Video Flame Detection Based on Multi-feature Fusion and Double-layer XGBoost

Yuanbin Wang, Yujie Li, Huaying Wu and Yu Duan

***Abstract*—One of the most common and dangerous natural disasters in human production and life is fire. Thus, the application of efficient fire warning technologies to complicated settings is crucial for study. To increase the accuracy and robustness of flame recognition, according to the findings of this research, an intelligent video flame detection system according to multi-feature fusion and double-layer XGBoost might be developed. Firstly, for color segmentation, a new color feature model based on YCbCr and HSV color spaces is created. Then, to filter away static interference objects of a similar hue to the flame, the ViBe algorithm with a new background update method is used, and the candidate is achieved. After that, the skeleton shape feature, Gabor texture feature, growth rate, and centroid change rate are adopted to train the independent XGBoost classifiers, respectively. Finally, the second layer XGBoost algorithm integrates each independent XGBoost classifier in the first layer and outputs flame classification results. The simulated annealing algorithm optimizes the parameters of the double-layer XGBoost model. The suggested technique enhances the detection rate and may be used in a variety of settings, according to the findings of experiments.**

***Index Terms*—shape context analysis, skeleton image, ViBe, XGBoost.**

I. Introduction

FIRE has brought great life danger and property loss to human society [1]. The traditional fire detection system based on the temperature sensor and smoke sensor [2] has a low recognition rate, especially in the case of complex environments. With the fast advancement of computer vision, video flame detection has become more popular.

The flame detection algorithm is mainly divided into traditional and deep learning algorithms. The traditional algorithm consists of three parts: extraction of suspected area, extraction of feature, and classification. Regarding extraction of suspected area, Qiu [3] proposed an algorithm based on edge detection to segment flame targets, but it is only applicable to the simple background and cannot segment correctly in complex environments. Chen [4] used the mixed Gaussian model to deal with RGB and HSV color information, obtained the probability density distribution function, and detected moving pixels using the inter-frame difference method. However, the robustness of this method is poor. The moving objects with a hue close to flame, as well as things lighted by flame, are easily detected mistakenly. Su [5] formulated different background modeling strategies for the suspected and non-suspected areas to extract the flame foreground under the intense light. In terms of feature extraction, Wu [6] put forward an algorithm based on the Support Vector Machine (SVM). Characteristics such as circularity, rectangularity, and gravity center height coefficient were analyzed. The false alarm rate was reduced. Mei [7] obtained two new early flame features to describe the overlapping rate and the ratio of motion intensity. Zhang [8] proposed to fuse the edge similarity features and the improved texture of the local binary mode for detection. Qi [9] adopted Lucas-Kanade sparse optical flow algorithm to track the movement area and obtain the main motion direction of the flame as the flame identification feature. Regarding the classification, Zeng [10] suggested an analytic hierarchy process-based technique for integrating motion feature detection, color feature analysis, and flame dynamic feature fusion, which lowered the missed alarm rate. Liao [11] used the Adaboost method in conjunction with the Support Vector Machine (SVM), which improved the classification performance when the distribution of positive and negative flame samples was unbalanced. Zhu [12] constructed a deep forest model. For the flame candidate region extracted by Gaussian mixture background modeling method based on frame rate enhancement, the features were first extracted using dual-perspective and deep multi-granularity scanning structure, and then the deep forest model was established for flame detection. In the field of deep learning, Tan [13] proposed an improved Probabilistic Neural Network (PNN) in deep learning, achieving good robustness. Yang [14] improved the accuracy of the Convolutional Neural Network (CNN) by extracting motion information of flames. Compared with traditional methods, the deep learning algorithm can compensate for information loss caused by manual feature extraction. However, the deep learning flame detection technique takes a vast amount of representative samples to train, and the deep learning model's complex forms make the process time-consuming [15].

This work offers a flame identification technique based on multi-feature fusion and the double-layer XGBoost model to increase the accuracy of flame recognition in difficult scenarios. Its advantages are as follows. (1) The color feature model constructed by YCbCr and HSV color space makes up for the vacancy of high-temperature flame segmentation. (2) The improved ViBe algorithm with a new background update strategy can completely extract the

Manuscript received November 11, 2021; revised May 1, 2022. This work was supported by the National Natural Science Foundation of China and the National Key Research and Development Program of Shaanxi Province, China(2019K-046).

Yuanbin Wang is an associate professor of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 13379232752@163.com).

Yujie Li is a postgraduate student of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 1678931975@qq.com).

Huaying Wu is a postgraduate student of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 1661349504@qq.com).

Yu Duan is a postgraduate student of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 307073082@qq.com).

flame region, conducive to further feature extraction. (3) In feature extraction, the skeleton shape feature, Gabor texture feature, growth rate, and centroid change rate are fused to make the algorithm have good robustness even in a complex environment. (4) The double-layer XGBoost model, which uses the simulated annealing algorithm for parameter optimization, takes advantage of the excellent performance and fast calculation of the XGBoost algorithm. Fig. 1 depicts the suggested algorithm's structural structure.

