Smoke-Detection Framework for
High-Definition Video Using Fused Spatial- and
Frequency-Domain Features

ZHE LIU\textsuperscript{1}, XIAOPING YANG, YING LIU, AND ZHIHONG QIAN\textsuperscript{2}, (Senior Member, IEEE)

College of Communication Engineering, Jilin University, Changchun 130012, China

Corresponding author: Zhihong Qian (dr.qzh@163.com)

This work was supported in part by the Special Fund Project of Jilin Province School Co-Construction Plan under Grant SXGJQY2017-9, in part by the High-Level Technology Innovation Team Building Project of Jilin University under Grant 2017TD-19, and in part by the National Natural Science Foundation of China under Grant 61771219.

ABSTRACT Video-based smoke detection is an effective method for fire alarm systems. Given the widespread use of high-definition cameras, a smoke detection method for high-definition video is needed. This paper proposes a smoke-detection framework for high-definition video, in which the main idea is to use the small smoke image blocks to match the image features of the motion area in the video and to use the support vector machine classifier for smoke recognition. The ViBe algorithm and other methods are used to effectively extract the areas for classification. This detection framework consists of spatial- and frequency-domain features. In the extraction of frequency domain features, we use local phase quantization (LPQ) features. In the local texture features of the spatial domain, we add the compensation of adjacent pixels and consider the gradient of the symmetrical pixels using the center-symmetric local binary pattern feature. To improve results, we also propose the trisection feature fusion scheme for features in the spatial and frequency domains. The experiments show that using the feature extraction and fusion schemes, our smoke-detection framework achieves the good performance in the detection of smoke in the video from different datasets.

INDEX TERMS High-definition video, local binary patterns, smoke detection, support vector machines.

I. INTRODUCTION

Fire is one of the major disasters that seriously endanger the safety of human life and property. Timely warning of fires is of great significance for reducing property losses. Generally speaking, smoke is the first to appear in the early stages of a fire, therefore, an effective smoke detection method that enables the fire to be quickly detected and controlled can effectively avoid the serious consequences of fire. Conventional fire smoke detection usually uses sensors such as light, smoke, and temperature sensors. However, these sensors have the drawbacks that only specific types of smoke can be detected and their use is limited in open space environments. Given the popularity of electronic cameras, a large number of video surveillance devices have been installed in many public places and buildings. Video surveillance-based systems provide effective coverage for larger areas and can be easily integrated into existing closed-circuit monitoring systems. With the continuous development of camera technology and the increasing density of monitoring points, smoke detection using video surveillance has become an important research topic.

At present, a number of smoke-detection algorithms using video images have been proposed. The main process of a video smoke detection algorithm is to determine candidate smoke areas, and then use a classifier to classify static features such as wavelets directly extracted from the candidate smoke area to determine the final smoke detection result. These algorithms directly detect the pixel values in each video frame, and the determination of the characteristics of the smoke pixel itself is often inaccurate. Jayavardhana \textit{et al.} \cite{1} proposed a video smoke detection method using wavelets to extract the features of smoke, and they trained a support vector machine (SVM) classifier using the extracted texture features to classify the smoke. Ye \textit{et al.} \cite{2} used the surfacelet wavelet transform and a hidden Markov tree (HMT) model to...
extract the texture features of smoke. For each image block in this method, the surfacelet transform is applied, then a three-dimensional HMT model is constructed to estimate the dynamic texture features from the coefficients of the blocks. Töreyin et al. [3] used temporal and spatial wavelet analysis to detect semi-transparent smoke using a static camera. These methods, which use improved wavelet transforms, have lower computational efficiency because they only determine the high-frequency components.

Moreover, many researchers have proposed smoke identification based on multiple stages. Such methods are described in [4], [5]. These identification methods are mainly divided into three stages, usually called the preprocessing stage, feature extraction stage, and classification stage. Tung and Kim [6] proposed a four-stage smoke detection algorithm. First, an approximate median method is used to obtain the moving region and then they cluster the candidate smoke regions using a fuzzy C-means clustering algorithm. They then extracted the spatial and temporal characteristics of candidate smoke regions. In final stage, the candidate regions are classified as smoke or non-smoke by an SVM classifier. This technique requires a long time to process each frame of the input videos. In [7], the author also combined temporal and spatial characteristics to achieve early fire detection; three effective static and dynamic visual features of smoke were used in this algorithm.

Local binary patterns (LBP) have many applications in the detection of smoke, and their advantage lies in their low computational cost. The success of LBP features in various computer-vision problems and applications has inspired much new research on different LBP variants. Yuan [8] computed the histograms of LBPs and proposed LBPs-based-on-variance (LBPV) pyramids, then used a neural network classifier to differentiate smoke. Zhao et al. [9] defined local binary motion patterns to describe the dynamic texture features of smoke. Tian et al. [10] used non-redundant LBP based features for smoke classification. Finally, Yonghua and Jin-Con [11] identified defects in wood based on texture analysis.

Most of the above articles use a public dataset from Bilkent University, which can be downloaded from http://signal.ee.bilkent.edu.tr/VisiFire/Demo/SampleClips.html. However, the dataset is currently quite dated and has a low video resolution that is not suitable for current practical applications. With the increasing number of devices such as cameras and mobile phones, it is difficult to meet current needs with methods designed for low-resolution images. In [12], a video-based smoke-detection method using dynamic texture feature extraction with volume LBPs was proposed. The author also provided a high-definition smoke video dataset on which the proposed method achieved good results.

