**INDUSTRIAL AND COMMERCIAL APPLICATION**

CrossMark

# An efficient edge extraction approach for flame image analysis

Vilas H. Gaidhane[1] · Yogesh V. Hote[2]

## Abstract

In this paper, a simple and robust approach for flame and fire image analysis is proposed. It is based on the local binary patterns, double thresholding and Levenberg–Marquardt optimization technique. The presented algorithm detects the sharp edges and removes the noise and irrelevant artifacts. The autoadaptive nature of the algorithm ensures the primary edges of the flame and fire are identified in the different conditions. Moreover, a graphical approach is presented which can be used to calculate the combustion furnace flame temperature. The various experimentations are carried out on synthetic as well as real flame and fire images which validate the efficacy and robustness of the proposed approach.

**Keywords** Flame image · Edge detection · Local binary patterns · Levenberg–Marquardt algorithm

## 1 Introduction

Concurrent and reliable monitoring of flames in fossil-fuel-fired furnace plays an important role in the improved understanding of energy conversion, pollutant formation and subsequent optimization of combustion processes. However, most of the practices for flame monitoring are largely limited to indicating whether the flame is present for safety reasons. In recent years, the trend of using low-quality fuels and fuel blends from a variety of sources has aggravated the problems of flame stability, poor combustion efficiency, high emissions and other operational problems. To meet the stringent standards on energy saving and pollutant emissions, advanced monitoring and characterization of flames have become highly desirable in the power generation industry as well as the research laboratory [1].

The physical characteristics of a flame and fire image can be identified by its geometry, luminance and thermodynamics parameters. In recent years, fire safety engineering research is mostly emphasized on the image processing-based system and increasingly used in flame and fire

detection systems [2–6]. Moreover, the image processing-based techniques are found more reliable as compared to the conventional flame detector techniques [7–10], due to their capability of remote detection of small-area fire, small-sized flame and other merits [11].

Edge detection is one of the important steps in fire and flame image processing which provide the foundation for analysis and monitoring of the system. It is due to the fact that the flame and fire edges form a basis for the qualitative analysis of the parameters such as area, size, shape, location and stability. Moreover, the edges can reduce the amount of redundant data and filter out the unwanted information such as background noise and ambiguities within the images. Furthermore, edges can be used to segment a group of flames which can be helpful for multiple flame monitoring in the industry [12].

In the last decades, several techniques have been proposed to identify the fire and flame image edges [13–15]. Bheemul et al. [16] presented an effective technique to extract the flame contours by detecting the changes of brightness in the horizontal direction line to line over the flame image. However, this technique is suitable only for steady and simple flame images. Toreyin et al. [17] proposed an edge detection algorithm based on the hidden Markov models and wavelet transform. Further, Wang et al. [18] presented a Chan–Vese active contour model for flame images, particularly obtained from the power plant. Jiang and Wang [19] improved the Canny edge detector which has been used to detect moving fire area in larger region fire images. Recently, Qiu et al. [20] proposed an autoadaptive

✉ Vilas H. Gaidhane
vilasgd612@gmail.com

1 Department of Electrical and Electronics Engineering, Birla Institute of Technology and Science, Pilani, Dubai Campus, International Academic City, 345055, Dubai, UAE

2 Department of Electrical Engineering, Indian Institute of Technology (IIT), Roorkee, Roorkee, India

edge detection algorithm for flame and fire image processing. It is based on the modified Canny edge detection technique and iterative threshold adjustment. Moreover, Gupta and Gaidhane [21] developed a simple and robust approach for flame image analysis, which is based on the local binary patterns (LBP) and double thresholding techniques. In this technique, the main features of an image are obtained by local binary patterns, and two-level thresholding is used to make the edges visible and clear. Although all these methods have its own advantages for flame and fire detection, they have some limitations such as unclear and discontinuous edge detection. Therefore, for detecting the size, shape and other geometric characteristic, and to obtain the clear, continuous and closed edges of the flame and fire images, an effective and efficient technique still remains a challenge for image processing researchers.

In this paper, some existing techniques have been tested to study the effectiveness in fire and flame image analysis. It is observed that the response of these techniques is affected by fragmentation and the edges of the fire and flame images are melted. To overcome these problems, an efficient and effective technique for fire and flame image analysis is proposed. It is based on the LBP operator and the double thresholding method. The double thresholding is carried out by the Levenberg–Marquardt (LM) optimization technique.

The rest of the paper is organized as follows: A brief overview of existing techniques is explained in Sect. 2. Sections 3 and 4 describe the proposed algorithm for fire and flame image edge detection and analysis, respectively. Section 5 presents the experimental results for various flame and fire images. Finally, concluding remarks and future research scopes are given in Sect. 6.

## 2 Overview of edge detection techniques

The tasks such as process, analysis and monitoring of flame images are of paramount importance in the power generation industry. It is increasingly used in temperature, flame and fire detection systems. It is due to the fact that the temperature calculation of visible area is able to provide additional information such as fire region, effects, gases [22–24]. In these techniques, edge detection is one of the important steps and acts as the foundation for other processing.

Generally, the edges of an image are represented by first- and second-order derivatives of the 2D image $f(x, y)$. Mathematically, it can be defined by a gradient vector as [25]

$$\nabla f(x,y) = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f(x,y)}{\partial x} \\ \frac{\partial f(x,y)}{\partial y} \end{bmatrix} \tag{1}$$

where $G_x$ and $G_y$ are the gradient in the $x$ and $y$ coordinates, respectively. For 2D digital image [25],

$$G_x = f(x + 1, y) - f(x,y) \text{ and } G_y = f(x, y + 1) - f(x,y) \tag{2}$$

The magnitude of the gradient vector is given by

$$\text{mag}(\nabla f) = \sqrt{G_x^2 + G_y^2} = \sqrt{\left( \frac{\partial f(x,y)}{\partial x} \right)^2 + \left( \frac{\partial f(x,y)}{\partial y} \right)^2} \tag{3}$$

The direction of the gradient vector is represented by the angle $\alpha$ as

$$\alpha(x,y) = \tan^{-1}\left( \frac{G_x}{G_y} \right) \tag{4}$$

Based on above mathematics and first-order derivative, Roberts, Prewitt, Sobel and Canny are presented in the literature and these methods are commonly used to extract the edges of an image. The second-order derivative-based method is Laplacian of Gaussian (LOG) also called as Mexican Hat or Marr–Hildreth operator. The Gaussian function is a nonlinear function given as

$$G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{5}$$

where $\sigma$ is a standard deviation called as space constant. Then the Laplacian of Gaussian (LOG) function can be written as [25]

$$\nabla^2 G(x,y) = \frac{\partial^2 G(x,y)}{\partial x^2} + \frac{\partial^2 G(x,y)}{\partial y^2},$$
$$\nabla^2 G(x,y) = \left[ \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right] e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{6}$$

In this method, the edges are detected by searching a second-order derivative over the entire image and search of zero crossing of LOG. However, these methods have limited performance as compared to Canny edge detection method.

