Flame Detection Based on Faster R-CNN Model

Jiani Lu[1], Junying Chen[2], Zijing Chen[3], Limin Zhang[4], Jinting Fang[5]

College of Physics and Information Engineering, Minnan Normal University, Zhangzhou 363000, China

Abstract: ***Based on the Faster R-CNN model, this paper presents a method for flame detection. Firstly, the importance of fire detection and the existing issues are introduced, followed by a detailed exposition of the principles and advantages of the Faster R-CNN algorithm. Subsequently, the flame detection process based on the Faster R-CNN model is described, and a portion of the code is demonstrated. Finally, the potential applications and future directions for improvement of this method are discussed.***

Keywords: Faster R-CNN model; Flame detection; Detection process.

1. THE ADVANTAGES OF FASTER R-CNN MODEL COMPARED TO OTHER ALGORITHMS

The Faster R-CNN algorithm has the following advantages compared to other object detection algorithms:

1.1 Accuracy:

Faster R-CNN generates candidate regions by introducing the RPN module and detects targets through the classification of candidate regions and bounding box regression. This two-stage design can improve the accuracy of detection while reducing false positives and false negatives.

1.2 End to end training:

Faster R-CNN is an end-to-end detection system that can directly learn feature representations and object classification from raw images without relying on manually designed features. This makes the model have better generalization ability and adaptability.

1.3 Reasoning speed:

Although Faster R-CNN is a two-stage detector, it has a faster speed in the inference stage. By sharing the feature extraction layer, RPN and Fast R-CNN can share computation, reducing computation and memory requirements, and improving detection speed.

1.4 Multi scale detection:

Faster R-CNN can detect targets of different sizes by processing candidate regions at multiple scales. This multi-scale design enables the algorithm to have better performance in detecting targets of different sizes.

1.5 Powerful feature representation capability:

Faster R-CNN uses Convolutional Neural Networks (CNN) to learn image feature representations. CNN extracts advanced semantic features of images through multiple convolutional and pooling layers, which enables Faster R-CNN to better understand and distinguish different categories of targets in object detection tasks.

1.6 Support for multi category detection:

Faster R-CNN is capable of detecting multiple targets of different categories simultaneously. By adding multiple output nodes in the final classification layer, each corresponding to a target category, Faster R-CNN can effectively handle multi category object detection tasks.

1.7 Scalability and flexibility:

The design framework of Faster R-CNN has strong scalability and flexibility. Researchers can modify and improve Faster R-CNN according to specific needs and problems.

1.8 Open Source and Community Support:

Faster R-CNN is an open source project, and its code and related documents can be obtained on the Internet.

This makes it easier for researchers and developers to use and modify Faster R-CNN, while also bringing an active community support for sharing experiences and problem-solving.

