Fire Technology

https://doi.org/10.1007/s10694-019-00831-x

Forest Fire Smoke Detection Based on Visual Smoke Root and Diffusion Model

*Yu Gao** and *Pengle Cheng**, *School of Technology, Beijing Forestry University, No. 35 East Tsinghua Road, Beijing 100083 Haidian, China*

Received: 3 September 2018/**Accepted:** 12 February 2019

Abstract. The damage caused by forest fire to forestry resources and economy is quite serious. As one of the most important characters of early forest fire, smoke is widely used as a signal of forest fire. In this paper, we propose a novel forest fire smoke detection method based on computer vision and diffusion model. Unlike the video-based methods that usually rely on image characters extraction, we try to find the shape of smoke that is at the generation stage. To combine vision and diffusion model together, the basic concept of smoke root is proposed. In the frame processing stage, none characters of fire smoke are extracted (like texture, color, frequency information etc.), and continuous frames are only used to extract stable points in dynamic areas as the smoke root candidate points. In the diffusion model simulation stage, all smoke root candidate points information is adopted by the model to generate the simulation smoke. Finally, the match algorithm based on color, dynamic areas and simulation smoke is implemented to get the final results. In order to reduce the complexity of computation, we ignored the simulation process of the smoke details, such as texture and turbulence, and only retained the contour features in two-dimensional form. Experiments show that under the condition of smoke root existing in the frames, the algorithm can obtain stable detection results and low false positive rate in cloudy scenes.

Keywords: Smoke detection, Navier–Stokes, Smoke root, Interactive modeling

1. Introduction

Monitoring and prevention of forest fires have already been matters of concern for a long time. The earliest monitoring method to protect the forest from wildfire depends on security patrol by forest managers. However, considering the characteristics of forests, wide and complicated, the manual patrol is not only high-cost, but also a potential threat to managers. The follow-up development of traditional methods, such as sensor distribution [15, 25] and large-scale sensors [19], still have defects in defense against large-scale forest fires. More consideration should be placed on many limitations, like power supply performance, signal transmission quality and the accuracy number of sensors.

* Correspondence should be addressed to: Yu Gao, E-mail: 1023743297@qq.com; Pengle Cheng, E-mail: chengpengle@bjfu.edu.cn

Published online: 28 February 2019

However, these defects have been significantly improved with the help of cameras and computer vision algorithms. On the one hand, the cameras placed on the high places, like a mountaintop, can extend their visual field to thousands of kilometers. On the other hand, the installation of cameras has lower cost and is more flexible than the method of large-scale sensors. Over the past decade, forest fire prevention based on image analysis has been developed rapidly, leading to the change of detection target from open flame to early smoke [13, 22, 28]. Therefore, the detection of smoke in the early stage of forest fire has become an important method to prevent the spread of fire.

1.1. Automatic Image Features Extraction and Classification

In recent years, with the rapid development of deep learning network, many network structures have been created for specific situations. Compared with traditional methods, deep learning nets can learn representations of date with multiple levels of abstraction and automatically learn the features [2]. Because of this advantage, Yin et al. [24] proposed a novel deep normalization and convolutional neural network (DNCNN) to extract smoke features and do classification. Tao et al. [17] improved the structure of AlexNet [10] avoiding complex functional image preprocessing, and successfully applied smoke recognition. Frizzi et al. [8] designed a dedicated 8 layers of CNN network that can detect both fire and smoke. However, given the fact that deep learning network requires enormous training sets, it is a serious problem that we cannot gain sufficient enough training data considering the particularity of forest fire. Actually the training sets used in [24, 26] have been processed by data enhancement techniques.

1.2. Network Classification Based on Manual Features Extraction

The common smoke detection architecture is composed of image feature extraction and classification machines. Prema et al. [5] proposed a method based on static and dynamic texture characters. Firstly, color information is used to segment the flame candidate region. Secondly, hybrid texture descriptors, 2D wavelet decomposition, 3D volumetric wavelet decomposition are adopted to obtain the texture features. Finally, Extreme Learning Machine(ELM) is used to classify the candidate flame regions. Pundir and Raman [13] computed the LECoPs to find the texture and intensity, and also used color space RGB and YCbCr to get color features. After that, DBN is used for classification of smoke and non-smoke pixels. Tung TX and Kim JM [18] presented a four-stage smoke-detection method, Firstly, an approximate median method is used to segment moving regions in a video frame. Secondly, a fuzzy c-means (FCM) method is used to cluster candidate smoke regions from these moving regions. In the third phase, a parameter extraction method is used to extract a set of parameters from spatial and temporal characteristics of the candidate smoke regions. Finally, the parameters are used as input feature vectors to train a support vector machine (SVM) classifier, which is then used by the smoke alarm to make a decision.

1.3. Smoke Detection Only Based on Features

The smoke area can be extracted by feature extraction and recognition algorithms designated for specific features. Han et al. [9] adopted Gaussian Mixture Model to subtract dynamic areas, and multi-color-based detection combining the RGB, HSI and YUV color space is employed to obtain possible fire regions. After that, the results of the two steps are combined to make the final decision. Zhou et al. [29] used Maximally Stable Extremal Region (MSER) detection method to extract local extremal regions of the smoke, these potential regions are tracked by searching the best-matched extremal regions in the subsequent frames. Then, a novel cumulated region approach, which can be effectively used to identify the distinctive expanding and rising motions of smoke, is implemented to give the final results. Ye et al. [23] presented a method can detect smoke and flame simultaneously. In the process of dynamic region extraction, the adaptive background subtraction is used. The optical flow-based movement estimation is applied to identify a chaotic motion. Finally, the spatial and temporal wavelet analysis, Weber contrast analysis and color segmentation are implemented to verify the real fire and smoke.

The main contributions of this paper includes these two parts:

1. Combining the computer vision technology with the essential physical characteristics of the smoke to decide whether the candidate smoke areas follow the physical principles.
2. The basic algorithm and concept of "smoke root" are proposed, which is also an entrance to apply Navier–Stokes equations. In this method, "smoke root" is the media and connection between computer vision and fluid mechanics.

The remainder of this paper is organized as follows. The details of our approach and contributions will be given in Sect. 2. Section 3 shows experimental results. The last section contains conclusions and discussions.

2. Proposed Method

The process of smoke detection method is shown in Algorithm 1. In the initialization step, the background of Vibe algorithm will be established, and five adjacent frames are needed to generate five adjacent skeleton images. *Cand_Root* is the information of merged skeleton endpoints position. During the detection phase, one frame is pushed into the queue at one loop to update *InitQueue*, which is a chain table of length 5. *Roots_Detection*() is a function designed for extracting smoke roots from candidate roots information. According to the results of *Roots*, different decisions will be executed. The first one skips the smoke simulation process and updates the chain table directly when the *Roots* is empty. The second decision is to perform the simulation process and make the final decision, after that, the chain table will be updated. As for physics model, a two-dimensional smoke diffusion model based on Navier–Stokes equations is applied in function *Smoke_Simulation*().

Algorithm 1 *SmokeRootDetection*

```
t = 0
i = 4
InitQueue = [];
Vibe_Init();
for t = 0 → 4 do
    Image_Temp = Load_Frame_Sequence(frame_t);
    Skeleton_image_t = Process_Image(Image_Temp);
    Cand_Roots_t = Candidate_Roots_Detection(Skeleton_image_t);
    InitQueue.append(Cand_Roots_t);
    t++;
end for
while (Start = True) do
    i++;
    Roots = Roots_Detection(InitQueue);
    if Roots == None then
        Image_Temp = Load_Frame_Sequence(frame_i);
        Skeleton_image_i = Process_Image(Image_Temp);
        Cand_Roots_i = Candidate_Roots_Detection(Skeleton_image_i);
        InitQueue.pop(InitQueue[0]);
        InitQueue.append(Cand_Roots_i);
    else
        Create_Computation_Domainn(Roots);
        Sim_Arrary = Smoke_Simulation(Roots);
        Simulation_Original_Match(Sim_Arrary);
        Image_Temp = Load_Frame_Sequence(frame_i);
        Skeleton_image_i = Process_Image(Image_Temp);
        Cand_Roots_i = Candidate_Roots_Detection(Skeleton_image_i);
        InitQueue.pop(InitQueue[0]);
        InitQueue.append(Cand_Roots_i);
    end if
end while
```

2.1. Dynamic Feature Extraction

The method assumes that the forest fire surveillance camera is stationary. ViBe algorithm [1] is adopted for background subtraction. The main idea of ViBe is to establish the background model for each pixel. Each pixel has N samples which are randomly picked from the eight-neighbor pixels. By comparing to the samples in collection, the new pixel belongs to background or not will be decided.

