A COMBINATION OF FASTER R-CNN AND YOLOv2 FOR DRONE DETECTION IN IMAGES

Pham Van Viet
Le Quy Don Technical University

ARTICLE INFO

Received: 09/5/2021
Revised: 28/5/2021
Published: 31/5/2021

KEYWORDS

Machine learning
Computer vision
Convolutional Neural Network
Faster R-CNN and YOLO
Drone detection

ABSTRACT

Today, drones are widely used for different purposes since they are not too expensive. Drones employed as explosive material, camera and illegal thing carriers can cause security threats. Computer vision can be applied to detect illegally acting drones effectively in a variety of conditions. A computer-based system using modern cameras is possible to discover small distant drones. The system can also become aware of low-speed and non-ground controlled drones. Furthermore, the system can display true drones. This makes the system friendly to users. This paper proposes a hybrid approach combining two emerging convolutional neural networks: Faster R-CNN and YOLOv2 to detect drones in images. Experimental results show that the approach can add up to almost 5% and more than 11% to precision and recall for Faster R-CNN and add up to 3% and more than 6% to these two metrics for YOLOv2. This better detection is resulted from the combination of the two networks. If a network is failed to detect drones in an image, the other network can help.

KẾT HỢP FASTER R-CNN VÀ YOLOv2 CHO VIỆC PHÁT HIỆN MÁY BAY KHÔNG NGƯỜI LÁI TRONG ẢNH

Phạm Văn Việt
Trường Đại học Kỹ thuật Lê Quý Đôn

THÔNG TIN BÀI BÁO

Ngày nhận bài: 09/5/2021
Ngày hoàn thiện: 28/5/2021
Ngày đăng: 31/5/2021

TỪ KHÓA

Học máy
Thị giác máy tính
Mạng nơ-ron tích chập
Faster R-CNN và YOLO
Phát hiện máy bay không người lái

TÓM TẮT

Ngày nay, máy bay không người lái (drone) được sử dụng rộng rãi cho các mục đích khác nhau vì chúng không quá đắt. Các drone được sử dụng làm các phương tiện mang vật liệu nổ, máy ảnh và vật bất hợp pháp có thể gây ra các mối đe dọa an ninh. Thị giác máy tính có thể được áp dụng để phát hiện các drone hoạt động bất hợp pháp một cách hiệu quả trong nhiều điều kiện khác nhau. Một hệ thống dựa trên máy tính sử dụng các camera hiện đại có thể phát hiện ra các drone nhỏ ở xa. Hệ thống cũng có thể nhận biết được các drone tốc độ thấp và không được điều khiển từ mặt đất. Hơn nữa, hệ thống có thể hiển thị các drone thực sự. Điều này giúp hệ thống thân thiện với người dùng. Bài báo này đề xuất một cách tiếp cận lai kết hợp hai mạng nơ-ron tích chập mới nổi: Faster R-CNN và YOLOv2 để phát hiện các drone trong ảnh. Kết quả thử nghiệm cho thấy rằng phương pháp này có thể thêm tới gần 5% và hơn 11% cho độ chính xác và độ tái hiện cho Faster R-CNN và thêm tới 3% và hơn 6% cho hai chỉ số này cho YOLOv2. Việc phát hiện tốt hơn này là kết quả của sự kết hợp của hai mạng. Nếu một mạng không thể phát hiện các drone trong một bức ảnh, mạng khác có thể trợ giúp.

DOI: https://doi.org/10.34238/tnu-jst.4465

Email: v.v.pham2012@gmail.com

1. Introduction

Today, drones are widely used for different purposes since they are not too expensive. Drones employed as explosive material, camera and illegal thing carriers can cause security threats. Discovering illegally acting drones can help to alert, prevent and track their operation.

Different types of sensors such as RADAR, LIDAR, acoustic and RF (Radio Frequency) sensors can be used to detect drones as reviewed in [1], [2]. However, small and low-speed drones challenge RADAR. LIDAR is problematic with large data output and cloud sensitivity. A long operational range and noisy environment makes an acoustic sensor less effective. An RF sensor cannot work with non-ground controlled drones that are automatically navigated, based on a predefined route.

Computer vision can be applied to detect illegally acting drones in a variety of conditions. A computer based system using modern cameras is possible to discover small distant drones. The system can also become aware of low-speed and non-ground controlled drones. Furthermore, the system can display true drones. This makes the system friendly to users. For these advantages, cameras are popularly integrated in modern drone detecting systems such as ND-BU001 [3] and DroneSentry [4].

