SURE-Based Filter For Flame Detection

Asha Joseph, Neethu Roy

Abstract— Flame detection is very important in fire alarm systems. To detect the flame or fire correctly its features should be extracted. Feature extraction will be correct only if we remove the noise from the flame image. We propose a SURE- based filter for removing the noise. It is a linear filter that preserves the edges of images and thus keeps the fine details and geometrical structure of the image. After the noise removal the features like optical mass transport and non smooth data of flame are extracted. These features are compared with the initially trained flame features and confirmed whether the input frame contains flame or not. Experimental results shows that noise removal is very effective and the detection of flame is very easy.

Index Terms— Feature extraction, Filtering, Flame detection, Noise removal, Optical Flow, Optimal Mass Transport, Non Smooth Data.

1 INTRODUCTION

There is a greater importance of fire alarms in public places, institutions, organizations etc. Purpose of this system is to produce alarms when there is fire attack. It should produce positive alarms. Some of these alarms produces false detections. To prevent it, it is needed to detect the flame correctly. The image captured by surveillance camera may be affected with noise. Features extracted from noisy image need not to match with the features of previously trained flame images. So for correct flame/fire detection noise is to be removed. There are many traditional methods of noise removal.

L.Rudin et al. [1] proposed nonlinear total variation based noise removal algorithms. The total variation of the image is minimized subject to constraints involving the statistics of the noise. Solution is based on a time dependent partial differential equation. It is Implemented as an iterative process which is usually slow. C.Tomasi et al.[2] introduced bilateral filtering for gray and color images. Bilateral filtering smoothens images while preserving edges, by means of a nonlinear combination of nearby image values. Implementation of bilateral filter is known to be slow . K.He[3] proposed another noise removal method which is guided image filtering. The guided filter generates the filtering output by considering the content of a guidance image, which can be the input image itself or another different image. Computational complexity is independent of the filtering kernel size. It may have halos near some edges. So to overcome these limitations SURE-based filtering is used.

Steins Unbiased Risk Estimate (SURE) is an edge preserving smoothing filter. It is a linear-time algorithm which can be used for many image processing tasks and helps for the feature extraction.

This paper is organized as follows. Section II deals with the SURE based method for noise removal. Section III handles the method for flame detection, section IV discusses feature extraction , section V explains classification via neural network and experimental results are shown in section VI. Finally in section VII concluding remarks are given

2 SURE BASED FILTER

The measurement model is considered as

$$y_i = x_i + n_i, \quad i=1\ldots\ldots\ldots N \tag{1}$$

x_i is the signal of interest at position i, n_i is the zero mean corrupting Gaussian noise with variance $\sigma^2$and y_i is the noisy measured signal[4]. The standard simplified denoising problem is to find good estimate $\hat{x}$ of x=$[x_1 \ldots\ldots\ldots \quad x_N]^T$ from the corresponding data set y= $[y_1 \ldots\ldots\ldots \quad y_N]^T$. The denoising performance is measured using peak signal-to-noise ratio(PSNR). Higher PSNR shows that the reconstruction is of higher quality. So our aim is to increase the PSNR or decrease the mean square error(MSE). MSE can be estimated by SURE which needs noisy image only. It does not need prior knowledge of unknown original signal. So SURE is seem to be a flexible method.
MSE of the denoised image with respect to noise free version is[4]

$$\text{MSE}[\hat{x}] = 1/N\, ||x - \hat{x}||^2 = 1/N \sum_{j=1}^{N} (||x_j - \hat{x}_j||^2 \tag{2}$$

Where $||.||^2$ is the Euclidean norm. PSNR is measured as

$$\text{PSNR} = 10 log_{10}\ (\text{MAX}(x^2) / \text{MSE}(\hat{x})) \tag{3}$$

It is possible to replace MSE by SURE without any assumptions on the original signal. It is specified by the following expression[4].