The first step of ViBe is to initiate the background model for each pixel. $M(x)$ in Eq. (1) indicates the model, which contains N samples picked from the eight-neighbor of the pixel randomly. v_i represents the pixel value of each sample.

$$M(x) = \{v_1, v_2, \ldots, v_n\} \tag{1}$$

The second step is to detect the background information. The Euclidean distance between New pixel value and N samples will be calculated and make a comparison to the stable threshold R. If the distance is smaller than R, the number of approximate sample points will increase. A pixel belongs to the background if this number bigger than Min. This process is shown in Eq. (2).

$$\{S_R(v(x)) \cap \{v_1, v_2, \ldots v_N\}\} \geq Min \tag{2}$$

The last step is to update the background model from time to time. ViBe mainly follows 2 updating rules:

(1) If a pixel is determined as a moving target for a long time, there must be an error occurring before. This pixel will be turned into background.
(2) If a pixel is considered as background, there is a possibility of $1/\varphi$ to update the collection it correspond to.

φ is the updating rate which determined the life period of the samples in the collection.

2.2. Morphology Processing

Under the situation that the area of noisy points is extremely smaller than large connected areas, the close morphological operation [20] is used to fill small noisy holes, connect adjacent objects and smooth the boundary. The essential process of the closed operation is the dilation and followed by erosion. The dilation and erosion kernels used in this paper are all rectangle with a range of 13×13 for dilation and 7×7 for erosion.

2.3. Connected Domain Skeletal Extraction

In order to generate smoke root nodes coordinates information, the skeleton images of connected areas can be adopted. The endpoints of skeleton parts can be defined as the basic position information for smoke root. With the single pixel link characteristics and only two kinds of color information, the corner point determination algorithm on skeleton images can be simple and efficient.

The algorithm of skeleton extraction used in this method is proposed by Zhang and Suen [27]. The main idea is to perform iterative erosion operation on the pixels that meet certain conditions. Four conditions are proposed to choose the pixels eroded, seen Eq. (3) and Fig. 1.

(6) $C6_{value}$	(7) $C7_{value}$	(8) $C8_{value}$
(5) $C5_{value}$	$I_{(x,y)}$	(1) $C1_{value}$
(4) $C4_{value}$	(3) $C3_{value}$	(2) $C2_{value}$

Figure 1. 8-Neighbor pixels.

$$\begin{cases} 2 \le B(I) \le 6 \\ A(I) = 1 \\ C7_{value} * C1_{value} * C3_{value} = 0 \\ C1_{value} * C3_{value} * C5_{value} = 0 \end{cases} \tag{3}$$

The first condition in Eq. (3) means the sum number of target pixels (value 1 in binary image) around I (Fig. 1) is between 2 and 6. The second condition represents the frequency that the value of adjacent pixels turn from 0 to 1 in the clockwise direction. As for Ci_{value} in condition 3 and 4 represents the pixel value in binary image (0 or 1).

2.4. Smoke Root Candidate Points Extraction

The main function of the smoke root is to distinguish the disturbances, such as clouds and fog. In natural phenomena, although the clouds and fog drift in the sky, these disturbances have no source, which is the biggest difference with the forest fire smoke. The extraction process of the root nodes can eliminate these interferences to some extent.

Considering the small diffusion range and relatively stable position of the combustion source in the early stage of the fire, it can be assumed that the source position of flame smoke is relatively fixed in the continuous frames. As the result, we define this relatively stationary position of combustion as "smoke root" in this paper.

As one of the most important elements to determine the initial conditions of Navier–Stokes equations, smoke root information directly affects the numerical calculation and smoke direction prediction of physical model. In addition, the smoke root information is also an important media between computer vision and fluid mechanics.

2.4.1. Skeleton Image Endpoints Position Extraction and Merging From the red circle area of image Fig. 2, it is apparent that the smoke root node is produced by the endpoints of the skeleton image. Since skeleton images are binary images, the endpoints extraction can be easily determined by the color information. Color distribution is shown in Fig. 1.

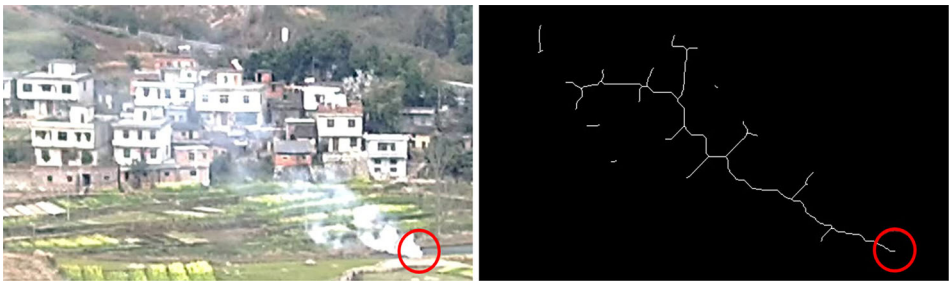

Figure 2. Smoke root in RGB and Skeleton image.

Where $I_{(x,y)}$ is the skeleton point in current operation frame. Ci_{value}, $(i = 1, 2, 3, 4, 5, 6, 7, 8)$ is the value of pixel i in clockwise direction, and the range of Ci_{value} is 0–255. Equation (4) indicates that when $I_{flag}(x, y)$ is equal to 1, $I_{(x,y)}$ can be inferred to be a skeleton endpoint.

$$I_{flag}(x,y) = \begin{cases} 1, & \sum_{i=0}^{8} Ci_{(x,y)} = 255 \\ 0, & else \end{cases} \tag{4}$$

To avoid the appearance of more than one smoke root at the same diffusion calculation plane, the merging of adjacent endpoints is required. Merging may lead to the loss of details in the simulated smoke shape results, but compared with particles distribution and visual effect of the details, the involvement of velocity distribution in the following computations is more important and thus the details losing can be completely accepted.

Center merging, which can be carried out by replacing two adjacent endpoints with the midpoint of the line connecting these two endpoints, is used to merge adjacent endpoints. In addition, when dealing with the merging of more than two endpoints, iterative merging is performed, which indicates only two endpoints are involved in each merging phase. One thing to note is that the result of each merging phase also carries over to the following merging phase. The coordinates of merged endpoints can be calculated by the Eq. (5).

$$\begin{cases} CP_x = (A_x + B_x)/2 \\ CP_y = (A_y + B_y)/2 \end{cases} \tag{5}$$

CP_x and CP_y are the horizontal and vertical coordinate values of a merged endpoint, A_x and A_y are the horizontal and vertical coordinate values of the endpoint A, B_x and B_y are the horizontal and vertical coordinate values of the endpoint B.

2.4.2. Template Building of Search Strategy Information from the merged endpoints is used to extract the candidate smoke roots. In the process of smoke roots extraction, adjacent frame information is required. Smoke root position remains

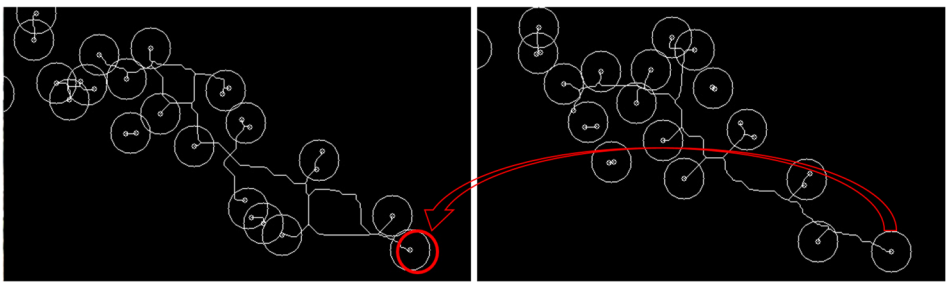

Figure 3. Smoke roots floating.

stable in the adjacent frames, so the basic idea of root extraction is to look for merged endpoints with the same position in the adjacent frames. However, the volatility of computations using the skeleton image extraction algorithm and dynamic feature extraction algorithm possibly make the result of the similar point in different computations fluctuate lightly, which is shown in Fig. 3. Statistical rules will be applied in this paper to solve the problem.

