ISSN: 2182-2069 / E-ISSN: 2182-2077

Design and Implementation of Anti-Drone System Based on Computer Vision with You Only Look Once (YOLO) Algorithm

Nabeel Aad Lafta[1*], and Saad Saffah Hreshee[2]

[1*]Department of Electrical Engineering, University of Babylon, Iraq.
eng876.nabeel.aad@student.uobabylon.edu.iq, https://orcid.org/0009-0008-8025-7522

[2]Department of Electrical Engineering, University of Babylon, Iraq.
eng.saad.saffah@uobabylon.edu.iq, https://orcid.org/0000-0002-7547-1443

Received: July 30, 2025; Revised: September 10, 2025; Accepted: October 17, 2025; Published: November 28, 2025

Abstract

Due to the rapid development of drone manufacturing technologies and their ease of use, many people can own and use drones for services such as photography, entertainment, scientific, security surveillance, and so on. However, terrorists and criminals have exploited drones in attacks on important places and vital institutions due to a lack of protection from anti-drone systems. This paper presents an anti-drone system based on computer vision using the You Only Look Once (YOLO11) algorithm to detect, track, and destroy unauthorized drones. The proposed system leverages computer vision techniques to detect drones in real-time with high accuracy and low latency. The system integrates the YOLO11 algorithm with automatic weapon control, enabling precise targeting and destruction of detected drones. The authors designed a new mathematical synchronization method to align firearm movements with drone coordinates for improved tracking performance. We implemented a prototype of the proposed project that can detect and track an unknown drone using a Raspberry Pi as a microcomputer, issue an audible alarm upon detection using an Arduino, track the drone, and destroy it using a robot arm as an automatic weapon that moves based on coordinates from the detection data. The practical experiments demonstrated high detection accuracy (precision: 95.7%, recall: 91.5%, mAP50: 91.1%, and FPS: 29.6), outperforming previous YOLO versions and other existing anti-drone solutions.

Keywords: Anti-Drone, Computer Vision, YOLO11 Algorithm, Automatic Weapon.

1 Introduction

In the last few years, drones, sometimes-called unmanned aerial vehicles (UAVs) have developed and become available in popular markets. They are widely accessible to the public due to their ease of use and low cost. Drones have many benefits and use in all areas of life; for example, they are used in the photography, scientific research, agriculture, health, entertainment, security, the military, surveillance, and other fields. Despite the many benefits provided by drones, they have deadly and destructive disadvantages and risks, as terrorists and criminals use them to achieve their goals, such as targeting airports, important government military institutions, and other important places (Srimuang, 2023). Criminals can also use them in drug trafficking, smuggling goods, and other criminal enterprises (Guru

Journal of Interner Services and Information Security (JISIS), volume: 15, number: 4 (November), pp. 324-344.
DOI: *10.58346/JISIS.2025.I4.024*

*Corresponding author: Department of Electrical Engineering, University of Babylon, Iraq.

Prasad & Badrinarayanan, 2025). The media has reported many terrorist and criminal incidents using drones (Tu et al., 2024).

Due to the small size and plastic body of drones and their ability to fly at low altitudes between buildings, military air defense systems cannot detect or shoot them down, incurring high costs for these systems in protecting important places such as government institutions and airports from the dangers drones pose (AbdAli & Maaroof, 2022). Therefore, it is necessary to design high-efficiency and low-cost anti-drone systems to protect these important areas (Najafi et al., 2025; Popescu, 2021; Maaroof & Bouhlel, 2025). Various researchers have attempted to design anti-drone systems using various scientific techniques and methods that are available (Kavitha, 2024). Some researchers examined the radio frequency (RF) tab by detecting drone control signals, considering most drones are ground controlled, using radio signals to communicate and transfer data between the drone and the control ground station. The limitations of the RF detection include the limited range of detection and the detection of self-programmed drones with no requirement for a ground station (Singh & Nair, A. 2022). Others have used acoustics to detect drones by their sound using artificial intelligence and deep learning methods. The limitation of the acoustics technology is the limited range of detection and the environmental noise sensitivity of detection. Others have used radar technology for the detection of drones. Although radar technology was good at detecting flying objects, it was not a good choice for detecting drones, because of their small size, low-flying altitude, and plastic nature (Shi et al., 2018). Therefore, in this study, I applied computer vision techniques using the YOLO11 algorithm that would detect unknown drones in real time and then destroy them is more useful than other protocols as a possibility for designing an anti-drone system.

1.1 Problem Statement

The research problem we seek to resolve in this study is unprotected important places (such as facilities and airports) with specialized protection systems that can eliminate the threat of hostile drones. Due to the cost of air defense weapons systems and the fact that radar is not efficient at detecting drones due to their small size and low-altitude flight in between buildings, especially in urban environments location there is a need to provide simple, suitable and effective anti-drone systems to protect important places.

1.2 Research Objectives

This study aims to design and implement a highly efficient, low-cost, and easy-to-use anti-drone system to protect important places such as government institutions from the threat of drone attacks. This system uses computer vision with the YOLO11 algorithm to detect drones and then issues an audio alarm; the hostile drone is then tracked and automatically destroyed by a firearm guided by the detection algorithm. In addition to technical contributions, this study also addresses the legal and ethical implications of deploying autonomous anti-drone systems.

1.3 Paper Organization

The following sections of this paper include; Section (2) explanation literature review of some similar studies. Section (3) explains the YOLO object detection methodology and its development. Section (4) describes the proposed project, its hardware components, and its software. Section (5) conducts a practical experiment of the proposed system, including data collection, training, detection, tracking, and destruction of hostile drones. Section (6) presents the results, discusses them, and compares the models trained in this study. Section (7) includes the conclusion.

2 Literature Review

As previously explained, there are numerous security problems resulting from terrorist and criminal gangs possessing drones, which threaten the lives of citizens and the security of important institutions. Therefore, providing an anti-drone system is very important to protect against potential drone attacks. Many researchers have directed their research activities to the field of air security.

Deep learning-based techniques have recently advanced drone detection, leading to systems using deep learning-based methods that are much more capable in practice. In the work of (Behera & Raj, 2020), the authors used the YOLOv4 model to detect various classes of drones in both static and dynamic conditions, with a mean average precision (mAP) of 0.74 after training YOLOv4 for 150 epochs. While the detection of drones was addressed, and the visual similarity of drones to birds is noted, (Sanae Fujii, 2021) identified a more recent important work with drone detection using a Center-Net detection model aided through data augmentation and hard negative mining. The authors achieved an accuracy of 72.03% and mAP of 72.13% vs. several competing methods.

The challenge of drone detection within aerial imagery is rooted in the multi-scale nature of object variation and the close proximity of some targets. To address this (Wang et al., 2021) designed SPB-YOLO, built on YOLOv5. The new model included a Strip Bottleneck module to help account for space variation better. The authors developed a Path Aggregation Network (PANet) and added a fourth detection head to better account for target proximity. The addition of these modules and structures improved mAP by 5.3%. A joint effort in YOLOv5 was produced by (Al-Qubaydhi et al., 2022) using transfer learning and 1359 drone images, which were subsequently augmented with preprocessing techniques. Their model reported 94.7% classification accuracy. similarly, (Liu & Luo, 2022) used previous studies to cite several improvements on YOLOv5; incorporating an Efficient Lite backbone, spatial feature fusion and a custom angle-based loss function, achieved 93.54% accuracy, 91.09% recall, and an inference speed of 94.82 milliseconds.

Synthetic data has also been used beneficially for their models (Wisniewski et al., 2022), trained DenseNet201 using artificially generated drone images and evaluated on real data, reporting 92.4% accuracy and their precision, recall and F1 were around 88%. (Samadzadegan et al., 2022) also used YOLOv4 with non-standard training procedures to get improved detection performance to distinguish drones from birds. Their results reported 83% accuracy, 84% mAP and 81% IOU. More recently and in terms of completeness (Yilmaz & Kutbay, 2024) applied YOLOv8 to drone detection tasks and used annotated images processed with Roboflow and trained in Google Colab to complete their project, obtaining an accuracy of 94.6%. Similar studies, such as (Suewongsuwan et al., 2024) combined YOLOv8 with the Deep SORT tracking algorithm and expanded their dataset with augmentation to include 7627 images achieving a detection accuracy of 95.2%.

The authors (Hakani & Rawat, 2024) developed a detection model that employed YOLOv9 trained on 6,996 drone images and validated on 2,999, showing a detection accuracy of 94.6%. The performance test was conducted and the detection results were found to be 94.6% accurate. The (Nguyen et al., 2025) designed a lightweight (LW UAV) YOLO (YOLOv10) for detecting drones, it achieved a maximum accuracy of 98.3% across all testing conditions.

Despite advancements, drone detection continues to encounter substantial challenges, the biggest being real-time detection, and the inability to discriminate the drone from similar objects in the air environment. Most detection models either use speed or accuracy as the principle performance measure, making them unsuitable for real-word applications. It is also common for most models to be trained across a few object classes, which limits the transferability and generalization to actual environments.

