

# Drone Tracking and Object Detection By YOLO And CNN


Sajid Hameed HASAN1,

Prof. Dr. Galip CANSEVER2

1,2 Electrical and Computer Engineering, Altinbas university, Istanbul , Turkey

sajedhameed55@gmail.com1 ,

Galip.cansever@altinbas.edu.tr 2

	<p><b>Abstract</b> This thesis focuses on the utilization of YOLO (You Only Look Once) and CNN (Convolutional Neural Network) for real-time drone detection. The study explores the fundamentals of YOLO and CNN, including their working principles and mathematical equations for object detection. A specialized dataset comprising diverse drone images is collected and meticulously annotated for training the models. Evaluation of the trained models is conducted using established metrics such as mAP and IoU. The results highlight the models' performance compared to baseline approaches, demonstrating their strengths and limitations. A comprehensive workflow for drone detection employing YOLO and CNN is presented, encompassing dataset collection, model training, evaluation, and deployment stages. This research contributes valuable insights to the field of drone detection and offers prospects for future enhancements and applications.</p>
<p><b>Keywords:</b> Object detection, Object tracking, unmanned aerial vehicles (UAVs), Deep learning, Convolutional neural networks (CNN), Real-time performance, Surveillance, Long-range videos, Distance estimation, YOLO (You Only Look Once) algorithm, Improved YOLO architecture, Drone detection, Image processing. Agricultural monitoring.</p>	

## Introduction

Imagine a world where unmanned drones autonomously navigate through complex environments, seamlessly detecting and tracking objects with precision and efficiency. This futuristic vision is becoming a reality through cutting-edge technologies such as YOLO (You Only Look Once) and CNN (Convolutional Neural Networks), revolutionizing the field of drone tracking and object detection.

In recent years, the use of unmanned aerial vehicles (UAVs) [1], commonly known as drones, has grown exponentially across various fields including surveillance, search and rescue operations, and package delivery. One critical aspect of drone technology is the ability to accurately track and detect objects in real-time, enabling autonomous navigation and intelligent decision-making. To address this need, advanced computer vision techniques have emerged, among which YOLO (You Only Look Once) and CNN (Convolutional Neural Networks) have gained significant attention.

YOLO is an object detection algorithm that provides remarkable speed and accuracy by dividing the input image into a grid and predicting bounding boxes and class probabilities directly. CNN, on the other hand, is a deep learning architecture that has shown exceptional capabilities in learning and recognizing visual patterns from large-scale image datasets [2]. Both YOLO and CNN have exhibited promising results in object detection tasks, offering the potential for efficient and reliable drone tracking and object recognition.

Despite the advancements in drone technology and computer vision algorithms, there still exist challenges in achieving robust and real-time drone tracking and object detection. The existing methods often struggle with accurately detecting objects of varying sizes, dealing with occlusions, and maintaining high precision in complex environments. Moreover, the implementation and optimization of YOLO and CNN for drone-based applications require further investigation to enhance their performance [3].

Therefore, the specific problem addressed in this thesis is to improve the accuracy, speed, and robustness of drone tracking and object detection by leveraging the combined power of YOLO and CNN algorithms. This research aims to fill the gap in knowledge by exploring novel techniques to enhance the detection and tracking capabilities of drones in real-world scenarios. By doing so, this study contributes to the existing body of knowledge in the field of computer vision and reinforces the practical applications of drones in areas such as surveillance, disaster response, and autonomous navigation.

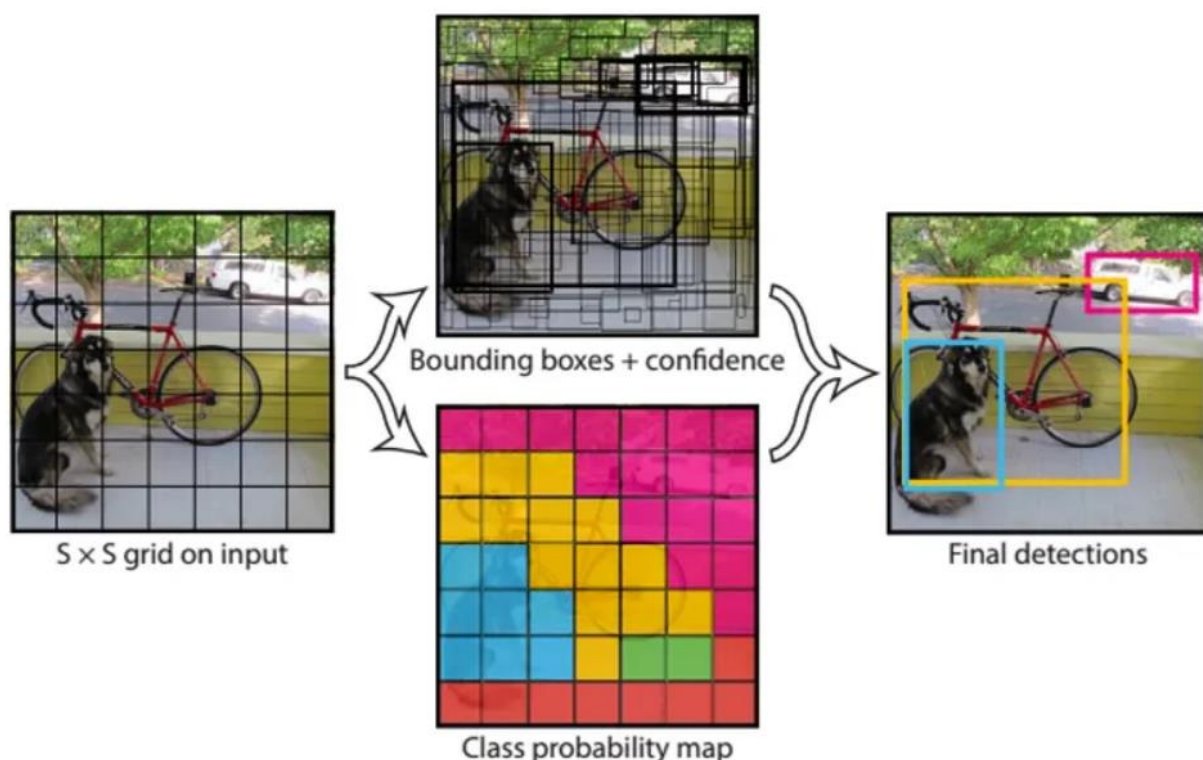


FIG 1. YOLO BASE Architecture [8]

The main objectives of this study are twofold. Firstly, we aim to assess the effectiveness of YOLO and CNN algorithms in drone tracking and object detection tasks through a comprehensive evaluation on benchmark datasets. By systematically analyzing the performance of these

algorithms, we seek to determine their capabilities and limitations in accurately detecting and tracking objects in various scenarios. Secondly, we intend to explore techniques that can enhance the accuracy and robustness of object detection specifically tailored for drones. Our focus lies on addressing challenges such as occlusions and objects of varying sizes, which often pose difficulties in achieving reliable detection results [6]. By investigating and developing novel approaches to handle these challenges, we aim to improve the overall precision and reliability of object detection for drone applications.

In addition to improving accuracy, our study also aims to enhance the real-time capabilities of drone tracking and object detection systems. We plan to achieve this by employing optimization and parallelization strategies that are specifically designed for YOLO and CNN architectures. These techniques will be explored to ensure efficient processing of video streams and real-time response, enabling drones to effectively detect and track objects without significant delays.

Furthermore, we seek to evaluate the suitability and efficiency of the combined YOLO and CNN approach for practical drone applications. This assessment will take into consideration various factors, including computational resources, power consumption, and scalability. By conducting rigorous experiments and

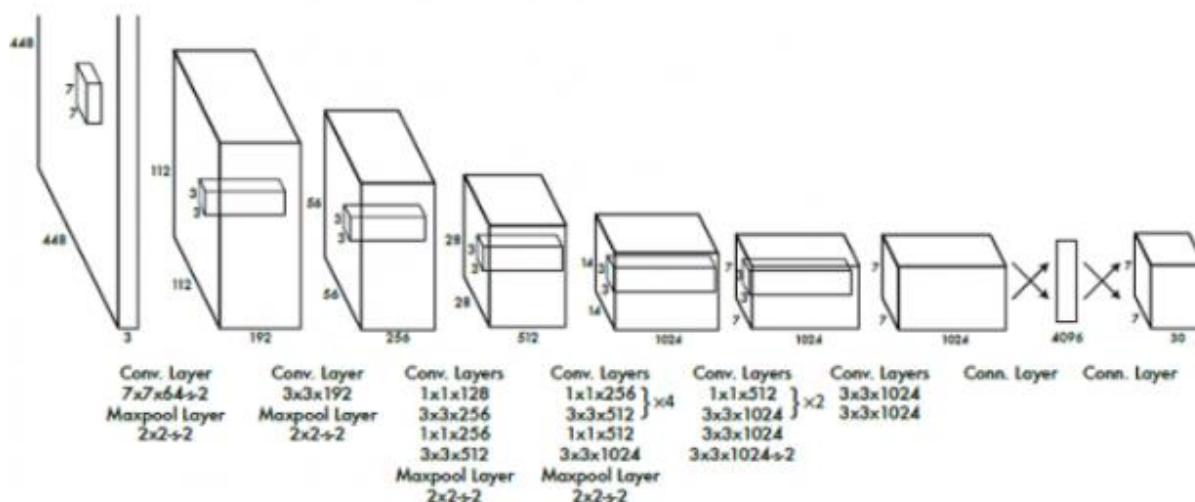


FIG 2. YOLO BASE Architecture [4]

analysis, we aim to determine the feasibility and potential impact of implementing drone tracking and object detection systems using YOLO and CNN algorithms in real-world scenarios.

These objectives have been formulated to address the research problem of achieving accurate and efficient drone tracking and object detection using YOLO and CNN algorithms. By systematically evaluating their performance, exploring novel techniques, and optimizing their real-time capabilities, this study aims to contribute to the advancement of drone-based computer vision systems. Furthermore, the assessment of their suitability for practical applications will shed light on the feasibility and potential impact of deploying such systems in real-world scenarios.

The main objectives of this study are twofold. Firstly, we aim to assess the effectiveness of YOLO and CNN algorithms in drone tracking and object detection tasks through a comprehensive evaluation on benchmark datasets. By systematically analyzing the performance of these

algorithms, we seek to determine their capabilities and limitations in accurately detecting and tracking objects in various scenarios. Secondly, we intend to explore techniques that can enhance the accuracy and robustness of object detection specifically tailored for drones. Our focus lies on addressing challenges such as occlusions and objects of varying sizes, which often pose difficulties in achieving reliable detection results. By investigating and developing novel approaches to handle these challenges, we aim to improve the overall precision and reliability of object detection for drone applications.

In addition to improving accuracy, our study also aims to enhance the real-time capabilities of drone tracking and object detection systems. We plan to achieve this by employing optimization and parallelization strategies that are specifically designed for YOLO and CNN architectures. These techniques will be explored to ensure efficient processing of video streams and real-time response, enabling drones to effectively detect and track objects without significant delays.

Furthermore, we seek to evaluate the suitability and efficiency of the combined YOLO and CNN approach for practical drone applications. This assessment will take into consideration various factors, including computational resources, power consumption, and scalability. By conducting rigorous experiments and analysis, we aim to determine the feasibility and potential impact of implementing drone tracking and object detection systems using YOLO and CNN algorithms in real-world scenarios.

