IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-4, Issue-5 E-ISSN: 2347-2693

An Efficient Framework for Fire Detection using Morphological Features

Mangesh S. Tambat[1*], Namrata Kodre[2], Shubhangi Shelke[3], Kimaya Chavan[4], Prof. Laxman Deokate[5]

[1,23,4,5]Dept. of IT, SAOE Pune-48, SPPU, India

Available online at: www.ijcseonline.org

Received: Apr/11/2016 Revised: Apr/19/2016 Accepted: May/14/2016 Published: May/31/2016

***Abstract*—** This paper gives the one of the best solution for the video surveillance in the fire detection. In the market there are most popular two software tools are used to detect the fire and smoke that are "VPlayer" for the fire and smoke detection and another one is "Precise Vision Fire Detection Graphics System" is these system one of the major drawback comes that is, user can get some times false positive result. And in this proposed system we try to reduce the false positive result. And in this system the technique involves fire features, fuzzy logic. And this proposed system is totally software based not any embedded system is used here.

***Keywords*—** RGB Model, Temperal Difference, Fire Morphology, Fuzzy Logic

I. INTRODUCTION

In the traditional approach firstly used the sensors for the fire detection, then after on words some changes are done with the various system approaches. Now a days the world become degitalized and used the everywhere digital devices hence video camera is common for the video surveillance. With the help of this approach that is video camera fire can detect integration with some logic and the model. In the proposed system we used the RGB model for the image feature extraction. And we used some of the filters with the RGB color model to reduce the false positive results. And fuzzy logic is used according to the color, shape and motion of the image. There are so much existing systems are working on the many models such as RGB, HSV, HIS, YCbCr . User can some amount of the error rate with this system, hence we upgrade the system with filter. User gets the correct result when the object is burning.

II. LITERATURE SURVEY

In the paper [1] the existing system RGB, YC_bC_r model are used to detect the fire. This fire detection system is works on the detection of fire using color and motion information. And the proposed system is works on the three of important factor that is color, shape & motion. Shape is also one of the most important factor for the fire detection. Due the shape, we can say that how big is the fire, how harmful & dangerous it is.

In this paper [2] the author is to detect and categorized hydrocarbon fires in aircraft dry bays and engine compartments. This system works on histogram and successive difference between the frames. In this proposed system there is also used a new one component that is the framegabber. In the framegabber there is collection of frames. The new one frame is compared with the old one & user can get the difference between them and get the result if there is actual fire is present or not.

In this paper [3] system is based on the motion or color. There is ordinary motion, color flame and fire clicker is detected. But one of the most advantage other this system is, we use results according to the shape. It increases the accuracy of the proposed system.

In this paper [4] this approach is based on the first order local differentiation method. There is pyramid technic is used to extract the smoke, motion features and gives the truth table based result on neural network. In the previous systems of this existing system there is high false positive result get to the user because of the some moving objects. Hence in the existing system there is not any moving objects are taken except the fire. In the existing system used to reduce the high false positive result with the help of optical flow method and pyramid technique. And in the proposed system there is filter concept is integrate with the RGB model and due to the used, it reduces the false positive results. And in this proposed system there is not any moving objects is taken only processing on the fire.

In the paper [5] author is used the RGB model in his system. But if we check the input video with their method with a pre partitioning scheme under some conditions, it is not able to refuse the similar fire such as alias. But in out proposed system comes with combination of the three main and important fact that is shape, color and motion by applying some algorithms mentioned in the proposed system section.

In the paper [6] the existing system extract the image on the basis of the hue and saturation. User can get the appropriate result. And in the proposed system there is each and every frame can compared with the previous one and we get the corrected result as compared to the existing system.

In the paper [7] the existing system is works on the HIS model for the detection of the fire flame and the feature extraction of fan image. This existing system is works on the color hypothesis. According to the color it gives the result. In this system color is roughly separated from a test image by separation methods. And according to the original fire and captured images are compared and gives the satisfactory result to the user. And the proposed system is also based on the same concept calculate the differences between the frames and gives the mentioned and corrected result to the user.

