Received 15 July 2023; revised 2 August 2023; accepted 25 August 2023. Date of publication 30 August 2023; date of current version 15 September 2023.

*Digital Object Identifier 10.1109/OJCOMS.2023.3310003*

# An Ensemble-Based IoT-Enabled Drones Detection Scheme for a Safe Community

**JASKARAN SINGH [1] (Student Member, IEEE), KESHAV SHARMA [1] (Student Member, IEEE), MOHAMMAD WAZID [1] (Senior Member, IEEE), ASHOK KUMAR DAS [2] (Senior Member, IEEE), AND ATHANASIOS V. VASILAKOS [3] (Senior Member, IEEE)**

[1]Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun 248002, India
[2]Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500032, India
[3]Center for AI Research, University of Agder, 4879 Grimstad, Norway

CORRESPONDING AUTHOR: A. V. VASILAKOS (e-mail: thanos.vasilakos@uia.no)

**ABSTRACT** With the increasing use of Internet of Things (IoT)-enabled drones for various purposes, including photography, delivery, and surveillance, concerns related to privacy and security have arisen. Drones have the potential to capture sensitive information, invade privacy, and cause security breaches. Therefore, the need for advanced technology for the automated detection of drones has become crucial. In this paper, we propose an ensemble-based IoT-enabled drones detection scheme (in short, EDDSBS). The presented model is part of a computer vision-based module and uses transfer learning for improved performance. Transfer learning allows the reuse of pre-trained models and their knowledge in a different but related domain, enabling better performance with less training data. To evaluate the performance of the proposed EDDSBS, we test it on benchmark datasets, including the Drone–vs–Bird Dataset and the UAVDT dataset. The proposed EDDSBS outperforms the existing schemes of drone detection (i.e., in terms of accuracy). The results of the presented scheme demonstrate the potential of deep learning-based technology for automated drone detection in critical areas, such as airports, military bases, and other high-security areas. Thus the paper introduces a comprehensive process methodology for drone detection that can be applied in real-world settings for a sustainable and secure environment, which is required for a safe community.

**INDEX TERMS** Internet of Things (IoT), drone detection, public safety, ensemble models, security and surveillance.

## I. INTRODUCTION

A SOCIETY that is safe and sustainable is an ecosystem in which the human, natural, and economic components are dependent on one another and derive their vitality from one another. Moreover, the subjects (i.e., human) feel safe and secure in that environment as they get protection from any kind of threats. In this particular environment, safety and sustainability remain for a long time [1].

The term "Internet of Things (IoT)" refers to actual physical things that include sensors, processing power, software, and other technologies, which can communicate to other systems and devices over the Internet to exchange their data [2]. The different applications of IoT are smart home automation, smart manufacturing, smart farming, intelligent transportation system, security and surveillance, smart healthcare, and many more [2], [3]. There are different variants of IoT, such as the Internet of Drones (IoD), Internet of Vehicles (IoV), Internet of Medical Things (IoMT), Industrial Internet of Things (IIoT), and Internet of Battlefield Things (IoBT) [4], [5].

With the advances of drones that are advanced and affordable for a wide range of consumers [6], [7], the field of aerial technology has revolutionized at a very rapid rate. Internet of Drones (IoD), also known as IoT-enabled drones, is an architectural framework that was created to facilitate rapid and secure communication between drones and users using the Internet as a medium [8], [9]. However, this progress has given rise to various challenges and problems, with the

major issues related to privacy concerns. This availability of low-cost IoT-enabled drones that are equipped with enhanced and high-powered cameras has made it easier for adversaries to spy and capture images and videos of others without their consent the [10]. This rapidly increasing misuse of drones, particularly in sensitive areas such as the defence sector, has caused rapid concerns [11], [12]. To address them, there is a growing need for IoT-driven automated drone detection systems using computer vision that can provide security and alert the authorities when drones venture into restricted airspace [13].

Machine learning (ML) has demonstrated its impact in multiple fields with its innovative capability in analysis and automation. ML in healthcare has been integrated into the Internet of Medical Things ecosystem for the diagnosis of diseases and developing personalized treatment plans [14]. ML in the finance sector is extensively being used for automated fraud detection and market trend analysis [15]. In the manufacturing sector, ML algorithms help to increase efficiency by predicting the failure of equipment and streamlining production workflow [16]. Lastly, In the transportation industry, the inclusion of self-driving cars significantly relies on optimized computer vision models to drive and navigate safely in a real-time environment [17].

Computer vision is a subset of the artificial intelligence domain that deals with the development of algorithms that help train computers to understand visual data for extracting information and patterns. With increasingly powerful machines and advanced ML techniques, computer vision is being continuously applied in a variety of fields like robotics, facial recognition, medical diagnosis, and enhanced surveillance [18], [19]. Transfer learning is a critical methodology in computer vision, enabling computers to utilize already trained models to be adapted for different tasks [20]. Transfer learning has helped in developing extremely efficient object detection models, particularly in areas with less availability of data, including drone detection.

Ensemble models in computer vision is a paradigm that combines multiple deep learning models to build a robust and accurate classifier [21], [22]. This technique involves training multiple models with the same or different architecture and layers that are then connected to each other to incorporate a single output. By capturing different aspects due to different architecture, these models are better in performance, notably by reducing biases and limitations that are prevalent in single models. In addition, ensemble models can help to improve generalization by reducing overfitting and providing a more stable prediction. Overall, ensemble models are a powerful technique in deep learning that can significantly improve the performance of machine learning models.

In the proposed scheme, the concepts of the Internet of Things (IoT) and deep learning are used. There is a Raspberry Pi, which acts as the central control server for an alarm detection module. The deployed cameras are equipped with the features of IoT and ensemble models of deep learning. The Raspberry Pi is equipped with a camera and is responsible for capturing and extracting images locally. These images are then sent to our Flask REST API, present in the cloud server for further processing and prediction. Once a drone is predicted, the Raspberry Pi triggered the alarm system, consisting of multiple buzzers and speakers, to alert the authorities of the drone's presence. To ensure timely action, we also connect the Raspberry Pi to Gmail SMTP Access to receive email notifications of drone detection. By integrating the Raspberry Pi with the alarm system and cloud-based API, we create an efficient and effective monitoring system to detect and respond to unauthorized drone activity in real-time.

### A. RESEARCH MOTIVATION

This work's motivation can be summarized as follows. Advanced Drones with affordable prices and advancing technology are being utilized in various domains. However, with this surge, they are increasingly posing a security threat, with a spike in use for criminal activities, including surveillance and smuggling. Therefore, there is a critical need for an automated drone detection process and prevention measures to handle these privacy concerns [23]. Computer vision envisioned techniques have shown promise in detecting drones from videos and CCTV footage, with advances towards real-time detection and monitoring. However, limitations in the algorithm are reflected in the number of false positives and false negatives, showing concerns about the robustness of the models. Ensemble models in deep learning have shown the potential to improve these baseline models' accuracy and robustness by reducing the impact of biases of the individual models. Hence it is crucial to continue research to develop effective drone detection schemes to counter these emerging threats [24], [25]. The proposed ensemble deep learning-based scheme for drone detection from videos aims to utilize transfer learning-based ensemble models and provide a global and complete approach for automated drone detection. The scheme's potential impact includes enhancing public safety and privacy and reducing the potential for the preachment of privacy through attacks using drones.