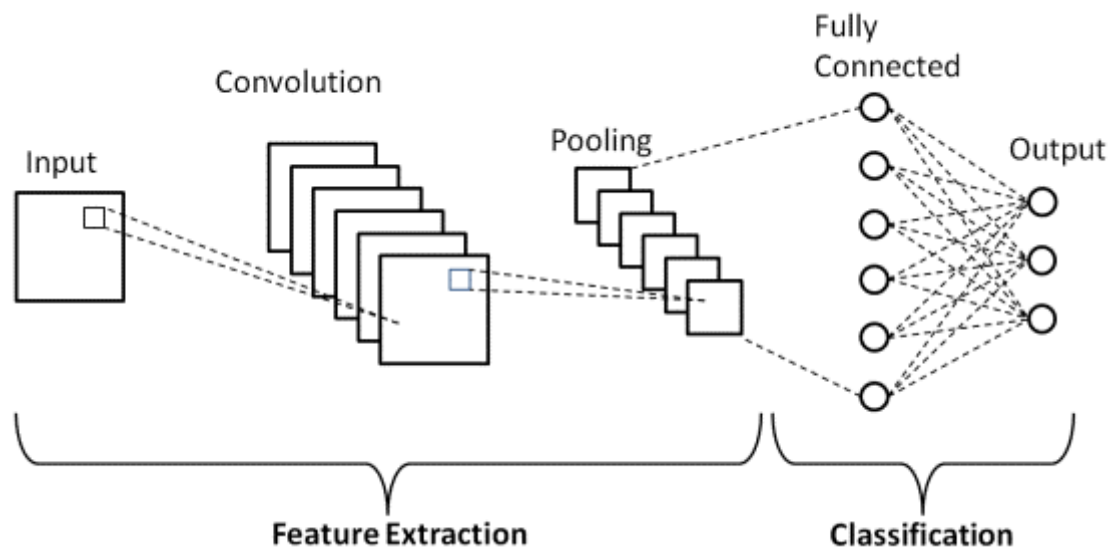


FIG- 3. CNN BASE Architecture [12]

These objectives have been formulated to address the research problem of achieving accurate and efficient drone tracking and object detection using YOLO and CNN algorithms. By systematically evaluating their performance, exploring novel techniques, and optimizing their real-time capabilities, this study aims to contribute to the advancement of drone-based computer vision systems. Furthermore, the assessment of their suitability for practical applications will shed light on the feasibility and potential impact of deploying such systems in real-world scenarios.

this research holds significant importance in advancing the field of drone tracking and object detection by leveraging YOLO and CNN algorithms. By undertaking a comprehensive evaluation

of these algorithms and exploring techniques to enhance their performance, this study addresses existing challenges and fills a crucial gap in the current knowledge.

The potential impact of this research is multifaceted. Firstly, the findings of this study contribute to the body of knowledge in computer vision and drone technology, specifically in the domains of tracking and object detection. The evaluation of YOLO and CNN algorithms, coupled with the exploration of novel techniques, provides valuable insights into the capabilities, limitations, and potential areas for improvement in the context of drone-based applications. This knowledge enhances our understanding of the strengths and weaknesses of these algorithms, enabling researchers and practitioners to make informed decisions regarding their implementation [7].

Secondly, the developed methodologies and optimizations have practical implications for real-world applications. Accurate and efficient drone tracking and object detection are essential in various domains such as surveillance, search and rescue operations, and autonomous navigation. By enhancing the precision, robustness, and real-time capabilities of drone systems, the proposed approach opens up new possibilities for improved performance and effectiveness in these practical scenarios [8]. This research has the potential to revolutionize the way drones operate, enabling them to navigate complex environments, detect objects with greater accuracy, and contribute to tasks such as disaster response and public safety.

Furthermore, the outcomes of this study can serve as a foundation for further research and development in the field of drone-based computer vision systems. The insights gained, along with the methodologies and optimization strategies proposed, offer a steppingstone for future investigations and advancements in this area. By addressing the challenges faced in drone tracking and object detection, this research facilitates the progress of the field, paving the way for new techniques, algorithms, and applications.

The potential impact of this research is multifaceted. Firstly, the findings of this study contribute to the body of knowledge in computer vision and drone technology, specifically in the domains of tracking and object detection. The evaluation of YOLO and CNN algorithms, coupled with the exploration of novel techniques, provides valuable insights into the capabilities, limitations, and potential areas for improvement in the context of drone-based applications. This knowledge enhances our understanding of the strengths and weaknesses of these algorithms, enabling researchers and practitioners to make informed decisions regarding their implementation.

Secondly, the developed methodologies and optimizations have practical implications for real-world applications. Accurate and efficient drone tracking and object detection are essential in various domains such as surveillance, search and rescue operations, and autonomous navigation. By enhancing the precision, robustness, and real-time capabilities of drone systems, the proposed approach opens new possibilities for improved performance and effectiveness in these practical scenarios. This research has the potential to revolutionize the way drones operate, enabling them to navigate complex environments, detect objects with greater accuracy, and contribute to tasks such as disaster response and public safety.

Furthermore, the outcomes of this study can serve as a foundation for further research and development in the field of drone-based computer vision systems. The insights gained, along with the methodologies and optimization strategies proposed, offer a steppingstone for future investigations and advancements in this area. By addressing the challenges faced in drone tracking

and object detection, this research facilitates the progress of the field, paving the way for new techniques, algorithms, and applications.

n by leveraging YOLO and CNN algorithms. By undertaking a comprehensive evaluation of these algorithms and exploring techniques to enhance their performance, this study addresses existing challenges and fills a crucial gap in the current knowledge.

The potential impact of this research is multifaceted. Firstly, the findings of this study contribute to the body of knowledge in computer vision and drone technology, specifically in the domains of tracking and object detection. The evaluation of YOLO and CNN algorithms, coupled with the exploration of novel techniques, provides valuable insights into the capabilities, limitations, and potential areas for improvement in the context of drone-based applications. This knowledge enhances our understanding of the strengths and weaknesses of these algorithms, enabling researchers and practitioners to make informed decisions regarding their implementation.

Secondly, the developed methodologies and optimizations have practical implications for real-world applications. Accurate and efficient drone tracking and object detection are essential in various domains such as surveillance, search and rescue operations, and autonomous navigation. By enhancing the precision, robustness, and real-time capabilities of drone systems, the proposed approach opens up new possibilities for improved performance and effectiveness in these practical scenarios. This research has the potential to revolutionize the way drones operate, enabling them to navigate complex environments, detect objects with greater accuracy, and contribute to tasks such as disaster response and public safety.

Furthermore, the outcomes of this study can serve as a foundation for further research and development in the field of drone-based computer vision systems. The insights gained, along with the methodologies and optimization strategies proposed, offer a steppingstone for future investigations and advancements in this area. By addressing the challenges faced in drone tracking and object detection, this research facilitates the progress of the field, paving the way for new techniques, algorithms, and applications.

## 1.2. Motivation

Drone technology has witnessed significant advancements in recent years, revolutionizing various industries and opening new possibilities in fields such as surveillance, agriculture, and disaster management. One crucial aspect of drone capabilities lies in their ability to accurately track and detect objects in real-time. However, achieving reliable and efficient object detection poses several challenges, including the complexities of real-world environments, varying object scales, occlusions, and limited computational resources on drones.

The motivation behind this thesis stems from the pressing need to address these challenges and enhance the performance of drone-based object detection systems. Accurate and efficient object detection is crucial for applications such as aerial surveillance, search and rescue operations, and autonomous navigation. Current object detection algorithms, including YOLO (You Only Look Once) and CNN (Convolutional Neural Network), have demonstrated promising results in computer vision tasks. However, their performance and suitability for drone-based applications require further investigation and optimization.

By exploring the potential of YOLO and CNN algorithms in the context of drone tracking and object detection, this research aims to overcome the limitations of existing techniques and

contribute to the advancement of drone-based computer vision systems. The utilization of these algorithms can provide real-time object detection capabilities to drones, enabling them to navigate complex environments, identify objects of interest, and respond to dynamic situations effectively. Moreover, the practical applications of this research are far-reaching. Improved drone tracking and object detection can greatly enhance the effectiveness of surveillance systems, allowing for quicker threat identification and response. In agriculture, drones equipped with robust object detection capabilities can aid in crop monitoring and pest control, optimizing resource allocation and improving yields. Furthermore, in disaster management scenarios, drones can play a crucial role in rapidly assessing affected areas and identifying survivors, thereby saving lives.

The significance of this research also extends to academia. By thoroughly evaluating the performance of YOLO and CNN algorithms in the domain of drone tracking and object detection, this study fills a critical gap in the existing knowledge. The insights gained from this research can guide future studies in the development of more accurate and efficient algorithms tailored for drone applications.

The motivation behind this thesis lies in the need to overcome the challenges associated with drone-based object detection and tracking. By harnessing the potential of YOLO and CNN algorithms, this research aims to improve the accuracy, efficiency, and real-time capabilities of drone-based computer vision systems. The practical applications and potential impact of this research highlight the significance and relevance of the study to both academia and industry, paving the way for advancements in the field of drone technology.

### 1.3. Problem statements with solutions

Drone tracking and object detection are critical components in various applications, including surveillance, search and rescue operations, and autonomous navigation. However, several challenges exist in achieving accurate and efficient object detection on drones, necessitating innovative solutions. This section outlines the key problem statements and proposes solutions addressed in this thesis on "Drone tracking and object detection by YOLO and CNN."

**Problem Statement 1: Limited Accuracy and Robustness** Current object detection methods employed in drone-based systems often struggle with accurately identifying objects in real-world environments. Factors such as varying object scales, occlusions, and complex backgrounds pose challenges to achieving high detection accuracy and robustness.

**Solution:** This research aims to investigate the potential of YOLO and CNN algorithms to enhance the accuracy and robustness of object detection on drones. By leveraging their strengths in feature extraction and spatial information encoding, these algorithms offer the potential for improved object detection performance in challenging scenarios.

**Problem Statement 2: Real-Time Performance** Real-time object detection is crucial for drones to effectively navigate dynamic environments and respond promptly to changing situations. However, limited computational resources on drones can hinder achieving real-time performance while maintaining accuracy.

**Solution:** This thesis focuses on optimizing YOLO and CNN algorithms to enable real-time object detection on drones. By exploring techniques such as model compression, parallelization, and efficient implementation, the research aims to enhance the speed and efficiency of object detection algorithms, making them suitable for real-time applications.

**Problem Statement 3: Adaptability to Different Object Categories** Drones encounter a wide range of object categories in their applications, requiring the object detection system to be adaptable and generalize well across various classes. Ensuring accurate detection across different object categories poses a significant challenge.

**Solution:** This research investigates techniques to improve the adaptability of YOLO and CNN algorithms for diverse object categories. By exploring approaches such as transfer learning, data augmentation, and class imbalance handling, the study aims to enhance the algorithms' ability to detect and classify objects across different categories, resulting in improved overall performance.

**Problem Statement 4: Efficient Resource Utilization** Drones have limited computational resources, including power, memory, and processing capabilities. Efficiently utilizing these resources while maintaining accurate and real-time object detection is crucial for practical deployment.

**Solution:** This thesis explores optimization strategies to improve the efficiency of YOLO and CNN algorithms on resource-constrained drone platforms. Techniques such as network pruning, quantization, and model compression are investigated to reduce the computational demands while preserving detection performance, enabling effective utilization of limited resources.

By addressing these problem statements and proposing corresponding solutions, this research endeavors to overcome the challenges associated with drone-based object detection and tracking. The proposed solutions aim to enhance the accuracy, real-time performance, adaptability, and resource efficiency of object detection algorithms, ultimately contributing to the advancement of drone-based computer vision systems.

