








Fire Detection Method Based on Infrared Image and Visible Image

Tong Zhang¹ , Hui Xue²  , Shiqiang Wang¹ , and Dongfang Zhang¹ 

¹ Beijing Institute of Remote Sensing Equipment, Beijing 100854, China

² XiangXi Vocational and Technical College for Nationalities, Xiangxi 416000, Hunan, China
3503112@qq.com

Abstract. This paper presents a fire detection method using infrared and visible images with the same angle of view. The gray values of each point of the infrared image are binarized, connected area analysis and shape judgment processing, which are used to detect the scene that has been on fire, and the fire detection and location are realized in combination with the specific location of the observation platform; The visible image is used to detect the scene with smoke, and the deep learning method based on the improved Yolo framework is used to process the image to realize the early warning of fire. After the actual deployment test, the probability of correctly identifying the characteristics of fireworks remains above 91%, which can realize the detection of fire and early warning of fire, give the specific location information of the fire point, and meet the requirements of real-time.

Keywords: Fire detection · Object detection · Infrared image · Visible image

1 Introduction

1.1 A Subsection Sample

Fire is a kind of disaster with the highest probability among many social and natural disasters, which seriously threatens human property and even life safety, destroys the peace of the ecological environment, and will cause huge economic losses and bad political and social repercussions. Analyzing the infrared image can detect the fire location according to the flame characteristics, but the details of the surrounding environment of the fire are not clear; Analyzing the visible light image can locate the fire location according to the smoke characteristics, but affected by a large amount of smoke, it can not accurately evaluate the fire scale.

The traditional method of flame detection by collecting the information of temperature and smoke sensors has the disadvantages of high false alarm rate and poor real-time performance. The flame detection method based on image recognition has the advantages of fast response and intuitive post tracking. Literature [1] comprehensively summarizes the characteristics of flame in infrared images. Literature [2] improves the success rate of smoke detection by analyzing the characteristics of flame changes between frames in the

video stream and adopting the method of deep learning model. Literature [3] focuses on smoke detection methods facing complex backgrounds. The above fire detection methods have not achieved outstanding results in real-time detection, and they all use a single video information for detection, which is not easy to trace back and repeat.

On this basis, a dual light pod UAV system is designed, which ensures the real-time performance and improves the accuracy of fire detection:

- (1) On the basis of the structure of the basic yolov2 model, the residual structure is added to avoid the problem of gradient disappearance, ensuring the accuracy of smoke detection as high as possible, and the convolution layer is used to replace the maximum pooling layer in the network structure, which not only improves the speed of smoke detection, but also expands the number of channels, ensuring that the fire detection system can add new information for disaster analysis in the future;
- (2) Based on the structure of the basic yolov2 model, the FPN structure of multi-scale prediction is added, which improves the detection ability of the target detection model for targets of different sizes and improves the accuracy of fire detection.

2 Fire Detection Principle

2.1 Flame Detection Based on Threshold Segmentation

Fire is a kind of uncontrolled combustion, and the occurrence of fire is a development process. Flame is an important physical feature of fire, and flame shows many posture characteristics in the process of fire. The so-called posture characteristics refer to the characteristics of the flame in an image, while the dynamic characteristics refer to the characteristics of the flame in the image as time changes. The static characteristics of flame include temperature characteristics, sharp angle and circular angle. Infrared imaging has the characteristics of all-weather and all-time, and has better detection effect than visible light imaging at night. Infrared thermal imaging technology has a long detection distance, a wide range, and is less affected by weather factors. For example, infrared detectors can penetrate fog, snow and other related obstacles to achieve the purpose of collecting infrared images, and are not affected by strong light and flash in the environment. However, the traditional method of smoke and temperature sensor detecting flame can not be applied outdoors.

When a disaster occurs, the temperature of the flame is generally higher than the temperature of the surrounding environment. Using infrared imaging, many non fire interference sources can be physically eliminated. In infrared images, the value range of pixel gray value is 0 to 255, 0 is black and 255 is white. The flame temperature is high, so the gray value is high; The temperature of the background is low, so the gray value is also low. The purpose of detection is to extract all targets in the infrared image. The specific steps are: binarization processing, connected area analysis and shape judgment of the original infrared image [4].

