IPSJ Transactions on Computer Vision and Applications

RESEARCH PAPER Open Access

Deep learning-based strategies for the detection and tracking of drones using several cameras

Eren Unlu[1]*, Emmanuel Zenou[1], Nicolas Riviere[2] and Paul-Edouard Dupouy[2]

Abstract

Commercial Unmanned aerial vehicle (UAV) industry, which is publicly known as *drone*, has seen a tremendous increase in last few years, making these devices highly accessible to public. This phenomenon has immediately raised security concerns due to fact that these devices can intentionally or unintentionally cause serious hazards. In order to protect critical locations, the academia and industry have proposed several solutions in recent years. Computer vision is extensively used to detect drones autonomously compared to other proposed solutions such as RADAR, acoustics and RF signal analysis thanks to its robustness. Among these computer vision-based approaches, we see the preference of deep learning algorithms thanks to their effectiveness. In this paper, we are presenting an autonomous drone detection and tracking system which uses a static wide-angle camera and a lower-angle camera mounted on a rotating turret. In order to use memory and time efficiently, we propose a combined multi-frame deep learning detection technique, where the frame coming from the zoomed camera on the turret is overlaid on the wide-angle static camera's frame. With this approach, we are able to build an efficient pipeline where the initial detection of small sized aerial intruders on the main image plane and their detection on the zoomed image plane is performed simultaneously, minimizing the cost of resource exhaustive detection algorithm. In addition to this, we present the integral system including tracking algorithms, deep learning classification architectures and the protocols.

Keywords: Unmanned aerial vehicle, Deep learning, Surveillance

1 Introduction

The exponentially increasing public accessibility of drones has been posing a great threat to the general security and confidentiality. The drone sales have been increasing consistently each year and they are expected to be much more widespread in the future [1]. To highlight the importance of the subject, several incidents with drones in recent years can be given as examples : the alarming security incident around the White House [2], mysterious appearance of multiple drones for several days around nuclear power plants in France [3], horrific near collision of an airliner and a drone near LAX airport [4] and the drone intrusion by an opposition party during a campaign of German chancellor, which has alerted the security officials [5]. Drones are also perfect tools for the illegal smugglers thanks to their low visibility. For instance, recently US officials have seized drug cartels while they were smuggling drugs from Mexico [6] and Chinese police has revealed the illegal trafficking of smart phones from Hong Kong to mainland China [7]. Drones are attempted many times to be used by inmates to smuggle things in and out of the prison [8]. With their potential to carry high explosive payloads, they are becoming a more significant concern for the public and the officials. Lots of more reported security incidents caused by drones in recent years can be found.

Based on these examples, it can be said that detecting and eliminating drones before lethal outcomes is at paramount of interest. This task has been investigated intensively by academia and the industry to commercialize anti-drone systems. Certain systems in the market and architectures proposed by researchers offer autonomous detection, tracking and identification of the UAVs, which is a highly important operational feature. The proposed

*Correspondence: eren.unlu@isae-supaero.fr
[1]ISAE-SUPAERO, 10, Avenue Edouard Belin, 31400 Toulouse, France
Full list of author information is available at the end of the article

systems use either RF signal detection (used for the communication between device and the ground operator) [9], acoustics [10], RADAR [11], LIDAR [12], or common passive optics (cameras) backed by computer vision algorithms [13].

In the following section of the article, we present these methods and discuss their pros and cons. The robustness and effectiveness of optics over other approaches will be highlighted. Among possible computer vision methods, the high performance and the possible affordability of the deep learning for this very specific task will be stated.

In this article, we present an autonomous drone detection, tracking and identification system based on optics and deep learning, composed of a static wide-angle RGB camera platform and a rotating turret, where a lower-angle RGB camera is mounted. The static wide-angle camera serves as a primary aerial object detection, where drones can be detected at relatively long range (up to $\sim$ 1 km), even as small as few dozens of pixels. These detections are tracked on the image plane of the wide camera and the ones which show specific motion and visual signatures are inspected by the narrow-angle RGB camera on the rotating turret. For detection of possible drones on the wide-angle camera's image plane, a lightweight version of YOLO deep learning algorithm is used, which has recently become a popular choice thanks to its robustness and speed [14]. This lightweight architecture is extensively trained for the detection of drones, also as small as 6×6 pixels, for backgrounds similar to the operational one. With diverse, well chosen, and augmented datasets ($\sim$ 10,000 images), we are able to use the same architecture for detection on two different image planes : wide -angle static RGB camera and narrow-angle RGB camera on the turret. By using a single lightweight YOLO detector on two frames with properly chosen different thresholds, we can economize GPU memory significantly. We have chosen to separate the detection and classification tasks to two different YOLO architectures, which yields lesser consumption of computational resources. One of the genuine contribution of this work is the introduction of a method, where the frames of the narrow-angle cameras are overlaid on the wide-angle camera's frame, thus detection can be performed simultaneously. This allows the uninterrupted tracking of the aerial objects on the main image plane, while system can identify the possible threats with the rotating turret's camera.

For tracking purpose, where the intruding airborne target is being tracked on wide-angle image plane, and verified by rotating zoom cameras; we have introduced a novel algorithm called *Target Candidate Track (TCT).* This is a collection of several policies, which mixes the motion signatures on static image plane and visual signatures on zoomed cameras; in order to prioritize the following of multiple candidates and yield a precisely defined duration for the assessment of possible threats. The main idea is to make sure that a potential target is not being followed for a long duration with rotating cameras, if it is a false alarm such as bird or commercial airplane, as this case might cause the missing of a real threat.

We present a fully autonomous optics-based UAV detection architecture with two cameras, where one of them is mounted on a rotating turret with low-angle lens for detailed inspection of certain flying objects. In this paper, after presenting the proposed methods for drone detection in the literature, especially the ones using the optical approach, we introduce the general scheme of the system. The detection of small aerial intruders on the main image plane (static wide-angle camera) and tracking of their movements is explained in the following section. Next, our policy to inspect and track suspicious aerial objects by low-angle camera is presented.

2 Drone detection and tracking

The proposed methods in market and academic literature can be grouped by the nature of their equipment : RADAR, LIDAR, acoustics, RF signal detection, and optics. RADAR technology has been used for decades to detect aerial vehicle; however, conventional ones are not feasible to detect small commercial UAVs. Also, they are flying at relatively much lower velocities, which decreases the Doppler signature [15, 16]. Even though such examples as [11, 15, 17, 18] exist, especially in K, S, X-band and with the exploitation of Doppler effect, generally they fail to classify other aerial objects such as birds and the background clutter due their increased sensitivity for this particular case [16]. Hence, RADAR technology has not been considered as an effective solution counter drones, especially for autonomous configurations. On the other end, LIDAR is a relatively new technology to be used for drone surveillance task, thus only few proposals such as [11, 19, 20] exist in the literature. Its feasibility and cost effectiveness is still questionable due to voluminous data output and sensitivity to the clouds etc.

Probably the most popular approach in the market for drone detection is the RF signal analysis, which intends to capture the communication between the drone and the ground operator [16]. However, the main issue with this approach is the fact that the drone may be operated without ground control at all but with a pre-programmed flight path.

Acoustics has been used also to detect drones by employing microphone arrays [11, 21]. The aim is to classify specific sound of rotors of drones; however, they fail to achieve high accuracy and operational range. Maximum range of audio-assisted systems stay below 200–250 m. Another disadvantage is the non-feasible nature of the system in urban or noisy environments such as airports.

2.1 Optical approach

Among other approaches for drone detection which are presented previously, optics distinct itself. It can be said that, optics has been regarded as the most convenient way to tackle this challenge due to its robustness, accuracy, range, and interpretability [22]. Hence, we observe a tendency to include cameras as the only or at least on of the sensors of the proposed system in the market. In addition to the aforementioned advantages, using optics has a supplementary advantage with the recently booming deep learning computer vision algorithms. With culminating open source data (i.e., images, videos), developed algorithms and affordable GPU resources, using deep learning for computer vision based on convolutional neural networks (CNNs) has already become the de facto approach for detection and recognition tasks [23, 24]. The breakthrough coming with the usage of deep learning in computer vision has already started to revolutionalize the industrial and scholar community. Therefore, one can deduce after synthesizing the listed advantages that using optics with deep learning is the most convenient approach to the drone detection challenge.