In the paper [8] the working of the existing system is totally depends of the "color", with the help of the some fuzzy logic user gets the result. This approach is very complicated in the real time, it requires the large amount of memory. And one of the most disadvantage is that is this system gives a high false alarm rate. Many of the objects color is same as the fire flame such as sun and light. To such problem related to the nature things many researchers provides their own solution over it. And due to avoid or resists such situation user gets the correct for of result.

In the paper [9] such existing system is works on the fire temperature and is related with color of that captured image. Actually when the fire temperature is low then the color changes from "red to yellow" , and when the temperature is high it becomes "white". This system is based on the 4 components such as:

1. Fire-colored pixel detection.
2. Moving pixel detection.
3. Color image segmentation.
4. Integration of spatial and temporal features.

There is one of the most important issue comes in the existing system that is related to the flicker. It can't be detect the rapid variation in the fire flame. And in the proposed system there is the some 2-3 second time interval proper images are captured and we get the result.

III. PROPOSED SYSTEM

This section reveals all the technique and methods which are being deployed for the detection of fire from the live video. Below mention steps represents the fire detection technique that our system incorporate as shown in the figure 1.

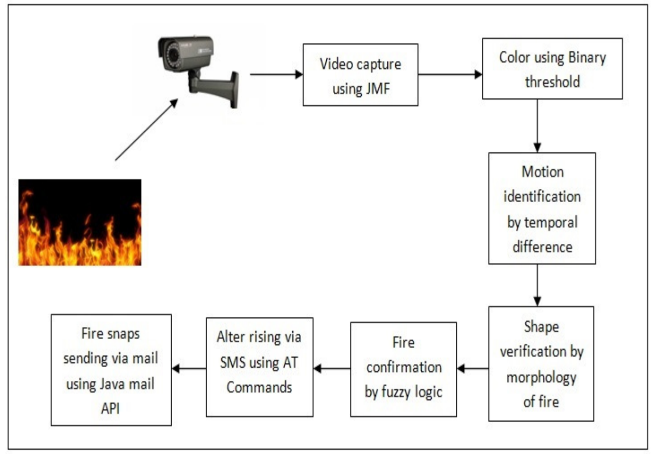

Figure 1: System Overview

Step 1:

- *This is the step of configuring the hardware webcam with our program. This process is successfully carried out by using a third party API called JMF(Java Media Files),which eventually helps to grab the live videos from the internal or external webcam attached to the system.*

- *Then by using frame grabbing technique relative frame from the video is been captured continuously in JPEG format for the set time in seconds. Finally these frames are then used to identify the fire with the below mention steps:*

Step 2:

- The successfully captured frame from the prior step is being used to identify the fire using color as its primary components. For this process our system uses a heuristic approach of converting the image into gray scale by using mean value of the RGB color components of the pixel. In the very next step this mean value of RGB is been verified for the threshold value of the brightness that eventually indicates the fire color (The threshold value is generally set more than 180).

- Whichever the pixels are crossed these threshold are tagged as fire pixel and then finally the count is been taken for these fire pixels. If the fire pixel count is greater than the threshold count set by our system based on the size of the image, then the image frame is labeled as fire containing frame. This step is been depicted by the below mentioned algorithm 1.

Algorithm 1: Gray scale Conversion and Binary threshold for Fire detection using color component

// Input: Video Frame **F**
// Output: Fire detected image

Step 0: Start
Step 1: Get Image path.
Step 2: Get Height and width of the Image **F** (L*W).
Step 3: FOR **i**=0 to width.
Step 4: FOR **j**=0 to Height.
Step 5: Get a Pixel at (i, j) as signed integer.
Step 6: Convert pixel integer value to Hexadecimal to get R, G, and B.
Step 7: **AVG**=(R+G+B) /3
Step 8: **IF AVG >T**
Step 9: Pixel at (i,j) is FIRE
Step 10: **ELSE**
Step 11: Pixel at (i, j) is NOT FIRE
Step 9: End of inner for
Step 10: End of outer for
Step 11: Stop

Step 3:

- This is the step where our proposed system identifies the shape of the fire by using co axial variance technique, Where our system keeps checking the ratio of the fire pixels which is been identified by the past step. The ratio is identified using the following two equations (1) and (2) for every pixel. And the stream of this ratio eventually indicates the shape vector or the morphology vector of the fire.