- *Asha Joseph is currently pursuing masters degree program in computer science with specialization in information system in M.G University, India. E-mail: ashajmanavalan@gmail.com*
- *Neethu Roy is currently working as Assistant Professor in ICET College,M. G University, India. E-mail: neerhuroy19@mail.com*

$$\text{SURE}(\hat{x}) = 1/N\,||y - \hat{x}||^2 + 2\sigma^2/N\,\text{div}_y\{\hat{x}\} - \sigma^2 \quad (4)$$

$$\text{Where} \quad \text{div}_y\{\hat{x}\} = \sum_{j=1}^{N} \partial\hat{x}_i / \partial\hat{y}_i \quad (5)$$

We start our procedure by dividing our image into patches and filtration is performed and then all filtered output image patches are averaged together to obtain the final filtered result.

Let w_i be a local window around the position i, y_w and $\hat{x}_w$ be an input image patch and a filtered output image patch respectively corresponding to the window w_i. Here, for simplicity, we consider w_i as a square window of pixels with a fixed size. Let

$$\hat{x}_{w_i} = a_i\, y_{w_i} + b_i, \qquad a_i, b_i \in R, a_i \geq 0 \quad (6)$$

where a_i and b_i are some affine transform coefficients assumed to be constants in window w_i. Thus, for any $j \in w_i$, we have $\hat{x}_j = a_i y_j + b_i$. Notice that we restrict a_i to be non-negative for preventing the filtered output results far from the input data. Plugging (6) into (4) we have

$$\text{SURE}(a_i, b_i) = 1/N_w\,||y_{w_i} - (a_i\, y_{w_i} + b_i\,||^2 + (2\sigma^2/N_w)\text{div}_y\{a_i\, y_{w_i} + b_i\} - \sigma^2$$

$$= 1/N_w||y_{w_i} - (a_i\, y_{w_i} + b_i\,||^2 + (2\sigma^2/N_w)a_i N_w - \sigma^2$$

$$= 1/N_w||y_{w_i} - (a_i\, y_{w_i} + b_i\,||^2 + (2\sigma^2\, a_i - \sigma^2 \quad (7)$$

Where $N_w = (2r+1)*(2r+1)$ is the pixel number in window w_i and r is the window radius .We can compute a_i *and* b_i by taking the first derivative of (7)

$$a_i = (\sigma_i^2 - \sigma^2) / \sigma_i^2$$

$$b_i = (1 - a_i)\,\bar{y}_i \quad (8)$$

Where $\bar{y}_i$ and σ_i^2 are the mean and variance respectively of input data in local window w_i. Since a_i is restricted as non negative, the optimal transform coefficients are determined finally as follows

$$a_i^* = \max(\sigma_i^2 - \sigma^2, 0) / (\sigma_i^2 + \varepsilon)$$

$$b_i^* = (1 - a_i^*)\,\bar{y}_i \quad (9)$$

where ε is a small constant number avoiding the denominator equal to zero. From (6) and (9) filtered output in local window w_i is

$$\hat{x}_j^{\,i} = a_i^*\, y_j + b_i^*, \; j \in w_i \quad (10)$$

As the window moves on the entire image transform coefficient will be as in (9) and filtered output will be as in (10). There will be an overlapping region when the window moves to entire region because we are considering the square window. So we obtain multiple estimates for pixels lies in the overlapping region. It is needed to aggregate these multiple estimate to get a single result for the overlapping region. Aggregation is done by weighted average of multiple estimates.

$$\hat{x}_j = \Sigma i \in w_j\, \lambda_i\, \hat{x}_j^{\,i} \quad (11)$$

Risk of estimate $\hat{x}_j$ is

$$R(\hat{x}_j, x_j) = E\,[(\hat{x}_j - x_j)^2] \quad (12)$$

Where E is with respect to the probability measure associated with the noise. Risk can be expressed in bias squared and variance terms as

$$R(\hat{x}_j, x_j) = E[(\hat{x}_j - E[\hat{x}_j] + E[\hat{x}_j] - x_j)^2]$$

$$= \text{var}(\hat{x}_j) + \text{bias}^2(\hat{x}_j) \quad (13)$$

To minimize the risk of $\hat{x}_j$ we need to minimize the variance.