We can already observe that most of the articles which are published in recent few years, proposing to use computer vision for autonomous drone surveillance task uses deep learning. Certain examples of these papers are [25–27], where all three of them use CNNs to detect and classify drones. The usage of optics is also a widespread application for commercial autonomous drone surveillence systems, such as [28–30]. Therefore, we have also chosen to follow an approach where RGB cameras are used with deep learning algorithms. The instantaneous detection and identification methods are widely addressed in the literature in the context of person/pedestrian recognition task such as [31, 32].

3 Main architecture of the system

As illustrated in Fig. 1, our system is composed of a static wide-angle camera placed on a stationary platform with adjusted angle and position according to demand, a rotating turret where a narrow-angle, zoomed RGB camera is mounted on it and a main computational unit (a Linux PC or embedded platform with NVIDIA GPU) is connected to them via ethernet. Both RGB cameras are high performance industrial cameras with same specs and model, except the one on the rotating angle carries an external professional zooming lens. The cameras are capable of delivering 2000 × 1700 pixels of resolution with approximately 25 FPS. The wide-angle camera has a lens with 16 mm of focal length, which corresponds to approximately 110° of field of view (FoV). The camera on the rotating turret has a 300 mm lens, thus having a diagonal FoV around 8.2°. One can see that the narrow-angle camera has a zoom capability around more than × 35, as we wish to detect drones as small as possible and at the same time be able to identify them at large distance.

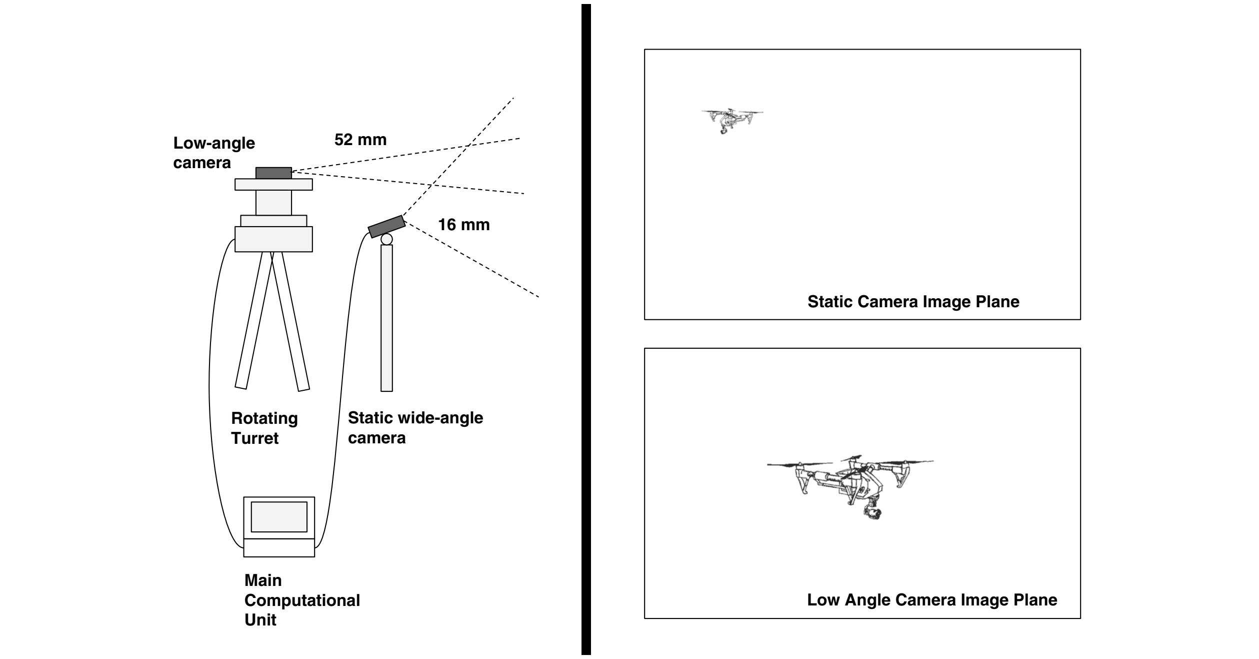

Fig. 1 The general setting of our drone detection system. Our drone detection system is composed of mainly three interconnected components; a static wide-angle camera, a rotating turret equipped with a low-angle camera and a Linux-based main control computer

Based on a modified lightweight YOLOv3 architecture, we detect the small intruders. At this first stage, false alarms up to a degree is acceptable, where they are tracked and based on their movements and visual signatures they may be inspected by rotating the turret toward it and analyzed with the narrow-angle camera.

Our system uses python as the main programming language due to its versatility and high performance. The deep learning algorithms for detection and classification are based on darknet YOLO architecture, which is written in C language but can be wrapped in python [14]. The deep learning algorithms are executed in the GPU of the main computational unit, which is a 2 Gb memory NVIDIA Geforce K620.

Initially, system tries to detect small UAVs which intrude and observe the horizon with the wide-angle static camera. In order to be coherent with the square input shape of the YOLO architecture, firstly we reshape the raw images coming from the wide-angle camera to 1600 × 1600 pixels. We have chosen to use YOLO for the detection and classification due to its high performance and processing speed, which is also preferred by many other researchers [26]. Different than other detection convolutional neural networks, YOLO uses a regression-based approach to locate objects on an image, which makes the process much more rapid. In the last version of YOLO (YOLOv3), a new concept called *upsampling* is introduced which boosts the small object detection performance drastically [33]. In this version, in the later layers of the architecture where the image plane size is diminished, an upsampling layer is introduced. The feature matrix is upsampled with ×2 and it is connected to a previous layer (a layer which has the same feature dimension with the new upsampled one.) via a *route layer*. As it is clear, this is an attempt to detect small patterns by rescaling the feature vector in positional axes.

The default version of YOLOv3 contains a total of 102 layers, where there is three detection layers, each for a scale. In other words, after the first detection layer the feature matrix is upscaled two times. The default version has two different options for input size 418 × 418 and 627 × 627. The detection layer of YOLO can be seen as a regression operation, which probabilistically locates the objects. Details of this default architecture can be found in [33].

We have found the default architecture of YOLOv3 is quite resource exhausting both in time and memory with our limited GPU. Especially, it does not allow to attain necessary FPS rates for the simultaneous detection and tracking of our system. Therefore, we have decided to use a lightweight version of the system. After an extensive experimentational campaign, we have found that narrowing the architecture rather than shortening it is a more reliable approach. In other words, minimizing the number of filters while keeping the number of layers same is found to be more effective for the very specific case of drone detection problem. As mentioned previously, detection and classification procedures are separated for effectiveness in our system [34]. Therefore, detection algorithm is trained to detect only drones; however, any miss detection is considered acceptable up to a degree, as the system tracks and inspects in detail with zoomed camera if necessary.

The aim of the design of our lightweight YOLO detector is to attain the best miss rate and false alarm performance for small intruding drones, while having an acceptable FPS and GPU memory usage. It was concluded that if the filter count can be minimized, the input shape can be increased to 832 × 832. The memory resource and computational time loss coming from increased frame size is decided to be more preferable, rather than having higher number of filters. Therefore, at the end, the best architecture for the lightweight YOLO detector is concluded to be as illustrated in Fig 2. As it can be seen, except for the last scale layers (the layers after the second detection layer), which are more responsible for detecting the smallest size objects, number of filters of each layer is set to 16. The number of filters of last scale layer are more than 16, in an attempt to boost the small object detection performance. Also note that, the number of convolutional filters before each detection layer is set to 18, which is a constrained as we have only a single class to detect due to the mechanism of the YOLO architecture [14]. Any number of filters lower than 16 found to be insufficient, while more has not been contributing to performance drastically when the resource constraints are considered. The performance of the detection architecture is presented in the Section 5 of the article.