In the last decade, some edge detection-based techniques have been proposed and applied in various applications for flame or fire image analysis [11–15]. Recently, Qiu et al. [20] presented an autoadaptive edge detection approach and applied to the flame and fire image analysis. Further, Habiboğlu et al. [2] proposed a fire and flame detection method based on the covariance matrix approach. This method works very well when the fire is close and visible. However, it poorly performs for small fire area. Moreover, it is observed that the above existing techniques are not enough suitable for flame and fire image analysis. Therefore, an efficient and simple approach is proposed for fire and flame image analysis.

## 3 Proposed algorithm for flame and fire image analysis

Generally, a flame region has a higher luminance as compared to its ambient and continuous background. Therefore, it is required to enlarge the difference among the flame and background. The basic idea is to extract the principle edges and remove the irrelevant edges for image analysis. To carry out this process, a new approach is proposed which is summarized in the following steps.

Step 1: *Extracting and adjusting the gray level*

In this step, the color image is converted to the grayscale image. It can be done by simple averaging of RGB components or separate average of colors prominent wise. Then the gray level of an image is adjusted according to its statistical distribution.

If $I(x,y)$ is the intensity value of a point $(x,y)$ in the original image $f(x,y)$, then the edge value is given as [26, 27]

$$V(x,y) = \frac{1}{m \times m} \sum_{i=-m}^{m} \sum_{j=-m}^{m} I(x+i, y+j) \tag{7}$$

Then the intensity $u(x,y)$ and local information $v(x, y)$ are given as

$$u(x,y) = \frac{I(x,y) - I_{\min}}{I_{\max} - I_{\min}} \tag{8}$$

$$v(x,y) = \begin{cases} -\frac{1}{2} \frac{[V(x,y)]^{\beta}}{V_{\min}}, & V(x,y) < 0 \\ \frac{1}{2} \frac{[V(x,y)]^{\beta}}{V_{\max}}, & V(x,y) < 0 \end{cases} \tag{9}$$

where $\beta$ is enhancement factor $0 \leq \beta \leq 1$.

The multi-peak generalized histogram enhancement is calculated as

$$g(x,y) = f(u(x,y), v(x,y)) \tag{10}$$

The density function is given as

$$p(g(x,y)) = h(f(u,v)) \tag{11}$$

The value of the histogram $H(p)$ can be calculated from the density function. Finally, the output image is obtained using the piecewise linearization of $H(p)$ as $\hat{f} = T(p)$.

Step 2: *Smoothing of the image*

Next step is to smoothen the image. This is done by filtering out the noise from an image before finding its edges. A Gaussian filter can be exploited for this process. The Gaussian function can be represented as

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{12}$$

where $\sigma$ represents the standard deviation. Based on Eq. (12), a Gaussian mask can be obtained. As the width of the Gaussian mask is increased, the clarity of the edges is also improved. After certain tests and evaluation, the Gaussian mask with a standard deviation $\sigma = 1.4$ is used in the implementation.

$$G_{\text{mask}} = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \tag{13}$$

However, further increase in the Gaussian mask also increases the local error and it becomes very difficult to distinguish between the false and true edge.

Step 3: *Local Binary Patterns (LBP)*

After smoothing, LBP are applied to the preprocessed image. In LBP, each pixel in $3 \times 3$ neighborhoods is replaced by calculating values using Eq. (14). This process is repeated for all the pixels of an image [28].

$$\text{LBP}_{P,R} = \sum_{p=0}^{p-1} s(g_p - g_c) 2^p \tag{14}$$

where

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0. \end{cases} \tag{15}$$

$P$ number of equally spaced pixels, $R$ radius of circle with center $g_c$. $g_c$ gray value of the center pixel of a local neighborhood, $g_p$ gray value of a sampling point in an evenly spaced circular neighborhood.

The LBP operator of size $3 \times 3$ is shown in Fig. 1. From Fig. 1, the binary code using LBP operator is 01011111 and its equivalent decimal code is 95. Thus, the center pixel is replaced by 95 pixel values. Averaging of the RGB components, smoothing and LBP operation on the colored images handle the spatial and temporal information together.

Step 4: *Thresholding of the LBP image*

It is observed that the best results on LBP processed image can be obtained by selecting the higher threshold values $k_2^*$ and lower threshold value $k_1^*$. For this, range of pixel value between $k_2^*$ and $k_1^*$ is selected. However, the lower value
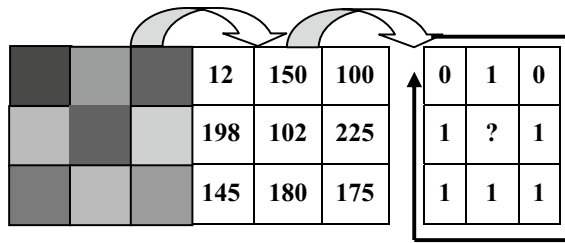
Binary equivalent = 01011111;   Decimal equivalent = 95

**Fig. 1** Example of LBP operator. Binary equivalent = 01011111; decimal equivalent = 95

of $k_1^*$ results into false edges and higher $k_2^*$ value eliminates the actual valid points. Therefore, the section of valid and correct threshold is an important task to process the LBP image. An optimal value of these thresholds is used in this paper. These multiple threshold values can be calculated using the global thresholding technique as given below.

The mean intensity of the pixels in the region $C_1$ (pixels $\{0, k\}$) and $C_2$(pixels$\{k + 1, L - 1\}$) is given as [25]

$$m_1 = \frac{1}{P_1} \sum_{i=0}^{k} i \cdot P_i \qquad (16)$$

$$m_2 = \frac{1}{P_2} \sum_{i=k+1}^{L-1} i \cdot P_i \qquad (17)$$

where $P_1$ and $P_2$ are the probabilities of the region $C_1$ and $C_2$, respectively. The global intensity value can be calculated as

$$m_G = \sum_{i=0}^{L-1} i \cdot P_i \qquad (18)$$

For double thresholding between-class variance can be obtained as

$$\sigma_B^2 = P_1 (m_1 - m_G)^2 + P_2 (m_2 - m_G)^2 \qquad (19)$$

Then the optimal thresholds for $k_1^*$ and $k_2^*$ can be computed as

$$\sigma_B^2 (k_1^*, k_2^*) = \max_{0 < k_1 < k_2 < L-1} \sigma_B^2 (k_1, k_2) \qquad (20)$$

At this stage, the preliminary image $\tilde{f}(x, y)$ with the edges is obtained from the original flame images. This image is processed further for closed edge extraction.

Step 5: *Obtaining the closed edges*

The LBP-based preliminary image $\tilde{f}(x,y)$ has thicker and unrelated edges wider than a pixel. Therefore, the non-maxima suppression technique is exploited for thinning down the broader edges.