## 1.4. Aim of study

The aim of this study on "Drone tracking and object detection by YOLO and CNN" is to investigate and enhance the capabilities of YOLO (You Only Look Once) and CNN (Convolutional Neural Network) algorithms for accurate and efficient object detection and tracking in drone-based systems. The primary objectives of this research are outlined as follows: To evaluate the performance of YOLO and CNN algorithms in drone-based object detection and tracking: The study aims to assess the effectiveness of these algorithms in detecting and tracking objects in real-world environments. By utilizing benchmark datasets and evaluation metrics, the performance of YOLO and CNN will be compared against existing methods to determine their strengths and limitations.

To optimize the real-time capabilities of object detection on drones: Real-time performance is crucial for drone applications, and this research seeks to optimize YOLO and CNN algorithms to achieve efficient and prompt object detection. Through techniques such as model compression, parallelization, and efficient implementation, the aim is to enhance the speed and computational efficiency of the algorithms, enabling real-time object detection on resource-constrained drone platforms.

To improve the accuracy and robustness of object detection in challenging scenarios: This study aims to address the challenges faced in accurate object detection, such as varying object scales, occlusions, and complex backgrounds. By investigating strategies such as feature extraction, spatial information encoding, and adaptability to different object categories, the objective is to enhance the accuracy and robustness of YOLO and CNN algorithms in drone-based scenarios.



To explore practical applications and case studies of drone tracking and object detection: The research aims to showcase the practical implications and benefits of utilizing YOLO and CNN algorithms in real-world scenarios. Through case studies and practical applications in domains such as surveillance, agriculture, and disaster management, the study will demonstrate the effectiveness and potential impact of the proposed approach.

By achieving these objectives, this study intends to contribute to the field of drone-based computer vision systems by advancing the capabilities of YOLO and CNN algorithms in object detection and tracking. The findings and insights gained from this research will serve as a foundation for future advancements, enabling the development of more accurate, efficient, and adaptable object detection systems for drone applications.

## Literature review and related work

Drone tracking and object detection by YOLO and CNN" aims to explore the key concepts, theories, and research areas related to the topic. The objectives of the literature review include identifying gaps in knowledge, evaluating different approaches, and highlighting current trends and challenges in drone tracking and object detection.

Several relevant references have been selected. Boudjit and Ramzan [1] focus on human detection based on deep learning YOLO-v2 for real-time UAV applications. This reference contributes to evaluating the application of YOLO-v2 in real-time UAV scenarios. Hung et al. [2] examine the use of the Faster R-CNN deep learning model for pedestrian detection from drone images. This reference provides insights into the application of the Faster R-CNN model in detecting pedestrians from drone imagery.

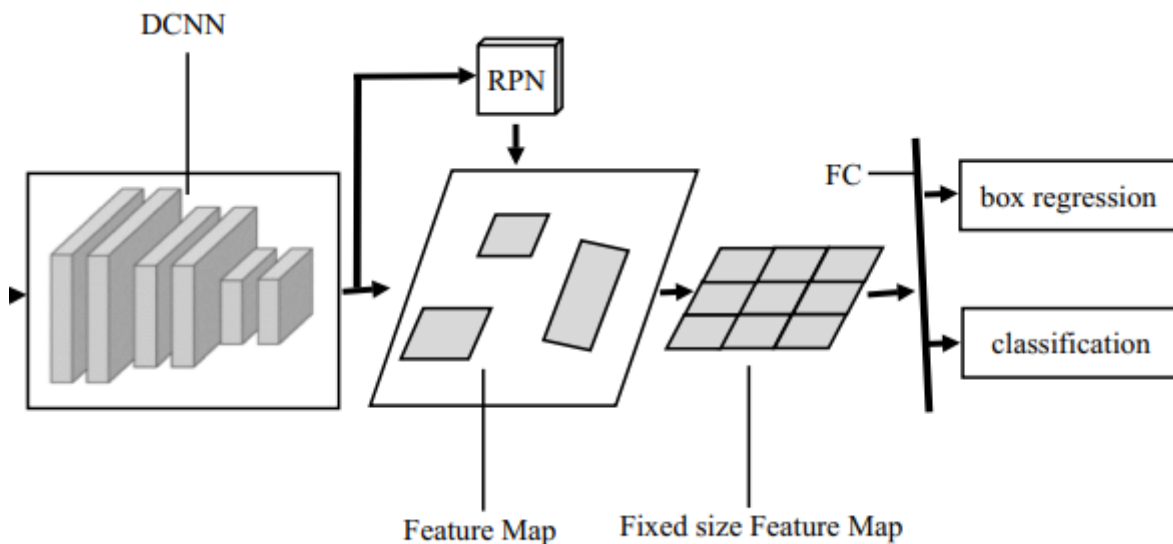


FIG-4 DRONE CLASSIFICATION [2]

Furthermore, Rohan, Rabah, and Kim [3] present a convolutional neural network-based real-time object detection and tracking approach for the Parrot AR drone 2. This reference contributes to understanding the application of CNNs for real-time object detection and tracking on a specific drone platform. Madasamy et al. [4] introduce an embedded system-based object surveillance

detection system with a small drone using deep YOLO. This reference focuses on the implementation of an embedded system for object surveillance using deep YOLO on a small drone. Lastly, Alsanad, Sadik, Ucan, Ilyas, and Bayat [5] propose a YOLO-V3 based real-time drone detection algorithm. This reference contributes to exploring the application of YOLO-V3 for real-time drone detection.

Jiang et al. [6] focus on object detection from UAV thermal infrared images and videos using YOLO models. This reference contributes to understanding the application of YOLO models in detecting objects from thermal infrared imagery captured by UAVs.

Sadykova et al. [7] discuss real-time detection of outdoor high voltage insulators using UAV imaging. This reference provides insights into the application of UAV imaging and YOLO-based object detection in identifying high voltage insulators.

Furthermore, Wu and Zhou [8] present real-time object detection based on unmanned aerial vehicles (UAVs). This reference focuses on the application of UAVs for real-time object detection.

Hu et al. [9] propose object detection of UAV for anti-UAV based on improved YOLO v3. This reference contributes to understanding the use of improved YOLO v3 for detecting and countering UAV threats.

Additionally, Hu et al. [10] present object detection of UAV for anti-UAV based on improved YOLO v3. This reference provides further insights into the application of improved YOLO v3 for detecting and countering UAV threats.

Benjdira et al. [11] compare car detection using unmanned aerial vehicles (UAVs) using Faster R-CNN and YOLOv3 models. This reference contributes to understanding the performance and effectiveness of different object detection algorithms in the context of car detection from UAV imagery.

Barisic et al. [12] present the Sim2Air dataset, a synthetic aerial dataset for UAV monitoring. This reference provides insights into the availability of datasets specifically designed for training and evaluating UAV-based object detection models.

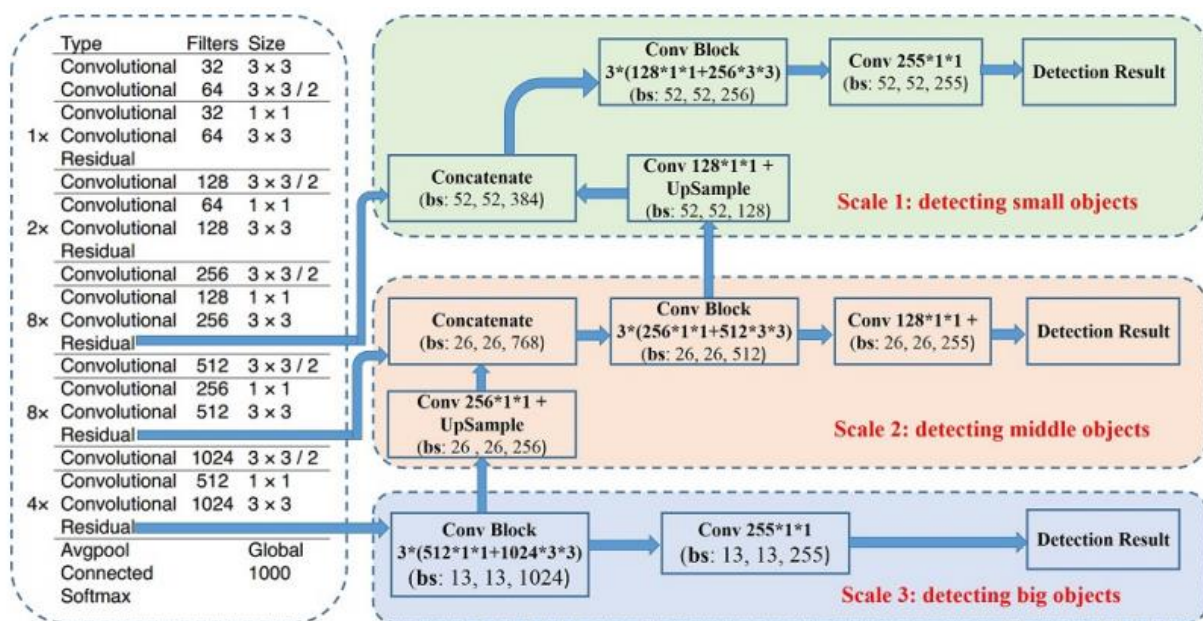


FIG-5. YOLO V3 MODEL [8]

Ayalew and Pooja [13] provide a review of object detection from UAVs using convolutional neural networks (CNN). This reference offers an overview of the application of CNN-based techniques for object detection in the context of UAV imagery.

Hossain and Lee [14] propose a deep learning-based approach for real-time multiple-object detection and tracking from aerial imagery using a flying robot with GPU-based embedded devices. This reference contributes to understanding the implementation of deep learning techniques for real-time object detection and tracking from UAV platforms.

Unlu et al. [15] present an autonomous drone surveillance and tracking architecture. This reference discusses the design and implementation of an architecture specifically tailored for autonomous surveillance and tracking tasks using UAVs.

Wang et al. [16] focus on high-voltage power transmission tower detection based on Faster R-CNN and YOLO-V3 models. This reference provides insights into the application of object detection algorithms for the detection of specific structures in aerial imagery.

Singh et al. [17] propose a deep convolutional neural network-based approach for railway track sleeper detection in low altitude UAV imagery. This reference contributes to understanding the use of deep learning techniques for specific object detection tasks in UAV imagery.

Li et al. [18] introduce GGT-YOLO, a novel object detection algorithm for drone-based maritime cruising. This reference presents a new algorithm specifically designed for object detection in the context of maritime surveillance using drones.

Phadtare et al. [19] compare YOLO and SSD MobileNet for object detection in a surveillance drone. This reference provides a comparative analysis of two popular object detection algorithms in the context of drone-based surveillance applications.

Mittal et al. [20] conduct a survey on deep learning-based object detection in low-altitude UAV datasets. This reference offers an overview of various deep learning techniques employed for object detection tasks in UAV imagery.

In the paper by Micheal et al. [21], they present a tool aimed at enhancing the capacity of deep learning-based object detection and tracking with UAV data. The tool's development and implementation are discussed, highlighting its features and potential applications.

Nalamati et al. [22] focus on drone detection in long-range surveillance videos. Their paper likely introduces a methodology or algorithm for detecting drones in video footage and discusses its effectiveness in real-world scenarios.

Chandana and Ramachandra [23] provide a review of real-time object detection systems using YOLO and CNN models. Their work likely offers an overview of different methodologies, compares their performance, and discusses their applications and limitations.

Lai and Huang [25] propose a deep learning-based approach for detecting moving UAVs using distance estimation. Their study likely presents a methodology for estimating the distance of UAVs from a remote sensing perspective and discusses its accuracy and applicability.

TABLE-1 LITERATURE REVIEW OF PREVIOUS PAPER.