3.1 Tracking on main image plane

After the lightweight YOLO detector locates a intruder, it is immediately assigned a new ID number and be tracked. The detection and tracking is performed in a frame by frame manner. Thus, it is an asynchronous process, as computation times may vary frame to frame; however, we have observed this effect is minimal. In other words, every time a frame is processed by detecting the objects on main plane, the tracking and classification operations follow it. When the operation for that frame is finished, the next frame is processed. Note that, a python thread is responsible of grabbing raw frames from the cameras with 25 FPS rates, thus the latest available frame is processed when demanded. As we would like to be able to detect objects as small as few pixels, motion-based tracking is adopted rather than visual signature-based tracking [35]. For this purpose, we have decided to use a Kalman tracking algorithm based on an already existing python library, optimized for algebraic operations [36].

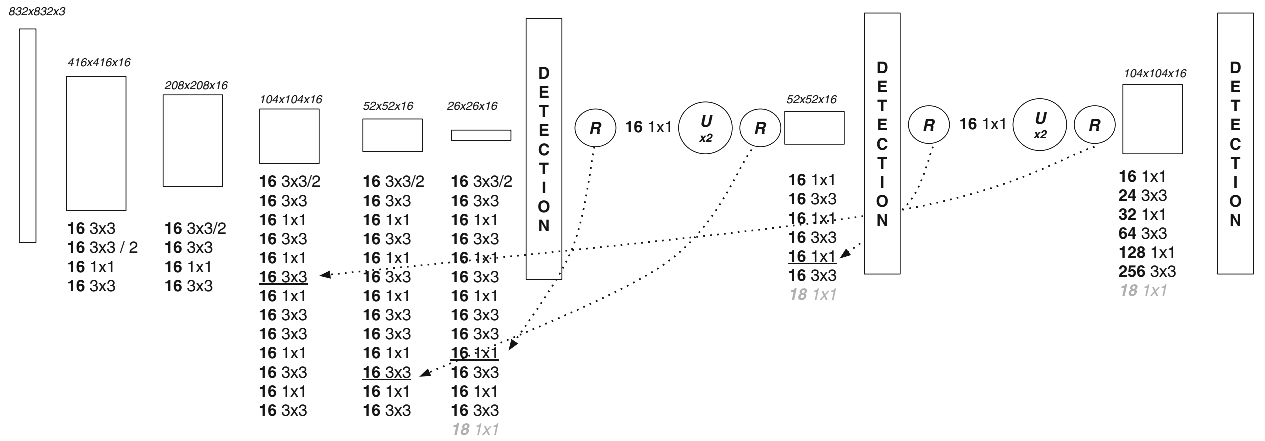

Fig. 2 Proposed lightweight YOLOv3 detector. Different than the default architecture, we prefer to use 812 × 812 × 3 input size to better detect the small drones. The circles with R denotes the *route layer* and the U denotes the *upsampling layer*. Note that, the previous layer filters, where the upsampled feature matrix is routed is underlined. Also, note that the layers where the positional dimensions of the feature matrix is divided in to two by proper strided max pooling are shown

The aim of tracking can be divided in to two as first being able to estimate the position of the object when the detection fails and being able to associate a new detection to an existing track in the previous frame [37]. In order to associate detections in the new frame to the existing tracks or assigning as new tracks, we use the well known Hungarian algorithm where the cost is the distance between the centroid of the bounding boxes [38].

3.2 Notion of target candidate track (TCT)

In this section, we introduce a concept which is called target candidate track (TCT), which refers to the tracks on the main image plane, where algorithm decides to inspect by the zoomed lens. Therefore, the rotating turret is directed toward the instantaneous location of the track. At an arbitrary time, the system only has a single TCT, where no new TCT is inspected until the current one is unassigned. So TCT is one of the tracks on the main image plane, which are being tracked by Kalman filtering, where it is also controlled visually with zoomed camera by rotating the turret.

3.2.1 Assignment of TCT from tracks on the main image plane

Each tracked object is followed at least for $T_{\min}^{\text{pre-track}}$ frames, where its instantaneous positions are recorded (centroid of the bounding box). These displacement information are used for the decision of whether the tracked object should be checked by the zoomed camera on the turret. For this purpose, we propose to analyze the movements of the tracks in last T_{mov} frames. As illustrated in Fig. 3, the Euclidean distances between last T_{mov} centroids and the last position is summed for each object, where we refer as the *cumulative Euclidean displacement*, ΔD. We use this metric as the motion signature. The main reason behind this decision is to favor the linear movements, which may potentially represent an outside intruder toward a direction. In Fig. 3, one can see that the plane in red bounding box,has a much larger movement score compared to others due to its linear movement. The calculated total displacement of each track on a frame in last ΔD is summed, and each track is assigned a motion score after dividing their ΔD with this sum. Thus, by normalizing the values, we can calculate a motion signature score for each track as a scalar between 0 and 1 : c_{mov}.

We also incorporate the visual information in the process of this decision; however, only if the size of the tracked object on the main image plane is larger than a value, $A_{\min}^{\text{pre-track}}$. This value is in pixels, where the minimum value of width or height of the bounding box is considered. The metric for the visual signature is the confidence score between 0 and 1 coming from the YOLO detection. Therefore, it is not calculated with a separate process. On the contrary, even the size is larger than $A_{\max}^{\text{pre-track}}$ pixels, it is considered as $A_{\max}^{\text{pre-track}}$. Also, if the size of the track is larger than $A_{\max}^{\text{TCT}}$ pixels, it can be counted as a TCT, but the zoomed camera is not rotated toward it due to the fact that, it would not even fit in the focal plane of the very low-angle camera. Hence, its visual appearance as a TCT object is checked based on the main image frame.

We use only motion signatures when the tracked object's size is very small and increment the utilization of visual signatures proportional to size. At the end, at each frame an overall score (c_{overall}) is calculated for each track from its visual and motion signature considering its size as

$$c_{\text{overall}} = \beta c_{\text{vis}} + (1 - \beta) c_{\text{mov}} \tag{1}$$

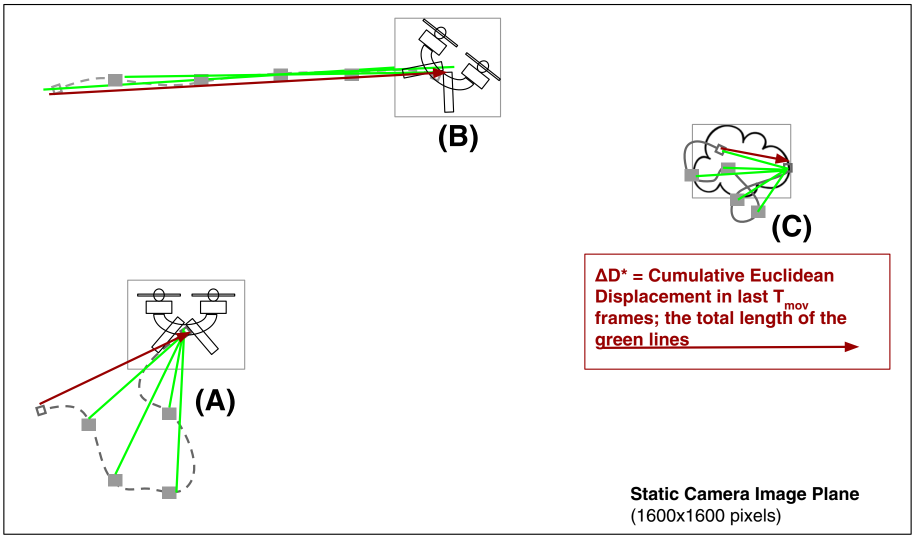
(B)
(C)
ΔD* = Cumulative Euclidean Displacement in last T_{mov} frames; the total length of the green lines
(A)
Static Camera Image Plane
(1600x1600 pixels)

