Article

Evaluating a Motion-Based Region Proposal Approach with Background Subtraction Methods for Small Drone Detection

Elif Ucurum *, Phil Birch, Xudong Han, Yueying Tian, Rupert Young and Chris Chatwin

School of Engineering and Informatics, University of Sussex, Brighton BN1 9RH, UK; p.m.birch@sussex.ac.uk (P.B.); xh218@sussex.ac.uk (X.H.); yt322@sussex.ac.uk (Y.T.); r.c.d.young@sussex.ac.uk (R.Y.); c.r.chatwin@sussex.ac.uk (C.C.)

* Corresponding author. E-mail: e.ucurum@sussex.ac.uk (E.U.)

Received: 25 December 2024; Accepted: 20 March 2025; Available online: 7 April 2025

ABSTRACT: The detection of drones in complex and dynamic environments poses significant challenges due to their small size and background clutter. This study aims to address these challenges by developing a motion-based pipeline that integrates background subtraction and deep learning-based classification to detect drones in video sequences. Two background subtraction methods, Mixture of Gaussians 2 (MOG2) and Visual Background Extractor (ViBe), are assessed to isolate potential drone regions in highly complex and dynamic backgrounds. These regions are then classified using the ResNet18 architecture. The Drone-vs-Bird dataset is utilized to test the algorithm, focusing on distinguishing drones from other dynamic objects such as birds, trees, and clouds. By leveraging motion-based information, the method enhances the drone detection process by reducing computational demands. Results show that ViBe achieves a recall of 0.956 and a precision of 0.078, while MOG2 achieves a recall of 0.857 and a precision of 0.034, highlighting the comparative advantages of ViBe in detecting small drones in challenging scenarios. These findings demonstrate the robustness of the proposed pipeline and its potential contribution to enhancing surveillance and security measures.

Keywords: Drone detection; Background subtraction; Small object detection; ResNet18; Motion region proposals

1. Introduction

Drones are being increasingly used in various applications due to their cost effectiveness, easy deployment, fast mobility, efficiency in time and effort, ability to perform repetitive and hazardous tasks, access remote areas, and suitability for real-time applications [1–4]. However, drones can be used in many malevolent operations that threaten communities. There have been numerous incidents where drones were used to eavesdrop, violating the privacy of people and properties, and to transport illegal substances such as drugs or sensitive information [5]. There are records of several drones colliding with aircraft or helicopters. Therefore, detecting drones is significant in terms of preventing malevolent usage of drones and addressing privacy and safety concerns. The primary objective of drone detection is to identify the presence of the drone and its location.

Detecting small drones, in particular, poses considerable challenges. When viewed from standard surveillance cameras, drones often appear as small objects within large, cluttered backgrounds. Although recent advances in computer vision have improved object detection, accurately identifying small drones remains problematic due to their low pixel count and susceptibility to occlusion or dynamic elements like moving trees, water ripples, and clouds. Moreover, the complexity of highly dynamic environments exacerbates this issue, as visual cues can be easily confused with natural background motion [6]. In computer vision literature, there are two main definitions of small objects. The first definition refers to objects with small physical sizes in the real world, such as insects, birds, tennis balls, and human faces in crowd scenes. The second definition relates to relative size, where the objects appear small due to their distance from the camera, as seen in aerial view scenes like synthetic aperture radar (SAR) imagery or images from unmanned aerial vehicles (UAVs). Therefore, the general definition accepted in the field is that small objects occupy a relatively small portion of a large scene. As per the MS-COCO dataset [7], small objects are those with areas less than or equal

https://doi.org/10.70322/dav.2025.10007

to 32 × 32 pixels, while medium-size objects are between 32 × 32 pixels and 96 × 96 pixels, and large objects cover an area more than 96 × 96 pixels.

Many object detection algorithms standardize on deep learning perform detection and classification on feature maps extracted by convolutional neural networks (CNNs). After multiple pooling layers, the image resolution is reduced, causing a significant loss of information on small objects, which already have poor and limited features. Yet high-level semantic features are needed for the detection and classification, so classical object detection algorithms often yield low accuracy when dealing with small targets [8]. Another challenge in drone detection is the dynamic environment, where drones can be easily lost against moving backgrounds. Environmental noise, such as weather conditions, trees, grass, sky, or buildings, makes detection particularly difficult since it relies on visual semantic features like colour, texture and shape. Consequently, distinguishing drones from other small, fast-moving objects such as birds or insects adds another layer of difficulty.

This paper proposes a robust pipeline that fuses background subtraction with deep learning-based classification to accurately detect and distinguish drones from other objects in video sequences. By leveraging spatiotemporal information across frames, the method isolates motion regions in complex, dynamic backgrounds and reduces the need to analyze the entire frame. The pipeline begins with background subtraction using either MOG2 (Mixture of Gaussians 2) [9] or ViBe (Visual Background Extractor) [10] to segment foreground elements. Candidate regions are generated by identifying blobs in the foreground mask that exceed a predefined size threshold. To prepare the dataset for training, regions labeled “drone” are extracted using ground truth annotations, while “not drone” regions come from other moving objects, or background noise (e.g., birds, clouds, trees). Each region is resized to 100 × 100 pixels to standardize the input for a ResNet18 classifier, which learns to differentiate drones from non-drones. During testing, the trained network evaluates only these motion-centric regions, reducing redundant computations and improving overall detection accuracy.

The proposed method leverages temporal information across video frames to isolate motion regions, effectively identifying areas likely to contain drones. In contrast to existing studies that rely solely on deep learning or background subtraction, our approach uniquely integrates both, thereby reducing the search space and computational overhead while preserving high-level semantic features for classification. This study compares two background subtraction techniques, MOG2 and ViBe, on a highly challenging dataset where target objects are often indistinguishable from dynamic background elements. Our systematic evaluation of these methods for small drone detection in cluttered scenes underscores the novelty of this work, as it provides a clear understanding of how motion-based foreground extraction can be effectively integrated with convolutional networks to achieve robust detection. Both methods demonstrate their ability to locate motion regions and detect drones in complex environments, with ViBe showing stronger performance in minimizing interference from dynamic background elements. This comprehensive assessment highlights the comparative strengths of MOG2 and ViBe while emphasizing the potential of integrating background subtraction with deep learning to detect small drones in highly dynamic and noisy environments. It lays the foundation for future research on real-time drone detection in challenging conditions.

The remainder of this paper is organized as follows. In Section 2, we review related work on background subtraction and drone detection, highlighting the challenges of detecting small objects in dynamic environments. Section 3 describes our proposed pipeline in detail, including the integration of background subtraction methods with a ResNet18-based classifier. In Section 4, we present the experimental setup, including dataset details, training parameters, and evaluation metrics, followed by a discussion of our results in Section 5. Finally, Section 6 concludes the paper and outlines directions for future research.

