Improved YOLOX Fire Scenario Detection Method

Jianfei Zhang and Sai Ke

School of Computer & Information Engineering, Heilongjiang University of Science & Technology, Heilongjiang 150027, China

Correspondence should be addressed to Jianfei Zhang; zjfnefu2021@163.com

Received 20 January 2022; Accepted 23 February 2022; Published 10 March 2022

1. Introduction

Conventional fire detection approaches are based on sensing detection of smoke and temperature or data monitoring with electric valve and vent valve. With the development of deep learning, Yu and Liu [1] employed the improved Mask R-CNN (Mask Regions with CNN Features) for identification and segmentation of flame images, adopting the bottom-up feature fusion and improved loss function for high precision detection. Wu [2] replaced the backbone network of YOLOv3 (You Only Look Once Version 3) [3] with DenseNet121 (Densely Connected Convolutional Networks), to improve the capture ability of the backbone network for flame and smoke features, and introduced focal loss for regression. Zhao et al. [4] proposed a CenterNet [5] algorithm-based target detection approach in complicated environment. Li et al. [6] proposed to use a network structure with depth separable convolution to improve the flame detection model and adopted several data augmentation techniques to increase the detection precision. Luo [7] adopted YOLOv4 (You Only Look Once Version 4) framework-based UAV (Unmanned Aerial Vehicle) for real-time flame detection.

However, the models of the above methods are complicated and difficult to deploy, with high calculation load and few detection targets. To solve such problems, an improved T-YOLOX detection model was proposed in our study, for multitarget detection of flame, smoke, and persons in complicated fire scenarios.

The method is based on YOLOX [8] architecture, with a light attention module for adjusting and improving the weight of each channel, to improve the overall feature extraction ability of the network; incorporated with the channel shuffle technique, to improve the communication ability between channels, increase the complexity level of the model, and avoid overfitting; and replaced the last layer of the backbone network with MobileViT (Mobile-friendly Vision Transformer) [9] module, for adding the learning ability of the backbone network for global features with this light transformer [10] module and improving the generality of the model. The experiment was conducted to demonstrate the effectiveness and advantages of the method.
2. Related Work

2.1. YOLOX. YOLOX is the result of the recent improvements to YOLO series, which incorporates the advantages of YOLO series network, the feature extraction network CSPDarknet (Cross Stage Partial Darknet) architecture of YOLOv4 [11–13], focus channel augmentation technique of YOLOv5, mosaic data augmentation, innovatively added decoupling prediction head, Anchor Free concept, and SimOTA (A label allocation strategy) dynamic positive sample matching.

2.2. Transformer. Transformer, a groundbreaking 4th-generation neural network in recent years, has been dominantly employed in the NLP (Natural Language Processing) field, and also in the CV (Computer Vision) field with dramatic effect. Initially, ViT [14] (Vision Transformer) was employed to train pictures with the text processing method and also in image classification with excellent effect, and BoTNet (Bottleneck Transformers for Visual Recognition) [15–18] was used for replacing the last layer of convolutional neural networks (CNN) with transformer module, to enhance the capability of backbone network for capturing global information.

2.3. Improved YOLOX Algorithm: T-YOLOX. Though YOLOX has shown satisfactory detection performance and inference speed, it still requires improvements in the following aspect for solving the problems of the study:

(1) CSPLayer in YOLOX contains high residual ratio, and residual error operation can effectively avoid vanishing gradient in deep network, but such residual error ratio will transmit feature information together with included noises to deep network, causing impact on training of backbone network

(2) All the residual error operations will connect input features to output features with residual error ratio, but the effect of merely connecting operation for the feature layer is not satisfactory, likely to cause channel information unable to be properly fused

(3) CNN-based CSPDarknet backbone network is used for YOLOX, to capture local feature information via convolution core, but the relationship between global feature information may be neglected

Therefore, considering the drawbacks of YOLOX in detecting complicated fire scenarios, we proposed T-YOLOX model, consisting of 3 parts, i.e., backbone, neck and head, with the architecture shown in Figure 1.

2.4. Light Attention Module. To prevent residual error operation bringing unnecessary noises to the following network, causing impact on the overall training of the network, a light attention module attention was added in the study, in addition to CSPLayer, for applying attention to original residual error edge and then adjusting the weight of each channel, to reduce the impact of noise on network training and facilitate the model to locate and identify the area of interest more accurately.