In this paper, a new framework is designed that uses a smoke image dataset to train a model of smoke features. The model is used to confirm whether a detected candidate region in the video contains smoke. This framework is especially suitable for high-definition video. Before using this framework, a feature extraction process is essential. For the feature extraction part, we propose a new LBP operator based on central symmetric gradient compensation that we call the central symmetric gradient compensation LBP (CSGC-LBP) operator. To describe the texture features, we use the spatial and frequency domains to extract the smoke features. In addition, a three-way feature fusion scheme is proposed for the two classification features in the spatial and frequency domains. Experimental results on two smoke video datasets show our method has strong discriminative capabilities for smoke detection.

The rest of this paper is organized as follows: Section II describes the extraction method for image texture features and the spatial and trisection feature fusion scheme. Section III introduces the smoke-detection framework. Section IV presents several experiments and a comparison of the performance of the proposed algorithm with respect to other LBP operators. Section V draws the conclusions of this study.

II. FEATURE EXTRACTION AND FUSION

A. CENTER-SYMMETRIC LBP

The LBP operator is a grayscale texture descriptor proposed by Ojala et al. [13] that can capture the spatial characteristics of images. It has strong classification ability and high computational efficiency. The basic LBP operator is defined as follows:

\[
LBP = \sum_{i=1}^{N} s(g_i - g_c) 2^i \tag{1}
\]

\[
s(x) = \begin{cases} 
1, & x \geq 0 \\
0, & x < 0 
\end{cases} \tag{2}
\]

where \(N\) indicates the total number of neighborhood, \(g_c\) is the central gray-scale pixel value of the neighborhood, and \(g_i\) is the gray-scale pixel value of the \(g_c\) neighborhood, in the basic LBP operator, \(i\) ranges from 1 to 8, as shown in Figure 1, assume that the numbers are arranged in a clockwise rotation. The basic principle of the LBP operator is to compare the grayscale value of the central pixel and the neighboring pixel, then use a Boolean function to represent the result. Finally, the grayscale value of the central pixel is obtained by a binary-weighted assignment as a texture feature.

\[FIGURE 1. \text{Pixels in the LBP and CS-LBP operator.}\]
The center-symmetric LBP (CS-LBP) operator not only keeps crucial local features such as LBP, but also has a lower number of feature dimensions. The aim of CS-LBP is to use a smaller number of LBP labels to produce shorter histograms. As a local texture description operator, CS-LBP encodes an image using a centrally symmetric idea based on the LBP operator. The basic method of CS-LBP compares the size of the neighborhood pixel value pairs \( g_i \) and \( g_{i+4} \) (\( i = 1, 2, 3, 4 \)) as shown in Fig. 1, which are symmetrically centered on the central pixel in the defined neighborhood. When the corresponding binary value is greater than 0, the position is recorded as 1. The resulting binary strings are then arranged in order and converted to a decimal number to form the CS-LBP encoding of the central pixel’s value. The CS-LBP operator is thus defined as follows:

\[
\text{CS-LBP} = \sum_{i=1}^{4} s(g_i - g_{i+4}) 2^i
\]

\[
s(x) = \begin{cases} 
1, & x \geq 0 \\
0, & x < 0 
\end{cases}
\]

**B. CSGC-LBP**

It is obvious that the CS-LBP operator ignores the information of the central pixel in its calculation. Moreover, CS-LBP has no mechanism to compensate for the sudden occurrence of noise in an image. Based on the gradient compensation method, we improved the CS-LBP operator and named it the central symmetric gradient compensation-LBP (CSGC-LBP). This operator is mainly composed of two parts: first, the central pixel is incorporated into the gradient calculation, and secondly, the influence of noise is reduced by gradient compensation.

To utilize the information provided by the central pixel, we take the central position pixel of the \( 3 \times 3 \) neighborhood as \( g_c \) and use the central symmetric gradient and difference from the neighborhood mean to weight the neighboring pixels. In the gradient weighting method, each set of symmetric pixels in the CS-LBP is split and the gradient values of the pixel and the central pixel are respectively obtained.

\[
s(g_i) = \begin{cases} 
g_i + g_{i+4} + g_i - g_c, & i \leq 4 \\
g_i - g_{i+4} + g_i - g_c, & i > 4 
\end{cases}
\]

Here, we take into account compensation from the neighborhood pixels to reduce the impact of sudden noise. The compensation scheme is described below. When calculating the weight of each pixel, a local compensation adjustment to the gradient is added. Specifically, each compensation neighborhood is defined as a \( 3 \times 3 \) neighborhood connecting each pixel and the central pixel, extending two unit pixels to each side. As shown in Fig. 2, for point \( g_1 \), the compensation pixels are \( g_{u1}, g_{u2}, g_{d1}, \) and \( g_{d2} \). Compensation pixels are calculated as follows:

\[
g_{cp} = \frac{\sum g_{in} + \sum g_{out}}{in + out}
\]

where \( g_{in} \) refers to the diagonal pixel of the pixel in the \( 3 \times 3 \) neighborhood (\( g_1, g_c, \) and \( g_5 \) in the example in Fig. 2). Pixel \( g_{out} \) has the compensation pixels \( g_{u1}, g_{u2}, g_{d1}, \) and \( g_{d2} \). Because the compensation of pixels in a symmetric pixel pair would be calculated twice, we do this for only one pixel in the symmetric pixel pair (\( i \leq 4 \)).

