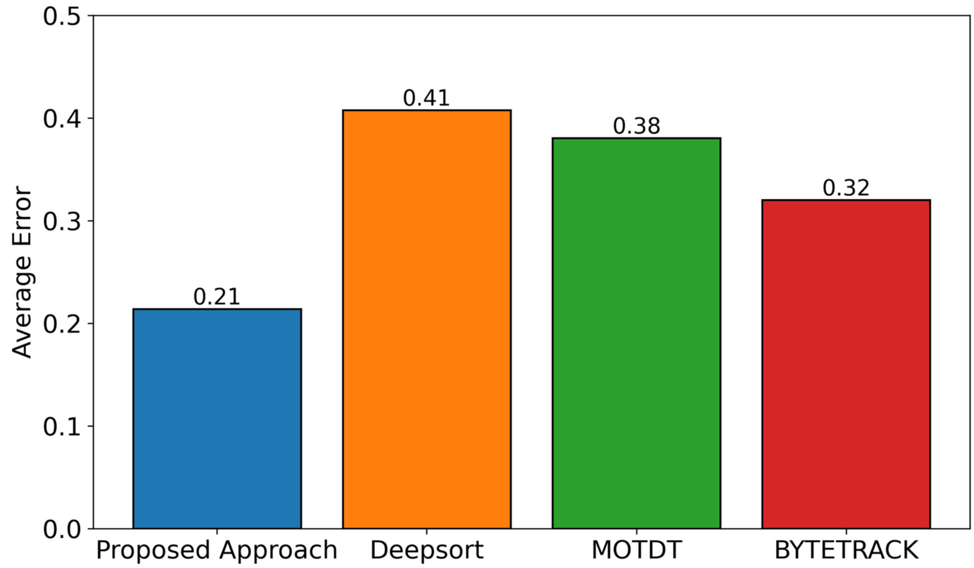


Fig. 28 Overall Average Error

iterations. This visualization technique allows for detailed comparison within the context of each approach while maintaining an overall view across the dataset.

In Fig. 28 a bar graph is constructed to display the average tracking error for each approach across 100 iterations. This graph highlights the numerical value of the error above each bar, enabling a clear comparison of the overall effectiveness of each tracking method. The proposed algorithm demonstrated superior accuracy in detecting and tracking drones within the swarm, showcasing its potential as a robust solution for drone tracking applications. The combined graphical representation in Fig. 29 the line graph shows the comparison of average error across all tracking algorithms (proposed approach, Deepsort, MOTDT, BYTETRACK) for the 10 drones based on the first 10 iterations. This comparative visualization makes it easier to see how each drone performed under different algorithms, providing further evidence of the proposed approach's enhanced performance.

5 Discussion

Drones or UAVs, have become more popular recently because of their adaptability and usability. But the growing number of drones has brought up concerns about privacy, protection, and personal security. In particular, public events, sensitive regions, and vital infrastructure can all be endangered by unauthorised drone use. To address this issue, this research explored methods for detecting and tracking drones using computer vision techniques. In the previous section,