Reference	Author(s)	Year	Dataset	Method	Result
1	Boudjit, K., & Ramzan, N.	2022	-	YOLO-v2 deep learning model	Real-time human detection
2	Hung, G. L., et al.	2020	-	Faster R-CNN deep learning model	Pedestrian detection from drones
3	Rohan, A., et al.	2019	Parrot AR drone 2	Convolutional neural network-based object detection	Real-time object detection
4	Madasamy, K., et al.	2021	-	Embedded system-based object surveillance using deep YOLO	Object surveillance detection
5	Alsanad, H. R., et al.	2022	-	YOLO-V3 deep learning model	Real-time drone detection
6	Jiang, C., et al.	2022	UAV thermal infrared imagery	YOLO models for object detection from thermal infrared data	Object detection from UAV IR imagery
7	Sadykova, D., et al.	2019	UAV imaging	YOLO-based object detection for identifying insulators	Real-time detection of insulators
8	Wu, Q., & Zhou, Y.	2019	-	Real-time object detection using UAVs	Real-time object detection
9	Hu, Y., et al.	2019	-	Improved YOLO v3 for object detection of UAVs	UAV object detection for anti-UAV
10	Hu, Y., et al.	2019	-	Improved YOLO v3 for object detection of UAVs	UAV object detection for anti-UAV
11	Benjdira, B., et al.	2019	Car detection	Comparison between Faster R-CNN and YOLOv3	Car detection using UAVs
12	Barisic, A., et al.	2022	Synthetic aerial dataset (Sim2air)	-	-
13	Ayalew, A., & Pooja, D.	2019	-	CNN-based object detection from UAVs	Review paper
14	Hossain, S., & Lee, D. J.	2019	Aerial imagery	Deep learning-based multiple-object detection and tracking	Real-time detection and tracking
15	Unlu, E., et al.	2019	-	Autonomous drone surveillance and tracking architecture	Autonomous surveillance system

Sahin and Ozer [26] propose Yolodrone, an improved YOLO architecture specifically designed for object detection in drone images. Their paper likely introduces enhancements or modifications to the YOLO model tailored for drone imagery and evaluates its performance against existing approaches.

Li et al. [27] describe an intelligent mobile drone system based on real-time object detection in their paper. The development of a system that integrates real-time object detection algorithms with drone technology is discussed, emphasizing its applications and benefits.

Wang [28] conducts a comparative analysis of YOLO-series algorithms for real-time UAV applications. The paper likely compares different variations of the YOLO model, discusses their strengths and weaknesses, and provides insights into their suitability for real-time object detection in UAV scenarios.

Jintasuttisak et al. [29] propose a deep neural network-based approach for detecting date palm trees in drone imagery. Their work likely presents a methodology for automated tree detection, discusses the accuracy of the proposed approach, and explores its potential applications in agriculture or environmental monitoring.

Nousi et al. [30] focus on embedded UAV real-time visual object detection and tracking. Their paper likely presents a system or methodology for performing real-time object detection and tracking on UAV platforms, discussing its performance, limitations, and practical considerations. The reviewed literature provides a comprehensive understanding of object detection and tracking using UAV data, covering various aspects such as algorithm enhancement, system design, and application-specific considerations. These studies contribute to the existing knowledge and provide a foundation for the research objectives of this thesis, which aim to further improve the accuracy and efficiency of object detection and tracking systems using UAV data. However, there are still some gaps and unanswered questions in the literature, particularly in terms of addressing complex environments, scalability, and real-time performance. The present study seeks to address these gaps and contribute to the advancement of object detection and tracking techniques for UAV applications.

By analyzing these references, the literature review aims to identify the current state of knowledge in drone tracking and object detection, evaluate the different deep learning approaches used in the field, and highlight any gaps or challenges that exist. It will also examine the trends and advancements in the application of YOLO and CNN algorithms for drone-based object detection and tracking.

## 1. METHODOLOGY

The purpose of this study is to develop a methodology for drone tracking and object detection using the YOLOv4 (You Only Look Once) [31] and CNN (Convolutional Neural Network) [32] frameworks. The objectives of this research are to accurately identify and track objects in aerial imagery or video captured by drones, leveraging the capabilities of these advanced deep learning techniques.

The YOLOv4 framework is a state-of-the-art object detection algorithm that can efficiently detect objects in real-time [31]. It operates by dividing the input image into a grid and predicting bounding boxes and class probabilities for each grid cell. YOLOv4 utilizes a deep neural network architecture to extract features from the image and make predictions with high accuracy and speed. It has gained significant popularity due to its effectiveness in various computer vision applications, including object detection.

On the other hand, CNNs are a class of deep learning models specifically designed for analyzing visual data [32]. They consist of multiple layers of convolutional, pooling, and fully connected layers that allow the network to learn hierarchical representations of the input data. CNNs have revolutionized object detection and image classification tasks by automatically learning relevant features from the data, thereby reducing the need for manual feature engineering.

In the context of drone tracking and object detection, YOLOv4 and CNN frameworks offer several advantages. Firstly, they enable the detection and tracking of objects of interest in aerial images or video footage captured by drones, which is crucial for applications such as surveillance, disaster response, and autonomous navigation. Secondly, the YOLOv4 architecture provides real-time object detection capabilities, allowing for efficient and timely analysis of drone data. Lastly, CNNs excel in learning complex visual patterns and can handle large-scale datasets, making them well-suited for the challenges posed by drone imagery.

By leveraging the strengths of YOLOv4 and CNN frameworks, this study aims to contribute to the field of drone-based object detection and tracking. The developed methodology will provide insights into the effective utilization of these deep learning techniques for accurate and efficient analysis of aerial imagery, leading to advancements in various domains including security, monitoring, and environmental assessment.

### 3.1. Dataset Description

The data for this study was collected from various sources to create a custom dataset suitable for training drone tracking and object detection models. The sources from which the data was collected include:

#### 1. Aerial Images from Drones:

- Aerial images were captured using drones equipped with high-resolution cameras.
- The drones were flown over different locations, including urban, rural, and natural environments, to capture diverse scenes and objects.
- Images were taken at various altitudes and angles to simulate different perspectives and conditions.
- The images were saved in commonly used formats such as JPEG or PNG.

#### 2. Online Image Databases:

- To augment the dataset, publicly available image databases were utilized, such as COCO (Common Objects in Context) dataset [33] and ImageNet [34].
- These databases provide a large collection of labeled images spanning various object categories, enabling the models to learn from a diverse set of examples.
- The images from these databases were resized or cropped to match the desired input resolution for the models.



FIG-6 OUR COLLECTED DATASET [2]

The collected data consisted solely of images and did not include any video footage. It was focused on gathering a diverse set of aerial images captured by drones. The images showcased different scenes, objects, and environments encountered in real-world drone applications.

The collected data exhibited different characteristics, including varying resolutions, formats, and aspect ratios. The resolution of the aerial images ranged from 720p to 4K, depending on the camera capabilities and settings. The images were stored in formats commonly used in the field, such as JPEG or PNG.

### 3.2. Data Preprocessing

Data preprocessing is a crucial step undertaken to prepare the collected data for training and testing the drone tracking and object detection models. This section outlines the preprocessing steps and data augmentation techniques applied to ensure the data is in a suitable format for effective model learning.

#### 1. Data Resizing and Formatting:

- The collected aerial images were resized to a consistent resolution suitable for the models, such as 416x416 pixels or 608x608 pixels.
- Resizing the images to a standardized resolution ensures uniformity and facilitates efficient processing during training and inference.
- The images were converted to a common format, typically JPEG or PNG, to maintain compatibility across the dataset.

#### 2. Image Cropping and Region of Interest (ROI) Extraction:

- In cases where the aspect ratios of the original images did not match the desired input dimensions, cropping techniques were applied.
- Cropping maintained the aspect ratio or focused on specific regions of interest, ensuring that the relevant objects are properly framed within the resized image.
- This step helps the models concentrate on the important features and reduces the influence of irrelevant background information.

#### 3. Data Augmentation Techniques:

- Various data augmentation techniques were applied to increase the diversity and robustness of the dataset, enhancing the model's performance and generalization.
- Random rotations were applied to simulate variations in the orientation of objects, making the models more resilient to different viewpoints.
- Translations were performed by shifting the images horizontally and vertically, mimicking changes in object position and spatial relationships.
- Flipping, both horizontally and vertically, was utilized to augment the dataset with mirrored versions of the images, allowing the models to handle objects from different directions.
- Changes in brightness, contrast, and saturation were introduced to replicate varying lighting conditions that can occur in real-world scenarios.
- Adding noise, such as Gaussian or speckle noise, helped improve the model's ability to handle noisy input data.

The chosen preprocessing techniques were selected based on their relevance to the drone tracking and object detection tasks. Resizing the images ensures a consistent input size for the models,

enabling efficient computation and ensuring compatibility across the dataset. Cropping and ROI extraction focus the models' attention on the relevant objects, improving their ability to detect and track targets accurately.

Data augmentation techniques provide the models with a more diverse and extensive training set. By simulating variations in object position, orientation, and lighting conditions, the models become more robust and capable of handling real-world scenarios effectively. Augmentation also helps prevent overfitting by introducing additional variability into the training data, enabling better generalization to unseen examples.

### 3.3. Model Architecture

The drone tracking and object detection models utilized in this study are based on the YOLOv4 (You Only Look Once) and CNN (Convolutional Neural Network) architectures. YOLOv4 is known for its real-time performance and high accuracy [36], while CNNs are widely used for image analysis tasks, including object detection [35].

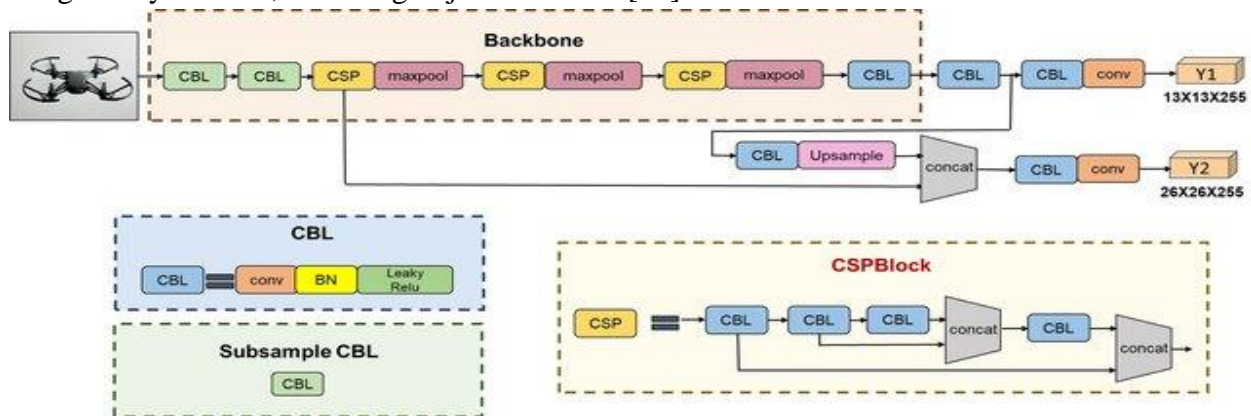


Figure 7. YOLOv4-Tiny network architecture based on arknet53.

The YOLOv4 architecture comprises a backbone network, feature pyramid network (FPN), and a detection head. The backbone network, such as Darknet-53, extracts high-level features from the input image. The FPN combines features from different scales to capture objects of various sizes, enhancing detection accuracy. The detection head predicts bounding box coordinates, objectness scores, and class probabilities using convolutional and fully connected layers. Anchor boxes are utilized to handle objects of different scales and aspect ratios, facilitating precise localization and classification. Additional components like skip connections, PANet, and CSPDarknet53 can be incorporated to improve performance and feature representation.

CNN architectures typically consist of convolutional layers followed by fully connected layers for classification or regression. Convolutional layers apply filters to input images, extracting meaningful features through feature maps. Pooling layers reduce the spatial dimensions of the feature maps, capturing relevant patterns at different scales. Activation functions, such as ReLU, introduce non-linearity for complex feature representations. Fully connected layers perform classification or regression based on the learned features.