### B. RESEARCH CONTRIBUTIONS

The research contributions of this paper are given below:

- This paper presents an ensemble-based IoT-enabled drones detection scheme (named EDDSBS).
- To assess its impact on various performance parameters, a practical demonstration of the EDDSBS is performed.
- The paper introduces a comprehensive process methodology for drone detection that can be applied in real-world settings for a secure environment.
- In the comparative study, it is observed that the proposed EDDSBS performs better than the other existing schemes.

### C. ROADMAP OF THE PAPER

The remaining part of this article is structured as follows. An analysis and review of the related existing schemes

is given in Section II. The details of the proposed drone detection scheme (EDDSBS) are provided in Section III. Then, the practical implementation of the proposed EDDSBS is conducted in Section IV. Furthermore, the performance comparison of the proposed EDDSBS with the other similar existing schemes is made in Section V. Finally, the paper is concluded in Section VI with some concluding remarks and future research works.

## II. RELATED WORK

This section outlines the specifics of several currently implemented methods for detecting drones using vision-based techniques.

While radar-based methods are commonly used for detecting and tracking drones in airspace, some studies have been conducted on visual-based identification of drones. These studies typically involve using computer vision techniques to analyze video or image data and identify drones based on their appearance.

Rozantsev et al. [26] employed a Convolutional Neural Network (CNN) model and Histograms of Gradients for drone detection on a dataset of UAV and aircraft images. The study used a multi-scale sliding window technique to generate spatiotemporal cubes for detection. Samadzadegan et al. [27] presents a novel deep learning-based approach for efficient detection and recognition of drones. The method uses a CSPDarknet53 feature extraction network and monitors the IoU loss function to distinguish between drones and birds. The proposed approach can detect and differentiate between two types of drones as well as differentiate them from birds. Carrio et al. [28] presents a drone detection method that utilizes 6000 synthetic depth maps of drones and includes a 3D localization module for the collision-free deployment of drones. The method achieves an average detection rate of 74.7% with a detection distance of 9.5 meters. Lv et al. [29] proposes a background reduction module that is combined with drone detection using SAG-YOLOv5 models. The model's speed is increased by using SimAM's attention modules to reduce background and increase FPS. The method achieves a detection speed of 13.2 FPS for drone detection from high-resolution images under a fixed camera. Peng et al. [30] introduces a Physically Based Rendering toolkit for creating a synthetic dataset of drones with varying positions, orientations, camera specifications, backgrounds, and post-processing techniques. The method improves a Faster R-CNN model with training weights from Resnet-101 and achieves a precision of 80.69%.

Al-Qubaydhi et al. [31] utilizes an optimized version of YOLO, specifically YOLOv5, for drone detection in videos with diverse contrasts, including low contrast, using a dataset consisting of images of drones with various backgrounds such as water, buildings, trees, and humans. Seidaliyeva et al. [32] proposes a drone detection method that employs a background subtraction module and a CNN model for classification to enhance the model's robustness. This approach allows for the accurate detection of drones on a static background, with a processing speed significantly higher than existing approaches while maintaining comparable accuracy. Wang et al. [33] demonstrated a quick and efficient detection method for unmanned aerial vehicles (UAVS) that was based on video pictures recorded by stationary cameras. The technology saved money and cut down on operational costs. They identified moving items in the video by applying "temporal median background subtraction and then extracted global Fourier descriptors and local HOG features" from the moving object pictures. After that, the combined features were sent to the Support Vector Machine (SVM) classifier. So that they could be classified and recognized.

Fang et al. [23] looked at whether or not it would be possible to use a multistatic SDPR (MSDPR) for drone detection in practice. Analyses of signal processing processes involving multipath energy, extracted reference signal purity, and receiving antenna were utilized for the investigation of SDPR's detection range. Kang et al. [24] gathered the complete frequencies of leakage signals and the radiation pattern of a drone equipped with the GPS module while it was operating in an anechoic room so that they could conduct an analysis of the leakage that occurred from the GPS module while it was in use. They measured the leakage signals in an open-air environment using the collected data. It was determined through measurements taken in the open air that the theoretical attenuation effect was consistent with the measured value despite the fact that the distance varied. Delleji et al. [25] advised using a method of drone detection that included deep learning-based categorization and localization tasks in order to protect sensitive places and restricted areas. Particularly, they went with the YOLOv3 family of one-stage object detectors known for their speed and precision. Therefore, to better recognize small objects, such as small drones, they used the YOLOv3 deep learning neural network and worked to improve it. To do this, they upgraded the architecture of the network and fine-tuned its parameters. Reddy et al. [34] proposed a deep learning-based object recognition model, in which YOLOv3 was applied to a particular dataset in order to improve the speed and precision with which drones could be identified. An image classification technique based on a convolutional neural network was proposed by Chen et al. [35]. This algorithm would convert the data collected by cooperative spectrum sensing at a sensing slot into a single image. In addition, to use more information and enhance the effectiveness of the detection process, they developed an algorithm for trajectory classification. This algorithm transformed the flying process of the drones into trajectory photos using consecutive multiple sensing slots. Furthermore, they did simulations to validate the performance of the presented technique using various parameter settings.

Unlu et al. [36] used vision-based features - 2-dimensional scale, rotation, and translation invariant Generic Fourier Descriptor (GFD), which were utilized to distinguish drones from a dataset of birds by training a CNN model.

Brown et al. [37] used various models for the detection of UAVs. They obtained a dataset of various images featuring UAVs and used it to build a classification model based on ResNet-18, VGG-16, MobileNetV2, and AlexNet. During the classification process, they took into account the view angle and elevation as crucial factors in determining the model's detection performance. Xun et al. [38] developed a drone surveillance system that employed the YOLOv3 model with pre-trained weights. The model was trained on a custom dataset and tested and validated in real-time settings using an NVIDIA Jetson TX2 computing device. Shi and Li [39] employed a drone detection framework that utilized three models - YOLOv4, YOLOv3, and SSD (Single-Shot-Detector) - with a CSPDarknet53 backbone network structure to ensure lightweight models for real-time drone detection. The models were trained and tested on an augmented dataset that included images collected from the Internet and their own collected images.

## III. THE PROPOSED DRONE DETECTION SCHEME: EDDSBS

In this section, we provide the details of EDDSBS. The architectural representation of EDDSBS scheme is given in Figure 1. Figure 2 illustrates the flow of execution for the various processes involved.

- The process involves installing the application on the security server and authenticating the cloud server hosting the Ensemble Deep Learning Model. The next step is to install the background subtraction module on the security server machine and carry out an authentication process between the alarm system and the security server. Meanwhile, the security server registers the installed camera with a secured channel.
- The installed camera system sends video surveillance of the airspace to the security server for monitoring.
- The security server (Raspberry Pi) extracts image frames out of the live video, applies image processing techniques such as dehazing, denoising, and resolution enhancement.
- The enhanced frames are sent to the installed background subtraction module.
- The background subtraction module uses the pre-trained model to split the frame into the background and foreground, where the foreground contains the object of interest.
- The image with the removed background is sent to the cloud server hosting the ensemble Deep learning Model for further analysis and prediction by the Raspberry Pi through Flask API.
- If the ensemble model predicts an object as a drone, the IoT-Enabled alarm system is activated, notifying the user through sound alerts or email notifications using SMTP (SMTP) service of the Raspberry Pi.