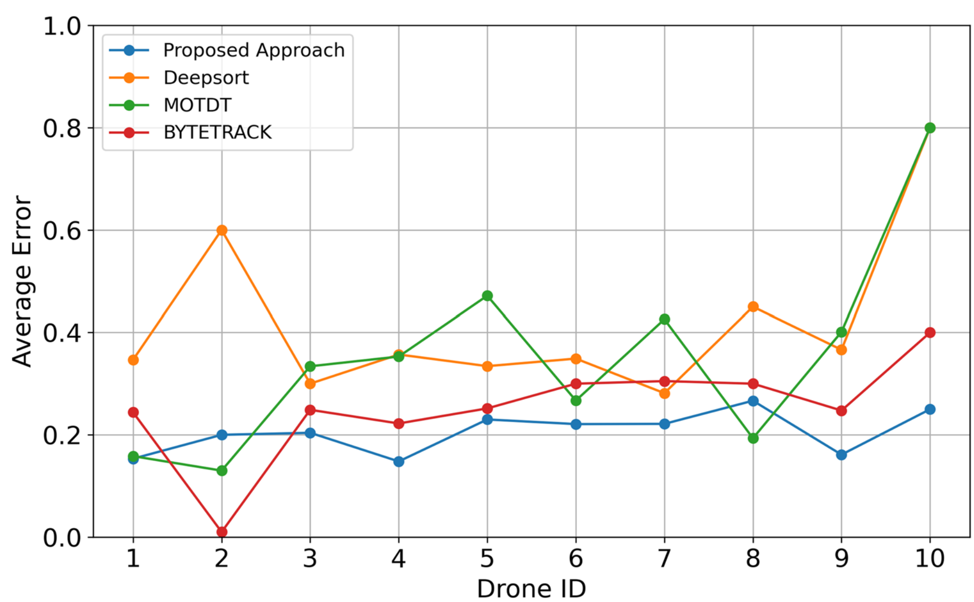


Fig. 29 Average Error for 10 Iterations

the results of drone swarm identification and tracking are presented using various methods. These findings highlight the efficacy of the presented method in detecting and tracking the maximum possible number of drones, both in dense and sparse networks.

In terms of the performance of the given framework, it is required that swarms exhibit sufficient distance between nodes to facilitate faster detection and tracking performance. However, as seen above, the very high level of density in the swarm results in poor performance of the algorithm. To overcome this challenge, utilizing images from a stereo camera for identifying drone swarms appears to be a feasible solution.

The Kalman filter-based tracking method possesses the smoothing and predictive properties necessary for a steady aim. Several experiments have been conducted to determine the algorithm's precision. Drone detection is tested using two drones at first and then the number of drones is gradually increased in multiple experiments. One problem that arose was that several bounding boxes, or false positives, were generated by the drone identification process. The NMS is employed to overcome this problem by obtaining reliable detections by extracting proper bounding boxes. These methods collectively contribute to a substantial improvement in the performance of the tracking framework for drone swarms, as evidenced by the experimental results. We have conducted extensive testing in varied environments and swarm densities, demonstrating that the presented framework significantly outperforms existing methods in terms of tracking accuracy, detection speed, and robustness against various challenges.

In addition to this another experiment for tracking drone swarms has been tested by changing drone numbers from 10 to 30. Every experiment computes the Euclidean distance between the actual and tracked location and takes the average mean. The 10 drones actual and tracked locations are very close, indicating good precision and with 20 drones, the average distance is and found average accuracy, indicating that the data is consistent and close to the mean. Ultimately, by raising the number of drones to 30 in the following experiment, some drones overlap and are not visible to the camera. As can be observed, when drone numbers are increased the swarm becomes denser due to this the accuracy decreases. It is evident that our algorithm performs well at the start but the performance is affected when the network is denser. Once the results were satisfying for detection and tracking of swarm drones, other experiments were tested by varying angles, varying number of drones, varying camera distance and finally varying density of drones. Overall, the proposed method is ideal for detecting multiple drones and tracking them with high precision. This can be further used to understand the behaviour/pattern in which swarm operations are taking place and tackle them accordingly.

In this study, a notable challenge was the dense arrangement of drones in the swarm, affecting the algorithm's performance. The findings indicate a reduction in accuracy, mainly due to drones overlapping when their number reached 30 in our experiments. Unfortunately, this overlap caused some drones to be hidden from the camera's view, making them undetectable.

The algorithm demonstrated effective performance in detecting and tracking swarm drones with high precision under less dense swarm conditions. However, as the density of the swarm increased, the algorithm's effectiveness declined, indicating a notable limitation.

To address this issue comprehensively, we conducted various experiments to assess the algorithm's robustness under different conditions, such as varying angles, the number of drones, camera distance, and drone density. While the algorithm showcased strong performance initially, the impact of swarm density became evident as these factors were altered.

Despite these challenges, it is crucial to emphasize that our proposed method proves to be well-suited for detecting multiple drones and tracking them with exceptional precision, particularly under less dense swarm scenarios. This capability holds significant promise for understanding and monitoring the behaviour and patterns of swarm operations, offering valuable insights for addressing them effectively.