The magnitude of the gradient of $\tilde{f}(x,y)$ intensity can be calculated as

$$\Im(x, y) = \sqrt{\tilde{f}_x^2 + \tilde{f}_y^2}, \qquad (21)$$

and the direction is given by

$$\theta(x, y) = \tan^{-1} \left[ \frac{\tilde{f}_x}{\tilde{f}_y} \right] \qquad (22)$$

The edge magnitude of two neighborhood edge pixels perpendicular to the edge direction is considered, and the one with a lesser edge magnitude is neglected. This non-maxima suppressed image may still contain few false edges. To remove these false edges, two thresholds $T_1$ and $T_2$ are chosen to create two different edge images $\tilde{f}_1$ and $\tilde{f}_2$, where $\tilde{f}_1$ is contained most of the false edges and $\tilde{f}_2$ has all continuous edges. The threshold selection algorithm starts with the edge points in $\tilde{f}_2$, linking the adjacent edge points in $\tilde{f}_2$ forming a contour, and the process continues till no more adjacent edge points are available. In order to speed up the process of finding the thresholds $T_1$ and $T_2$, a Levenberg–Marquardt (LM) optimization algorithm is used which optimizes a mean square error-based cost function [29, 30]. In this step, $T_1$ and $T_2$ need to be autoadjusted. Initially, the value of threshold $T_1$ and $T_2$ will be given based on the initial results. Further, $T_2$ can be adjusted to different values assuming $T_1$ fixed at initial value. The mean square error deviation $E(\mu_m)$ of a contour start from point $C_{start}(x_s, y_s)$ and end at $C_{end}(x_e, y_e)$ is used as a cost function to judge the new value of $T_2$. Suppose $E(\mu_m)$ small enough, the computation process terminates. If $E(\mu_m)$ is still greater than the desired value, a further error computation process is applied to $C_{start}(x_s, y_s)$. Thus, the cost function $\chi(\mu_m)$ can be written as

$$\chi(\mu_m) = \sum_{i=1}^{N} [E(\mu_m)]^2 \qquad (23)$$

where

$$E(\mu_m) = C_{start}(x_s, y_s) - C_{end}(x_e, y_e) \qquad (24)$$

The second-order Taylor series results in updated initial guess

$$\chi(\mu_{m+1}) = \chi(\mu_m + \Delta\mu_m),$$
$$\approx \chi(\mu_m) + g_m^T \Delta\mu_m + \frac{1}{2} \Delta\mu_m^T A_m \Delta\mu_m \qquad (25)$$

where $g_m = \nabla \chi(\mu)|_{\mu=\mu_m}$ and $A_m = \nabla^2 \chi(\mu)$. If we calculate the gradient of the above equation with respect to $\Delta \mu_m$ and set it equal to zero, then another equation can be obtained as

$$\begin{aligned} \chi(\mu_{m+1}) &= \chi(\mu_m + \Delta \mu_m), \\ \Delta \mu_m &= -A_m^{-1} g_m \end{aligned} \tag{26}$$

Thus, the Newton's method can be defined as

$$\mu_{m+1} = \mu_m - A_m^{-1} g_m \tag{27}$$

The Levenberg–Marquardt algorithm is a variation of Newton's statistical method which is designed for minimizing the sums of squares of nonlinear functions. Here, $\chi(\mu_m)$ is a sum of square functions as given in Eq. (23). The gradient of $\chi(\mu_m)$ can be written as [30]

$$\nabla \chi(\mu_m) = 2J^T(\mu_m)E(\mu_m) \tag{28}$$

where $J(\mu_m)$ is the image Jacobian matrix. Similarly, Hessian matrix can be obtained as

$$\nabla^2 \chi(\mu_m) = 2J^T(\mu_m)J(\mu_m) \tag{29}$$

Then the updated guess value can be calculated as

$$\begin{aligned} \mu_{m+1} &= \mu_m - \left[2J^T(\mu_m)J(\mu_m)\right]^{-1} 2J^T(\mu_m)E(\mu_m), \\ \mu_{m+1} &= \mu_m - \left[J^T(\mu_m)J(\mu_m)\right]^{-1} J^T(\mu_m)E(\mu_m) \end{aligned} \tag{30}$$

This Eq. (30) is called as Gauss–Newton method. Further, Hessian matrix ($H$) can be approximated as $G = H + \lambda_m I$. This leads to the LM algorithm

$$\begin{aligned} \mu_{m+1} &= \mu_m - \left[J^T(\mu_m)J(\mu_m) + \lambda_m I\right]^{-1} J^T(\mu_m)E(\mu_m), \\ \Delta \mu_m &= -\left[J^T(\mu_m)J(\mu_m) + \lambda_m I\right]^{-1} J^T(\mu_m)E(\mu_m) \end{aligned} \tag{31}$$

where $\lambda_m$ is a positive scalar parameter and $I$ is the identity matrix. The free parameter $\lambda_m$ works as an adaptive parameter of the step length during the iterations. The choice of this parameter has a great impact on the convergence of an error. This algorithm has very important features such that as $\lambda_m$ increases, it approaches the steepest descent algorithm and as $\lambda_m$ decreases to zero, the algorithm becomes Gauss–Newton.

Initially, the LM algorithm starts with $\lambda_m = 0.01$. If a step does not reach to smaller value for $E(\mu)$, then the step is repeated by multiplying $\lambda_m$ by the factor $\tau \geq 1$. This approaches to the steepest descent, but results in slow convergence. On the other hand, if a step produces a smaller value for $E(\mu)$, then $\lambda_m$ is divided by $\tau \geq 1$ for the next step so that the algorithm will move toward the Gaussian–Newton which provides the faster convergence.

Step 6: *Analysis of flame image*

After successful clear and continuous edge detection the various parameters such as area, mean, standard deviation, entropy are calculated to analyze the flame and fire images.
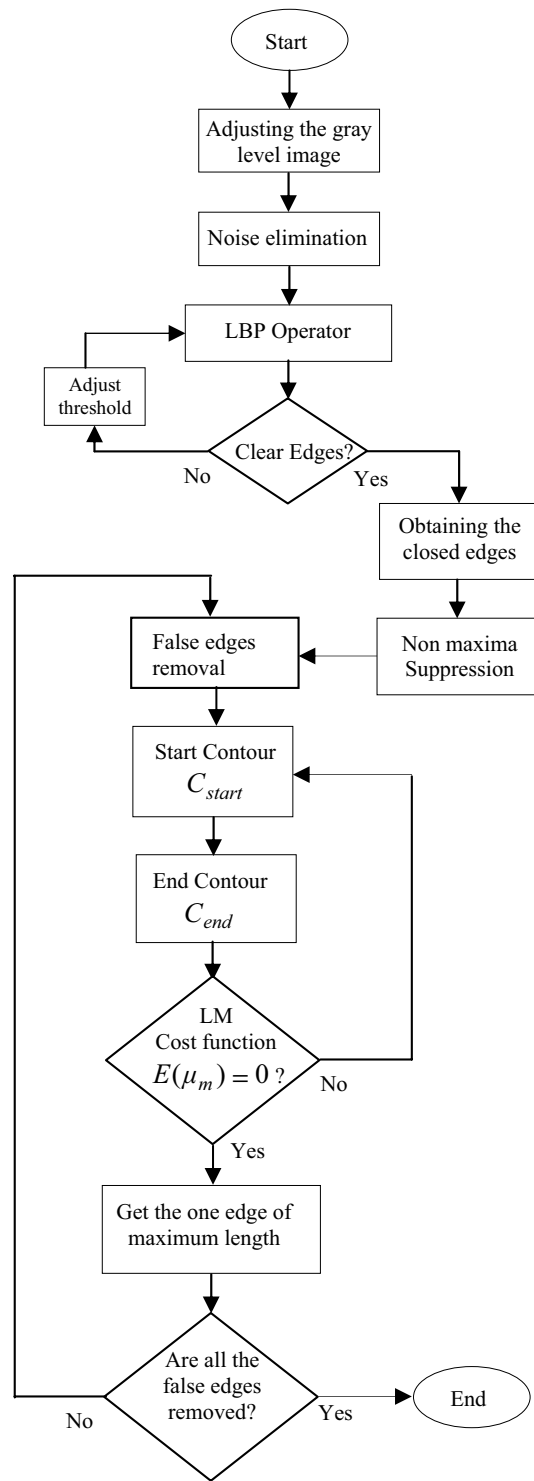


**Fig. 2** Proposed concept of flame and fire image edge detection

The whole process of flame and fire image edge detection is summarized using the flowchart as shown in Fig. 2

## 4 Flame image analysis

The various image parameters such as *Area*, *Mean*, *Standard Deviation* and *Variance* are used to analyze the edged flame and fire images. These parameters are described in this section.