$$M(x) = \sum_{i=1}^{N} P(i,j)/WIDTH \qquad (1)$$
$$M(y) = \sum_{i=1}^{N} P(i,j)/HEIGHT \qquad (2)$$

Where M(x) – Morphology vector related to X axis.
M(y) – Morphology vector related to Y axis.
P(i,j) – Pixel at position i and j
N – Number of pixels in the image

Step 4:

- Here in this step for every given time T, grabbed frame is been assigned to the past frame for the motion detection of the fire. In this process for every time the difference between the current and the past frame is been calculated for the fire pixels which was identified through the color parameter in the prior steps. If the difference is crossed the threshold value then the frame is been labeled for the fire image. This process can be depicted in the figure 2.

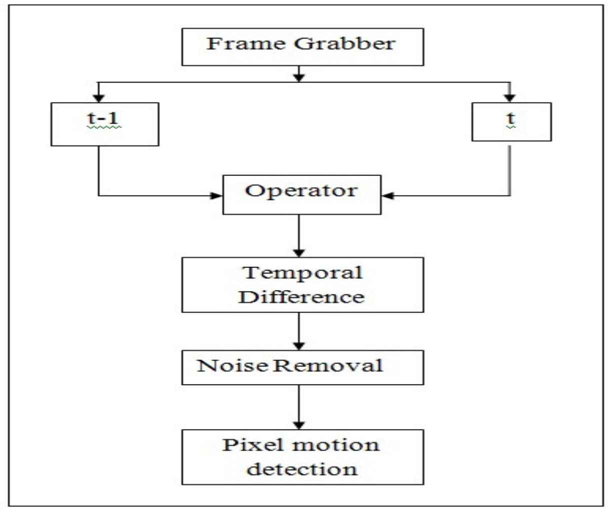

Figure 2: Overview of Fire detection by motion

The algorithm 2 clearly indicates the details of this step as follows.

Algorithm 2: Fire Detection by motion.
// Input: Time **T**, Frame $\mathbf{F_c}$, Frame $\mathbf{F_p}$, Threshold Fire pixels $\mathbf{T_h}$
// Output: Fire Detection through motion
Step 0: **Start**
Step 1: **WHILE** (TRUE)
Step 2: for each time **T**
Step 3: $F_p \rightarrow F_c$
Step 4: calculate pixel positions of $\mathbf{F_p}$ in an vector $\mathbf{V_p}$
Step 5: calculate Pixel positions of $\mathbf{F_c}$ in an vector $\mathbf{V_c}$
Step 6: IF ABSOLUTE DIFF ($\mathbf{V_p} - V_c$) > $\mathbf{T_h}$
Step 7: Label Frame for Fire
Step 8: **END IF**
Step 9: **END WHILE**
Step 10: Stop

Step 5:

- This is the last step of our system where false positivesness can be reduce by using fuzzy logic. This process receives the all the three parameters from the past three steps. That are fire detection by color, motion and shape, the received parameters are been tagged between the value 0 and 1.

- So by using the Fuzzy crisp values, which are divided in between the ranges as follows

  - ✓ VERY LOW – 0 TO 0.2
  - ✓ LOW -- 0.21 TO 0.4
  - ✓ MEDIUM -- 0.41 TO 0.6
  - ✓ HIGH _ 0.61 TO 0.8
  - ✓ VERY HIGH --0.81 TO 1.0

- So any frame whose mean parameter values falls in between the HIGH and VERY HIGH is considered as the fire frame and then the relevant alarm will be raised by the system. The parameters which are come under the range VERY LOW, LOW and MEDIUM will represent the false fire detected frames.

Step 6: Sending alert to fire department via SMS

- Here in this step a SMS will be sending to the fire department through a GSM modem connected to the system using Attention commands and Java Communication API.

Step 7: Sending Fire snaps to fire department via mail

- Here in this step an Email will be sending to the concern person by using Java mail API which is powered by Gmail host.