This compensation adjustment method is employed in the proposed CSGC-LBP operator, which is computed as follows:

\[
\text{CSGC-LBP} = \sum_{i=1}^{8} u_{\text{csgc-lbp}}(s(g_i)) 2^i
\]

\[
s(g_i) = \begin{cases} 
g_i + g_{i+4} + g_i - g_c, & i \leq 4 \\
g_i - g_{i+4} + g_i - g_c, & i > 4 
\end{cases}
\]

\[
u_{\text{csgc-lbp}}(x) = \begin{cases} 
1, & x \geq 0 \\
0, & x < 0 
\end{cases}
\]

The proposed CSGC-LBP is used as the spatial-domain feature in the proposed smoke-detection framework. A 256-dimensional vector is obtained in each image block after processing.

**C. LOCAL PHASE QUANTIZATION**

The local phase quantization (LPQ) algorithm is a type of texture feature descriptor with invariance to blur. The spatial blur of an image can be expressed as a convolution of the image intensity and a point spread function (PSF). In the frequency domain, it can be described as \( G = F \ast H \), where \( G \) represents the discrete Fourier transform of the blurred image, \( H \) corresponds to the original image, and \( F \) stands for the PSF. Therefore, the phase relationship of the spectrum can be expressed as \( \angle G = \angle F + \angle H \). When the PSF is a central symmetric function, the Fourier transform of \( H \) is always a real number, that is, \( \angle H \in \{0, \pi\} \).
The shape of the Fourier transform of $H$ of the conventional PSF is considered to be a Gaussian or sinc function, which at least guarantees that $H$ is a constant value at low frequencies, for example, when $\angle H = 0$, then $\angle G = \angle F$.

LPQ calculates the phase of the neighborhood $N_x$ of each pixel $x = [x_1, x_2] \in \mathbb{R}^2$ on image $f(x)$. The local spectrum $F(u, x)$ is computed using the discrete short-time Fourier transform (STFT), defined by

$$F(u, x) = \sum_y f(x - y)e^{-j2\pi u^T y}$$

(10)

where $u$ is the frequency; LPQ is calculated at four frequency points, $\mu_1 = \{a, 0\}^T$, $\mu_2 = \{0, a\}^T$, $\mu_3 = \{a, -a\}^T$, and $\mu_4 = \{-a, a\}^T$, which are used to compute the local Fourier coefficients. Here, $a$ is sufficient for $H(\mu_i) > 0$, and each pixel in the neighborhood can be represented by the vector

$$F(x) = [F(\mu_1, x), F(\mu_2, x), F(\mu_3, x), F(\mu_4, x)]$$

(11)

The phase information of the Fourier coefficients is represented by the sign function of the real and imaginary parts of each component in $F(x)$. This step is performed using a simple scalar quantizer.

$$q_j = \begin{cases} \begin{array}{ll} 1, & g_j \geq 0 \\ 0, & g_j < 0 \end{array} \end{cases}$$

(12)

where $g_j(x)$ is the $j$th component of $G_x = [\text{Re}\{F_x\}, \text{Im}\{F_x\}]$. The final LPQ feature is calculated as

$$f_{LPQ}(x, y) = \sum_{j=1}^{8} q_j(x, y)2^{j-1}$$

(13)

In the subsequent work, we use LPQ as the frequency domain feature. Here, the feature of LPQ is 256-dimensional.

D. TRISECTION FEATURE FUSION

The CSGC-LBP feature is a spatial-domain feature, while LPQ is a feature in the frequency domain. Therefore, these two features are not directly comparable. Our aim in this study is to use a combination of the spatial- and frequency-domain features to develop a better smoke classification model.

Traditionally, two or more sets of features that are not comparable are linearly connected. The usual approach is to directly extract the two sets of features, which are then normalized and joined to form a new vector. In the SVM classifier, the consequence of directly extracting features is that the learning factors and other parameters are often not properly used, and the consequences of linear connections are often not obvious. In fact, such a linear concatenation sometimes reduces the classification performance. For smoke detection, the two features are not ideal as a cascade. Hence, we considering fusing features after the two sets of features have been extracted and normalized. The scheme adopts the idea of feature weighting and sets the ratio of spatial domain feature vector to frequency domain feature vector as a variable. Algorithm 1 describes the trisection feature fusion procedure.

The ratio of the two sets of features defined as $\lambda$, and the interval of $\lambda$ is set to an initial range from $\lambda_{\text{min}}$ to $\lambda_{\text{max}}$. Two points are set on the trisection in the range of $\lambda$ and marked as $g_1$ and $g_2$. The SVM classifier is used for each prediction in each iteration. The range of $\lambda$ is updated by comparing the accuracy from SVM prediction. In fact, this is a negative feedback process and screening scope. The advantage of this fusion algorithm compared to other multi-step methods is that it reduces the steps needed for re-prediction, which is highly advantageous for accelerating the smoke detection time. Because the training process of the image blocks is separated from the video prediction process, the proposed feature fusion scheme can substantially improve the system efficiency.