Each pixel in the infrared image has only a gray value, so a threshold value is determined in advance, and the pixel values in the image with gray values greater than this threshold are set to 255, and those less than the threshold are set to 0. In this way, the flame and the background are separated, and a binary image is formed, that is, all pixel values in the image are 0 or 255.

In practical application, the segmentation threshold is different in different scenes, and it is necessary to test the appropriate detection threshold according to the specific situation. However, at this time, the segmented target image is incomplete, and too many independent targets will be formed. Therefore, it is also necessary to mark the connected region of the binary image, fuse the adjacent target points into a target, and determine whether it is a flame target in combination with the shape criterion.

2.2 Smoke Detection Based on Deep Learning

Unlike the infrared image, which only contains gray information, each pixel of the visible image also contains chrominance information, so the threshold segmentation method cannot be used for smoke detection. In addition, in this system, the infrared video signal is 50 Hz. After testing, the deep learning method is used to detect the flame, and the detection time of each frame is more than 20 ms. Therefore, the deep learning method is not used for infrared image flame detection.

Smoke is an obvious feature in the early stage of fire. Traditional smoke detection is to detect the concentration of smoke through the concentration sensor, and then make a response judgment. Early common fire alarms used sensors that could detect smoke and temperature. Due to their own properties, these sensors are sensitive to the temperature, smoke particles, thermal infrared and other reactions of the fire site, and can detect the corresponding information is to trigger the alarm device. Although these traditional detectors are cheap and have advantages in detection speed and accuracy, it is difficult for traditional smoke sensors to achieve good detection results in outdoor open spaces such as forests.

The deep learning network structure is mainly composed of backbone network, neck layer and prediction head. Backbone network refers to the network that extracts features. Its function is to extract the information in the image for the later network to use. Prediction head refers to the network that has the ability to obtain the output content of the network, and uses the previously extracted features to predict the type and location of the target. The neck layer is a network layer between the backbone network and the prediction head, which is used to collect the feature maps in different stages, so as to make better use of the features extracted by the backbone network.

Yolov2 detection framework [5, 6] is a classic deep learning target detection framework. Even today, with the rapid development of a variety of target detection methods, the structure of its source code written in C language still maintains good readability, which allows researchers to easily modify the network structure.

Its network structure is as follows (Fig. 1):

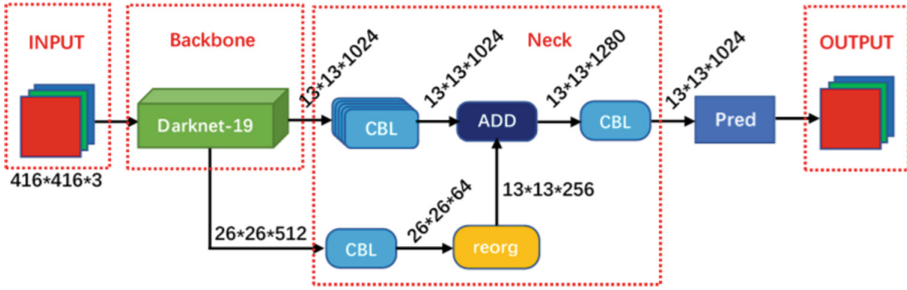


Fig. 1. Yolov2 structure diagram

Among them, CBL module is composed of a convolution layer, a BN layer and LeakyReLU activation function (Fig. 2).



Fig. 2. CBL module composition

First, the image data of $416 \times 416 \times 3$ is sent to the darknet-19 backbone network. After several convolution and pooling operations, the feature matrices of $13 \times 13 \times 1024$ and $26 \times 26 \times 512$ are output. After multi-scale training, the generated $13 \times 13 \times 1024$ feature map is sent to the prediction head to obtain the final detection result matrix. The number of channels is changed from 1024 to $K(1+c+4)$, where k is the number of target types to be predicted, 1 indicates whether there are targets in this grid, and the presence is 1, Otherwise, it is 0; 4 represents the horizontal and vertical coordinates and width and height of the target, and C represents the probability of various types of targets.