### 4.1 Flame and fire image parameters

The *Area* is defined as the total number of nonzero pixels in an image. The principle of calculating the area is that as the area of the flame increases the size of the flame is also increasing. The area of edged images can be calculated as [25]

$$A = \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} I_{ij}, \quad \forall A \in \{0, w \times h\} \tag{32}$$

and

$$I_{ij} = \begin{cases} 0, & P(i, j) = 0 \\ 1, & P(i, j) > 0. \end{cases} \tag{33}$$

where $P(x, y) \in \{0, 255\}$ represent the intensity of pixels at the position $(x, y)$, $x \in \{0, w-1\}$, $y \in \{0, h-1\}$. *Mean* ($\mu$) is the overall average intensity of the pixels of the flame image. It can be represented as [25]

$$\mu = \frac{1}{A} \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} P(i, j), \quad \forall \mu \in \{0, 255\} \tag{34}$$

*Standard deviation* ($\sigma$) and *variance* ($v$) give the measures of the spread of the distribution or data about the mean.

$$v = \frac{1}{A} \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} [P(i, j) - \mu]^2 \text{ and } \sigma = \sqrt{v} \tag{35}$$

### 4.2 Formulation of relation between flame area and temperature

Image area plays an important role in flame and fire analysis. It is one of the simplest features and also acts as the foundation of other parameters computation. After various experimentations and careful analysis, it has been observed that the area of a red color component of the flame or fire images is directly proportional to the temperature. Moreover,

after detailed study of literature, it has been observed that the maximum fuel efficiency and minimum environmental pollution can be obtained at 1299 °C combustion furnace flame temperature [4]. Therefore, in this paper a flame image with temperature 1299 °C has been considered as a reference temperature image for an efficient working of the furnace.

It is well proven in image processing that the graphical and visual presentations are more suitable for the human vision system as compared to statistical and mathematical analysis. Therefore, in this paper a graphical approach is also presented which predicts the relation between temperature and the area of the flame image. The proposed approach for estimation of the temperature of the furnace using the graphical analysis is summarized in the following steps:

Step 1: Obtain the edged contour using the proposed approach as given in Sect. 3.
Step 2: Calculate the area of edge contour using Eq. (32).
Step 3: Calculate the radius of the reference and test flame images as

$$r_{\text{ref}} = \sqrt{\frac{A_{\text{ref}}}{\pi}}, \text{ and } r_{\text{test}k} = \sqrt{\frac{A_{\text{test}k}}{\pi}}, \quad \forall k = 1, 2, 3, \dots, n \tag{36}$$

Step 4: The circles $c_{\text{ref}}\{0, r_{\text{ref}}\}$ and $c_{\text{test}k}\{0, r_{\text{test}k}\}$ can be drawn for reference and test flame image, respectively.
Step 5: The temperature of the furnace can be obtained by the relation

$$T = 10^3 \times \left[ \alpha_0 (r_{\text{test}k} - r_{\text{ref}}) + \alpha_1 \right] \pm 5 \,^{\circ}\text{C} \tag{37}$$

where $\alpha_0 = 0.1038$ and $\alpha_1 = 1.3179$ are the constants and can be calculated using the polynomial curve fitting analysis.

## 5 Experimental results and analysis

After implementing the proposed algorithm as described in Sects. 3 and 4, the numbers of flame and fire images are processed to evaluate the effectuality of the proposed approach. All the experimentations are carried out on a system equipped with Intel(R) Core (TM) i7-4500U, CPU @ 2.40 GHz, 8 GB RAM in a MATLAB R2013a software environment.

### 5.1 Edge detection using proposed approach

In this paper, the conventional edge detection techniques have been applied with the appropriate parameters to process the flame and fire images. For fare comparison, the image database reported by Qiu et al. [20] and Gupta and Gaidhane [21] is considered for the experimentation.

These flame images are taken for propane Bunsen flames burning in open air with moving camera. Firstly, the existing edge detection techniques are tested on the flame images. Ideally, the extracted flame edges should be clear, continuous and uninterrupted. Although many parameters are adjusted in Sobel, Roberts, Prewitt and Laplacian operator techniques, flame edges are unclear and discontinuous. Therefore, it is not always possible to obtain ideal edges from real-life images using these existing techniques. Figure 3a–f shows the examples of results obtained by the various existing edge detection techniques. Among all these conventional methods, the Canny method gives the most clear, continuous and uninterrupted edges.

However, the result cannot be called as optimum as it gives the multiple responses. Therefore, it is clear that none of the traditional methods are able to give the optimum edges. Therefore, in this paper an attempt has been made and a new approach is proposed for clear and continuous edge detection as explained in Sect. 3.

The effectuality of the proposed approach is evaluated on the refinery flame image. The LBP operator of size $3 \times 3$ pixel and a Gaussian mask represented by Eq. (13) is considered in all these experiments. Moreover, the initial values of lower and higher threshold are set to $T_1 = 0.35$ and $T_2 = 0.98$, respectively. Assuming threshold $T_1$ as fixed in order to adjust $T_2$, the threshold optimization is initiated