Algorithm 1 Trisection Feature Fusion

Input: Training datasets $s = \{(x_i, y_i)\}$ with labels $y \in \{-1, +1\}$.  
Set $\lambda_{\text{min}} \leftarrow 0.01$, $\lambda_{\text{max}} \leftarrow 100$ as the end points of coefficient ratio; $\text{iters}$ as the number of iterations; $\text{iter}_{\text{max}}$ as the maximum of the iterations 
while $\text{iters} < \text{iter}_{\text{max}}$ do 
compute $g_1 = (\lambda_{\text{max}} - \lambda_{\text{min}})/3 + \lambda_{\text{min}}$ 
compute $g_2 = (\lambda_{\text{max}} - \lambda_{\text{min}})/3$ 
compute $\text{acc}_g$ as the accuracy when the coefficient ratio is $g_1$ 
compute $\text{acc}_g$ as the accuracy when the coefficient ratio is $g_2$ 
\quad \text{case } \text{acc}_g > \text{acc}_{g_2} 
\quad \lambda_{\text{max}} \leftarrow g_2 
\quad \text{case } \text{acc}_g < \text{acc}_{g_2} 
\quad \lambda_{\text{min}} \leftarrow g_1 
\quad \text{iters} \leftarrow \text{iters} + 1 
end while 

III. VIDEO-BASED SMOKE-DETECTION FRAMEWORK

A. SMOKE-DETECTION FRAMEWORK

For a long time, the conventional approach to detecting smoke in video has been to cut the image from the original video into small image blocks and classify each image block. The disadvantage of this is that for each grid, a discriminating process is required. In high-definition video, this takes a long time, and sometimes misclassification is caused by some grids that cannot completely cover the smoke.

In this section, a smoke-detection framework for high-definition video is proposed. This framework is divided into two stages: an image training stage and a video prediction stage. Using the training of small image blocks, a training model is obtained. This model needs to have a good classification effect on smoke and has a fast pro-
cessing speed. During the detection process of the video, the motion area to be detected is found, and the image blocks of the same size as the training stage are divided in the motion area, and it is hence only necessary to determine whether the image blocks contain smoke in the detection process. The proposed smoke-detection framework is described in Fig. 3.

First, during the image training stage, we use small images to obtain the characteristics of smoke. The size of the small image blocks are 100 × 100 pixels. SVM classification is employed to classify the extracted features. This process yields a prediction model for small smoke images. Then, using this prediction model, we predict and mark the smoke in the video.

The processing of the video is divided into the following steps, as shown in Fig. 3. First, the motion regions in the video are extracted using the ViBe algorithm. Then, morphological processing of the motion regions is performed for edge detection. Specifically, the segmented area is reduced by the closing operation and the edge contour of the motion region is detected by the Sobel operator. Image blocks of 100 × 100 pixels (the same size as those used for training) centered on the center of the graphic are created. Using the model obtained in the first step, these image blocks are classified, and those predicted to contain smoke are marked in the original image.

In this framework, the detection of high-definition video actually only needs to detect the area left after edge detection. The advantage is that the time needed for detection is substantially reduced. The result of the detection follows the position of the smoke in the image instead of that of the grid.

Note that the dataset is prepared in advance in the first stage of learning; however, the images processed in the second stage are not related to those in the first stage when processing the video. Therefore, pre-training the prediction model before detecting smoke in a video is an efficient solution. Once an optimized model is obtained, the detection in the second stage is reliable.

B. ViBe for Motion Detection

In the ViBe algorithm [14], when motion is detected, it is determined whether the current pixel is a foreground pixel by comparing the current pixel value with the corresponding historical pixel value in the background model. Assume that the value of pixel $x$ in the current frame image is $v(x)$, as shown in Fig. 4, where $C_1$ and $C_2$ are components of the two-dimensional color space ($C_1$, $C_2$). In addition, $S(R)$ is a region with $v(x)$ as the center and $R$ as the radius. This region contains all points with a Euclidean distance to $v(x)$ of less than $R$. When the Euclidean distance of $v(x)$ is greater than the threshold, it is determined that $v(x)$ is the background; otherwise, it is labelled as foreground. In Fig. 4, the background points are $v_1$, $v_3$, and $v_4$, and point $v_2$ belongs to the foreground.

To update the model, the ViBe algorithm uses a random update strategy with an update probability of $\lambda$. After the current pixel $v(x)$ has been determined to be a background pixel, it will be used as a new sample to replace an old sample in the background sample model. This replaced sample is generated by random decision making. Specifically, $v(x)$ has...
TABLE 1. Image data sets for smoke detection.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Smoke Images</th>
<th>Non-smoke Images</th>
<th>Total Number</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set 1</td>
<td>552</td>
<td>831</td>
<td>1,383</td>
<td>Training</td>
</tr>
<tr>
<td>Set 2</td>
<td>688</td>
<td>817</td>
<td>1,505</td>
<td>Testing</td>
</tr>
<tr>
<td>Set 3</td>
<td>2,201</td>
<td>8,511</td>
<td>10,712</td>
<td>Testing</td>
</tr>
<tr>
<td>Set 4</td>
<td>2,254</td>
<td>8,363</td>
<td>10,617</td>
<td>Testing</td>
</tr>
</tbody>
</table>