Another issue is the robustness of the existing models while imaging drones. Imaging of the drones is affected by several dynamic environments (e.g. altered weather, altitude variations and speeds of ops drone) and there are often birds and other airborne objects in the image frame, which compound problems with false classifications and false negatives. The above concerns highlight the need for more smarter, more adaptable, and efficient systems for detecting drones. This motivation is to fill in the gaps mentioned above by proposing solutions that establish a balance between high detection accuracy and fast inference speed, while ensuring that the solution is robust enough to deploy in complicated real-world scenarios such as aerial surveillance.

3 Methodology

Computer vision techniques detect objects using deep learning algorithms. The most well-known of these algorithms is YOLO, which has proven its worth in detecting objects in real-time with high speed and accuracy, as shown in the following sections.

3.1 Computer Vision

Computer vision is a field of artificial intelligence that aims to enable computers to understand and process images and videos in a similar way to how humans understand the visual world. Computer vision relies on developing algorithms and mathematical models that allow computers to analyze images and extract information from them, recognizing objects, identifying faces, tracking motion, and understanding three-dimensional scenes. Object detection using computer vision is divided into two types: two-stage detection and one-stage detection (Hakani & Rawat, 2024).

3.1.1 Two-Stage Object Detection

The first phase in two-stage detection is identifying a set of regions that contain the observed or desired target. The second phase produces a bounding box around the detected target, as shown in Figure 1. Examples of two-stage detection techniques include CNN (Mohamed S. Sawah et al., 2023), mask RCNN (Mahmoud et al., 2020), and Faster RCNN (Ren et al., 2016). The backbone of this model is a convolutional neural network (CNN), typically VGG-16 or Res-Net, which produces feature maps. The region proposal network then generates suggestions for locations where objects are likely to occur (He et al., 2017).

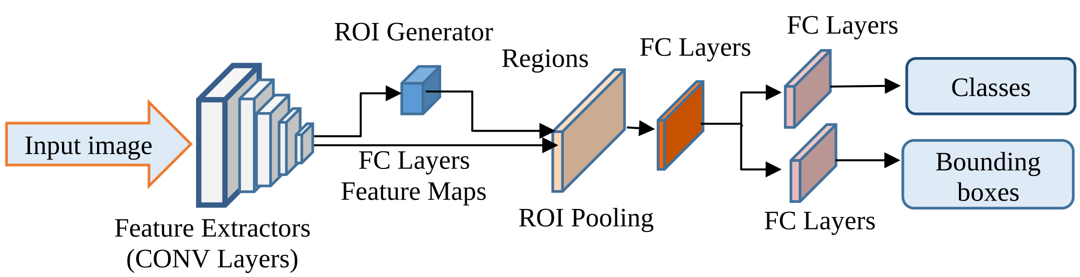

Figure 1: Architecture of two-stage object detection (Hakani & Rawat, 2024)

Region-of-interest pooling takes these suggestions and the corresponding feature maps and performs max pooling to produce fixed-size feature maps. The output from region-of-interest clustering is then fed into a classifier to predict the object class and draw a specified box. Two-stage detection techniques are highly accurate, but can take a long time to detect objects (Cheng et al., 2018; He et al., 2017).

3.1.2 One-Stage Object Detection

With single-stage object detection methods the target is detected, identified in a category, and a bounding box is drawn around it without much loss in accuracy, as seen in Figure 2. Single- stage object detection methods are faster and better approximate real-time detection response than two-stage methods (Carranza-García et al., 2020). One such theoretically interchangeable method that is most used in a single-stage object detection method is called You Only Look Once (YOLO), which has quickly become the leader in real-time object detection research. Within this method, the image is divided into an equal grid of cells, where according to the area within the fixed grid cell each cell predicts the presence of the object and draw a predetermined square around the target, which will be detailed explained in the extended version of the following sections. The benefits of using this method include the high accuracy and speed of detecting objects, and the possibility that YOLO could detect the objects in a real time frame such was required in the design of the intended project (Hakani et al., 2024).

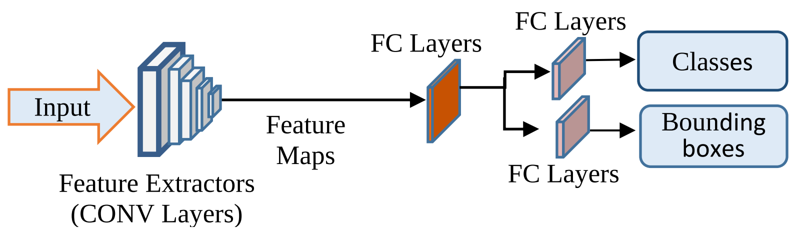

Figure 2: Architecture of one-stage object detection (Hakani & Rawat, 2024)

3.2 You Only Look Once (YOLO) Algorithm

In 2015, Joseph Redmon presented YOLO, a single-stage object detection algorithm (Redmon et al., 2016). The single-stage detection method is considered the best for object detection compared with two-stage algorithms. YOLO detects objects in one stage and at high speeds, faster than those using two-stage detection (Aria Hendrawan et al., 2024; Redmon et al., 2016). YOLO is commonly used in video surveillance for real-time object detection applications, the architecture of it as shown Figure 3.

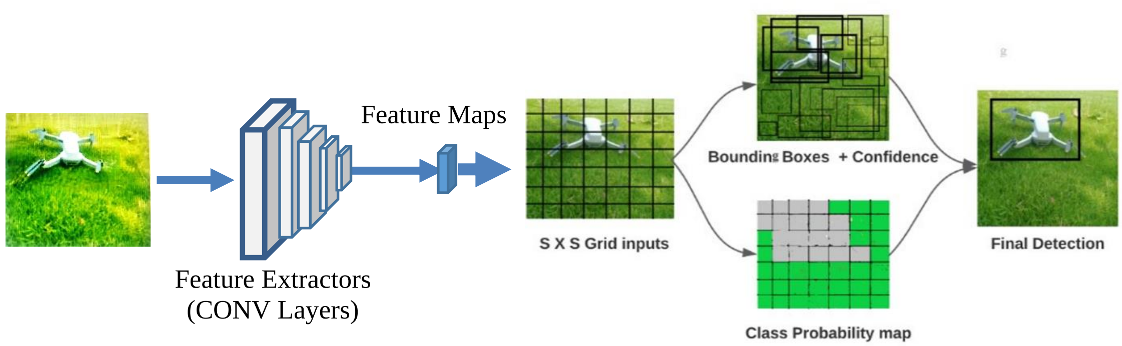

Figure 3: Architecture of the YOLO algorithm

The technical architecture of this algorithm includes three steps. First, images are passed through multiple CNNs to extract image features, called the backbone, wherein the data are passed through two fully connected layers to obtain general features. Then, the data are reshaped into a two-dimensional space to predict the presence of objects in one stage for each network, called the neck. Finally, the image is divided into a symmetrical grid, objects are detected in each square, and a bounding box is created for the detected objects, called the head (Alnajjar et al., 2024).

The algorithm divides the image into an (S by S) grid, and for each grid cell, it predicts the bounding boxes (B), the center coordinates of box is (x, y), the height and width of the box is (h, w), the confidence for those boxes is (P) and the class probabilities (C). These predictions are encoded as [S * S, (B * (x, y, h, w, p) +C)] (Shi & Li, 2020). To clarify how YOLO works, it is necessary to focus on important concepts related to evaluating its performance as a single-stage object detection method based on the parameters TP, FP, and FN, that is, the true positive, false positive, and false negative results (Zhao et al., 2024), respectively (Srimuang & Doungmala, 2023).

Precision (P) is the percentage of detected objects within the entered data, symbolized by a positive match. Calculating the detection accuracy for the entered class helps, us know the extent to which this class matches the real class; the higher the percentage, the more reliable the detection result.

$$P = \frac{TP}{TP + FP} \tag{1}$$

Recall (R) is a calculation of the percentage of all positive inputs for a single category. If there is more than one category, it is calculated separately for each category.

$$R = \frac{TP}{TP + FN} \tag{2}$$

Mean Average Precision (mAP) represents the average accuracy in detecting the input classes; in other words, the bounding box of the class is compared with the bounding box of the actual class using the prediction of the proposed algorithm (Dadrass Javan et al., 2022).

$$AP = \int_0^1 P(R)dR \tag{3}$$

$$mAP = \frac{1}{N}\sum_{n\epsilon N} AP(n) \tag{4}$$

Where N is the number of classes.

3.3 YOLO Development

Researchers have continued to develop YOLO after its initial version, and many modified and improved versions have been released. An advanced version of this algorithm has been released almost every year, sometimes more than one. Researchers developing YOLO have focused on detection speed and results accuracy, and modern versions are compatible with real-time detection applications (Khanam & Hussain, 2024). Figure 4 shows YOLO's stages of development and the techniques used (Jegham et al., 2024).