**Fig. 3** Comparison results of existing edge detection techniques. **a** Original image, **b** Sobel method, **c** Roberts method, **d** Prewitt method, **e** Laplacian method and **f** Canny method
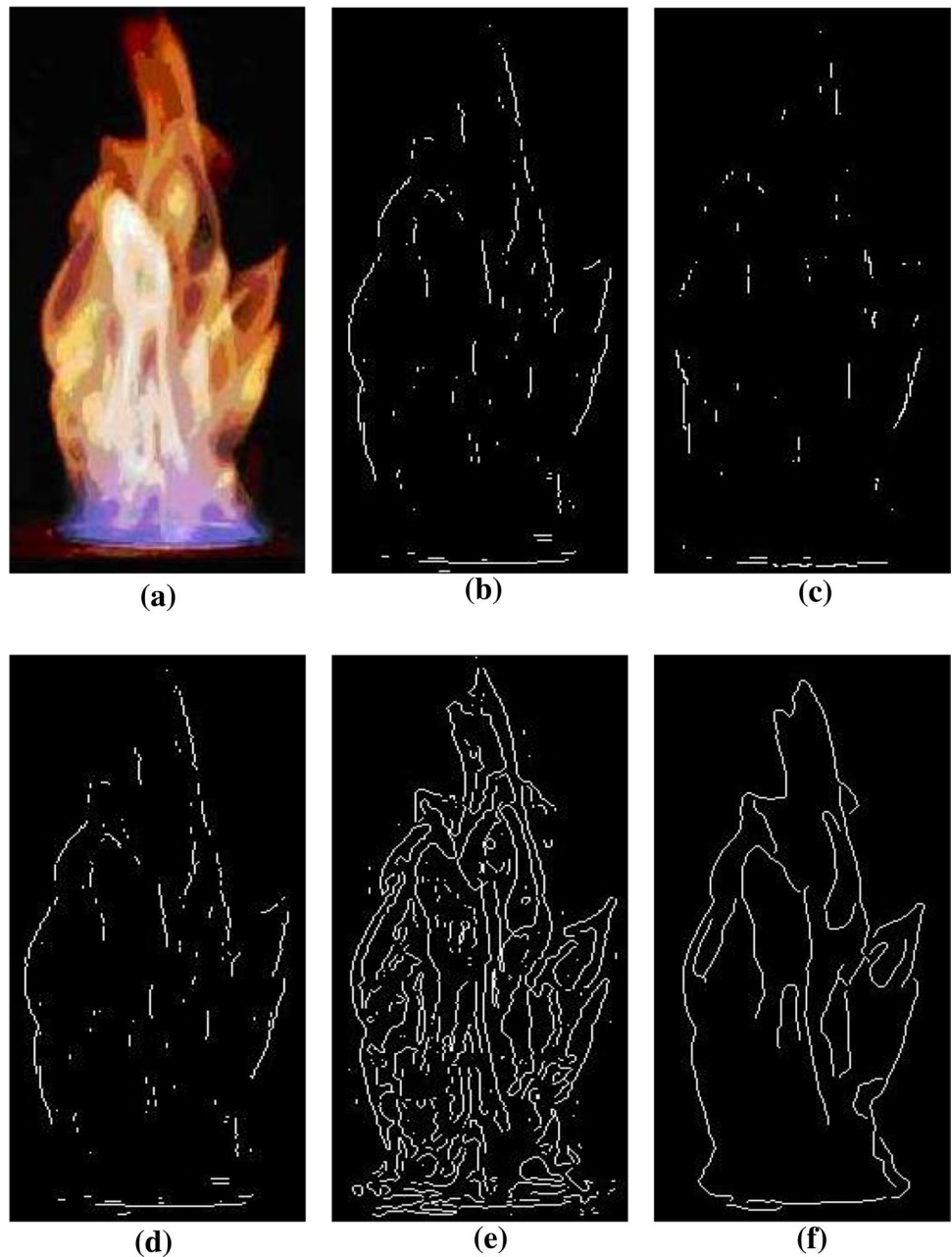
using the Levenberg–Marquardt (LM) algorithm. The LM algorithm is configured as one input layer, one hidden layer and one output layer. The LM optimization process starts with $\lambda_m = 0.01$. The various experimentations are carried on refinery flame image. The intermediate stage and final edge detection results of the proposed algorithm are shown in Fig. 4. Figure 4b shows a LBP pattern image, Fig. 4c shows the LBP plus a threshold image, and Fig. 4d shows the edged image obtained using the proposed approach. It is observed from Fig. 4d that the proposed approach generates the clear, continuous and uninterrupted edges.

Recently, the new computing algorithms [20, 21] are presented in the literature which is based on the strategy to remove the redundant edges and identifying the principal edges using a Sobel operator and thresholding technique. On the other hand, the proposed approach is based on LBP operator $(3 \times 3)$, double thresholding technique and LM-based threshold optimization which modifies the target image before the process of threshold adjustment
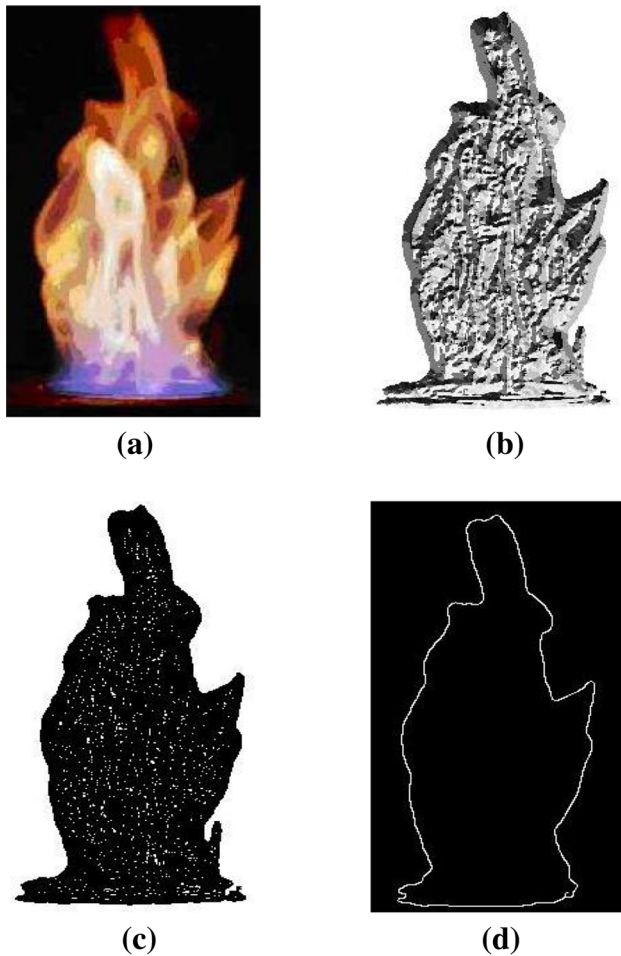


**Fig. 4** Illustration of proposed approach. **a** Original image, **b** LBP pattern image, **c** image after threshold and **d** edge detection using the proposed approach
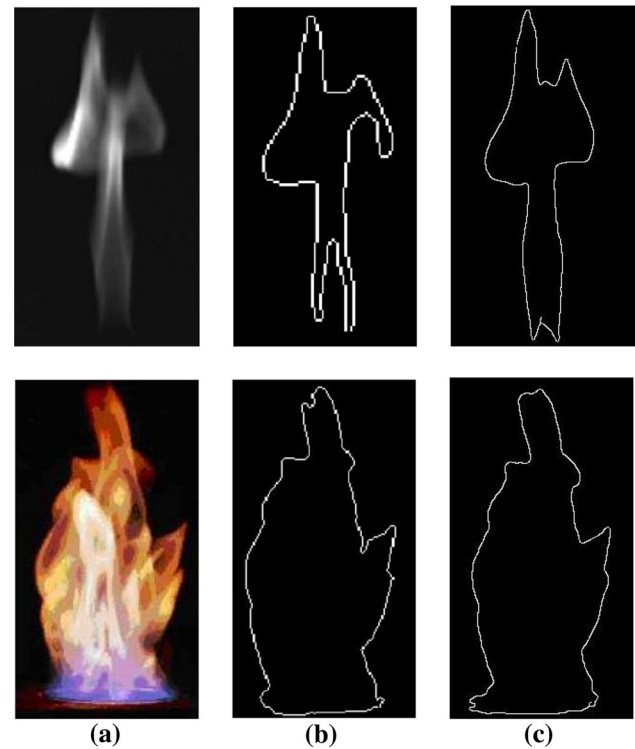


**Fig. 5** Comparison of results of autoadaptive [20] and proposed approach. **a** Original images, **b** autoadaptive technique, **c** proposed approach

and optimization of the edges. This makes the proposed approach more efficient than the existing approaches [20, 21]. In this experiment, the results of the proposed approach are compared with the autoadaptive edge detection technique. Figure 5 shows the edge detection results of autoadaptive technique [20] and proposed approach. It is clearly observed from Fig. 5 that the edges detected by the proposed approach are clearer and represent all the edge features of an image.