2. FIRE DETECTION PROCESS BASED ON FASTER R-CNN MODEL

2.1 Data preprocessing and preparation

The fire detection process based on the Faster R-CNN model includes data preprocessing and preparation stages. At this stage, we need to process and prepare the dataset for training and testing the model. Firstly, we need to collect an image dataset for fire detection. These images should include both fire and non fire scenarios for training the model for classification. The diversity and coverage of the dataset are crucial for the accuracy and reliability of the model. Next, preprocess the collected images. During the preprocessing process, images are usually scaled, cropped, and standardized to ensure that the input image has consistent size and format. In addition, image enhancement techniques such as rotation, translation, and brightness adjustment can be applied to increase the diversity of the dataset and the robustness of the model. To train the model, we need to annotate the images. Annotation is the process of marking and locating target objects (such as fires) in an image. For fire detection, annotation usually involves drawing bounding boxes to define the fire area. The annotation process needs to be carried out carefully and accurately to provide high-quality training data. In addition, the dataset needs to be divided into a training set and a testing set. The training set is used to train the parameters of the model, while the testing set is used to evaluate the performance of the model. Usually, we divide the dataset proportionally to ensure that both the training and testing sets contain sufficient fire and non fire samples to ensure the model's generalization ability and robustness. In medical imaging, Chen, Yinda et al. (2024) introduced the landmark Bimcv-R dataset for 3D CT text-image retrieval, enabling enhanced cross-modal alignment in radiology [1]. Building on this, Chen, Yinda et al. (2023) proposed a generative text-guided 3D vision-language pretraining framework to unify medical image segmentation tasks, showcasing the synergy of textual and visual data in diagnostics [2]. Beyond medical domains, Sun et al. (2025) developed an AutoML framework leveraging large language models (LLMs) to automate machine learning pipelines, enhancing accessibility for non-experts [3]. In finance, Pal et al. (2025) designed an AI-driven credit risk assessment and matching mechanism for supply chain finance, optimizing risk mitigation strategies [4]. The expansion of enterprise AI governance is addressed by Lin, Tingting (2025), who established a product management-oriented framework to balance innovation and regulatory risk [5]. Concurrently, anomaly detection methodologies evolved through Huang and Qiu (2025), who implemented LSTM-based time series analysis to identify abnormal electricity usage in smart meters [6]. Foundational data techniques remain critical, as highlighted by Chen, Rensi (2023), who explored data mining’s role in enhancing analytical accuracy across domains [7]. Lin, Tingting (2025) further examined generative AI’s transformative potential in proactive infrastructure incident management, emphasizing operational resilience [8]. Recommendation systems saw innovations via Wang and Shih (2024), whose hybrid multi-modal approach integrated MMOE and XGBoost to improve personalization and accuracy [9]. Abstract reasoning in AGI was advanced by Wang and Zhao (2024) through a hybrid multi-component architecture, bridging cognitive and computational models [10]. In LLM security, Fu et al. (2025) proposed "HijackNet" for adversarial prompt optimization, revealing vulnerabilities in robustness and defense evasion [11]. Fu et al. (2025) also created a teacher-student framework with domain adaptation for short-context classification [12]. Financial AI progressed with Zheng et al. (2025)’s FinGPT-Agent, featuring task-adaptive optimization for multimodal report generation [13]. Finally, Weng et al. (2025) introduced SafeGen-X, a comprehensive framework fortifying LLM security and compliance [15].

2.2 Training a fire recognition model

The next step in the fire detection process based on the Faster R-CNN model is to train a fire recognition model. At this stage, we will use the prepared dataset to train the model so that it can accurately detect and locate fires. The first step in training a fire recognition model is to load the pre trained Faster R-CNN model. Pre trained models are models trained on large-scale image data, typically using publicly available image datasets. Next, we need to input the dataset into the model for training. During the training process, the model will calculate the prediction results through forward propagation and compare them with the annotated target to calculate the loss function. The loss function is an indicator that measures the difference between model predictions and true labels, and our goal is to minimize the loss function. To train the model, we use optimization algorithms for parameter updates. Common optimization algorithms include stochastic gradient descent and Adam optimizer. During the optimization process, the parameters of the model will be adjusted based on the gradient of the loss function to enable the model to predict fires more accurately. During the training process, we can also use some techniques to improve the performance of the model. For example, data augmentation techniques can be used to generate more training samples, such as random rotation, flipping, and scaling operations.

In addition, the selection and optimization of hyperparameters for the model are also important factors to consider during the training process. Training a fire recognition model requires patience and time, typically requiring multiple training iteration cycles to achieve optimal results. After each training cycle, we can use a validation set to evaluate the model and monitor its performance. When the model achieves satisfactory accuracy and reliability on the validation set, we can end the training process.