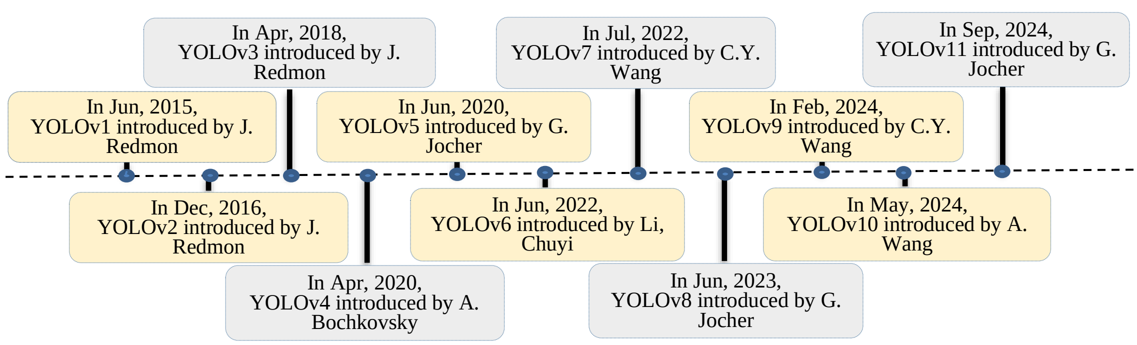

Figure 4: Shows YOLO's stages of development and the techniques used (Jegham et al., 2024)

3.4 YOLO11 Architecture

YOLO11 is the latest version, developed in 2024 by Ultralytics. It is an extension of its previous versions, with improvements in features such as detection speed, results accuracy, and ease of implementation. YOLO11 is faster and more accurate than previous versions due to significant improvements made to its infrastructure (Jegham et al., 2024), making it suitable for real-time object detection. This fits the requirements of the proposed project in this study. Figure 5, shows a block diagram of the YOLO11 algorithm, demonstrating that it is an improved or enhanced version of the YOLOv8 algorithm. One of the most important modifications in the new version is the addition of the C2PSA (Cross-Stage Partial with Self-Attention) unit, which works on the output of the partial networks across the detection stages.

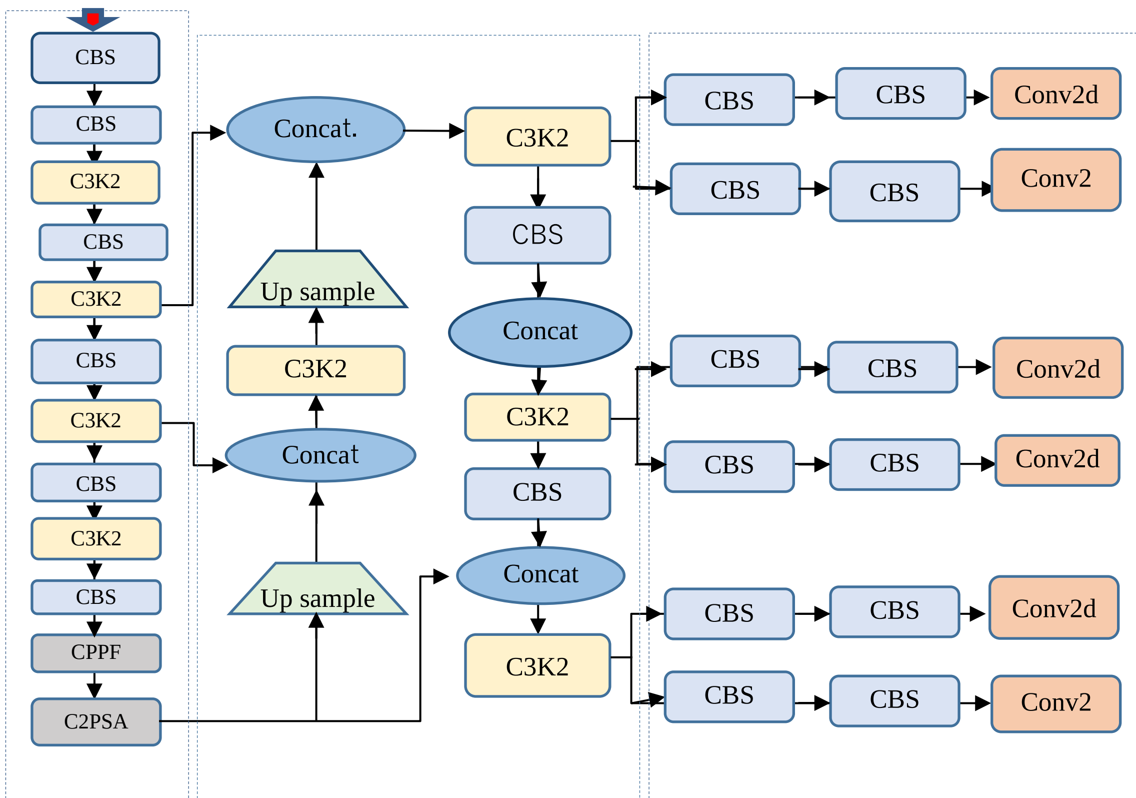

Figure 5: Show the architecture of yolo11 algorithm (backbone, neck and head) (Jegham et al., 2024)

It improves the process of capturing accurate information, supporting high detection speeds and increasing accuracy even when detecting small objects (Sapkota et al., 2024; Sharma et al., 2024). In addition to other modifications to the new algorithm, it replaces block C3K2 instead of C2f. This replacement improves the performance of the bottleneck, as it uses two small convolutions, improving on YOLOv8, which uses one large convolution. This new block uses a smaller kernel with high efficiency and speed. The technologies added to the latest version of YOLO make it faster and more accurate than previous versions. These improvements include the performance curves of all YOLO versions in terms of performance speed and accuracy (Jegham et al., 2024), as shown in Figure 6. The features that distinguish YOLO11 make it suitable for detecting and tracking drones in real-time, so it was used to design the proposed project in this study.

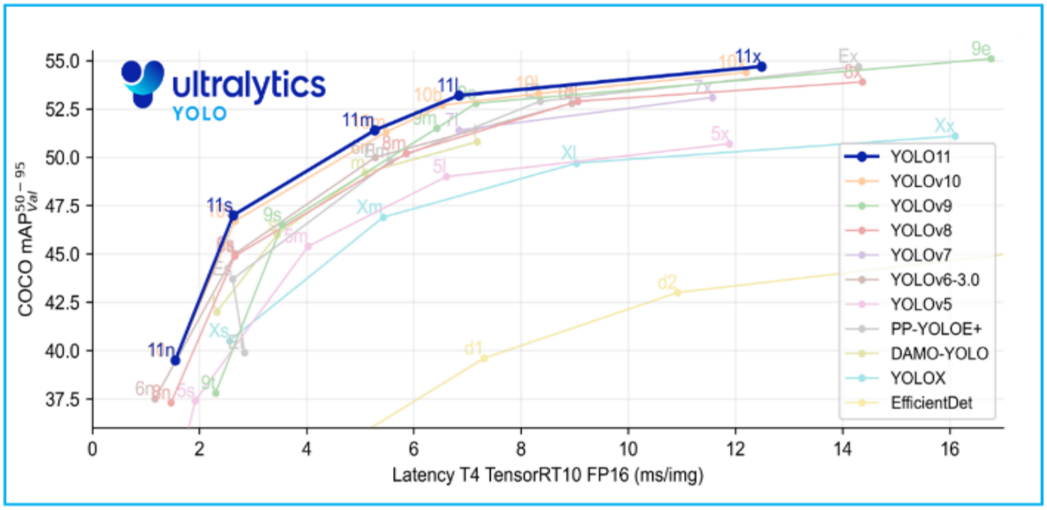

Figure 6: Latency–accuracy curves of YOLO versions (Sapkota & Karkee, 2024)

4 The Proposed System

The proposed system in this study is an anti-drone system using the YOLO11 algorithm. Its features are listed below, and Figure 7, shows its design.

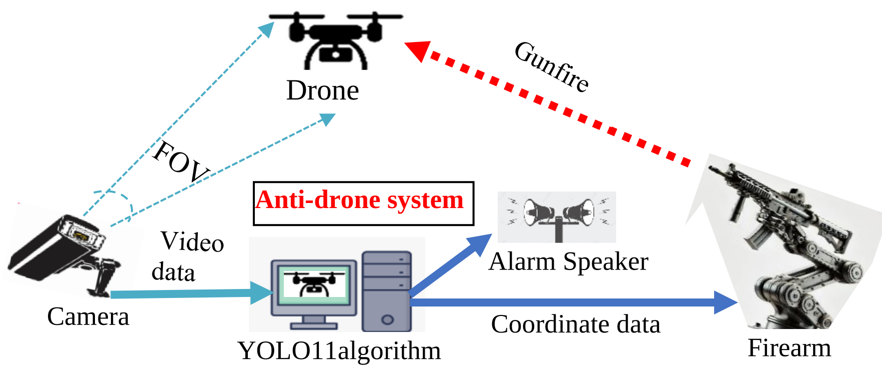

Figure 7: The proposed anti-drone system

- A camera for capturing video as surveillance data.
- Drone detection via computer vision in real-time using a microcomputer programmed with the YOLOv11 algorithm.
- An alarm unit to provide an early warning for the detection of an unknown drone.
- An automated weapon unit to destroy unknown drones with a robot arm equipped with a firearm.