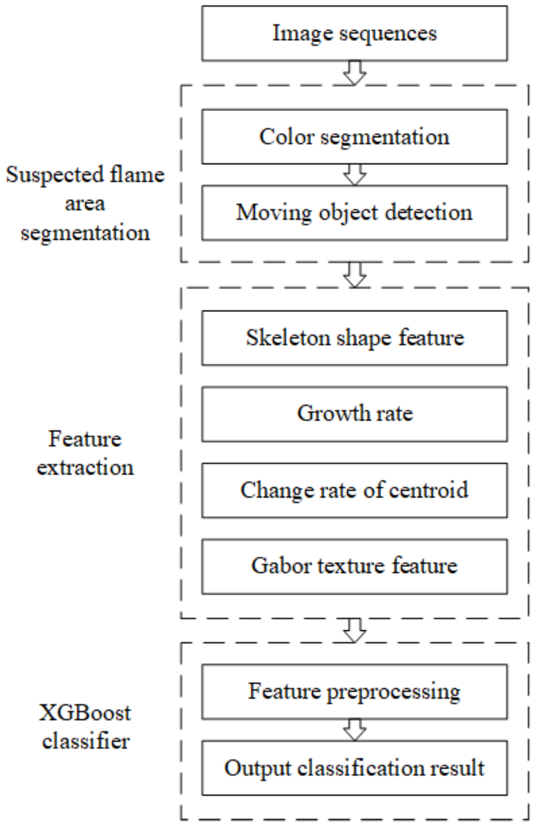


Fig. 1. Structural diagram of the proposed algorithm

II. Suspected Flame Area Segmentation

A. Color Segmentation

At the early stage of the flame, it presents red and yellow. As the temperature rises, the flame gradually turns white. According to the YCbCr color space, Chen [16] summarized the characteristics of flame pixels at lower temperatures. Color and brightness information may be detected using the HSV color model. Hence, it is used to detect the white flame at high temperatures. With regard to the white color, the H component has a value range of 0-180, the S component has a value range of 0-30, and the V component has a value range of 221-255, as shown in the HSV table. Thus, by combining the decision criteria in two color spaces, a new flame color model according to YCbCr and HSV color space is created. The improved color criterion is shown in formula (1). If rule 1-8 or 9 is met, it is taken as a flame. Fig. 2 depicts the color segmentation algorithm's flow chart.

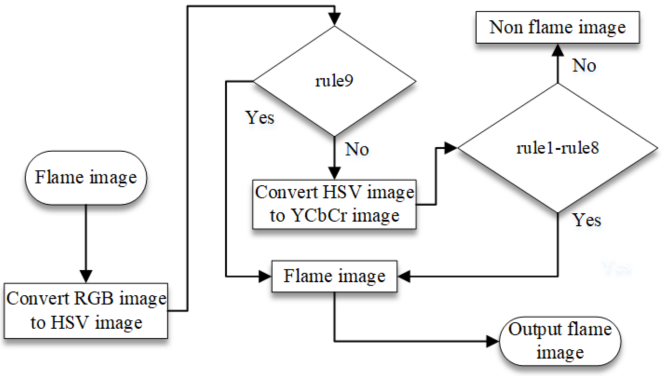
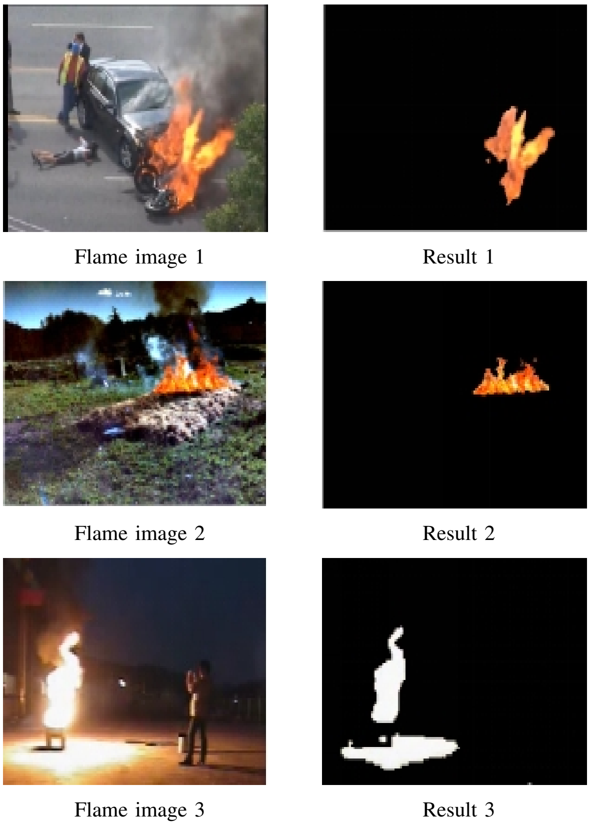


Fig. 2. Flow chart of the color segmentation algorithm

$$\begin{cases} rule1 : |Cb(i,j) - Cr(i,j)| \geq 150 \\ rule2 : Y(i,j) > Ymean \\ rule3 : Cb(i,j) < Cbmean \\ rule4 : Cr(i,j) > Crmean \\ rule5 : Y(i,j) > Cb(i,j) \\ rule6 : Cr(i,j) > Cb(i,j) \\ rule7 : Cr(i,j) > 150 \\ rule8 : Cb(i,j) < 100 \\ rule9 : S(i,j) < 30, V(i,j) > 221 \end{cases} \tag{1}$$

Where, $Ymean$, $Cbmean$, and $Crmean$ are the pixel values on the respective channels on average.