2. Related Work

2.1. Background Subtraction

Background subtraction is a fundamental task in computer vision. It is applied to video streams to segment images into background and foreground. The background represents the static elements of a scene, while the foreground includes dynamic elements, such as moving objects. The process of background subtraction includes three main steps: background initialization, where a background model is built based on a fixed number of frames; foreground detection, which compares the current frames with the background model to extract the foreground; and background maintenance, where the background model is updated to include new static elements, such as objects that have stopped moving for a while [11]. Background subtraction is used primarily in video analysis and is widely used for detecting and tracking moving objects. It plays a key role in various applications, such as video surveillance, people counting, anomaly detection, content-based video coding, target tracking, intrusion detection, vehicular traffic analysis, and human activity recognition [12].

One of the simplest approaches related to background subtraction is frame differencing, which involves calculating the difference between two consecutive frames to identify changes. While this method can work under ideal conditions, it struggles with challenges in real-world scenarios, such as lighting changes, dynamic background elements, shadows, irregular object motion, weather variations, and low frame rates. To address these challenges, more sophisticated background subtraction algorithms have been developed over the years. These algorithms rely on feature extraction methods, which can be broadly categorized into hand-crafted features and learning-based features.

The study by [11] comprehensively reviews traditional background subtraction methods and evaluates their performance on the Background Models Challenge (BMC) dataset. The methods are assessed using evaluation metrics such as precision, recall, F-measure, D-score, and SSIM, while considering various challenging scenarios. According to this study, methods such as PBAS, MultiLayerBGS, LBAdaptiveSOM, DPWrenGABGS, and Mixture of Gaussians demonstrate the best robustness and performance under difficult conditions. A more recent comparative study by [12] evaluates the performance of 12 background subtraction methods, including traditional methods such as GMM, KDE, and SuBSENSE, as well as deep learning-based methods like Cascaded CNN, BSUV-Net, and FgSegNet. The results indicate that deep learning-based methods significantly outperform traditional approaches, with the FgSegNet models achieving the best overall performance in terms of the F-measure across various challenging scenarios.

A major breakthrough occurred in 2016 when convolutional neural networks (CNNs) were specifically applied to the background subtraction task [13]. Since then, various types of neural networks, such as fully convolutional neural networks (FCNNs) and 3D-CNN-based algorithms, have been proposed for background subtraction by capturing the temporal changes in videos. Additionally, some studies have employed generative adversarial networks (GANs) to model the background and extract the foreground [12].

Although deep learning-based methods perform better than many traditional methods, in most cases, they require large amounts of training data, which may not always be available. Moreover, these methods often struggle with unseen videos due to poor generalization and significantly increase computational costs, making real-time processing more challenging. Comparative background subtraction methods rely heavily on benchmark datasets such as CDnet (Change Detection), BMC (Background Models Challenge), and LASIESTA, which contain various moving objects like cars, humans, and animals [12]. However, these datasets often lack specific small-object classes, such as birds or drones, which present unique challenges due to their size, speed, and irregular motion. One of the key challenges in background subtraction is detecting small moving objects in dynamic environments. Scenarios involving waving trees or water ripples are particularly difficult, as these background elements exhibit motion but are still part of the background. Methods often fail to eliminate these from the foreground, mistakenly classifying them as moving objects.

The study by [14] reviews background subtraction techniques applied in real-world applications, ranging from video surveillance in various environments to human and animal detection in diverse contexts. Among these, bird detection emerges as one of the most similar applications to drone detection. According to their evaluation, detecting birds poses significant challenges, particularly in handling varying lighting conditions and capturing the motion of small, fast-moving objects. These challenges are highly relevant to drone detection tasks, where small drones can be difficult to distinguish from background noise and natural motion.

2.2. Background Subtraction in Drone Detection

There are two main approaches to vision-based drone detection: traditional methods and deep learning-based methods. Traditional methods rely on hand-crafted feature extraction techniques like Histogram of Oriented Gradients (HOG), SIFT, and Haar Wavelets, combined with machine learning classifiers such as SVM and random forests. While these methods are computationally less intensive, they struggle with small objects like drones, dynamic backgrounds, and real-time applications [2,3,5,15]. Deep learning-based methods, on the other hand, automatically extract features using convolutional neural networks (CNNs). These methods are categorized into one-stage detectors, like YOLO and SSD, which perform detection in a single pass, and two-stage detectors, like Faster R-CNN and DETR, which involve region proposal followed by classification. One-stage detectors are faster and suitable for real-time applications, while two-stage detectors offer higher accuracy but are computationally more demanding [16,17]. A recent study by [18] compared state-of-the-art algorithms such as Faster R-CNN, SSD512, YOLOv3, and DETR for drone detection. It also evaluated tracking algorithms like SORT and DeepSORT for leveraging temporal information, highlighting that detection typically serves as the backbone for tracking. However, tracking algorithms depend on the object detector; if the detector fails, the tracking algorithm also fails.

Although background subtraction is a well-established technique for detecting medium- or large-sized moving objects, it is rarely utilized in drone detection. This underuse stems largely from drones' small size and frequent appearance in complex, dynamic environments, where background subtraction methods must handle varied illumination, adapt to moving elements, and accurately segment tiny objects in motion. Most drone detection approaches each frame independently, overlooking temporal context that could improve accuracy. A tracking algorithm is typically employed alongside object detection when spatiotemporal information is incorporated. However, if the drone's visual cues are weak, sometimes imperceptible even by the human eye, temporal cues become critical in confirming a drone's presence.

Despite these challenges, some recent studies have successfully leveraged background subtraction as a preprocessing step to enhance detection efficiency. For instance, the study by [19] demonstrated the potential of using a simple background subtraction method to create candidate regions in high-resolution videos, significantly reducing computational overhead. They integrated this approach with the YOLOv5 network, achieving improved detection efficiency in high-resolution imagery. Similarly, [20] employed a motion detector based on a two-point background subtraction algorithm to identify moving objects, which were subsequently classified using a CNN-based classifier into drones, birds, or background elements. In a related study, [21] proposed a deep learning-based detection and classification framework for micro/mini drones using an enhanced YOLOv3 architecture and a custom multi-class drone dataset, underscoring the challenges of detecting small-scale drones. Moreover, recent datasets, such as the Long-Range Drone Detection Dataset [22], focus on detecting drones from far away—where they appear as small objects—highlighting the importance of developing robust detection methods for extreme scale variations. Collectively, these studies emphasize that, despite its limitations, background subtraction can still play a critical role as a preprocessing step in small drone detection. By isolating motion regions, these methods address some of the computational inefficiencies associated with processing entire video frames while enhancing detection accuracy in dynamic and complex environments.

