A Deep Learning Based Forest Fire Detection Approach Using UAV and YOLOv3

Zhentian Jiao[1], Youmin Zhang[2*], Jing Xin[1], Lingxia Mu[1], Yingmin Yi[1], Han Liu[1] and Ding Liu[1]
[1]*Shaanxi Key Laboratory of Complex System Control and Intelligent Information Processing, Xi'an University of Technology, Xi'an, Shaanxi, 710048, China*
[2]*Dept. of Mechanical, Industrial and Aerospace Engineering, Concordia University, Montreal, Quebec H3G 1M8, Canada*
Email: youmin.zhang@concordia.ca

Abstract—**Unmanned aerial vehicles (UAVs) are increasingly being used in forest fire monitoring and detection thanks to their high mobility and ability to cover areas at different altitudes and locations with relatively lower cost. Traditional fire detection algorithms are mostly based on the RGB color model, but their speed and accuracy need further improvements. This paper proposes a forest fire detection algorithm by exploiting YOLOv3 to UAV-based aerial images. Firstly, a UAV platform for the purpose of forest fire detection is developed. Then according to the available computation power of the onboard hardware, a small-scale of convolution neural network (CNN) is implemented with the help of YOLOv3. The testing results show that the recognition rate of this algorithm is about 83%, and the frame rate of detection can reach more than 3.2 fps. This method has great advantages for real-time forest fire detection application using UAVs.**

Index Terms—**Unmanned aerial vehicles, forest fire, YOLOv3, real-time detection**

1. Introduction

Forests are an important part of natural resources. They can provide habitat for animals, maintain biodiversity and purify the air. Known as the "Lung of the Earth", the forest has rich natural and social economic values. However, current environmental conditions, unfortunately, have made occurrence of wildfires more frequent, causing sizable areas of forest loss each year [1, 2]. The sensational California fire, occurred in November 2018, showed the serious harm of forest fires once more. Particularly, in view of the rapid spread of forest fires and the long burning time, early prevention of forest fires is one of the important means to protect natural resources and people [3].

With the development of technology, UAVs will hopefully become the most powerful tool for early detection of forest fires [1, 4]. It has enabled a large variety of applications, such as tracking [5], surveillance [6], and in particular mapping and land surveying [7]. They are also used in detection applications given their ability to cover open areas at different altitudes and provide high-resolution videos and images. Based on the information above, many related research activities on forest fire monitoring and detection based on UAVs have been carried out in recent years [2, 8–10].

In the researches that have been carried out, vision-based fire detection techniques are usually based on three characteristics: color, motion, and geometric features [11]. In particular, most researchers tend to combine the color and dynamic characteristics of the flame to provide a more reliable recognition. To the best of authors' knowledge, one of the first works on fire detection based on image processing was proposed in [12]. In this work, threshold processing is used in the region of interest (ROI) to distinguish the flame region and the non-flame region. The RGB/HSI color model was used in [13], dynamically analyzing the disorder characteristics of the flame and verifying the possibility of fire. The work in [8–10] solved the problem of effectively extracting fire pixels by taking advantage of the Lab color model. In addition, otsu segmentation method is used in [9], whose experimental results verify that the method can effectively extract forest fire pixels.

This paper aims to find an effective approach that can be applied to forest fire prevention with UAVs, and to improve the efficiency of fire detection. Therefore, we use a combination of aerial image and YOLOv3 algorithm [14]. Since 2012, researchers have proposed a variety of object detection algorithms and architectures, such as a region-based convolutional neural network (R-CNN) and its variants [15–17]. In 2016, Joseph Redmon proposed a method called "YOLO" (You Only Look Once) [18]. Unlike traditional region-based approaches, YOLO is a one-stage algorithm, passing the image only once in a fully convolutional neural network (FCNN), which makes it quite fast towards real-time applications. Compared to region-based technology, YOLOv2 [19] overcomes the relatively high localization error and low recall by making batch-normalization and higher resolution classifier. In 2018, YOLOv3 [14] was released with incremental improvement.

The rest of this paper is organized as follows. Section 2 gives a preliminary description of forest fire detection system with UAV. The image detection algorithm based on

This work is partially supported by the National Natural Science Foundation of China (No. 61573282, 61833013, and 61873200), and the Natural Sciences and Engineering Research Council of Canada

the YOLOv3 network will be introduced in Section 3. The experimental testing results of the algorithm are given and analysed in Section 4. At last, the final section summarizes the paper and looks forward to future works.

