An Improved YOLOv5 for Real-time Mini-UAV Detection in No Fly Zones

Tijeni Delleji [1,2][a] and Zied Chtourou[1][b]

[1] *Science and Technology for Defense Lab (STD), Military Research Center, Taeib Mhiri City, Aouina, 2045, Tunis, Tunisia*

[2] *Digital Research Center of Sfax, 3021 Sfax, Tunisia*

Keywords: Mini-UAV, YOLOv5, Dahua Multi-sensor Camera, Object Detection, Tiny and Small Objects, Air Image, Real-time, No Fly Zones.

Abstract: In the past few years, the manufacturing technology of mini-UAVs has undergone major developments. Therefore, the early warning optical drone detection, as an important part of intelligent surveillance, is becoming a global research hotspot. In this article, the authors provide a prospective study to prevent any potential hazards that mini-UAVs may cause, especially those that can carry payloads. Subsequently, we regarded the problem of detecting and locating mini-UAVs in different environments as the problem of detecting tiny and very small objects from an air image. However, the accuracy and speed of existing detection algorithms do not meet the requirements of real-time detection. For solving this problem, we developed a mini-UAV detection model based on YOLOv5. The main contributions of this research are as follows: (1) a mini-UAV dataset of air pictures was prepared using Dahua multi-sensor camera; (2) a tiny and very small object detection layers are added to improve the model's ability to detect mini-UAVs. The experimental results show that the overall performance of the improved YOLOv5 is better than the original. Therefore, the proposed mini-UAV detection technology can be deployed in monitor center in order to protect a No Fly Zone or a restricted area.

1 INTRODUCTION

The International Civil Aviation Organization (ICAO) denotes by "drone" any Unmanned Aerial Vehicle (UAV). Furthermore, the Air Force Special Operations Command (AFSOC) gave additional three names for a drone: a flying robotic system, an Unmanned Aircraft System (UAS), and a micro air vehicle (MAV)(United States Air Force, 2009),(Doyle, 2013). So, to simplify, an UAV is an aircraft either controlled by pilot via RF remote controlller or increasingly, autonomously following a mission planner through flight controller. In the same context, Maddalon et al.(Maddalon et al., 2013) and Lykou et al.(Lykou et al., 2020)have mentioned that for the NATO (North Atlantic Treaty Organization) classification, UAVs weighting between 2 and 25 kg are called mini-UAVs. So, a mini-UAV can carry an operating payload up to 15 kg, e.g. the DJI MATRICE 600 which weighs 10kg is capable of carrying a 6 kg payload for 16 minutes (DJI, 2021). Over the last few years, the manufacturing technology of mini-Unmanned aerial vehicles (mini-UAVs), also known mini drones, has been experiencing a significant evolution. There are multiple usages for mini-UAVs, including: precision agriculture for Spraying Operation, professional aerial photography and industrial applications (Seidaliyeva et al., 2020). However, the polyvalence of this type of flying gadgets made it accessible to everyone, particularly to terrorist groups. Therefore, we can conclude that the detection of mini-UAVs before serious attacks in restricted areas, especially for "No Fly Zones"(NFZ), is of the utmost interest. Pointing out that NFZs , i.e. territories over which no aircraft are allowed to fly, include the world's major airports, the borders between two sovereign countries or regions, major cities/regions, etc.

Consequently, in this work we will treat the issue of detecting and localizing mini-UAV in diverse environments as a problem of very small object detection from an aerial perspective image. To set the record straight, an air image or ground-to-aerial perspective image is mostly a picture of a flying object that must include sky background, taken by a ground-

[a] https://orcid.org/0000-0003-1323-8520

[b] https://orcid.org/0000-0001-7154-6906

Delleji, T. and Chtourou, Z.
An Improved YOLOv5 for Real-time Mini-UAV Detection in No Fly Zones.
DOI: 10.5220/0011065400003209
In *Proceedings of the 2nd International Conference on Image Processing and Vision Engineering (IMPROVE 2022)*, pages 174-181
ISBN: 978-989-758-563-0; ISSN: 2795-4943

based imaging system, typically used to monitor a No Fly Zone or a restricted area. The real-time object detection applied to UAV monitoring is really crucial. Nevertheless, these applications need early detection of objects so that they can be used later as inputs for other activities. Due to early detection, the appearance of the objects is generally very small. In general, the aim of tiny object detection is to detect objects that belong to the image and are tiny in size, which implies that the objects of interest are objects that either have a large physical appearance but occupy only a tiny area in an image, or have a really tiny appearance. Improvements in object detection algorithms allow faster and more accurate results.