$$\text{var}(\hat{x}_j) = \text{var}(\Sigma i \in w_j\, \lambda_i\, \hat{x}_j^{\,i})$$
$$= \Sigma i \in w_j\, \lambda_i^2\, \text{var}(\hat{x}_j^{\,i}) \quad (14)$$

$\Sigma i \in w_j\, \lambda_i = 1$. Using Lagrangian equation this can be as

$$\lambda_i = \text{var}^{-1}(\hat{x}_j^{\,i}) / \Sigma k \in w_j\, \text{var}^{-1}(\hat{x}_k^{\,i}) \quad (15)$$

This kind of aggregation procedure is called Variance based Weighted Average (WAV).

Final filtered output is

$$\hat{x}_j = \Sigma i \in w_j \lambda_i \hat{x}_j^{\,i}$$
$$= \Sigma i \in w_j\, (\sigma_i^{-2})/(\Sigma k \in w_j\, (\sigma_k^{-2}))(a_i^*\, y_j + b_i^*)$$
$$= 1/\Sigma k \in w_j\, (\sigma_k^{-2}) \Sigma i \in w_j\, (\sigma_i^{-2}\, a_i^*\, y_j + (\sigma_i^{-2}\, b_i^*)$$
$$= \bar{a}_j\, y_j + \bar{b}_i \quad (16)$$

Where $\bar{a}_j = (1/w_j)\, \Sigma i \in w_j\, (\sigma_i^{-2}\, a_i^*$
$\bar{b}_j = (1/w_j)\, \Sigma i \in w_j\, \sigma_i^{-2}\, b_i^*$ and $w_j = \Sigma i \in w_j\, \sigma_i^{-2}$

This linear filter preserves the edge and smoothness. If the local variance of a window is not greater than the entire noise variance, then it is a flat region and contains no edge. That is if $\sigma_i^2 - \sigma^2 \leq 0$ then $a_i^* = 0$ and $b_i^* = \bar{y}_i$ its value is average of the pixels nearby. If $\sigma_i^2 - > \sigma^2$ then a_i^* becomes close to 1 and b_i^* becomes close to 0. That means location i is a high variance region or an edge. So its value remain unchanged. Noise variance is an important threshold that determines whether it is an edge or not. Taking of window size is the another important thing. If we choose a large window size then the region that contain edge will become large and denoising performance will be poor. If window size is too small then it will lead to unwanted block like artifacts. Experimental research shows that window size r=2 or r=3 is an optimal choice.

2.1. Extended SURE method

If we are having two undesirable images of the same scene then we create a satisfactory image by combining them. For that we extended the SURE approach. After the Extended SURE (E-SURE) approach we get more smoothed image[4].

$$\text{E-SURE}(a_i, b_i) = 1/N_{w_i} \|f_{w_i} - (a_i y_{w_i} + b_i\|^2 + (2\sigma^2/N_w) \mathrm{div}_y\{a_i y_{w_i} + b_i\} - \sigma^2$$

$$= 1/N_w \|f_{w_i} - (a_i y_{w_i} + b_i\|^2 + (2\sigma^2 a_i - \sigma^2 \quad (17)$$

Where f_{w_i} and y_{w_i} are the filter input patch and the guidance image patch corresponding to the window w_i. From (17) we get

$$a_i = (\mathrm{cov}(f_{wi}, y_{wi}) - \sigma^{2/}) / \sigma_i^2$$

$$b_i = \bar{f}_i - a_i \bar{y}_i \quad (18)$$

Where $\bar{f}_i$ is the mean of input signal f in local window w_i i , $\bar{y}_i$ and σ_i^2 are the mean and variance of guided signal y in window wi_i.