The details of the proposed drone detection scheme (EDDSBS) are given in Algorithm 1. It works as follows. The detection process is executed for all flying drones. There is an installation of an application on the security server $SS_i$. Then, the application server authentication with the cloud server $CS_j$ is performed using some standard method. The $CS_j$ hosts ensemble deep learning model $EDL_{CS_j}$. Moreover, there is a deployment of alarm system $AS_k$. There is the installation of a background subtraction module on $SS_i$. Also, we perform an authentication process between $AS_k$ and $SS_i$. Again, $SS_i$ registers the cameras $CM_l$ before they are deployed. A drone $DR_r$ is flying randomly in the deployment area. $CM_l$ sends video recording $VD_{CM_l}$ of surveillance to $SS_i$ for monitoring. At this point, $SS_i$ extracts image frames from $VD_{CM_l}$ by applying image processing techniques such as dehazing, denoising, and resolution enhancement. After that, $SS_i$ sends the enhanced frames to the installed background subtraction module $BSM_m$. $BSM_m$ uses the pre-trained model to split the frame into the background and foreground. The image with the removed background is sent to $CS_j$ for further analysis and prediction. Here, $EDL_{CS_j}$ does the analysis. If $EDL_{CS_j}$ detects $DR_r$ as a drone, then $AS_k$ notifies authority through sound alerts or email notifications for proper action taking. Otherwise, it continues the detection process. At the end, $CS_j$ adds information about the detected drone $DR_r$ to the list of detected drones $L_{DR}$.

**Algorithm 1** Drone Detection Scheme (EDDSBS)

**Output:** List of detected drones $L_{DR}$

**for** All flying drones **do**
  Installation of application on security server $SS_i$.
  Authentication of application server with cloud server $CS_j$.
  $CS_j$ hosts ensemble deep learning model $EDL_{CS_j}$.
  Deployment of alarm system $AS_k$.
  Install background subtraction module on $SS_i$.
  Carry out an authentication process between $AS_k$ and $SS_i$.
  $SS_i$ registers camera $CM_l$. Then $CM_l$ is deployed.
  Drone $DR_r$ is flying randomly.
  $CM_l$ sends video recording $VD_{CM_l}$ of surveillance to $SS_i$ for monitoring.
  $SS_i$ extracts image frames from $VD_{CM_l}$ by applying image processing techniques such as dehazing, denoising, and resolution enhancement.
  $SS_i$ sends enhanced frames to installed background subtraction module $BSM_m$.
  $BSM_m$ uses pre-trained model to split the frame into the background and foreground.
  The image with the removed background is sent to $CS_j$ for further analysis and prediction.
  $EDL_{CS_j}$ does the analysis.
  **if** $EDL_{CS_j}$ detects $DR_r$ **then**
    $AS_k$ notifies authority through sound alerts or email notifications for proper action taking.
  **else**
    Continue detection process.
  **end if**
  $CS_j$ adds information of $DR_r$ in $L_{DR}$.
**end for**

### A. DATASET ACQUISITION AND PREPROCESSING

For the implementation of our proposed approach, we collected a hybrid dataset consisting of drone and bird images.

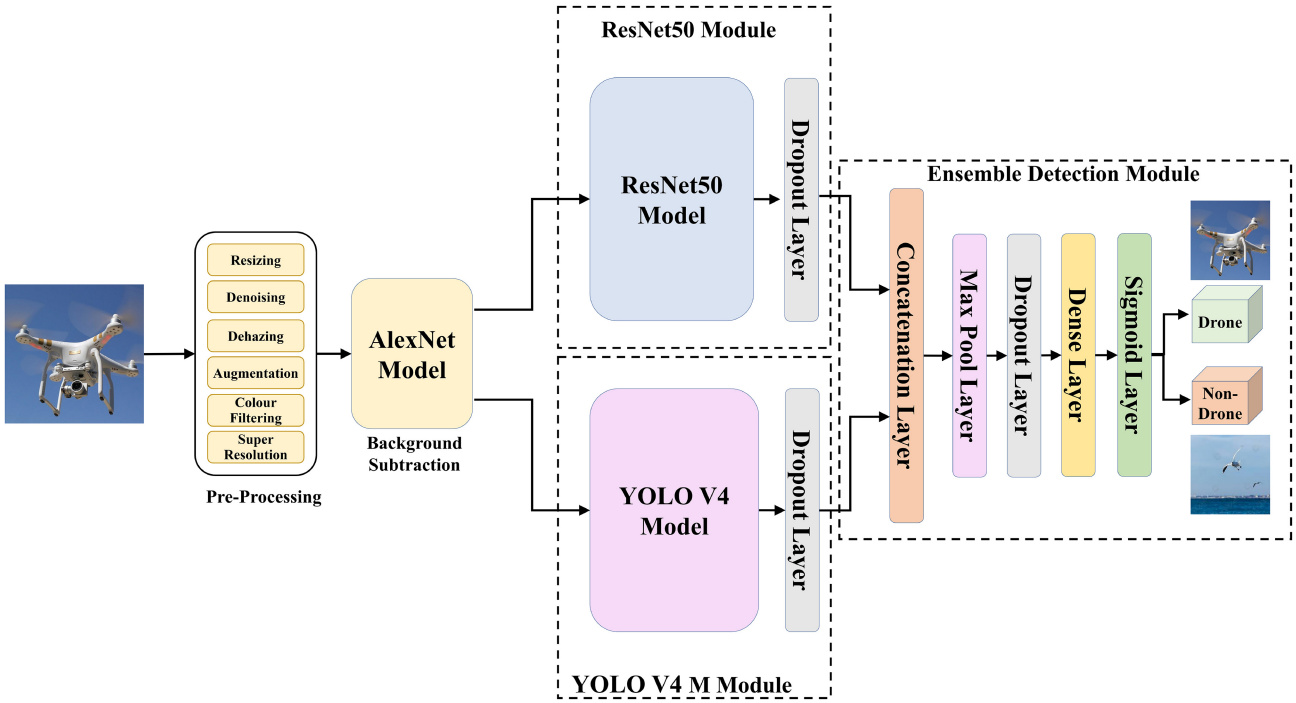


FIGURE 1. Architectural representation of the proposed EDDSBS.