The most recent methods using deep Convolutional Neural Networks (Deep CNN) usually involve several steps. First, specify the objects of interest in the image, then pass them through the Deep CNN for feature extraction and then classify them using supervised classification techniques. Finally, mixing the results between the objects to properly mark the bounding box. In Deep CNN models there are mainly two categories of current state-of-art object detectors: single-stage and two-stage detectors. On one hand, the single stage detectors, are represented by SSD (Single Shot multibox Detector)(Liu et al., 2016) that runs a convolutional network on input image only once, calculates a feature map and predicts a detection; and YOLO (You Only Look Once)(Redmon et al., 2016), that treats object detection as a simple regression problem by tacking an input image and learning the class probabilities and bounding box coordinates. Such models (SSD and YOLO) are proposed by considering both accuracy and processing time. On the other hand, the two-stage detectors, include the Faster R-CNN (Region-based convolutional Neural Networks) (Ren et al., 2015) that uses a region proposal networks to generate regions of interests in the first stage; and Mask R-CNN (He et al., 2017) that sends the region proposals down the pipeline for object classification and bunding box regression. Such models perform well in term of accuracy, in particular the faster R-CNN with an accuracy of 73% mAP. But due to their very complex pipeline, these two-stage detectors perform poorly in terms of speed with 7 frames per second (FPS), which restricts them for real-time object detection.

Since real-time is a challenge in optical early warning UAV detection, in our work, we will propose a CNN architecture based on a detection method with fast processing speed. Especially YOLO performs well compared to previous region-based algorithms in terms of speed with 45 FPS while maintaining a good detection accuracy more than 63% mAP (Rahim et al., 2021). Although the speed and accuracy were good, YOLOv1 (YOLO first version) (Redmon et al., 2016) made some remarkable localization errors. In other words, the bounding boxes predicted by YOLOv1 are not accurate. So, to overcome the deficiencies of YOLOv1, the creators of YOLO launched YOLOv2 (YOLO second version) (Redmon and Farhadi, 2017) where the similarity of predicted bounding box to the ground truth bounding box, and the percentage of total relevant objects correctly classified, were mainly focused without impairing the accuracy of the classification. Moreover, YOLOv2, which called also YOLO9000 (Redmon and Farhadi, 2017), gained a speed of 59 FPS and mAP of 77.8% in experiments on the PASCAL VOC 2007 dataset(Everingham et al., 2014), (Everingham et al., 2010). The YOLOv3 (the third version of YOLO) (Redmon and Farhadi, 2018), whose main improvement is the addition of multi-scale prediction, has brought further improvements in speed and accuracy. In experimenting with MS COCO (Lin and Maire, 2014), (Kim, 2017)dataset, YOLOv3 obtained 55% AP score and achieved a real-time speed of approximately 200 FPS on Tesla V100. YOLOv4 (YOLO fourth version) was released on 23 April 2020 and YOLOv5 on 10 June 2020. However, YOLOv4 (Bochkovskiy et al., 2020), (Wang et al., 2021d) was released in the Darknet framework, while YOLOv5 (Wang et al., 2021d) ,(Ultralytics, 2021), (Ahmed and Kharel, 2021), (Wang et al., 2021b), (Yan et al., 2021), (Yang et al., 2020) has been released in the Ultralytics PyTorch framework. Despite the fact that YOLOv4 can reach 43.5% AP on MS COCO (COCO, 2021)and 65 FPS speed, the developers of YOLOv5 claim that in a YOLOv5 Collab notebook, running a Tesla P100, they found inference times of up to 0.007 seconds per image, meaning 140 frames per second (FPS) (Yan et al., 2021). In contrast, YOLOv4 achieved 50 FPS after having been converted to the same Ultralytics PyTorch library (Ultralytics, 2021). Not only that, they also mentioned that YOLOv5 is smaller. Specifically, the YOLOv5 file weights 27 megabytes. However, the weights file for YOLOv4 (with Darknet architecture) is 244 megabytes. So, YOLOv5 is about 88% smaller than YOLOv4 (Roboflow, 2021). The development of new versions of YOLO is not finished. In Oct 28, 2021 Yuxin et al. (Fang et al., 2021) have launched the YOLOS (You Only Look at One Sequence) . It is a series of object detection models based on the vanilla Vision Transformer with the fewest possible modifications, region priors, as well as inductive biases of the target task. However, despite that other variants of YOLO are developed such as YOLOX (Ge et al., 2021), YOLOv5 remains more practical in real time tasks

All in all, YOLOv5, with its latest v6.0 version released in January 2022, claims to be fast, has a very light model size, trains quickly, makes inferences quickly, and is comparable to YOLOv4 in accuracy. (Adibhatla et al., 2021).