Drone detection using computer vision is to determine the existence of a drone and its location in an input image. A drone is located by its bounding box. The study in [2] provides a review of methods to solve this problem. Some researches [5], [6] first represent a drone by feature vectors that are extracted from a set of training images by feature descriptors such as SIFT (Scale Invariant Feature Transform), SURF (Speeded-Up Robust Features), HOG (Histogram of Oriented Gradients). A classifier (e.g. Support Vector Machine) trained on the extracted vectors is applied to detect drones on sliding windows in an input image. This method requires skillful extraction to acquire relevant information for detection. Furthermore, the sliding window technique causes computationally costly exhaustive search. In [5], CBCs (Cascades of Boosted Classifiers) are trained on Haar feature (feature achieved by Harr-like transformation), HOG feature and LBP (Local Binary Pattern) feature for drone detection. In [6], SURF feature and Neural Network are used for drone detection.

The study in [7] first preprocesses an image by morphological operations to highlight potential drones. Then, hidden Markov models are employed to track and detect drones. The detection decision is made after target information is collected and collated over a period of time.

The method in [8] partitions video into overlapping slices. Each slice contains N frames. The accuracy of drone detection can be improved by increasing the number of overlapping frames. Spatio-temporal cubes (st-cubes) with different scales for width, height and time duration are created by sliding window technique. A motion compensation algorithm is used for st-cutes to create st-cubes with a target object (drone) at center. Then, boosted trees or Convolutional Neural Networks (CNN) are employed to categorize each st-cute as containing a drone or not. If more than one drones are detected for the same spatial location at different scales, the most confident one is reserved.

In [9], the Contiguous Outlier Representation via Online Low-rank Approximation (COROLA) technique is first employed for detecting the appearance of a small moving object in a frame and the CNN algorithm is applied for drone recognition.

Deep neural networks in some studies are used as begin-to-end drone detection models. YOLOv2 [10] and YOLOv3 [11] are used in [12], [13], [1] and Faster R-CNN [14] is used in [2] for drone detection.

In this paper, we propose a method combining two emerging convolutional neural networks: Faster R-CNN and YOLOv2 to detect drones in images. They both have lower layers that are convolutional layers. These convolutional layers take an image as input and output feature maps. The feature maps are then inputs for object localization and classification.

Faster R-CNN (for more detailed see [14]) joins the region proposal network RPN and the object detection network Fast R-CNN [15]. The two networks share convolutional layers. These layers have input of an image and output of feature maps. RPN takes the input image and produces region proposals and their objectness score. Region proposals are generated by sliding a small network with fully connected convolutional layers over the feature map. A spatial window of the feature map is taken as input for the small network. Each sliding window is mapped to a lower-dimensional feature. This feature is then taken as input for two sibling fully connected layers: a box-regression layer that outputs the encoded coordinates of k anchor boxes (also called anchors), and a box-classification layer that outputs 2-k scores estimating probability of object or not-object for each proposal. Fast R-CNN begins with convolutional and max pooling layers that take the input image and generate feature maps. Then, a region of interest (RoI) pooling layer uses max pooling to convert the features inside a region proposed by RPN into a small feature map with a fixed spatial extent. Next, fully connected layers map the small feature map to a feature vector. Finally, two fully connected sibling layers process the feature vector and outputs N bounding boxes with respect to N object classes and N+1 probability estimates for N object classes and background. A non-maximum suppression technique is independently used for each class to remove low confidence bounding boxes.

YOLOv2 (for more detailed see [10]) also starts with convolutional and max pooling layers. These lower layers are trained to extract high-level features. Then, the features from the layers at the two highest levels are combined to get the final feature map of an input image. YOLOv2 views the input image as a grid of SxS cells. Each grid cell is in relation with a set of anchor boxes. These anchor boxes' centers are the same with the grid cell's one. Their widths and heights are predefined by k-mean based on the objects' dimensions in the training data so that these dimensions best present the objects' dimensions. For each anchor box, YOLOv2 predicts a bounding box, a confidence score that reflects how confident the bounding box containing an object is, and conditional probabilities that the object belongs to classes. Then, low confidence bounding boxes are also filtered out as in Faster R-CNN.

The difference between Faster R-CNN and YOLOv2 is that Faster R-CNN proposes potential regions (containing objects) for classification and continues refining these regions while YOLOv2 preforms region detection and classification in just one time.

The following sections include: section 2 presents the proposed method, section 3 presents the experiments and results of drone detection by Faster R-CNN, YOLOv2 and the proposed method, and the last section is about conclusion.