TABLE 2. Description of comparison methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBP</td>
<td>Original LBP with U2, RI, and RIU2, respectively [13]</td>
</tr>
<tr>
<td>HOG</td>
<td>Histogram of oriented gradient [15]</td>
</tr>
<tr>
<td>PLBP</td>
<td>Pyramid local binary patterns with U2, RI, and RIU2, respectively [16]</td>
</tr>
<tr>
<td>CLBP</td>
<td>Completed local binary patterns with U2, RI, and RIU2, respectively [17]</td>
</tr>
<tr>
<td>NRLBP</td>
<td>Noise-resistant local binary patterns with U2 [18]</td>
</tr>
<tr>
<td>POEM</td>
<td>Patterns of oriented edge magnitudes with U2, RI, and RIU2, respectively [19]</td>
</tr>
<tr>
<td>LTrP</td>
<td>Local tetra patterns with U2, RI, and RIU2, respectively [20]</td>
</tr>
<tr>
<td>LDP4</td>
<td>Local derivative pattern with U2, RI, and RIU2, respectively [21]</td>
</tr>
<tr>
<td>PRICoLBP</td>
<td>Pairwise rotation-invariant co-occurrence local binary pattern [22]</td>
</tr>
</tbody>
</table>

a probability of $1/\lambda$ of updating the background sample model of the pixel and randomly replacing one of the samples. The probability that each sample will remain after time $t$ is

$$p(t) = e^{-\ln\left(\frac{n}{N-1}\right)t}$$  \hspace{1cm} (14)

This strategy guarantees a smooth life cycle of the sample. At the same time, to maintain the consistency of the pixel space, the ViBe algorithm uses the same method to randomly select one pixel from the 8-connected neighborhood to update the background model.

C. EDGE DETECTION

The process of edge detection is divided into two aspects: integrating the motion area and extracting the edge area.

After the ViBe algorithm, some small areas of motion are marked. These small areas may be background noise, or the motion area may be too small. The goal of integration is to bring these small areas that are close in distance together into a large area, and far-away (outlier) areas can be judged to be background noise and discarded. For an image $G$ and a defined structural element $S$, the closing operation is mathematically the result of dilation and erosion as follows:

$$G \ast S = (G \oplus S) \ominus S$$  \hspace{1cm} (15)

where $\oplus$ denotes the dilation operation and $\ominus$ denotes the erosion operation. Closing operations can eliminate small holes (black areas) and smooth the outline of an object on a binary image. Unlike opening operations, closing operations generally connect narrow gaps to form elongated bends and fill holes that are smaller than the structural elements. After extracting the motion regions, many unconnected areas are often generated, and these areas can be reduced by the closing operation for subsequent processing.

To effectively segment the background region, the edge of each connected domain needs to be extracted by the Sobel operator. This operation calculates the vertical and horizontal gradients of the image and combines them to obtain the final gradient value. Finally, the edge of the image is determined by the result of a threshold comparison.

The Sobel operator is an operator based on a first-order differential edge detection. The traditional Sobel operator
<table>
<thead>
<tr>
<th></th>
<th>Set 2</th>
<th>Set 3</th>
<th>Set 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TPR</td>
<td>FPR</td>
<td>ACC</td>
</tr>
<tr>
<td>CSGC-LBP</td>
<td>96.3</td>
<td>3.63</td>
<td>96.7</td>
</tr>
<tr>
<td>LPQ</td>
<td>93.2</td>
<td>6.83</td>
<td>93.8</td>
</tr>
<tr>
<td>Trisection</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature Fusion</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LBP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2</td>
<td>98.5</td>
<td>3.79</td>
<td>97.3</td>
</tr>
<tr>
<td>RI</td>
<td>98.1</td>
<td>3.79</td>
<td>97.1</td>
</tr>
<tr>
<td>RIU2</td>
<td>98.8</td>
<td>6.98</td>
<td>95.7</td>
</tr>
<tr>
<td>HOG</td>
<td>89.4</td>
<td>11.5</td>
<td>88.9</td>
</tr>
<tr>
<td>PLBP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2</td>
<td>91.0</td>
<td>1.35</td>
<td>95.2</td>
</tr>
<tr>
<td>RI</td>
<td>95.5</td>
<td>1.59</td>
<td>97.1</td>
</tr>
<tr>
<td>RIU2</td>
<td>97.7</td>
<td>2.20</td>
<td>97.7</td>
</tr>
<tr>
<td>CLBP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2</td>
<td>97.1</td>
<td>2.57</td>
<td>97.3</td>
</tr>
<tr>
<td>RI</td>
<td>98.1</td>
<td>1.59</td>
<td>98.3</td>
</tr>
<tr>
<td>RIU2</td>
<td>99.0</td>
<td>4.28</td>
<td>97.2</td>
</tr>
<tr>
<td>NRLBP</td>
<td>81.3</td>
<td>18.7</td>
<td>81.3</td>
</tr>
<tr>
<td>POEM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2</td>
<td>67.7</td>
<td>16.6</td>
<td>76.2</td>
</tr>
<tr>
<td>RI</td>
<td>74.3</td>
<td>27.3</td>
<td>73.4</td>
</tr>
<tr>
<td>RIU2</td>
<td>76.0</td>
<td>32.7</td>
<td>71.3</td>
</tr>
<tr>
<td>LTrP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2</td>
<td>57.3</td>
<td>0.12</td>
<td>80.4</td>
</tr>
<tr>
<td>RI</td>
<td>94.9</td>
<td>0.36</td>
<td>97.5</td>
</tr>
<tr>
<td>RIU2</td>
<td>97.5</td>
<td>1.10</td>
<td>98.3</td>
</tr>
<tr>
<td>LDP4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2</td>
<td>77.6</td>
<td>0.98</td>
<td>89.2</td>
</tr>
<tr>
<td>RI</td>
<td>97.5</td>
<td>2.33</td>
<td>97.6</td>
</tr>
<tr>
<td>RIU2</td>
<td>98.1</td>
<td>3.33</td>
<td>97.6</td>
</tr>
<tr>
<td>PRICoLBP</td>
<td>98.2</td>
<td>1.84</td>
<td>98.2</td>
</tr>
</tbody>
</table>