IV. MATHEMATICAL MODEL

A. *Set Theory*

1. S= { } be as system for fire detection

2. Identify Input as V= {V_1, V2..... V_n}
   Where V_n= Frame numbers

3. Identify F as Output i.e. Fire detection
   S= {V_n, F}

4. Identify Process P

   S= {V_n, F, P}
   P= {V_f, C_i M_i S_i F_l}
   Where V_f = Video frame
   C_i =Color identification
   M_i =Morphology identification
   S_i = Shape identification
   F_l = Fuzzy logic

5. S = { V_n, F, V_f, C_i M_i S_i F_l }

B. *Set Description:*

1. Video frame:
   Set V_f:
   V_{f1} = Capturing camera content
   V_{f2}= Adding to video track
   V_{f3}= Grab frame content
   V_{f4}= prepare image object

2. Color identification:

   Set C_i:
   C_{i1}= Read RGB value in pixel
   C_{i2}= Convert into YCbCr model
   C_{i3}= Calculate component difference
   C_{i4}=Check for the component rule
   C_{i5}= Identify fire pixel

3. Morphology identification:

   Set M_i:
   M_{i1}= Get image height and weight
   M_{i2}= Get RGB values of pixel
   M_{i3}= Get the axis distance
   M_{i4}= Calculate co-axial ratio
   M_{i5}= Create morphological vector

4. Shape identification

   Set S_i:
   S_{i1}= Grab a frame in given time
   S_{i2}= Calculate RGB values
   S_{i3}= Label pixel position
   S_{i4}=Compare pixel position with next frame
   S_{i5}=Identify temporal difference

5. Fuzzy logic

   Set F_l:
   F_{l0}=Crisp values
   F_{l1}=Fuzzyfier
   F_{l2}=Defuzzyfication
   F_{l3}=If-then Rules
   F_{l4}=Summary

C. *Representation of Sets and its operation:*

1. Union Representation:

   1.1 Set V_f = { V_{f1}, V_{f2}, V_{f3}, V_{f4}}
   Set C_i = {C_{i1}, C_{i2}, C_{i3}, C_{i4}, C_{i5}}
   Set (V_f U C_i) = {V_{f1}, V_{f2}, V_{f3}, V_{f4}, C_{i1}, C_{i2}, C_{i3}, C_{i4}, C_{i5}}

   1.2 Set S_i ={ S_{i1}, S_{i2}, S_{i3}, S_{i4}, S_{i5}}
   Set (V_f U C_i U S_i) = { V_{f1}, V_{f2}, V_{f3}, V_{f4}, C_{i1}, C_{i2}, C_{i3}, C_{i4}, C_{i5}, S_{i1}, S_{i2}, S_{i3}, S_{i4}, S_{i5}}

1.3 Set F_l = { $F_{l0}, F_{l1}, F_{l2}, F_{l3}, F_{l4}$}

Set (V_f U $C_i S_i$ U F_l) = { $V_{f1}, V_{f2}, V_{f3}, V_{f4}, C_{i1}, C_{i2}, C_{i3}, C_{i4}, C_{i5}, S_{i1}, S_{i2}, S_{i3}, S_{i4}, S_{i5}, F_{l0}, F_{l1}, F_{l2}, F_{l3}, F_{l4}$}

D. Venn Diagram of Union of set V_f, C_i, Si and F_l.

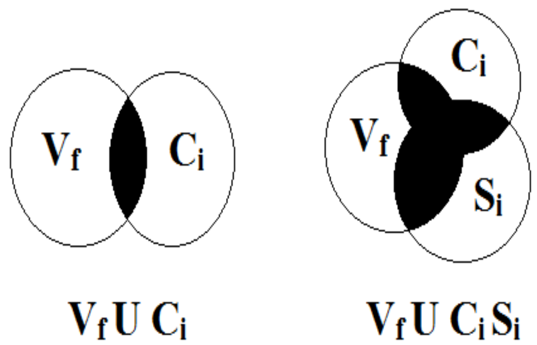

Figure 3: Relation between V_f, C_i and S_i

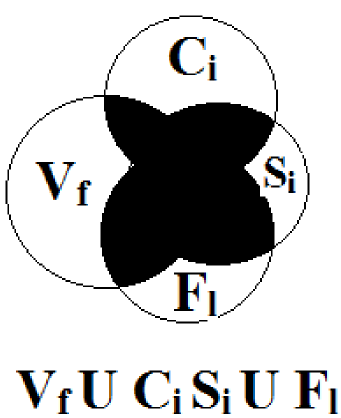

Figure 4: Relation between V_f, C_i, S_i and F_l

E. NP HARD:

1.1 Computations, Decisions and Languages:

The most common resource to analyze software is time and number of execution steps, this is generally computed in terms of n. We will use an informal model of a computer and an algorithm. All the definitions can be made precise by using a model of a computer such as a Turing machine.