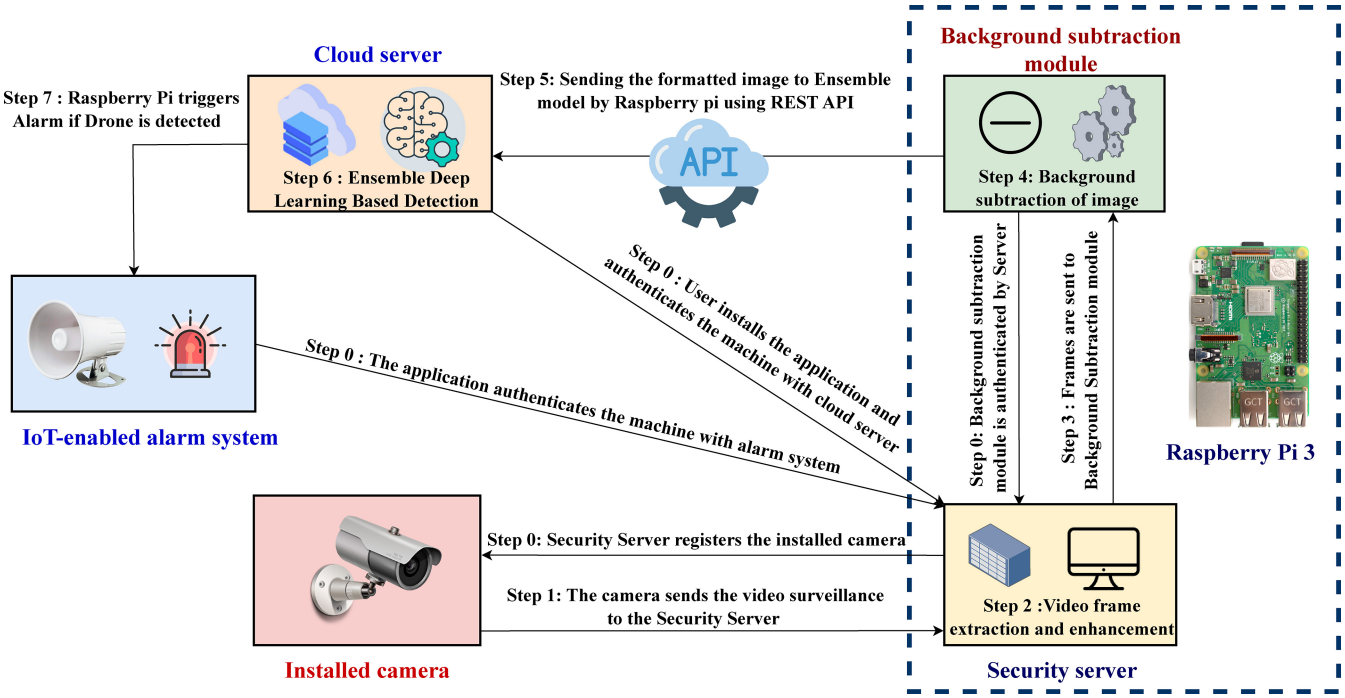


FIGURE 2. Process flow diagram of the proposed EDDSBS.

We utilized two publicly available datasets, namely Birds Vs Drone and UAV dataset, as well as gathered additional images by Web scraping. This comprehensive dataset comprises 3000 images, consisting of 1500 images of flying drones and 1500 images of birds, which will be used to test the classification accuracy of our model.

In order to make the image data suitable for further analysis, the first preprocessing step we implemented in this study was resizing. This was done to standardize the image size, making it easier to work with during subsequent stages of the study. Next, we utilized dehazing to remove any atmospheric haze or fog that may have been present in the images,

which could have affected the accuracy of subsequent analyses. We also applied denoising to remove any random noise that may have been present in the images, thus improving the overall clarity and detail. Finally, we applied color filtering to enhance specific colors or remove unwanted color casts. Overall, these preprocessing techniques were crucial in improving the overall quality of the image data, which ultimately aided in obtaining reliable and meaningful results from the subsequent analyses.

In addition to the previously mentioned preprocessing techniques, data augmentation techniques were also employed to increase the amount of training data available for the machine learning model. The data augmentation techniques we used included cropping, flipping, rotation, scaling, brightness, and contrast adjustments. Cropping involves selecting a specific region of an image and removing the rest to create a new variation. Flipping was used to create a mirror image of an image, which can often help the machine learning model learn more robust and invariant features. Rotation was applied to rotate the image by a certain degree, which can help the model recognize objects from different perspectives. Scaling was used to resize the image to a larger or smaller size, creating a new variation with different image dimensions. Brightness and contrast adjustments were applied to alter the brightness and contrast of the image, which can help the model learn to recognize objects under varying lighting conditions. Overall, applying these data augmentation techniques generated new and diverse variations of the original images and increased the training data by a factor of five, improving the model's robustness and performance on unseen test data.

The dataset was split into two subsets, one for training the model and the other for testing its classification accuracy. The training subset consisted of 75% of the images (1125 drone images and 1125 bird images), while the testing subset contained the remaining 25% (375 drone images and 375 bird images). The dataset was balanced to ensure that there was an equal number of images in each class.

The image processing procedure is explained in Algorithm 2. It executes for all the given images $SS_i$. Then, the deployed mechanism resizes the images to the standard size for easy analysis. It also applies dehazing to remove atmospheric haze or fog. Furthermore, it applies denoising to remove random noise and improve clarity. Again, it applies color filtering to enhance specific colors or remove unwanted casts. In addition, it does cropping of images to select specific regions and create new variations. After that, it applies the flipping of images to create mirror images for learning robust features. Next, it performs the rotation of images by certain degrees to recognize objects from different perspectives, and also perform the adjustment of brightness and contrast of images to learn object recognition under varying lighting conditions. It generates new and diverse variations of the original images to improve the model's robustness and performance on unseen test data. Finally, the dataset is splitted into training and testing subsets.

**Algorithm 2** Image Processing Procedure

**Output:** Preprocessed and augmented image dataset

**for** all given images $SS_i$ **do**
  Do resize of images to the standard size for ease of analysis.
  Apply dehazing to remove atmospheric haze or fog.
  Apply denoising to remove random noise as well as improve clarity.
  Apply color filtering to enhance specific colors or remove unwanted casts.
  Do cropping of images to select specific regions and create new variations.
  Apply the flipping of images to create mirror images for learning robust features.
  Do rotate images by certain degrees for recognizing objects from different perspectives.
  Do scaling of images to larger or smaller sizes create new variations with different dimensions?
  Do adjusting of brightness and contrast of images to learn object recognition under varying lighting conditions.
  Generate new and diverse variations of the original images to improve the model's robustness and performance on unseen test data.
  Split dataset into training and testing subsets.
**end for**

### B. BACKGROUND SUBTRACTION

To perform background subtraction of drone images using a pre-trained AlexNet [40], we first loaded the AlexNet model and replaced the last fully connected layer with a new layer that has only two output neurons, one for the foreground class and one for the background class. Then, we froze the weights of all layers except for the last layer so that only the new layer is trained on the drone dataset. Next, we collected a dataset of drone images and labeled them as either foreground or background, with the background being the areas of the image that do not contain the drone. We then trained the modified AlexNet model on this labeled dataset using a Stochastic Gradient Descent optimizer and binary cross-entropy loss function.