The improved color segmentation algorithm is applied to the flame images in three different scenes, and Fig. 3 depicts the experimental outcomes.


Flame image 1

Result 1

Flame image 2

Result 2

Flame image 3

Result 3

Fig. 3. Result of color segmentation

In Fig. 3, the improved method can effectively remove the interference of withered grass and smoke. The flame can be completely segmented. Furthermore, it is also suitable for the segmentation of white flame.

B. Moving Object Detection

In order to eliminate the interference of static objects with a similar color to the flame, ViBe is carried out after color segmentation. ViBe algorithm [17] is an efficient pixel-level background modeling algorithm. Model initialization, foreground detection, and background model update are the three main components. The schematic diagram of ViBe is shown in Fig. 4.

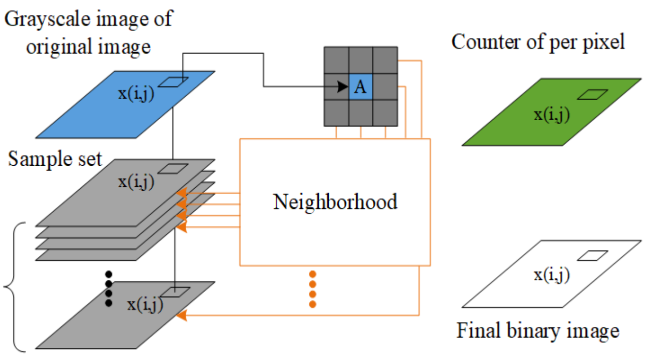
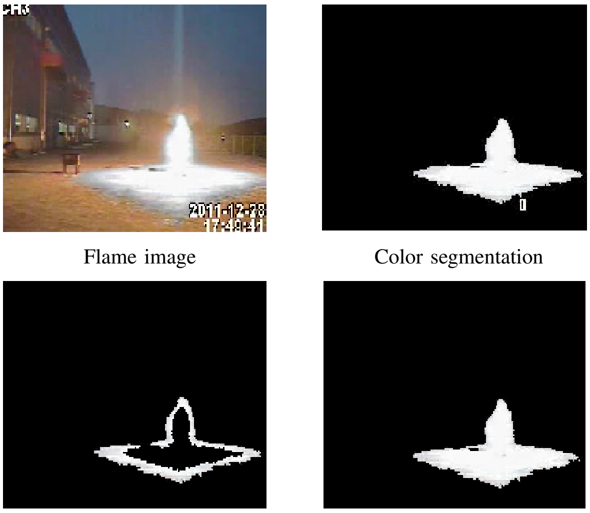


Fig. 4. Schematic diagram of the ViBe algorithm

In the traditional ViBe algorithm, when a pixel is detected as the foreground for consecutive times, it will be updated as the background point so that the moving object will be misjudged as the background. The improved ViBe algorithm proposes to carry out color segmentation first and then accomplish foreground extraction. If the current pixel point is judged as the foreground point many consecutive times but has been segmented as the flame before, it will not be updated to the background. The moving object will not be misjudged as background by adding this judgment condition. In addition, in the stage of background model update, the improved ViBe algorithm selects the sample with the most remarkable difference from the current frame for updating to improve background updating efficiency. The experimental result is shown in Fig. 5.


Flame image

Color segmentation

Traditional Vibe

Improved Vibe

Fig. 5. Result of moving object detection

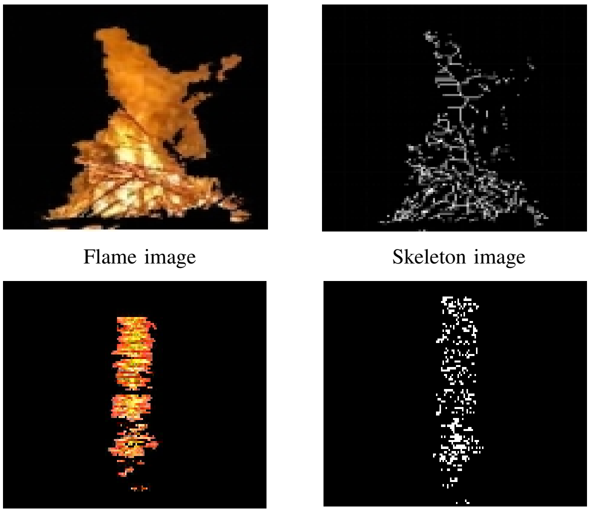
It can be seen from Fig. 5 that the moving object extracted by the traditional ViBe algorithm has a severe hollow phenomenon. In contrast, the improved ViBe algorithm can completely extract the flame. In addition, the time subscript in the original image, whose color is similar to the flame, is filtered out through moving object detection.

III. Flame Feature Extraction

Color segmentation and moving object detection can eliminate the interference of most objects. However, it is still necessary to further extract the static and dynamic characteristics of flame to improve detection accuracy. It is found that the shape and texture of the flame are different from other objects, and the flame has the characteristics of spreading and a stable centroid. This paper extracts skeleton shape feature described by shape context and Gabor texture feature as static features. Skeleton images' growth rate and centroid change rate are extracted as dynamic features. Static features and dynamic features are combined to achieve flame detection.