4.1 Hardware system

The hardware of the prototype anti-drone system comprises the parts shown in Figure 8 they are listed as follows:

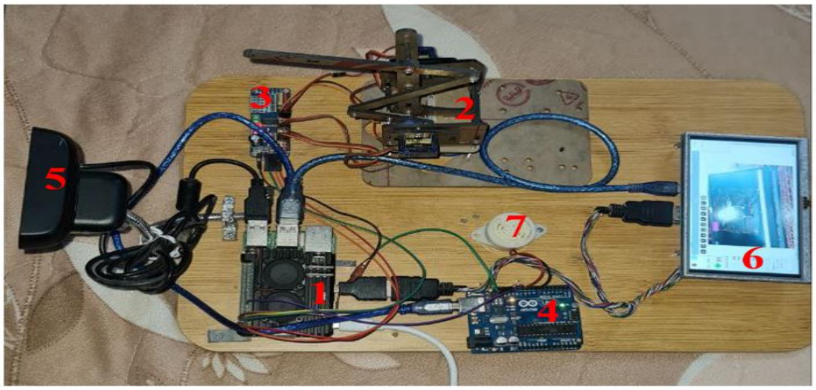

Figure 8: Hardware components

1. Raspberry Pi 5: A small, single-board computer featuring an ARM1176JZF-S CPU with 8 GB of RAM, programmed with the YOLO11 algorithm.
2. 4-DOF: An acrylic arm with MG996R servos that is easy to assemble as an automatic weapon. It is controlled by a serial communication protocol that sets the position of each servo, using a firearm to destroy the detected enemy drone.
3. PWM Servo Driver: A board designed to control servos or motors (type: PCA9685 16-Channel). It is commonly used in robotics, automation, and other projects involving multiple servomotors.
4. Arduino Uno: This operates the speaker with three different alarm tones for alerts.
5. Web camera: This is used for video surveillance.
6. Screen (5 inches): This shows the detection results.
7. Speaker: This issues alarm sounds to alert users to unknown drone threats.

4.2 Software System

The software of the proposed system includes several sequential stages, as shown in Table 1.

Table 1: Software system

Name	Version	Description
Raspberry Pi 5 OS	Raspbian/Linux/12	Operating system (Bookworm)
Python	V / 3.11	Programming language
Pytorch	V / 2.2.0	Detection library
Ultralytics	V / 8.3.73	YOLO object detection library
Open-cv	V / 4.9.0	Video and image library
Thonny	V / 4.2.0	Code management
PWM	Servo-Kit/1.0.0	Servomotor controller

4.3 System Cost, Scalability, and Deployment Potential

The cost estimate provided in this study (350–400) $ pertains exclusively to the prototype developed for experimental and proof-of-concept purposes. This reflects the basic expenses associated with components such as the Raspberry Pi 5, Arduino Uno, servomotors, camera, screen, speaker, and related accessories used to validate the detection, tracking, and control algorithms in a controlled environment.

It is important to clarify that this estimate does not account for the cost of the automated firearm or kinetic countermeasure intended for real-world drone neutralization. The inclusion of an operational automated weapon system would significantly increase the overall cost, given the price of the firearm, its mechanical platform, safety mechanisms, and the associated legal and regulatory compliance requirements.

In terms of scalability, the proposed system was designed with a modular and programmable architecture, enabling its replication and adaptation across different sites and applications. The YOLO-based detection algorithm can be used for deployment on more powerful edge devices, such as NVIDIA Jetson or Coral TPU platforms, to accommodate larger-scale or more complex installations. Furthermore, the open-source software stack offers flexibility for customization according to specific use cases, such as critical infrastructure protection, border security, or event surveillance. However, the actual cost of deploying a fully operational system would depend heavily on various factors, including the choice of countermeasure technology site-specific requirements, environmental conditions, and compliance with national and international regulations governing the use of automated weapons and airspace control. Therefore, the prototype cost presented should be regarded as a baseline figure, and not a comprehensive budget estimate for commercial or governmental deployment.

5 Practical Experiment to Implement the Proposed System

To implement the proposed system and explain the practical implementation steps, we will describe the work in four stages: Figure 9 (a, b, c, d) show each implemented stage sequentially. The first stage is data construction using the Rob flow server.

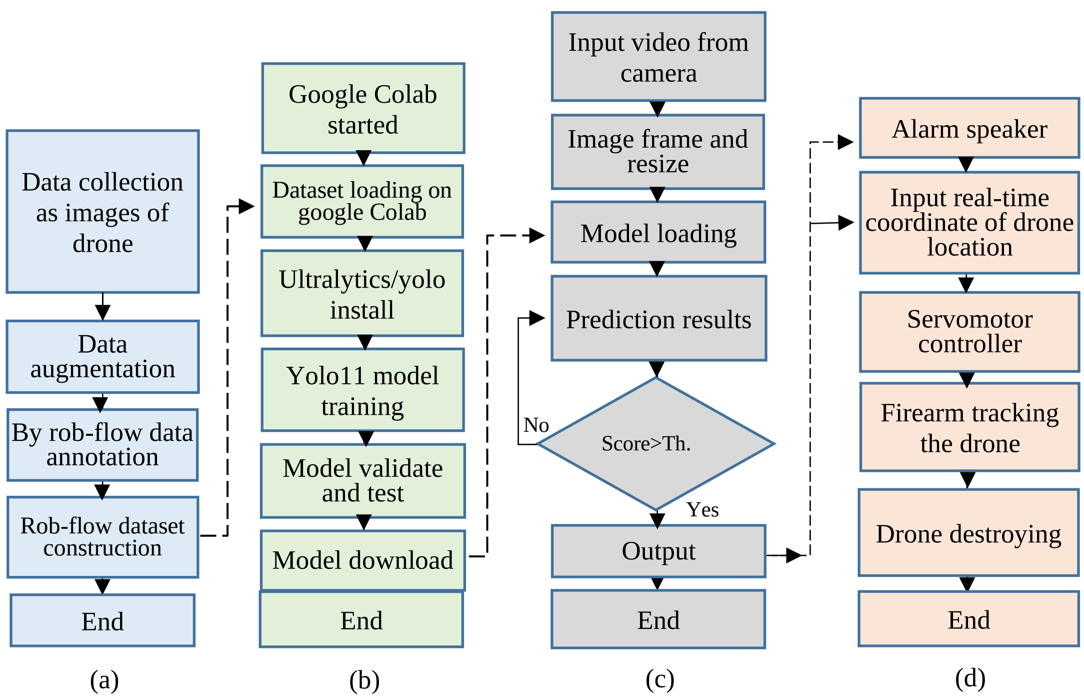

Figure 9: The flowchart of practical experiment of anti-drone system (a) the data construction, (b) training, (c) drone detection, (d) destroying the unknown drone

The second step is training the YOLO model using the Google Colab service. The third stage is drone detection based on the trained model resulting from the second stage, followed by predicting the coordinates of the detected drone in real-time. The last stage is the alarm speaker, followed by the firearm tracking the unknown drone and destroying it.

5.1 Data Construction

To obtain high-accuracy training results, numerous drone images were collected. To make the results accurate, numerous bird images were collected to train the model to distinguish between classes, as shown in Figure 9 (a). The images were collected based on public sources and public websites. In total, 1330 drone images and 1008 bird images were collected, totaling 2338 images. Using the Roboflow server, identification labeling and augmentation were performed for all drone and bird images. The dataset was divided into 70% training images, 20% validation images, and 10% test images. All images were issued with the coordinates of drones and birds to build the dataset in the YOLO (v8, v9, v10, v11) format; Figure 10: shows the Rob flow details.

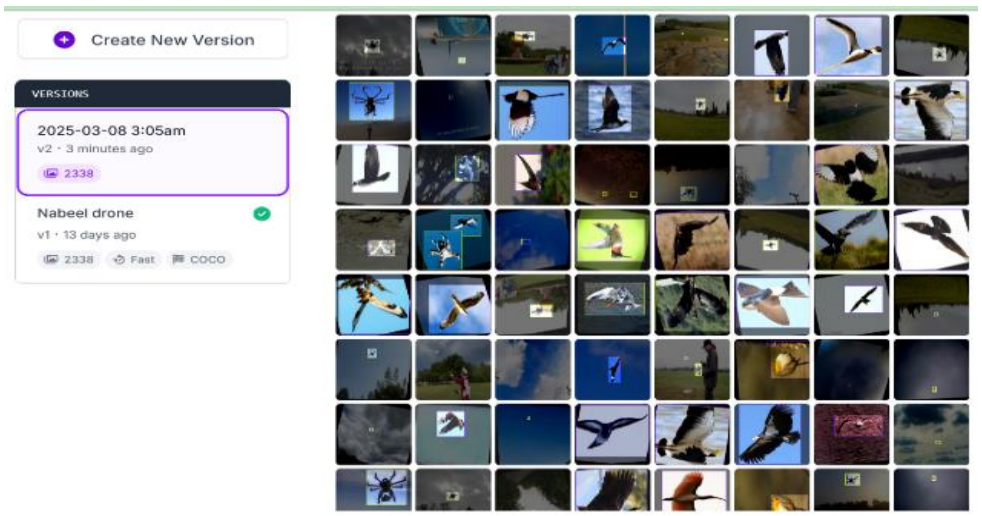

Figure 10: Data construction details using Roboflow service

5.2 Model Training

The YOLO models were trained using the Google Colab service by installing libraries required for the work environment, as shown in Table 2. Then, the prepared training data (images) were loaded from Roboflow in the YOLO (v8, v9, v10, v11) format for three sections: training, validation, and testing. The suitability of the data was checked for training. The models were trained using these data with a set of suitable parameters, as shown in Figure 9(b).