Although this works conclusions are built on the algorithm's performance, they are generalizable and should apply to other algorithms that rely on effective communication between drones to discover the best possible answer. Extending the current ideas to include a large number of drone swarm detection and overcoming the overlapping issue with improved accuracy using an extended Kalman filter (EKF) will be the focus of future studies in this area.

In terms of the real-world performance of a drone swarm tracking system that combines YOLOv6 detection with Kalman filter can be influenced by several factors, including the processing power, density and speed of drone swarm, accuracy requirements frame rate, processing time and the number of tracked objects or swarms. The initial stage in utilizing YOLOv6 tiny involves the production of a custom dataset for the purpose of training. In general, it is recommended that each training class has a minimum of 2000 images. The frame rate refers to the number of frames (images) processed per second by the tracking system. A higher frame rate allows for more frequent updates and smoother tracking. However, increasing the frame rate also demands more computational resources. Processing time refers to the time it takes for the tracking system to process each frame, including YOLOv6 detection, Kalman filtering, and any other computations. Lower processing time is desirable for real-time applications to ensure timely updates. As the number of tracked objects or swarms increases, the processing time per frame generally increases. More objects require more YOLOv6 detections and more Kalman filter updates, resulting in longer processing times. If the processing time per frame exceeds the frame rate, the system may fall behind real-time, causing delays in updates and potentially affecting the quality of tracking. Hence In practice, frame rates for drone swarm scenarios can vary widely. For a general starting point, one might consider frame rates in the range of 15 to 30 frames per second (FPS). This range is commonly used for many real-time computer vision applications.

The objective of this research is to gain insights into solutions to infer the communication network topology of highly dynamic swarms. By identifying high-impact nodes we will fracture the swarm and will use a co-evolutionary technique to detect these high-impact nodes and to evolve countermeasures in the swarm that make them more robust towards targeted elimination. Processes developed in this project can be used to study bird flocks and fish schools. This work can also be used for defence purposes when one wants to fracture hostile swarms or one wants to deploy swarms that can withstand targeted eliminations style attacks.

In addition to the presented findings, further insights into the main results shed light on the significance and implications of our research:

1. **Swarm Density Impact:** The observed decrease in algorithm accuracy under high swarm density underscores the critical need for methods that can effectively handle densely packed drone swarms. This insight emphasizes the challenges associated with real-world scenarios where drones may operate in close proximity.
2. **Algorithm Robustness:** Despite the challenges posed by high-density swarms, our proposed method exhibited commendable robustness under less dense swarm conditions. This highlights the adaptability of the algorithm in scenarios where drones are more sparsely distributed, showcasing its versatility in various operational contexts.
3. **Extended Kalman Filter (EKF) Potential:** The identified challenge of decreased accuracy under high drone density points to the potential efficacy of an Extended Kalman Filter (EKF) as a future enhancement. Future research focusing on integrating an EKF could address the overlapping issue, further improving the algorithm's performance in dense swarm scenarios.
4. **Real-World Applicability:** The considerations regarding frame rate, processing time, and the number of tracked objects underscore the practical challenges of implementing drone swarm tracking systems in real-world settings. Striking a balance between these factors becomes imperative for achieving real-time tracking with optimal accuracy.

Parametric uncertainty, referring to the lack of precise knowledge about specific characteristics or settings within a model or system [75, 76], can significantly affect the performance of detection and tracking methods. In the case of YOLO for drone detection, this uncertainty might arise due to variations in lighting conditions, camera angles, and the visual appearance of drones in images. The Kalman filter used in tracking, uncertainties in the initial state estimate, imprecise knowledge of system dynamics, and variations in sensor measurements contribute to parametric uncertainty. We are already exploring this idea by tweaking different sets of parameters to see how it impacts the results. However, in the current study, I've been using a fixed set of parameters consistently without introducing any variations. In the current approach, a fixed set of parameters has been consistently employed for all experiments.

We utilized pre-trained model weights from the COCO dataset with YOLOv6. Fine-tuning our model on these pre-trained weights allowed us to benefit from existing knowledge while tailoring the model to our specific application.