This paper focuses on detect mini-drones based on ground to aerial perspective images, more precisely the AI techniques used for early detection and localization. The goal is to obtain a real-time and accurate deep-CNN object detector which will be able to correctly detect and locate mini-drones supporting a payload, in order to start a neutralization system. The main contributions of this work can be summarized as follows:

(1) We collect images of mini-UAVs in a real environment, most of which contain flying mini-UAVs in poor visibility conditions. Subsequently, we build a custom dataset, called "mini-UAV dataset", which provides a benchmark to evaluate the performance of the proposed detection model.

(2) We develop a mini-UAV detection model by redesigning the YOLOv5 object detector(Ultralytics, 2021), (Wang et al., 2021b). Moreover, we implement key modifications to the network to improve the behavior of the model in terms of performance. So, the redesigned model uses features learned by a deep CNN to focus on very small flying object detection in aerial perspective.

The remainder of this paper is partitioned as follows. We present the issues of object detection and the neural architecture of the YOLO model in Section 2. A mini-UAV targets real-time detection algorithm based on improved YOLOv5 is presented in Section 3, and the results are discussed in Section 4. Finally, Section 5 concludes the study.

2 RELATED WORKS

2.1 Issues in Deep Object Detection

Deep Object detection is a Deep Learning powered computer vision technique that consists of identifying and locating instances of an object of a certain class within an image or a video. The deep learning-based object detectors (i.e,. Deep detectors) usually have two parts: one is a skeleton or encoder that takes an air image as input and passes it through a sequence of blocks and layers that learn to extract statistical features used to locate and annotate flying objects. And the other called a head or a decoder, it is the main part used to predict bounding boxes and labels of objects. In addition, object detectors developed in recent years usually have some layers inserted between the skeleton and the head, and usually used to collect feature maps at different stages. We can call it the neck of the object detector (Bochkovskiy et al., 2020), (Yan et al., 2021). So, the detector needs to meet the following conditions:

i.Higher input network scale (resolution)-used to detect multiple very small objects;

ii.Higher layers – higher receptive fields to cover the expanding scale of the input network;

iii.More parameters improve the model's ability to detect multiple objects of different sizes in a single image.

In summary, the general object detector consists of the parts presented by Figure 1.

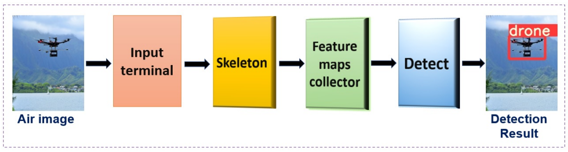


Figure 1: Concept of architectural object detection for ground to aerial perspective image.

Although these detectors remain benchmarks, research in this area is far from complete and many difficulties remain. An interesting summary of some of the challenges is presented in the review by Agarwal et al.(Agarwal et al., 2019).

• Occlusions: This problem, which exists in most applications, is an issue since some of the information is hidden. Thus, providing examples containing occlusions in the training dataset may partially solve the problem but will not represent all forms of occlusion.

• Very small or tiny objects: detecting very small objects is more difficult than detecting medium or large ones. This is due to many factors such as lack of associated information, inaccurate localization and confusion of objects with the background image. So, to overcome this problem, solutions vary in terms of complexity from simple scaling to the use of surface networks, coarse and fine networks to a super-resolution method that could be implemented with a variant of GAN learning (Wang et al., 2021c) to represent very small objects with higher resolutions. In addition, low image resolution could cause the same problems and thus require a super-resolution method.