A. Skeleton shape feature described by shape context

The skeleton image is a simplified representation of the target shape. The skeleton of the image is generally located in the symmetric center of the image contour, similar to the human skeleton, supporting the entire contour. Skeleton images extracted from flame and non-flame objects are shown in Fig. 6. As shown in the figure, the skeleton image of the flame is different from another image. Therefore, we use the shape context algorithm to describe this difference further.


Flame image

Skeleton image

Reflection of the sunset

Skeleton image

Fig. 6. Skeleton image of flame and non-flame object

The shape context algorithm [18] uses a set consisting of a finite number of coordinate points to describe the shape. A point is selected arbitrarily from the set as the coordinate origin to establish a log-polar coordinate system. The coordinate system is divided into N regions, and the number of coordinate points in each area is calculated.

Then this point can be described by a $1 \times N$ vector. The shape of a coordinate point can be expressed as:

$$h_i(k) = \#\{c_j \neq c_i \& c_j \in bin(k)\}, i \neq j \tag{2}$$

Where, $\#$ is the statistical operator of the number for points falling in the kth region; bin(k) represents the kth region, $\mathrm{c_i}$ and $\mathrm{c_j}$ are pixels. Then a shape containing M coordinate points can be represented by an $\mathrm{M} \times \mathrm{N}$ matrix.

B. Growth Rate Based on Skeleton Image

The flame movement has the characteristics of diffusion. The traditional method is to calculate the area change rate of the original flame image. It is found that the number of pixels in the skeleton image changes correspondingly with the flame diffusion and growth. Fig. 7 shows the line chart of the number of pixels in flame skeleton images and candle skeleton images in videos.

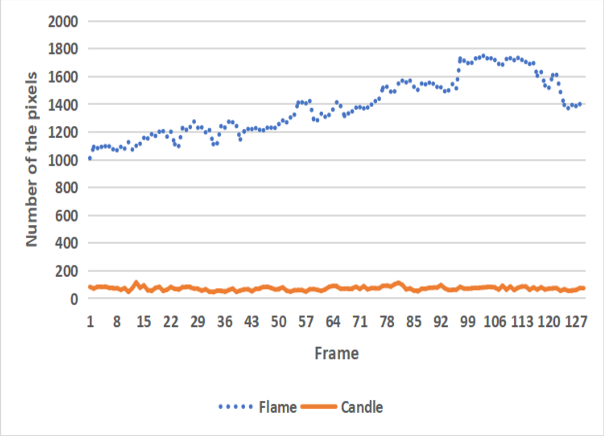


Fig. 7. Line chart of the number of pixels in skeleton images in videos

As seen in Fig. 7, the number of pixels in the flame skeleton images changes dramatically, and the standard deviation is 206.9. The number of pixels in the candle skeleton images is more concentrated than in the flame images, and the standard deviation is 13.32. Therefore, the growth rate can be obtained as a feature by formula (3) for moving objects in continuous video frames.

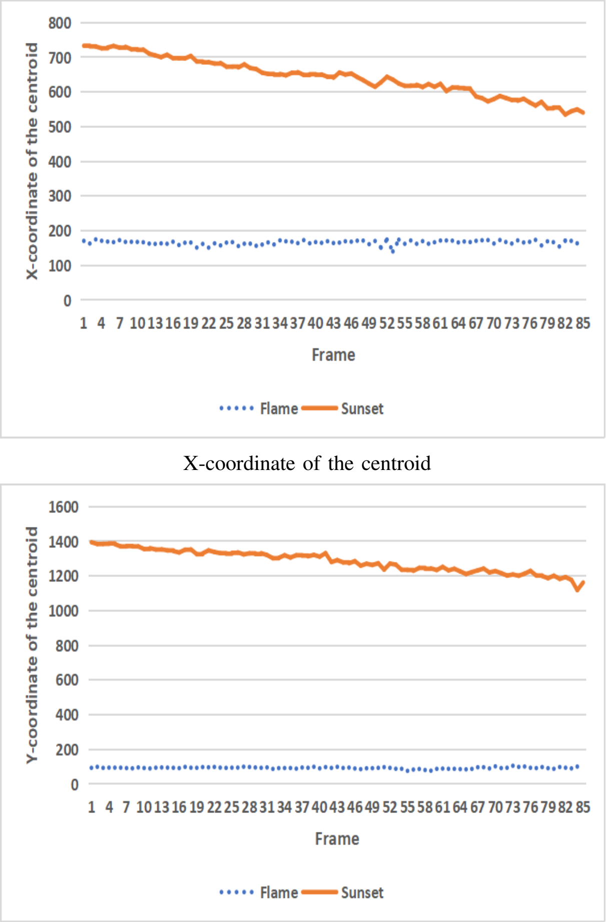
$$g = \frac{N_{t+\Delta t} - N_t}{(t + \Delta t) - t} \tag{3}$$

Where, g is the growth rate, $N_{t+\Delta t}$ and N_t denote the quantity of pixels of the skeleton image in the frame $t + \Delta t$ and t, respectively.

C. Change rate of centroid based on skeleton image

The flame moves and changes along the comburent or the wind direction, but the centroid does not change. This characteristic is different from other rigid objects. As the centroid of the original flame image moves, the centroid of the skeleton image changes correspondingly. The line graphs of the centroids in the flame skeleton images and the sunset skeleton images of videos are shown in Fig. 8.