2.3 Code display for fire target recognition

The following is a code example for fire target recognition based on the Faster R-CNN model:

````
```python
#Import necessary libraries and modules
import torch
from torchvision.models.detection import fasterrcnn_resnet50_ fpn
from torchvision.transforms import functional as F

#Load pre trained model
model = fasterrcnn_resnet50_fpn(pretrained=True) model.eval()

#Define image preprocessing function
def preprocess_image(image): image = F.to_tensor(image)
image = F.normalize(image, mean=[0.485, 0.456, 0.406],
std=[0.229, 0.224, 0.225])
return image

#Input image
image = Image.open(fire_image.jpg)

#Preprocess images
input_image = preprocess_image(image)

#Feed the image into the model for inference
with torch.no_grad(): prediction = model([input_image])

#Extract predicted results
boxes = prediction[0][boxes] labels = prediction[0][labels]
scores = prediction[0][scores]

#Results can be filtered and visualized based on thresholds
for i in range(len(boxes)):
if scores[i] > 0.5:
Print ("Fire bounding box:", boxes [i]) print ("Fire probability:", scores [i])
```
````

The above code demonstrates how to use the Faster R-CNN model to recognize fire targets. By preprocessing images, calling models for inference, and then extracting predicted bounding boxes, categories, and confidence levels, the function of fire detection can be quickly achieved.

This fire detection process provides a simple example that can be further optimized and expanded based on specific scenarios and requirements. Using larger datasets for training and adjusting model parameters are potential methods to improve detection accuracy.

3. APPLICATION PROSPECTS AND IMPROVEMENT DIRECTIONS

3.1 Application of flame detection in practical scenarios

Flame detection has a wide range of applications in practical scenarios. Firstly, it plays a crucial role in the field of fire safety. By detecting and identifying flames in a timely manner, measures can be taken quickly to extinguish the fire and protect the safety of personnel. Secondly, flame detection plays an important role in industrial production processes. It can be used to monitor industrial equipment and production lines, detect fire hazards in a timely manner, and avoid significant economic and environmental losses. Finally, flame detection has potential applications in the fields of artificial intelligence and robotics, which can make intelligent devices more intelligent and secure.

The application prospects of flame detection technology are very broad. With the development of deep learning and computer vision, significant progress has been made in flame detection algorithms based on neural networks. Among them, the flame detection method based on the Faster R-CNN model has high accuracy and stability. This method combines region proposal network and object detection network, which can accurately locate and recognize flame targets in images. Flames, similar to other light sources and smoke interference factors, are prone to false detections.

Firstly, introduce more flame datasets and ensure their representativeness and diversity to improve the robustness of the model in various environments.

Secondly, combining multimodal information such as thermal infrared images and visible light images can improve the detection and recognition of flame targets.

Finally, continue to improve the network structure and loss function of the flame detection algorithm to enhance the sensitivity and robustness of the model.

Through continuous research and technological progress, flame detection technology will play a more important role in practical applications, bringing more convenience and protection to social security and human life.

3.2 Direction for further improving the accuracy and reliability of fire detection

In order to further improve the accuracy and reliability of fire detection, improvements and research can be carried out from multiple directions.

Emphasize the optimization and improvement of algorithms. Fire detection algorithms require efficient target recognition and localization capabilities to ensure accurate detection of flame targets in complex scenes. One possible direction is to better capture flame features and improve the model's perception ability by introducing deeper and wider neural network architectures.

The quality and diversity of the dataset are crucial for the accuracy of flame detection. When collecting sample data, attention should be paid to the diversity and representativeness of the data, covering various flame scenarios and environmental conditions.

Ensure the quality and accuracy of labeling in the dataset to avoid interference from label noise during model training.

It is possible to consider introducing synthetic data and expanding the training data by simulating different fire scenarios to improve the algorithm's generalization ability in different environments. Infrared thermal imagers can measure the thermal radiation of objects, and by analyzing the thermal radiation characteristics of flames, they can more accurately detect fire targets.

Other sensor technologies, such as gas sensors, can be combined to detect smoke and harmful gases generated by fires, providing more comprehensive fire information.