Table 2: Parameters of the YOLO models training using Google Colab

Parameter	Value
Epochs	100
Optimizer	Adam_W
Batch size	16
Image size	(640,640)
Learning rate	0.0001-0.01)
Data split	(70,20,10) %

Figure 11. shows improvement in the training results, demonstrating a decrease in losses in the bounding box predictions for training and validation and a decrease in the losses in class predictions. The decreases were gradual, corresponding to an increase in the epoch until reaching the fewest training losses (approaching zero) when the epoch approached 100.

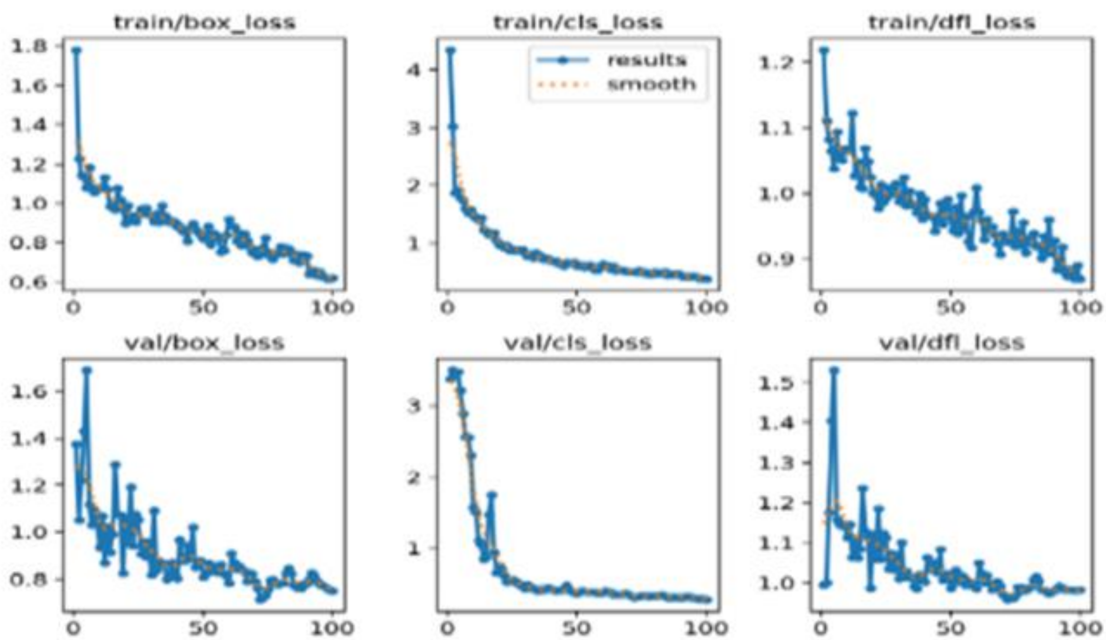

Figure 11: Decrease in training losses with increasing epochs

Figure 12. shows the improvement steps in accuracy, efficiency, and rate during the training stages, demonstrating their gradual arrival at the highest level. This depended on the increasing epoch until it reached 100. After the training was completed, the trained model was loaded to detect drones.

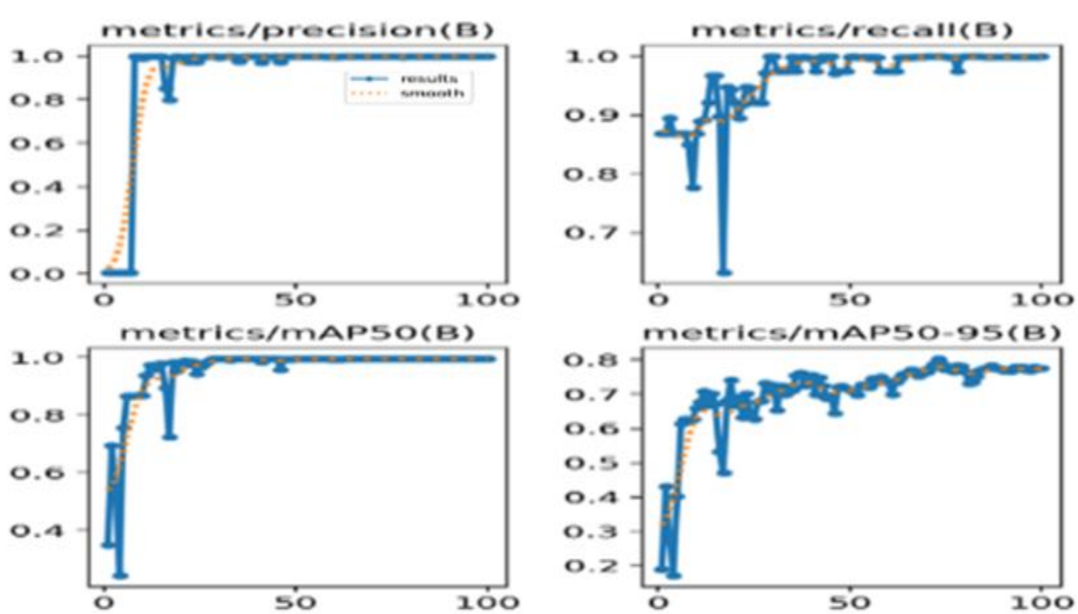

Figure 12: Maximum value of accuracy criteria with increasing epochs

The evaluation results, where the images were classified as test images (totaling: 234), were used to test the training model. This stage ensures the training model's ability to classify and detect drones and birds. Figure 13 (a) shows the confidence–recall curve, which has the best values when confidence is greater than 80%. The same is true of the confidence–precision curve shown in Figure 13 (b).

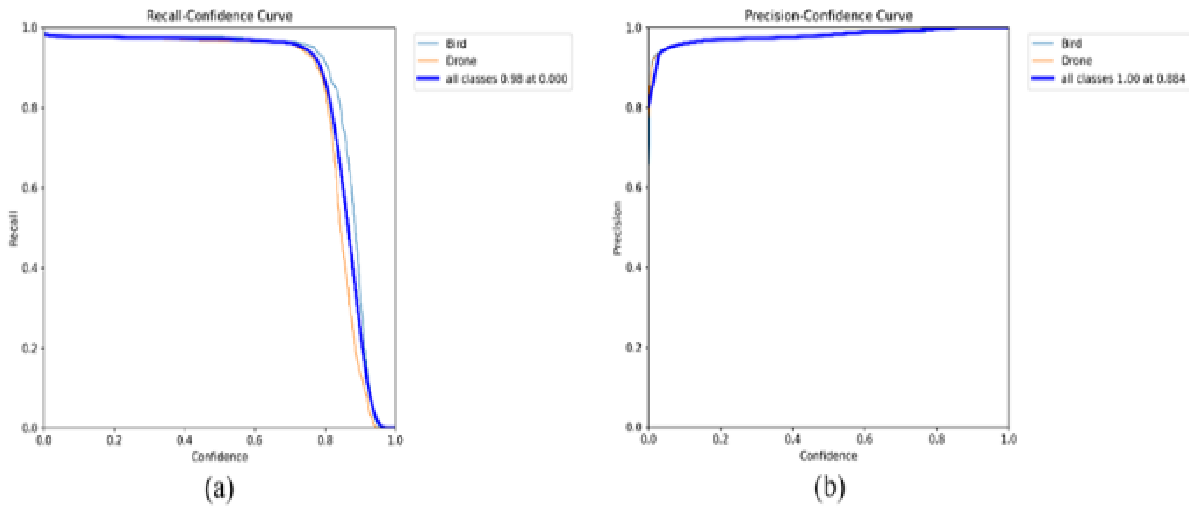

Figure 13: (a) Confidence–recall curve and (b) confidence–precision curve

5.3 Practical Detection in Real-time

The video surveillance data from the camera in the prototype is received and then converted into several frames of a suitable size. Each frame is predicted and detected by the YOLO11 algorithm and based on the weights loaded in the model during training in the previous step as shown in Figure 9 (c). The results demonstrate the detection accuracy, classes, and target location coordinates. A bounding box is drawn around the target, and all of these results are almost in real-time due to the speed of the YOLO11 algorithm. Figure 14 shows the practical detection results in real-time.