Once the model was trained, we used it as an inference model to perform this subtraction on new drone input images by passing the images through the trained model and extracting the foreground pixels as an output. The resulting foreground pixels thus represent the drone, while the background pixels represent the unnecessary background behind the drone that would not be considered for detection and identification. This approach was quite useful in our drone detection scheme as it allows for accurate segmentation and detection of the drone from the background, helping us in tracking and monitoring only the drone's movements and neglecting the background.

### C. ENSEMBLE MODEL

Resnet50 [41] is a 50-layer deep convolutional neural that has found intensive use in image classification and object detection tasks. It incorporates multiple techniques, including its bottleneck design and skip connections, which help to

minimize the problem of vanishing gradients by allowing the propagation of gradients. It provides high accuracy and allows to build of more dense layers, and thus has been used extensively in complex problems in medical image analysis and facial recognition.

YOLOv4, (You Only Look Once version 4) [42], is a well-known state-of-the-art object detection model that can efficiently detect objects in a real-time environment. It uses a one-stage detection algorithm as its architectural methodology, enabling it to predict the bounding boxes of objects in a single pass, along with classification into different object classes with prediction probabilities. YOLOv4, along with its multiple versions, is well renowned for its speed and accuracy and finds usage in multiple use cases, including autonomous vehicles and surveillance systems.

To build the proposed model, we utilized a transfer learning-based approach to create an ensemble model consisting of YOLOv4 and ResNet50 models as the solo components. The input image sent from the background subtraction module is duplicated and sent to both the ResNet50 and YOLOv4 models. The output from each of these models is then passed through a dropout layer, and the resulting two feature vectors are concatenated using a concatenation layer. This concatenated feature vector is then further reduced in dimensionality by a max-pooling layer and sent to a final dropout and dense layer. A sigmoid activation function is applied to this output layer to classify the image as a drone or non-drone.

The procedure for training the model is explained in Algorithm 3. It first loads the augmented training data, and then reads through each image in the training data. Moreover, background subtraction is applied to separate the image's foreground and background. If the foreground is empty, it goes to the next image. Otherwise, it continues with the process. It then sends the foreground image to both the Resnet50 and YoloV4 models, and obtains the output features from both models. After that, it sends each output to a dropout layer, and concatenates the output features from both dropout layers using the concatenation layer. Furthermore, it passes the concatenated feature vectors through the max-pooling and dense layers. After that, it passes the output of the dense layer through the sigmoid layer for classification. At this stage, there is a calculation of the loss between the predicted output and the true label. It again uses back-propagation to update the weights of the model. If the condition: $P_{ACC} < \Theta_{ACC}$ is met, it goes to Step 2 in Algorithm 3 with an increment in the learning rate of the optimizer. In this way, by using this procedure, the required model is trained.

**Algorithm 3** Model Training Procedure

**Output:** Trained ensemble model with prediction accuracy $P_{ACC}$ greater than threshold accuracy $\Theta_{ACC}$

1: **for** all augmented training data $CS_j$ and $BSM_m$ **do**
2: Load the augmented training data.
3: Read through each image in the training data.
4: Apply background subtraction to separate the foreground and background of the image.
5: **if** Foreground is empty **then**
6: Go to the next image.
7: **else**
8: Continue with the next step.
9: **end if**
10: Send foreground image to both the Resnet50 model and the YoloV4 model.
11: Get the output features from both models.
12: Send each output to a dropout layer.
13: Concatenate the output features from both dropout layers using the concatenation layer.
14: Pass the concatenated feature vector through the maxpooling and dense layers.
15: Pass the output of the dense layer through the sigmoid layer for classification.
16: Calculate the loss between the predicted output and the true label.
17: Use back-propagation to update the weights of the model.
18: **if** $P_{ACC} < \Theta_{ACC}$ **then**
19: Go to Step 2 with an increment in the learning rate of the optimizer.
20: **else**
21: Required model is trained.
22: **end if**
23: **end for**

### D. IOT BACKEND

In the proposed EDDSBS), the concepts of the Internet of Things (IoT) and deep learning are used. There is a Raspberry Pi, which acts as the central control server for an alarm detection module. The deployed cameras are equipped with the features of IoT and ensemble models of deep learning. The Raspberry Pi is equipped with a camera and is responsible for capturing and extracting images locally. These images were then sent to our Flask REST API, present in the cloud server for further processing and prediction. Once a drone was predicted, the Raspberry Pi triggered the alarm system, consisting of multiple buzzers and speakers, to alert the authorities of the drone's presence. To ensure timely action, we also connected the Raspberry Pi to Gmail SMTP Access to receive email notifications of drone detection. By integrating the Raspberry Pi with the alarm system and cloud-based API, we created an efficient and effective monitoring system to detect and respond to unauthorized drone activity in real time.

## IV. PRACTICAL IMPLEMENTATION

This section provides practical details on the implementation of the proposed EDDSBS, including information on the hardware and software utilized. The setup had a processor of 2 X Intel Xeon and 12 GB of random access memory (RAM). The implementation was done over the Google Colab platform via a Ubuntu 18.04.5 LTS platform. The programming was done through Python 3.8 along with the Tensorflow library with Keras API.

We provide the details of the used UAVDT dataset, which is an unmanned aerial vehicle (UAV) detection and tracking benchmark dataset. It contains around 80,000 sample frames from ten hours of raw videos. There are three essential and fundamental responsibilities, namely object detection (abbreviated as DET), single object tracking (abbreviated as SOT), and multiple object tracking (abbreviated as MOT). The dataset was collected by using UAVs in a variety of challenging environments. Vehicles are the primary focus of attention in this benchmarking exercise. The frames have bounding boxes and a few other helpful attributes, such as the category of the vehicles and occlusion that have been manually annotated. The UAVDT benchmark comprises one hundred video sequences that are chosen from more than ten hours' worth of movies acquired with a UAV platform in various sites in metropolitan regions. The locations include squares, arterial streets, toll stations, highways, crossings, and T-junctions [43].

The required model was trained and validated on the desired dataset. We conducted "K4 cross-validation by dividing the dataset into four equal folds". After that, they were used to train and test the model each time for the average accuracy measurement. The "Stochastic Gradient Descent (SGD) optimizer and binary cross-entropy loss function" were used for the hyper-parameters for the compilation of the model. Then it was executed for fifty epochs. Inside the training process, the monitoring of performance metrics, i.e., training accuracy and loss, validation accuracy and loss, was estimated. It ensured that the model was not overfitting the training data.

### A. EVALUATION METRIC

The proposed EDDSBS was evaluated using four key parameters, like "true positive (TP), false positive (FP), true negative (TN), and false negative (FN)." The TP and TN parameters provide the measurement of the number of correctly identified drones and non-drones (like, birds and airplanes), respectively. The FP and FN parameters provide the measurement of the number of incorrectly identified non-drones as drones and drones as non-drones, respectively [44], [45].