The principles involve feature fusion and residual error conversion, to enhance channel information and reduce impact from noises. The module chiefly includes 3 branches, i.e., X1, for compressing high dimension features through pooling global average values (AvgPool) and then for $F_{X1}$ compression of features with fully connected layers (FC) and $\delta$ (ReLU) activation function, with the actual operation shown in

\[ F_{X1} = \delta(FC(AvgPool,X1)). \]  

After completion, FC and $\sigma$ (Sigmoid) activation functions are used for expansion to obtain $F_{X2}$, and the eventually extracted attention weight $F_{X2}$ dot product is applied on X2, with the final output shown in

\[ F_{X2} = \sigma(FC(F_{X1})), \]  

\[ X2 = F_{X2} \odot Input. \]

X3 is for stacking of residual error blocks and feature extraction. Finally, X2 and X3 converge through connection ($\oplus$). Figure 2 shows the overall structure of improved CSPLayer-attention.

2.5. Channel Shuffle (CS) Technique. The channel shuffle [19] module is used for channel shuffling the feature layer after connecting operation, to improve communication between channels. The connecting operation of the feature layer is shown in

\[ \left( \begin{array}{ccc} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{array} \right) \odot \left( \begin{array}{ccc} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{m1} & \cdots & b_{mn} \end{array} \right) = \left( \begin{array}{ccc} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{array} \right). \]  

The channel shuffling process is demonstrated with matrix $A_{mn}$, $B_{mn}$, where $a_{ij}$ represents the weight of features, $1 \leq i \leq m$, $1 \leq j \leq n$, $i, j \in Z$; $\odot$ represents feature connecting operation, dividing the feature matrix into several column vectors, e.g., $a_i = [a_{i1} \cdots a_{im}]^T$, $b_j = [b_{1j} \cdots b_{mj}]^T$. Corresponding groups are extracted from different matrices by one for merging operation, to obtain matrix $c_j = [a_{ij}, b_{ij}]$; and then, the connecting operation “$\oplus$” is performed for each $c_j$. Formula (5) shows the matrix after shuffling:

\[ \left( \begin{array}{ccc} a_{11} & b_{11} \\ \vdots & \ddots & \vdots \\ a_{m1} & b_{m1} \end{array} \right) \oplus \cdots \oplus \left( \begin{array}{ccc} a_{1j} & b_{1j} \\ \vdots & \ddots & \vdots \\ a_{mj} & b_{mj} \end{array} \right) \oplus \cdots \oplus \left( \begin{array}{ccc} a_{1n} & b_{1n} \\ \vdots & \ddots & \vdots \\ a_{mn} & b_{mn} \end{array} \right). \]  

Then, matrix $c_j$ is deemed as a group, and “$\oplus$” connecting operation is conducted between groups, with no change to the feature dimension in the entire process.
With the same computing resources, the method can enhance the communication ability between channels, increase the network complexity, and avoid overfitting.

The flow for realizing channel shuffle is shown in Figure 3.

2.6. Light Transformer: MobileViT. The MobileViT module was introduced, and the last layer of backbone network CSPLayer was replaced with MobileViT Block, to increase the sensing ability of the network for global and local information and improve the ability of the model for feature extraction. In MobileViT Block, for given input feature $X \in \mathbb{R}^{H \times W \times C}$ (where $H$, $W$, and $C$ represent the channel, height, and width of the tensor, respectively), $n \times n$ $(n = 3)$ convolution was used for coding local space information, and $1 \times 1$ convolution was used for learning the linear combination of input channel, and the tensor was projected to a high dimension space $(d: \text{dimension}, \text{and } d > C)$, to obtain $X_{1} \in \mathbb{R}^{H \times W \times d}$ after adjustment, and the channel unfolding operation was performed for the feature layer to obtain $X_{\text{Unfold}} \in \mathbb{R}^{P \times N \times d}$, where $P = wh$, $P$ represents the pixel count of the patch with the width of $w$ and height of $h$; $N = HW \div P$; $N$ represents the total number of patches $(h \leq n, w \leq n)$. Then, the information between patches was coded with the transformer module, to obtain $X_{G} \in \mathbb{R}^{P \times N \times d}$, with the actual operation shown in

$$X_{G}(p) = \text{Transformer}(X_{\text{Unfold}}(p)), \quad 1 \leq p \leq P. \quad (6)$$