In summary, our adoption of YOLOv6, and strategic use of pre-trained weights collectively contribute to the efficiency and adaptability of our deep-learning model for drone swarm tracking.

6 Conclusion and Future Work

The paper proposed a reliable identification and tracking method for drone swarms to understand the behaviour of each drone in a swarm. The proposed framework utilized a stereo-vision camera to observe a drone in a swarm. A deep learning-based detection method YOLOv6 is employed to detect the drone swarms in conjunction with NMS, and then, each detected drone swarm is tracked by using the Kalman filter technique. This research enhances the basic capabilities of YOLOv6 through specific improvements and the integration of advanced techniques, significantly improving drone swarm tracking. Integrating these components allows for the detection and tracking of multiple drones with high accuracy. YOLO efficiently detects drones in real-time, NMS reduces redundant detections, and the Kalman Filter accurately predicts the drones movements, considering both the current and past state uncertainties. The performance of the proposal was evaluated under a number of different scenarios by varying the position of the stereo camera, swarm density, number of drones, and observing camera distance. The findings show conclusively that the suggested framework is capable to identify and track drones in a swarm.

The experimental analysis clearly shows that the proposed tracking algorithm works better for tracking drones as compared to other methods. The comparative visualizations not only highlight its performance but also emphasize the algorithm's reliability, making it a potential tool for future drone tracking applications. The detailed research and creative approach to designing this algorithm open up new possibilities for more precise, dependable drone tracking solutions. This work highlights how important it is to develop technology specifically for drones as they become more common and their uses continue to grow.

To further improve the performance of the proposed framework in terms of accuracy and timely detection, the possibility of embedding an extended Kalman filter will be investigated. This would allow extending the current work to create a system that can handle uncertainty and noise, making it more robust in real-world circumstances.

In future work, there is a recognition of the necessity to go beyond the current focus on tracking individual drones. Drones in practical scenarios often operate in swarms with complex communication dynamics and diverse organisational structures (centralized, decentralized, adaptive intelligence). To tackle this complexity, active development is underway to create methodologies for visually analysing and interpreting coordinated swarm behaviours. This involves not only monitoring individual drone trajectories but also analysing patterns arising from communication and collaboration within the swarm.

Visually analysing swarm behaviour poses unique challenges due to its asynchronous nature. Unlike observing individual drones, where movements are relatively independent, swarm coordination involves intricate interactions that may not be readily apparent through visual observation alone. The asynchronous communication among swarm members further complicates the analysis, requiring innovative approaches to capture and interpret these dynamic interactions. Even with these challenges, gaining insights into the visual representation of swarm coordination is expected to significantly contribute to the robustness and applicability of simulated experiments, bringing a step closer to replicating real-world conditions.

Author Contributions All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Nisha Kumari. Experiments were conducted by Nisha Kumari. The first draft of the manuscript was written by Nisha Kumari and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

Code or data availability The code used during the current study is available from the corresponding author on a reasonable request.

Declarations

Conflicts of interest The authors declare that they have no conflicts of interest to this work.

Ethics Approval Not applicable.

Consent to Participate Not applicable.

Consent for Publication Not applicable.