2.2 Visualization of YOLO Network Architecture

The YOLO is a technique based on regression. Instead of selecting the relevant part of an image, it predicts classes and bounding boxes for the entire image in a single run of the algorithm. So, the idea of YOLO originated from the extension of the basic CNN (Convolutional Neural Network) idea from classification tasks to detection. The YOLO series (from YOLOv0 to YOLOv5) is a regression method based on deep learning. So, the last version: YOLOv5(Wang et al., 2021d) ,(Ultralytics, 2021), (Ahmed and Kharel, 2021), (Wang et al., 2021b), (Yan et al., 2021), is basically modified on the structure of YOLOv3(Redmon and Farhadi, 2018). As shown in Figure 2, the YOLO series architecture is divided into three functionally different parts, called backbone network, neck network and head or detect network (Bochkovskiy et al., 2020), (Yan et al., 2021). This is a division found in the architecture of many recent image detection models (Yao et al., 2021):

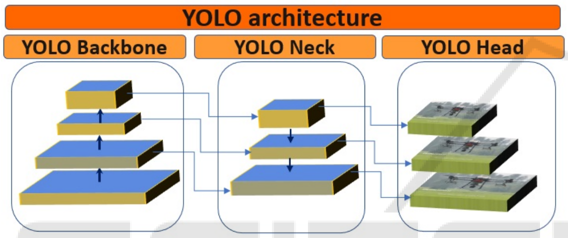


Figure 2: Basic architecture of the YOLO series network presented as Backbone, Neck and head.

The backbone is the ”body” of the network, which will enable all the decisions made by the network. In simple terms, it can be seen as a ”converter” that converts the input image, a data format as such difficult to process by AI (Artificial intelligence), into a set of information that characterizes its content called ”features” (such as the presence of shapes, colors, textures, ...) from which it is easy to recognize objects. It is thus composed of a series of successive layers, and extracts feature maps, i.e. maps indicating which features are present at which locations in the image. The backbone is usually trained separately on image classification competitions such as the ImageNet challenge (ImageNet, 2021), which include hundreds of thousands of images with a wide range of content such as animals, vehicles, plants, etc. This diversity of content forces the backbone to learn a wide variety of features in terms of size, color and shape of the elements it observes and thus be more robust and able to extract useful features regardless of the image presented to the backbone. The second part of Yolov5, the neck, has the role of extracting the relevant features from all the layers of the backbone, and combining them into useful features for our detection task. Indeed, not all the layers included in the backbone learn the same information: the first set of layers, generally of higher spatial resolution, will detect features that are often simpler (the presence of lines, colors) and smaller. The last set are the lower resolution layers that tend to provide more complex features (e.g. the combination of specific shapes and colors such as a metal circle with a hole for a car rim) and large objects. The neck makes it possible to integrate and combine features of different resolutions and complexities, to allow detection of small and large, simple and complex features. Finally, the head is responsible for the final decision of the network. Based on the information provided by the neck, it will detect the elements of interest by drawing bounding boxes around them and it will, furthermore, give the nature of every object present in each bounding box. In terms of general architecture, Yolov5 is similar to its predecessors Yolo and other models in the literature. It is therefore time to see the real reason for the difference in performance. The ”bag of freebies” is a set of enhancements with no impact on the architecture of a network, which can be used ”free of charge”, with no cost of modification on an existing network. It thus gathers all the improvements that apply during the network learning such as: the loss function, data augmentation, cross mini-batch normalization. The ”bag of specials” is, on the contrary, a bag containing improvements that requiring specific modifications to the architecture of a network. It contains recent advances in the scientific literature that improve the performance of the network without decrease its speed.

3 THE IMPROVED YOLOv5 ALGORITHM

In order to implement an optical early warning mini-UAV detection system, a flying target, which necessarily has a small or even tiny appearance, must be detected as much as possible. Thus, the size distant mini-UAVs, in the sky background, is very small; and the receptive field size of YOLOv5 is not enough to detect these tiny flying objects. Hence the reason to improve the architecture of YOLOv5. As shown in Figure 3, there are two improvements to the original YOLOv5 architecture: i) a fourth scale (marked with yellow rectangle in Figure 3) is added to the three scales of YOLOv5 feature maps to capture more texture and contour information of tiny mini-UAVs. ii) feature maps from the backbone network are brought into the added fourth scale (represented by the red line) to reduce feature information loss of tiny mini-UAVs.