To avoid losing the location information between patches and the pixel information in each patch, $X_{G} \in \mathbb{R}^{P \times N \times d}$ was refolded and recovered to obtain $X_{\text{Fold}} \in \mathbb{R}^{H' \times W' \times C'}$. $X_{\text{Fold}}$ was transmitted to $1 \times 1$ convolution network, and projected to low dimension space (C dimension) to obtain $\hat{X} \in \mathbb{R}^{H' \times W' \times C'}$. $\hat{X}$ and $X$ were connected (CONCAT) to obtain $\hat{X} \in \mathbb{R}^{H' \times W' \times 2C}$, and finally, $n \times n$ $(n = 3)$ convolution was employed to fuse local feature $\hat{X}$ and global feature $X$ and obtain output $Y \in \mathbb{R}^{H' \times W' \times C}$. Figure 4 shows the structure of MobileViT Block.

---

**Figure 1:** Architecture of T-YOLOX model.

**Figure 2:** Structure of CSPLayer-attention module.

**Figure 3:** Channel shuffle flow.
3. Experiment

3.1. Dataset. The early flame datasets were mainly used for image classification, with low image resolution and poor ability for identification and feedback of information, so such datasets cannot meet the requirements for current detection tasks. In addition, for detection of complicated fire scenarios, it shall not be limited to merely detecting the flame. Targeted at the problems of existing datasets, the fire dataset was developed for the experiment, including the contents on flame, smoke, and persons, and 5,000 pictures related to fire were collected from internet, covering forest fire, industrial fire, urban fire, indoor fire, vehicle fire, etc. Figure 5 shows part of the dataset contents.

The collected data was screened and organized, and the pictures were labeled with LabelImg tool, to create the fire dataset, including 3 categories, i.e., fire, smoke, and person. The labeled pictures were finally stored in an xml file, including the categories and coordinate information of the detection targets. Figure 6 shows the operation of LabelImg scripts.

The experiment is intended for realizing detection of flame, smoke, and persons with the datasets. Figure 7 visualizes the information of the dataset. As shown in visualized data plot 7 (a), nearly 7,000 flames were labeled, and about 2,000 smokes and persons were labeled; and plot 7 (b) shows the uniform distribution of labeled frames, Figure 7(c) shows the location distribution of target frame, and Figure 7(d) shows the percentage target frame relative to the picture. It is apparent that the distribution and percentage of labeled data are uniform and varied.

3.2. Data Augmentation. Data augmentation can effectively expand diversity of samples, to ensure the higher robustness of the model in different environments. In our experiment, in addition to conventional data augmentation methods, e.g., random zooming, cutting, rotation, and other operation, mosaic data augmentation was also employed, for splicing 4 pretreated images into one large image, to enrich the background of the object for detection. Figure 8 shows the operation of mosaic data augmentation.

3.3. Experiment Environment and Program Design. The experiment was conducted in the environment of Python3.8, CUDA11.1, PyTorch 1.9.1, and all the models were trained and tested on NVIDIA RTX3060 GPU.

Before network training, the model was initiated with He et al. [20]. Upon training, Python script was employed to randomly divide the dataset into the training set and test set as per 8:2, and Adam [21] optimizer and cosine annealing learning rate were adopted for training, with 4 training lots, initial learning rate of 0.0001, and totally 300 training iterations. Totally, 640, 640, 3 pictures were input in the experiment. After the training, the performance of the proposed model was compared with that of CenterNet, YOLOv3, and YOLOX models.
3.4. Model Evaluation. For the target detection task, the precision and recall can be calculated for each category, and a PR curve can be plotted for each category based on reasonable calculation. Each labeled picture contains a detection targets and \( b \) detection categories; through detection of pictures, \( c \) Bounding Boxes (BB) is obtained, and each BB contains the coordinate information and category information of the target, with which the IOU value of the ground truth is calculated.

In our study, the commonly used evaluation indexes for target detection models mAP (mean average precision) and FPS (frames per second) were used for model evaluation. AP is the area under PR (precision-recall) curve, and mAP is the mean average value of AP of each category (the higher value of AP and mAP, the better), calculated with the following:

\[
\text{Precision} = \frac{TP}{TP + FP},
\]

(7)

\[
\text{Recall} = \frac{TP}{TP + FN}.
\]

(8)

TP (true positives) refers to positive samples, correctly segmented.