2. Proposed method

The proposed method in this paper is rooted from our observation that some drones in images are detected by YOLOv2 while they are not detected by Faster R-CNN and vice versa. The method combines Faster R-CNN and YOLOv2 to detect drones so that when some drones are not detected by one network, they are detected by the other.

shows the activity diagram of the proposed method. A rounded rectangle represents an activity. A rectangle describes an activity's input or output data. A black circle represents the start node of the diagram. An encircled black circle denotes the end node of the diagram. A solid arrow represents a transition from one activity to another. A condition for a transition is written in square brackets. A dash arrow denotes a connection between an activity and its input or output data. A synchronization bar (a filled rectangle) indicates start or end of parallel activities. At first, Faster R-CNN and YOLOv2 are trained separately for drone detection. The input of this step is training images and the ground truth bounding boxes of drones in the training images. The output of this step is Faster R-CNN and YOLOv2 networks. Then, for each image taken from a set of testing images, the two networks are used to detect drones in parallel. This step produces the bounding boxes of detected drones and corresponding confidence scores. If Faster R-CNN's

minimum confidence score is greater than YOLOv2's one or YOLOv2's minimum confidence score is less than 0.5, then Faster's R-CNN detection results are selected, otherwise YOLOv2's ones are selected.

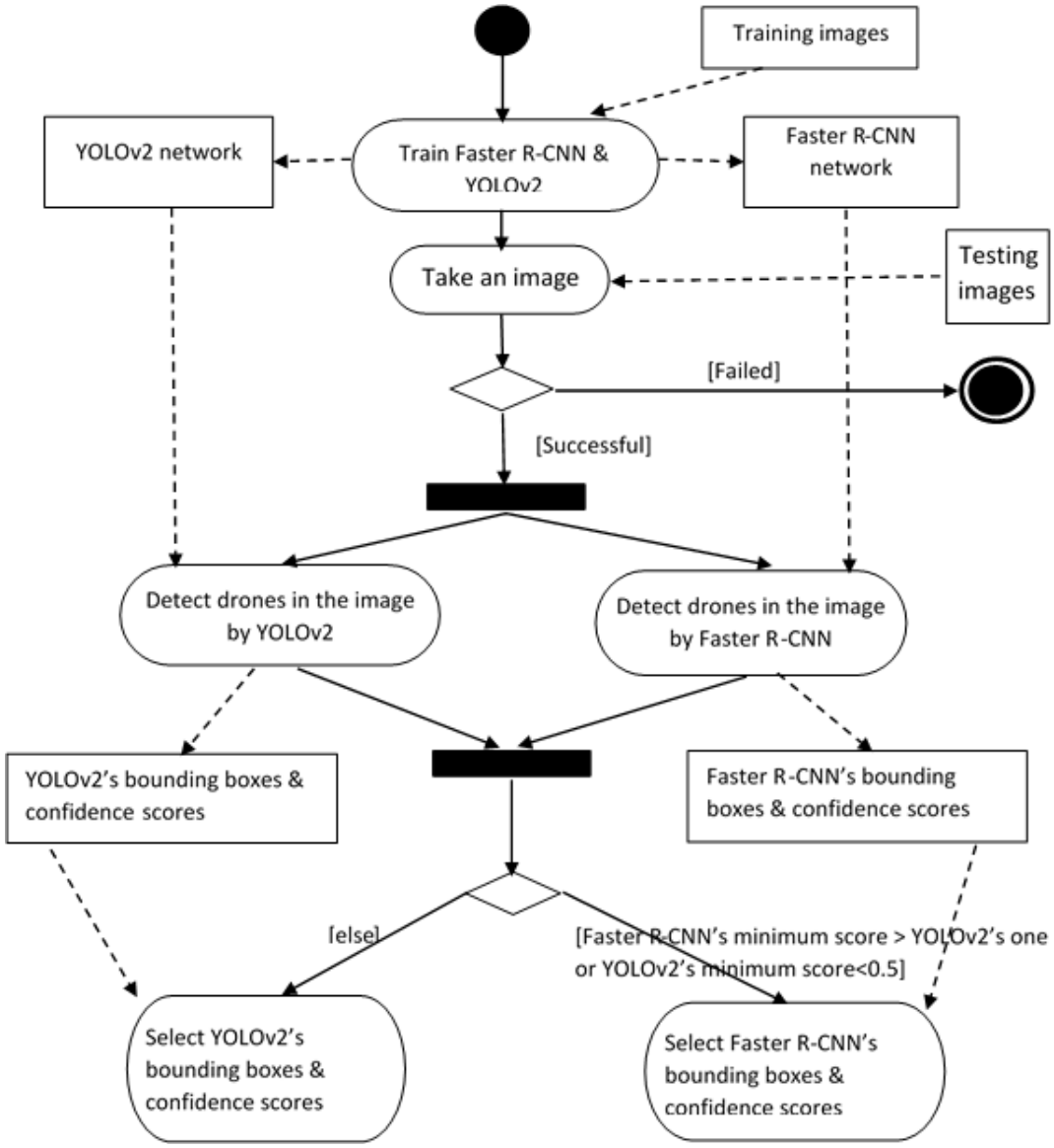

Figure 1. *Proposed method*

3. Experiments and results

In this section, dataset for training and testing Faster R-CNN and YOLOv2 networks for drone detection is first described. Parameters for training Faster R-CNN and YOLOv2 are then presented. The experimental results of testing Faster R-CNN, YOLOv2 and the hybrid approach are presented at last.

3.1. Training and testing dataset