2. UAV-based Forest Fire Detection System

A typical UAV-based forest fire detection system is shown in Fig. 1. The UAV is equipped with a visible/infrared camera for image acquisition, and the onboard computer carried by UAV can perform local real-time image processing and mission planning. The data transmission system is responsible for the transmission of images data and flight commands. The ground station would detect and diagnose forest fire, after receiving the images and location information of potential fire spot(s) transmitted by the UAV in real time. On the other hand, the ground terminal can send desired operational command to the UAV for path planning and re-planning. Such a UAV system combines real-time control of the UAV and wireless communication and alarm for fire detection. Different algorithms could be implemented on this platform, which provides a more reliable guarantee to forest fire detection.

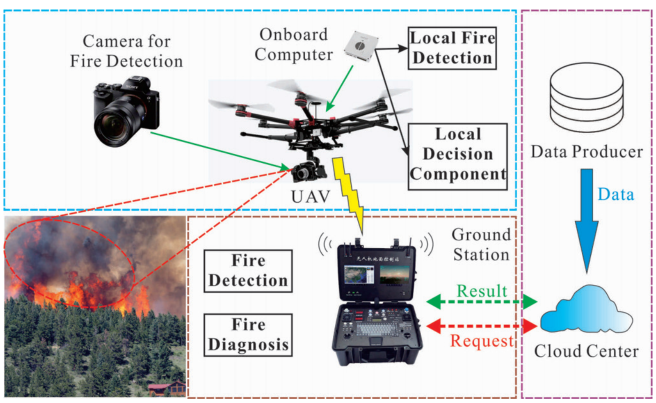

Figure 1: Components and functions of a UAV-based forest fire detection system.

The computing hardware configuration of the UAV and ground station is shown in Table 1.

TABLE 1: Configuration of the UAV system

Platform	Parameter	Description
UAV	Onboard computer	DJI MANIFOLD
	CPU	NVIDIA 4-Plus-1 ARM Cortex-A15
	RAM	2GB
	System environment	Ubuntu 14.04LTS
Gound Station	CPU	i7-8700K
	RAM	16GB
	Graphics card	RTX2080-8G
	System environment	Ubuntu 18.04LTS

In the process of forest fire detection, a large number of image data will undoubtedly be produced, which is a challenge to the system. At the same time, cloud computing is no longer suitable for such a video processing and transmission problem, because the transmission of large amounts of data in the network may cause network congestion, and the privacy of video data is difficult to guarantee.

Therefore, as shown in Fig. 1, edge computing is proposed in a forest fire detection system. The edge computing allows the cloud center to decentralize related requests. On the other hand, each edge node processes the request with the local video data and then returns only the relevant results to the cloud center, which not only reduces the data traffic but also ensures the privacy of the users. In this system, the UAV can be regarded as a user node, and the ground station and the site server are regarded as edge nodes. Using an edge computing architecture, we can increase the response time to image recognition of the system. In addition, with some of the computing tasks are offloaded from the cloud to the edge, energy consumption of the entire system could be reduced by 30% to 40%. What is more, it can be sped up by 20 times in terms of data integration and migration.

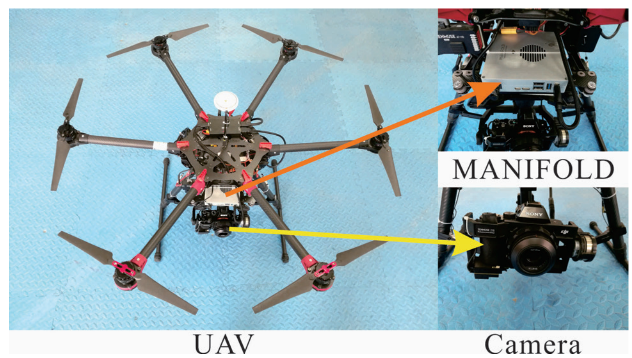

Figure 2: The UAV platform.

As shown in Fig. 2, a UAV platform with onboard computer and camera is built.

3. Detection Algorithm

The onboard computer has the characteristics of small size, lightweight and low power consumption, but its computing power is limited. In view of this, this paper proposes a YOLOv3-based algorithm, called YOLOv3-tiny, to improve the detection speed with reliable accuracy.