The YOLOv5 final part consists of three detec-

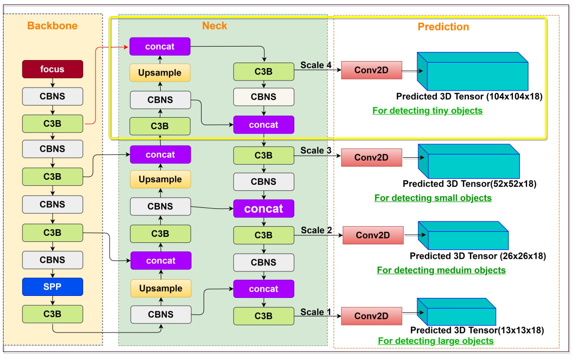

Backbone
Neck
Prediction
focus
CBNS
C3B
CBNS
C3B
CBNS
C3B
CBNS
SPP
C3B
concat
Upsample
CBNS
C3B
concat
Upsample
CBNS
C3B
concat
Upsample
CBNS
C3B
CBNS
concat
C3B
CBNS
concat
C3B
CBNS
concat
C3B
Scale 4
Scale 3
Scale 2
Scale 1
Conv2D
Conv2D
Conv2D
Conv2D
Predicted 3D Tensor (104x104x18)
For detecting tiny objects
Predicted 3D Tensor(52x52x18)
For detecting small objects
Predicted 3D Tensor (26x26x18)
For detecting meduim objects
Predicted 3D Tensor(13x13x18)
For detecting large objects

Figure 3: Anatomy of the improved YOLOv5 for mini-UAV detection.

tion tensors. So, YOLOv5 applies 8,16, and 32 downsampling of the initial image to detect objects at different resolutions. For example, the final outputs of YOLOV5 are three tensors of predictions 52x52x18,26x26x18 and 13x13x18 for an initial image of resolution 416*416. In fact, the problem of lacking appearance information is related to different image resolutions. For example, if the image resolution is low, it may prevent the detector from detecting very small objects. In these cases, the information needed to detect very small objects will be very limited. Indeed, in YOLOv5, if the object of interest occupies a size of 8*8 pixels on an image with a resolution of 416*416, then it will be represented by only one pixel in the final feature maps. Therefore, any object smaller than 8*8 will be disappeared. Subsequently, this architecture of YOLOv5 is insufficient for the detection of tiny objects. Therefore, the main idea of our proposal is to add a detection level (scale 4 in Figure 3) with a high resolution that is able to extract more features for tiny objects. For this purpose, we added a level that reduces the resolution only four times. In fact, our proposed architecture aims to detect tiny objects, which is why we have added a higher resolution detection level (104*104). The addition of the later consists of adding seven layers as indicated in Figure 3 by yellow boxe, of which the upsample layer increases the resolution and then the output of this layer will be concatenated with the output of layer three of the Backbone part. In addition, the connection represented by the red line is added to bring the feature information from the backbone network into the added fourth scale of the neck network. Based on the idea of residual networks, this connection can improve gradient backpropagation, to prevent the gradient from being erased, and reduce the loss of the feature information of very small flying objects.

4 EXPERIMENTAL RESULTS AND EVALUATION

4.1 Custom Dataset Construction

Our custom dataset, called “mini-UAV dataset”, was collected and constructed by us “the anti-drone project team” for the HANNIBAL Air defense system. This dataset is captured by a Dahua multisensor Network PTZ camera (Dahua, 2021), in various complex scenarios. We record various videos of several UAV types flying in the air. In order to ensure the diversity of data, UAVs, mainly including rotor mini-UAV, like four-rotor UAV (i.e., DJI-Phantom4, DJI-Marvic), and six-rotor UAV (i.e., DJI-Matrice 600, DJI Agars T16) (DJI, 2021). The videos recorded include some useful attributes, e.g., Illumination Variation (IV), Weather Conditions (WC), and Diverse Background (DB). In addition, the captured videos are stored in an MP4 files with a frame rate of 25 FPS. The frames, which have a resolution of 1920*1080, are manually annotated with bounding boxes. Thus, a total of 4560 sample images are used in this experiment which are divided, randomly, into 3400 images for training and 1160 images for testing purposes.