Further, the efficiency of the proposed approach is tested under the different noisy conditions. In this experiment, the different types of random noise such as salt and peppers $(\eta = 0.2)$ and Gaussian noise $(\rho = 0.3)$ are added to the reference flame images. Although the noise is added to the reference image, the edges obtained by the proposed approach are more clear and continuous than the existing method. The various results in the presence of salt and peppers and Gaussian noise are shown in Fig. 6.

The computational complexity is an another evaluation parameter for any algorithm. Therefore, the computational time of the proposed approach is compared with the existing conventional methods. This experiment is carried out for a set of five different images. Table 1 shows the average execution time of five images for existing methods as well as the proposed approach. It is observed from Table 1 that the time
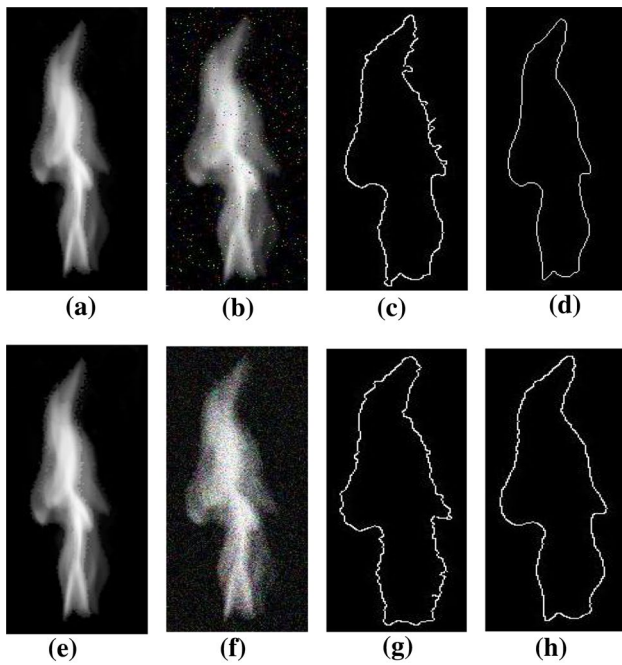
**Fig. 6** Results in noisy conditions. **a**, **e** Original image; **b** with salt and pepper noise, **f** with Gaussian noise, **c**, **g** edged image using auto-adaptive method [20] and **d**, **h** edge-detected image using proposed approach

taken by the proposed approach is little more than existing methods. It is due to the iterative nature Levenberg–Marquardt error optimization-based algorithm. This algorithm requires comparatively more time for error optimization. However, more computational time is compensated by the accuracy and effectiveness of the proposed approach.

In flame image analysis, it is very difficult to find clear and continuous edges of real and high resolution. Therefore, in this paper the effectiveness of the proposed approach is also tested on the high-resolution images. Figure 7a shows a wood gas flame image of high resolution of size 846 × 1721 pixel taken from a flame gasifier using moving camera. Similarly, Fig. 7d shows a high-resolution campfire image of dimension 1200 × 1600 pixel with static camera. In this experiment, the results of the proposed approach are compared with the existing Canny-based edge detection method. Figure 7b, e shows the Canny-based edge-detected images, whereas Fig. 7c, f shows the results of proposed approach. In Canny-based results (Fig. 7b, e) the edges are unclear and broken in certain areas, whereas the edges detected by proposed approach are comparatively clear and continuous. Thus, it is clear that the

propose approach performs better than the conventional edge detection methods for static as well as moving camera.

## 5.2 Edge detection based on red color component using proposed approach

In this experiment, the database [4] of flame images is used to detect the edges related to the red color component of RGB images. Further, these edged images are considered in the calculation of flame characteristics such as area, radius, mean, standard deviation, temperature. Figure 8 demonstrates an example of furnace combustion flame image database and their respective red color component-based edge detection results. It is well known that the red color component in an image contains most of the information about the temperature. Therefore, in this paper red color component is used for study and analysis of flame images. Figure 8a–d shows the furnace flame images with temperature 1197, 1299, 1397 and 1499 °C, respectively. These images are taken from blackbody furnace using flame image detector-based Samsung SCC-833P color charged-coupled device (CCD) camera installed in different height of the furnace wall.

In this experiment, a red color component is extracted to define the temperature of the flame image. From Fig. 8a–d, it is observed that the red color component increases as the temperature rises above the minimum level. In the literature [4], it is reported that the red color component starts growing above 1000 °C temperature in the furnace flame image. The temperature of these images can be obtained using the different methods [7–10]. However, these methods are more costly and require the dedicated instruments as well as the trained operator. On the other hand, image processing-based methods are reliable, simple and require less equipments. Therefore, in this paper an attempt has been made and a simple approach is proposed to calculate the temperature of a furnace as explained in Sect. 4. Here, the edges of the furnace flame images are extracted using the proposed approach as shown in Fig. 8e–h. It is observed that the edges of the red component of the flame image for temperature 1499 °C go beyond the expected area. It indicates the overheating temperature of the furnace and it can be used as a danger alarm in the system.

The area and radius of the different images can be obtained using Eqs. (32) and (36), respectively. Table 2 shows the area and radius of Fig. 8a–d. It is observed from Table 2 that as the temperature increases, the radius also increases. This property can be used to calculate the temperature of the furnace using image processing algorithms. However, the radius of Fig. 8d

**Table 1** Time comparison of different exiting methods and the proposed edge detection

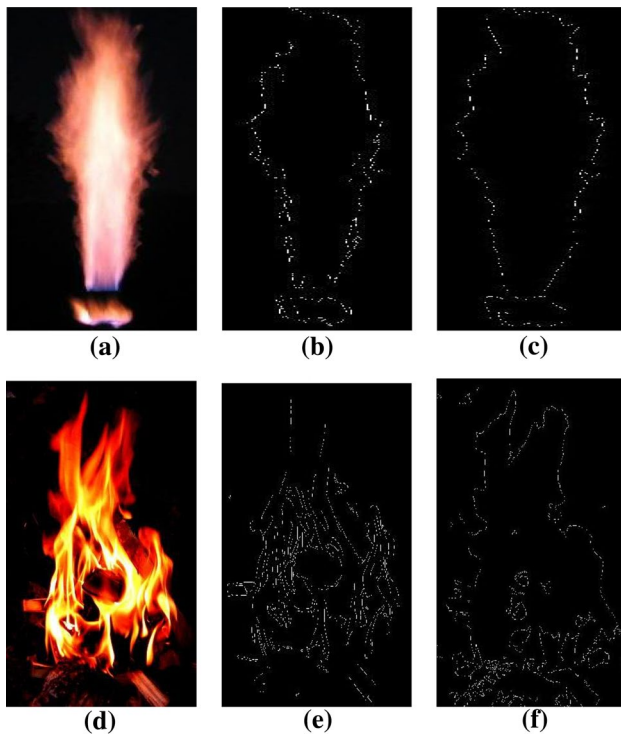| Methods | Sobel | Roberts | Prewitt | Log | Canny | Autoadaptive | Proposed |
|---------|-------|---------|---------|-----|-------|--------------|----------|
| Time (s) | 0.789 | 0.755 | 0.757 | 0.780 | 0.916 | 1.312 | 1.003 |