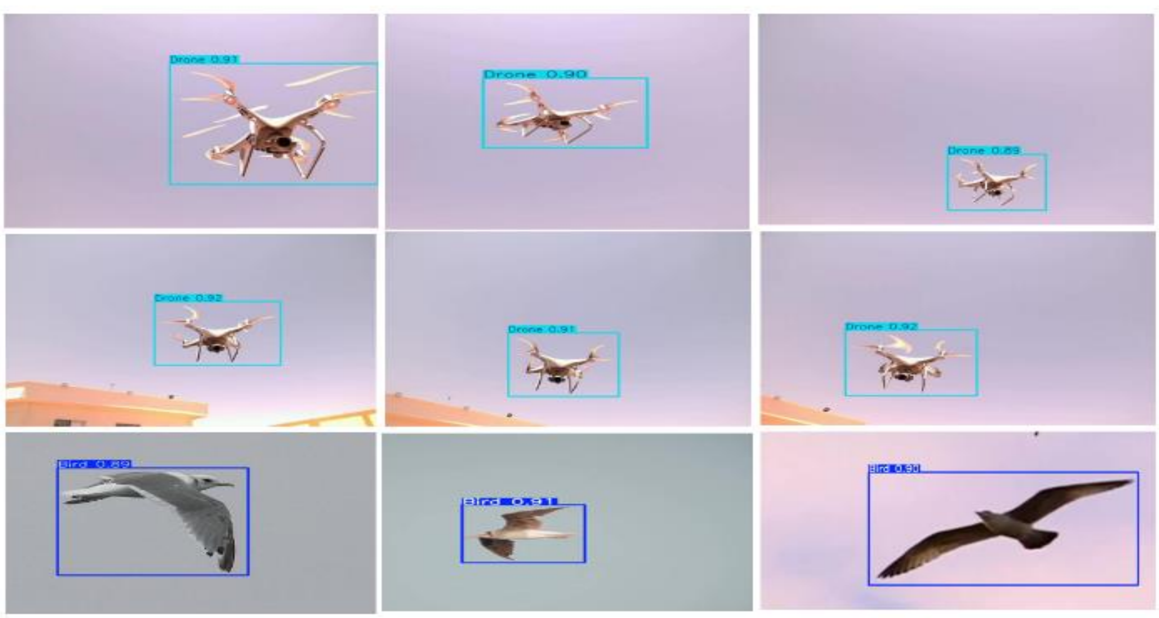

Figure 14: Practical detection results in real-time

The confidence matrix in Figure 15 shows the detection results ratios. A practical test was carried out; a drone and birds were displayed in front of the surveillance camera, with (3) minutes for the drone and (1) minute for the birds. The results were based on the detection ratio of the frames resulting from dividing the surveillance video. The detection-and-classification ratio for the drone was 98%, the classification-and-detection ratio for the birds was 98%, and the background ratio without any objects was 100%. These results demonstrate the success and high accuracy of the training process, which will be discussed in the following sections.

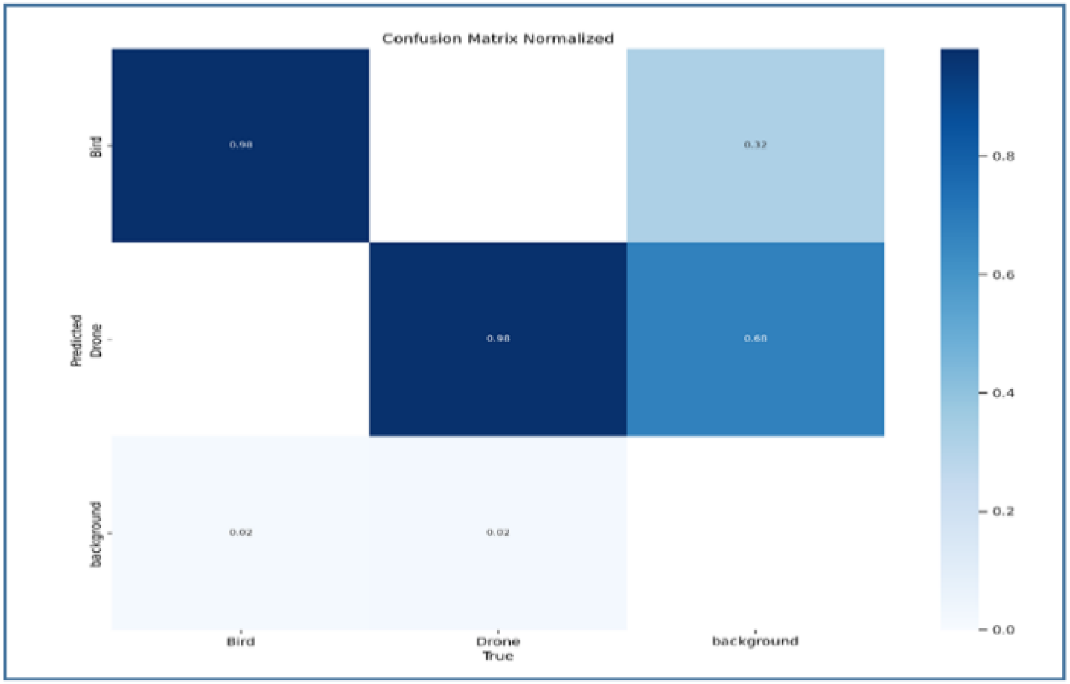

Figure 15: Normalized confidence matrix for the detection results

5.4 Tracking and Destroying the Unknown Drone

Through extensive experimentation and real-world system implementation, the researcher developed a novel targeting theory that bridges the gap between 2D visual information and 3D physical weapon actuation. This theory was derived from empirical observation and refined through iterative testing in semi-controlled environments. It represents a significant contribution to the field of autonomous defense systems, particularly those based on computer vision. The theory introduces a proportional relationship between the displacement of a drone in the image frame (measured in pixels) and the angular displacement required by the weapon system (measured in degrees) to align with the target. This simple yet powerful principle enables precise targeting using only camera data, without reliance on other external sensors.

In this theory, the center of the image frame is adopted as the primary reference point for measuring visual displacement. The center of the frame corresponds to the optical axis of the camera, which in turn is aligned with the aiming axis of the firearm, both fixed along the same line in the proposed system. This geometric alignment provides faster mechanical response and higher angular calculation accuracy, Figure 16 shows the positioning plan of firearm, and camera toured the detection drone. Thus, using the frame center as a reference ensures more accurate and stable conversion between the two-dimensional visual space and the three-dimensional guidance space. It also simplifies the computational process required for real-time targeting and reduces overall system complexity.

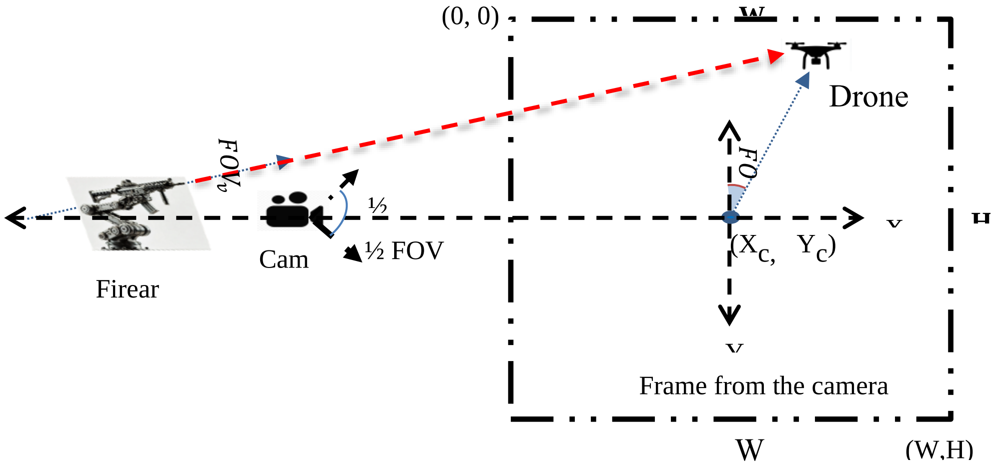

Figure 16: Shows the positioning plan of firearm, and camera toured the detection drone

The Theoretical Formulation: The detection data produced by the YOLO algorithm consists of the coordinates of the top-left and bottom-right points of the bounding box surrounding the detected drone, as illustrated in the Figure 17.