- *Accuracy:* It is a very important performance parameter, which is measured as all correctly identified cases [45]. Therefore, utilizing accuracy is imperative when classes are equally important. It is estimated as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}. \tag{1}$$

- *Recall:* The number of "positive class predictions made out of all positive examples in the dataset is calculated as recall [45]." It is estimated as follows:

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{2}$$

- *Precision:* The number of "positive class predictions that actually belong to the positive class is measured by precision [45]." It is estimated as follows:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{3}$$

- *F1-score:* Also referred to as "F1-measure, which is calculated through the harmonic mean of precision and recall." It provides the "exact estimate of the incorrectly classified cases than the accuracy [45]." It is mathematically denoted as:

$$\text{F1} - \text{score} = \frac{2(P \times R)}{P + R} \tag{4}$$

where $P$ is Precision and $R$ is Recall

- *False Positive Rate (FPR):* False Positive Rate is a metric that measures the percentage of times that a negative class instance is wrongly forecasted as having a positive outcome [44]. It is estimated as follows:

$$\text{FPR} = \frac{FP}{TN + FP}. \tag{5}$$

- *False Negative Rate (FNR):* False negative rate is a metric that measures the proportion of correctly predicted positive class instances that end up being wrongly labeled as negative [44]. It is estimated as follows:

$$\text{FNR} = \frac{FN}{TP + FN}. \tag{6}$$

**TABLE 1.** Performance evaluation of the proposed EDDSBS.

| Accuracy | Precision | Recall | F1-Score | FPR | FNR |
|---|---|---|---|---|---|
| 91.20% | 89.92% | 92.80% | 0.9134 | 10.40% | 7.20% |

### B. RESULTS AND DISCUSSIONS

Table 1 contains the details of the performance metrics of EDDSBS. It includes parameters, like accuracy, precision, recall, F1-score, false positive rate (FPR), and false negative rate (FNR). The values obtained for various performance parameters for the proposed EDDSBS with respect to accuracy, precision, recall, F1-score, FPR, and FNR are 91.20%, 89.92%, 92.80%, 0.9134, 10.40%, and 7.20%, respectively.

Figure 3 depicts the visual representation of the accuracy parameter for training and validation of the proposed EDDSBS. The obtained confusion matrix of the proposed EDDSBS is given in Figure 4. It provides information about various parametric values. For example, there are 348 true positives, 27 false negatives, 39 false positives, and 336 true negatives for the proposed EDDSBS.

## V. COMPARATIVE STUDY

In this section, we have compared the effectiveness of the proposed EDDSBS with other existing schemes. The comparisons are made in terms of accuracy and other important features.

The accuracy of the considered schemes of Unlu et al. [36], Brown et al. [37], Xun et al. [38],

**TABLE 2.** Performance comparison of various schemes.

| Scheme | Methodology | Limitations | Accuracy (%) |
|---|---|---|---|
| Unlu *et al.* [36] | Extracted 2-dimensional scale, rotation and translation invariant Generic Fourier Descriptor (GFD) features for drone vs. bird classification. | Dataset used was small, not diverse, and cannot be used in a real-time setting with non-ideal conditions. | 85.30 |
| Brown *et al.* [37] | Used four transfer learning models over a three drone collected dataset along with cross-validation and error analysis. | Dataset is limited with similar looking drones, and a video-based detection scheme is not present. | 85.90 |
| Xun *et al.* [38] | Real-time drone detection using YOLOv3 model with integration of NVIDIA Jetson TX2 for real-time drone detection. | Dataset was limited along with baseline model that was outdated. | 88.90 |
| Shi *et al.* [39] | Utilization of multiple transfer learning models on a large dataset for drone identification. | Detection failed with a complex background, due to entire image classification without background subtraction. | 89.32 |
| Proposed scheme (EDDSBS) | Using transfer learning-based ensemble models and a background subtraction module, presented a complete IoT-enabled detection scheme. | Further improvement can be done by inducing attention channels. | 91.20 |

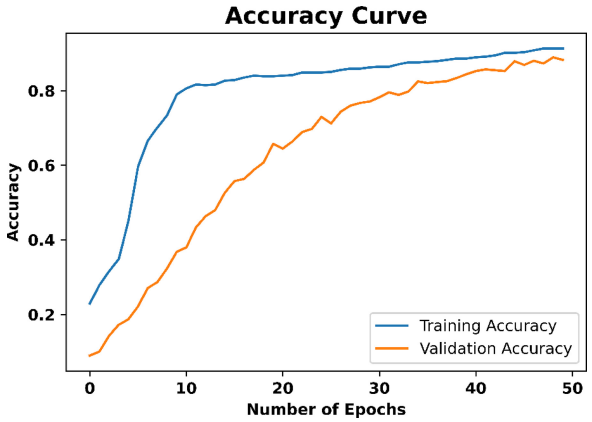


**FIGURE 3.** Accuracy curve of the proposed EDDSBS.

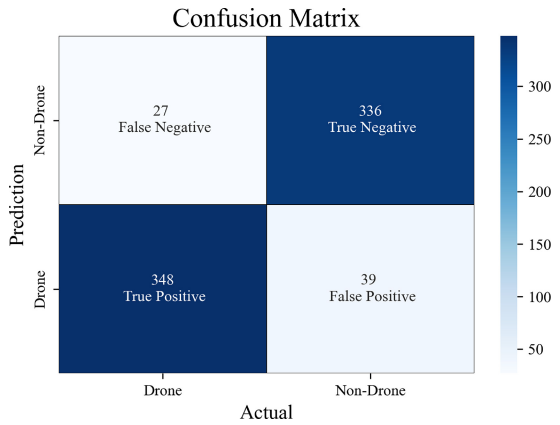


**FIGURE 4.** Obtained confusion matrix.

Shi and Li [39], and the proposed EDDSBS are 85.30%, 85.90% (VGG-16), 88.90%, 89.32% (YOLOv4), and 91.20%, respectively. The details of the comparison between the proposed EDDSBS and existing schemes are presented in Table 2. The results demonstrate that the proposed EDDSBS performed better than the other existing approaches.

## VI. CONCLUSION AND FUTURE WORK

In this article, we presented a novel ensemble-based IoT-enabled drones detection scheme using transfer learning and background subtraction technique (EDDSBS). The presented results demonstrate that the proposed EDDSBS can effectively detect drones with higher accuracy and precision. The proposed EDDSBS also showed superior performance as compared to the competing existing schemes in terms of detection accuracy. Therefore, we believe that the proposed EDDSBS has significant potential to provide security and safety against unauthorized flying drones, which is applicable in various day-to-day applications.

In the future, we would like to extend the proposed EDDSBS to incorporate other advanced machine learning techniques, such as attention-enabled deep learning and reinforcement learning, in order to improve its accuracy and efficiency further. We would also like to add some cyber security mechanisms (for example, authentication, key management, and intrusion detection) in the presented scheme as well.

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers and the editor for their valuable suggestions and comments which helped in improving the presentation and technical quality of this article.

## REFERENCES

[1] G. Jie and L. Jiahui, "Media attention, green technology innovation and industrial enterprises' sustainable development: The moderating effect of environmental regulation," *Econ. Anal. Policy*, vol. 79, pp. 873–889, Sep. 2023, doi: 10.1016/j.eap.2023.07.003.

[2] S. N. Swamy and S. R. Kota, "An empirical study on system level aspects of Internet of Things (IoT)," *IEEE Access*, vol. 8, pp. 188082–188134, 2020.