**Fig. 7** Edge detection results for real images. **a** Wood gas flame image, **b** edge detection using Canny method, **c** edge detection using proposed approach, **d** campfire image, **e** edge detection using Canny method and **f** edge detection using proposed approach

is infinite which can be used to show the danger alarm or overheating in the system.

### 5.3 Calculation of flame parameters

The different flame image parameters such as area, mean, standard deviation and entropy can be calculated using Eqs. (32)–(35). These parameters show the characteristics like stability, intensity and effects. Table 3 shows the calculated values of the various flame image features discussed in Sect. 4. It is observed from Table 3 that the area of the red component of the flame image is directly proportional to the temperature. Similarly, the mean and the standard deviation can be used to find the stability of the flame by calculating the coefficient of variance.

The values of mean and standard deviation increase with the increase in temperature. Moreover, it has been observed from Table 3 that the increase in temperature results into more entropy. Thus, these flame parameters can be used to predict the stability of the flame images.

### 5.4 Computation of flame temperature

The visual perception plays an important role in image processing and computer vision research area. Therefore, in this

paper an attempt has been made and proposed a graphical approach for the comparison of furnace flame temperature. In this approach, furnace flame temperature 1299 °C is considered as a reference or the ideal temperature at which the system gives an optimum output [4]. Here, the main aim is to calculate the real time temperature of the furnace with respect to the reference temperature. Moreover, one can calculate the temperature of the furnace flame using the proposed technique as explained in Sect. 4. The graphical method can be used for visual analysis of the furnace temperature.

In this experiment, furnace flame image database [4] is considered for the analysis and calculation of the temperature. Figure 9 shows an example of the proposed graphical approach. In this experiment, two images with unknown temperature and one reference image with known temperature (1299 °C) are considered for analysis. The radius of each image is calculated using the proposed approach as given in Sect. 3 as well as Sect. 4. Then the respective circles are drawn as shown in Fig. 9. The middle circle shown in Fig. 9 with radius 22.5 is drawn for the reference image. Similarly, the outer circle and inner circle are drawn for unknown temperature with calculated radius 23.45 and 21.26, respectively.

Now, Eq. (37) can be used to calculate the unknown temperature of the flame image with respect to the reference image. After various experimentations and rigorous analysis the value of constant $\alpha_0$ and $\alpha_1$ are set to 0.1038 and 1.3179, respectively. The temperature for outer circle and the inner circle is obtained which is found to be 1416 and 1189 °C, respectively.

Further, the results of the computation of temperature for various flame images are validated using the database consisting of 100 different flame images with known temperature. The proposed graphical method is found suitable for visual inspection. If the temperature of the flame image under test is equal to or closed to the reference flame image, then the circles will overlap on each other. Thus, operator can predict the under-heating or overheating of the furnace by simple visual inspection. Moreover, it is observed that the proposed approach is comparatively effective and efficient as compared to the conventional methods. Therefore, it can be used to calculate the edges and unknown temperature of the flame images.

## 6 Conclusions and future scope

In this paper, a simple and efficient flame and fire image analysis approach has been proposed and evaluated in comparison with the conventional techniques. Experimental results have shown that the developed algorithm is effective in identifying the clear, continuous and uninterrupted edges

**Fig. 8** Edge detection of red color component of database images using proposed approach **a–d** 1197, 1299, 1397 and 1499°C, respectively, and **e–h** edge detection results using proposed approach
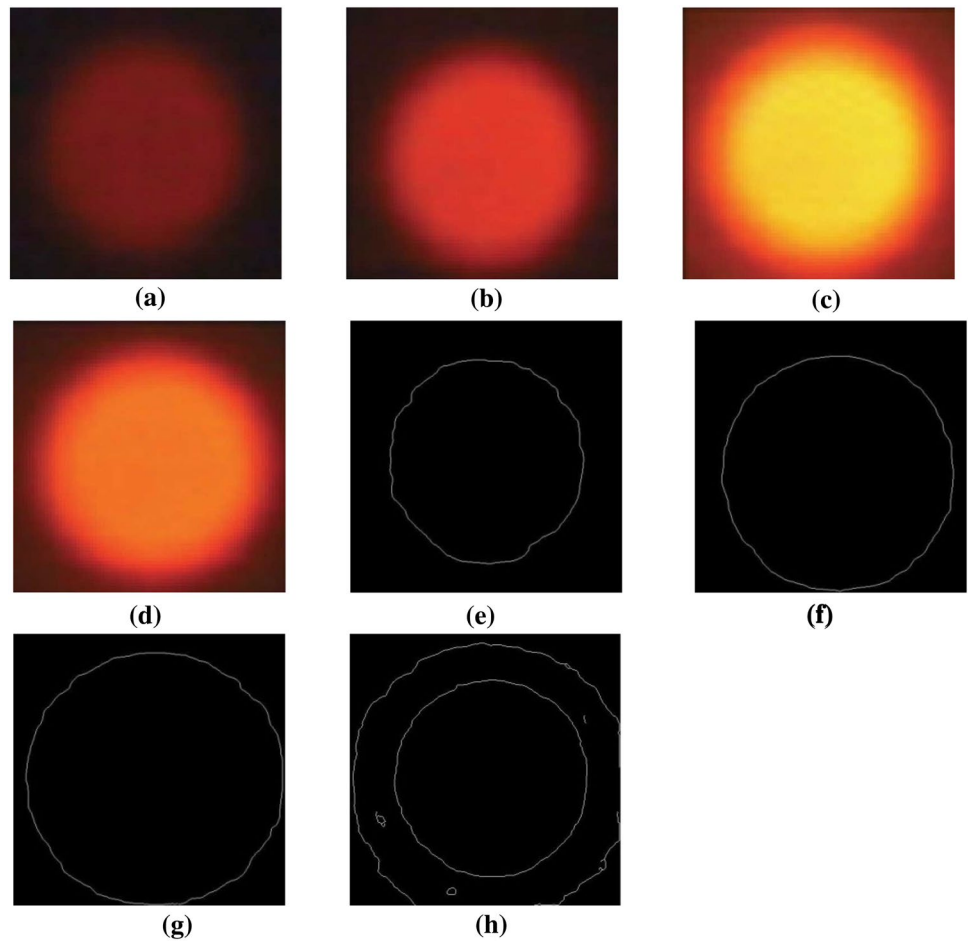


(a)　　　(b)　　　(c)