$$\mathrm{cov}(f_{wi}, y_{wi}) = (1/N_{w_i}) \sum_{j=1}^{N_{w_i}} f_{j_i} y_{j_i} - \bar{f}_i \bar{y}_i$$ is the covariance between f_{wi} and y_{wi}

$$a_i^* = \mathrm{soft}(\mathrm{cov}(f_{wi}, y_{wi}), \sigma^2) / (\sigma_i^2 + \varepsilon)$$

$$b_i^* = \bar{f}_i - a_i \bar{y}_i \quad (19)$$

Where $\mathrm{soft}(x,a) = \mathrm{sign}(x)\max(|x| - a, 0)$.
After finding a_i^* and b_i^* final filtered output is obtained by

$$\hat{f}_j = \sum_{i \in w_j} \lambda_i (a_i^* y_j + b_i^*) = \bar{a}_j y_i + \bar{b}_i \quad (20)$$

Where $\bar{a}_j = (1/w_j) \sum_{i \in w_j} \sigma_i^{-2} a_i^*$
$\bar{b}_j = (1/w_j) \sum_{i \in w_j} \sigma_i^{-2} b_i^*$ and $w_j = \sum_{i \in w_j} \sigma_i^{-2}$

(a) (b)
Fig 1. (a) Noisy image (b) Filtered image

3.FIRE / FLAME DETECTION

After filteration we check for whether the frame contain flame or not. Flame is recognized by optical flow estimation. Flame features such as Optimal Mass Transport and Non Smooth Data are extracted to confirm it as flame.

3.1 Optimal Mass Transport (OMT) optical flow

Optical flow computes the correspondence between pixels in the current and the previous frame of an image sequence. Classical optical flow assumes that moving objects preserve their intensity constancy. This assumption leads to[5]

$$(d/dt)I = I_x u + I_y v + I_t = 0 \quad (21)$$

Where flow vector $(u,v) = (x_t, y_t)$ points into the direction where the pixel (x,y) is moving. We can not stick on this concept because of two reasons. One is fire does not satisfy intensity constancy assumption. That is change of intensity occurs during the burning process due to fast pressure and heat dynamics. Second is related to smoothness regularization. That is fire is having non smooth motion field.

Generalized mass in this context is image intensity. Mass conservation law is written as

$$It + \nabla \cdot (\mathbf{u} I) = 0 \quad (22)$$

Where $\hat{u} = (u, v)^T$ .Equation (21) is known as continuity equation. Conservation of mass together with conservation of momentum and conservation of energy gives motion of inviscid fluids such as fire. OMT optical flow minimizes the total energy defined as

$$\min_{\hat{u}} (1/2) \int_0^T \int (It + \nabla \cdot (\mathbf{u} I))^2 + \alpha \|\hat{u}\| I \, dt \, dx \, dy \quad (23)$$

Equation (22) is a hard constraint. A solution to this problem is a” discretize – then - optimize” approach. That is

$$\min_{\hat{u}} (\alpha/2) (\hat{u}^T \Gamma \hat{u}) + (1/2) (It + [Dx I \;\; Dy I] \hat{u})^T (It + [Dx I \;\; Dy I] \hat{u}) \quad (24)$$

where $\hat{u}$ is acolumn vector containing u and v and Γ is amatrix containing average intensity values (I0+ I1)/2 on its diagonal. The derivatives are discretized by It = I1– I0 and the central difference sparse matrix derivative operators Dx and Dy. Quantities b=-1, A=[Dx I Dy I] are defined . Minimized equation becomes

$$\min_{\hat{u}} (\alpha/2)(\hat{u}^T \Gamma \hat{u}) + (1/2) (A\hat{u} - b)^T(A\hat{u} - b) \quad (25)$$

$$\hat{u} = (\alpha \Gamma + A^T A)^{-1} (A^T b) \quad (26)$$

3.1.1 Flame color as generalized mass

Generalized mass is the pixel intensity I. It is represented in HSV color space. (H,S,V ∈ [0,1]) which is chosen to be H_c = 0.083 , V_c =S_c= 1 a fully color saturated and bright orange. Generalized mass is computed as[5]

$$I = f (\min \{ |Hc - H|, 1 - | Hc - H | \}) \cdot S \cdot V \quad (27)$$

Where is the logistic function

$$f (x) = 1 - (1 + \exp(-a .(x-b))) \quad (28)$$

(a) (b) (c) (d)

Fig.2.Two examples for the generalized mass transformation Eq. (9). (a) and (c): Original images.