TABLE 3. Experimental results.

contains two sets of $3 \times 3$ matrices, denoted by $M_x$ and $M_y$.

$$M_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$$ (16)

$$M_y = \begin{bmatrix} -1 & -2 & +1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$$ (17)

where $M_x$ detects the vertical gradient direction for the horizontal edges; $M_y$ detects the horizontal gradient direction for the vertical edges. For original image $G$, the calculation of the edges is as follows:

$$G_x = M_x \ast G$$ (18)

$$G_y = M_y \ast G$$ (19)

$$G'(i, j) = \sqrt{G_x^2 + G_y^2}$$ (20)

In the above formula, it is determined whether it is an edge point by comparison with a threshold. Finally, $G'(i, j)$ is the image obtained by the Sobel operator.
D. CLASSIFICATION AND PREDICTION

An SVM was selected as a classifier for the features in the proposed framework. As a supervised machine learning algorithm, SVM is extensively used in smoke detection due to its robust performance for two-class classification (smoke or non-smoke). For non-linear data, a kernel function is used to map the initial data into a high dimensional feature space. By projecting data into the feature space and finding the separating hyperplane that maximizes the margin between the data, SVM transform a nonlinearly separable problem into a linearly separable problem with different kernel functions. Several kernel functions such as sigmoid, polynomial, and Gaussian radial basis function (RBF) kernels have been used in different applications. We selected RBF as kernel after an analysis of a number of smoke video clips. This kernel is defined as follows:

\[ k(x, x') = e^{-\frac{\|x-x'\|^2}{2\sigma^2}} \]  

(21)

where vectors \( x \) and \( x' \) are the input patterns and \( \sigma \) determines the width of the kernel function. For large values of \( \sigma \), the kernel function forms an oval around the data points without defining the shape or pattern. If the \( \sigma \) values are small, overtraining, in which the basis function is wrapped tightly around the data points, can occur.
In our study, \( \sigma \) was tuned using pattern search by optimization tools. In the optimization process, the objective function is formulated to consider the classification error of 5-fold cross-validation. A pattern-based search technique is used to minimize the objective function using the initial values of the kernel parameters. The optimization process was run 10 times. Finally, the \( \sigma \) value that corresponded to the minimum objective function of the 10 experiments was selected.

We took the centroid point coordinates of each closed area detected by the Sobel operator in \( G'(i,j) \) and recorded them. Using each centroid point as the center, a \( 100 \times 100 \) square region in the original image of the corresponding frame was extracted. In this step we discarded the centroid point at the...
edge of the image. After that, we converted each square image block to grayscale, then we used the SVM-based smoke feature model to classify them. Image blocks identified as smoke are outlined the original image.

IV. EXPERIMENTS

A. EVALUATION OF THE TRISECTION FEATURE FUSION

To measure the performance of the fusion scheme, we compare the fusion scheme with existing algorithms. The experimental data set is publicly available from http://sim.jxufe.cn/yfn/vsd.htm. Table 1 lists four subsets of the data set: sets 1–4. In the experimental comparison, to evaluate generalization performance, the smallest set (set 1) is used as the training set, and sets 2, 3, and 4 are used as the test set. In the comparative experiment, the image was resized to $48 \times 48$ and converted to a grayscale image. The comparison algorithms are described in Table 2, and the results of the experiment are listed in Table 3, where TFF is the trisection feature fusion procedure proposed in Section II. We use the true positive rate (TPR), false positive rate (FPR), and accuracy (ACC) as performance metrics.

Using our feature fusion algorithm, the initial $\lambda_{\text{min}}$ is taken as 0.01 and $\lambda_{\text{max}}$ is taken as 100. That is, the ratio of the characteristics of LPQ and CSGC-LBP range from 0.01 to 100, we set the number of iterations to 50. The result of feature fusion is shown in Fig. 5. Note that the ACC of the generation in Fig. 5 was obtained by cross-validation in set 1 and is not the true ACC.

The LBP improvement operator in Table 2 is based on the spatial domain, and the LPQ and HOG [15] operators extract other features of the image. Because it is difficult to confirm whether there are correlations between different extended LBP operators, they should be compared separately. Table 3 shows that the LPQ feature performs better on the three datasets than the HOG feature, so it is suitable as a joint domain feature. Compared with other extended LBP operators, CSGC-LBP has the advantage of fast calculation because only the gradient and regional compensation are calculated, and the derivative and wavelet transform are avoided.

The results are compared in Fig. 6. A good operator should have both high TPR and low FPR. A larger interval between the TPR and FPR indicates a better classification performance. In the comparison results, the performance of the HOG algorithm indicates that it is not suitable for describing blurry features such as smoke. Some algorithms, such as NRLBP [18] and POEM [19], do not express smoke characteristics well. LTrP achieves a lower FPR, but its lower TPR means that it is not sufficiently suitable for describing smoke. The algorithm proposed in this paper achieves a higher level of TPR than the U2 and RI modes of PLBP [16] and LDP4 [21]. Some schemes, such as PRICoLBP [22] and LDP RIU2, are slightly better than our algorithm, but spend more time in the feature extraction process.