3.1. Architecture of YOLOv3

The YOLOv3 model can usually be divided into a feature extraction layer and a processing output layer. The feature extraction layer is a mixture of Darknet-19 and ResNet-like network, and the processing output layer is similar to the feature pyramid network (FPN). The basic component of the feature extraction layer is DBL, as shown in Fig. 3, which is the combination of Convolutional layer, BN layer (Batch Normalization in the middle) and Leaky ReLU layer. It can effectively prevent over-fitting without Dropout layer by using the BN layer. The Leaky ReLU

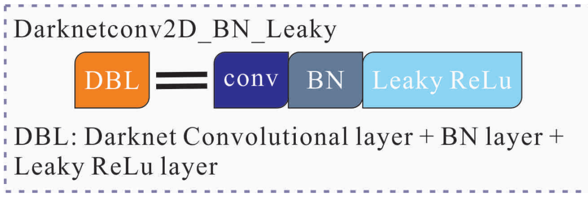

Figure 3: The structure of DBL.

layer retains the negative axis information of the training part.

Compared to the powerful hardware configuration on the ground station side, the Darknet-53 will no longer be suitable for the hardware platform on the UAV side. The network of YOLOv3-tiny is constituted by DBL with the classic max-pooling layer, which is used for features extracting from the image. The network structure is shown in Fig. 4. In particular, we added 4 DBL layers to the extraction of small feature map, which will enhance the ability of small target detection.

The first five max-pooling layers of the network, their $size = 2 \times 2$ and $stride = 2$, so the feature map will be reduced to 1/32 of the original input image size.

Due to the smoke and flame are not fixed in shape, they could appear in various sizes. Therefore, two YOLO layers are used for the detection to speed up the algorithm. When the input is 416×416, the size of the first time of output is 13×13×24 when the feature map is transfered to the 20th layer. Besides, the second feature extraction is performed on layer 27, resulting in an output of 26×26×24. 24 represents the network tensor size, which can be calculated by Eq. (1).

$$tensor = N \times N \times [(bounding\ box) \times (offset + object + class)] \quad (1)$$

where $N = 1$. Three bounding boxes are used in each YOLO layer, and each box needs to have five basic parameters $(x, y, w, h, confidence)$ [14], so $offset = 4$, $object = 1$. In order to improve the detection accuracy, we define the forest fire scenes as three scenarios of smoke, fire, smoke plus fire cases, so $class = 3$.

3.2. YOLOv3 bounding box prediction

YOLOv3 has unparalleled advantages in the speed and accuracy of object detection, mainly due to its superior algorithm.

The anchor boxes of YOLOv3 is learned from the anchor mechanism in the faster R-CNN region proposal network (RPN). However, YOLOv3 discards the approach of manual setting and uses the dimensional clustering method instead of determining the anchor box in apriori, which is obtained by K-means clustering [19]. In our network, the $priors = 9$. This method uses the IOU (intersection over union) score as the final evaluation standard, and more suitable bounding boxes can be found automatically [18]. Through the clustering method, six anchor boxes are selected based on the average IOU to predict the bounding box. The distance function used by clustering is shown in Eq. (2).

$$d\left(box, centroid\right) = 1 - IOU\left(box, centroid\right) \quad (2)$$

The idea of the algorithm for calculating the target bounding box is as follows. When an image is inputted, the target is first selected in the network to determine the center point. Then the input image will be divided into multiple $s \times s$ cells. After the coordinates of the point which is the center of the cell are calculated, the predicted bounding box will be produced by the coordinates of the center point. The coordinate calculation is as shown in Eq. (3).

$$\begin{cases} b_x = \sigma\left(t_x\right) + c_x \\ b_y = \sigma\left(t_y\right) + c_y \\ b_w = P_w e^{tw} b_h = P_h e^{th} \end{cases} \quad (3)$$

where the coordinates of the center point are (t_x, t_y, t_w, t_h), which represents the coordinates, width and height of the center point of the bounding box. (p_w, p_h) represents the width and height of the cell. (c_x, c_y) indicates the offset coordinate.

$$P_r\left(object\right) \times IOU\left(b, object\right) = \sigma\left(t_o\right) \quad (4)$$

YOLOv3 uses logistic regression when predicting bounding-box. Each time YOLOv3 predicts the bounding-box, it will output $(t_x, t_y, t_w, t_h, t_o)$ and then calculate the absolute (x, y, w, h, c) by Eqs. (3) and (4). Logistic regression is used to make an objectness score on the part enclosed by the anchor.

4. Experiments

We first trained the model on the desktop, then put the model on the DJI MANIFOLD embedded on a quadrotor UAV for testing. The model has been trained for 60,000 steps, and 64 images are used in each step. The network performance results are shown in Table 2.