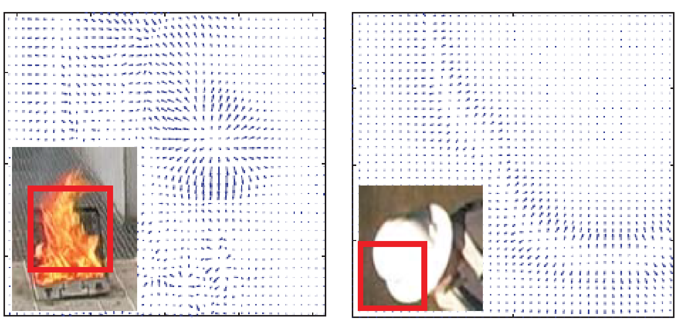

Fig 3. OMT flow fields: fire with dynamic texture (left) and a white hat (right) moving up/right. The red box indicates the area for which the flow field is shown.

3.2 Non-Smooth Data (NSD) Optical Flow

Fire blobs are saturated under unfavorable lighting conditions thus violating the concept of dynamic texture of fire. Such fire will be of non smooth condition. Non Smooth Data optical flow is tailored to saturated fire blob.

$$\min_{\hat{u}} (1/2) \iint (It + \nabla \cdot (\mathbf{u}I))^2 + \alpha \, || \hat{u} || \, I \, dt\, dx\, dy \quad (29)$$

Minimization of equation (29) can be as

$$\frac{\partial l}{\partial u} = It + \nabla \cdot (\mathbf{u}I)\, Ix + \alpha u = 0 \quad (30a)$$

$$\frac{\partial l}{\partial v} = It + \nabla \cdot (\mathbf{u}I)\, Iy + \alpha v = 0 \quad (30b)$$

$$u= - (Ix\, It) / ||\nabla I|| + \alpha \qquad v= - (Iy\, It) / ||\nabla I|| + \alpha$$

Again, *Ix* , *Iy*, and *It* are pre-computed image derivatives andα is the regularization parameter.

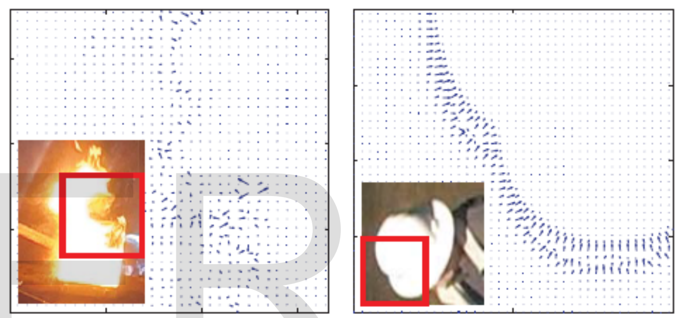

Fig.4 NSD flow fields: saturated fire (left) and a white hat (right) moving up/right. The red box indicates the area for which the flow field is shown.

4. FEATURE EXTRACTION

Following four features are extracted and compared with the trained flame image features. If it matches then it is clear that the input frame contains flame contents otherwise not.

4.1 OMT Transport Energy: Mean OMT transport energy per pixel is

$$F1= \text{Mean}(½ \, || \hat{u}\, OMT||) \quad (31)$$

After the color transformation fire and other moving objects in the fire-colored spectrum produces high value for this feature.