The CLBP RI [17] operator is superior to CSGC-LBP in terms of indicators, but it takes a longer time to compute. If accuracy is sufficiently high, CSGC-LBP is sufficient to meet smoke detection needs. The trisection feature fusion scheme combines the CSGC-LBP feature with the LPQ feature, and the obtained result is better than the results of independent features. Moreover, the combined features
perform well on the three sets of data, which also shows the effectiveness of the proposed scheme.

### B. EVALUATION OF THE VIDEO-BASED SMOKE-DETECTION FRAMEWORK

1) DATASET

In the evaluation, two types datasets are employed. An image dataset is used to acquire the model used for classification and compare the quality of the extracted features while a video dataset is used to measure the performance of the smoke-detection framework.

For the image dataset, we use the datasets of Yuan et al. and Lin et al. Because the ratios of positive and negative samples in these two datasets are not equal, we reassembled and filtered the images of the training set. Two new datasets were formed and uniformly scaled to $100 \times 100$ pixels. The number of images in set 1 is relatively small, while the number of images in set 2 is large. Samples from the image datasets are shown in Fig. 7. We employed the receiver operating characteristic (ROC) metric to evaluate algorithm performance.

For the smoke video dataset, we also included two sets of videos for testing. Although our proposed smoke-detection framework is suitable for high-definition video, most of the current smoke detection methods use smoke videos made by Bilkent University. However, these videos are low definition. To compare the proposed method with other algorithms, we upsampled the video to $1,280 \times 960$ pixels using bilinear interpolation. Lin et al. [12] contributed a video dataset that was taken by a high-definition camera during their research. We also evaluated the methods on this dataset. Note that although the image dataset was extracted from video, the video used to extract the image training set is not included in the videos used for testing.

The images in Fig. 8(1)–8(5) are smoke videos from Bilkent University. Videos 6–10 are high-definition videos with smoke from the China University of Science and Technology. Videos 11–16 are non-smoke high-definition videos from the China University of Science and Technology. We use these videos to compare several commonly used algorithms such as LBP U2 [13], CS-LBP, LBPV [23], and MBLBP [24].

2) FEATURE FUSION EXPERIMENTS

First, we converted the images in from Table 4 to grayscale. Then, we extracted the CSGC-LBP and LPQ features and calculated their histograms to form the spatial- and frequency-domain joint feature using the proposed feature fusion scheme. We then normalized the extracted features according

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Use</th>
<th>Smoke Images</th>
<th>Non-smoke Images</th>
<th>Total Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Training</td>
<td>550</td>
<td>777</td>
<td>1,327</td>
</tr>
<tr>
<td></td>
<td>Testing</td>
<td>688</td>
<td>777</td>
<td>1,465</td>
</tr>
<tr>
<td>2</td>
<td>Training</td>
<td>4,000</td>
<td>4,000</td>
<td>8,000</td>
</tr>
<tr>
<td></td>
<td>Testing</td>
<td>1,500</td>
<td>1,500</td>
<td>3,000</td>
</tr>
</tbody>
</table>
Table 5. Results of the methods for smoke detection in videos 1–5.

<table>
<thead>
<tr>
<th>Tested video</th>
<th>Number of frames</th>
<th>Number of smoke frames</th>
<th>MBLBP</th>
<th>LBPV</th>
<th>LBP U2</th>
<th>CS-LBP</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>TP (frame)</td>
<td>TPR (%)</td>
<td>TP (frame)</td>
<td>TPR (%)</td>
<td>TP (frame)</td>
</tr>
<tr>
<td>Video 1</td>
<td>900</td>
<td>821</td>
<td>744</td>
<td>90.6</td>
<td>755</td>
<td>92.0</td>
<td>817</td>
</tr>
<tr>
<td>Video 2</td>
<td>630</td>
<td>600</td>
<td>435</td>
<td>72.5</td>
<td>598</td>
<td>99.7</td>
<td>600</td>
</tr>
<tr>
<td>Video 3</td>
<td>1650</td>
<td>519</td>
<td>466</td>
<td>89.8</td>
<td>430</td>
<td>82.9</td>
<td>497</td>
</tr>
<tr>
<td>Video 4</td>
<td>900</td>
<td>900</td>
<td>864</td>
<td>96.0</td>
<td>900</td>
<td>100</td>
<td>900</td>
</tr>
<tr>
<td>Video 5</td>
<td>244</td>
<td>219</td>
<td>193</td>
<td>88.1</td>
<td>195</td>
<td>89.0</td>
<td>195</td>
</tr>
</tbody>
</table>

In the classification process, we used an SVM and with an RBF kernel. Then, we compared the proposed algorithm with fused features with other algorithms. We measured the TPR, FPR, dimension of feature vector, and ACC. In some studies, TPR is also called the detection rate and FPR is also called the false alarm rate. Higher dimensions in a feature vector mean that a longer time is spent in the SVM classification. It is obvious that during smoke detection, an excessive detection time is unacceptable. The experimental comparison algorithms are CS-LBP, LBPV [23], MBLBP [24], and LBP U2. We compared these algorithms using ROC curves.

As can be seen from the ROC curves, MBLBP and LBPV clearly do not recognize smoke, and CS-LBP performs only slightly better. The features extracted by CSGC-LBP and those extracted by LBP U2 lead to good smoke detection results. On large data sets, CSGC-LBP appears to perform better. The performance of LPQ in these two data sets demonstrates the more descriptive abilities of the frequency domain characteristics of smoke. When the CSGC-LBP and LPQ algorithms are using the trisection feature fusion scheme, we achieve the best classification performance in both data sets, which indicates that this classifier can be applied to smoke detection.