Nevertheless, applying background subtraction to small drone detection remains underexplored, and there is a lack of comparative studies investigating how well these methods adapt to small-object detection in highly dynamic backgrounds. Drones may occupy only a few pixels, so parameters like learning rates, morphological operation kernels, and updating thresholds must be carefully fine-tuned: overly aggressive settings can suppress the drone entirely, while looser settings risk flooding the foreground mask with noise from water ripples, moving clouds, or waving trees. These natural environmental motions resemble small moving blobs and can be indistinguishable from drones in low-resolution frames. Consequently, examining how background subtraction methods perform under these conditions is essential for ensuring reliable drone detection, minimizing false positives, and capturing the often-subtle motion cues of genuinely small targets.

Given these challenges, our work investigates how MOG2 and ViBe perform under small-object conditions with significant environmental motion, as detailed in the following section. We propose a pipeline that integrates background subtraction with a deep convolutional classifier, focusing on and classifying drones from a broad range of dynamic backgrounds.

3. Materials and Methods

In this work, we focus on detecting motion regions in videos captured with a static camera using background subtraction methods, followed by classifying these regions with the ResNet18 network architecture. This approach aims to improve detection accuracy by isolating motion regions and avoiding processing static background areas. We implemented two conventional background subtraction methods, MOG2 and ViBe, to conduct a comparative study on a highly challenging dataset. Our study utilizes the Drone vs Bird dataset [23], which includes challenging scenarios such as water ripples, waving trees, moving clouds, illumination changes, and the presence of birds in dynamic environments.

Unlike most background subtraction studies, which typically focus on medium or large-sized objects, this work emphasizes the detection of small objects in highly dynamic backgrounds. Traditional drone detection studies often rely solely on direct object detection, where features of small objects are frequently lost in the pooling layers of neural network models. To address this limitation, we propose focusing on motion regions identified by background subtraction methods to ensure more accurate detection of small and dynamic objects in complex environments. Furthermore, we compare the performance of two widely used background subtraction methods to evaluate their effectiveness in isolating small targets in dynamic environments.

In the proposed detection algorithm, background subtraction serves as a crucial first step in generating a foreground mask. MOG2 or ViBe identifies moving pixels by comparing each incoming video frame against an evolving background model. Once the foreground mask is created, morphological operations (opening and filtering) are applied to remove noise and preserve small moving objects. Next, the cleaned mask's connected components (or blobs) are located via contour detection. These blobs are considered candidate regions that may contain drones. Finally, each candidate region is cropped from the original frame and resized to 100 × 100 pixels before being passed to the ResNet18 network for classification. Hence, MOG2 and ViBe directly feed into the bounding-box proposal stage of the detection pipeline, and the ResNet18 classifier subsequently determines whether each proposed region is a drone or not. This integration ensures that only motion-containing portions of the frame are analyzed in detail, reducing computational overhead and improving detection performance in dynamic scenes. Figure 1 illustrates the pipeline of the proposed algorithm for detecting drones, accompanied by sample images demonstrating each step of the process.

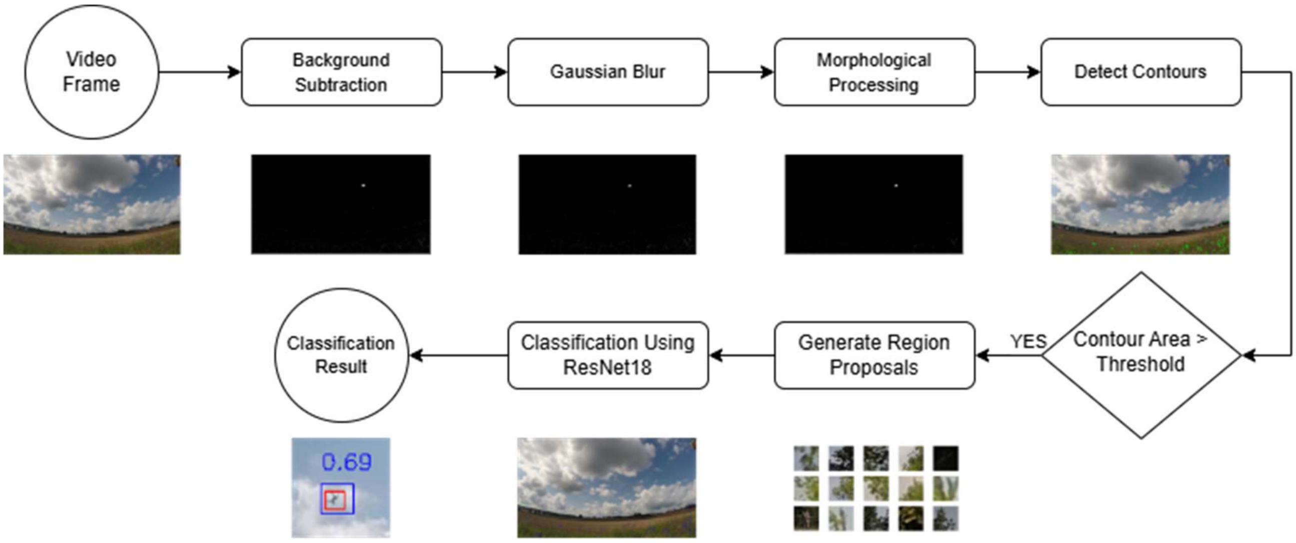


Figure 1. Flowchart of the proposed algorithm with sample images.

3.1. Motion Region Proposal with Background

We implemented background subtraction methods to create a foreground mask, a binary image where pixels with a value of 1 (white) represent areas with motion. These foreground masks were generated by applying two background subtraction methods: MOG2 and ViBe.

The Mixture of Gaussians (MOG) model, introduced by Chris Stauffer and W.E.L. Grimson in 1999 [24], remains one of computer vision's most widely used background subtraction methods. This method models each pixel as a mixture of Gaussian distributions, enabling the detection of dynamic background elements. However, MOG's fixed number of Gaussian components limited its adaptability to sudden changes and noisy environments. Addressing these challenges, Zivkovic and van der Heijden developed MOG2 in 2006 [9], introducing adaptive mechanisms that dynamically determine the optimal number of Gaussian distributions per pixel. This enhancement significantly improved the algorithm's ability to adapt to complex scenes, handle gradual illumination changes, and detect shadows more effectively. MOG2's robustness and, adaptability, and widespread implementation in popular libraries like OpenCV have made it a preferred choice for real-time video analysis applications.