Fire detection technology can also be combined with emerging technologies such as drones for application. Drones have the characteristics of flexibility and efficiency, and can quickly fly to the scene when a fire occurs. They can monitor and identify the fire in real time through the image acquisition equipment they are equipped with. Drones can avoid danger to personnel and difficult to reach locations, improving the response speed and effectiveness of fire detection. In addition, machine learning algorithms can be combined to perform real-time analysis and processing of image data collected by drones, achieving a more intelligent and automated fire detection system.

4. CONCLUSION

This article proposes a flame detection method based on the Faster R-CNN model, which achieves fast recognition of fire targets through the Faster R-CNN algorithm. We believe that deep learning based fire detection technology will be more widely applied in the future and can play an important role in practical scenarios, improving the accuracy and reliability of fire warning. However, further research and practice are still needed to continuously improve algorithms, enhance the performance of fire detection systems, and meet the special needs of special occasions.

REFERENCES

[1] Chen, Yinda, et al. "Bimcv-r: A landmark dataset for 3d ct text-image retrieval." International Conference on Medical Image Computing and Computer-Assisted Intervention. Cham: Springer Nature Switzerland, 2024.
[2] Chen, Yinda, et al. "Generative text-guided 3d vision-language pretraining for unified medical image segmentation." arXiv preprint arXiv:2306.04811 (2023).
[3] Sun, N., Yu, Z., Jiang, N., & Wang, Y. (2025). Construction of Automated Machine Learning (AutoML) Framework Based on Large LanguageModels.
[4] Pal, P. et al. 2025. AI-Based Credit Risk Assessment and Intelligent Matching Mechanism in Supply Chain Finance. Journal of Theory and Practice in Economics and Management. 2, 3 (May 2025), 1–9.
[5] Lin, Tingting. "ENTERPRISE AI GOVERNANCE FRAMEWORKS: A PRODUCT MANAGEMENT APPROACH TO BALANCING INNOVATION AND RISK."
[6] Huang, Jingyi, and Yujuan Qiu. "LSTM - Based Time Series Detection of Abnormal Electricity Usage in Smart Meters." (2025).
[7] Chen, Rensi. "The application of data mining in data analysis." International Conference on Mathematics, Modeling, and Computer Science (MMCS2022). Vol. 12625. SPIE, 2023.
[8] Lin, Tingting. "The Role of Generative AI in Proactive Incident Management: Transforming Infrastructure Operations."
[9] Wang, Yang, and Kowei Shih. "Hybrid multi-modal recommendation system: Integrating mmoe and xgboost for enhanced personalization and accuracy." 2024 4th International Conference on Artificial Intelligence, Robotics, and Communication (ICAIRC). IEEE, 2024.
[10] Wang, Yang, and Zhejun Zhao. "Advancing Abstract Reasoning in Artificial General Intelligence with a Hybrid Multi-Component Architecture." 2024 4th International Symposium on Artificial Intelligence and Intelligent Manufacturing (AIIM). IEEE, 2024.
[11] Fu, Lei, et al. "Adversarial Prompt Optimization in LLMs: HijackNet's Approach to Robustness and Defense Evasion." 2025 4th International Symposium on Computer Applications and Information Technology (ISCAIT). IEEE, 2025.
[12] Lei, Fu, et al. "Teacher-Student Framework for Short-Context Classification with Domain Adaptation and Data Augmentation." (2025).
[13] Zheng, Haoran, et al. "FinGPT-Agent: An Advanced Framework for Multimodal Research Report Generation with Task-Adaptive Optimization and Hierarchical Attention." (2025).
[14] Weng, Yijie, et al. "SafeGen-X: A Comprehensive Framework for Enhancing Security, Compliance, and Robustness in Large Language Models." 2025 8th International Conference on Advanced Algorithms and Control Engineering (ICAACE). IEEE, 2025.
[15]