[3] P. Nayak and G. Swapna, "Security issues in IoT applications using certificateless aggregate signcryption schemes: An overview," *Internet Things*, vol. 21, Apr. 2023, Art. no. 100641.

[4] P. K. Donta, S. N. Srirama, T. Amgoth, and C. S. R. Annavarapu, "Survey on recent advances in IoT application layer protocols and machine learning scope for research directions," *Digit. Commun. Netw.*, vol. 8, no. 5, pp. 727–744, 2022.

[5] J. M. Talavera et al., "Review of IoT applications in agro-industrial and environmental fields," *Comput. Electron. Agricul.*, vol. 142, pp. 283–297, Nov. 2017.

[6] S. Wulfovich, H. Rivas, and P. Matabuena, "Drones in healthcare," *Digital Health: Scaling Healthcare to the World*. Cham, Switzerland: Springer, pp. 159–168, 2018.

[7] D. Floreano and R. J. Wood, "Science, technology and the future of small autonomous drones," *Nature*, vol. 521, no. 7553, pp. 460–466, 2015.

[8] A. Derhab et al., "Internet of drones security: Taxonomies, open issues, and future directions," *Veh. Commun.*, vol. 39, Feb. 2023, Art. no. 100552.

[9] P. Boccadoro, D. Striccoli, and L. A. Grieco, "An extensive survey on the Internet of Drones," *Ad Hoc Netw.*, vol. 122, Nov. 2021, Art. no. 102600.

[10] A. Cavoukian, *Privacy and Drones: Unmanned Aerial Vehicles*. Inf. Privacy Commiss. Ontario: Toronto, ON, Canada, 2012.

[11] S. Son, D. Kwon, S. Lee, Y. Jeon, A. K. Das, and Y. Park, "Design of secure and lightweight authentication scheme for UAV-enabled intelligent transportation systems using blockchain and PUF," *IEEE Access*, vol. 11, pp. 60240–60253, 2023.

[12] A. Sripesh, M. Wazid, D. P. Singh, A. K. Das, and B. Verma, "BAKP-IoDA: Blockchain driven authentication and key agreement protocol for Internet of Drones based applications," in *Proc. 5th Int. ACM Mobicom Workshop Drone Assist. Wireless Commun.5G Beyond*, Sydney, NSW, Australia, 2022, pp. 25–30, doi: 10.1145/3555661.3560859.

[13] B. Bera, A. K. Das, and A. K. Sutrala, "Private blockchain-based access control mechanism for unauthorized UAV detection and mitigation in Internet of Drones environment," *Comput. Commun.*, vol. 166, pp. 91–109, Jan. 2021.

[14] M. Wazid, J. Singh, A. K. Das, S. Shetty, M. K. Khan, and J. J. Rodrigues, "ASCP-IoMT: AI-enabled lightweight secure communication protocol for Internet of Medical Things," *IEEE Access*, vol. 10, pp. 57990–58004, 2022.

[15] A. Khattak et al., "An efficient supervised machine learning technique for forecasting stock market trends," *Inf. Knowl. Internet Things*. Cham, Switzerland: Springer, 2022, pp. 143–162.

[16] T. P. Carvalho, F. A. Soares, R. Vita, R. D. P. Francisco, J. P. Basto, and S. G. Alcalá, "A systematic literature review of machine learning methods applied to predictive maintenance," *Comput. Ind. Eng.*, vol. 137, Nov. 2019, Art. no. 106024.

[17] P. Bhavsar, I. Safro, N. Bouaynaya, R. Polikar, and D. Dera, "Machine learning in transportation data analytics," in *Data Analytics for Intelligent Transportation Systems*. Amsterdam, The Netherlands: Elsevier, 2017, pp. 283–307.

[18] D. Canedo and A. J. Neves, "Facial expression recognition using computer vision: A systematic review," *Appl. Sci.*, vol. 9, no. 21, p. 4678, 2019.

[19] J. Gao, Y. Yang, P. Lin, and D. S. Park, "Computer vision in healthcare applications," *J. Healthc. Eng.*, vol. 2018, Mar. 2018, Art. no. 5157020.

[20] L. Torrey and J. Shavlik, "Transfer learning," in *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*. Hershey, PA, USA: IGI Global, 2010, pp. 242–264.

[21] Z.-H. Zhou, *Ensemble Learning*. Singapore: Springer, 2021, pp. 181–210. [Online]. Available: https://doi.org/10.1007/978-981-15-1967-3_8

[22] A. Jamali, M. Mahdianpari, B. Brisco, J. Granger, F. Mohammadimanesh, and B. Salehi, "Comparing solo versus ensemble convolutional neural networks for wetland classification using multi-spectral satellite imagery," *Remote Sens.*, vol. 13, no. 11, p. 2046, 2021.

[23] G. Fang, J. Yi, X. Wan, Y. Liu, and H. Ke, "Experimental research of multistatic passive radar with a single antenna for drone detection," *IEEE Access*, vol. 6, pp. 33542–33551, 2018.

[24] H. Kang, K.-B. Bae, M.-H. Jung, and S.-O. Park, "Measurement and analysis of radiation leakage from a GPS module for the detection of drones," *IEEE Antennas Wireless Propag. Lett.*, vol. 19, no. 9, pp. 1610–1614, Sep. 2020.

[25] T. Delleji, H. Fekih, and Z. Chtourou, "Deep learning-based approach for detection and classification of Micro/Mini Drones," in *Proc. 4th Int. Conf. Adv. Syst. Emerg. Technol. (IC_ASET)*, Hammamet, Tunisia, 2020, pp. 332–337, doi: 10.1109/IC_ASET49463.2020.9318281.

[26] A. Rozantsev, V. Lepetit, and P. V. Fua, "Detecting flying objects using a single moving camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 879–892, May 2017.

[27] F. Samadzadegan, F. Dadrass Javan, F. A. Mahini, and M. Gholamshahi, "Detection and recognition of drones based on a deep convolutional neural network using visible imagery," *Aerospace*, vol. 9, no. 1, p. 31, Jan. 2022. [Online]. Available: http://dx.doi.org/10.3390/aerospace9010031

[28] A. Carrio, S. Vemprala, A. Ripoll, S. Saripalli, and P. Campoy, "Drone detection using depth maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Madrid, Spain, 2018, pp. 1034–1037, doi: 10.1109/IROS.2018.8593405.

[29] Y. Lv, Z. Ai, M. Chen, X. Gong, Y. Wang, and Z. Lu, "High-resolution drone detection based on background difference and SAG-YOLOv5s," *Sensors*, vol. 22, no. 15, p. 5825, 2022.

[30] J. Peng, C. Zheng, P. Lv, T. Cui, Y. Cheng, and S. Lingyu, "Using images rendered by PBRT to train faster R-CNN for UAV detection." 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:56260675

[31] N. Al-Qubaydhi et al., "Detection of unauthorized unmanned aerial vehicles using YOLOv5 and transfer learning," *Electronics*, vol. 11, no. 17, p. 2669, Aug. 2022. [Online]. Available: http://dx.doi.org/10.3390/electronics11172669

[32] U. Seidaliyeva, D. Akhmetov, L. Ilipbayeva, and E. T. Matson, "Real-time and accurate drone detection in a video with a static background," *Sensors*, vol. 20, no. 14, p. 3856, 2020.