ViBe (Visual Background Extractor) was introduced by Barnich and Van Droogenbroeck in 2009 [10], and it is a non-parametric background subtraction algorithm designed for dynamic and complex environments. Unlike parametric models like MOG2, ViBe uses a history-based approach, maintaining a sample set of recent pixel values for each location. Pixels are classified as background or foreground by comparing them to these samples, making ViBe highly effective in handling challenges like moving vegetation, water ripples, and gradual lighting changes. ViBe's probabilistic update mechanism enhances adaptability while remaining computationally efficient. However, it is sensitive to initialization quality and lacks built-in shadow detection, which can lead to errors in scenes with abrupt changes or irregular motion. ViBe was chosen for this study because it can handle dynamic and noisy backgrounds effectively, making it suitable for detecting motion regions in drone detection tasks. Its robustness in scenarios with complex environmental noise, such as waving trees and water ripples, aligns with the challenges presented by the Drone-vs-Bird dataset. Additionally, its computational efficiency and simplicity make it a practical choice for real-time applications.

Once the background subtraction methods created the foreground mask, a Gaussian blur filter with a 3 × 3 kernel for MOG2, and a 5 × 5 kernel for ViBe was applied to reduce noise and smooth the mask, minimizing isolated noise pixels that could interfere with the detection of meaningful motion regions. To further refine the mask and eliminate small, irrelevant regions of motion, morphological operations were performed. An opening operation (erosion followed by dilation) with a 3 × 3 kernel was carefully implemented to remove noise while ensuring the small drones, which could appear as very small regions, were not inadvertently removed during the refinement process.

After the refinement, contours were drawn around the blobs in the foreground mask to identify discrete motion regions. To eliminate irrelevant regions, a size threshold was applied, and contours with an area smaller than 30 pixels were ignored. The remaining valid contours were then extracted from the original video frames and resized to 100 × 100 pixels to ensure dataset consistency and compatibility with the neural network classifier, ResNet18. This pipeline effectively identifies motion regions, minimizes noise, and preserves small drone regions, ensuring accurate classification even in highly dynamic and challenging scenarios.

3.2. Drone Classification Network

For the classification task, we utilized the ResNet18 architecture, a deep residual neural network designed to address the vanishing gradient problem and enable the training of very deep networks using skip connections. ResNet18's structure, which incorporates residual blocks, ensures efficient gradient flow during backpropagation, making it well-suited for our task of distinguishing drones from background regions. Its robust feature extraction capabilities allow it to learn complex representations from small and challenging regions. ResNet18 was chosen for this study due to its balance between computational efficiency and strong performance on small datasets. Compared to larger architectures, ResNet18 provides sufficient depth to learn complex features while remaining lightweight enough for efficient training and inference.

The motion regions identified during the background subtraction step were extracted from the original video frames and resized to 100 × 100 pixels to ensure consistency across all input samples. These regions were then normalized to improve convergence during training and testing. By resizing motion regions to a fixed 100 × 100 resolution, the proposed approach mitigates the feature loss typically observed in deep learning architectures during pooling or stride operations, especially for small objects like drones. This ensures that the drone occupies the entire input frame, effectively 'magnifying' its features and enabling more robust feature extraction. Additionally, the resizing step isolates regions of interest, removing background clutter and ensuring consistent input dimensions for the network, which simplifies training and improves computational efficiency. By focusing on motion regions proposed by background subtraction instead of processing entire frames, the proposed approach minimizes computational overhead. This selective processing reduces the input size for the classifier, enabling efficient analysis of dynamic regions while ignoring static or irrelevant areas.

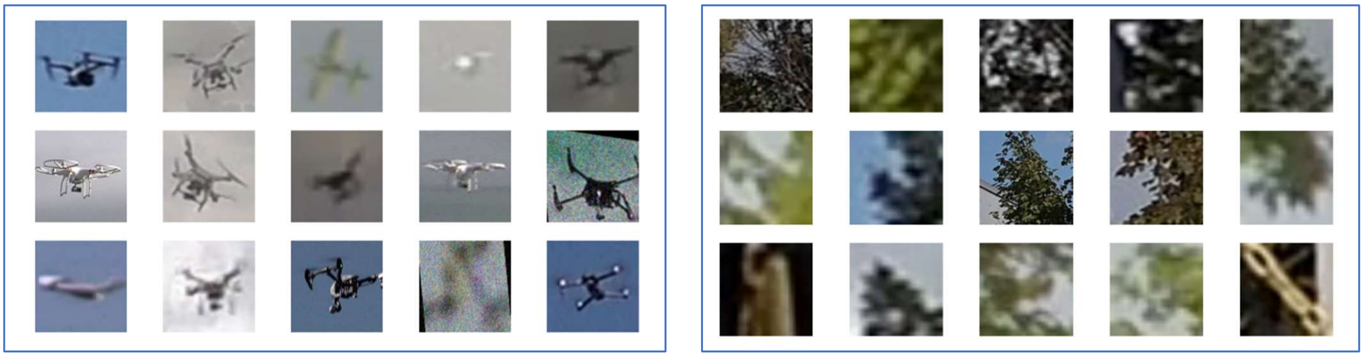
To improve classification reliability and reduce false positives, a confidence score threshold of 0.7 was applied during the prediction stage. This ensures that only regions with a sufficiently high likelihood of containing a drone are considered as detections, helping to balance precision and recall. During training, dropout regularization with a rate of 0.7 was applied to prevent overfitting, ensuring the model did not rely excessively on specific neurons. Additionally, L2 regularization with a value of 0.02 was implemented in the loss function to penalize large weights, further improving generalization and reducing the risk of overfitting. The network was optimized using the Adam optimizer with a learning rate 0.0001, ensuring stable and efficient convergence. The loss function used was binary cross-entropy, appropriate for the binary classification task of distinguishing drones from non-drone regions. A batch size 32 was used to balance computational efficiency and gradient stability during training. The combination of dropout and L2 regularization enabled the model to achieve robust performance even in challenging scenarios with small motion regions and dynamic backgrounds.

4. Experiments

4.1. Dataset and Metrics

The Drone-vs-Bird Detection Challenge dataset consists of 77 video sequences, combining footage from both static and moving cameras. Static cameras account for 32 sequences, while moving cameras contribute 45 sequences and were introduced during later iterations of the challenge. In total, these videos contain several thousand frames, each of which may include zero, one, or multiple drones. The dataset provides annotations for drones, averaging 1.12 drones per frame, but does not include annotations for birds despite their frequent presence in over one-third of the sequences. The pixel sizes of drones vary significantly, ranging from 15 pixels to over 1,000,000 pixels, with most drones being less than 16^2 pixels or between 16^2 and 32^2 pixels, posing a significant detection challenge. The dataset features diverse environments, including urban areas, woodlands, rivers, and maritime scenes, captured under varying weather and lighting conditions. Each video sequence is accompanied by annotations provided as bounding boxes detailing the precise location of drones in each frame. These video sequences were originally collected by the dataset creators to represent real-world scenarios where drones frequently appear alongside other moving objects (e.g., birds) and background motion (e.g., waving trees, water ripples).