A dataset of 498 images of the quadcopter DJI Phantom 3 obtained from Google image search tool, and screenshots from videos from YouTube [16] were used for training and testing Faster

R-CNN and YOLOv2 network and testing the hybrid approach. Training data took 350 images and testing data took 148 images.

The training data was augmented by randomly flipping original images and the bounding boxes horizontally at each iteration of a training epoch. This helps diversify the training data without having to increase the number of labeled training samples. The testing data was not augmented for unbiased evaluation. Figure 2 presents an original image (the left image) and its modified image (the right image) created by horizontal flip.

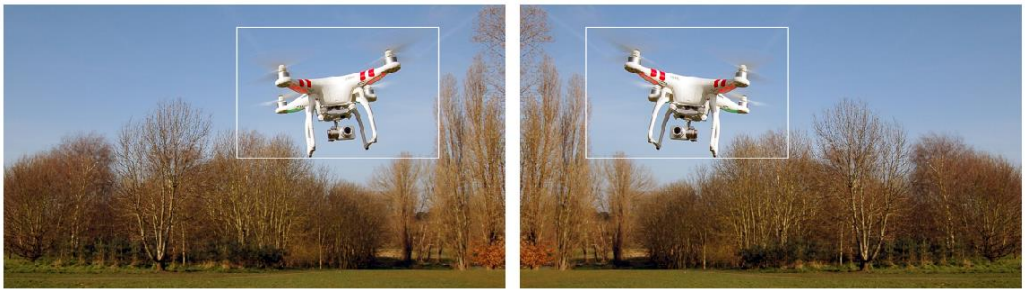

Figure 2. *Data augmentation*

3.2. Parameter settings

The values of training parameters for Faster R-CNN were set as the best ones that were experimentally determined in [2]. These parameters include learning rate, momentum co-efficient, maximum number of epochs, IoU ranges for negative and positive anchor boxes at each sliding window position, number of images to sample mini-batchs, number of anchor boxes at each sliding window and pretrained network. IoU is the ratio of intersection over union of a ground truth bounding box and an anchor box. These training parameters were described detailed in [2]. Their values are shown in Table 1**.** Table 2 presents those for YOLOv2**.** These values were determined after several trials. YOLOv2 does not require IoU ranges for positive and negative anchor boxes as Faster R-CNN.

Table 1. *Faster R-CNN training parameters*

Parameter	Value
Learning rate	0.001
Momentum co-efficient	0.09
Maximum number of epochs	30
IoU range for negative anchors	[0 0.3]
IoU range for positive anchors	[0.6 1]
#images to sample mini-batches	1
#anchor boxes	10
Pretrained network	vgg19

Faster R-CNN and YOLOv2 were respectively trained with parameter settings in Table 1 and Table 2 by stochastic gradient descent. Then, Faster R-CNN and YOLOv2 networks were combined as described in section 2 to detect drones in testing images.