IOU refers to the IOU of the prediction box and the real box. PR refers to the probability that the box contains objects. It is not a classification prediction for each category, but can be understood as the prediction of foreground and background.

When yolov2 detects objects in pictures, each grid area will have the same number and size of anchor boxes [7], or known as a priori box, which is used to provide the prior information of the size of the bounding box, so that the network can adjust the anchor box to get the final prediction box only by learning the offset (Fig. 3).

The blue box is the prediction box, and the dotted box is the anchor box. The parameters of the prediction box are calculated as follows:

$$b_x = \sigma(t_x) + c_x \tag{1}$$

$$b_y = \sigma(t_y) + c_y \tag{2}$$

$$b_w = p_w e^{t_w} \tag{3}$$

$$b_h = P_h e^{t_h} \tag{4}$$

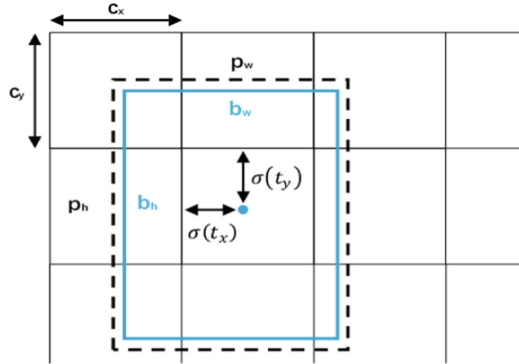


Fig. 3. Anchor box prediction process

$$\sigma(t_o) = PR(object) * IoU(b, object) \tag{5}$$

When calculating the horizontal and vertical coordinates of the target center of the prediction box, the sigmoid function is used to limit the predicted TX ty to ensure that it will not exceed 1, and then add the number of grids from the grid where the target center is located to the upper left corner of the image to obtain the horizontal and vertical coordinates of the target prediction center. When calculating the width and height of the prediction box, the exponential function is used to limit the prediction value TW ty to ensure that it is greater than 0.

Analyze the difference between the sample label and the network output, that is, the loss function, combined with the construction of the labeled real label of the object, train the depth network. The loss function of yolov2 consists of five parts.

The expression of frame center point error is:

$$L_{XY} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B I_{ij}^{obj} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] \tag{6}$$

where λ_{coord} is to adjust the parameters of the loss function, S represents the size of the grid, S^2 represents the number of grids, B represents the number of anchor boxes, and I_{ij}^{obj} represents that if there are targets in the current and lower grids, it is 1, otherwise it is 0. x represents the abscissa value of the predicted target center, \hat{x} represents the abscissa value of the real target center, y represents the ordinate value of the predicted target center, and \hat{y} represents the ordinate value of the real target center.

The expression of border width and height error is:

$$L_{WH} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B I_{ij}^{obj} \left[(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \right] \tag{7}$$

where w represents the width of the predicted target, \hat{w} represents the width of the real target, h represents the height of the predicted target, and \hat{h} represents the height of the real target.

When there are objects in the grid, the expression of confidence error is:

$$L_{obj} = \sum_{i=0}^{S^2} \sum_{j=0}^B I_{ij}^{obj} (C_i - \hat{C}_i)^2 \quad (8)$$

where, the value of \hat{C} is determined by whether the grid cell's bounding box is responsible for detecting an object. If it is responsible, it is 1, otherwise it is 0. Since each grid contains anchor boxes with the number of B , there is an anchor box, and the IOU value of the ground truth box of the current object is the maximum, then under the current j , the value is 1, otherwise the value is 0.

When there is no object in the grid, the expression of confidence error is:

$$L_{NOobj} = \lambda_{NOobj} \sum_{i=0}^{S^2} \sum_{j=0}^B I_{ij}^{NOobj} (C_i - \hat{C}_i)^2 \quad (9)$$

where, I_{ij}^{NOobj} means that if there is a target in the current and lower grids, it is 1, otherwise it is 0. λ_{NOobj} is to adjust the parameters of the loss function, because this formula is used to calculate the grid without the target center point. In an image, the grid without the target center point accounts for the majority. In order to prevent this item from causing too much impact on the result of the loss function, this parameter is introduced.