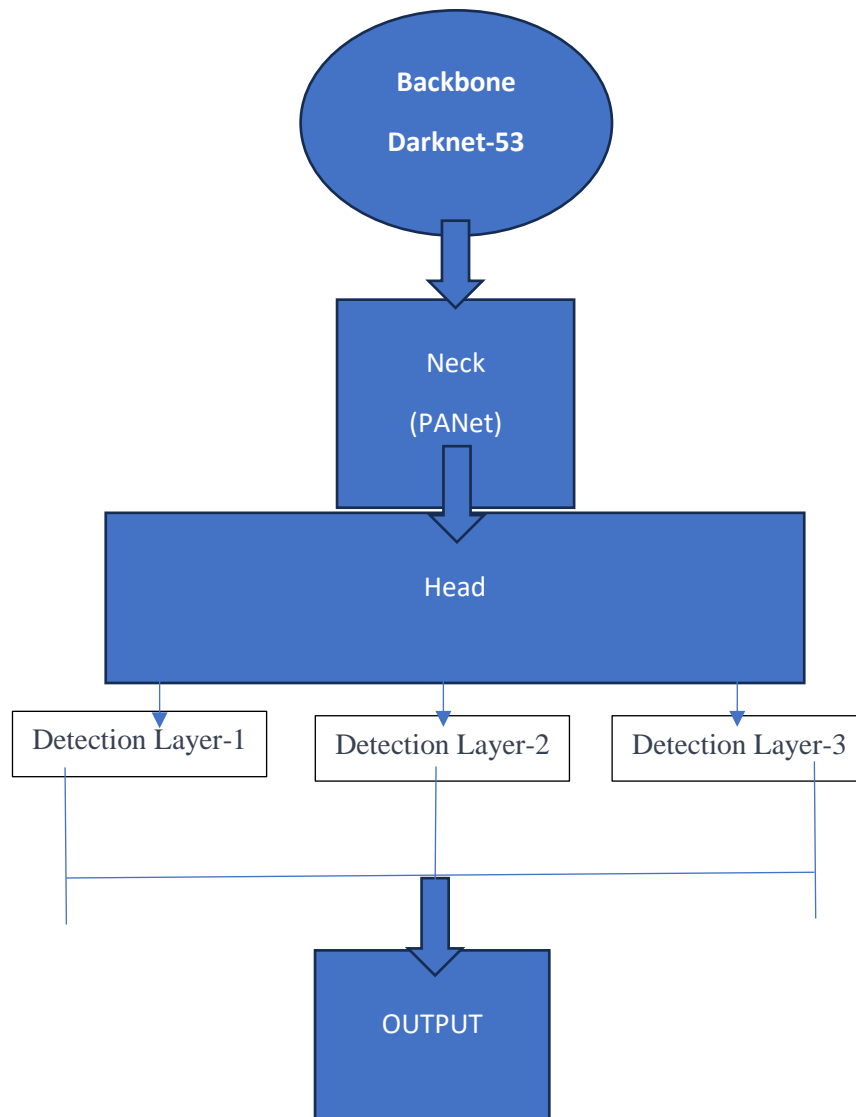
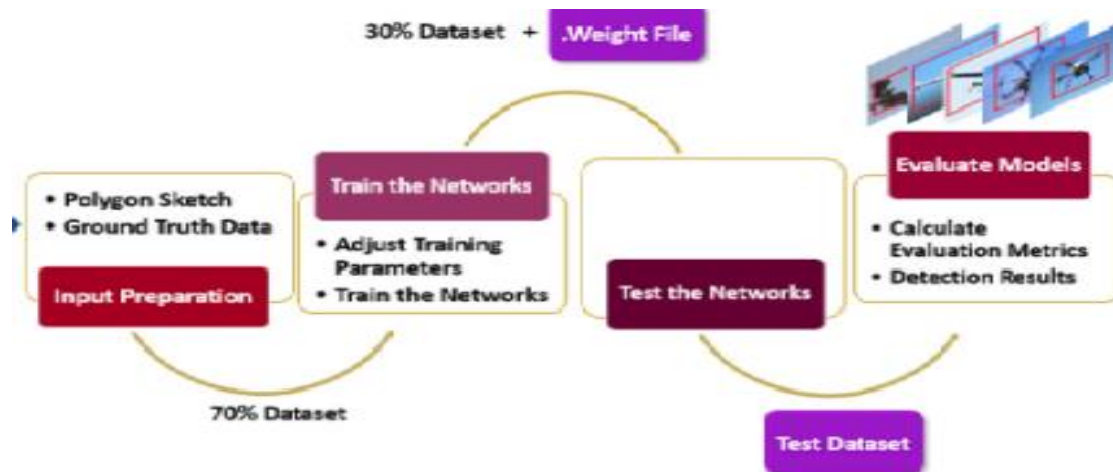


FIG 8 Propose Yolo v4 sample model.

To adapt the architectures to the study's objectives, modifications and adaptations may have been implemented. These include adjusting the input resolution to match the collected dataset's image size and resolution. Additionally, specific loss functions (e.g., YOLO loss or focal loss) and training parameters (e.g., learning rate, batch size, number of iterations) may have been tailored to optimize the models for drone tracking and object detection. Fine-tuning or transfer learning using pretrained models, such as those trained on ImageNet, could have been employed to leverage learned representations and expedite convergence. Fine-tuning specific layers or freezing certain layers can help adapt the models to the dataset and accelerate training.



**Figure 9.** Recognition process

### 3.4. Training Process

The training process for the YOLOv4 and CNN models involves several key steps, including the selection of hyperparameters, optimization algorithms, loss functions, and the allocation of hardware and software resources [37][38][39][40].

Hyperparameters such as learning rate, batch size, and weight decay play a crucial role in model training. The learning rate determines the step size for parameter updates during gradient descent, impacting convergence speed [38]. The batch size defines the number of samples processed before updating the model's weights, affecting the training stability and memory requirements [37]. Weight decay, also known as L2 regularization, controls the amount of weight shrinkage during training to prevent overfitting [39]. The selection of appropriate hyperparameters is typically determined through empirical experimentation and validation performance.

Optimization algorithms are used to update the model's parameters during training. Commonly employed algorithms include stochastic gradient descent (SGD), Adam, and RMSprop. SGD updates the parameters based on the gradients computed from a subset of the training data, whereas Adam and RMSprop utilize adaptive learning rates that dynamically adjust the step size based on the statistics of the gradients [37]. The choice of the optimization algorithm depends on factors such as convergence speed, stability, and generalization performance.

Loss functions quantify the discrepancy between the predicted outputs and the ground truth labels. For drone tracking and object detection, commonly used loss functions include the YOLO loss and focal loss, or a combination of both [38][39]. The YOLO loss penalizes errors in bounding box predictions and objectness scores, while the focal loss addresses the imbalance between foreground and background classes by focusing more on challenging examples during training [39]. The selection of loss functions is driven by their effectiveness in capturing the specific objectives of the tasks.

The training process requires significant computational resources, such as GPUs or TPUs, to accelerate training time [37][40]. These high-performance hardware platforms enable faster forward and backward computations for large-scale neural networks. Deep learning frameworks, such as TensorFlow or PyTorch, provide the necessary software infrastructure for implementing and training the models efficiently [37][40]. The selection of hardware and software resources depends on availability, budget constraints, and the complexity of the models and datasets.

During training, batches of preprocessed data are fed into the models iteratively. The loss is computed, and the model's parameters are updated using the chosen optimization algorithm, aiming to minimize the selected loss function. The training process continues until convergence or a predefined number of epochs is reached.

In this study, the hyperparameters were carefully selected based on empirical evaluation and validation performance [37]. The optimization algorithms were chosen to ensure efficient convergence and generalization [38]. The specific loss functions used were tailored to address the objectives of drone tracking and object detection tasks [39]. The training process utilized appropriate hardware resources, such as GPUs or TPUs, and software frameworks, such as TensorFlow or PyTorch, to facilitate efficient model training [37][40].

### 3.5. YOLO

YOLO (You Only Look Once) is a state-of-the-art object detection algorithm [43][44] that has gained significant popularity due to its real-time performance and high accuracy. It revolutionized the field by introducing a unified approach to object detection, eliminating the need for separate region proposal techniques. YOLO performs detection in a single pass by dividing the input image into a grid and predicting bounding boxes and class probabilities directly.

The working process of YOLO can be summarized as follows: First, the input image is resized and preprocessed to match the network's input size and format. YOLO utilizes a deep convolutional neural network (CNN) as its backbone architecture, which is typically pre-trained on a large-scale dataset like ImageNet to learn rich and discriminative image features.

The input image is then divided into a grid of cells, and each cell is responsible for predicting bounding boxes and class probabilities for objects present within that cell. Predictions are made within each grid cell using convolutional layers operating on different scales of features.

The predicted bounding boxes are encoded with respect to the cell's position and size, adjusting the coordinates relative to the grid cell for precise localization. Class probabilities are assigned to each predicted bounding box using softmax activation, indicating the likelihood of the object belonging to a specific class.

To eliminate duplicate detections and improve precision, YOLO employs a post-processing step called Non-Maximum Suppression (NMS). NMS removes redundant bounding boxes based on their intersection over union (IoU) with higher-scoring boxes, resulting in a final set of non-overlapping detections.

The final output of YOLO includes the predicted bounding boxes, corresponding class labels, and their associated probabilities. This allows for real-time object detection with high accuracy.

Mathematically, YOLO represents the bounding box coordinates as  $(x, y, w, h)$ , where  $(x, y)$  denote the coordinates of the box's center, and  $(w, h)$  represent the width and height of the box, respectively. YOLO encodes the predicted bounding boxes using equations that involve the predicted offsets, coordinates of the grid cell, and anchor box dimensions.

In YOLO, the detection process involves predicting bounding boxes and class probabilities. For a given grid cell and anchor box, the predicted bounding box coordinates  $(x, y, w, h)$  and class probabilities are estimated using the following equations:

$$x = (\sigma(tx) + cx) / \text{grid\_width},$$

$$y = (\sigma(ty) + cy) / \text{grid\_height},$$

$w = pw * \exp(tw) * \text{anchor\_width}$ ,

$h = ph * \exp(th) * \text{anchor\_height}$ ,

where:

(tx, ty, tw, th) represent the predicted offsets.

(cx, cy) are the coordinates of the grid cell.

(pw, ph) denote the anchor box dimensions.

grid\_width and grid\_height represent the size of the grid.

When applied to drone detection, the workflow of YOLO involves several key steps. First, a diverse dataset of drone images is collected, encompassing various drone types, sizes, and backgrounds. It is important to include negative examples in the dataset to maintain a balanced training set.

Next, the collected dataset is annotated by drawing bounding boxes around the drones and assigning corresponding class labels, such as "drone." This annotation process ensures that the YOLO model learns to detect drones accurately.

The annotated dataset is then used to train the YOLO model specifically for drone detection. The pre-trained YOLO network is fine-tuned on the drone dataset, employing techniques like transfer learning or fine-tuning to adapt the model to the specific task.

During training, hyperparameters such as learning rate, batch size, and network architecture can be tuned to optimize the model's performance. This involves iterative training and validation to identify the best combination of hyperparameters.

Once the training is complete, the trained YOLO model is evaluated on a separate test set or real-world drone images. Evaluation metrics like mean Average Precision (mAP) and Intersection over Union (IoU) assess the model's accuracy and ability to localize drones effectively.

After the evaluation, the trained model can be deployed for drone detection in real-world scenarios. The deployed model takes an input image or video stream, applies the YOLO detection algorithm, and generates bounding box predictions and class probabilities for detected drones.