Fig. 3 Detection and TCT tracking algorithm. The movement score that we use for evaluating tracks to decide or prioritize to assign as TCT is calculated as the total euclidean displacement in last T_{mov} frames. In other words it is the total length of green lines in this figure, where $T_{mov} = 6$ for illustration purposes. The very last position and the position T_{mov} frames before are marked with empty squares, while centroids on other frames are marked with filled squares

where β is a scalar between 0 and 1 determining the importance of visual and motion signatures proportional to the instantaneous size of the tracked object:

$$\beta = \frac{A - A_{\text{min}}^{\text{pre-track}}}{A_{\text{max}}^{\text{pre-track}} - A_{\text{min}}^{\text{pre-track}}} \tag{2}$$

where A is the instantaneous size.

c_{overall} determines if a track should be inspected by rotating the low-angle camera (i.e., assigned as TCT) and if there are multiple candidate tracks which one we should first assign as a TCT. This overall score is a scalar between 0 and 1. The tracks having c_{overall} smaller than $c_{\text{TCT}}^{\text{min}}$ are not considered as candidates for being TCT. Hence, if there is no TCT on a given frame, and there are multiple tracks on the main image plane which fulfill the condition to be a TCT, the one with the maximum c_{overall} is assigned as a TCT, if and only if it was not assigned as TCT before (It means, this track was already inspected with zoomed camera and decided to be a non-threatening object) (Table 1).

3.2.2 *Treatment of an assigned TCT*

As mentioned in the previous section, there can only be one TCT object at a given instance. When a specific track with a specific ID is assigned as TCT, the turret is immediately rotated toward the instantaneous position of it. Then the procedure for the treatment of a TCT begins, where frames coming from zoomed camera is evaluated. A TCT is inspected for at least $T_{\text{min}}^{\text{TCT}}$ frames, in order to allow a minimum fair time to decide whether it is a drone or not. The tracking process on the main image plane of the static wide-angle camera, which is explained in previous section continues normally, even during the presence of a TCT.

A TCT is evaluated in periodic windows temporally, where each time unit is $T_{\text{window}}^{\text{TCT}}$ frames long. The frames coming from the zoomed camera is processed simultaneously with the main image plane for detection with the

Table 1 Descriptions and chosen values of various parameters of the system

Symbol	Description	Value
$T_{\text{min}}^{\text{pre-track}}$	Minimum number of frames for a track to be considered as a TCT	18 frames
T_{mov}	Number of last centroids and frames to calculate the movement score	16 frames
$A_{\text{min}}^{\text{pre-track}}$	Minimum width of height of a track bounding box to be considered as a TCT	32 pixels
$A_{\text{max}}^{\text{pre-track}}$	Maximum width of height of a track bounding box of TCT, where main image planes is used for visual signatures.	256 pixels
$T_{\text{window}}^{\text{TCT}}$	The length of a periodic window for TCT	8 frames
$T_{\text{min}}^{\text{TCT}}$	Minimum number of windows for TCT to decide to whether unassign or continue	12 windows
α	Moving average filter parameter	0.85
$K_{\text{min}}^{\text{TCT}}$	Minimum overall score of TCT to decide to continue	0.85

same YOLO architecture, thanks to the novel algorithm we present in the next section. Each frame, the candidate objects (if exists any) are detected and located on the zoomed camera frame. After detection are located, they are cropped from the frame and classified by the separate YOLO classifier algorithm. Note that as explained previously, if the size of the TCT is larger than a threshold, its appearance on the main image plane is used.

As mentioned previously, the detection and classification is performed separately with two different YOLO architectures, even the nature of YOLO permits for combined detection/classification. The reason behind this is the fact that increasing input size exhausts quadrically more GPU resources. Hence, we have shown that following a *divide and conquer* strategy is better in this context, where the objects are detected with a lightweight detector and classified with a separate, more sophisticated architecture, whose input size is 64 × 64. We have found that this input size attains a sufficient compromise between memory consumption and accuracy. The layers of the classifier architecture can be seen in Fig. 4. Based on our field observations, we have designed and trained to system to analyze 4 different classes : *drones*, *birds*, *airplanes*, and *background clutter*. We have observed that considering helicopters confuse the algorithm due to their visual similarity with certain types of drones. It was concluded that, it is better to exclude helicopters for our specific field case, where they appear much more less frequent compared to commercial airplanes and birds. Also note that, jet fighters were not included in dataset due to the same reason. One can find high number of open-source videos and images, whereas the most time-consuming part is the labeling of the dataset. As a future enhancement, we would like to add semi-autonomous approaches such as in [39], where a deep learning-based model is trained initially with a smaller, human-labeled dataset and can automatically generate highly accurate labelings for voluminous datasets.

For each frame, the detection and classification is performed for the current TCT, and the confidence scores of classification of each detection is recorded. If there is no detection, the classification score is set to 0 for drone recognition. Note that, there may be multiple overlapping detections for the same objects. For each temporal window $\left(T_{\mathrm{window}}^{\mathrm{TCT}} \text{ frames}\right)$, the maximum drone classification score, k_t^{TCT} is evaluated among all detections in that window. The motivation behind this scheme is the fact that due to rapid motion of the zoomed camera and the object, the object may not be present in the image plane. Also, degrading effects due to blurry frames caused by motion is also compensated. Another advantage of this scheme is the chance of evaluating different poses of the same object, where the maximum score among them shall give a more accurate result.

The maximum scores of each time windows are averaged by a regular moving average filter as follows :

$$K_t^{\mathrm{TCT}} = \alpha K_{t-1}^{\mathrm{TCT}} + (1-\alpha) k_t^{\mathrm{TCT}} \tag{3}$$

where α is a scalar determining the effect of the history. If the age of a TCT is larger than $T_{\mathrm{min}}^{\mathrm{TCT}}$ frames and its K_t^{TCT} is smaller than $K_{\mathrm{min}}^{\mathrm{TCT}}$; the object is considered not to be a drone and it is unassigned as a TCT. Note that this object shall continue to be tracked on the main image plane; however, it would not be assigned as TCT another

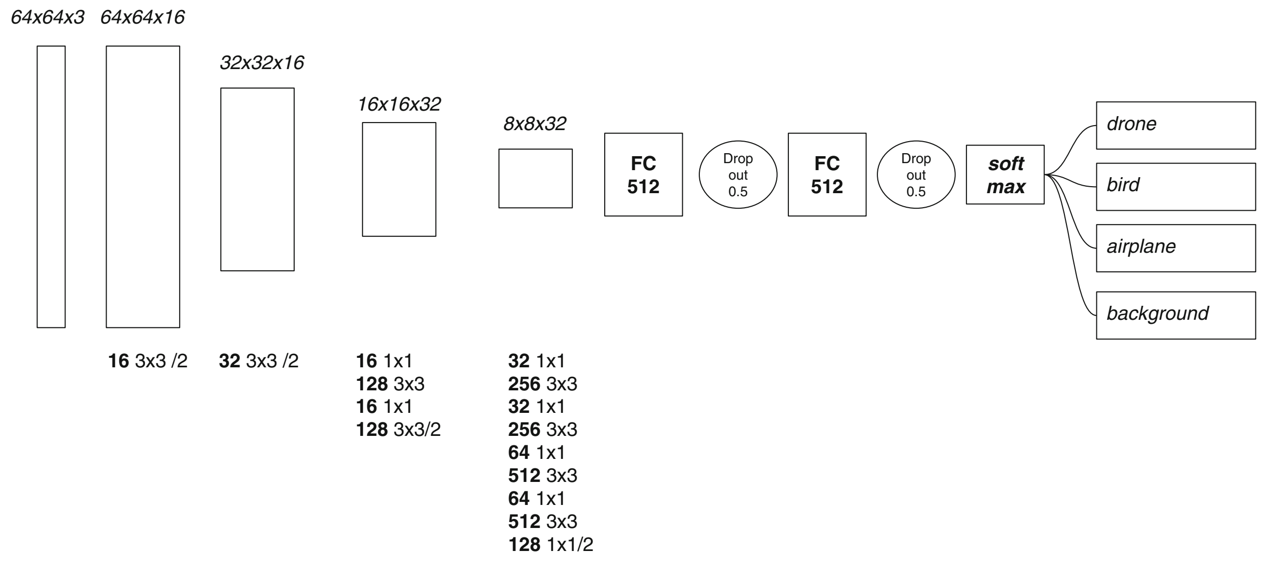

Fig. 4 Proposed lightweight YOLOv3 classifier. The YOLO architecture used for classification purpose with 64 × 64 input size. Note that, classification of feature vectors are performed by 2 consecutive fully connected layer with 512 neurons with 0.5 dropout between them

time as it has been checked before. Then, if there are other candidate tracks, which has the highest c_{overall} is assigned as TCT, immediately.