The expression of object classification error is:

$$L_{classes} = \sum_{i=0}^{S^2} \sum_{c \in classes} [p_i(c) - \hat{p}_i(c)]^2 \quad (10)$$

where AA represents the category of the real box and BB represents the category of the prediction box.

The complete loss function is the sum of the above five formulas:

$$L_{total} = L_{XY} + L_{WH} + L_{obj} + L_{NOobj} + L_{classes} \quad (11)$$

3 Methods Used in This Article

3.1 Infrared Image Flame Detection

The connected region refers to the pixel set composed of adjacent pixels with the same pixel value. In a binary image, adjacent sets with the same pixel values are the targets. We can use connectivity analysis to find each connected area and give it a unique identification. After marking, we can get all the targets in the infrared image. The method of finding the connected region in the image adopts the twice traversal method. Each scan follows a principle: start from the upper left pixel, scan a line from left to right, and then continue scanning from the leftmost pixel in the next line.

During the first scan, read the gray value of the pixel at the current position. If the gray value here is 255, continue to judge the gray value of its neighborhood. The

neighborhood of a pixel refers to its upper adjacent pixels, left adjacent pixels and upper left adjacent pixels. If the pixel values in its neighborhood are all 0, the pixel is given a new label value; If there is a pixel with a gray value of 255 in its neighborhood, the minimum label value in its neighborhood is assigned to the pixel, and it is recorded that the labels of these neighborhoods belong to the same connected area.

Then, the image is scanned twice, and the gray value of the pixel point at the current position is read. If the value is 255, the minimum label value belonging to the same connected region is found, which is assigned to the pixel. After two scans, the adjacent pixels in the image are combined into a region, and all the targets in the image are obtained, as shown in the figure (Fig. 4).

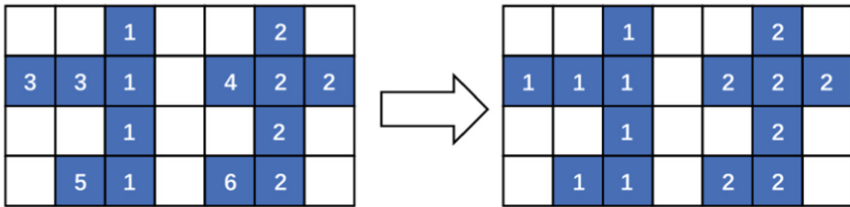


Fig. 4. Analysis process of quadratic ergodic connected domain

After obtaining all the targets in the image, some of them are not flames, which requires further judgment. Because the flame is of irregular shape, and most of the other interferences are of regular shape, the shape is used to judge the flame target. This method uses two indicators to judge its shape, roundness and rectangularity, which measure the degree of shape regularity.

Among them, the calculation formula of roundness is:

$$Roundness = \frac{4\pi S}{L^2} \tag{12}$$

where, S is the area of the target and L is the perimeter of the target. Analyze the flame image. According to the data set, the roundness of the flame is generally less than 0.3, and the roundness of other high-temperature interference objects is generally more than 0.5. However, if the distance is far or the flame area is too small, because there are too few pixels, the roundness will also be higher.

Another measure is rectangularity, which is calculated as follows:

$$rect = \frac{S}{S_R} \tag{13}$$

where s is the target area and S_R is the area of the smallest rectangle surrounding the target.

Analyze the flame image, in which the rectangularity of the flame is generally between 0.2 and 0.5, while the rectangularity of other high-temperature interference objects is generally more than 0.6, and only a few angles occasionally fall below 0.5.

Combining the above two indicators, the flame target is obtained by constraining the area, roundness and rectangularity of the target obtained after the connected domain analysis.

3.2 Smoke Detection Based on Improved Yolo Model

New Convolution Module. Based on darknet-19, the backbone network used by yolov2, and referring to RESNET’s residual layer idea [8], the backbone network is improved, and the specific network structure is introduced:

First, the convolution layer is responsible for convolution operation, mapping high-dimensional image information to low-dimensional feature map; BN layer allows users not to set initialization parameters, but also to choose a relatively large initial learning rate to improve the training speed, and can also prevent over fitting to improve the generalization ability of the network; The leakyrelu activation function can control the gradient explosion and prevent the gradient from disappearing during the training process, and can accelerate the convergence speed and improve the training speed.