FP (false positives) refers to positive samples, incorrectly segmented.

FN (false negatives) refers to negative samples, incorrectly segmented.

3.5. Experiment Design and Result Analysis. For our study, we totally designed 1 group of ablation experiments and 1 group of comparative experiments and to meet the
requirements for model deployment, size—a light model was chosen, and the ablation experiment was conducted to analyze the effect of different improvements on network performance under YOLOX architecture, and then, the comparative experiment was performed for T-YOLOX and mainstream networks (CenterNet, YOLOv3) to comprehensively analyze the model performance.

All the comparisons were made with the self-developed fire dataset, the dataset was divided into the training set and the test set as per 8:2, and 10% was sampled from the training set as the verification set. The experiment results were verified with the above two methods. The loss of the proposed T-YOLOX model after training for 300 iterations is shown in Figure 9. As seen in the figure, the loss decreases quickly at the early stage of training, and with the increases of training rounds, the loss curve gradually falls down and becomes flat in the end. When Epoch reaches around 200, the model gradually converges, and no overfitting was discovered in the training process.

### 3.5.1. Ablation Experiment

To analyze the effect of the proposed improvements on the model performance, 3 groups of experiments were designed for analysis and comparison of the improvements, and each group of experiments was conducted with the same training parameters and different model contents. Table 1 shows the detection results of the model performance, where “✓” represents the strategy used in the improved model and “×” represents the strategy not used in the improved model. According to the analysis on the results in Table 1, for Improvement 1, after connecting operation (CONCAT), channel shuffle module is used, to increase the communication ability between channels and avoid overfitting, increasing mAP to some extent; for Improvement 2, in addition to the above, light attention module is added, and an attention enhancing edge is added at CSPLayer, to improve the attention of the channel to the space information and reduce the impact of noise on deep network, increasing mAP by 1.05%; and for Improvement 3, MobileViT module is added, and CNN and transformer are fused, to realize the learning ability of backbone network for local and global information, increasing mAP by 1.02%.

### 3.5.2. Model Comparison

To verify the detection performance of improved T-YOLOX model, comparative experiment was conducted to the model and mainstream target detection models, e.g., YOLOv3, CenterNet, and YOLOX, adopting mAP and FPS for evaluating each algorithm, with the results shown in Table 2. As seen in the table, mAP of T-YOLOX algorithm was 69.54%, increased by 2.24% as compared with the original YOLOX algorithm, and according to the analysis on the average AP values for fire, smoke, and person in the table, the proposed method showed the

### Table 1: Experiment results with different improvement methods.

| Method        | SC | Attention | MobileViT | mAP   | FPS   |
|---------------|----|-----------|-----------|-------|-------|
| YOLOX         | ×  | ×         | ×         | 67.28%| 63.69 |
| Improvement 1 | ✓  | ×         | ✓         | 67.47%| 63.09 |
| Improvement 2 | ✓  | ✓         | ✓         | 68.52%| 58.35 |
| Improvement 3 | ✓  | ✓         | ✓         | 69.54%| 54.17 |

### Table 2: Comparison of performance between mainstream target detection models.

| Model   | Fire (IoU = 0.5) | Smoke | Person | mAP   | FPS   |
|---------|------------------|-------|--------|-------|-------|
| CenterNet | 0.59             | 0.43  | 0.17   | 39.80 | 60.29 |
| YOLOv3   | 0.74             | 0.58  | 0.54   | 62.05 | 48.60 |
| YOLOX    | 0.72             | 0.61  | 0.69   | 67.28 | 63.69 |
| T-YOLOX  | 0.75             | 0.62  | 0.72   | 69.54 | 54.17 |
improved AP values in detecting flame, smoke, and persons as compared with that of original YOLOX algorithms to different extent, with detection performance better than other mainstream target detection models (CenterNet, YOLOv3). For detection of victims, T-YOLOX showed the significant advantages, with FPS not falling significantly, and the detection speed still higher than that of the mainstream models, while maintaining the high precision detection.