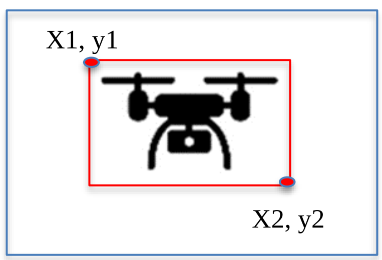

Figure 17: Shows data produced by the YOLO, which consists of the coordinates of the bounding box

Let (Xc, Yc) are the center coordinates of the frame:

$$(\mathrm{Xc}, \mathrm{Yc}) = \left(\frac{\mathrm{W}}{2}, \quad \frac{\mathrm{H}}{2}\right) \tag{5}$$

Where:

W and H are the width and height of the frame. The center of drone is (x, y) in pixel:

$$(\mathrm{x}, \mathrm{y}) = \left(\frac{\mathrm{x1} + \mathrm{x2}}{2}, \quad \frac{\mathrm{y1} + \mathrm{y2}}{2}\right) \tag{6}$$

The theory is that the relative displacement of a target in the image frame is directly proportional to the angular displacement of the camera's field of view from its central axis. Figure 17 illustrates the locations of the proposed system components, showing that the frame center, camera field of view center, and firing arm center are all located on a uniform horizontal line. It also includes an explanation of the drone's location and the angles required to access it. It can be expressed that the rate of displacement of the drone from the center of the image frame is equal to the rate of angular displacement of the camera's line of sight from its central axis. The mathematical relationship is formulated as follows:

$$\frac{\Delta P}{\mathrm{P}} = \frac{\Delta FOV}{\mathrm{FOV}} \quad (7)$$

Where: ΔP is the pixel displacement of the drone from the center of the image frame, and P is the total number of pixels in the frame axis (horizontal or vertical). ΔFOV: required change in weapon angle to align with the drone, and FOV is the camera field of view in degrees for the corresponding axis (horizontal or vertical). There for vertical position, it became:

$$\frac{\Delta y}{\mathrm{H}} = \frac{\Delta FOV_v}{\mathrm{FOV_v}} \quad (8)$$

Where:

Δy is the pixel displacement of the drone from the center of the image frame in Y-axes, and H is the high of the frame. ΔFOVv is the required change in weapon angle to align with the drone, and FOV_v is the camera field of view in degrees for the vertical axis. Then the vertical form became:

$$\frac{y - Yc}{H} = \frac{\Delta FOV_v}{FOV_v} \quad (9)$$

$$\frac{y - H/2}{H} = \frac{\Delta FOV_v}{FOV_v} \quad (10)$$

$$\frac{y}{H} - \frac{1}{2} = \frac{\Delta FOV_v}{FOV_v} \quad (11)$$

$$\Delta FOV_v = \frac{FOV_v(2y - H)}{2H} \quad (12)$$

Similarly, for the horizontal direction, the angle can be computed as:

$$\Delta FOV_h = \frac{FOV_h(2x - W)}{2W} \quad (13)$$

Where:

x is the pixel position of the drone on the image frame in the X-axis. ΔFOV_h Is the required change in horizontal weapon angle to align with the drone, and FOV_h is the camera field of view in degrees for the horizontal axis.

The amount of vertical and horizontal adjustment required in the firearm's orientation to align with the drone corresponds to the angular values that must be added to or subtracted from the central angle of the camera's field of view. Therefore, the final aiming angle (Θ, Ø) of the firearm toward the drone can be expressed as follows:

$$\Theta = \tfrac{1}{2} FOV_h + \Delta FOV_h \quad (14)$$

$$\Theta = \frac{1}{2} FOV_h + \frac{FOV_h(2x - W)}{2W} \quad (15)$$

The firearm horizontal angle:

$$\Theta = FOV_h \left(\frac{1}{2} + \frac{(2x - W)}{2W} \right) \quad (16)$$

Similarly, the firearm vertical angle:

$$\emptyset = FOV_v \left(\frac{1}{2} - \frac{(2y - H)}{2H} \right) \quad (17)$$

This theory represents a new scientific contribution by the researcher and is the outcome of original practical experimentation. It provides a lightweight, efficient, and cost-effective alternative to traditional targeting mechanisms. Its advantages include:

- No need for triangulation or distance estimation
- No dependence on external sensors
- Real-time computation with minimal latency
- Simple mathematical mapping between visual and physical domains

It is particularly suitable for smart defense systems deployed in resource-limited environments or where stealth and simplicity are paramount. Future extensions of this theory may include multi-axis compensation, adaptive FOV calibration, and swarm drone tracking applications.

6 Legal and Ethical Considerations

The proposed anti-drone system is technically capable of real-time detection and autonomous destruction of unauthorized drones; however, the important aspects of legality and ethics must be examined in civilian contexts. The use of autonomous weapon systems raises challenging considerations particularly under international humanitarian law (IHL), privacy legislation, and public safety regulations. An autonomous weapon system that autonomously destroys aerial targets without verifying with a human operator, risks violating property rights, endangering civilians, and mistaking non - hostile objects for threats. Therefore, prior to deploying a system such as this in the real world, the system must include a reliable human-in-the-loop verification step prior to destruction.

Furthermore, the use of firearms or automated kinetic countermeasures in urban or populated areas must comply with national regulations on weapons use, airspace control, and data protection laws. Failure to adhere to these frameworks can result in serious legal consequences and ethical controversies. Accordingly, while the current work serves as a proof-of-concept for technological feasibility, future implementations must incorporate legal review, ethical oversight, and built-in safeguards to ensure responsible and lawful deployment.

7 Results Discussion

All the requirements of the proposed system have been fulfilled and practical testing and experimentation of the trained models YOLO (v8, v9, v10, v11) have been conducted on the data explained in Section 4 as shown in Table 3. In this study, four models were trained on the most popular versions of the YOLO algorithm and the same training environment and same dataset .This was done to identify the most suitable version for the success of the proposed project (anti-drone system), which requires a high detection speed close to real-time detection, while maintaining the highest detection accuracy to ensure the proposed project meets the desired purpose. Table 3 includes the test results of the trained models, showing that YOLOv11 outperforms the other versions in terms of accuracy, which reaches (mAP-0.5: 0.911, Recall: 0.915, Precision: 0.957), the lowest number of training layers (238), and the fewest parameters (2169701). Since the primary purpose of designing the proposed project is to protect against the threat or drone attacks, the detection process must be at real-time. The number of detected frames per second measures the detection speed. The results obtained from this study show that YOLO 11 is the fastest detection algorithm, with the number of frames per second (FPS: 29.6 f/s), indicating that it is suitable for real-time drone detection.

Table 3: Comparison of the results models of YOLO v (8, 9, 10, 11)

Models	mAP-0.5	mAP 0.5-0.95	Precision	Recall	Layers	FPS	Model Size (MB)	Parameters
YOLOv8n	0.844	0.542	0.932	0.81	168	24.7	6.3	3,005,833
YOLOv9n	0.892	0.526	0.921	0.832	486	20.3	4.7	2,921,036
YOLOv10n	0.813	0.5	0.933	0.854	285	27.0	5.9	2,754,321
YOLOv11n	0.911	0.556	0.957	0.915	238	29.6	5.4	2,169,701

8 Conclusion

This study used computer vision to design an anti-drone system to protect important places from unknown drone attacks. A prototype of the proposed system was practically designed using the YOLO 11 algorithm to detect unknown drones in real-time. Images were collected and classified using the Roboflow server, and the model was trained using the Google Colab server. A practical experiment was conducted to detect a drone: the detection results were high (precision: 95.7%, recall: 91.5%, and mAP50: 91.1%, and FPS:29.6) and superior to other studies. After the drone detection process, an audio alarm is issued, and the target coordinates are extracted and sent to a robotic arm carrying a weapon in real-time to track and destroy the enemy drone. We faced a challenge in synchronizing the tracking of the moving drone and converting the coordinates from a pixel system into angles in degrees. Thus, we designed a new mathematical method that provides the angles of the weapon arm's movement as a line of sight toward the detection drone. This method was subsequently tested, and the accuracy was very high.

In future work, we plan to expand testing to include a broader range of drone models and flying scenarios (e.g., different altitudes, lighting conditions, and backgrounds) to further evaluate the robustness of the proposed system. Due to current financial and logistical constraints, such extended testing was not feasible in the present study.

References

[1] AbdAli, J. K., & Maaroof, M. K. A. (2022). Theoretical Investigations of Wing Coating on the Performance of UAV. *International Journal of Advances in Engineering and Emerging Technology (IJAEET) Vol*, *13*.

[2] Alnajjar, I. A., Almazaydeh, L., Odeh, A. A., Salameh, A. A., Alqarni, K., & Ban Atta, A. A. (2024). Anomaly Detection Based on Hierarchical Federated Learning with Edge-Enabled Object Detection for Surveillance Systems in Industry 4.0 Scenario. *International Journal of Intelligent Engineering & Systems*, *17*(4). https://doi.org/10.22266/ijies2024.0831.49

[3] Al-Qubaydhi, N., Alenezi, A., Alanazi, T., Senyor, A., Alanezi, N., Alotaibi, B., Alotaibi, M., Razaque, A., Abdelhamid, A. A., & Alotaibi, A. (2022). Detection of Unauthorized Unmanned Aerial Vehicles Using YOLOv5 and Transfer Learning. *Electronics*, *11*(17), 2669. https://doi.org/10.3390/electronics11172669

[4] Behera, D. K., & Raj, A. B. (2020, May). Drone detection and classification using deep learning. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 1012-1016). IEEE. https://doi.org/10.1109/ICICCS48265.2020.9121150

[5] Carranza-García, M., Torres-Mateo, J., Lara-Benítez, P., & García-Gutiérrez, J. (2020). On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data. *Remote Sensing*, *13*(1), 89. https://doi.org/10.3390/rs13010089

[6] Cheng, B., Wei, Y., Shi, H., Feris, R., Xiong, J., & Huang, T. (2018). Revisiting rcnn: On awakening the classification power of faster rcnn. In *Proceedings of the European conference on computer vision (ECCV)* (pp. 453-468).