References

1. Navarro, I., Matia, F.: An introduction to gswarm robotics. Int Scholarly Res Notices **2013** (2013)
2. Khaldi, B., Cherif, F.: An overview of swarm robotics: Swarm intelligence applied to multi-robotics. Int J Comput Appl **126**(2), 31–37 (2015)
3. Saha, H.N., Das, N.K., Pal, S.K., Basu, S., Auddy, S., Dey, R., Nandy, A., Pal, D., Roy, N., Mitra, D., et al.: A cloud based autonomous multipurpose system with self-communicating bots and swarm of drones. In: 2018 IEEE Annual Computing and Communication Workshop and Conference (CCWC), pp. 649–653 (2018)
4. Wang, X., Green, D., Barca, J.C.: Guidelines for improving the robustness of swarm robotic systems through adjustment of network topology. In: 2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), pp. 1399–1405 (2017)
5. Sparrow, R.J.: Killer robots: Ethical issues in the design of unmanned systems for military applications. In: Handbook of Unmanned Aerial Vehicles, pp. 2965–2983 (2015)
6. Gupta, M., Saxena, D., Kumari, S., Kaur, D.: Issues and applications of swarm robotics. Int. J. Res. Eng., Technol. Sci. **6**, 1–5 (2016)
7. Xiaohong, W., Zhang, Y., Lizhi, W., Dawei, L., Guoqi, Z.: Robustness evaluation method for unmanned aerial vehicle swarms based on complex network theory. Chinese J. Aeronautics **33**(1), 352–364 (2020)
8. Scharre, P.: Counter-swarm: a guide to defeating robotic swarms. https://warontherocks.com/2015/03/counter-swarm-a-guide-to-defeating-robotic-swarms (Accessed 02 April 2023)
9. Scharre, P.: Army of None: Autonomous Weapons and the Future of War. WW Norton & Company, New York (2018)
10. Beauchamp, G.: Social Predation: How Group Living Benefits Predators and Prey. Elsevier, UK (2013)
11. Procaccini, A., Orlandi, A., Cavagna, A., Giardina, I., Zoratto, F., Santucci, D., Chiarotti, F., Hemelrijk, C.K., Alleva, E., Parisi, G., et al.: Propagating waves in starling, sturnus vulgaris, flocks under predation. Animal Behaviour **82**(4), 759–765 (2011)
12. Hettiarachchige, Y., Khan, A., Barca, J.C.: Multi-object tracking of swarms with active target avoidance. In: 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), pp. 1204–1209 (2018)
13. Scharre, P.: How swarming will change warfare. Bullet. Atomic Scientists **74**(6), 385–389 (2018)
14. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: 2016 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)