Fig. 8 shows that the centroids of the flame skeleton images in the video are stable, but the centroids of the sunset skeleton images fluctuate wildly. The standard variances of the X and Y coordinates of the centroids of flame skeleton images are 6.58 and 5.13, respectively, and the standard variances of the X and Y coordinates of the centroids of the sunset skeleton images are 197.2 and 384.7, respectively.



X-coordinate of the centroid

Y-coordinate of the centroid

Fig. 8. Line chart of centroid in skeleton image

In the case of a moving item in a series of video frames, the centroid can be caculated as follows:

$$x_i = \sum_{(x,y) \in S} x / N_s \tag{4}$$

$$y_i = \sum_{(x,y) \in S} y / N_s \tag{5}$$

Where S represents the skeleton image; N_s is the number of pixels of skeleton image; (x_i,y_i) is the centroid coordinates of the skeleton image.

The formulas for the change rate of the centroid are as follows:

$$h_1 = \frac{x_{i_{t+\Delta t}} - x_{i_t}}{(t + \Delta t) - t} \tag{6}$$

$$h_2 = \frac{y_{i_{t+\Delta t}} - y_{i_t}}{(t + \Delta t) - t} \tag{7}$$

Where $(x_{i_{t+\Delta t}}, y_{i_{t+\Delta t}})$ and (x_{i_t}, y_{i_t}) represent the centroid of the skeleton image in the frame $t + \Delta t$ and t, respectively.

D. Gabor texture feature

The Gabor texture feature reflects the static characteristics of the flame. Gabor transform is a windowed

Fourier transform, which can extract the target's spatial and frequency domain features well and obtain direction and scale characteristics simultaneously. The wavelet kernel of the two-dimensional Gabor wavelet transform is constructed using the two-dimensional Gabor function as the wavelet basis. Then the image is convoluted to obtain the Gabor features.

The multi-scale and multi-directional Gabor feature maps contain some redundant information, and using the information of all the feature maps will improve the computational complexity of the algorithm. Therefore, the spatial domain is chosen for feature extraction. In the spatial domain, the mathematical expression of the Gabor filter is shown in equations (8) and (9).

$$g(x,y;\lambda,\theta,\psi,\gamma)=\exp(-\frac{x'^2+\gamma^2y'^2}{2\sigma^2})\exp(i(2\pi\frac{x'}{\lambda}+\psi)) \tag{8}$$

$$\begin{cases} x'=x\cos\theta+y\sin\theta \\ y'=-x\cos\theta+y\cos\theta \end{cases} \tag{9}$$

Where x and y are pixel information, λ denotes the wavelength. θ represents the direction of the filter; ψ is the phase shift of the tuning function. γ represents the longitudinal ratio of space and determines the shape of the filter; σ denotes the standard deviation of the Gaussian function.

Gabor texture feature of the flame image and the flag image are shown in Fig. 9. As shown in Fig. 9, the filters in four directions can enhance the vertical, diagonal, and horizontal features of the flame image, which are significantly different from the flag image.

IV. XGBoost Classifier

After extracting the features of the suspected flame, a flame detection method based on multi-feature fusion and double-layer XGBoost [19] is proposed. The first layer is about feature processing. Shape feature descriptors, texture feature descriptors, growth rate, and centroid change rate are sent into a separate XGBoost model for training and classification. After that, the accuracy rate and recall rate are calculated. The outcome of each independent XGBoost classifier are employed as features, and the accuracy rate and recall rate are adopted to sort the features, which are sent into the second-layer XGBoost model. The structure diagram of the double-layer XGBoost model is displayed in Fig. 10.

A. XGBoost Algorithm

XGBoost, the limit gradient lifting algorithm, is used to supervise the learning problem and predict the target variables using the training data. As a supervised integrated learning algorithm, XGBoost algorithm can be understood as the summation model of multiple decision trees. Its formula is:

$$\widehat{y_i}=\varphi(x_i)=\sum_{k=1}^{n}f_k(x_i), f_k\in F \tag{10}$$

Where k refers to the number of decision trees; $f(k)$ is the independent function of function space; F is the function space, which is composed of decision trees.

There are n samples and m features in the given sample set, and its formula is:

$$D=\{x_i,y_i\}(|D|=n,x_i\in R^m,y_i\in R) \tag{11}$$

Where x_i represents the $i-th$ sample and y_i represents the $i-th$ label.

The function space formula of the above decision tree is:

$$F=\{f(x)=\omega_{q(x)}\} \tag{12}$$

Where $q(x)$ means to assign sample x to a leaf node; ω is the weight of leaf node.

The objective function of XGBoost algorithm is separated into two parts: loss function and regular term. The loss function describes the difference between the predicted value and the real value of the target. Regular terms control the complexity of the tree and prevent over fitting. The objective function formula is:

$$\ell(\phi)=\sum_i l(\widehat{y_i},y_i)+\sum_k\Omega(f_k) \tag{13}$$

$$\Omega(f)=\gamma T+\frac{1}{2}\lambda\|\omega\|^2 \tag{14}$$

Where T is the number of decision trees; γ is the penalty coefficient of T; λ is the coefficient of regularization penalty term.

XGBoost, as a robust algorithm, has many advantages. It adopts a strategy similar to the random forest in each iteration, supports data sampling, and can significantly improve the running speed of the model. In addition, it does not need to deal with missing values. Therefore, we apply it to flame detection.