4. Conclusions

Considering the problems of existing target detection models difficult to give timely and effective feedback on fire detection in complicated fire scenarios, a fire scenario detection model T-YOLOX for improving YOLOX was proposed in the study. On the basis of the YOLOX model, the method is added with the channel shuffle module with enhanced channel communication ability and CSPLayer_attention module for channel attention weight, and MobileViT module integrating CNN and transformer, to complete target detection for persons, flame, and smoke in complicated fire scenarios, with the detection results shown in Figure 10. As seen in the experiment, the proposed detection method demonstrated the good performance in detection in complicated fire scenarios. In the future work, consideration shall be made on how to further improve the detection accuracy of smoke in complicated fire scenarios.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research was funded by the National Natural Science Foundation of China (grant number 61803148).

References

[1] L. C. Yu and J. Q. Liu, "Flame image recognition algorithm based on improved mask R-CNN," Computer Engineering and Applications, vol. 56, no. 21, pp. 194–198, 2020.
[2] F. Wu, Research and Implementation of Fire Detection Algorithm Based on Deep Learning, Hangzhou Dianzi University, 2020.
[3] J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," 2018, arXiv: 1804.02767.
[4] M. Zhao, Y. L. Ge, and N. Ding, "Object detection in complex environment based on CenterNet algorithm," Journal of China Academy of Electronics Science, vol. 16, no. 7, pp. 654–660, 2021.
[5] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Q. Tian, Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6569–6578, Seoul Korea, 2019.
[6] X. J. Li, D. S. Zhang, and L. L. Sun, "CNN-based lightweight flame detection method in complex scenes," Pattern Recognition and Artificial Intelligence, vol. 34, no. 5, pp. 415–422, 2021.
[7] Z. H. Luo, Research on Forest Fire Monitoring and Path Planning Based on UAV, Xi'an University of Technology, 2021.
[8] Z. Ge, S. Liu, and F. Wang, "YOLOX: exceeding YOLO series in 2021," 2021, arXiv: 2107.08430.
[9] S. Mehta and M. Rastegari, "MobileViT: light-weight, general-purpose, and mobile-friendly vision transformer," 2021, arXiv: 2110.02178.
[10] A. Vaswani, N. Shazeer, and N. Parmar, “Attention is all you need,” 2017, arXiv: 1706.03762.
[11] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: optimal speed and accuracy of object detection," 2020, arXiv:2004.10934.
[12] Z. Ly, L. Qiao, M. S. Hossain, and B. J. Choi, "Analysis of using blockchain to protect the privacy of drone big data," IEEE Network, vol. 35, no. 1, pp. 44–49, 2021.
[13] T. Wang, W. Liu, J. Zhao, X. Guo, and V. Terzija, "A rough set-based bio-inspired fault diagnosis method for electrical substations," International Journal of Electrical Power & Energy Systems, vol. 119, article 105961, 2020.
[14] A. Dosovitskiy, L. Beyer, and A. Kolesnikov, “An image is worth 16x16 words: transformers for image recognition at scale,” https://arxiv.org/abs/2010.11929.
[15] A. Srinivas, T. Y. Lin, and N. Parmar, "Bottleneck transformers for visual recognition," 2021, arXiv: 2101.11605.
[16] Z. Lv, L. Qiao, and I. You, “6G-enabled network in box for internet of connected vehicles,” IEEE transactions on intelligent transportation systems, vol. 22, no. 8, pp. 5275–5282, 2021.

[17] B. Li, G. Xiao, R. Lu, R. Deng, and H. Bao, ”On feasibility and limitations of detecting false data injection attacks on power grid state estimation using D-FACTS devices,” IEEE Transactions on Industrial Informatics, vol. 16, no. 2, pp. 854–864, 2020.

[18] Z. Lv and W. Xiu, ”Interaction of edge-cloud computing based on SDN and NFV for next generation IoT,” IEEE Internet of Things Journal, vol. 7, no. 7, pp. 5706–5712, 2020.

[19] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, ”ShuffleNet V2: practical guidelines for efficient CNN architecture design,” in Computer Vision – ECCV 2018, pp. 116–131, Springer, Cham, 2018.

[20] K. He, X. Zhang, and S. Ren, ”Delving deep into rectifiers: surpassing human-level performance on ImageNet classification,” 2015, arXiv: 1502.01852.

[21] D. P. Kingma and J. Ba, ”Adam: a method for stochastic optimization,” 2017, arXiv:1412.6980.