4 Detection on multiple overlaid images with a single architecture

One of the most pioneering aspect of this article is the presentation of a new kind of scheme to locate objects on multiple frames with a single deep learning detector. When there is no TCT, as mentioned previously, only a single frame coming from the wide-angle static camera is evaluated. At this configuration, the YOLO detector tries to locate objects only on this image. However, when there is a TCT, we have to process the images coming from the zoomed camera also. We can do this by using two separate detectors for two frames at the same time or using the same detector architecture and process the frames consecutively. However, at the first case, we consume the GPU memory and we lose time in the latter case, by the order of 100%. In this system, the main temporal bottleneck is execution time of YOLO detector (0.07 s), which mostly determines the resulting FPS. Therefore, this improvement is primordial for a reliable operation.

Our GPU memory was not sufficient for the instantaneous operation of two copies of the YOLO detector. And we have observed that halving FPS significantly reduces the performance of the surveillance. Therefore, we suggest to locate objects on a single-montaged frame, where the image coming from the low-angle camera is overlaid on the main image plane. This way, the detector detects drones on this single image, and based on the positions, we can locate the object in the image planes of both cameras. If detections are on the borders between the montaged frames, the detection is associated to the camera where most of its area resides.

Note that, this overlaid image is only used to locate the bounding boxes of the objects. In other words, based on the location of the detection bounding box on the small image in the corner (low-angle camera frame), with proper scaling of the coordinates; we can crop the image from the raw frame of the low-angle camera (thus, resolution loss does not occur due to downsizing for overlaying). Next, this cropped images can be classified by the YOLO classifier architecture.

Based on our experimentations, it was concluded that the overlaid image coming from low-angle camera shall be 1/4 of the main image plane. Note that, even the raw main image plane size is 1600 × 1600 pixels, it is automatically downsized to 832 × 832 pixels, due to YOLO architecture. However, when regions are being cropped, raw images are used with proper transformation between coordinates of different scales. So, even small objects' bounding boxes can be acquired with high resolution. Also note that, the size of the small image to be overlaid can be determined on the fly, according to changing demands (Fig. 5).

5 Experimental results

We have tested our system in field and with test videos in order to stress the performance under different conditions. First of all, let us examine the detection accuracy of the proposed lightweight YOLO architecture and compare to the several conventional object detection methods. As mentioned previously, we would like to be able to locate the intruding small drones at large distance; therefore, the performance under these conditions are evaluated in depth. As a conventional measure, any video used

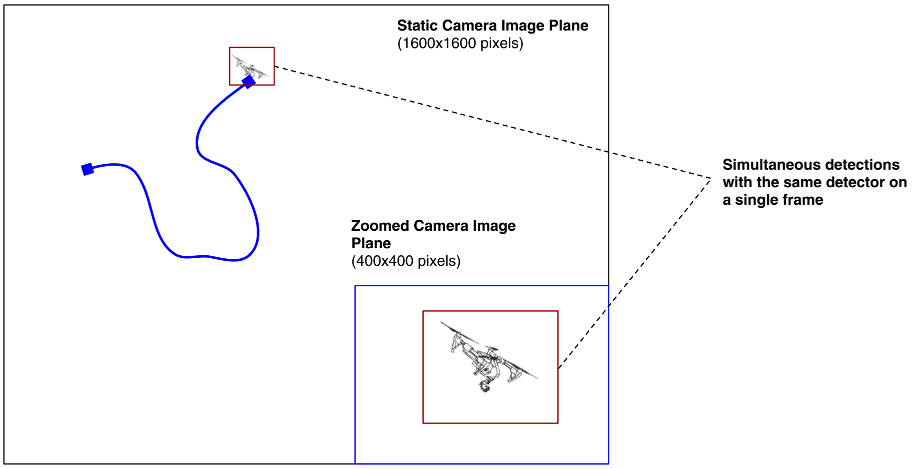
Static Camera Image Plane
(1600x1600 pixels)
Simultaneous detections with the same detector on a single frame
Zoomed Camera Image Plane
(400x400 pixels)

Fig. 5 Proposed montaged multiple image detection. The frame coming from the zoomed camera is rescaled and overlaid on the main image plane. The position of this overlay is determined according to presence of tracks. We try to enforce to overlay the new image, as far as possible from existing tracks, in order to allow the system to continue tracking. A similar attempt has not been proposed in the literature to the best of our knowledge

for experimentation has not been used for training or validation process. We have also tried to test the system with as many different types of drones, birds, and commercial airplanes.

To highlight the performance of the lightweight deep learning detector, we compare it to two conventional object detection approaches. First one is the cascaded Haar feature classifier, which works in a sliding window manner [40]. We have designed a 20 layer extended set Haar classifier with Adaboost algorithm. This cascaded algorithm is also trained with the same dataset for the YOLO detector. As a second counterpart, we have used a Gaussian mixture model (GMM) background subtraction algorithm, as the wide-angle camera is static [41]. The background subtractor is pretrained for 400 frames. The results for overall accuracy and precision are based on 800 frames from 20 videos (40 from each video, non-consecutive frames).

For instance, in Fig. 6, there exists two small birds flying from left hand side toward right hand side (two small green bounding boxes on the upper half of the image) and a small drone approaching from horizon (small green bounding box below). The green, red, and blue bounding boxes correspond to detections by our lightweight YOLO detector, cascaded Haar detector and background subtraction algorithm, respectively.

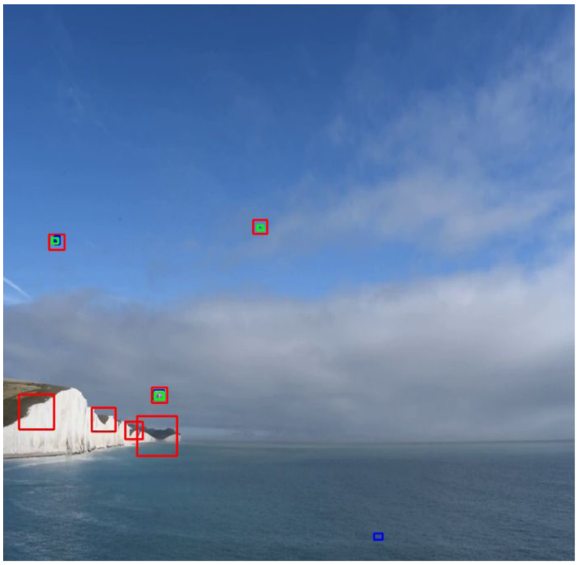

Fig. 6 Comparison of detectors in a marine environment at far distance. Detections produced by lightweight YOLO architecture (green), cascaded Haar classifier (red) and Gaussian mixture model background subtractor (blue). There exists two birds, flying from left hand side toward right hand side on the upper half of the image and a drone approaching from horizon on the bottom half of the image, all three enclosed by green boxes. The footage is provided by [43]

One can see that, lightweight YOLO architecture has detected 3 aerial objects one drone and two birds, while producing no other false alarms. Considering the complexity of background, this is a remarkable performance. Even with high number of layers, cascaded detector has produced high number of false alarms. The background subtraction method also produces false alarms caused by the motion of sea waves. Also note that, the enclosing of bounding boxes around three objects by lightweight YOLO detector is much better compared to others. This is thanks to the semantic context which can be retrieved by deep learning and regression-based approach, even with the minimal number of filters.