B. Feature Preprocessing

In the XGBoost algorithm, the input order of features will affect the experimental results: the more advanced the input, the more significant the impact. In addition, the number and format of feature descriptors of different features are different. Thus, shape feature descriptors, texture feature descriptors, growth rate, and centroid change rate of the image are separately sent into a single XGBoost model for training and classification. The accuracy rate and the recall rate are calculated, then the F-score is obtained. After that, features are input in descending order according to the value of the F-Score of each feature. The calculation formula for F-score is shown in equation (15).

$$F-score=2*\frac{precision*recall}{precision+recall} \tag{15}$$

$$precision=\frac{TP}{TP+FP} \tag{16}$$

$$recall=\frac{TP}{TP+FN} \tag{17}$$

Where TP is the number of correctly predicted positive samples, TN is the number of correctly predicted negative samples, FP is the number of negative samples incorrectly predicted, and FN is the number of positive samples incorrectly predicted.

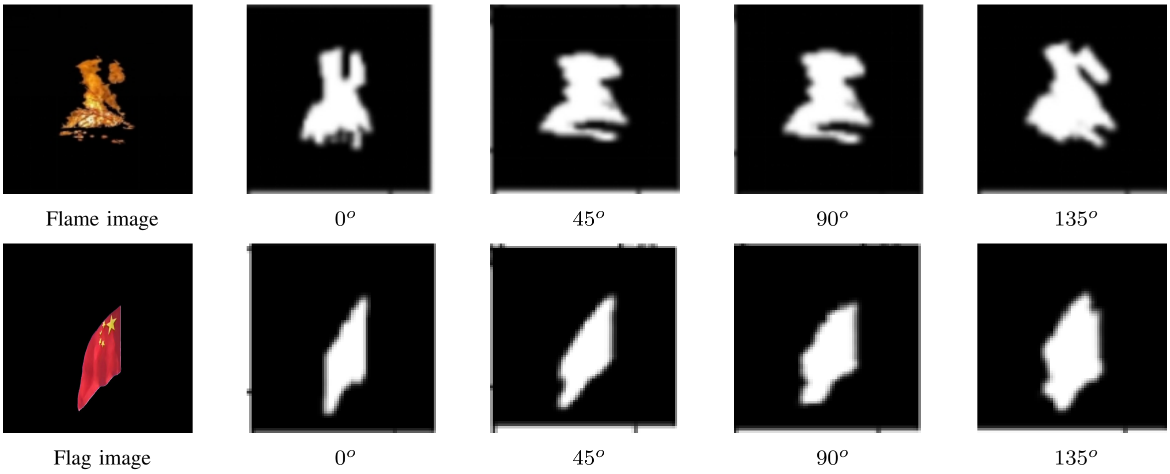


Fig. 9. Gabor texture feature of flame image and flag image

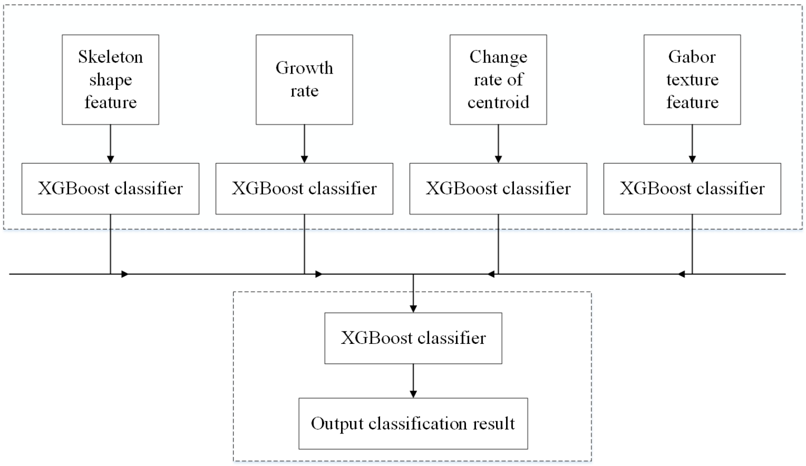


Fig. 10. Structure diagram of the double-layer XGBoost model

C. Construction of the Double-layer XGBoost Model

In the classification process, parameters need to be adjusted to take full advantage of the XGBoost algorithm. The parameters are as follows. The weak estimator is the tree model, and the objective function is logarithmic loss. The table below lists the variable parameters.

TABLE I
PARAMETERS OF THE FIRST-LAYER XGBOOST CLASSIFIER

Parameter	Meaning
$n\_estimaters$	number of trees
max_depth	maximum depth of tree
$subsample$	proportion of samples
$colsample\_bytree$	percentage of random samples
$gamma$	penalty subtracted for each additional leaf
eta	step size of an iterative decision tree
$scale\_pos\_weight$	proportion of positive and negative samples

The simulated annealing algorithm [20] can effectively solve the local optimal problem, and its operating speed is better than that of the grid search method. Thus, it is employed to optimize the parameters in the XGBoost model.

The construction process of double-layer XGBoost model is as follows:

1) Division of data sets: In this paper, from the pool of samples, there are 909 negative samples and 1114 positive samples selected. In the training set, 40 percent of the data is included, while in the verification set, 30 percent of the data is included.