While we are interested in the difficulty of a computation, we will focus our hardness results on the difficulty of yes - no questions. These results immediately generalize to questions about general computations. It is also possible to state definitions in terms of languages, where a language is defined as a set of strings: the language associated with a question is the set of all strings representing questions for which the answer is Yes.

1.2 The Class P:

The collection of all problems (Algorithms or methods that we are using in our project) that can be solved in polynomial time is called P. That is, a decision question is in P if there exists an exponent k and an algorithm for the question that runs in time $O(n^k)$ where n is the length of the input.

P roughly captures the class of practically solvable problems. Or at least that is the conventional wisdom. Something that runs in time 2^n requires double the time if one adds one character to the input. Something that runs in polynomial time does not suffer from this problem.

P is robust in the sense that any two reasonable (deterministic) models of computation give rise to the same definition of P.

1.3 The Class NP:

The collection of all problems that can be solved in polynomial time using non determinism is called NP. That is, a decision question is in NP if there exists an exponent k and an nondeterministic algorithm for the question that for all hints runs in time $O(n^k)$ where n is the length of the input.

1.4 P versus NP:

It would seem that P and NP might be different sets. In fact, probably the most important Unsolved problems in Mathematics and Computer Science today is:

Conjecture. $P \neq NP$

If the conjecture is true, then many problems for which we would like efficient algorithms do not have them. This would be sad. If the conjecture is false, then much of cryptography is under threat. Which would be sad.

1.5 NP Completeness:

While we cannot determine whether P = NP or not, we can, however, identify problems that are the hardest in NP. These are called the NP-complete problems. They have the property that if there is a polynomial-time algorithm for any one of them then there is a polynomial-time algorithm for every problem in NP.

1.5.1 Definition:

A decision problem S is defined to be NP-complete if

a) It is in NP; and

b) For all A in NP it holds that $A \leq_P S$.

1.5.2 Note that this means that:

- If S in NP-complete and S in P, then P=NP.
- If S is NP-complete and T in NP and $S \leq_P T$, then T is NP-complete.

1.5.3 Example:
We can state this even in simpler form, like as shown below
Let we consider a module V_f (Video frame) in our system called S,
Then If V_f is set to change in time T
V_f' (changed module) will be the changed module
If (V_f' ∈ S) ≤ T
Then system is considered as NP Complete. Our System unconditionally satisfies this problem, so we can conclude our system as NP Complete.

V. RESULTS AND DISCUSSIONS

To show the effectiveness of the proposed system some experiments are conducted on java based windows machine using Netbeans as IDE. To measure the performance of the system we set the bench mark by considering the system with more number of fire images.

We collected over 50 images from internet as of now and these data were considered to be the source database for the evaluation experiment.

For evaluation the system is required to submit a ranked list of five opinions to a fire image. Each image received a score equal to the inverse of the rank at which the first correct opinion was found.

That is called the Reciprocal Rank (RR), that the values of RR are 1,1/2,1/3,1/4,1/5,0. E.g., if a correct rank appears on the second rank, then it is one over two, so the score will be 0.5, etc. If none of the top five responses contained a correct rank, then the score was zero. The mean reciprocal rank (MRR) is the average score over all Fire Images.

$$MRR = \frac{\sum_{i=1}^{N} 1/Rank_i}{N}$$

Where, $Rank_i$ is the rank of the first correct occurrence in the top five ranks for Fire images i; N is the number of test images asked for the fire opinion asked; If for an image i, the correct rank is not in the top five responses then it is taken to be zero.

We performed an experiment to evaluate the rank retrieval using the MRR metric up to the top three responses, defined as follows. The result is shown in figure 6: Result of the Experiments.

As the result, the proposed method achieved average MRR of 0.775 for ranks of special images of fore in different forms like Traffic, Office, Home, Corridor and Greenery outdoor.