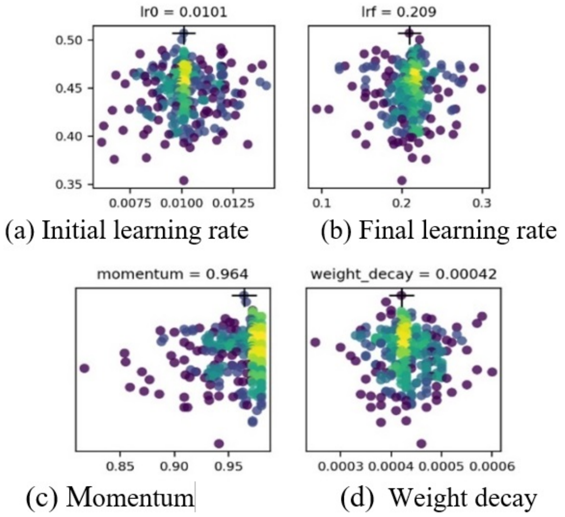
4.2 Experimental Setting

Experiments in this paper have been performed using the machine learning framework PyTorch 1.9. At the beginning of our work, training tests were performed, with 100 epochs, on the kaggle platform with a GPU NVIDIA TESLA P100, 16 GB of memory, Driver version: 450.119.04, and CUDA version: 11.0. Then, hyperparameters evolution is performed on a workstation with AMD Ryzen 9 5900X 12-Core Processor 3.70 GHz, NVIDIA Geforce RTX 3070 AORUS MASTER (8GB memory) GPU, CUDA 11.1.0, cuDNN v8.2.2 and 64GB of memory. In our work, the base scenario, of optimizing the hypermeters that are shown in Figure 4, is trained during 100 of GPU hours. Afterwards, a final training operation is performed using the hyperparameters generated by the optimization algorithm (Wicaksono and Supianto, 2018), with an input image size of 640*640, and a batch size on a GPU of 16 images.

Figure 4 shows optimization of some hyperparameters of YOLOv5, which has in total 30 of them.

4.3 Experimental Analysis

Table 1 shows mAP, precision, and recall of two models. It can be seen that, after 300 epochs, our method has better performance. Compared with the results of



(a) Initial learning rate

(b) Final learning rate

(c) Momentum

(d) Weight decay

Figure 4: Hyperparameter optimization of the improved YOLOv5.

the baseline, the precision of the improved YOLOv5 model is increased by 6.84% and the recall rate is increased by 9.04%. Moreover, the mAP_0.5:0.95 is increased by 40.57% and the mAP_0.5 has improved by 9.9%. These results confirm what was mentioned at the beginning of this interpretation, that the accuracy of our model is higher than that of the baseline.

Table 1: The model evaluation indicators for both the improved-YOLOv5 and the Baseline models.

	Performance metrics			
detection model	mAP_0.5	mAP _0.5:0.9	Precision	Recall
Improved YOLOv5	0.8606	0.9836	0.9804	0.9693
Beseline	0.4549	0.8846	0.912	0.8789

The loss function indicates the performance of a given predictor in detecting the input data points in a dataset. The smaller the loss, the better the detector is at modeling the relationship between the input data and the output targets. To evaluate our work, we have used two different types of loss: the confidence loss or objectness loss ($Loss^{obj}$) and the predicted bounding box loss($Loss^{box}$). In other words, the box loss represents how well the model can locate the center of an object and how well the predicted bounding box covers an object. While, the objectness loss determine whether there are objects in the predicted bounding box. Let's mention that classification loss ($Loss^{class}$) is not used for the evaluation, since our custom "mini-UAV dataset" is composed of a single class called "mini-UAV".Table 2 shows that after 300 epochs of training, our model has the lowest total loss value, which makes it perform better than the baseline model.

Table 2: The loss functions for both the imroved-YOLOv5 and the Baseline.

	Loss function		
detection model	$Loss^{box}$	$Loss^{obj}$	$Loss^{total}$
Improved YOLOv5	0.01257	0.001986	0.014556
Baseline	0.02621	0.0007704	0.0269804

To highlight the performance of the improved YOLOv5 detector, we compare it to the baseline. The results of the test are based on 400 frames from YouTube video sequences captured in outdoor environment with different drone models, and from visible video clips shot with our Dahua multi-sensor camera. So, an illustration of detected results of baseline model and the improved YOLOv5 for some samples in air images (i.e. ground to aerial perspective images) is shown in Figure 5.f where the red, and green bounding boxes correspond to detections by the improved YOLOv5 detector, and the baseline detector, respectively.

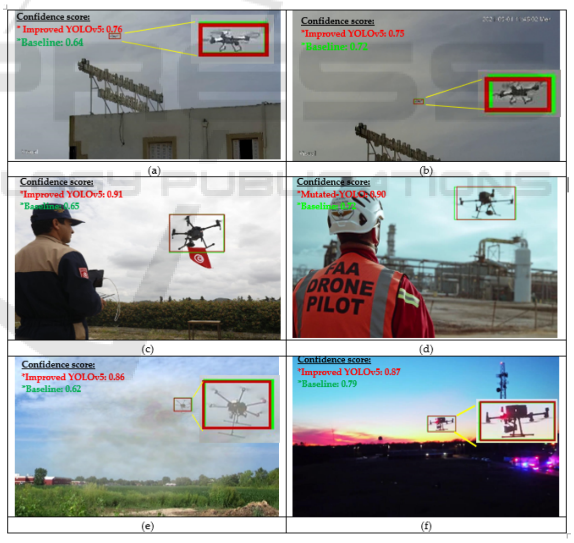


Figure 5: Comparison between the original YOLOv5 (the baseline) and the improved-YOLOv5 in aerial perspective at diverse distances and with different visibility conditions.