The red circle in the left frame of Fig. 3 represents the smoke root position in the frame on the right. It is obvious that smoke roots in different frames do not completely coincide with one another. Therefore, the corrective statistics of the smoke roots position is the essential operation in this step.

First, five skeleton images need to be converted into merged endpoint position information. Second, all the information will be restored on an empty image template that is the same size as the frames. Finally, the template marked with position information is used to select candidate smoke roots by executing the search strategy.

When it comes to identifying matching nodes in different frames, the option is to turn to the positional similarity principle. This is because there are no other principles or information in binary images to indicate matching nodes, such as the gray scale invariance principle in the L–K optical flow method [12]. Inspired by DoG in the SIFT algorithm [11], the restored process of merged endpoints is shown in Fig. 4.

The white sheets represent five continuous frames, and the black points in these sheets represent the merged endpoints position information. Information on all 5 frames will be restored in a single channel template that has pixels all with values are zero. As for the points in the template, different gray values are used to mark them. The range of the template pixel value is between 0 and 255 and each merged point in the frames will increase the value of corresponding coordinate points by 51 (255 max value/5 frames = 51). If there is more than one merged points in the same position, the operation of accumulation will be performed.

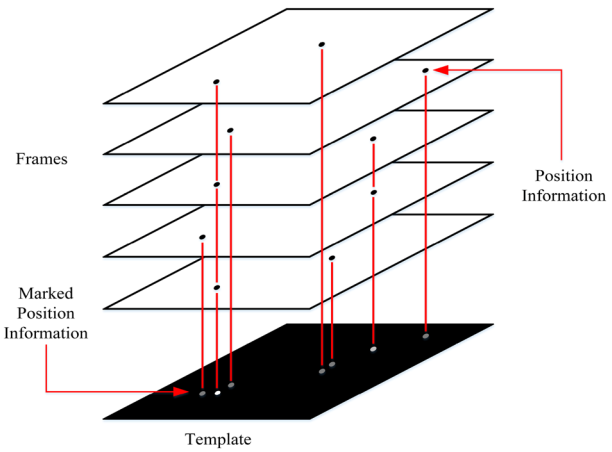

Figure 4. The restored process of merged endpoints.

According to this method, the pixel values of the three points on the left side in the template shown in Fig. 4 are 51, 153, and 51. When all merged points in the five frames are counted, the template map can be used to implement the search strategy.

2.4.3. Extraction of Smoke Root Candidate Points The searching process is done in two steps. In the first step, all pixels in the template with values above 102 are chosen (153, 204 and 255). The pixels with a value of 153 are used as the center of the circle and *Rd* is set as the radius to perform circular searching. If there are 2 non-zeros pixels within the circumference, this center will be classified as the "smoke root". If the number of rounding points is 0 or 1, the center will be defined as the "pseudo smoke root". For pixels with a value of 204, we do the same operation as was done with the pixels with a value of 153, but the difference is that there is a condition of rounding pixels. If there is 1 non-zero pixel within the circumference, this center will be classified as the "smoke root". If the number is equal to 0, the center will be defined as a "pseudo smoke root". All pixels wih a value of 255, will be directly classified as a "smoke root". "Pseudo smoke root" has the same function as "smoke root" during the simulation process, and the only difference is that if the same "pseudo smoke root" appears more than a certain number of times (*TiT*) in continuous frames, it will be upgraded to a "smoke root". "Pseudo smoke roots" have no elimination mechanism. The first step of the searching strategy can be seen in Fig. 5.

"Pseudo smoke root" is used because the skeleton extraction algorithm can not provide a precise measurement of the range of fluctuations, which means the merged endpoints will miss detection to some extent. On the other hand, if the results of dynamic region extraction are not completely continuous, it may lead to the fracture of skeleton images. Because the simulation process of using diffusion models has a relationship with time and begins with the condition of no-smoke, if the simulation process is interrupted by missed detection of roots, all simulation results will be abandoned and will start again. Therefore, it is important to provide stable smoke root information.

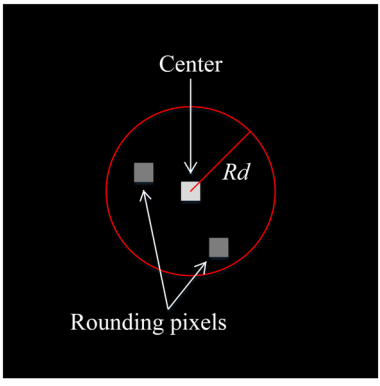

Figure 5. The first step of searching strategy.

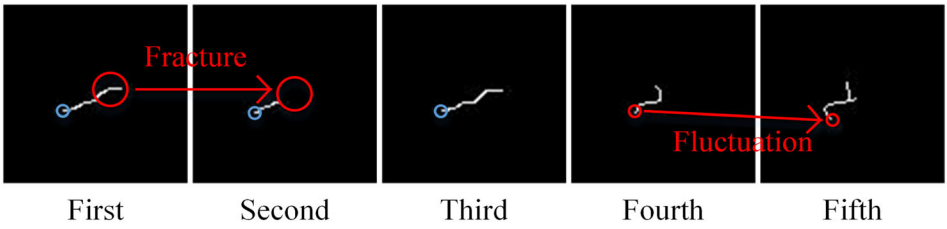

Figure 6. Fluctuation and fracture with frames.

As shown in Fig. 6, there are stable merged endpoints (blue circle) in the first, second, third, and fourth frames, while the fifth frame loses the merged points because of skeleton fluctuations. But this merged point is the root that we need. The fracture happens in the third frame, although it has no effect on the results.

The second step of the search process is also based on the template. When the first step is finished, all smoke roots and pseudo smoke roots with their rounded pixels values will be set to zero. The main purpose of this operation is to erase the pixel information that has been used in the first step. The remaining pixels whose value are not zero will be treated as the center to execute the circle searching operation mentioned in the last step (Fig. 5), and if all pixel values within the radius (including center point itself) are more than 153 (204, 255), these merged endpoints would be classified as ''pseudo smoke roots''. ''Smoke roots'' will not be defined in this step. Another difference from first step is that the involved pixel values are set to zero within the radius after successful detection of the pseudo smoke root in each search process. The points that failed to become roots, will not be operated on.

2.5. *Determination of Smoke Regions Based on Navier–Stokes Equation*

For each smoke root r, computing domain DOM_R needs to be established. A two-dimensional space is adopted in the Navier–Stokes equation. Although the proposed method try to simulation the shape of smoke, the accuracy of details and clarity of textures are not required. Because the velocity distribution information is sufficient to obtain the final results. Due to this implication, complicated phenomena, such as turbulence, are not considered in the calculation process.

2.5.1. Establishment of Computation Domain The number of computation domain is decided by the number of smoke roots. As for each candidate root, the proposed algorithm will establish a two-dimensional Descartes coordinate system, which origin is the current candidate smoke root node. The smoke generated by the combustion source is driven by the rise of hot gas, which makes the smoke root lower than actual smoke. However, different angles between camera and real smoke may break this rule. For example, the smoke existing in video material ''TEST17'' is distributed under the smoke root. Therefore, all four quadrants need to be considered, and then the appropriate simulation domain will be made to

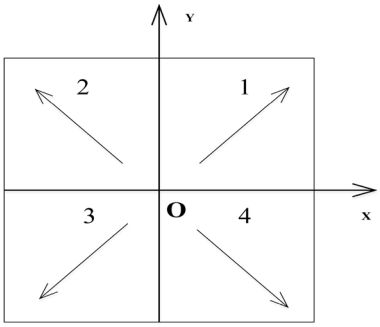

Figure 7. Rules for establishing the computation domain.

implement simulation computation. The rules for establishing the computation domain are shown in Fig. 7.

Where origin O is the position of candidate smoke root. The arrow in each quadrant represents the direction of real floating smoke, and the computation quadrant is determined according to the direction of the smoke. For example, the second quadrant is adopted as the computation domain for the smoke in Fig. 2. As for the smoke floating around the X-axis or the Y-axis, according to the direction of the skeleton distribution, the two quadrants are combined and selected as the computation domain. For example, the computation domain for the smoke, which is in the second row and the third column in Fig. 8, is selected as the first and second quadrants.