For our experiments, we curated a training set and a test set from these videos. Specifically, we used 29 static-camera sequences and all 45 moving-camera sequences for training, while 3 static-camera videos were reserved for testing. The raw drone bounding boxes were directly extracted from the ground-truth annotations whenever drones were present in the frame, and not-drone bounding boxes were sampled from other moving objects or dynamic background elements. Figure 2 shows random samples of images belonging to the drone and not-drone classes from the training dataset. To further balance the dataset and enhance diversity, data augmentation was applied to the drone images. Augmentations included adding Gaussian noise, applying Gaussian blur with a 5 × 5 kernel, JPEG quality reduction, random geometric distortions, and brightness, saturation, contrast, and hue adjustments. Random flips and rotations (90°, 180°, 270°) were also used, simulating real-world conditions and increasing the dataset's variability. In total, after data augmentation, we obtained 126,725 drone images and 126,725 not-drone images for training, resulting in 253,450 labeled samples.



(**a**) (**b**)

Figure 2. (**a**) Samples from 'drone' class in training dataset, (**b**) Samples from 'not-drone' class in training dataset.

From the Drone-vs-Bird dataset, the videos used for testing purposes are: *00_02_45_to_00_03_10_cut.mpg*, *00_06_10_to_00_06_27.mp4*, *2019_09_02_GOPR5871_1058_solo.mp4*. The first video, *00_02_45_to_00_03_10_cut.mpg*, was recorded on a cloudless and bright day, with nearly one-third of the scene covered by water. This setup introduces water ripples and birds as dynamic elements against a bright sky. The second video, *00_06_10_to_00_06_27.mp4*, was captured at sundown under low-light conditions; here, water ripples and moving clouds add additional motion in the background. Finally, *2019_09_02_GOPR5871_1058_solo.mp4* features an urban environment with buildings and a cloudy sky. As the drone flies over, the background periodically includes waving trees and structures, creating further complexity. Collectively, these videos represent bright daytime, low-light, and mixed urban scenes with multiple sources of motion and environmental noise, thereby providing a stringent test of our pipeline's ability to handle real-world drone detection challenges. All videos in the dataset were recorded in Full HD (1920 × 1080). Notably, the video *00_02_45_to_00_03_10_cut.mpg* is captured at 50 FPS, *00_06_10_to_00_06_27.mp4* runs at approximately 30 FPS, and *2019_09_02_GOPR5871_1058_solo.mp4* is recorded at 25 FPS, introducing varied temporal characteristics while maintaining a consistent spatial resolution. The videos in our dataset feature drones moving at varying velocities—from slow hovering to rapid flight—and our pipeline maintained robust detection performance across these conditions, consistent with the high-speed detection capabilities demonstrated by [25] using an enhanced YOLO-V8 mode.

The Intersection over Union (IoU) metric is used to evaluate the quality of predicted bounding boxes in object detection tasks. IoU calculates the overlap between the predicted and ground-truth bounding boxes by dividing the intersection area by the union area of the boxes, providing a clear measure of localization accuracy. A threshold of 0.4 is applied to the IoU to determine the accuracy of detections. A detection is classified as a true positive if the IoU between the predicted bounding box and the ground-truth annotation meets or exceeds this threshold. Conversely, detections with an IoU below the threshold are considered false positives, and any ground-truth annotations not matched to detection are classified as false negatives, indicating missed objects. Figure 3 illustrates the IoU results, where the green bounding boxes represent the ground truth, and the red boxes indicate the predictions, with the model prediction scores displayed above each predicted region. Object detection models are evaluated using metrics such as Average Precision (AP) and Mean

Average Precision (mAP), which assess performance in terms of precision and recall. Precision measures the accuracy of positive predictions, while recall evaluates the proportion of correctly identified positive cases.

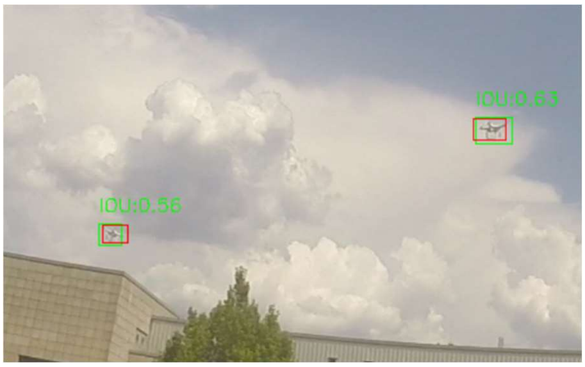


Figure 3. IoU results, where the green bounding boxes represent ground truth, and the red boxes indicate the predictions.

4.2. Experimental Setup

All experiments were conducted on a workstation running Ubuntu 20.04 with Python 3.11.7. The system included an Intel® Core™ i7-7800X CPU at 3.50 GHz, 32 GB RAM, and an NVIDIA GeForce GTX 1080 GPU. PyTorch 1.11.0 (compiled with CUDA 11.3 support) is used for accelerated model training. This setup provided sufficient computational resources to handle data preprocessing and deep learning operations efficiently. For training, the model was run over 20 epochs with a batch size of 32 and an input shape of 100 × 100 × 3, using the Adam optimizer with a learning rate of 0.0001 and binary cross-entropy loss. To mitigate overfitting, we applied a dropout rate of 0.7 and L2 regularization of 0.02.

In addition, the parameters for the background subtraction methods, including thresholding and morphological operations—were fine-tuned through extensive trial and error to optimize performance for our dataset. Inference was performed with a confidence threshold of 0.7, meaning that only predictions with a probability equal to or exceeding 0.7 were accepted as valid detections.