The second is residual module res unit. The residual structure of RESNET model is used for reference, and the problems of gradient explosion and gradient disappearance are alleviated through cross layer connection (Fig. 5).



Fig. 5. Res unit structure diagram

Finally, the ResX module, which is composed of a convolution layer and a residual module with a number of X, completes the downsampling operation instead of using the maximum pooling layer, while retaining the semantic features of the deep convolution results and the image features of the shallow convolution results (Fig. 6).



Fig. 6. ResX unit structure diagram

After the output of different number of resx modules, it can be used as the output of feature maps of different sizes. The function of the output of feature maps of different sizes will be mentioned later.

Multiscale Prediction. After the feature map of the target is extracted from the backbone network, in order to better integrate the extracted features [9], the output of the backbone network is sent to the rear network. After up sampling, the feature map with the size of 13 * 13 is sent to the concat layer together with the feature map with the size of 26 * 26 output from the middle layer of the backbone network for tensor splicing, that is, keep the size of the feature map unchanged, and connect the number of channels to the size of 26 * 26 * 512; Similarly, carry out the sampling operation on the feature

map with the size of $26 * 26$ to obtain the feature map with the size of $52 * 52$. This operation is to make better use of the features extracted from the backbone network to form an FPN operation.

In convolution networks, semantic features are more obvious after deep network processing, while image features are more preserved after shallow network processing. The FPN operation in the figure makes the shallow detail information and deep semantic information of the network fully integrate, forming a pyramid structure with strong image features on multiple scales. The improved Yolo model can detect targets of different sizes more accurately and improve the accuracy of fire smoke detection.

The improved Yolo network structure is as follows (Fig. 7):

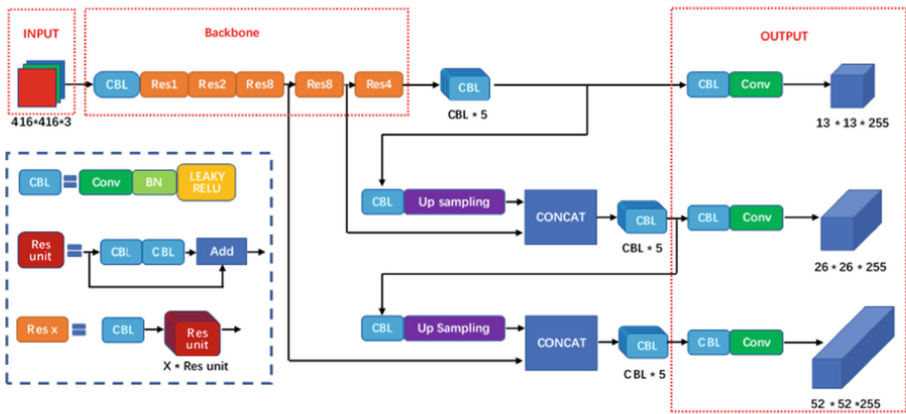


Fig. 7. Improved Yolo network structure diagram

Structure of Loss Function. According to the training process, for each input image, the model will output three sizes of feature maps. And on each cell, a prediction box will be generated according to the anchor box of three sizes, which together with the probability of foreground or background and the probability of category constitute the prediction information output by the target detection network.

From the perspective of supervised learning, the input image itself has the existence, location and category information of the actual target box, and then according to the anchor box correspondence mechanism, the information of the prediction box and the real box can be correlated correspondingly, that is, under the premise of three different size feature maps, the deviation between the prediction box and the real box can be calculated from the probability, location information and category information of the foreground or background, In this way, the loss function under each size is obtained, and the overall loss function is the sum of the loss functions under the three sizes.