2.5.2. The Initial Velocity of the Computation Domain Using the principles shown in Sect. 2.5.1, we try to combine the computation domain selection process with

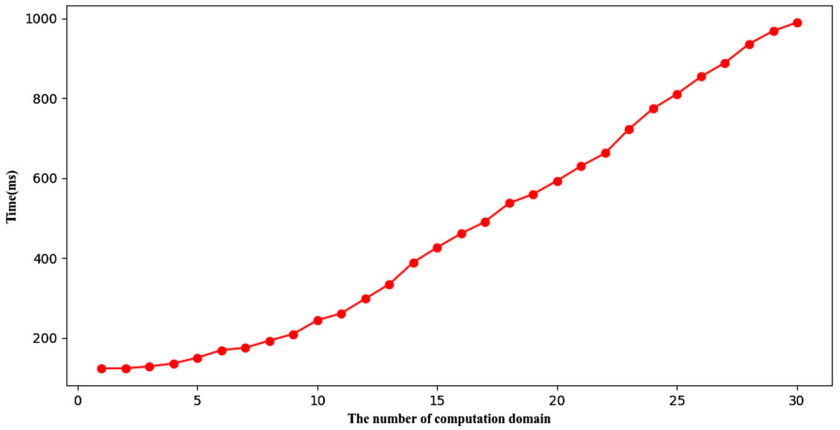

Figure 8. The relationship between the number of computation domain and simulation time cost.

settlement of initial velocity to perform a unified calculation. By completing these two tasks at the same time, redundant computations can be significantly reduced.

Although the smoke distribution in the different position of the image has different actual wind speed, the shape of the smoke is relatively stable, so we select the virtual velocity according to the shape of smoke. In the computation domain, we assume the horizontal velocity of all points are the same. The maximum velocity given in an axis direction is 1 and the minimum value is 0. The more parallel to the Y axis the shape of the smoke skeleton part is, the more horizontal velocity moves closer to 0. As such, the more parallel to the X axis the shape of the smoke skeleton is, the more horizontal velocity moves closer to 1. The data from each computation domain is independent. The initial velocity is calculated by Eq. (6), where V_x is horizontal velocity, and θ is the clockwise angle between the skeleton part and the X axis.

$$V_x = -\frac{2}{\pi}\theta + 1 \tag{6}$$

We only set the vertical velocity for the points that are in the skeleton part, while setting the vertical velocity to 0 for the points that are outside the skeleton. Skeleton points were chosen as vertical velocity bearing points because the skeleton information must be included in the dynamic region of the smoke extracted from the continuous frame image. The initial velocity value in the vertical direction of the velocity field is 0. The velocity calculation is simulated by the method used in [16], but the difference is that the dimension adopted is only two-dimension. Influential factors of vertical velocity are calculated by adding a simple linear function of density and temperature in the vertical direction. The equation is as follows:

$$F_{buoy} = -\alpha z + \beta(T - T_{amb})z \tag{7}$$

$z = (0, 1)$ is the direction vector pointing to the vertical direction. T_{amb} is the ambient temperature of the air, and α and β are two positive constants with appropriate units to ensure Eq. (7) is physically meaningful. What is different from [16] is that external force is not added to the global computation domain but to the skeleton points. The method assumes that the virtual smoke density does not change when it is transmitted between grids, because they have the same temperature information at the same Y coordinates.

2.5.3. Navier–Stokes Basic Equations Navier–Stokes equations are the basic equations describing the motion of an incompressible fluid [3]. Low-speed flowing air can be treated as an incompressible fluid, because the polymerization of the combustion particles is not considered, so it meets the application requirements of the Navier–Stokes equations. The specific equations are as follows:

$$\begin{cases} \frac{\partial u}{\partial t} = -(u \cdot \nabla)u - \frac{1}{\rho}\nabla p + \gamma \nabla^2 u + f \\ \nabla \cdot u = 0 \end{cases} \tag{8}$$

In equation set (8), the first equation is the momentum equation, which is used to ensure the conservation of momentum. The second equation is the continuity equation, ensuring the conservation of mass. u is the term of velocity, p is pressure, ρ is the density of the fluid, γ is viscosity coefficient, and f is external force. Momentum equation in a two-dimensional coordinate system is as follows:

$$\begin{cases} \frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} - v\frac{\partial u}{\partial y} - \frac{1}{\rho}\frac{\partial p}{\partial x} + \gamma\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) + f \\ \frac{\partial v}{\partial t} = -u\frac{\partial v}{\partial x} - v\frac{\partial v}{\partial y} - \frac{1}{\rho}\frac{\partial p}{\partial y} + \gamma\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) + f \end{cases} \tag{9}$$

In equation set (9), u is the horizontal component of velocity, and v is the vertical component of velocity. In the calculation process, density is set to 1. In order to solve the pressure term, the Poisson equation required is formed as follows:

$$\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} = -\rho\left(\frac{\partial u}{\partial x}\frac{\partial u}{\partial x} + 2\frac{\partial u}{\partial y}\frac{\partial v}{\partial x} + \frac{\partial v}{\partial y}\frac{\partial v}{\partial y}\right) \tag{10}$$

The position, on which the velocity given to grids are set, mentioned in [7], is set in the center of the grid center. During the calculation of each round, the smoke source is set as the origin and will diffuse with the velocity field to form simulated smoke. The method for solving the Poisson equation is an iterative method. For the discretization of Navier–Stokes equations, the method proposed by Fedkiw et al. [6] is adopted. Boundary conditions of the Poisson equation are the pure Neumann boundary condition.

$$\begin{cases} \left.\frac{\partial p}{\partial x}\right|_{x=1} = 0 \\ \left.\frac{\partial p}{\partial x}\right|_{x=40} = 0 \\ \left.\frac{\partial p}{\partial y}\right|_{y=1} = 0 \\ \left.\frac{\partial p}{\partial x}\right|_{y=40} = 0 \end{cases} \tag{11}$$

The boundary conditions for velocity are set as the followings:

$$\begin{cases} u_{(i,0)} = u_{(i,1)} \\ u_{(i,40)} = u_{(i,1)} \end{cases} \tag{12}$$

$$\begin{cases} v_{(0,i)} = v_{(1,i)} \\ v_{(40,i)} = v_{(41,i)} \end{cases} \tag{13}$$

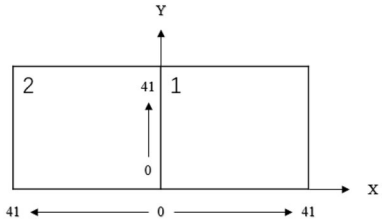

Figure 9. Index for the first quadrant.

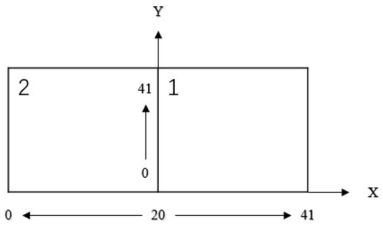

Figure 10. Index for both the first and the second quadrant.

In (10) and (11), i represents the number of rows and j represents the number of columns. The index number order also differs depending on the quadrants. For example, if the computation domain is the first quadrant, the index number is shown in Fig. 9. If the domain is the first and second quadrant combined, the index is shown in Fig. 10.

During the generation of simulated images, the smoke source is initially setup at coordinate origin (0, 0), follows the velocity field to perform iso-density transmission. When the virtual smoke reaches the boundary, the computational information exceeding the boundary will be abandoned directly.

After getting the simulation image, the color information is used as the final judgment condition to get the smoke region in the RGB image. The grids in the computational simulation domain correspond to pixels of the RGB image intercepted region one by one. If more than 70% of pixels belonging to the smoke distribution region have their counterparts in the RGB image and dynamic region image. If all of the counterparts three channels' pixel values are over a fixed threshold in RGB image, these pixels are determined to be part of the real smoke region.

3. Results and Discussion

3.1. Environment and Database

The video acquisition frequency is 25 frames per second. The program is implemented using C++ in Visual Studio 2013 and OpenCV 3.1.0 on PC. In order to visualize the predicted image based on the smoke eigenvalue generated in the pro-

cess, OpenGL is selected. The CPU used in our PC is the Intel Core i7-5700HQ, while the GPU used is the GTX960M.