TABLE 2: The network performance results

Parameter	Value
mAP@0.5	79.84%
mAP@0.75	65.91%
Average_IOU	70.23%
Precision	0.82
Recall	0.79
F1-score	0.81

The YOLOv3-tiny network has low FPS (frames per second) performance on DJI MANIFOLD onboard computer, but it can still achieve real-time detection. The results are shown in Table 3.

With the use of 60 images to test the model, the detection rate can reach 83%. Some of the testing results are shown in Fig. 5.

It can be seen from Fig. 5 that the algorithm sacrifices some detection accuracy. After adding the convolution layer to the YOLOv3-tiny network, the detection performance on the small target becomes better.

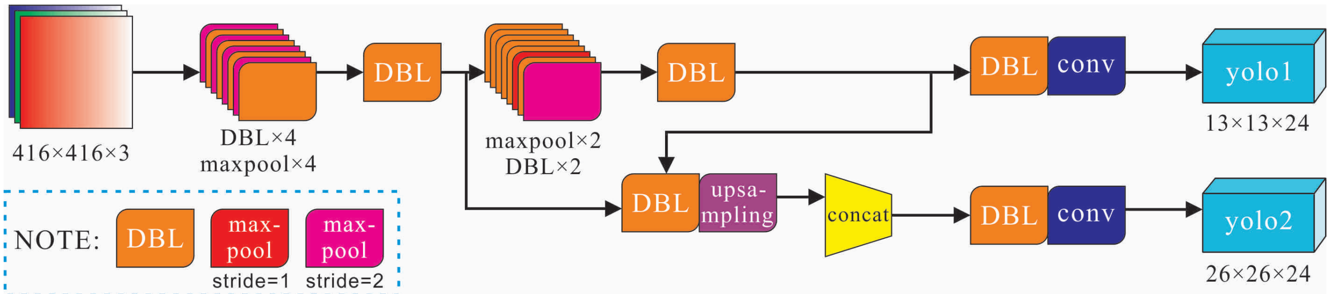

Figure 4: The structure of YOLOv3-tiny.

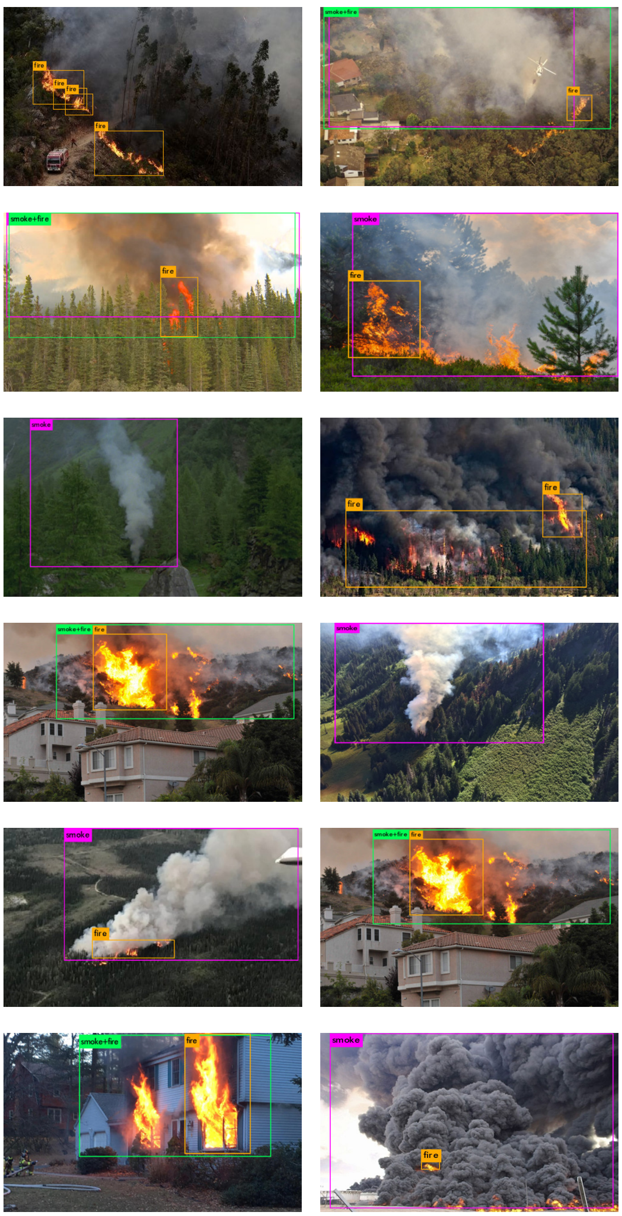

Figure 5: Testing results with developed detection algorithm.