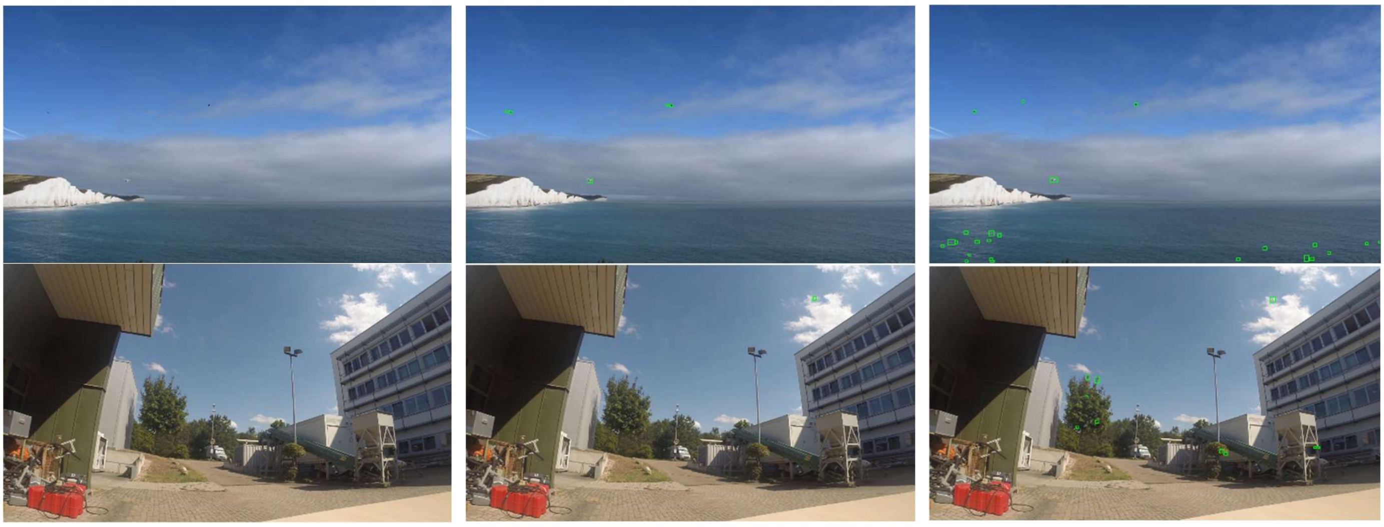
5. Results and Discussion

To evaluate the performance of the background subtraction methods, we conducted an ablation study using MOG2 and ViBe to generate motion regions, which were then classified using the ResNet18 architecture. Table 1 presents the precision and recall values obtained after applying these background subtraction methods.

Table 1. Precision and Recall Metrics for Classification Based on Background Subtraction Methods.

Metrics	ViBe	MOG2
Recall	0.956	0.857
Precision	0.078	0.034
F1-score	0.144	0.065

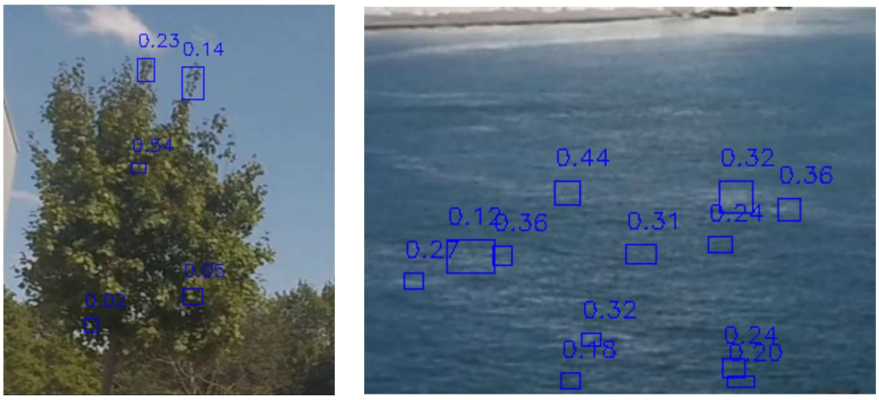
MOG2 demonstrated strong adaptability to gradual illumination changes and dynamic backgrounds, contributing to improved classification performance in less complex scenarios. However, in environments with significant environmental noise, such as water ripples, MOG2 generated a higher number of region proposals, which could lead to increased computational effort during classification. Conversely, ViBe was recognized for its robustness in managing noise in dynamic backgrounds like waving trees. It achieved consistent recall values but occasionally struggled with initialization errors, resulting in minor missed detections, particularly in the initial frames. While MOG2's reliance on Gaussian modeling enables effective handling of general dynamic changes, it often results in over-smoothing in highly dynamic backgrounds or when dealing with small objects. ViBe's history-based approach, on the other hand, makes it more robust to noise and particularly effective in managing stationary movement in water and trees. Figure 4 illustrates the detected motion regions by these two background subtraction methods in challenging scenarios, such as water ripples and waving trees. The figure demonstrates that both methods successfully extract motion regions containing drones in most scenarios. However, MOG2 tends to generate a greater number of region proposals, while ViBe focuses more accurately on relevant motion regions, especially in scenarios involving dynamic background elements.



(**a**) (**b**) (**c**)

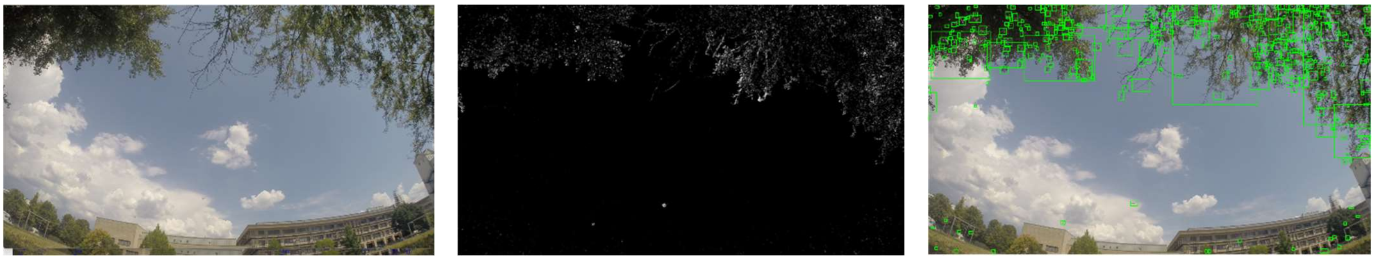
Figure 4. (**a**) Original frames; Detected contours after applying (**b**) ViBe and (**c**) MOG2.

These findings underscore the complementary strengths and weaknesses of MOG2 and ViBe. MOG2 performs well in handling dynamic illumination and motion but may create redundant region proposals in complex environments. Figure 5 illustrates the motion regions proposed by MOG2 in challenging environments such as water ripples and waving trees. Although MOG2 successfully identifies regions of motion, these regions often correspond to areas where motion elements are stationary and belong to the background. This highlights MOG2's tendency to generate an excessive number of motion region proposals in highly dynamic scenarios, which can increase computational overhead and reduce classification accuracy by introducing non-relevant regions. In contrast, ViBe's precision in managing dynamic background noise often provides better focus on relevant motion regions, particularly in scenarios involving stationary background elements. However, in some scenarios where moving stationary elements, such as vegetation, are closer to the camera, and the drone is very small and far away, ViBe struggles to eliminate dynamic background elements from the foreground, as depicted in Figure 6. This occurs because ViBe avoids overly aggressive filtering to ensure that the drone region is not mistakenly removed.