Firstly, the expression of position error is given, which represents the difference between the prediction box generated by the network and the marked real box:

$$L_{location} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B I_{ij}^{obj} \left[(x_i^j - \hat{x}_i^j)^2 + (y_i^j - \hat{y}_i^j)^2 + (w_i^j - \hat{w}_i^j)^2 + (h_i^j - \hat{h}_i^j)^2 \right] \quad (14)$$

where λ_{coord} is to adjust the parameters of the loss function, S represents the size of the grid, S^2 represents the number of grids, B represents the number of anchor boxes, and I_{ij}^{obj} represents that if there is a target in the grid under the current i and j , it is 1, otherwise it is 0. x represents the abscissa value of the predicted target center, \hat{x} represents the abscissa value of the real target center, y represents the ordinate value of the predicted target center, and \hat{y} represents the ordinate value of the real target center. w represents the width of the predicted target, \hat{w} represents the width of the real target, h represents the height of the predicted target, and \hat{h} represents the height of the real target.

The anchor box calculates the confidence error when it is responsible for a goal or not. The weight coefficient is added to calculate the loss part without objects. In an image, most meshes have no targets, so after training, the model will be more inclined to calculate all meshes as having no targets, which is not conducive to model training. Therefore, we should reduce the mesh weight without object parts. Here is the expression of confidence error:

$$L_{confidence} = \sum_{i=0}^{S^2} \sum_{j=0}^B I_{ij}^{obj} \left[\hat{c}_i^j \lg(c_i^j) + (1 - \hat{c}_i^j) \lg(1 - c_i^j) \right] - \lambda_{NOobj} \sum_{i=0}^{S^2} \sum_{j=0}^B I_{ij}^{NOobj} \left[\hat{c}_i^j \lg(c_i^j) + (1 - \hat{c}_i^j) \lg(1 - c_i^j) \right] \quad (15)$$

Finally, the classification error, choose cross entropy as the loss function. Only when the j th anchor box of the i th grid is responsible for a real target, the bounding box generated by this anchor box will calculate the classification loss.

$$L_{classes} = \sum_{i=0}^{S^2} I_{ij}^{obj} \sum_{c \in classes} \left[\hat{p}_i^j \lg(p_i^j) + (1 - \hat{p}_i^j) \lg(1 - p_i^j) \right] \quad (16)$$

where, p_i^j represents the category of the real box, \hat{p}_i^j represents the category of the prediction box, and also adopts the form of binary cross entropy.

After obtaining the loss function, you can start training and update the network parameters in reverse iteration, so that the value of the loss function becomes smaller and smaller. The final loss function is as follows:

$$L_{total} = L_{location} - L_{confidence} - L_{classes} \quad (17)$$

4 Experimental Analysis

4.1 Experimental Setup

The CPU used in the experiment is Intel Core i9-10900x, the GPU is NVIDIA geforce GTX 1080ti, the operating system is Ubuntu 18.04 LTS, the programming language is python3.6.9, the acceleration framework is cuda10.2 and cudnn7.6.5, the machine learning framework is pytorch1.6.0, and the target detection program finally runs on the GPU development board NVIDIA Jetson xavier nx.

Suitable super parameters can train the network more effectively, get better results, and prevent the phenomenon of over fitting or unable to fit. The super parameters used include: 79 for batch, 79 for batch_ The size is 128, the iteration is 79, the learning rate is 0.01, and 300 epochs are trained.

4.2 Dataset

This paper is to detect and recognize the fire, using the video data taken in the laboratory and outdoors by the infrared imager and visible imager, and get the required infrared image and visible image by the method of frame extraction. After cooperating with the emergency management department of a province, the video data of some local forest fires were obtained. It is also mixed with some fire pictures collected on the network to get the data set used in this training. After random allocation, 7777 pieces were selected as the training set and 3333 pieces as the test set.

The data annotation format is TXT text, the content is the horizontal and vertical coordinates of the normalized target center point and the normalized target box width and height, and the category of the target is recorded.

4.3 Experimental Result

Results After Introducing the new Residual Module. First, we use the network structure of yolov2 to train the data set, train 300 epochs, get the optimal training weight by close fitting, and recognize it on the test set, and get the test result. The map value is 80.33%, and the average time to recognize an image is 40 ms.

Then a new residual module is introduced, and the same number of epochs are trained on the training set. The test results show that the AP value is 82.56%, and the average time to recognize an image is 25 ms.