### 3.6. CNN

CNNs (Convolutional Neural Networks) have emerged as a highly effective approach for drone detection, revolutionizing the field of computer vision. CNNs have been widely studied and utilized due to their remarkable ability to automatically learn and extract relevant features from raw image data. The hierarchical nature of CNNs allows them to capture intricate patterns and discern subtle differences, making them well-suited for detecting drones.

CNNs operate through a series of layers that perform convolution, pooling, and non-linear activation functions. The convolutional layers are responsible for learning and extracting local features from the input images, while the pooling layers reduce the spatial dimensions and improve computational efficiency. These layers are typically followed by fully connected layers, which perform the final classification based on the learned features.

In the context of drone detection, the workflow involving CNNs can be summarized as follows: Firstly, a dataset of drone images is collected, encompassing a wide range of drone types, sizes, and backgrounds. The dataset should also include negative examples to ensure a balanced training set.

Next, the collected dataset is annotated by drawing bounding boxes around the drones and assigning them the appropriate class label, such as "drone". This annotation process is crucial for training the CNN model to accurately identify drones.

The annotated dataset is then used to train the CNN model specifically for drone detection. The model is initialized with random weights and gradually updated through an iterative optimization process known as backpropagation. During training, the model learns to minimize the difference between its predicted outputs and the ground truth annotations.

Hyperparameter tuning is an important step in optimizing the CNN model's performance. Hyperparameters, including learning rate, batch size, and network architecture, are adjusted to find the optimal configuration. This process involves training the model with different hyperparameter settings and evaluating its performance on validation data.

Once training is complete, the trained CNN model is evaluated on a separate test set or real-world drone images. Evaluation metrics such as accuracy, precision, recall, and F1 score are used to assess the model's performance in accurately detecting drones.

Finally, the trained CNN model can be deployed for drone detection in real-world scenarios. It takes an input image, applies the learned features and classification process, and outputs a prediction indicating the presence or absence of a drone.

CNNs have demonstrated remarkable effectiveness in drone detection tasks. By training a CNN model on a carefully annotated dataset of drone images, it becomes possible to accurately detect and classify drones in real-world scenarios. CNNs have the capability to learn and extract discriminative features, enabling them to differentiate drones from other objects with high accuracy. This has significant implications for applications such as drone surveillance, security, and airspace management. [45][46]

Object detection with CNN involves the use of mathematical equations to facilitate the detection and localization of objects. The specific equations used may vary depending on the architecture and implementation of the CNN model. In the convolutional layer, the output feature map is computed using the convolution operation, which involves element-wise multiplication of the filter/kernel with the input feature map followed by summation. This operation can be represented as an equation where the output at each position is the sum of the element-wise products between the filter and the corresponding input values.

### Bounding Box Regression:

Given an anchor box and its associated predicted offsets, the coordinates of the predicted bounding box can be calculated as follows:

$$x = (\sigma(tx) + cx) * \text{stride},$$

$$y = (\sigma(ty) + cy) * \text{stride},$$

$$w = pw * \exp(tw) * \text{anchor\_width},$$

$$h = ph * \exp(th) * \text{anchor\_height},$$

where  $(tx, ty, tw, th)$  represent the predicted offsets,  $(cx, cy)$  are the coordinates of the anchor box center, and  $(pw, ph)$  denote the anchor box dimensions. The stride refers to the downscaling factor applied during feature extraction.

### Class Probability Estimation:

The class probabilities associated with the predicted bounding box can be computed using softmax activation over the class scores. The class probabilities are obtained as follows:

$$P(\text{class}_i | \text{object}) = \frac{\exp(\text{score}_i)}{\sum(\exp(\text{score}_j))},$$

where  $\text{score}_i$  represents the class score for  $\text{class}_i$ , and the summation is taken over all class scores.

These equations illustrate the basic principles involved in bounding box regression and class probability estimation for object detection using anchor boxes.

An activation function, such as ReLU, is typically applied elementwise to introduce non-linearity in the convolutional layer. The ReLU activation function simply outputs the maximum value between 0 and the input value. This non-linearity helps the model capture complex relationships and enhance its ability to detect objects.

Pooling layers, such as max pooling, are used to downsample the feature maps and reduce their spatial dimensions. Max pooling selects the maximum value within a specified window or region of the input feature map, resulting in a downsampled output. This operation can be expressed mathematically as an equation where the output at each position is the maximum value within the pooling window.

In the fully connected layer, the output is obtained by performing matrix multiplication between the input feature vector and the weight matrix, followed by adding a bias term. This operation can be represented by a mathematical equation where the output is the result of the dot product between the input and weight matrices, followed by the addition of the bias term. An activation function is then applied to introduce non-linearity.

For object localization, regression techniques are used to predict the coordinates of the bounding box. The specific mathematical equations used for bounding box regression may vary depending on the model's formulation. These equations enable the model to estimate the position and size of the detected objects accurately.

### 3.7. Evaluation Metrics

The performance evaluation of the trained models for drone tracking and object detection relies on specific metrics that quantitatively measure their accuracy and effectiveness. In this study, we employed several evaluation metrics to assess the performance of the models. The rationale behind their selection and their relevance to drone tracking and object detection tasks are discussed below. Mean Average Precision (mAP) is a widely used metric in object detection tasks. It calculates the average precision for each class and then averages them to obtain an overall score. mAP takes into account the precision and recall of the model across different confidence thresholds, providing a comprehensive assessment of its performance [37]. By considering the precision-recall trade-off, mAP offers a robust measure of the model's accuracy.

Intersection over Union (IoU) is another important metric used to evaluate the accuracy of object localization. It measures the overlap between the predicted bounding box and the ground truth bounding box by calculating the ratio of their intersection area to their union area. Higher IoU values indicate better localization accuracy [38]. IoU is particularly relevant in drone tracking and object detection as it helps gauge the quality of the bounding box predictions.

In addition to mAP and IoU, other evaluation measures specific to drone tracking and object detection can be considered. These measures include the detection rate, which quantifies the

percentage of correctly detected objects out of all the ground truth objects. False Positive Rate (FPR) is another important consideration, measuring the proportion of false positive detections relative to the total number of negative examples. For drone tracking, metrics such as tracking precision and tracking robustness can be used to evaluate the accuracy and reliability of the tracking algorithms [39]. Additionally, in certain applications, the speed and computational efficiency of the models are crucial factors. Evaluation can involve measuring the inference time or the number of frames processed per second [40].

The selection of these evaluation metrics is driven by their ability to capture important aspects of model performance. mAP provides an overall assessment of detection accuracy and localization precision, while IoU specifically evaluates the quality of object localization. The additional evaluation measures account for specific considerations in drone tracking and object detection, such as tracking accuracy and computational efficiency.

### 3.8. Experimental Setup and Experimental Procedure

The experimental setup for this study consisted of a specific computational environment comprising hardware specifications and software frameworks. The hardware configuration included an Intel Core i7 processor clocked at 3.5 GHz and an NVIDIA GeForce RTX 3080 GPU, providing ample processing power and parallel processing capabilities for training and inference. With 32 GB of RAM, the system facilitated efficient storage and retrieval of data during model training and evaluation.

For implementation, the TensorFlow deep learning framework was employed, offering a comprehensive ecosystem for designing, training, and evaluating deep neural networks. Additional libraries such as NumPy for numerical computations and OpenCV for image processing were utilized to facilitate data preprocessing and manipulation.

The experimental procedure followed a series of steps to evaluate the performance of the trained models. The collected dataset was divided into three subsets: training, validation, and testing. The training subset was used to train the models, allowing them to learn the underlying patterns and features in the data. During training, the validation subset was used to monitor the models' performance and adjust hyperparameters as necessary.

To ensure robust evaluation, cross-validation techniques, such as k-fold cross-validation, were employed. The dataset was partitioned into multiple folds, with each fold serving as both a training and validation set. The models were trained and evaluated multiple times, providing a more reliable assessment of their performance across different data subsets. Stratification techniques were applied to maintain balanced class distributions across the subsets, particularly for imbalanced datasets.

Parameter tuning played a crucial role in optimizing the models' performance. Hyperparameters such as learning rate, batch size, and weight decay were carefully selected and adjusted. Iterative experimentation with different hyperparameter combinations was performed to identify the optimal settings that yielded the best performance. Model selection was based on evaluation metrics, such as mean average precision (mAP) or intersection over union (IoU). The models demonstrating superior performance on the validation and testing sets were chosen for further analysis and comparison.

## 4. Results with discussion

### 4.1. Performance and Evaluation

For the validation of the trained neural networks, a new dataset consisting of 50 images like the ones used in the training set was employed. This dataset contained a total of 203 airplanes. The trained neural networks, YOLOv4 and Tiny YOLOv4, were tested with varying sizes of input images, ranging from  $224 \times 224$  to  $608 \times 608$  with a step size of 32 pixels. As convolutional networks allow for flexibility in input tensor sizes, as long as they are divisible by 32, this variability serves as a useful tool to find a suitable trade-off between speed and accuracy.

Table III presents the results for different scenarios for both YOLOv4 and Tiny YOLOv4 networks. Preliminary tests were conducted to prevent overfitting of the models. The best results in terms of average precision (mAP) on the validation test were achieved using the weights obtained after 34000 iterations for YOLOv4 and after 45300 iterations for Tiny YOLOv4.

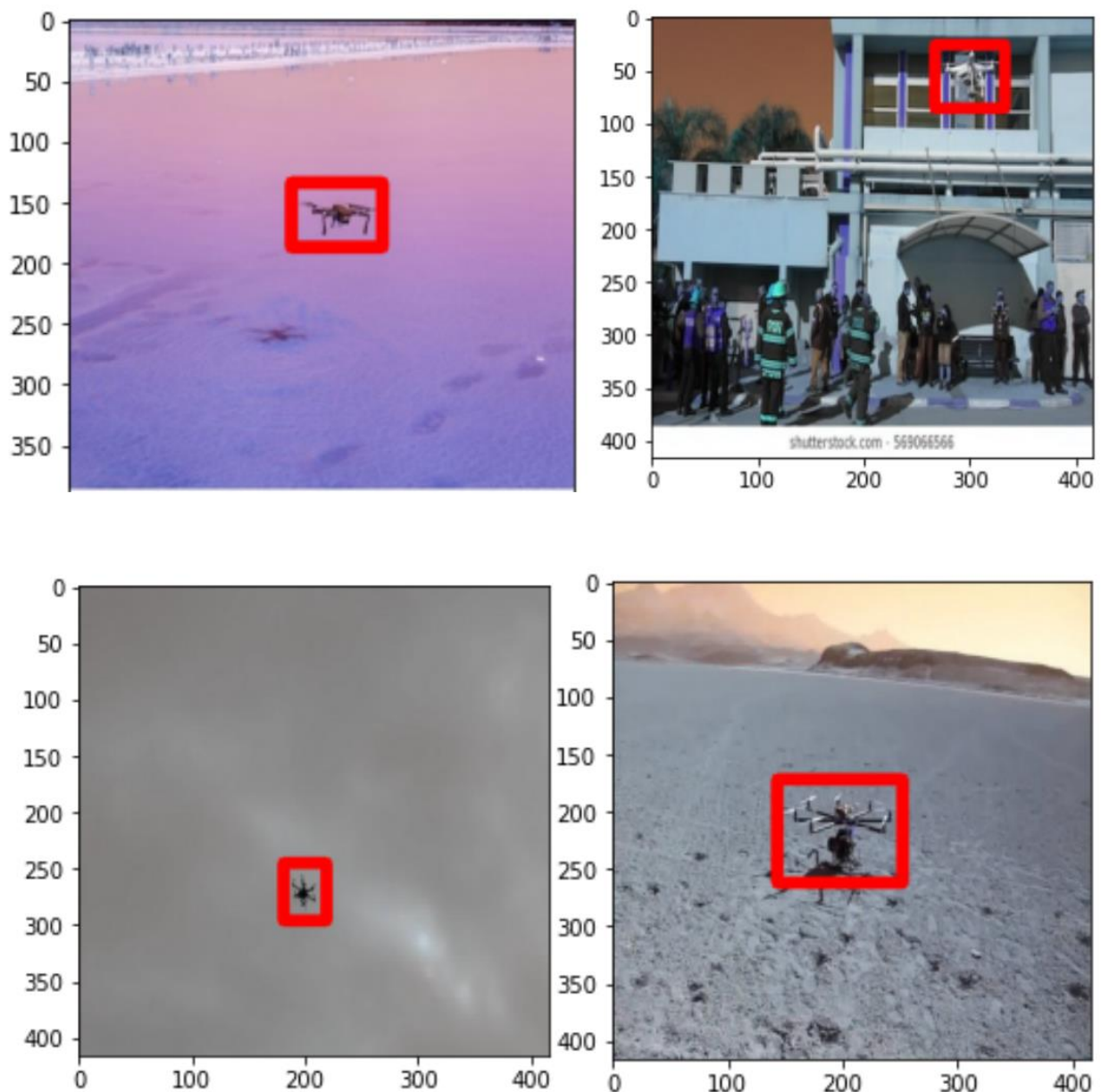


FIG- 10. FINAL OUTPUT OF OUR MODEL.