2) Construction of the first layer model: The XGBoost model of each feature is trained with the training set and the verification set. The parameters and F-score value of each feature are shown in Table II.

3) Construction of the second layer model: According to the F-score of each feature, the features are sent into the second-layer classifier in the order of shape feature, texture feature, centroid change rate, and growth rate. After the first-layer model is constructed, 40% of the test data is divided into the training set to train the second-layer model, 30% into the verification set to optimize parameters, and 30% into the test set for testing. The parameter values of the second layer model are shown in the Table III.

V. ANALYSIS OF EXPERIMENTAL RESULTS

In order to test the performance of the proposed algorithm, we chose four flame videos and three non-flame videos for experiments. The samples of the flame videos are shown in Fig. 11. Ture Positive Rate (TPR) and False Negative Rate (FNR) are used as evaluation standards for flame detection, and the calculation formulas are shown in equations (18) and

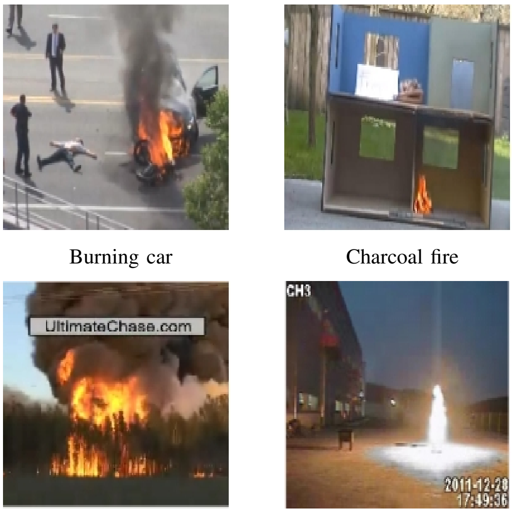
TABLE II
PARAMETERS OF THE FIRST-LAYER XGBOOST CLASSIFIER

Parameter	Texture feature	Shape feature	Growth rate	Centroid change rate
$n\_estimaters$	75	60	65	70
max_depth	20	5	10	15
$subsample$	0.8	0.9	0.9	0.6
$colsample\_bytree$	0.8	1.0	0.6	1.0
eta	0.4	0.4	0.3	0.2
$gamma$	0.1	0.05	0.1	0.15
$scale\_pos\_weight$	600	30	500	600
$F-score$	0.88	0.99	0.73	0.74

TABLE III
PARAMETERS OF THE SECOND-LAYER XGBOOST CLASSIFIER

Parameter	Value	Parameter	Value
$n\_estimaters$	5	max_depth	25
$subsample$	0.5	$colsample\_bytree$	0.8
$gamma$	0.5	eta	0.1
$scale\_pos\_weight$	50		

(19). Table IV describes the situation and detection results of the four videos.


Burning car

Charcoal fire

Forest fire

White fire

Fig. 11. Samples of the flame videos

$$TPR = \frac{TP}{TP+FN} \tag{18}$$

$$FNR = \frac{FN}{TP+FN} \tag{19}$$

TABLE IV
EXPERIMENTAL RESULTS OF FLAME VIDEOS

Number	Video description	TP	FN	TPR	FNR
1	Burning car	182	17	0.915	0.085
2	Charcoal fire	189	10	0.950	0.050
3	Forest fire	185	14	0.930	0.070
4	White fire	191	8	0.96	0.040

Samples of non-flame videos are shown in Fig. 12. Ture Negative Rate (TNR) and False Positive Rate (FPR) are adopted as evaluation indicators for non-flame detection, and the calculation formulas are shown in equations (20) and (21). Table V describes the situation and detection results of the three videos.

Smoke

Street lamp

Headlight

Fig. 12. Samples of the non-flame videos

$$TNR = \frac{TN}{TN+FP} \tag{20}$$

$$FPR = \frac{FP}{TN+FP} \tag{21}$$

TABLE V
EXPERIMENTAL RESULTS OF NON-FLAME VIDEOS

Number	Video description	TN	FP	TNR	FPR
1	Smoke	181	18	0.910	0.090
2	Street lamp	196	3	0.985	0.015
3	Headlight	193	6	0.970	0.030

As shown from Table IV, the algorithm's average detection rate is 0.939, and its average underreporting rate is 0.061. The algorithm's average misreport rate is 0.045, as indicated in Table V. Owing to the modified color model, the detection rate of the white flame video is up to 0.96. The detection rate of forest flame with thick smoke is 0.93. The detection rate of the flame generated by automobile spreading over the car and the charcoal flame shaking with the wind is 0.915 and 0.95, separately. The suggested approach achieves a high detection rate and is suited for a variety of difficult settings, according to experimental data.

Several flame detection algorithms are selected to compare with the proposed algorithm. The comparison is shown in Table VI. Although the misreport rate is reduced in the literature [6], the underreporting rate and detection rate are worse than the algorithm of this paper. The detection rate of literature [10] is lower than that of the program in this study, and the misreport rate is high. The Experimental result means that the proposed algorithm can achieve a great detection rate, effectively reduce the misreport rate and underreporting rate, and is suitable for various complex scenes.