The input video materials that we use in the experiments have already been uploaded to the https://pan.baidu.com/s/1Z7GFX9wGDbSelTIhhZm6Kw. In order to evaluate the effectiveness of the proposed method, we also use the data base from the website http://signal.ee.bilkent.edu.tr/VisiFire/Demo/FireClips. There are four folders in our database: Smoke1 folder is provided by Qingdao Haohai Network Technology JointStock Co.,LTD, Smoke2 folder is made by ourselves using the iPhone6, and Smoke3 folder contains the same forest smoke materials as the "ForestSmoke" folder in the above mentioned website. The fourth folder contains 22 videos that are the formatted materials used in the test process with a resolution of 480×320, and a frame rate of 25 fps.

3.2. Parameter Settings

3.2.1. Dynamic Region Extraction The parameters of the Vibe algorithm are determined experimentally. The Number of pixel samples N is 15, threshold Min is 2, the default search radius of pixel value R is 20, and the probability of random sample φ is 16. Because the parameters will affect the sensitivity of the algorithm to dynamic regions, the parameters mentioned above are the values that we determined during the test process to ensure the dynamic regions can be detected. The algorithm we propose allows multiple smoke roots, even it takes more time. But the persistent missing detection will lead to errors in the results. Regarding this concern, the parameters used have to ensure the algorithm sensitive to dynamic regions.

3.2.2. Morphology and Smoke Root Extraction Dynamic regions determined by Vibe are dot matrices composed of pixels, which can not guarantee the continuity of a wide range, so the size of dilation kernels are much bigger than erosion kernels to ensure that the dot pixels can form a connected domain. Although this work can be handled by the Vibe+ algorithm [4] and has a better effect, the processing speed of Vibe+ is much lower than Vibe, which is why we did not adopt it. Besides, during the image processing stage, the algorithm we proposed does not involve detailed computations of items such as texture, so the dilation and erosion operation will not affect the results.

During the process of smoke root extraction, there are two main parameters that need to be adjusted. The first one is the merging radius of skeleton endpoints in Sect. 2.4.1, and the second one is the searching radius Rd of smoke roots in Sect. 2.4.2. The merging process of endpoints only make sense when the two parameters are equal, and the main factor that determines parameters is the resolution of the image. The larger the parameters are, the more obvious fluctuating phenomenon of merged endpoints will be. If the parameters are too small, the skeleton endpoints will be incompletely merged, and the smoke root detection results will be discontinuous between frames. In our experiments, the parameters are set to 5 pixels. TiT, which is also an empirical value, is set to 10.

3.2.3. Diffusion Model The diffusion model computation process requires fixed-size computation areas to generate simulated smoke. The main factors determining the size of the areas are the resolution of the frames and the number of pixels occupied by the smoke in the frames. Under the premise that forest fires are of a similar scale, the number of pixels occupied by smoke that far from the camera is less than those occupied by smoke close to the camera. Therefore, in the simulation process, it is necessary to ensure that the size of the computation areas can contain the roots of the nearby smoke as much as possible.

The size of the computation domain affects calculation and accuracy of the algorithm directly. The larger the computation domain is, the more accurate the final result and the distribution information of the smoke area will be. But the cost of computation will increase greatly. The range of the computation domain is also affected by the number of candidate smoke root regions. The range of computation domain is 40×40 pixels. Because of the limitations of boundary conditions, the computing scale is set to 42×42 pixels. The pixels around the region are not included in the reference range of the simulated image.

During the final matching computation process, the RGB color threshold that we chose for TEST1, TEST3, and TEST11 is 180, while the others had a threshold of 200.

3.3. Performance Evaluation and Comparison

3.3.1. Image Processing Stage In this stage, we focus on the process of obtaining the coordinates generated by smoke root detection algorithm from the original video materials. The total number of videos we used in the experiments is 22, which is detailed description in Sect. 3.1.

The first evaluating indicator is the time cost of image process, the average processing time of each frame per video (250 frames total in each video) is adopted as the evaluation standard. We also list the max and the min time for each video as reference. The result is shown in Table 1.

The second evaluating indicator is the accuracy of the smoke root detection algorithm. We first set the artificial Region Of Interest (ROI) in the smoke root region of the image, which is similar to the target marking process in the creation of the deep learning database. The difference between the two processes is that the ROI we set will not be marked (red box) in the frames, it will just be an item of information stored in the chain table. Each result of smoke root coordinates extracted from each frame is compared to the ROI information. If the ROI exists in the current frame and smoke roots (any number except zero) are existing in the ROI at the same time, successful detection is made. For different video materials, the size of the ROI is also different. The ROI shape that we have used in all materials is rectangle. The specific size of the ROI and detection rate are shown in Table 2 and the example of the ROI setting is shown in Fig. 11. The background model initialization process of the Vibe algorithm is not included in the 250 frame test results.

Table 1
Time Cost in Frame Processing Stage

	Time (ms/frame)				Time (ms/frame)		
Videos	Ave	Max	Min	Videos	Ave	Max	Min
TEST01	27.95	59.81	27.02	TEST12	36.06	54.57	31.37
TEST02	28.63	66.34	27.07	TEST13	33.87	48.90	31.15
TEST03	27.85	46.09	27.06	TEST14	35.68	97.73	26.79
TEST04	29.89	71.56	27.36	TEST15	38.16	90.76	26.19
TEST05	32.91	51.92	27.51	TEST16	38.52	107.63	26.67
TEST06	32.19	90.54	27.99	TEST17	35.78	70.77	27.19
TEST07	33.33	58.56	30.01	TEST18	29.40	70.24	26.74
TEST08	31.52	56.51	28.86	TEST19	33.83	72.98	26.73
TEST09	33.64	60.96	28.76	TEST20	27.75	55.08	26.57
TEST10	36.16	88.46	31.77	TEST21	27.49	40.01	26.45
TEST11	29.06	42.88	27.61	TEST22	32.38	81.60	27.84

Table 2
Smoke Roots Detection Accuracy

Videos	Smoke/ROI (frames)	Positive detection (frames)	Detection rate (%)	ROI size (pixels × pixels)
TEST01	250	227	90.80	9 × 7
TEST02	250	241	96.40	10 × 10
TEST03	250	172	68.80	10 × 10
TEST04	250	247	98.80	10 × 10
TEST05	250	233	93.20	30 × 30
TEST06	250	249	99.60	10 × 30
TEST07	250	237	94.80	10 × 10
TEST08	250	228	91.20	30 × 30
TEST09	250	205	82.00	10 × 10
TEST10	250	192	76.80	40 × 30
TEST11	250	147	58.80	20 × 20
TEST12	250	247	98.80	15 × 15
TEST13	250	235	94.00	10 × 10
TEST14	250	231	92.40	10 × 10
TEST15	250	238	95.20	10 × 10
TEST16	250	238	95.20	15 × 15
TEST17	250	179	71.60	20 × 20
TEST18	250	167	66.80	15 × 15
TEST19	250	242	96.80	10 × 10
TEST20	250	250	100.00	10 × 10
TEST21	250	240	96.00	10 × 10
TEST22	250	240	96.00	15 × 15

3.3.2. Simulation Stage Time cost and simulation accuracy of the computation are a concern in this section. The simulation time cost is mainly determined by the number of smoke root nodes and the size of the computation domain area. Since

Figure 11. Example of ROI setting.

the computation domain area of all materials is the same, we first consider the impact of the roots number. The average simulation time of each frame is adopted as the evaluation standard and we also give the longest and the shortest simulation time for each video. The corresponding number of smoke root nodes is listed in Table 3.

We also determine the relationship between the number of computation domains and time cost. All computation domains that we used have the same condition as those used in the simulation process (40×40, including boundary conditions, iteration times, etc). The velocity data is taken from the ''.txt'' documents, which have a velocity matrix with a size of 42×42. The velocity data is given from 0 to 1 randomly and the final result is shown in Fig. 11.

In order to examine the simulation results more intuitively, we show the graphs in the simulation process for the six scenes mentioned in Fig. 11 respectively. The graphs shown in Fig. 12 represents the corresponding simulation results of the first line in Fig. 11, and all three scenes are taken from the public website mentioned in Sect. 3.1. In Fig. 12, the graphs in first line represent the computation domain and the skeleton parts in the smoke root region, and the second line represents the visualization results of smoke captured during the simulation process. Figure 13 manifest the results for the second line in Fig. 11, which is taken from our own database.