[7] Dadrass Javan, F., Samadzadegan, F., Gholamshahi, M., & Ashatari Mahini, F. (2022). A modified YOLOv4 Deep Learning Network for vision-based UAV recognition. *Drones*, *6*(7), 160. https://doi.org/10.3390/drones6070160

[8] Fujii, S., Akita, K., & Ukita, N. (2021, July). Distant bird detection for safe drone flight and its dataset. In *2021 17th international conference on machine vision and applications (MVA)* (pp. 1-5). IEEE. https://doi.org/10.23919/MVA51890.2021.9511386

[9] Guru Prasad, S., & Badrinarayanan, M. K. (2025). A study on the adoption of threat prevention and dark web monitoring for information security management in India. *Indian Journal of Information Sources and Services*, *15*(2), 154–159. https://doi.org/10.51983/ijiss-2025.IJISS.15.2.21

[10] Hakani, R., & Rawat, A. (2024). Edge computing-driven real-time drone detection using YOLOv9 and NVIDIA Jetson nano. *Drones*, *8*(11), 680. https://doi.org/10.3390/drones8110680

[11] Hakani, R., Prajapati, S., & Rawat, A. (2024). Optimizing UAV Detection Performance with YOLOv5 Series Algorithms. *Int. J. Microsyst. IoT*, *2*, 991-995. https://doi.org/10.5281/zenodo.132936

[12] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961-2969).

[13] Hendrawan, A., Gernowo, R., Nurhayati, O. D., & Dewi, C. (2024). A Novel YOLO-ARIA Approach for Real-Time Vehicle Detection and Classification in Urban Traffic. *International Journal of Intelligent Engineering & Systems*, *17*(1). https://doi.org/10.22266/ijies2024.0229.38

[14] Jegham, N., Koh, C. Y., Abdelatti, M., & Hendawi, A. (2024). Evaluating the evolution of yolo (you only look once) models: A comprehensive benchmark study of yolo11 and its predecessors. *arXiv e-prints*, arXiv-2411. https://arxiv.org/abs/2411.00201

[15] Kavitha, M. (2024). Embedded system architectures for autonomous vehicle navigation and control. *SCCTS Journal of Embedded Systems Design and Applications*, *1*(1), 25-28. https://doi.org/10.31838/ESA/01.01.06

[16] Khanam, R., & Hussain, M. (2024). Yolov11: An overview of the key architectural enhancements. *arXiv preprint arXiv:2410.17725*.

[17] Liu, B., & Luo, H. (2022). An improved Yolov5 for multi-rotor UAV detection. *Electronics*, *11*(15), 2330. https://doi.org/10.3390/electronics11152330

[18] Maaroof, M. K. A., & Bouhlel, M. S. (2025). Drone Image Localization by Faster R-CNN Algorithm and Detection Accuracy. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, *16*(1), 172-189. https://doi.org/10.58346/JOWUA.2025.I1.010

[19] Mahmoud, A., Mohamed, S., El-Khoribi, R., & AbdelSalam, H. (2020). Object detection using adaptive mask RCNN in optical remote sensing images. *Int. J. Intell. Eng. Syst*, *13*(1), 65-76. https://doi.org/10.22266/ijies2020.0229.07

[20] Najafi, J., Mirzakuchaki, S., & Shamaghdari, S. (2025). Autonomous drone detection and classification using computer vision and prony algorithm-based frequency feature extraction. *Journal of Intelligent & Robotic Systems*, *111*(1), 8. https://doi.org/10.1007/s10846-024-02216-x

[21] Nguyen, P. T., Nguyen, G. L., & Bui, D. D. (2025). LW-UAV–YOLOv10: A lightweight model for small UAV detection on infrared data based on YOLOv10. *Geomatica*, *77*(1), 100049. https://doi.org/10.1016/j.geomat.2025.100049

[22] Popescu, L. R. (2021). The existing technologies on anti-drone systems. In *International conference knowledge-based organization* (Vol. 27, No. 3, pp. 83-91). https://doi.org/10.2478/kbo-2021-0093

[23] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

[24] Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, *39*(6), 1137-1149. https://doi.org/10.1109/TPAMI.2016.2577031

[25] Samadzadegan, F., Dadrass Javan, F., Ashtari Mahini, F., & Gholamshahi, M. (2022). Detection and recognition of drones based on a deep convolutional neural network using visible imagery. *Aerospace*, *9*(1), 31. https://doi.org/10.3390/aerospace9010031

[26] Sapkota, R., Meng, Z., & Karkee, M. (2024). Synthetic meets authentic: Leveraging llm generated datasets for yolo11 and yolov10-based apple detection through machine vision sensors. *Smart Agricultural Technology*, *9*, 100614. https://doi.org/10.1016/j.atech.2024.100614

[27] Sapkota, R., & Karkee, M. (2024). Comparing YOLOv11 and YOLOv8 for instance segmentation of occluded and non-occluded immature green fruits in complex orchard environment. *arXiv preprint arXiv:2410.19869*. https://doi.org/10.48550/arXiv.2410.19869

[28] Sawah, M. S., Taie, S. A., Ibrahim, M. H., & Hussein, S. A. (2023). Deep Neural Network-based Approach for Accurate Vehicle Counting. *International Journal of Intelligent Engineering & Systems*, *16*(2). https://doi.org/10.22266/ijies2023.0430.21

[29] Sharma, A., Kumar, V., & Longchamps, L. (2024). Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species. *Smart Agricultural Technology*, *9*, 100648. https://doi.org/10.1016/j.atech.2024.100648

[30] Shi, Q., & Li, J. (2020, October). Objects detection of UAV for anti-UAV based on YOLOv4. In *2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT* (pp. 1048-1052). IEEE. https://doi.org/10.1109/ICCASIT50869.2020.9368788

[31] Shi, X., Yang, C., Xie, W., Liang, C., Shi, Z., & Chen, J. (2018). Anti-drone system with multiple surveillance technologies: Architecture, implementation, and challenges. *IEEE Communications Magazine*, *56*(4), 68-74. https://doi.org/10.1109/mcom.2018.1700430

[32] Singh, T., & Nair, A. (2022). Reinforcement learning model-oriented autonomous drone navigation and its applications. *International Academic Journal of Science and Engineering*, *9*(3), 21–25.

[33] Srimuang, c., & doungmala, P. (2023). Autonomous flying drones: Agricultural supporting equipment. *International Journal of communication and computer Technologies*, *11*, 7-12.

[34] Suewongsuwan, K., Angsuseranee, N., Wongkamchang, P., & Phasinam, K. (2024). Comparative analysis of UAV detection and tracking performance: Evaluating YOLOv5, YOLOv8, and YOLOv8 DeepSORT for enhancing anti-UAV systems. *Edelweiss Applied Science and Technology*, *8*(5), 708–726. https://doi.org/10.55214/25768484.v8i5.1737

[35] Tu, X., Zhang, C., Zhuang, H., Liu, S., & Li, R. (2024). Fast drone detection with optimized feature capture and modeling algorithms. *IEEE Access, 12,* 108374–108388. https://doi.org/10.1109/access.2024.3438991

[36] Wang, X., Li, W., Guo, W., & Cao, K. (2021, April). SPB-YOLO: An efficient real-time detector for unmanned aerial vehicle images. In *2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)* (pp. 099-104). IEEE. https://doi.org/10.1109/ICAIIC51459.2021.9415214

[37] Wisniewski, M., Rana, Z. A., & Petrunin, I. (2022). Drone model classification using convolutional neural network trained on synthetic data. *Journal of Imaging*, *8*(8), 218. https://doi.org/10.3390/jimaging8080218

[38] Yilmaz, B., & Kutbay, U. (2024). YOLOv8-based drone detection: Performance analysis and optimization. *Computers*, *13*(9). https://doi.org/10.3390/computers13090234

[39] Zhao, Y., Wang, L., Lei, G., Guo, C., & Ma, Q. (2024). Lightweight UAV small target detection and perception based on improved YOLOv8-E. *Drones*, *8*(11), 681. https://doi.org/10.3390/drones8110681

Authors Biography

Nabeel Aad Lafta is a Ph.D. student in the Department of Electrical Engineering, College of Engineering, at the University of Babylon, Iraq. He holds a Master's degree in Electronics and Communications Engineering from the same institution. Her research interests include a wide range of topics in electronics and telecommunications, particularly in the areas of signal processing, embedded systems, and wireless communication.

Saad Saffah Hreshee is a professor in the Department of Electrical Engineering at the College of Engineering, University of Babylon, Iraq. He received his professor Dr. in Electronic and Communication Engineering and has extensive academic and research experience in wireless communication, embedded systems, and digital signal processing. He has supervised numerous graduate students and contributed to several national and international research projects. His research interests include wireless sensor networks, IoT systems, and AI applications in communication systems.