[33] Z. Wang, L. Qi, Y. Tie, Y. Ding, and Y. Bai, "Drone detection based on FD-HOG descriptor," in *Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discov. (CyberC)*, Zhengzhou, China, 2018, pp. 433–4333, doi: 10.1109/CyberC.2018.00084.

[34] K. V. S. Reddy, V. R. Molabanti, S. T. P. Dumpala, and U. R. Nelakuditi, "YOLOV3 based real time drone detection for counter drone system," in *Proc. IEEE 3rd Int. Conf. Technol. Eng. Manag. Soc. Impact Using Market. Entrepreneurship Talent (TEMSMET)*, Mysuru, India, 2023, pp. 1–5, doi: 10.1109/TEMSMET56707.2023.10149935.

[35] H. Chen, Z. Wang, and L. Zhang, "Collaborative spectrum sensing for illegal drone detection: A deep learning-based image classification perspective," *China Commun.*, vol. 17, no. 2, pp. 81–92, Feb. 2020.

[36] E. Unlu, E. Zenou, and N. Riviere, "Using shape descriptors for UAV detection," *Electron. Imag.*, vol. 2018, pp. 1–5, Jan. 2018.

[37] J. Brown, Z. Gharineiat, and N. Raj, "CNN based image classification of malicious UAVs," *Appl. Sci.*, vol. 13, no. 1, p. 240, Dec 2022. [Online]. Available: http://dx.doi.org/10.3390/app13010240

[38] D. T. W. Xun, Y. L. Lim, and S. Srigrarom, "Drone detection using YOLOv3 with transfer learning on NVIDIA Jetson TX2," in *Proc. 2nd Int. Symp. Instrum. Control Artif. Intell. Robot. (ICA-SYMP)*, Bangkok, Thailand, 2021, pp. 1–6, doi: 10.1109/ICA-SYMP50206.2021.9358449.

[39] Q. Shi and J. Li, "Objects detection of UAV for anti-UAV based on YOLOv4," in *Proc. IEEE 2nd Int. Conf. Civil Aviat. Safety Inf. Technol. (ICCASIT*, Weihai, China, 2020, pp. 1048–1052, doi: 10.1109/ICCASIT50869.2020.9368788.

[40] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015. [Online]. Available: https://arxiv.org/abs/1512.03385.

[42] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao. "Yolov4: Optimal speed and accuracy of object detection." 2020. [Online]. Available: https://arxiv.org/abs/2004.10934.

[43] D. Du et al., "UAVDT (unmanned aerial vehicle benchmark object detection and tracking)," 2023. [Online]. Available: https://paperswithcode.com/dataset/uavdt. Accessed on July 2023.

[44] M. Wazid, P. R. Dsouza, A. K. Das, V. Bhat, N. Kumar, and J. J. P. C. Rodrigues, "RAD-EI: A routing attack detection scheme for edge-based Internet of Things environment," *Int. J. Commun. Syst.*, vol. 32, no. 15, 2019, Art. no. e4024.

[45] M. Wazid, A. K. Das, V. Chamola, and Y. Park, "Uniting cyber security and machine learning: Advantages, challenges and future research," *ICT Exp.*, vol. 8, no. 3, pp. 313–321, 2022.

**JASKARAN SINGH** (Student Member, IEEE) is currently pursuing the Bachelor of Technology degree in computer science and engineering (with specialization in Data Science) with the Department of Computer Science and Engineering, Graphic Era Deemed to be University Dehradun, India. His area of research is machine learning, data science, cyber security, and Internet of Things.

**KESHAV SHARMA** (Student Member, IEEE) is currently pursuing the Bachelor of Technology degree in computer science and engineering (with specialization in Data Science) with the Department of Computer Science and Engineering, Graphic Era Deemed to be University Dehradun, India. His area of research is data science, cyber security, and Internet of Things.

**MOHAMMAD WAZID** (Senior Member, IEEE) received the Master of Technology degree in computer network engineering from Graphic Era University, Dehradun, India, and the Ph.D. degree in computer science and engineering from the International Institute of Information Technology, Hyderabad, India. He is currently working as a Professor with the Department of Computer Science and Engineering, Graphic Era University, Dehradun, India, where he is also the Head of the Cybersecurity and IoT Research Group. Prior to this, he was an Assistant Professor with the Department of Computer Science and Engineering, Manipal Institute of Technology, MAHE, Manipal, India. He was also a Postdoctoral Researcher with the Cyber Security and Networks Laboratory, Innopolis University, Innopolis, Russia. His current research interests include security, remote user authentication, the Internet of Things, and cloud computing. He has published more than 100 papers in international journals and conferences in the above areas. He was a recipient of the University Gold Medal and the Young Scientist Award from UCOST, the Department of Science and Technology, Government of Uttarakhand, India.

**ASHOK KUMAR DAS** (Senior Member, IEEE) received the M.Sc. degree in mathematics, the M.Tech. degree in computer science and data processing, and the Ph.D. degree in computer science and engineering from IIT Kharagpur, India. He is currently a Full Professor with the Center for Security, Theory and Algorithmic Research, IIIT, Hyderabad, India. He was also a Visiting Faculty with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, VA, USA. His Google Scholar h-index is 80 and i10-index is 230 with over 18,000 citations. His research interests include cryptography, system and network security, blockchain, security in Internet of Things, Internet of Vehicles, Internet of Drones, smart grids, smart city, cloud/fog computing, intrusion detection, AI/ML security, and post-quantum cryptography. He has authored over 360 papers in international journals and conferences in the above areas, including over 310 reputed journal papers. He was a recipient of the Institute Silver Medal from IIT Kharagpur. He is/was on the editorial board of IEEE SYSTEMS JOURNAL, *Journal of Network and Computer Applications* (Elsevier), *Computer Communications* (Elsevier), *Journal of Cloud Computing* (Springer), *Cyber Security and Applications* (Elsevier), *IET Communications*, *KSII Transactions on Internet and Information Systems*, and *International Journal of Internet Technology and Secured Transactions* (Inderscience). He also severed as one of the Technical Program Committee Chairs of the first International Congress on Blockchain and Applications (BLOCKCHAIN'19), Avila, Spain, in June 2019, the International Conference on Applied Soft Computing and Communication Networks (ACN'20), in October 2020, Chennai, India, and the second International Congress on Blockchain and Applications (BLOCKCHAIN'20), L'Aquila, Italy, in October 2020. He has been listed in the Web of Science (Clarivate$^{TM}$) Highly Cited Researcher 2022 in recognition of his exceptional research performance.

**ATHANASIOS V. VASILAKOS** (Senior Member, IEEE) is with the Center for AI Research, University of Agder, Grimstad, Norway. He served or is serving as an Editor for many technical journals, such as the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON NANOBIOSCIENCE, IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE, *ACM Transactions on Autonomous and Adaptive Systems*, and the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS. He was a WoS Highly Cited Researcher from 2016 to 2021.