The standard for evaluating the accuracy of the simulation are based on the overlap rate, which is mentioned in the final position of Sect. 2.5.3. When the coverage exceeds the set threshold, the current computation domain is determined as the smoke area. The comparison results of different overlap rate thresholds are depicted in Table 4. The True Positive Rate (TPR) is the rate of how often real smoke is correctly detected as smoke, the False Positive Rate (FPR) is the rate of

Table 3
The Time Cost in Simulation Stage

Videos	Ave time (ms/frame)	Max time (ms/frame)	Min time (ms/frame)	Roots number (Max)	Roots number (Min)
TEST01	193.17	244.89	117.44	9	1
TEST02	508.21	644.83	121.21	21	2
TEST03	277.74	310.11	120.17	12	2
TEST04	146.09	194.68	109.91	6	1
TEST05	708.22	766.32	144.54	23	1
TEST06	738.26	840.27	200.20	26	7
TEST07	799.65	975.13	277.85	27	12
TEST08	383.41	406.42	166.83	14	6
TEST09	401.19	434.21	162.69	15	5
TEST10	681.89	826.82	170.28	26	5
TEST11	160.51	204.90	130.39	8	1
TEST12	355.18	432.53	292.67	14	10
TEST13	348.33	488.24	160.11	15	4
TEST14	148.30	175.69	125.43	6	2
TEST15	152.03	180.06	112.69	6	1
TEST16	137.21	154.98	119.71	4	1
TEST17	188.85	215.24	126.25	8	2
TEST18	188.86	247.34	151.04	10	4
TEST19	148.92	151.78	122.17	4	1
TEST20	118.80	123.20	116.78	3	1
TEST21	140.22	146.71	114.12	4	1
TEST22	266.67	318.35	185.97	12	7

how often non smoke is detected as smoke, and OLP represents the threshold of overlap rate.

3.3.3. Evaluation of the Complete Method The method that we propose consists of two parts and the corresponding evaluation standard have been given in the previous section. In this section, we mainly discuss the particularity of our method. Compared with vision-based algorithms, our method cannot be applied to all scenes that contain smoke. For example, a frame made entirely of smoke without root node information cannot be successfully detected. Our method is based on the forest fire smoke detection process, which means that there must be a combustion source(s) or a similar smoke-generating object (like a chimney) in the frames. We also list some situations in which our algorithm is ineffective in Fig. 14 and the reason why is that no stable root exists in the pictures.

Besides the image processing stage and the simulation stage, the complete algorithm also has additional procedures that ensure the connection between these two stage allow for the production of the final results. These parts also should be considered in the overall processing time of the method. Time cost of complete algorithm is shown in Table 5.

In order to test the effectiveness of the proposed smoke detection method, the performance is compared with those of three state-of-art algorithms. The first one

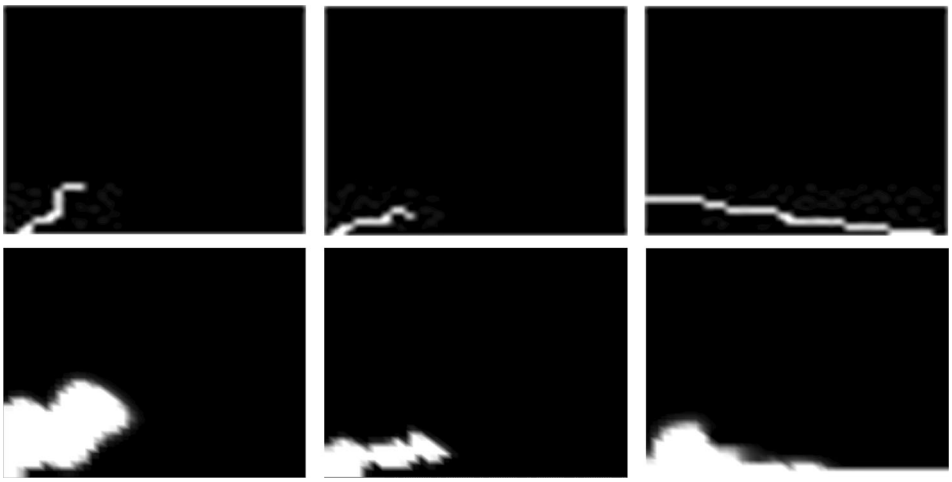

Figure 12. Simulation graphs for the scenes in Fig. 11 first line.

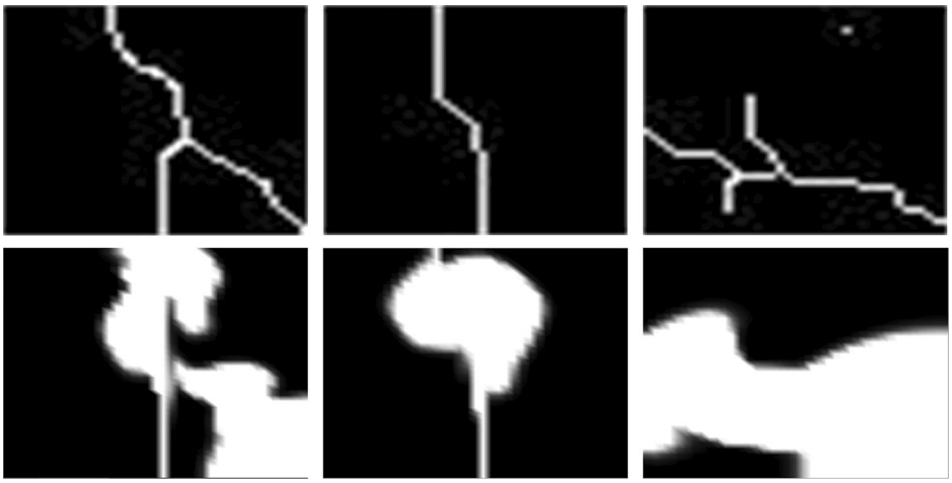

Figure 13. Simulation graphs for the scenes in Fig. 11 second line.

is ''Smoke Detection Method Based on LBP and SVM from Surveillance Camera'' [14], and the second one is ''Video smoke detection using shape,color and dynamic features'' [21]. The results is shown in Table 6. The data used in our method is taken from Table 4, and the OLP is set to 70%. The threshold θ_m mentioned in the first method is 5.

3.3.4. Algorithmic Performance in Cloudy Environment It can be seen from Table 6 that the proposed method has relatively stable accuracy and FPR in cloudy conditions (TEST13–TEST22). The result is mainly due to stable smoke roots. In order to show the performance of smoke roots in a better way, we accel-

Table 4
The Relationship Between Overlap Rate and Accuracy

Videos	OLP TPR (%)	10% FPR (%)	OLP TPR (%)	30% FPR (%)	OLP TPR (%)	50% FPR (%)	OLP TPR (%)	70% FPR (%)	OLP TPR (%)	90% FPR (%)
TEST01	90.80	34.00	90.80	25.20	90.80	9.60	81.60	3.60	62.80	0.00
TEST02	96.40	10.40	96.40	8.80	92.40	4.40	87.60	4.40	68.40	0.80
TEST03	68.80	60.80	68.80	42.80	65.60	17.20	64.00	8.40	49.60	2.80
TEST04	98.80	41.60	98.80	32.40	98.40	12.40	92.80	4.40	86.00	2.80
TEST05	93.20	7.20	93.20	6.80	93.20	3.60	92.40	0.80	84.40	0.00
TEST06	99.60	31.20	99.60	25.60	98.40	10.80	91.60	9.20	74.80	4.40
TEST07	94.80	21.20	94.80	17.60	94.80	6.40	88.40	6.00	87.60	0.40
TEST08	91.20	37.20	91.20	36.00	82.40	26.80	77.20	18.00	58.00	4.00
TEST09	82.00	44.00	82.00	21.60	81.60	8.40	80.40	2.80	66.80	0.00
TEST10	76.80	66.80	76.80	53.60	74.00	24.80	74.00	17.20	72.80	5.20
TEST11	52.80	77.20	52.80	65.60	42.80	40.00	38.40	19.20	36.00	12.40
TEST12	98.80	4.40	98.80	3.60	94.40	0.80	94.00	0.80	91.60	0.00
TEST13	94.00	22.00	94.00	19.20	94.00	6.40	94.00	2.80	90.00	0.00
TEST14	92.40	71.60	92.40	53.20	92.40	12.40	92.40	4.40	88.40	2.80
TEST15	95.20	30.40	95.20	24.00	92.40	5.60	91.20	5.60	82.80	0.00
TEST16	95.20	6.80	95.20	3.60	94.80	2.40	94.80	0.40	81.60	0.00
TEST17	71.60	64.40	71.60	45.20	60.00	40.80	53.60	16.40	37.60	11.20
TEST18	66.80	81.60	66.40	58.80	66.40	28.80	66.40	9.20	55.60	3.20
TEST19	96.80	15.20	96.80	11.60	96.40	6.80	96.40	4.80	92.00	0.00
TEST20	100.00	28.80	100.00	21.20	99.60	10.40	99.60	2.80	96.40	0.40
TEST21	96.00	20.40	96.00	18.00	96.00	5.20	96.00	5.20	93.20	0.00
TEST22	96.00	7.60	95.20	7.60	95.20	4.40	95.60	4.00	88.00	1.20