Due to limited space, more visual comparison of detectors with images is not given; however, the overall performance for similar settings can be examined from Table 2. As it can be seen from the results, while YOLO produces no false alarms, their counterparts show a remarkably low performance. Also, we can see that YOLO has a slightly less true positive rate. Considering the fact that even they system misses detections on certain frames, it may be recovered within the tracking-based framework. Even with minimal number of filters, the system can attain very promising accuracy for a primary detection mechanism thanks to the semantic nature of deep learning-based YOLO algorithm. These results are important in the sense of compromising width of the architecture, we can still manage to achieve substantial results; if detection and fine-grade classification is separated between two different architectures (Figs. 7, 8, and 9).

Especially in the context of small airborne target detection, where most of the time the background of the target and its periphery shall be uniform such as sky. This situation contains a high-degree semantic context, where deep learning algorithms can grasp. In the small target detection task, we have observed that if upscaling feature of new YOLOv3 is used, width (i.e., number of filters) is less important compared to the depth, if and only if detection and detailed object classification tasks are separated. In addition, to further increase the small object detection accuracy, one can increase the number of filters in the last scale. Even though classification is performed with a different architecture, another advantage of this detector

Table 2 Overall approximate true positive and false alarm rates of three different detectors, for different settings and environments

	True positive	False alarm
Lightweight YOLO	0.91	0
Cascaded Haar	0.95	0.42
GMM back. sub.	0.98	0.31

The results are based on 800 frames from 20 videos (40 from each video, non-consecutive frames.)

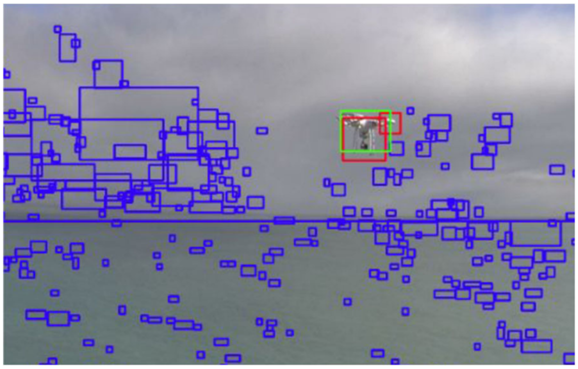

Fig. 7 Comparison of detectors in a marine environment at low distance. Performance of detectors for low distance case in same marine environment. For instance, even very small camera vibrations or sudden illumination perturbations can cause the drastically . The footage is provided by [43]

scheme is the resulting confidence score, which can be used as a metric in the system.

The operation of multiple overlaid images detection with a single detector is shown in Fig. 10. As it can be seen, very small drone present in the main image plane is detected, even in the presence of low visibility and complex background. At the same time, we also detect the drone on the secondary image, where we do not have to run the algorithm again. We have explained in previous sections that this scheme is only used for bounding box localization where we crop the region of interest from the raw image after rescaling coordinates.

Tracking of the drones with the scheme explained previously is shown in Figs. 11, 12, 13, 14, and 15, where the path they have followed in previous frames are traced. The detector and Kalman tracking works with high accuracy, even when the size of drone is smaller than few dozens of pixels. Kalman tracking also makes sure that the trace of the object is never lost by periodic prediction. Note that, in these examplary three frames, there

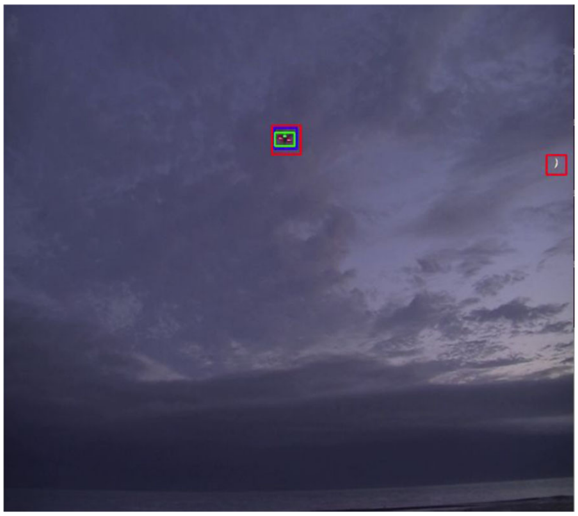

Fig. 8 Comparison of detectors in a nocturnal setting. Performance of detectors in a nocturnal marine environment, with very low visibility. The footage is provided by [43]

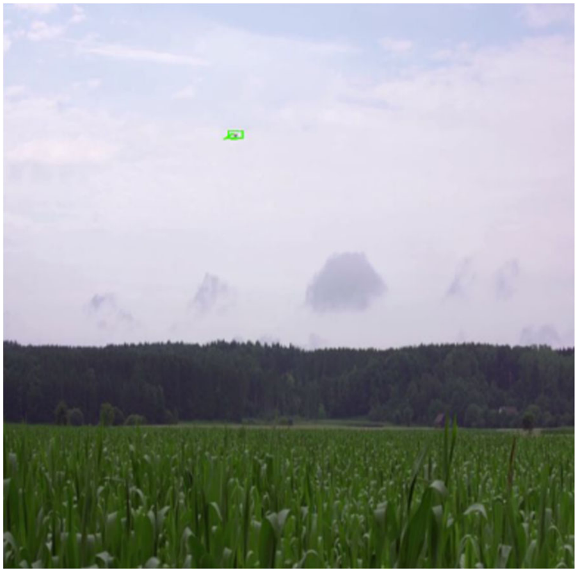

Fig. 9 Performance demonstration of our YOLOv3 detector on a open field at large distance

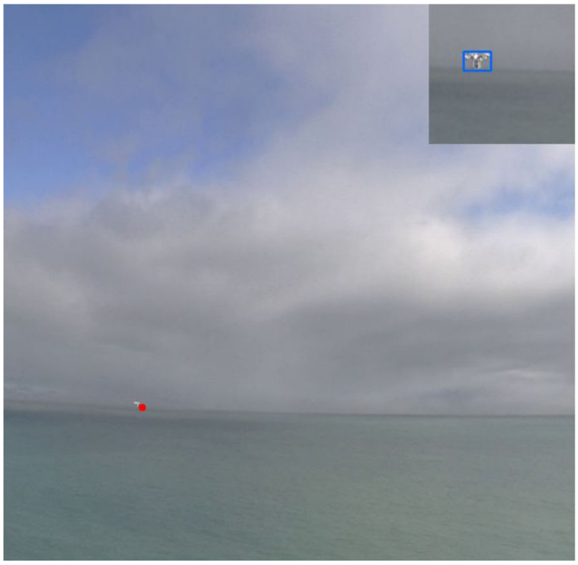

Fig. 10 Detecting on multiple overlaid images. Simultaneous detection on a single overlaid frame composed of images coming from wide-angle camera and lowangle camera, with a single lightweight YOLO architecture. Also note that, even for very small sizes and difficult visibility conditions, the system can detect the drone on the horizon, while producing no false alarm [43]

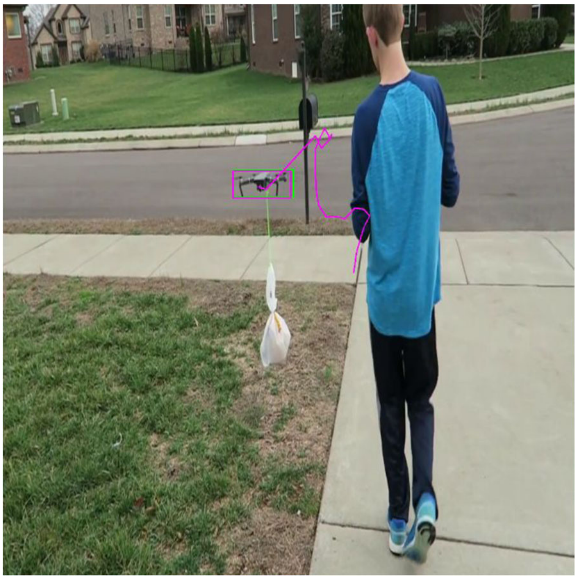

Fig. 11 Performance of tracking. Lightweight YOLOv3 detector and TCT-based tracking perform well even with complex background

is no false alarms being tracked by the system despite highly complex background, low visual signature, and different models. Preference to favor linear direction with the methods explained previously can be considered consistent for these examples also.