(**a**) (**b**)

Figure 5. Motion region proposal by MOG2 in regions of (**a**) waving trees, (**b**) water ripple.



(**a**) (**b**) (**c**)

Figure 6. (**a**) Original frame (**b**) Foreground mask obtained using ViBe with filtering applied (**c**) Region proposals.

In our experiments, the proposed pipeline using ViBe achieved a recall of 0.956 and a precision of 0.078, while MOG2 yielded a recall of 0.857 and a precision of 0.034. These results underscore that background subtraction can effectively isolate candidate drone regions in dynamic scenes despite the relatively high false positive rate. Compared to contemporary deep learning approaches, such as the YOLO-based system presented by [25], our method offers a notable advantage in computational efficiency by drastically reducing the search space through motion-based region proposals. Although YOLO-based methods deliver robust detection even for high-speed or long-range scenarios, our approach demonstrates competitive performance for small drone detection in cluttered backgrounds, with the potential for further improvement via integration with tracking algorithms or enhanced classification strategies. Overall, our study highlights that leveraging background subtraction as a preprocessing step alleviates the computational burden and provides a complementary solution to fully end-to-end deep learning methods in resource-constrained environments.

The results highlight the performance of MOG2 and ViBe in detecting small drones in dynamic and noisy environments. Through an ablation study, precision and recall metrics were evaluated after applying each background subtraction method, demonstrating their impact on the classification results. The results show that while both methods effectively extract drone regions from complex backgrounds, they also struggle with many motion region proposals and false positives. Despite achieving high recall scores, the precision remains significantly low, highlighting the challenge of accurately distinguishing drones from dynamic background elements and minimizing false detections. This issue arises partly due to low-resolution drone samples in the training dataset resembling non-drone regions. Figure 7 illustrates an example where birds are mistakenly detected as drones. This misclassification occurs due to the similarity in visual patterns and size between birds and drones, compounded by the limitations of the feature extraction process during classification. This highlights the challenge of distinguishing drones from other small, fast-moving objects in dynamic environments, a well-documented issue in drone detection tasks [22]. Additionally, background subtraction methods require a set amount of time to initialize the background model, often leading to inaccuracies in detecting moving regions during the initial frames of videos.

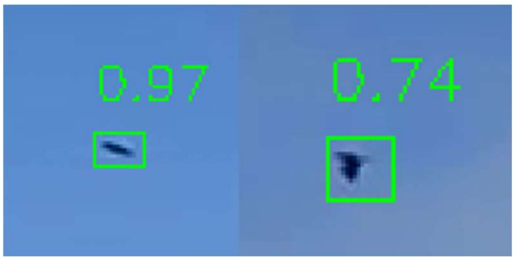


Figure 7. Examples of birds are incorrectly identified as drones due to similar visual patterns and size.

The study highlights a key trade-off: while both methods successfully extract drone regions from challenging backgrounds, they are prone to false positives. This is partly attributed to low-resolution drone samples in the training dataset that resemble non-drone regions. Additionally, the initialization period required by both methods introduces inaccuracies in detecting moving regions during the initial frames of videos. Despite these limitations, the results emphasize that background subtraction methods can effectively extract small drone regions in dynamic environments without compromising resolution. This preservation of resolution is critical for ensuring that small drones are accurately classified, even in highly cluttered and noisy scenes. The results also demonstrate that the selection of a background subtraction method significantly influences classification outcomes, underscoring the importance of tailoring the method to specific environmental challenges. In summary, the results indicate that while MOG2 and ViBe have distinct strengths and weaknesses, both methods prove valuable as preprocessing steps for drone detection in dynamic and noisy environments. ViBe demonstrates superior motion region extraction compared to MOG2, as reflected in its higher precision and recall. This is attributed to ViBe's ability to distinguish drones from background elements better, reducing the number of false region proposals and improving detection accuracy. These findings underscore the importance of further refining background subtraction techniques to mitigate false positives and address initialization challenges, paving the way for more robust drone detection pipelines.

6. Conclusions and Future Work

This study evaluates the performance of MOG2 and ViBe in detecting small drones within dynamic and noisy environments. The results demonstrate their effectiveness in extracting motion regions containing drones, even in challenging scenarios. While both methods face limitations, such as initialization challenges and high numbers of region proposals, they show potential for isolating small objects in complex backgrounds. To enhance drone detection, future work could explore integrating tracking algorithms to leverage spatiotemporal information better. Further advancements might include refining background subtraction techniques, improving initialization processes, improving drone detection in moving camera videos, and adopting adaptive models to reduce region proposals and improve detection accuracy. These steps aim to overcome limitations and establish a more robust and efficient drone detection pipeline.

Although our experiments were conducted on a high-performance workstation (Intel® Core™ i7-7800X CPU, 32 GB RAM, and an NVIDIA GeForce GTX 1080 GPU), the core components of our pipeline are designed with computational efficiency in mind. The initial background subtraction stage (using MOG2 or ViBe) is inherently lightweight, significantly reducing the amount of data processed by isolating motion regions. Furthermore, the subsequent classification step employs ResNet18, a relatively lightweight deep network architecture. These factors suggest that our solution can be adapted for deployment on low-cost hardware platforms similar to the YOLO-based detection and classification system on a Raspberry Pi 4B demonstrated by [26] with additional optimizations such as model quantization, pruning or the use of efficient inference frameworks (e.g., TensorRT or OpenVINO), our pipeline has the potential to operate in real-time even on resource-constrained systems. Future work will include an experimental evaluation of such platforms to validate these adaptations.

Looking forward, future work will focus on several key areas:

- Integration with Tracking Algorithms: Incorporating tracking methods could help leverage spatiotemporal information, reduce false positives, and improve the continuity of detection, especially in scenarios involving high-speed drones or intermittent motion.
- Enhancement of Background Subtraction Techniques: Refining the background subtraction methods, particularly improving initialization processes and adapting the algorithms for moving camera scenarios, could lead to further improvements in detection accuracy.
- Adaptive Model Development: Investigating adaptive models that dynamically adjust parameters based on the scene could help reduce redundant region proposals and better distinguish drones from dynamic background elements.
- Experimental Evaluation on Low-Cost Platforms: Future studies will include comprehensive evaluations of the proposed pipeline on resource-constrained hardware (e.g., Raspberry Pi), which would validate its real-world applicability in mobile and surveillance systems.

Author Contributions

Conceptualization: E.U. and X.H.; Methodology: E.U. and Y.T.; Formal analysis and investigation: E.U.; Writing—original draft preparation: E.U. and P.B.; Writing—review and editing: E.U. and R.Y.; Supervision: R.Y., C.C. and P.B.