TABLE 3: The network experiment results

Video name	Resolution	Frame rate	Frame rate of detection
Test1.mp4	400×240	29.97fps	6.5fps
Test2.avi5	1920×1080	25.00fps	3.2fps
Test3.mp4	1920×1080	30.10fps	3.2fps
Test4.mp4	1140×544	26.47fps	6.5fps
USB-camera	640×480	30.0fps	6.2fps

5. Conclusions

This paper aims at developing an efficient and reliable UAV testbed and image-based detection method for forest fire detection that could be implemented on small-scale UAVs. Therefore, the hardware platform for forest fire detection is developed first. At the same time, according to the hardware of the UAV platform, a corresponding YOLOv3 image detection algorithm is proposed. The experimental testing results show that accuracy and speed have reached the expected level, which proves the effectiveness and the feasibility of the developed UAV platform and the deep learning based fire detection algorithm.

However, we also found some problems in the experiments. The detection algorithm is sensitive to large-area forest fires, and the performance needs further improvements in small-scale (such as the small fire spots in the forest). That is mainly due to the lack of a large number of initial forest fire sets available for the training set. As our future works, more in-depth study with better detection performance will be achieved to meet practical requirements towards fast and precise forest fires early detection and prevention.

References

[1] C. Yuan, Y. M. Zhang, and Z. X. Liu, “A survey on technologies for automatic forest fire monitoring, detection, and fighting using unmanned aerial vehicles and remote sensing techniques,” *Canadian Journal of Forest Research*, vol. 45, no. 7, pp. 783–792, 2015.
[2] J. R. Martinez-de Dios, B. C. Arrue, A. Ollero, L. Merino, and F. Gómez-Rodríguez, “Computer vision techniques for forest fire perception,” *Image and Vision Computing*, vol. 26, no. 4, pp. 550–562, 2008.
[3] D. Kolarić, K. Skala, and A. Dubravić, “Integrated system for forest fire early detection and management,”

Periodicum Biologorum, vol. 110, no. 2, pp. 205–211, 2008.

[4] J. Everaerts *et al.*, "The use of unmanned aerial vehicles (uavs) for remote sensing and mapping," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 37, no. 2008, pp. 1187–1192, 2008.

[5] A. Koubaa and B. Qureshi, "Dronetrack: Cloud-based real-time object tracking using unmanned aerial vehicles over the internet," *IEEE Access*, vol. 6, pp. 13 810–13 824, 2018.

[6] G. R. Ding, Q. H. Wu, L. Y. Zhang, Y. Lin, T. A. Tsiftsis, and Y. D. Yao, "An amateur drone surveillance system based on the cognitive internet of things," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 29–35, 2018.

[7] A. Tariq, S. Osama, and A. Gillani, "Development of a low cost and light weight UAV for photogrammetry and precision land mapping using aerial imagery," in *2016 International Conference on Frontiers of Information Technology*. IEEE, 2016, pp. 360–364.

[8] C. Yuan, Z. X. Liu, and Y. M. Zhang, "UAV-based forest fire detection and tracking using image processing techniques," in *2015 International Conference on Unmanned Aircraft Systems*. IEEE, 2015, pp. 639–643.

[9] C. Yuan, Z. X. Liu, and Y. M. Zhang, "Fire detection using infrared images for uav-based forest fire surveillance," in *2017 International Conference on Unmanned Aircraft Systems*. IEEE, 2017, pp. 567–572.

[10] C. Yuan, Z. X. Liu, and Y. M. Zhang, "Aerial images-based forest fire detection for firefighting using optical remote sensing techniques and unmanned aerial vehicles," *Journal of Intelligent & Robotic Systems*, vol. 88, no. 2-4, pp. 635–654, 2017.

[11] T. Celik, H. Ozkaramanlt, and H. Demirel, "Fire pixel classification using fuzzy logic and statistical color model," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, vol. 1. IEEE, 2007, pp. I–1205.

[12] V. Cappellini, L. Mattii, and A. Mecocci, "An intelligent system for automatic fire detection in forests," in *Recent issues in pattern analysis and recognition*. Springer, 1989, pp. 351–364.

[13] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in *2004 International Conference on Image Processing, 2004. ICIP'04.*, vol. 3. IEEE, 2004, pp. 1707–1710.

[14] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

[16] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1440–1448.

[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[19] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7263–7271.