3) SMOKE VIDEO EXPERIMENTS

In Videos 1 to 5, smoke appears after the first frame. We compare the TPR indicator of algorithms like MBLBP, LBPV, LBP U2, CS-LBP, and the proposed method, which means the fusion feature composed of CSGC-LBP and LPQ here. It can be seen that in this framework, each algorithm achieves a relatively high TPR. Among them, the combination of CSGC-LBP and LPQ achieves a higher TPR. Although the prediction accuracy of each video cannot be measured by the average value, it can be seen in Table 5 that the algorithm with feature fusion has advantages over other algorithms. The specific detection result of each video is shown in Fig. 11.

In the second video dataset, Videos 6 to 10 are videos with white smoke (Fig. 13). Because the videos themselves are high definition, we did not pre-process them. It can be seen from Table 6 that in the smoke-detection framework proposed in this article, in addition to the unsatisfactory prediction of the individual algorithms in Video 9, these algorithms obtain a relatively high TPR. Moreover, the feature fusion scheme proposed in this paper still has superior performance on high-definition video.

Finally, the results of an experiment on high-definition non-smoke videos is presented in Fig. 14 and 15. In this experiment, we added the LBP RI and LBP RIU2 operators,
which are based on the basic LBP operator. The appearance of FPR is often a misjudgment of some image blocks. Although our smoke-detection framework reduces the workload of SVM after filtering the images, there are still some misjudgments. This is because the texture features formed by the image blocks after conversion to grayscale have a greater degree of similarity with smoke. Although the gray trousers in Fig. 15 do not correspond to smoke in RGB space, the texture features are roughly similar to those of smoke. Despite this, the proportion of false detections is still not very high, and this can be compensated for using an alarm mechanism with continuous frames. The misdetections of the LBP RIU2 and MBLBP operators are more serious. CS-LBP and LBP U2 are better than some algorithms, and our feature fusion scheme obtains the lowest FPR on these videos.
V. CONCLUSION
Smoke is often a precursor to fires, and the detection of smoke in videos is particularly important. With the popularity of high-definition cameras and videos, the performance of an original method is often not satisfactory. This paper presents a smoke-detection framework for high-definition video. In this framework, the ViBe algorithm is used to extract the motion regions, the number of regions to be classified is reduced using morphological processing and filtering of image blocks, and, finally, extracted small image blocks are fed to the SVM for classification.

During the design of the detection process, we improved the original texture feature description method and proposed a local texture description operator that considers the compensation of the surrounding pixels and the center gradient. In addition, we also combined the characteristics of spatial and frequency domains and proposed the trisection feature fusion scheme.

In the experiments that compared various texture description operators, the proposed texture extraction operator was found to be more consistent with smoke images than conventional features. The features merged using the proposed feature fusion scheme obtained the best accuracy. We evaluated the proposed method on different datasets and obtained high values for TPR and ACC. These experiments show that our smoke-detection framework has a strong performance in the detection of smoke in high-definition videos.

REFERENCES

ZHE LIU received the B.Eng. degree from the College of Communication Engineering, Jilin University, Jilin, China, in 2017, where he is currently pursuing the M.S. degree. His major research interests include image processing, computer vision, and pattern recognition.

XIAOPING YANG received the B.Eng. degree in detection technology and instrument, the M.S.E. degree in communication and electronic system, and the Ph.D. degree in control theory and control engineering from the Jilin University of Technology, Changchun, China, in 1985, 1988, and 2007, respectively, where she is currently a Professor with the College of Communication Engineering. She has been committed to teaching and research, including communication, signal, and information processing for years. Her current research interests include optimization control of data transmission, congestion control in wireless communication networks, and intelligent surveillance systems.
YING LIU received the B.Eng. degree in information engineering from the College of Communication Engineering, Jilin University, Jilin, China, in 2018, where she is currently pursuing the M.S. degree. Her major research interests include image processing and machine learning.

ZHIIHONG QIAN received the M.S. degree in communication and electronics systems from the Jilin University of Technology (JUT), in 1991, and the Ph.D. degree in communication and information systems from Jilin University, China, in 2001. He was a Visiting Researcher with the University of Massachusetts, USA, in 2001, The University of Texas, USA, in 2005, and Virginia Tech, USA, in 2006. He is currently a Professor of information and communication engineering with Jilin University and a Ph.D. Students Supervisor. He is the author of three monographs, has been granted 11 patents, has completed 27 research projects with his co-operators as a Principal Investigator, and authored or coauthored more than 170 research papers in National or International Academic Journals and Conferences. His research interests include wireless networks and communication systems, including key technologies of wireless sensor networks (WSN), the Internet of Things (IoT), the next generation mobile communication system and signal analysis, and the process of wireless network system. He serves on the Editorial Board for Journal of Communications, Journal of Electronics (China), China Communications, Acta Electronica Sinica, and Chinese Journal of Electronics. He serves as the Associate Editor-in-Chief of Journal of Electronics and Information Technology. He hosted ITA2013, SECS-2013, AICE2014, CNIS2015, CSMA 2015, ICNISC2016, CEIS2016, WCNA 2017, and CSAE2018 as the General Chair, and moreover, delivered keynote speech for national or international conferences respectively in the last 5 years.

* * *