Fire image Type	Images	MRR
Traffic	25	0.875
Office	13	0.5
Home	15	0.5
Corridor	16	1
Greenery	20	1
	TOTAL =89	MEAN=0.775

Figure 5: Result of the Experiment

In the figure 6, we observe that the tendency of average MRR is 0.775 for the mentioned fire images types in figure 5. So this is actually a better performance in our very first attempt of fire detection system

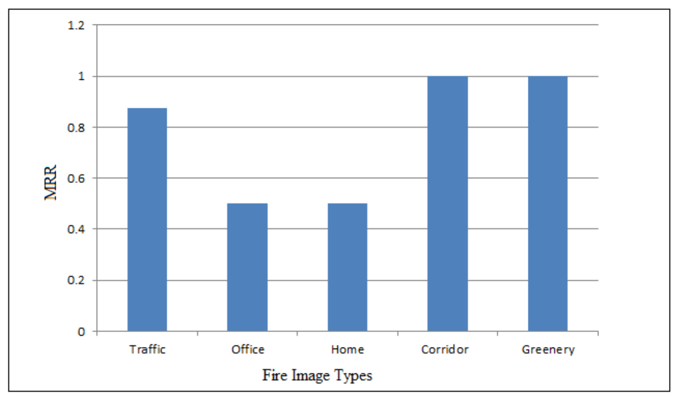

Figure 6: MRR of the Fire image Types

VI. CONCLUSION

In the system the presence of flame and fire is detected by monitoring the surveillance camera and collecting the information related to grey scale, color, shape and motion of fire. Major components of proposed fire detection system are: fire color detection, motion identification, shape verification and fuzzy logic approach.

The system is based on a combination of various methods used for fire detection using surveillance cameras. It delivers us facility to adjust the system by using different combinations of video image processing based fire detection methods and implementing the system according to different area requirement. It also provides us the best possible method for accurately detecting the flame and fire in terms of decreased false fire detection rate and hence increasing the accuracy of the system.

VII. ACKNOLEDGEMENT

The Authors would like to tank Prof. Laxman J. Deokate for their assistance and guidance in preparing this manuscript.

VIII. REFERENCES

[1] W. Phillips III, M. Shah, and N.V.Lobo, “Flame Recognition in Video”, in Proceedings of the Fifth IEEE Workshop on Applications of Computer Vision, December 2000, pp. 224-229.

[2] Simon Y. Foo, “A Rule-Based Machine Vision System for Fire Detection in Aircraft Dry Bays and Engine Compartments”, Knowledge-Based Systems, volume 9, pp. 531-541.

[3] B.Ugur.Toreyin, Y. Dedeoglu, U.Gudukbay, and A.Enis Cetin, “Computer Vision Based Method for Real-Time Fire and Flame Detection”, Pattern Recognition Letters, 2006, pp. 49-58.

[4] B. Lucas, and T. Kanade, “An iterative image registration technique with an application to stereo”, Proc. 7th IJCAI 1981, August 24-28, Vancouver, British Columbia, pp. 674-679

[5] G. Healey, D. Slater, T. Lin, B. Drda, and A. Goedeke, “A system for real-time fire detection,” in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Jun. 1993, pp. 605 –606.

[6] T.-H. Chen, C.-L. Kao, and S.-M. Chang, “An intelligent real time fire-detection method based on video processing,” in IEEE 37th Annual 2003 International Carnahan Conference on Security Technology, Oct. 2003, pp. 104 – 111.

[7] W.-B. Horng, J.-W. Peng, and C.-Y. Chen, “A new image-based real-time flame detection method using color analysis,” in IEEE Networking, Sensing and Control, Mar. 2005, pp. 100 – 105.

[8] S.-J. Wang, M.-T. Tsai, Y.-K. Ho, and C.-C. Chiang, “Video-based early flame detection for vessels by using the fuzzy color clustering algorithm,” in Proceedings of the International Computer Sympoium, vol. 3, 2006, pp. 1179–1184.

[9] Liang-Hua Chen, Wei-cheng Haung, “Fire Detection Using Spatial-Temporal Analysis,” in World Congress on Engineering, volume III, 2013, pp. 2078-0958.