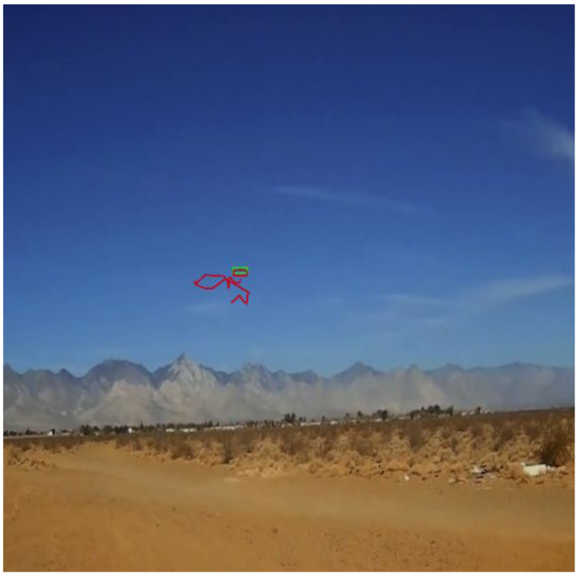

Fig. 12 Performance for small size in open sky. Lightweight YOLOv3 detector and TCT-based tracking can also perform well for drones at high range with almost non-existent false alarms

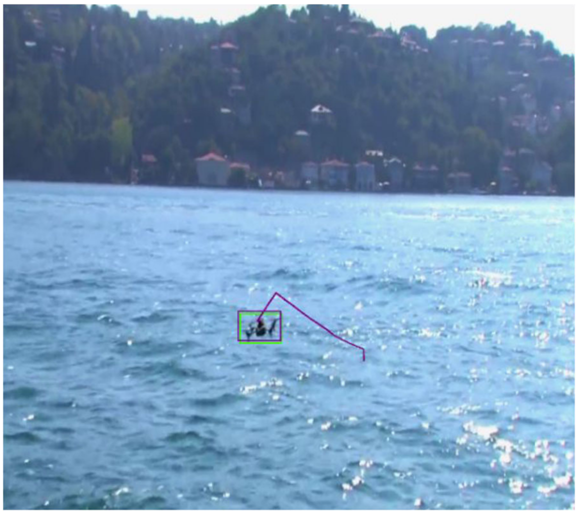

Fig. 13 Performance of tracking in a marine environment with complex background patterns. Lightweight YOLOv3 detector and TCT-based tracking perform with high performance in a marine background, where there is no false alarms

6 Conclusion

We propose an end to end, complete autonomous drone surveillance system based on RGB cameras and computer vision. Considering the needs of this two camera system, a detailed framework has been built, consisting of algorithms and policies for detection, tracking, and recognition. The proposed scheme can be used partially

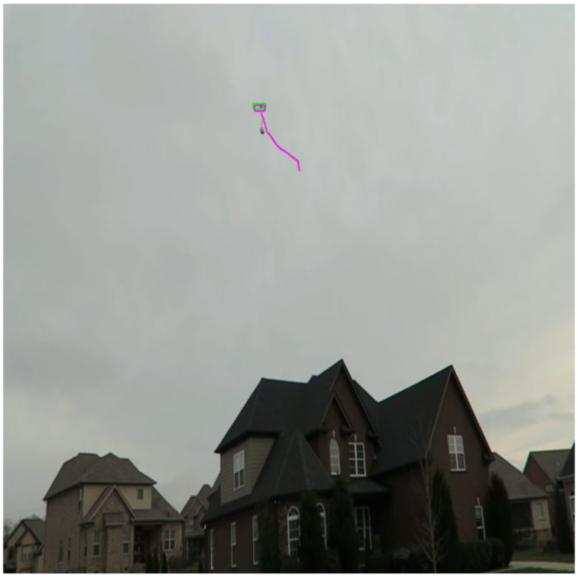

Fig. 14 Performance for small size in an urban setting. Our proposed system can still provide good results, with almost zero false alarm rate, even at large distance in an urban setting, where complex shapes are present

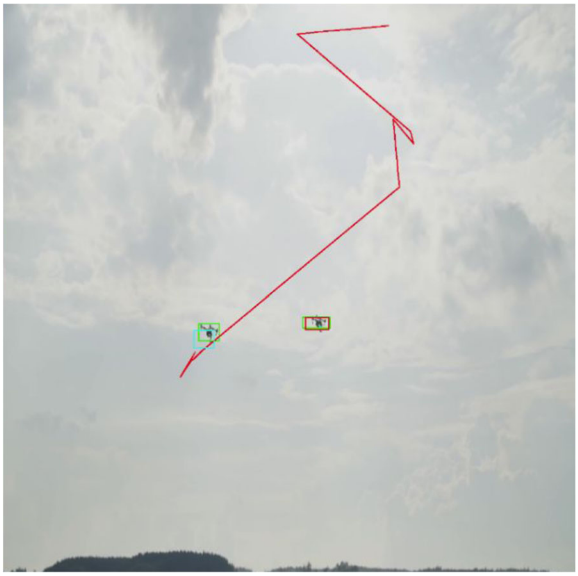

Fig. 15 Detection and tracking for multiple targets. System continues to operate with accuracy even with presence of multiple drones

or fully for other video surveillance tasks rather than counter drone activity, after proper modifications. Our system, in collaboration of a static wide-angle camera and a rotating low-angle camera, has been proven to provide plausible results based on simulations and field tests. One advantage of the system is the limited GPU memory requirement which makes it affordable.

After in depth experimentation, it has been concluded that separating the resource exhausting detector architecture from the lower input size classifier can be a conceivable strategy. A lightweight version of the YOLOv3 architecture is used for detection task with number of filters as small as possible. We have observed that even with this significantly thinner architecture, small drones can be detected with drastically low false alarm rate. However, this strategy would not always detect only intended objects; therefore, this part of the system should be treated as a primary filter for candidate targets.

In addition to this, we have developed and presented an autonomous intelligent tracking policy, where suspicious airborne targets are examined in detail with a lower-angle camera. Probably, the most innovative contribution of this paper is the proposal of a basic method, where frames coming from multiple cameras are overlaid with a proper configuration and object detection algorithm with deep learning is executed once. To the best of our knowledge, this is the first similar attempt in the literature.

Our multi-camera scheme can be accompanied by more complex re-identification (ReID) algorithms in future, which offers significant performance augmentation. Literature on ReID primarily have been focused on person/pedestrian tracking such as [42]. In future developments, we would like to apply a similar approach to track drones (and other airborne objects), starting from their initial detection on primary, static wide-angle camera, until the end of the recognition process with secondary zoomed camera.

To draw a conclusion, we can firmly state that using a very lightweight (in terms of filter count) deep YOLO architecture (properly and adequately trained with a voluminous dataset) shall give a high performance in terms of precision and accuracy compared to conventional object detection methods, while attaining a similar FPS and memory consumption. As mentioned previously, this architecture would not be a front-end recognition system, but serve as a primary candidate target location finder.

Acknowledgements

This work is supported by French Ministry of Defence, Directorate General of Armament (DGA).

Authors' contributions

EU and EZ have presented the ideas, carried out the experiments and written the paper. NR and PED have provided the required materials, helped with the experiments and production of paper. All authors read and approved the final manuscript.

Funding

This work is funded by French Ministry of Defence, Directorate General of Armament (DGA).

Availability of data and materials

The datasets generated during and/or analyzed during the current study are not publicly available due to confidentiality.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]ISAE-SUPAERO, 10, Avenue Edouard Belin, 31400 Toulouse, France. [2]ONERA, 2, Avenue Edouard Belin, 31000 Toulouse, France.