Ethics Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

This study uses the Drone-vs-Bird dataset, which is publicly available and can be accessed as described in [27].

Funding

This research received no external funding.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Mohsan SAH, Khan MA, Noor F, Ullah I, Alsharif MH. Towards the Unmanned Aerial Vehicles (UAVs): A Comprehensive Review. *Drones* **2022**, *6*, 147.
2. Heidari A, Jafari Navimipour N, Unal M, Zhang G. Machine Learning Applications in Internet-of-Drones: Systematic Review, Recent Deployments, and Open Issues. *ACM Comput. Surv.* **2023**, *55*, 1–45.
3. Khan MA, Menouar H, Eldeeb A, Abu-Dayya A, Salim FD. On the Detection of Unauthorized Drones—Techniques and Future Perspectives: A Review. *IEEE Sens. J.* **2022**, *22*, 11439–11455.
4. Ubina NA, Cheng SC. A Review of Unmanned System Technologies with Its Application to Aquaculture Farm Monitoring and Management. *Drones* **2022**, *6*, 12.
5. Taha B, Shoufan A. Machine Learning-Based Drone Detection and Classification: State-of-the-Art in Research. *IEEE Access* **2019**, *7*, 138669–138682.
6. Cheng G, Yuan X, Yao X, Yan K, Zeng Q, Xie X, et al. Towards Large-Scale Small Object Detection: Survey and Benchmarks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 13467–13488.
7. Lin TY, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, et al. Microsoft COCO: Common Objects in Context. *arXiv;* 2015. Available online: http://arxiv.org/abs/1405.0312 (accessed on 26 May 2024).
8. Xue Z, Chen W, Li J. Enhancement and Fusion of Multi-Scale Feature Maps for Small Object Detection. In Proceedings of the 2020 39th Chinese Control Conference (CCC), Shenyang, China, 2020. p 7212–7217. Available online: https://ieeexplore.ieee.org/document/9189352/ (accessed on 2 May 2022).
9. Zivkovic Z, van der Heijden F. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognit. Lett.* **2006**, *27*, 773–780.
10. Barnich O, Van Droogenbroeck M. ViBe: A Universal Background Subtraction Algorithm for Video Sequences. *IEEE Trans. Image Process* **2011**, *20*, 1709–1724.
11. Sobral A, Vacavant A. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Comput. Vision. Image Underst.* **2014**, *122*, 4–21.
12. Kalsotra R, Arora S. Background subtraction for moving object detection: explorations of recent developments and challenges. *Vis. Comput.* **2022**, *38*, 4151–4178.
13. Braham M, Van Droogenbroeck M. Deep background subtraction with scene-specific convolutional neural networks. In *2016 International Conference on Systems, Signals and Image Processing (IWSSIP)*. IEEE: Bratislava, Slovakia, 2016; pp. 1–4. Available online: http://ieeexplore.ieee.org/document/7502717/ (accessed on 7 December 2024).
14. Garcia-Garcia B, Bouwmans T, Rosales Silva AJ. Background subtraction in real applications: Challenges, current models and future directions. *Comput. Sci. Rev.* **2020**, *35*, 100204.
15. Ajakwe S, Ihekoronye V, Lee JM, Kim DS. DRONET: Multi-Tasking Framework for Real-Time Industrial Facility Aerial Surveillance and Safety. *Drones* **2022**, *6*, 46.
16. Pansare A, Sabu N, Kushwaha H, Srivastava V, Thakur N, Jamgaonkar K, et al. Drone Detection using YOLO and SSD A Comparative Study. In Proceedings of the 2022 International Conference on Signal and Information Processing (IConSIP), Pune, India, 26–27 August 2022. pp. 1–6.
17. Zou Z, Chen K, Shi Z, Guo Y, Ye J. Object Detection in 20 Years: A Survey. *Proc. IEEE* **2023**, *111*, 257–276.
18. Isaac-Medina BKS, Poyser M, Organisciak D, Willcocks CG, Breckon TP, Shum HPH. Unmanned Aerial Vehicle Visual Detection and Tracking using Deep Neural Networks: A Performance Benchmark. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, 2021. p 1223–32. Available online: https://ieeexplore.ieee.org/document/9607468/ (accessed on 2 July 2023).
19. Lv Y, Ai Z, Chen M, Gong X, Wang Y, Lu Z. High-Resolution Drone Detection Based on Background Difference and SAG-YOLOv5s. *Sensors* **2022**, *22*, 5825.
20. Seidaliyeva U, Akhmetov D, Ilipbayeva L, Matson ET. Real-Time and Accurate Drone Detection in a Video with a Static Background. *Sensors* **2020**, *20*, 3856.
21. Delleji T, Fekih H, Chtourou Z. Deep Learning-based approach for detection and classification of Micro/Mini drones. In Proceedings of the 2020 4th International Conference on Advanced Systems and Emergent Technologies (IC_ASET), Hammamet, Tunisia, 15–18 December 2020; pp. 332–337. doi:10.1109/IC_ASET49463.2020.9318281.
22. Rouhi A, Umare H, Patal S, Kapoor R, Deshpande N, Arezoomandan S, et al. Long-Range Drone Detection Dataset. In Proceedings of the 2024 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 6–8 January 2024; pp. 1–6. doi:10.1109/ICCE59016.2024.10444135.
23. Coluccia A, Fascista A, Sommer L, Schumann A, Dimou A, Zarpalas D. The Drone-vs-Bird. Detection Grand. Challenge at ICASSP 2023: A Review of Methods and Results. In *IEEE Open Journal of Signal Processing*; IEEE: New York City, NY, USA, 2024; pp. 1–15.

24. Stauffer C, Grimson WEL. Adaptive background mixture models for real-time tracking. In Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat No PR00149) 1999. pp. 246–252 Volume 2. Available online: https://ieeexplore.ieee.org/document/784637/?arnumber=784637 (accessed on 8 December 2024).
25. Kim J-H, Kim N, Won CS. High-Speed Drone Detection Based on Yolo-V8. In Proceedings of the ICASSP 2023—2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–2. doi:10.1109/ICASSP49357.2023.10095516.
26. Cherubin S. YOLO object detection and classification using low-cost mobile robot. *Electrotech. Rev.* **2024**, *1*, 31–35, doi:10.15199/48.2024.09.04.
27. Coluccia A, Fascista A, Schumann A, Sommer L, Dimou A, Zarpalas D, et al. Drone vs. Bird Detection: Deep Learning Algorithms and Results from a Grand Challenge. *Sensors* **2021**, *21*, 2824. doi:10.3390/s21082824.