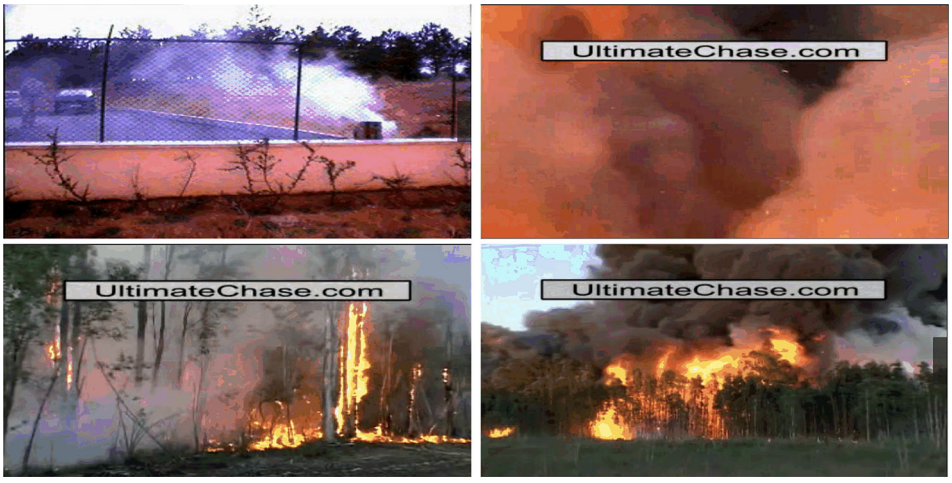

Figure 14. Invalid scenes for proposed method.

Table 5
Time Cost for Complete Algorithm

Videos	Time (ms/frame) Ave	Videos	Time (ms/frame) Ave
TEST01	447.36	TEST12	586.31
TEST02	847.19	TEST13	577.44
TEST03	506.39	TEST14	322.99
TEST04	329.77	TEST15	347.12
TEST05	966.25	TEST16	368.78
TEST06	973.98	TEST17	509.14
TEST07	1142.69	TEST18	501.11
TEST08	613.87	TEST19	409.83
TEST09	665.24	TEST20	344.87
TEST10	901.73	TEST21	375.61
TEST11	391.66	TEST22	534.48

Table 6
Precision for Different Method

	Method1		Method2		Propoed method	
Videos	TPR (%)	FPR (%)	TPR (%)	FPR (%)	TPR (%)	FPR (%)
TEST01	83.20	4.80	77.60	6.00	81.60	3.60
TEST02	81.60	2.80	91.20	3.20	87.60	4.40
TEST03	77.60	12.00	67.20	19.20	64.00	8.40
TEST04	94.40	2.80	82.00	12.00	92.80	4.40
TEST05	99.60	2.00	94.40	2.00	92.40	0.80
TEST06	88.00	12.00	71.20	14.40	91.60	9.20
TEST07	84.00	6.00	88.40	4.80	88.40	6.00
TEST08	74.00	8.00	64.40	20.80	77.20	18.00
TEST09	91.20	4.80	89.60	14.40	80.40	2.80
TEST10	93.20	5.20	81.60	7.20	74.00	17.20
TEST11	70.80	12.80	71.20	8.00	38.40	19.20
TEST12	88.40	7.20	76.00	2.00	94.00	0.80
TEST13	91.20	32.00	96.00	24.80	94.00	2.80
TEST14	89.60	20.40	91.60	14.80	92.40	4.40
TEST15	93.20	13.60	90.40	8.00	91.20	5.60
TEST16	97.60	16.80	90.00	10.80	94.80	0.40
TEST17	81.20	28.00	60.00	30.80	53.60	16.40
TEST18	93.20	31.20	73.20	27.20	66.40	9.20
TEST19	92.40	12.00	87.20	8.80	96.40	4.80
TEST20	70.80	21.20	90.00	20.40	99.60	2.80
TEST21	77.20	17.20	95.20	11.20	96.00	5.20
TEST22	88.00	10.00	96.00	8.00	95.60	4.00

erated the TEST20 video by 1.5 times, making the Vibe algorithm more sensitive to cloud movement. As can be seen in Fig. 15, the region of clouds do extract the wrong smoke root results, but we also can find that the positions of these roots

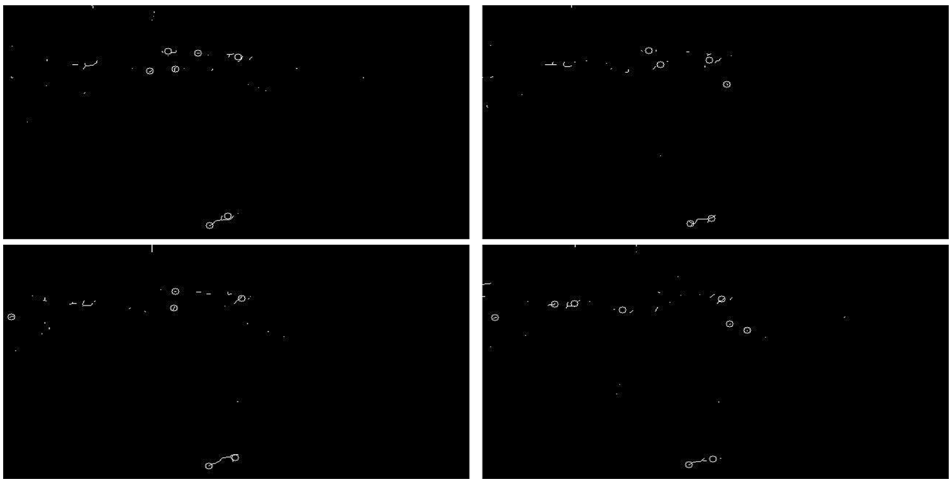

Figure 15. Roots performance in cloudy condition with 1.5 times acceleration.

are not stable in these frames, almost only for a moment. The computation domain established by these roots will also exist for a short moment and cannot be used in a simulation process. The stable roots at the bottom of these frames can still produce stable simulation results to ensure stable accuracy and low FPR. The low detection rates for TEST17 and TEST18 will be discussed in Sect. 3.4.

3.4. Discussion

In this section, we will summarize the data presented above and present our conclusions. At the beginning, we mainly discuss the sensitivity of the algorithm to time.

First, time cost of the simulation stage is at least 50% of the overall algorithm. Different resolutions and video materials in different environment will also produce different time results. Even for the videos that have the same resolution, time cost can be different greatly. For example, the simulation time of TEST06 spends nearly seven times as long as TEST20 in Table 3. The essential reason of this difference is the number of dynamic regions in the frames. Even though TEST13 and TEST15 have a common scene, there are dynamic branches on the right side of TEST13 that produce some false smoke roots, which leads to a longer simulation computation time.

Second, the more smoke roots detected, the more time will be spent handling data connections. As shown in Table 5, The average time cost of TEST07 reached 1142 ms, while the total time of image the processing stage and simulation stage was about 833 ms, which means the remainder of the program took about 300 ms. But the corresponding time cost in TEST20 was only about 200 ms. It can be inferred that when the number of smoke roots increase, not only the simulation stage, but also the time of data transfer stage will increase.

In a word, the proposed method used with high resolution cameras or a high frequency dynamic background (like the forest swings with the wind), will lead to a longer time cost, which may not fully guarantee real-time detection.