Mark the prediction box of the test results on the test image. It can be seen that after the introduction of the new residual module, the recognition of smoke is more accurate, and it is known that the new residual module can improve the effect of smoke detection.

Multiscale Prediction Results. The multi-scale prediction of FPN structure is introduced into the network structure, and the input image size is controlled at 416 * 416.

Firstly, the network of yolov2 detection feature layer is used to train the data set, and then the FPN structures of $13 * 13$, $26 * 26$ and $52 * 52$ are introduced into the network structure to train 300 epochs respectively. After the introduction of FPN layer, the map value of network test is 85.22%, which is higher than 81.57% of yolov2.

On the image, it can be seen that more smoke positions are recognized by using the FPN structure, which shows that the structure of this paper can make the model have better detection performance for targets of various sizes. The comparison of detection before and after the introduction of multi-scale prediction is shown below (Fig. 8).



Fig. 8. Before (left) after (right)

4.4 Deploy Experimental Results

This algorithm is deployed on the edge computing device NVIDIA Jetson xavier NX and installed on the UAV of the dual light pod, as shown in the figure (Fig. 9):



Fig. 9. UAV with NVIDIA Jetson xavier NX

The identification process and schematic diagram are shown in the figure (Fig. 10):



Fig. 10. Fire image detection results

The recognition results are tabulated as follows:

Table 1. Identification result statistic

Image type	Total videos	Correct detection	Accuracy rate	Detection time per frame
Infrared image	35	32	91.43%	12 ms
Visible image	80	73	91.25%	25 ms

It can be seen from Table 1 that this method can effectively identify the simulated ignition point with high accuracy, and the detection time is about 25 ms, which can meet the image frame rate of 25 Hz and the requirements of real-time.

In this paper, a real-time fire detection and recognition method using image is proposed, which uses infrared image as threshold segmentation to detect fire and visible image as deep learning to detect smoke, respectively, to get more accurate detection results. The accuracy rate of infrared image detection is 91.43%, and the accuracy rate of visible image detection is 91.25%. After the transplantation of edge intelligent equipment, it can meet the real-time requirements of image frame rate of 25 Hz. It can effectively detect the ignition point information in ground experiments and flight experiments, which can meet the needs of fire detection in engineering. It can receive infrared and visible images from the same angle of view, combine the advantages of the two images to detect the fire, and record the location and time of the fire, which can be used by disaster relief personnel to remotely evaluate the disaster situation, and is conducive to the follow-up rescue process and post review summary.

References

1. Qi, M., Chen, B.: Forest fire detection algorithm based on aerial image. In: Liang, Q., Wang, W., Mu, J., Liu, X., Na, Z. (eds.) *Artificial Intelligence in China*. LNEE, vol. 854, pp. 465–472. Springer, Singapore (2022). https://doi.org/10.1007/978-981-16-9423-3_58
2. Li, M., et al.: Early forest fire detection based on deep learning. In: *3rd International Conference on Industrial Artificial Intelligence* (2021)

3. Zhao, M., Zhang, W., Wang, X., Liu, Y.: A smoke detection algorithm with multi-texture feature exploration under a spatio-temporal background model. *Hsi-An Chiao Tung Ta Hsueh/J. Xi'an Jiaotong Univ.* **52**(8), 67–73 (2018)
4. Shao, Z., Wang, H., Dong, Z., Yuan, Y., Li, J., Zhao, L.: Early bruises detection method of apple surface based on near infrared camera imaging technology and image threshold segmentation method. *Nongye Jixie Xuebao/Trans. Chin. Soc. Agric. Mach.* **52**, 134–139 (2021)
5. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 273–279 (2016)
6. Redmon, J., Divvala, S., Girshick, R., et al.: You only look once: unified, real-time object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788 (2016)
7. Zhang, S., Chi, C., Yao, Y., Lei, Z., Li, S.Z.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 9756–9765 (2020)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 9 December 2016
9. Lin, T.-Y., Goyal, P., Cirshick, R., He, K., Dollar, P.: Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **42**(2), 318–327 (2020)