For instance, in Figures 5.a and 5.b, we used two frames of size 1920*1080 taken by our Dahua camera, which contain each other a very far mini-drones. Indeed, the improved YOLOv5 has detected the far mini-drones with a confidence score higher than 0.76, which is superior than that of the baseline (i.e., between 0.64 and 0.72). Accordingly, Figures 5.a and 5.b show that our model was efficient and outperformed the baseline in the detection of mini-UAV of tiny and small appearance. Furthermore, the results in Figures 5.c and 5.d shows that the bounding boxes of the improved YOLOv5 (red bounding boxes) are more adjusted with the detected mini-UAVs. This was consistent with the previous evaluation, and this is shows that our method has the lowest box loss. Finally, the last Figures (lack of lighting for Figure 5.e and fog phenomena for Figure 5.f) show that our model performs well even under low visibility conditions.

5 CONCLUSIONS

In this research, Deep learning technology was applied to tiny and very small flying object detection in aerial perspective image (i.e., an image of a flying object on a sky background). And based on YOLOv5 object detector(Ultralytics, 2021), a high-precision mini-UAV detection model was proposed. So, we firstly collected images of mini-UAVs in a real environment, using our Dahua Thermal Network PTZ Camera (Dahua, 2021). Most of them consist of mini-UAVs flying in poor visibility conditions. Then, we constructed a custom dataset designed by "mini-UAV dataset", which provides a benchmark to evaluate the performance of the proposed detection model, especially under low-visibility condition. As a result, in order to reduce the total loss, we implemented a mini-UAV detection model based on YOLOv5, which has recently appeared. The proposed detector uses features learned by a deep CNN to focus on very small flying object detection in air image. This paper mainly researches and develops drone related threats under the requirement of real-time flying object detector. However, fast detection still needs specific hardware configuration. In the future, we will continue to optimize YOLOv5 especially by inputting a Small Target Motion Detection-bsed model (STMD) (Wang et al., 2021a) for early warning. At the same time, we will attempt to deploy and integrate our model with a flying object tracker such as DeepSORT (Wojke et al., 2017), with the goal of establishing a monitoring system in a No Fly Zone.

ACKNOWLEDGEMENTS

This work is financially supported by the Tunisian Ministry of National Defense, Science and Technology for Defense lab (STD), and Military Research Center through an R&D project entitled: Development of a Tactical System for Detection and Neutralization of Mini-UAVs.

REFERENCES

Adibhatla, V., Chih, H.-C., Hsu, C.-C., Cheng, J., Abbod, M., and Shieh, J.-S. (2021). Applying deep learning to defect detection in printed circuit boards via a newest model of you-only-look-once. *Mathematical Biosciences and Engineering*, 18:4411–4428.

Agarwal, S., Terrail, J. O. D., and Jurie, F. (2019). Recent advances in object detection in the age of deep convolutional neural networks.

Ahmed, K. R. and Kharel, S. (2021). Potholes detection using deep learning and area estimation using image processing.

Bochkovskiy, A., Wang, C., and Liao, H. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *CoRR*, abs/2004.10934.

COCO (2021). COCO, Common Object in Context. Availbleonline:https://cocodataset.org/#home. (accessed on 17 September 2021).

Dahua (2021). Dahua Technology. Availableonline:https://www.dahuasecurity.com/products/All-Products/Thermal-Cameras/Wizmind-Series/TPC-8-Series/TPC-PT8621C. (accessed on 04 August 2021).

DJI (2021). MATRICE 600PRO SIMPLY PROFESSIONAL PERFORMANCE. Availableonline:https://www.dji.com/matrice600-pro/info. (accessed on 23 Mars 2021).

Doyle, D. (2013). *Real-Time, Multiple, Pan/Tilt/Zoom, Computer Vision Tracking, and 3D Position Estimating System for Small Unmanned Aircraft System Metrology*. AIR UNIVERSITY, Wright-Patterson Air Force Base, Ohio, USA.