Received: 21 November 2018 Accepted: 31 May 2019
Published online: 24 July 2019

References

1. Valentak Z (2018) Drone market share analysis. http://www.dronesglobe.com/news/drone-market-share-analysis-predictions-2018/
2. Jansen B (2015) Drone crash at white house reveals security risks. USA Today
3. Jouan A (2014) Survols de centrales: un expert reconnu s' inquiète. http://www.lefigaro.fr/actualite-france/2014/11/25/01016-20141125ARTFIG00024-survols-de-centrales-un-expert-reconnu-s-inquiete.php
4. Serna J (2016) Lufthansa jet and drone nearly collide near lax. LA Times
5. Gallagher S (2013) German chancellor's drone 'attack' shows the threat of weaponized uavs. Ars Technica
6. Dinan S (2017) Drones become latest tool drug cartels use to smuggle drugs into u.s. https://www.washingtontimes.com/news/2017/aug/20/mexican-drug-cartels-using-drones-to-smuggle-heroi/
7. Zhang L, Young S (2018) China busts smugglers using drones to transport smartphones: state media. https://www.reuters.com/article/us-china-crime-smartphones-smugglers/china-busts-smugglers-using-drones-to-transport-smartphones-state-media-idUSKBN1H60BT
8. BBC (2018) Charges over drone drug smuggling into prisons. https://www.bbc.com/news/uk-england-43413134

9. Nguyen P, Ravindranatha M, Nguyen A, Han R, Vu T (2016) Investigating cost-effective rf-based detection of drones. In: Proceedings of the 2nd Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use. ACM. pp 17–22
10. Liu H, Wei Z, Chen Y, Pan J, Lin L, Ren Y (2017) Drone detection based on an audio-assisted camera array. In: Multimedia Big Data (BigMM), 2017 IEEE Third International Conference On. IEEE. pp 402–406
11. Hommes A, Shoykhetbrod A, Noetel D, Stanko S, Laurenzis M, Hengy S, Christnacher F (2016) Detection of acoustic, electro-optical and radar signatures of small unmanned aerial vehicles. In: Target and Background Signatures II, vol. 9997. International Society for Optics and Photonics. p 999701
12. Hammer M, Hebel M, Laurenzis M, Arens M (2018) Lidar-based detection and tracking of small uavs. In: Emerging Imaging and Sensing Technologies for Security and Defence III; and Unmanned Sensors, Systems, and Countermeasures, vol. 10799. International Society for Optics and Photonics. p 107990
13. Müller T (2017) Robust drone detection for day/night counter-uav with static vis and swir cameras. In: Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR VIII, vol. 10190. International Society for Optics and Photonics. p 1019018
14. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp 779–788
15. Mendis GJ, Randeny T, Wei J, Madanayake A (2016) Deep learning based doppler radar for micro uas detection and classification. In: Military Communications Conference, MILCOM 2016-2016 IEEE. IEEE. pp 924–929
16. Ganti SR, Kim Y (2016) Implementation of detection and tracking mechanism for small uas. In: Unmanned Aircraft Systems (ICUAS), 2016 International Conference On. IEEE. pp 1254–1260
17. Drozdowicz J, Wielgo M, Samczynski P, Kulpa K, Krzonkalla J, Mordzonek M, Bryl M, Jakielaszek Z (2016) 35 ghz fmcw drone detection system. In: Radar Symposium (IRS), 2016 17th International. IEEE. pp 1–4
18. Kwag Y-K, Woo I-S, Kwak H-Y, Jung Y-H (2016) Multi-mode sdr radar platform for small air-vehicle drone detection. In: Radar (RADAR), 2016 CIE International Conference On. IEEE. pp 1–4
19. Laurenzis M, Hengy S, Hommes A, Kloeppel F, Shoykhetbrod A, Geibig T, Johannes W, Naz P, Christnacher F (2017) Multi-sensor field trials for detection and tracking of multiple small unmanned aerial vehicles flying at low altitude. In: Signal Processing, Sensor/Information Fusion, and Target Recognition XXVI, vol. 10200. International Society for Optics and Photonics. p 102001
20. Kim BH, Khan D, Bohak C, Kim JK, Choi W, Lee HJ, Kim MY (2018) Ladar data generation fused with virtual targets and visualization for small drone detection system. In: Technologies for Optical Countermeasures XV, vol. 10797. International Society for Optics and Photonics. p 107970
21. Hauzenberger L, Holmberg Ohlsson E (2015) Drone detection using audio analysis
22. Nam SY, Joshi GP (2017) Unmanned aerial vehicle localization using distributed sensors. Int J Distrib Sensor Networks 13(9):1550147717732920
23. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436
24. Taigman Y, Yang M, Ranzato M, Wolf L (2014) Deepface: Closing the gap to human-level performance in face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp 1701–1708
25. Schumann A, Sommer L, Klatte J, Schuchert T, Beyerer J (2017) Deep cross-domain flying object classification for robust uav detection. In: Advanced Video and Signal Based Surveillance (AVSS), 2017 14th IEEE International Conference On. IEEE. pp 1–6
26. Aker C, Kalkan S (2017) Using deep networks for drone detection. arXiv preprint arXiv:1706.05726
27. Saqib M, Khan SD, Sharma N, Blumenstein M (2017) A study on detecting drones using deep convolutional neural networks. In: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). IEEE. pp 1–5
28. (2018) How Droneshield works?. https://www.droneshield.com/how-droneshield-works/. Accessed: 22 Oct 2018
29. (2018) Introduction to Dedrones Airspace Security Platform. https://www.dedrone.com/webinars/introduction-to-dedrones-airspace-security-platform-11-28-2018. Accessed: 22 Oct 2018
30. (2018) Gryphon Skylight System. Detect, track and classify moving objects in your airspace. https://www.srcinc.com/pdf/Radars-and-Sensors-Gryphon-Skylight.pdf. Accessed: 22 Oct 2018
31. Fan H, Zheng L, Yan C, Yang Y (2018) Unsupervised person re-identification: Clustering and fine-tuning. ACM Trans Multimed Comput Commun Appl (TOMM) 14(4):83
32. Zheng L, Shen L, Tian L, Wang S, Wang J, Tian Q (2015) Scalable person re-identification: A benchmark. In: Proceedings of the IEEE International Conference on Computer Vision. pp 1116–1124
33. Redmon J, Farhadi A (2018) Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767
34. Cheng B, Wei Y, Shi H, Feris R, Xiong J, Huang T (2018) Decoupled classification refinement: Hard false positive suppression for object detection. arXiv preprint arXiv:1810.04002
35. Hu W, Tan T, Wang L, Maybank S (2004) A survey on visual surveillance of object motion and behaviors. IEEE Trans Syst Man Cybern Part C (Appl Rev) 34(3):334–352
36. Wojke N, Bewley A, Paulus D (2017) Simple online and realtime tracking with a deep association metric. IEEE
37. Wang H, Kirubarajan T, Bar-Shalom Y (1999) Precision large scale air traffic surveillance using imm/assignment estimators. IEEE Trans Aerosp Electron Syst 35(1):255–266
38. Yilmaz A, Javed O, Shah M (2006) Object tracking: A survey. ACM Comput Surv (CSUR) 38(4):13
39. Dong X, Zheng L, Ma F, Yang Y, Meng D (2018) Few-example object detection with model communication. IEEE Trans Pattern Anal Mach Intell
40. Lienhart R, Maydt J (2002) An extended set of haar-like features for rapid object detection. In: Image Processing. 2002. Proceedings. 2002 International Conference On, vol. 1. IEEE
41. Zivkovic Z (2004) Improved adaptive gaussian mixture model for background subtraction. In: Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference On, vol. 2. IEEE. pp 28–31
42. Wu Y, Lin Y, Dong X, Yan Y, Bian W, Yang Y (2019) Progressive learning for person re-identification with one example. IEEE Trans Image Process
43. Coluccia A, Ghenescu M, Piatrik T, De Cubber G, Schumann A, Sommer L, Klatte J, Schuchert T, Beyerer J, Farhadi M, et al (2017) Drone-vs-bird detection challenge at ieee avss2017. In: Advanced Video and Signal Based Surveillance (AVSS), 2017 14th IEEE International Conference On. IEEE. pp 1–6

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.