TABLE VI
COMPARISON OF DIFFERENT DETECTION ALGORITHMS

Algorithm	TPR	FPR	FNR
Reference [6]	91.1	4.2	8.9
Reference [10]	93.2	6.8	2.5
Proposed	93.9	4.5	6.1

VI. CONCLUSION

This study provides a video flame identification method based on multi-feature fusion and double-layer XGBoost. First and foremost, a novel flame color feature model based on the YCbCr color space and the HSV color space is created for the purpose of color segmentation. The resulting information is sent into the ViBe algorithm, which allows it to refine its background update approach in order to thoroughly extract the suspected flame zone. In the next step, the static and dynamic characteristics of the flame are combined in order to increase the resilience in complicated situations. In addition, a double-layer XGBoost model is created with the goals of accuracy and instantaneity. In order to improve the model parameters, the simulated annealing method is used. The experimental findings in complicated situations demonstrate that the suggested approach is capable of reliably detecting flames. It may be utilized in a variety of applications including indoor and outdoor fire detection, and forest fire detection.

REFERENCES

[1] Y. Yan, J. Du, S. Gao, J. Zhou, and Y. Liu, "Video flame detection based on fusion of multi-feature," *Journal of Computer-Aided Design and Computer Graphics*, vol. 27, no. 3, pp. 433–440, 2015.

[2] H. Huang, P. Kuang, L. Fan, and H. Shi, "An improved multi-scale fire detection method based on convolutional neural network," in *17th International Computer Conference on Wavelet Active Media Technology and Information Processing*, 2020.

[3] Q. Tian, "An autoadaptive edge-detection algorithm for flame and fire image processing," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 5, pp. 1486–1493, 2012.

[4] L. H. Chen and W. C. Huang, "Fire detection using spatial temporal analysis," *Lecture Notes in Engineering & Computer Science*, vol. 2206, no. 1, pp. 2222–2225, 2013.

[5] X. G. Su and W. Zhang, "Flame foreground extraction under the interference of intense light," *Computer Engineering & Science*, vol. 39, no. 11, pp. 2060–2065, 2017.

[6] Q. Y. Wu, Y. Y. Yan, J. Du, and S. B. Gao, "Fire detection based on fusion of multiple features," *Journal of Graphics*, no. 2, pp. 240–247, 2017.

[7] J. J. Mei and Z. W., "Early fire detection algorithm based on vibe and machine learning," *Acta Optica Sinica*, vol. 38, no. 7, pp. 52–59, 2018.

[8] C. Zhang, Q. H. Meng, and T. Jing, "Video flame detection algorithm based on improved gmm and multi-feature fusion," *Laser & Optoelectronics Progress*, vol. 58, no. 4, pp. 128–137, 2021.

[9] W. J. Qi, Y. H. Wang, and X. Z. Guo, "Research on large space image flame detection method," *Modern Electronics Technique*, vol. 42, no. 4, pp. 76–79,84, 2019.

[10] S. T. Zeng, H. B. Wu, and P. H. Shen, "Video fire detection based on fusion of multiple features," *Journal of Graphics*, vol. 38, no. 4, pp. 549–557, 2017.

[11] Y. T. Liao, H. Q. Wang, Q. Chai, Y. Lu, and Z. F. Ma, "On applying adaboost algorithm in image fire detection," *Computer Applications and Software*, no. 4, pp. 153–155,180, 2015.

[12] X. Y. Zhu, Y. Y. Yan, Y. A. Liu, and S. B. Gao, "Flame detection based on deep forest model," *Computer Engineering*, vol. 44, no. 7, pp. 264–270, 2018.

[13] Y. Tan, L. B. Xie, H. W. Feng, L. Peng, and Z. D. Zhang, "Flame detection algorithm based on image processing technology," *Laser & Optoelectronics Progress*, vol. 56, no. 16, pp. 107–113, 2019.

[14] Z. Yang, L. Bu, T. Wang, P. Yuan, and O. Jineng, "Indoor video flame detection based on lightweight convolutional neural network," *Pattern Recognition and Image Analysis*, vol. 30, no. 3, pp. 551–564, 2020.

[15] J. T. Cao, Y. Y. Qin, and X. F. Ji, "Review on video based flame detection algorithm," *Journal of Data Acquisition & Processing*, vol. 35, no. 1, pp. 35–52, 2020.

[16] T. Y. Chen, S. T. Zeng, and H. B. Wu, "Flame image segmentation method based on ycbcr color space," *Transducer and Microsystem Technologies*, vol. 30, no. 10, pp. 62–64, 2011.

[17] Y. B. Wang, Q. Han, Y. Y. Li, and Y. J. Li, "Video smoke detection based on multi-feature fusion and modified random forest," *Engineering Letters*, vol. 29, no. 3, pp. 1115–1122, 2021.

[18] S. Shi, Y. Yu, Z. J. Yang, and M. Yu, "Face image matching algorithm based on surf and shape context," *Application Research of Computers*, vol. 35, no. 10, pp. 3197–3200, 2018.

[19] G. Y. Li, W. Li, X. L. Tian, and Y. F. Che, "Short-term electricity load forecasting based on the xgboost algorithm," *High Voltage Engineering*, vol. 7, no. 4, pp. 274–285, 2021.

[20] M. El Alaoui and M. Ettaouil, "An adaptive hybrid approach: Combining neural networks and simulated annealing to calculate the equilibrium point in max-stable problem." *IAENG International Journal of Computer Science*, vol. 48, no. 4, pp. 893–898, 2021.