(d)　　　(e)　　　(f)

(g)　　　(h)

**Table 2** Variation of temperature, area and radius of the flame images

| Temperature (°C) | Area | Radius |
|---|---|---|
| 1197 | 1413 | 21.21 |
| 1299 | 1589 | 22.49 |
| 1397 | 1706 | 23.30 |
| 1499 | Undefined | Infinite |

**Table 3** Variation of flame parameter

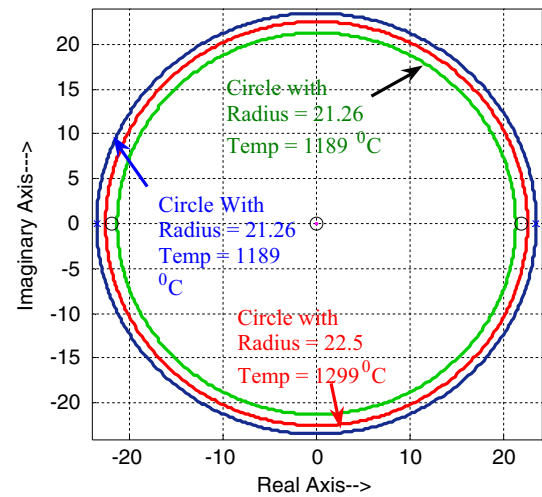| Characteristics parameter | Temperature (°C) | | | |
|---|---|---|---|---|
| | 1197 | 1299 | 1397 | 1499 |
| Threshold | 30 | 40 | 80 | 105 |
| Area | 1413 | 1585 | 1706 | – |
| Mean | 38.543 | 64.608 | 91.325 | 123.278 |
| Std. deviation | 30.2219 | 61.861 | 76.190 | 84.525 |
| Entropy | 5.3361 | 6.4051 | 6.8191 | 7.0607 |



**Fig. 9** Graphical comparison of the unknown temperature with the reference temperature

and effective area of the flame and fire images. The clearly defined flame area plays an important role in quantification of flame parameters such as flame surface area, flame volume, flame or fire spread speed, temperature. Moreover, the proposed approach may be useful in detailed understanding and advanced image processing-based monitoring of the furnace combustion flame.

# References

1. Zhang S, Shen G, An L, Niu L (2015) Online monitoring of the two-dimensional temperature field in a boiler furnace based on acoustic computed tomography. Appl Therm Eng 72:958–966
2. Habiboğlu YH, Günay O, Çetin AE (2012) Covariance matrix-based fire and flame detection method in video. Mach Vision Appl 23:1103–1113
3. Cetin AE, Bart M, Güna O, Töreyi BU, Verstockt S (2016) Methods and techniques for fire detection: signal, image and video processing perspectives. Academic Press, Cambridge
4. Luo Z, Zhou H-C (2007) A combustion-monitoring system with 3-d temperature reconstruction based on flame-image processing technique. IEEE Trans Instrum Meas 56:1877–1882
5. Tung TX, Kim JM (2011) An effective four-stage smoke-detection algorithm using video images for early fire-alarm systems. Fire Saf J 46:276–282
6. Yan Y, Qiu T, Lu G, Hossain MM, Gilabert G, Liu S (2012) Recent advances in flame tomography. Chin J Chem Eng 20:389–399
7. Coppo P (2015) Simulation of fire detection by infrared images from geostationary satellites. Remote Sens Environ 162:84–98
8. Kong SG, Jin D, Li S, Kim H (2016) Fast fire flame detection in surveillance video using logistic regression and temporal smoothing. Fire Saf J 79:37–43
9. Romero C, Li X, Keyvan S, Rossow R (2005) Spectrometer-based combustion monitoring for flame Stoichiometry and temperature control. Appl Therm Eng 25:659–676
10. Wojcik W, Golec T, Kotyra A (2004) Combustion assessment of pulverized coal and secondary fuel mixtures using the optical fiber flame monitoring system. In: Proceedings of international conference on modern problems of radio engineering, telecommunications and computer science, pp 464–467
11. Razmi SM, Saad N, Asirvadam VS (2010) Vision-based flame analysis using motion and edge detection. In: Proceedings of the IEEE international conference on intelligent and advanced systems (ICIAS), pp 1–4
12. Qiu T, Yan Y, Lu G (2011) A new edge detection algorithm for flame image processing. In: Proceedings of the IEEE international conference on instrumentation and measurement (I2MTC), pp 1–4
13. Bheemul HC, Lu G, Yan Y (2002) Three-dimensional visualization and quantitative characterization of gaseous flames. Meas Sci Technol 13:1643–1650
14. Ko BC, Cheong KH, Nam JY (2009) Fire detection based on vision sensor and support vector machines. Fire Saf J 44:322–329
15. Chi R, Lu ZM, Ji QG (2017) Real-time multi-feature based fire flame detection in video. IET Image Process 11:31–37
16. Bheemu HC, Lu G, Yan Y (2001) 3D monitoring and characterisation of gaseous flames using contour extraction methods. In: Proceedings of conf on sensors and their applications XI, pp 383–389
17. Toreyin BU, Dedeoglu Y, Cetin AE (2005) Flame detection in video using Hidden Markov models. In: Proceedings of IEEE international conference on image processing, pp 1230–1233
18. Wan XF, Huang DS, Xu H (2010) An efficient local Chan-Vese model for image segmentation. Pattern Recognit 43:603–618
19. Jiang Q, Wang Q (2010) Large space fire image processing of improving canny edge detector based on adaptive smoothing. In: Proceedings of international conference on CICCITOE. 2010, pp 264–267
20. Qiu T, Yan Y, Lu G (2012) An autoadaptive edge-detection algorithm for flame and fire image processing. IEEE Trans Instrum Meas 61:1486–1493
21. Gupta P, GaidhaneVH (2014) A new approach for flame image edges detection. In: Proceedings of IEEE ICRAIE, pp 1–6
22. Yan Y, Lu G, Colechin M (2002) Monitoring and characterization of pulverized coal flames using digital imaging techniques. Fuel 81:647–655
23. Roddy D (2004) Advanced power plant materials, design and technology. Woodhead Publications, Cambridge
24. Lu G, Yan Y, Colechin M (2004) A digital imaging based multifunctional flame monitoring system. IEEE Trans Instrum Meas 53:1152–1158
25. Gonzalez RC, Woods RE (2009) Digital image processing. Pearson Prentice Hall Publications, 3nd Edition
26. Stark JA (2000) Adaptive image contrast enhancement using generalizations of histogram equalization. IEEE Trans Image Process 9:889–896
27. Cheng HD, Shi XJ (2004) A simple and effective histogram equalization approach to image enhancement. Digit Signal Process 14:158–170
28. Guo Z, Zhang D (2010) A completed modeling of local binary pattern operator for texture classification. IEEE Trans Image Process 19:1657–1663
29. Singh V, Gupta I, Gupta HO (2007) ANN-based estimator for distillation using Levenberg–Marquardt approach. Eng Appl Artif Intell 20:249–259
30. Gaidhane VH, Hote YV, Singh V (2016) Emotion recognition using eigenvalues and Levenberg Marquardt algorithm-based classifier. Sadhana 41:118–123