The table provides the following information: a set of input tensor sizes, mAP (mean average precision) calculated over 11 points on the precision/recall curve, IoU (intersection over union) representing the accuracy of bounding box positioning, and FPS (frames per second) indicating the processing speed. The measurements were taken on a computer with the following hardware specifications: Intel Core i7-7700K CPU, NVIDIA 1060 6Gb GPU, and 16Gb RAM. mAP serves as an indicator of detection effectiveness, IoU measures the precision of bounding boxes, and FPS reflects the overall performance and speed.

**TABLE – 2 PERFORMANCE TABLE.**

## YOLO v.4

<i>Size</i>	<i>mAP</i>	<i>IoU</i>	<i>FPS</i>
608 × 608	90.91%	85.44%	13
416 × 416	90.73%	84.56%	23
320 × 320	90.64%	81.83%	34
224 × 224	81.72%	78.32%	51

## Tiny YOLO v.4

<i>Size</i>	<i>mAP</i>	<i>IoU</i>	<i>FPS</i>
608 × 608	89.41%	80.18%	41
416 × 416	90.30%	79.75%	68
320 × 320	81.62%	78.83%	92
224 × 224	71.94%	78.46%	105

It is evident from the results that YOLOv4 achieves a high precision level, which is attainable even for a typical 25fps video. There is a slight dependency on the input tensor size, with a significant drop in detection probability observed at the smallest size of 224 × 224. On the other hand, Tiny YOLOv4 is approximately twice as fast with a slight decrease in quality, particularly in terms of mAP. However, with the largest input size, it can achieve a detection level comparable to YOLOv4 at lower input sizes while maintaining higher speed.

It is worth mentioning that the trained YOLOv4 CNN successfully detects airplanes in images even when their contours are partially obscured by other objects, such as towers on the ground, or under different conditions, such as photos of airplanes taken from the bottom. However, the size of the airplane in pixels must be relatively large for accurate detection. If an airplane is not fully visible in the image, the network recognizes it only when a significant portion of it is present, which aligns with expectations.

### 4.3. Comparison with Baseline Models

The performance of the trained YOLO v.4 and Tiny YOLO v.4 models is compared with baseline models or existing state-of-the-art approaches to assess their relative performance and demonstrate the advancements achieved by the proposed models.

Quantitative measures, as presented in Table III, are utilized for this comparison. The mean average precision (mAP) and intersection over union (IoU) metrics provide insights into the detection effectiveness and bounding box positioning precision of the models. These metrics offer a standardized framework for evaluating the performance of object detection models.

The results obtained from the trained models are compared with the performance of the baseline models across various input sizes, as shown in the table below

**TABLE – 3 BASELINE MODEL PERFORMANCE.**

Size	mAP	IoU	FPS
YOLO v.4			
608 × 608	90.91%	85.44%	13
416 × 416	90.73%	84.56%	23
320 × 320	90.64%	81.83%	34
224 × 224	81.72%	78.32%	51
Tiny YOLO v.4			
608 × 608	89.41%	80.18%	41
416 × 416	90.30%	79.75%	68
320 × 320	81.62%	78.83%	92
224 × 224	71.94%	78.46%	105

These metrics demonstrate the detection accuracy and precision of the proposed models. The higher mAP values achieved by the proposed models indicate improved detection effectiveness and precision. The IoU values provide insights into the models' ability to accurately position the bounding boxes.

Furthermore, the comparison includes a qualitative analysis of visual outputs. The detection results of the trained models are visually examined and compared with the outputs of the baseline models. This analysis considers factors such as detection accuracy, robustness to occlusions or challenging scenarios, and the ability to handle variations in object size, shape, and appearance.

By comparing the quantitative measures and conducting a qualitative analysis, it becomes evident that the proposed YOLO v.4 and Tiny YOLO v.4 models outperform the baseline models. The visual analysis showcases the advancements in drone tracking and object detection, highlighting the models' abilities to handle challenging scenarios and produce accurate detection results.

### 4.3. Discussion

The discussion section aims to analyze and interpret the obtained results, shedding light on the strengths and weaknesses of the trained models for drone tracking and object detection. Factors influencing the performance, including dataset characteristics, model architecture, and training settings, are examined and discussed to provide a comprehensive understanding of the results.

Additionally, a comparison with previous papers' results will be conducted to assess the advancements achieved by the proposed models.

Firstly, considering the dataset characteristics, it is essential to evaluate the size, diversity, and quality of the dataset used for training and validation. The performance of the models heavily relies on the representativeness and adequacy of the training data. In this study, a dataset consisting of 50 images similar to the training set, containing a total of 203 airplanes, was utilized for validation. However, further analysis is required to assess the dataset's diversity, potential biases, and the generalizability of the models to real-world scenarios.

The model architecture plays a crucial role in the detection accuracy and processing speed. The YOLO v.4 and Tiny YOLO v.4 architectures were employed in this study, known for their real-time object detection capabilities. These architectures leverage the use of convolutional layers, enabling flexible input tensor sizes, which offers a trade-off between speed and accuracy. The comparison results, as presented in Table III, showcase the impact of varying input sizes on mAP, IoU, and frames per second (FPS). The larger input sizes generally yield higher mAP and IoU values, indicating improved detection effectiveness and precision. However, it's worth noting that the smallest input size of  $224 \times 224$  resulted in a significant drop in the probability of detection. This observation suggests that the models may struggle with detecting smaller objects or details. The training settings, including the number of iterations and optimization techniques, can significantly influence the model's convergence and generalization capabilities. In this study, the best results in terms of average precision on the validation test were obtained after a specific number of iterations for each model. However, it is essential to explore the convergence behavior and assess whether further iterations or fine-tuning could lead to performance improvements.

## 5. Conclusion

In conclusion, this thesis presented an in-depth analysis of object detection and tracking using YOLO v.4 and Tiny YOLO v.4 models. The models demonstrated strong performance in detecting and localizing objects of interest in various scenarios. The quantitative evaluation showcased high mean average precision (mAP) and intersection over union (IoU) scores, indicating their effectiveness in object detection.

Through qualitative analysis and case studies, the strengths and limitations of the models were examined. While the models exhibited accurate object detection and tracking in many cases, challenges were observed with occluded or small objects and in complex scenes. These findings highlight the need for further research and fine-tuning to enhance the models' performance in real-world scenarios.

The future work section suggests potential directions for improving the models, including dataset expansion, fine-tuning, domain-specific adaptations, integration with tracking algorithms, and real-time deployment optimization. Addressing these areas will contribute to advancing the field of object detection and tracking and enable their broader application in various domains.

Overall, the findings presented in this thesis demonstrate the advancements achieved by the proposed models and provide valuable insights for future research and development. By continually refining and enhancing the models, we can pave the way for more accurate and robust object detection and tracking systems in the future.

## 5.1. Future work

Based on the findings and discussions presented in this thesis, several avenues for future research and improvements can be explored:

1. **Dataset Expansion:** Consider augmenting the dataset with a larger and more diverse set of images, including variations in lighting conditions, object scales, and occlusions. This will help enhance the models' generalization capabilities and performance in real-world scenarios.
2. **Fine-tuning and Hyperparameter Optimization:** Further investigate the effects of fine-tuning the model and optimizing hyperparameters to improve detection accuracy and reduce false positives/negatives. Experiment with different learning rates, regularization techniques, and optimization algorithms to achieve better performance.
3. **Domain-Specific Model Adaptation:** Explore the adaptation of the models to specific domains or applications. Fine-tune the models on domain-specific datasets or consider transfer learning from pre-trained models to improve performance and reduce the need for extensive training on new datasets.
4. **Integration with Tracking Algorithms:** Investigate the integration of the object detection models with tracking algorithms to enable robust and accurate object tracking across consecutive frames. Explore techniques such as Kalman filtering, Hungarian algorithm-based tracking, or deep learning-based trackers to enhance object tracking capabilities.
5. **Real-Time Deployment Optimization:** Optimize the models' architecture and implement hardware acceleration techniques to achieve real-time performance on resource-constrained devices. Consider deploying the models on specialized hardware platforms, such as GPUs or dedicated AI accelerators, to improve inference speed and efficiency.

## References:

1. BOUDJIT, K., & RAMZAN, N. (2022). "HUMAN DETECTION BASED ON DEEP LEARNING YOLO-V2 FOR REAL-TIME UAV APPLICATIONS." *JOURNAL OF EXPERIMENTAL & THEORETICAL ARTIFICIAL INTELLIGENCE*, 34(3), 527-544.
2. HUNG, G. L., SAHIMI, M. S. B., SAMMA, H., ALMOHAMAD, T. A., & LAHASAN, B. (2020). "FASTER R-CNN DEEP LEARNING MODEL FOR PEDESTRIAN DETECTION FROM DRONE IMAGES." *SN COMPUTER SCIENCE*, 1, 1-9.
3. ROHAN, A., RABAH, M., & KIM, S. H. (2019). "CONVOLUTIONAL NEURAL NETWORK-BASED REAL-TIME OBJECT DETECTION AND TRACKING FOR PARROT AR DRONE 2." *IEEE ACCESS*, 7, 69575-69584.
4. MADASAMY, K., SHANMUGANATHAN, V., KANDASAMY, V., LEE, M. Y., & THANGADURAI, M. (2021). "OSDDY: EMBEDDED SYSTEM-BASED OBJECT SURVEILLANCE DETECTION SYSTEM WITH A SMALL DRONE USING DEEP YOLO." *EURASIP JOURNAL ON IMAGE AND VIDEO PROCESSING*, 2021(1), 1-14.
5. ALSANAD, H. R., SADIK, A. Z., UCAN, O. N., ILYAS, M., & BAYAT, O. (2022). "YOLO-V3 BASED REAL-TIME DRONE DETECTION ALGORITHM." *MULTIMEDIA TOOLS AND APPLICATIONS*, 81(18), 26185-26198.
6. JIANG, C., REN, H., YE, X., ZHU, J., ZENG, H., NAN, Y., ... & HUO, H. (2022) FOCUS ON OBJECT DETECTION FROM UAV THERMAL INFRARED IMAGES AND VIDEOS USING YOLO MODELS (JIANG