15. Campion, M., Ranganathan, P., Faruque, S.: A review and future directions of uav swarm communication architectures. In: 2018 IEEE International Conference on Electro/information Technology (EIT), pp. 0903–0908 (2018)
16. Teague, E., Kewley Jr, R.H.: Swarming unmanned aircraft systems. Operations Research Center of Excellence (ORCEN) Technical Report DSE-TR-0808. West Point, NY: US Military Academy ORCEN (2008)
17. Burkle, A., Segor, F., Kollmann, M.: Towards autonomous micro uav swarms. J. Intell. & Robotic Syst. **61**, 339–353 (2011)
18. Husseini, T.: Gremlins are coming: DARPA enters Phase III of its UAV programme. https://www.army-technology.com/features/gremlins-darpa-uav-programme/ (Accessed 19 February 2023)
19. Airforce Technology: Initial Operating Capability for Swarm Drone Technology Could Be Achieved in Less Than Ten Years: Poll. https://www.airforce-technology.com/news/initial-operating-capability-for-swarm-drone-technology-could-be-achieved-in-less-than-ten-years-poll/ (Accessed 02 April 2023)
20. Bakhshipour, M., Ghadi, M.J., Namdari, F.: Swarm robotics search & rescue: A novel artificial intelligence-inspired optimization approach. Appl. Soft Comput. **57**, 708–726 (2017)
21. Liao, Y.-L., Su, K.-L.: Multi-robot-based intelligent security system. Artif. Life Robotics **16**(2), 137 (2011)
22. Shiomi, M., Kamei, K., Kondo, T., Miyashita, T., Hagita, N.: Robotic service coordination for elderly people and caregivers with ubiquitous network robot platform. In: 2013 IEEE Workshop on Advanced Robotics and Its Social Impacts, pp. 57–62 (2013)
23. Gerkey, B.P., Thrun, S., Gordon, G.: Parallel stochastic hill-climbing with small teams. In: Multi-Robot Systems. From Swarms to Intelligent Automata vol. III, pp. 65–77 (2005)
24. Lemaire, T., Alami, R., Lacroix, S.: A distributed tasks allocation scheme in multi-uav context. In: 2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004, vol. **4**, pp. 3622–3627 (2004)
25. D'Emidio, M., Frigioni, D., Navarra, A.: Exploring and making safe dangerous networks using mobile entities. In: 2013 International Conference on Ad-Hoc Networks and Wireless, pp. 136–147 (2013)
26. Ducatelle, F., Di Caro, G.A., Forster, A., Gambardella, L.: Mobile stigmergic markers for navigation in a heterogeneous robotic swarm. In: 2010 International Conference on Swarm Intelligence, pp. 456–463 (2010)
27. Kallenborn, Z.: The Era of the Drone Swarm is Coming, and We Need to be Ready for it. https://mwi.usma.edu/era-drone-swarm-coming-need-ready/ (Accessed 25 March 2023)
28. Embention: Drone Swarm Performance and Applications. https://www.embention.com/news/drone-swarm-performance-and-applications/ (Accessed 10 April 2023)
29. Chang, X., Yang, C., Wu, J., Shi, X., Shi, Z.: A surveillance system for drone localization and tracking using acoustic arrays. In: 2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM), pp. 573–577 (2018)
30. Chang, X., Yang, C., Shi, X., Li, P., Shi, Z., Chen, J.: Feature extracted doa estimation algorithm using acoustic array for drone surveillance. In: 2018 IEEE 87th Vehicular Technology Conference (VTC Spring), pp. 1–5 (2018)
31. Yang, C., Wu, Z., Chang, X., Shi, X., Wo, J., Shi, Z.: Doa estimation using amateur drones harmonic acoustic signals. In: 2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM), pp. 587–591 (2018)
32. Ganti, S.R., Kim, Y.: Implementation of detection and tracking mechanism for small uas. In: 2016 International Conference on Unmanned Aircraft Systems (ICUAS), pp. 1254–1260 (2016)
33. Hommes, A., Shoykhetbrod, A., Noetel, D., Stanko, S., Laurenzis, M., Hengy, S., Christnacher, F.: Detection of acoustic, electro-optical and radar signatures of small unmanned aerial vehicles. In: 2016 Target and Background Signatures II, vol. **9997**, p. 999701 (2016)
34. Hauzenberger, L., Holmberg Ohlsson, E.: Drone detection using audio analysis (2015)
35. Abdelkader, M., Guler, S., Jaleel, H., Shamma, J.S.: Aerial swarms: Recent applications and challenges. Current Robotics Reports, 1–12 (2021)
36. Sevil, H.E., Dogan, A., Subbarao, K., Huff, B.: Evaluation of extant computer vision techniques for detecting intruder suas. In: 2017 International Conference on Unmanned Aircraft Systems (ICUAS), pp. 929–938 (2017)
37. Hwang, S., Lee, J., Shin, H., Cho, S., Shim, D.H.: Aircraft detection using deep convolutional neural network in small unmanned aircraft systems. In: 2018 AIAA Information Systems-AIAA Infotech@ Aerospace, p. 2137 (2018)
38. Censi, A., Strubel, J., Brandli, C., Delbruck, T., Scaramuzza, D.: Low-latency localization by active led markers tracking using a dynamic vision sensor. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 891–898 (2013)
39. Nguyen, P.H., Kim, K.W., Lee, Y.W., Park, K.R.: Remote marker-based tracking for uav landing using visible-light camera sensor. Sensors **17**(9), 1987 (2017)
40. Ruiz, C., Pan, S., Bannis, A., Chen, X., Joe-Wong, C., Noh, H.Y., Zhang, P.: Idrone: Robust drone identification through motion actuation feedback. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **2**(2), 1–22 (2018)
41. Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., Kim, T.-K.: Multiple object tracking: A literature review. Artif Intell **293**, 103448 (2021)
42. Senanayake, M., Senthooran, I., Barca, J.C., Chung, H., Kamruzzaman, J., Murshed, M.: Search and tracking algorithms for swarms of robots: A survey. Robot. Autonomous Syst. **75**, 422–434 (2016)
43. Cimino, M.G., Massimiliano, L., Monaco, M., Vaglini, G., et al.: Adaptive exploration of a uavs swarm for distributed targets detection and tracking. In: 2019 Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods, vol. **1**, pp. 837–844 (2019)
44. Kumari, N., Lee, K., Ranaweera, C., Barca, J.C.: A comparison of clustering vs yolo for drone swarm centroid detection. In: 2023 8th International Conference on Robotics and Automation Engineering (ICRAE), pp. 106–111 (2023). https://doi.org/10.1109/ICRAE59816.2023.10458512
45. Kumari, N., Lee, K., Ranaweera, C., Barca, J.C.: Visually detecting drones in drone swarm formations topologies. In: International Conference on Information Technology and Applications, pp. 21–30 (2022). https://doi.org/10.1007/978-981-99-8324-7_3 . Springer
46. Coops, N.C., Goodbody, T.R., Cao, L.: Four steps to extend drone use in research. Nature **572**(7770), 433–435 (2019)
47. Johnson, J.: Artificial intelligence, drone swarming and escalation risks in future warfare. RUSI J. **165**(2), 26–36 (2020)
48. Yang, B., Huang, C., Nevatia, R.: Learning affinities and dependencies for multi-target tracking using a crf model. In: 2011 CVPR, pp. 1233–1240 (2011)
49. Pellegrini, S., Ess, A., Schindler, K., Van Gool, L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 261–268 (2009)
50. Case, E.E., Zelnio, A.M., Rigling, B.D.: Low-cost acoustic array for small uav detection and tracking. In: 2008 IEEE National Aerospace and Electronics Conference, pp. 110–113 (2008)
51. Tang, Q., Yu, F., Ding, L.: A grouping method for multiple targets search using swarm robots. In: 2016 International Conference on Swarm Intelligence, pp. 470–478 (2016)
52. Wang, X.: Intelligent multi-camera video surveillance: A review. Pattern Recognition Lett. **34**(1), 3–19 (2013)