4.2 NSD Flow Magnitude : Mean of the regularization term of NSD optical flow energy constitute

$$F2= \text{Mean}(1/2|| \hat{u}\, NSD ||) \quad (32)$$

F1 and f2 will have high values for fire colored moving objects. Last two faetyres distinguish fire from other rigid objects.

4.3 OMT Sink/Source Matching : The third feature is designed to quantify how well an ideal source flow template matches the computed OMT flow field. The template is defined as

$$\hat{u}t(x,y) = (ut(x,y)\ vt(x,y)) = \exp\left(-\sqrt{x^2+y^2}\right)(x\ y) \qquad (33)$$

$$F3 = \max\left|\,(ut * u\,OMT / \hat{u}\,OMT) + (vt * v\,OMT / \hat{v}\,OMT)\right| \qquad (34)$$

Where * denotes convolution

4.4 NSD Directional Variance: It distinguishes boundary motion of saturated fire blobs from rigidly moving objects.Implementation is done by estimating histogram.

$$h(r, \varphi) = \mathrm{Hist}(\hat{u}\,NSD) \qquad (35)$$

Where $r = \sqrt{u^2+v^2}$ and $\varphi = \arctan(v/u)$

$$F4 = \mathrm{var}\{s_i,\ i = 0 \ldots n-1\}$$

$$S_i = \left(\int_0^{Inf} \int_{2pi/n}^{2pi(i+1)/n} h(r,\varphi)\, d\varphi\, dr\right) \Big/ \left(\int_0^{Inf} \int_0^{2\,pi} h(r,\varphi)\, d\varphi\, dr\right) \qquad (36)$$

Feature *f*4 is more versatile for detecting rigid motion

5.CLASSIFICATION VIA NEURAL NETWORK

Better results can be achieved by learning the classification boundary with a machine learning approach such as neural networks. Training the neural network (NN) means performing a non-linear regression in the feature space to best separate the labeled training data into classes. Features extracted are trained using NN. For the input frames features are compared with these trained features and checked the similarity. If it matches we can conclude that it is a flame image.

6.EXPERIMENTAL RESULTS

A set of 100 noisy images of size 320 × 180 were given as input. From that we could easily remove the noise in seconds. Features were extracted from those noise free images and checked for the flame content in it. We could successfully classify the flame and flameless images from that set. 0.046478 seconds is the time taken to show it as a flame image after its features are analyzed. For the following image we got the MSE and PSNR as follows

MSE	0.0145
PSNR	41.5333

Table 1 shows MSE and PSNR values of given frame

(a) (b)

Fig 5. (a)shows input frame (b) shows its filtered output

7. CONCLUSION

A linear SURE based filter is used to denoise the flame images for flame detection. It is an edge preserving smoothing filter preserves the fine and geometrical characteristics of objects. The method obtains least mean square error and good PSNR value proves its filtering efficiency. After denoising its features are extracted and compared with previously trained flame features. Optimal Mass Transport and Non Smooth Data optical flow are the most important flame features that are analysed to confirm it as flame. This work can be used in fire alarm systems which is free from false fire detection. Further research is needed for the real life flame or fire images.

ACKNOWLEDGEMENT

The authors would like to thank the ICET college authority and the guide for their valuable support. They would also like to thank the reviewers for their valuable feedback and suggestions

REFERENCES

[1]. L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, vol. 60, pp. 259–268, Nov. 1992.*

[2]. C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. 6th Int. Conf. Comput. Vis., Bombay, India,* pp. 839–846, *Jan. 1998*.

[3] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proc. Eur.Conf. Comput. Vis.,pp. 1–14, 2010.*

[4]. Tianshuang Qiu, Aiqi Wang, Nannan Yu, and Aimin Song,"LLSURE: Local Linear SURE-Based Edge-Preserving Image Filtering", IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 22, NO. 1, JANUARY 2013

[5]. Martin Mueller, Peter Karasev, Ivan Kolesov, and Allen Tannenbaum "Optical Flow Estimation for Flame Detection in Videos" , IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 22, NO. 7, JULY 2013