Everingham, M., Eslami, S. M. A., Gool, L. V., Williams, C. K. I., Winn, J. M., and Zisserman, A. (2014). The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111:98–136.

Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J. M., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.*, 88(2):303–338.

Fang, Y., Liao, B., Wang, X., Fang, J., Qi, J., Wu, R., Niu, J., and Liu, W. (2021). You only look at one sequence: Rethinking transformer in vision through object detection.

Ge, Z., Liu, S., Wang, F., Li, Z., and Sun, J. (2021). YOLOX: exceeding YOLO series in 2021. *CoRR*, abs/2107.08430.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988.

ImageNet (2021). Avaiableonline:https://www.image-net.org/challenges/LSVRC/. (accessed on 15 June 2021).

Kim, D.-H. (2017). Evaluation of coco validation 2017 dataset with yolov3. *Journal of Multidisciplinary Engineering Science and Technology (JMEST).*, 6:10356–10360.

Lin, T.-Y. and Maire, M. (2014). Microsoft coco: Common objects in context. cite arxiv:1405.0312Comment.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., and Reed, S. (2016). Ssd: Single shot multibox detector. In *Computer Vision – ECCV 2016*, pages 21–37, Cham. Springer International Publishing.

Lykou, G., Moustakas, D., and Gritzalis, D. (2020). Defending airports from uas: A survey on cyber-attacks and counter-drone sensing technologies. *Sensors*, 20(12).

Maddalon, J., Hayhurst, K., Koppen, D., Upchurch, J., Morris, A., and Verstynen, H. (2013). Perspectives on unmanned aircraft classification for civil airworthiness standards nasa sti program. .. in profile.

Rahim, A., Maqbool, A., and Rana, T. (2021). Monitoring social distancing under various low light conditions with deep learning and a single motionless time of flight camera. *PLOS ONE*, 16(2):1–19.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788.

Redmon, J. and Farhadi, A. (2017). Yolo9000: Better, faster, stronger. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6517–6525.

Redmon, J. and Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *arXiv.org*, pages 1–6.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Roboflow (2021). Availableonline:https://models.roboflow.com/object-detection/yolov5. (accessed on 03 December 2021).

Seidaliyeva, U., Akhmetov, D., Ilipbayeva, L., and Matson, E. T. (2020). Real-time and accurate drone detection in a video with a static background. *Sensors*, 20(14).

Ultralytics (2021). Ultralytics YOLOv5 and Vision AI. Availableonline:http://www.ultralytics.com. (accessed on 20 December 2021).

United States Air Force, W. (2009). Unmanned aircraft systems flight plan 2009-2047, technical report.

Wang, H., Zhao, J., Wang, H., Peng, J., and Yue, S. (2021a). Attention and prediction guided motion detection for low-contrast small moving targets.

Wang, X., Wei, J., Liu, Y., Li, J., Zhang, Z., Chen, J., and Jiang, B. (2021b). Research on morphological detection of fr i and fr ii radio galaxies based on improved yolov5. *Universe*, 7(7).

Wang, X., Xie, L., Dong, C., and Shan, Y. (2021c). Real-esrgan: Training real-world blind super-resolution with pure synthetic data.

Wang, Z., Wu, Y., Yang, L., Thirunavukarasu, Arjun an-Wand Evison, C., and Zhao, Y. (2021d). Fast personal protective equipment detection for real construction sites using deep learning approaches. *Sensors*, 21(10).

Wicaksono, A. S. and Supianto, A. A. (2018). Hyper parameter optimization using genetic algorithm on machine learning methods for online news popularity prediction. *International Journal of Advanced Computer Science and Applications*, 9(12).

Wojke, N., Bewley, A., and Paulus, D. (2017). Simple online and realtime tracking with a deep association metric. *CoRR*, abs/1703.07402.

Yan, B., Fan, P., Lei, X., Liu, Z., and Yang, F. (2021). A real-time apple target21s detection method for picking robot based on improved yolov5. *Remote Sensing*, 13(9).

Yang, G., Feng, W., Jin, J., Lei, Q., Li, X., Gui, G., and Wang, W. (2020). Face mask recognition system with yolov5 based on image recognition. In *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, pages 1398–1404.

Yao, J., Qi, J., Zhang, J., Shao, H., Yang, J., and Li, X. (2021). A real-time detection algorithm for kiwifruit defects based on yolov5. *Electronics*, 10(14).