Next, the factors that affect detection accuracy will be explored. According to Table 4, with a different value of OLP, FPR is more sensitive than TPR. The higher the OLP value is set, the more the simulation domain is eliminated. If the OLP is set to 0, the results of TPR will be equal to FPR. But when the value of OLP is high enough, some of the correct results will be eliminated. For example, when the plane of drifting smoke is not parallel to the lens, the two-dimensional smoke simulation algorithm may not be completely accurate, like in TEST8 and TEST17. In TEST17, the direction of the smoke is almost perpendicular to the camera lens, which make the simulation results quite different from the actual situation. This distribution of smoke will be recognized as going in a vertical direction. This is also the reason why the detection rate is not good enough in those two scenarios.

4. Conclusion

In this paper, a new method is proposed to predict the result of smoke distribution through the theory of fluid mechanics. The algorithm is robust and has a good detection efficiency compared with the algorithms that only rely on image feature extraction, especially in cloudy situations. First, the dynamic background is extracted from the continuous frame images. Second, the smoke root extraction method is used to find the points that are relatively stable in continuous frame images. These points are fire source candidate points. Then, computation domains are created with the smoke roots as origins and Navier–Stokes equations are used to simulate the image generation process. Finally, the simulation results are compared with the original RGB images to obtain final results. The sensitivity of the method to time cost depends on the resolution and scenarios displayed in the video. In our experiments, with a resolution of 480×320, it can achieve a detection rate of up to 0.35 s/frame. In cloudy conditions, we have achieved an accuracy rate higher than 90% and have maintained an FPR below 10% at the same time. However, when an angle exists between the direction of smoke and camera lens, the accuracy of the algorithm decreases to some extent.Publisher's Note-Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Acknowledgements

The authors are grateful to Qingdao Haohai Network Technology Joint Stock Co., LTD. For video materials provided during the entire research process.

References

1. Barnich O, Droogenbroeck MV (2011) Vibe: a universal background subtraction algorithm for video sequences. IEEE Trans Image Process 20(6):1709–1724. https://doi.org/10.1109/TIP.2010.2101613
2. Bengio Y (2009) Learning deep architectures for AI. Now Found Trends. https://doi.org/10.1561/2200000006
3. Chorin AJ (1968) Numerical solution of the Navier–Stokes equations. Math Comput 22(104):745–762
4. Droogenbroeck MV, Paquot O (2012) Background subtraction: experiments and improvements for vibe. In: 2012 IEEE computer society conference on computer vision and pattern recognition workshops, pp 32–37. https://doi.org/10.1109/CVPRW.2012.6238924
5. Emmy Prema C, Vinsley SS, Suresh S (2018) Efficient flame detection based on static and dynamic texture analysis in forest fire detection. Fire Technol 54(1):255–288. https://doi.org/10.1007/s10694-017-0683-x
6. Fedkiw R, Stam J, Jensen HW (2001) Visual simulation of smoke. In: Proceedings of the 28th annual conference on computer graphics and interactive techniques, SIGGRAPH '01. ACM, New York, pp 15–22. https://doi.org/10.1145/383259.383260
7. Foster N, Metaxas D (1996) Realistic animation of liquids. Graph Models Image Process 58(5):471–483. https://doi.org/10.1006/gmip.1996.0039
8. Frizzi S, Kaabi R, Bouchouicha M, Ginoux J, Moreau E, Fnaiech F (2016) Convolutional neural network for video fire and smoke detection. In: IECON 2016-42nd annual conference of the IEEE industrial electronics society, pp 877–882. https://doi.org/10.1109/IECON.2016.7793196
9. Han XF, Jin J, Wang MJ, Jiang W, Gao L, Xiao LP (2017) Video fire detection based on Gaussian mixture model and multi-color features. Signal Image Video Process 11:1419–1425. https://doi.org/10.1007/s11760-017-1102-y
10. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Proceedings of the 25th international conference on neural information processing systems, vol 1, NIPS'12. Curran Associates Inc., pp 1097–1105
11. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94
12. Lucas BD, Kanade T (1981) An iterative image registration technique with an application to stereo vision. In: Proceedings of the 7th international joint conference on artificial intelligence, IJCAI'81, vol 2. Morgan Kaufmann Publishers Inc., San Francisco, pp 674–679. http://dl.acm.org/citation.cfm?id=1623264.1623280
13. Pundir AS, Raman B (2017) Deep belief network for smoke detection. Fire Technol 53(6):1943–1960. https://doi.org/10.1007/s10694-017-0665-z
14. Russo AU, Deb K, Tista SC, Islam A (2018) Smoke detection method based on LBP and SVM from surveillance camera. In: 2018 International conference on computer, communication, chemical, material and electronic engineering (IC4ME2), pp 1–4. https://doi.org/10.1109/IC4ME2.2018.8465661
15. Solrzano A, Fonollosa J, Fernndez L, Eichmann J, Marco S (2017) Fire detection using a gas sensor array with sensor fusion algorithms. In: 2017 ISOCS/IEEE international symposium on olfaction and electronic nose (ISOEN), pp 1–3. https://doi.org/10.1109/ISOEN.2017.7968889
16. Stam J (1999) Stable fluids. In: Proceedings of the 26th annual conference on computer graphics and interactive techniques, SIGGRAPH '99. ACM Press/Addison-Wesley Publishing Co., New York, pp 121–128. https://doi.org/10.1145/311535.311548

17. Tao C, Zhang J, Wang P (2016) Smoke detection based on deep convolutional neural networks. In: 2016 International conference on industrial informatics: computing technology, intelligent technology, industrial information integration (ICIICII), pp 150–153. https://doi.org/10.1109/ICIICII.2016.0045
18. Tung TX, Kim JM (2011) An effective four-stage smoke-detection algorithm using video images for early fire-alarm systems. Fire Saf J 46(5):276–282. https://doi.org/10.1016/j.firesaf.2011.03.003
19. Utkin AB, Fernandes A, Simes F, Lavrov A, Rui V (2003) Feasibility of forest-fire smoke detection using lidar. Int J Wildland Fire 12(2):159
20. Vincent L (1992) Morphological area openings and closings for grey-scale images. In: Proceedings of the workshop shape in picture. Springer, pp 197–208
21. Wang S, He Y, Wang K, Wang J (2017) Video smoke detection using shape, color and dynamic features. J Intell Fuzzy Syst 33(1):305–313. https://doi.org/10.3233/JIFS-161605
22. Wang Y, Wu A, Zhang J, Zhao M, Li W, Dong N (2016) Fire smoke detection based on texture features and optical flow vector of contour. In: 2016 12th World congress on intelligent control and automation (WCICA), pp 2879–2883. https://doi.org/10.1109/WCICA.2016.7578611
23. Ye S, Bai Z, Chen H, Bohush R, Ablameyko S (2017) An effective algorithm to detect both smoke and flame using color and wavelet analysis. Pattern Recognit Image Anal 27(1):131–138. https://doi.org/10.1134/S1054661817010138
24. Yin Z, Wan B, Yuan F, Xia X, Shi J (2017) A deep normalization and convolutional neural network for image smoke detection. IEEE Access 5:18429–18438. https://doi.org/10.1109/ACCESS.2017.2747399
25. Yu L, Wang N, Meng X (2005) Real-time forest fire detection with wireless sensor networks. In: Proceedings. 2005 International conference on wireless communications, networking and mobile computing, 2005, vol 2, pp 1214–1217. https://doi.org/10.1109/WCNM.2005.1544272
26. Zhang Q, Lin G, Zhang Y, Xu G, Wang J (2018) Wildland forest fire smoke detection based on faster R-CNN using synthetic smoke images. Procedia Eng 211:441–446. https://doi.org/10.1016/j.proeng.2017.12.034
27. Zhang TY, Suen CY (1984) A fast parallel algorithm for thinning digital patterns. Commun ACM 27(3):236–239. https://doi.org/10.1145/357994.358023
28. Zhao Y, Li Q, Gu Z (2015) Early smoke detection of forest fire video using CS Adaboost algorithm. Optik: Int J Light Electr Opt 126(19):2121–2124. https://doi.org/10.1016/j.ijleo.2015.05.082
29. Zhou Z, Shi Y, Gao Z, Li S (2016) Wildfire smoke detection based on local extremal region segmentation and surveillance. Fire Saf J 85:50–58. https://doi.org/10.1016/j.firesaf.2016.08.004