53. Candamo, J., Shreve, M., Goldgof, D.B., Sapper, D.B., Kasturi, R.: Understanding transit scenes: A survey on human behavior-recognition algorithms. IEEE Trans. Intell. Transportation Syst. **11**(1), 206–224 (2009)
54. Li, X., Hu, W., Shen, C., Zhang, Z., Dick, A., Hengel, A.V.D.: A survey of appearance models in visual object tracking. ACM Trans. Intell. Syst. Technol. (TIST) **4**(4), 1–48 (2013)
55. Wu, Y., Lim, J., Yang, M.-H.: Online object tracking: A benchmark. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2411–2418 (2013)
56. Leal-Taixe, L., Milan, A., Reid, I., Roth, S., Schindler, K.: MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking (2015)
57. Scott-Samuel, N.E., Holmes, G., Baddeley, R., Cuthill, I.C.: Moving in groups: how density and unpredictable motion affect predation risk. Behavioral Ecol. Sociobiol. **69**(6), 867–872 (2015)
58. Ioannou, C., Tosh, C., Neville, L., Krause, J.: The confusion effect–from neural networks to reduced predation risk. Behavioral Ecol. **19**(1), 126–130 (2008)
59. Cavagna, A., Giardina, I., Orlandi, A., Parisi, G., Procaccini, A.: The STARFLAG handbook on collective animal behaviour: Part II, three-dimensional analysis (2008)
60. Singha, S., Aydin, B.: Automated drone detection using yolov4. Drones **5**(3), 95 (2021)
61. Jung, H.-K., Choi, G.-S.: Improved yolov5: Efficient object detection using drone images under various conditions. Appl. Sci. **12**(14), 7255 (2022)
62. Aydin, B., Singha, S.: Drone detection using yolov5. Eng **4**(1), 416–433 (2023)
63. Alsanad, H.R., Sadik, A.Z., Ucan, O.N., Ilyas, M., Bayat, O.: Yolo-v3 based real-time drone detection algorithm. Multimed. Tools Appl. **81**(18), 26185–26198 (2022)
64. Fan, H., Lin, L., Yang, F., Chu, P., Deng, G., Yu, S., Bai, H., Xu, Y., Liao, C., Ling, H.: Lasot: A high-quality benchmark for large-scale single object tracking. In: 2019 Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5374–5383 (2019)
65. Le, T.: Real-time object detection and tracking on drones. Undergraduate Res. & Mentoring Program **25** (2018)
66. Fan, H., Du, D., Wen, L., Zhu, P., Hu, Q., Ling, H., Shah, M., Pan, J., Schumann, A., Dong, B., et al.: Visdrone-mot2020: The vision meets drone multiple object tracking challenge results. In: 2020 European Conference on Computer Vision, pp. 713–727 (2020)
67. Li, Q., Li, R., Ji, K., Dai, W.: Kalman filter and its application. In: 2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS), pp. 74–77 (2015)
68. Kalman, R.E.: A new approach to linear filtering and prediction problems. J. Basic Eng. **82**(1), 35–45 (1960)
69. Cipra, T., Romera, R.: Robust kalman filter and its application in time series analysis. Kybernetika **27**(6), 481–494 (1991)
70. Lo, L.-Y., Yiu, C.H., Tang, Y., Yang, A.-S., Li, B., Wen, C.-Y.: Dynamic object tracking on autonomous uav system for surveillance applications. Sensors **21**(23), 7888 (2021)
71. Kumari, D., Kaur, K.: A survey on stereo matching techniques for 3d vision in image processing. Int. J. Eng. Manuf. **4**, 40–49 (2016)
72. Pereira, R., Carvalho, G., Garrote, L., Nunes, U.J.: Sort and deep-sort based multi-object tracking for mobile robotics: Evaluation with new data association metrics. Appl. Sci. **12**(3), 1319 (2022)
73. Zhu, J., Yang, H., Liu, N., Kim, M., Zhang, W., Yang, M.-H.: Online multi-object tracking with dual matching attention networks. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 366–382 (2018)
74. Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., Wang, X.: Bytetrack: Multi-object tracking by associating every detection box. In: European Conference on Computer Vision, pp. 1–21 (2022). Springer
75. Campbell, M.E., Grocott, S.C.: Parametric uncertainty model for control design and analysis. IEEE Trans. Control Syst. Technol. **7**(1), 85–96 (1999)
76. Bhattacharyya, S.: Robust control under parametric uncertainty: An overview and recent results. Annual Rev. Control **44**, 45–77 (2017)