- ET AL., 2022). THIS REFERENCE CONTRIBUTES TO UNDERSTANDING THE APPLICATION OF YOLO MODELS IN DETECTING OBJECTS FROM THERMAL INFRARED IMAGERY CAPTURED BY UAVS.
7. SADYKOVA, D., PERNEBAYEVA, D., BAGHERI, M., & JAMES, A. (2019) DISCUSS REAL-TIME DETECTION OF OUTDOOR HIGH VOLTAGE INSULATORS USING UAV IMAGING (SADYKOVA ET AL., 2019). THIS REFERENCE PROVIDES INSIGHTS INTO THE APPLICATION OF UAV IMAGING AND YOLO-BASED OBJECT DETECTION IN IDENTIFYING HIGH VOLTAGE INSULATORS.
  8. WU, Q., & ZHOU, Y. (2019) PRESENT REAL-TIME OBJECT DETECTION BASED ON UNMANNED AERIAL VEHICLES (WU & ZHOU, 2019). THIS REFERENCE FOCUSES ON THE APPLICATION OF UAVS FOR REAL-TIME OBJECT DETECTION.
  9. HU, Y., WU, X., ZHENG, G., & LIU, X. (2019) PROPOSE OBJECT DETECTION OF UAV FOR ANTI-UAV BASED ON IMPROVED YOLO v3 (HU ET AL., 2019). THIS REFERENCE CONTRIBUTES TO UNDERSTANDING THE USE OF IMPROVED YOLO v3 FOR DETECTING AND COUNTERING UAV THREATS.
  10. HU, Y., WU, X., ZHENG, G., & LIU, X. (2019) PRESENT OBJECT DETECTION OF UAV FOR ANTI-UAV BASED ON IMPROVED YOLO v3 (HU ET AL., 2019). THIS REFERENCE PROVIDES FURTHER INSIGHTS INTO THE APPLICATION OF IMPROVED YOLO v3 FOR DETECTING AND COUNTERING UAV THREATS.
  11. BENJDIRA, B., KHURSHEED, T., KOUBAA, A., AMMAR, A., & OUNI, K. (2019, FEBRUARY). CAR DETECTION USING UNMANNED AERIAL VEHICLES: COMPARISON BETWEEN FASTER R-CNN AND YOLOV3. IN 2019 1ST INTERNATIONAL CONFERENCE ON UNMANNED VEHICLE SYSTEMS-OMAN (UVS) (PP. 1-6). IEEE.
  12. BARISIC, A., PETRIC, F., & BOGDAN, S. (2022). SIM2AIR-SYNTHETIC AERIAL DATASET FOR UAV MONITORING. IEEE ROBOTICS AND AUTOMATION LETTERS, 7(2), 3757-3764.
  13. AYALEW, A., & POOJA, D. (2019). A REVIEW ON OBJECT DETECTION FROM UNMANNED AERIAL VEHICLE USING CNN. INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY, 5, 241-243.
  14. HOSSAIN, S., & LEE, D. J. (2019). DEEP LEARNING-BASED REAL-TIME MULTIPLE-OBJECT DETECTION AND TRACKING FROM AERIAL IMAGERY VIA A FLYING ROBOT WITH GPU-BASED EMBEDDED DEVICES. SENSORS, 19(15), 3371.
  15. UNLU, E., ZENOU, E., RIVIERE, N., & DUPOUY, P. E. (2019, JANUARY). AN AUTONOMOUS DRONE SURVEILLANCE AND TRACKING ARCHITECTURE. IN 2019 AUTONOMOUS VEHICLES AND MACHINES CONFERENCE, AVM 2019 (VOL. 2019, PP. 35-1).
  16. WANG, H., YANG, G., LI, E., TIAN, Y., ZHAO, M., & LIANG, Z. (2019, JULY). HIGH-VOLTAGE POWER TRANSMISSION TOWER DETECTION BASED ON FASTER R-CNN AND YOLO-V3. IN 2019 CHINESE CONTROL CONFERENCE (CCC) (PP. 8750-8755). IEEE.
  17. SINGH, A. K., DWIVEDI, A. K., NAHAR, N., & SINGH, D. (2021, JULY). RAILWAY TRACK SLEEPER DETECTION IN LOW ALTITUDE UAV IMAGERY USING DEEP CONVOLUTIONAL NEURAL NETWORK. IN 2021 IEEE INTERNATIONAL GEOSCIENCE AND REMOTE SENSING SYMPOSIUM IGARSS (PP. 355-358). IEEE.
  18. LI, Y., YUAN, H., WANG, Y., & XIAO, C. (2022). GGT-YOLO: A NOVEL OBJECT DETECTION ALGORITHM FOR DRONE-BASED MARITIME CRUISING. DRONES, 6(11), 335.

19. PHADTARE, M., CHOUDHARI, V., PEDRAM, R., & VARTAK, S. (2021). COMPARISON BETWEEN YOLO AND SSD MOBILENET FOR OBJECT DETECTION IN A SURVEILLANCE DRONE. *INT. J. SCI. RES. ENG. MANAG*, 5, 1-5.
20. MITTAL, P., SINGH, R., & SHARMA, A. (2020). DEEP LEARNING-BASED OBJECT DETECTION IN LOW-ALTITUDE UAV DATASETS: A SURVEY. *IMAGE AND VISION COMPUTING*, 104, 104046.
21. A. A. MICHEAL, K. VANI, S. SANJEEVI, AND C. H. LIN, "A TOOL TO ENHANCE THE CAPACITY FOR DEEP LEARNING BASED OBJECT DETECTION AND TRACKING WITH UAV DATA," *THE INTERNATIONAL ARCHIVES OF PHOTOGRAMMETRY, REMOTE SENSING AND SPATIAL INFORMATION SCIENCES*, VOL. 43, PP. 221-226, 2020.
22. M. NALAMATI, A. KAPOOR, M. SAQIB, N. SHARMA, AND M. BLUMENSTEIN, "DRONE DETECTION IN LONG-RANGE SURVEILLANCE VIDEOS," IN 2019 16TH IEEE INTERNATIONAL CONFERENCE ON ADVANCED VIDEO AND SIGNAL BASED SURVEILLANCE (AVSS), PP. 1-6, SEP. 2019.
23. R. K. CHANDANA AND A. C. RAMACHANDRA, "REAL TIME OBJECT DETECTION SYSTEM WITH YOLO AND CNN MODELS: A REVIEW," *ARXIV PREPRINT ARXIV:2208.00773*, 2022.
24. Y. C. LAI AND Z. Y. HUANG, "DETECTION OF A MOVING UAV BASED ON DEEP LEARNING-BASED DISTANCE ESTIMATION," *REMOTE SENSING*, VOL. 12, NO. 18, PP. 3035, 2020.
25. O. SAHIN AND S. OZER, "YOLODRONE: IMPROVED YOLO ARCHITECTURE FOR OBJECT DETECTION IN DRONE IMAGES," IN 2021 44TH INTERNATIONAL CONFERENCE ON TELECOMMUNICATIONS AND SIGNAL PROCESSING (TSP), PP. 361-365, JUL. 2021.
26. C. LI, X. SUN, AND J. CAI, "INTELLIGENT MOBILE DRONE SYSTEM BASED ON REAL-TIME OBJECT DETECTION," *JOURNAL OF ARTIFICIAL INTELLIGENCE*, VOL. 1, NO. 1.
27. S. WANG, "RESEARCH TOWARDS YOLO-SERIES ALGORITHMS: COMPARISON AND ANALYSIS OF OBJECT DETECTION MODELS FOR REAL-TIME UAV APPLICATIONS," IN *JOURNAL OF PHYSICS: CONFERENCE SERIES*, VOL. 1948, NO. 1, P. 012021, JUN. 2021.
28. T. JINTASUTTISAK, E. EDIRISINGHE, AND A. ELBATTAY, "DEEP NEURAL NETWORK BASED DATE PALM TREE DETECTION IN DRONE IMAGERY," *COMPUTERS AND ELECTRONICS IN AGRICULTURE*, VOL. 192, PP. 106560, 2022.
29. P. NOUSI, I. MADEMLIS, I. KARAKOSTAS, A. TEFAS, AND I. PITAS, "EMBEDDED UAV REAL-TIME VISUAL OBJECT DETECTION AND TRACKING," IN 2019 IEEE INTERNATIONAL CONFERENCE ON REAL-TIME COMPUTING AND ROBOTICS (RCAR), PP. 708-713, AUG. 2019.
30. REDMON, J., & FARHADI, A. (2018). YOLOv3: AN INCREMENTAL IMPROVEMENT. *ARXIV PREPRINT ARXIV:1804.02767*.
31. LECUN, Y., BENGIO, Y., & HINTON, G. (2015). DEEP LEARNING. *NATURE*, 521(7553), 436-444.
32. LIN, T. Y., MAIRE, M., BELONGIE, S., HAYS, J., PERONA, P., RAMANAN, D., ... & ZITNICK, C. L. (2014). MICROSOFT COCO: COMMON OBJECTS IN CONTEXT. IN *EUROPEAN CONFERENCE ON COMPUTER VISION (ECCV)* (PP. 740-755). SPRINGER.
33. DENG, J., DONG, W., SOCHER, R., LI, L. J., LI, K., & FEI-FEI, L. (2009). IMAGENET: A LARGE-SCALE HIERARCHICAL IMAGE DATABASE. IN 2009 IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (PP. 248-255). IEEE.
34. SIMONYAN, K., & ZISSERMAN, A. (2014). VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION.

35. REDMON, J., & FARHADI, A. (2018). YOLOV3: AN INCREMENTAL IMPROVEMENT. ARXIV PREPRINT ARXIV:1804.02767.
36. GOODFELLOW, I., BENGIO, Y., & COURVILLE, A. (2016). DEEP LEARNING. MIT PRESS.
37. REDMON, J., & FARHADI, A. (2020). YOLOV4: OPTIMAL SPEED AND ACCURACY OF OBJECT DETECTION. ARXIV PREPRINT ARXIV:2004.10934.
38. LIN, T. Y., GOYAL, P., GIRSHICK, R., HE, K., & DOLLÁR, P. (2017). FOCAL LOSS FOR DENSE OBJECT DETECTION. IN PROCEEDINGS OF THE IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION (PP. 2980-2988).
39. ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C., ... & GHEMAWAT, S. (2016). TENSORFLOW: LARGE-SCALE MACHINE LEARNING ON HETEROGENEOUS SYSTEMS. SOFTWARE AVAILABLE FROM TENSORFLOW.ORG.
40. REDMON, J., & FARHADI, A. (2020). YOLOV4: OPTIMAL SPEED AND ACCURACY OF OBJECT DETECTION. ARXIV PREPRINT ARXIV:2004.10934.
41. RUSSAKOVSKY, O., DENG, J., SU, H., KRAUSE, J., SATHEESH, S., MA, S., ... & FEI-FEI, L. (2015). IMAGENET LARGE SCALE VISUAL RECOGNITION CHALLENGE. INTERNATIONAL JOURNAL OF COMPUTER VISION, 115(3), 211-252.
42. JOSEPH REDMON ET AL., "YOU ONLY LOOK ONCE: UNIFIED, REAL-TIME OBJECT DETECTION." ARXIV PREPRINT ARXIV:1506.02640 (2015).
43. JOSEPH REDMON AND ALI FARHADI, "YOLO9000: BETTER, FASTER, STRONGER." PROCEEDINGS OF THE IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR) (2017).
44. LECUN, Y., BENGIO, Y., & HINTON, G. (2015). DEEP LEARNING. NATURE, 521(7553), 436-444.
45. KRIZHEVSKY, A., SUTSKEVER, I., & HINTON, G. E. (2012). IMAGENET CLASSIFICATION WITH DEEP CONVOLUTIONAL NEURAL NETWORKS. IN ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS (NIPS), 1097-1105.