Table 2. *YOLOv2 training parameters*

Parameter	Value
Learning rate	0.001
Momentum co-efficient	0.9
Maximum number of epochs	30
#images to sample mini-batches	5
#anchor boxes	7
Pretrained network	resnet50

3.3. Results

We used precision and recall metrics on the whole set of test images to evaluate the proposed method, where the metrics were calculated as the following equations. *TP*, *FP* and *FN* are the numbers of true positives, false positives, and false negatives of the prediction on the whole set of testing images respectively. A positive detection is true if the ratio of intersection over union of its predicted box and a ground truth box is greater than or equal to 0.5, otherwise it is false. The number of false negatives is the number of drones that were not detected.

$$presion = \frac{TP}{TP + FP} \quad (1)$$

$$recall = \frac{TP}{TP + FN} \quad (2)$$

Table 3 shows the precisions and recalls of Faster R-CNN, YOLOv2 and the hybrid approach. We can see that the precision and recall of the hybrid approach are almost 5% and more than 11% higher than those of Faster R-CNN, and are 3% and more than 6% higher than those of YOLOv2. This shows that Faster R-CNN and YOLOv2 can work together to improve the accuracy of drone detection.

Table 3. *Precision and recall comparison between different methods*

Method	Precision	Recall
Faster R-CNN	0.877	0.796
YOLOv2	0.896	0.847
Proposed method	0.926	0.908

4. Conclusion

In this paper, a hybrid method combining two emerging deep neural networks Faster R-CNN and YOLOv2 for drone detection was proposed. The two networks in the hybrid approach are first trained independently. Then, they are both used to detect drones in parallel. If the drone detection results of YOLOv2 are not confident, then those of Faster R-CNN are selected. The experimental results show that the hybrid approach can increase precision by almost 5% and 3%, and increase recall by more than 11% and 6% for Faster R-CNN and YOLOv2 respectively. This shows that Faster R-CNN and YOLOv2 can work together to more precisely detect drones.

REFERENCES

[1] E. Unlu, E. Zenou, N. Riviere, and P.-E. Dupouy, "Deep learning-based strategies for the detection and tracking of drones using several cameras," *IPSJ Transactions on Computer Vision and Applications*, vol. 11, no. 7, pp. 1-13, 2019.

[2] V. V. Pham, "A new approach using computer vision for drone detection," *TNU Journal of Science and Technology*, vol. 225, no. 11, pp. 11-18, 2020.

[3] NovoQuad, "ND-BU001 Standard Anti-Drone System," 2020. [Online]. Available: https://www.nqdefense.com/products/anti-drone-system/nd-bu001-standard-anti-drone-system/. [Accessed Jan. 5, 2021].

[4] DRONESHIELD, "DroneSentry: Autonomous Drone Detection & Countermeasure," 2020. [Online]. Available: https://www.droneshield.com/sentry. [Accessed Mar. 15, 2020].

[5] G. Fatih, Ü. Göktürk, S. Erol, and K. Sinan, "Vision-Based Detection and Distance Estimation of Micro Unmanned Aerial Vehicles," *Sensors*, vol. 15, no. 9, pp. 23805-23846, 2015.

[6] T. Ahmed, T. Rahman, B. B. Roy, and J. Uddin, "Drone Detection by Neural Network Using GLCM and SURF," *Journal of Information Systems and Telecommunication*, vol. 9, no. 33, pp. 15-24, 2021.

[7] L. Mejias, S. McNamara, J. Lai, and J. Ford, "Vision-based detection and tracking of aerial targets for UAV collision avoidance," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, 2010

[8] A. Rozantsev, V. Lepetit, and P. Fua, "Detecting Flying Objects Using a Single Moving Camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 5, pp. 879-892, 2016.

[9] A. Sharjeel, S. A. Z. Naqvi, and M. Ahsan, "Real time drone detection by moving camera using COROLA and CNN algorithm," *Journal of the Chinese Institute of Engineers*, vol. 44, no. 2, pp. 128-137, 2021.

[10] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017.

[11] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018. [Online]. Available: arXiv:1804.02767. [Accessed Jan. 5, 2021].

[12] C. Aker and S. Kalkan, "Using Deep Networks for Drone Detection," *IEEE International Conference on Advanced Video and Signal Based Surveillance*, Lecce, Italy, 2017.

[13] M. Wu, W. Xie, X. Shi, P. Shao, and Z. Shi, "Real-Time Drone Detection Using Deep Learning Approach," *International Conference on Machine Learning and Intelligent Communications*, Hangzhou, China, 2018.

[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Conference on Neural Information Processing Systems*, Montreal, Canada, 2015.

[15] R. Girshick, "Fast R-CNN," *IEEE International Conference on Computer Vision*, Santiago, Chile, 2015.

[16] C. Reiser, "Bounding box detection of drones (small scale quadcopters) with CNTK Fast R-CNN," 2017. [Online]. Available: https://github.com/creiser/drone-detection. [Accessed Jan. 5, 2021].