Nisha Kumari is a PhD scholar at the School of Information Technology at Deakin University. She received her Master of Engineering degree in Information Technology from Mehran University of Engineering and Technology, Pakistan, in 2018 and her Bachelor of Engineering degree in Computer System Engineering from Quaid-e-Awam University of Engineering, Science, and Technology, Pakistan, in 2012. Her research interests include machine learning, robotics, computer vision, and swarm robotics.

Kevin Lee is an Associate Processor in Software Engineering and Internet of Things (IoT) at the School of Information Technology in Deakin University. Previously, he held academic positions at Nottingham Trent University, Murdoch University, University of Mannheim, and University of Manchester. He received his PhD in Computer Science from Lancaster University, UK in 2006. He is a recognised researcher in the area of distributed systems and autonomic computing. He has worked in the areas of IoT, Cloud Computing, Robotics, Embedded Systems, cyber-physical systems. He has over 100 publications in high quality journals and conferences, with a H-Index of 19 and over 1900 citations. His current main active research area is in IoT and its integration with Cloud Computing in support of dynamic IoT applications.

Jan Carlo Barca is affiliated with the Defence Science and Technology Group in Australia. He is also an Honorary Fellow at the school of information technology, Deakin University. Jan Carlo has worked on technology for swarms of robots and swarm intelligence projects since 2009. He has managed research projects for academia, defence and industry.

Chathurika Ranaweera received B.Sc. and Ph.D. degrees from the University of Peradeniya, Sri Lanka, and the University of Melbourne, Australia, respectively. She is currently an Associate Professor and Course Director of Master of Information Technology at Deakin University, Melbourne, Australia. She has published more than 90 papers in prestigious journals and international conferences so far. Her research interests include network optimization, optical wireless convergence, optical transport network architectures, secure IoT connectivity, energy efficient communications and networks, and machine learning applications in networking.