

# “DETECTION OF FOREST FIRE USING MATLAB”

## A MINI PROJECTREPORT

***Submitted by:***

**KIRAN JANGID (1NH18EC726)**

**MERVIN FRANCIS (1NH18EC738)**

*In partial fulfilment for the award of the degree of*

### BACHELOR OF ENGINEERING

**IN**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**ACKNOWLEDGEMENT**

The satisfaction that accompanies the successful completion of any task would be, but impossible without the mention of the people who made it possible, whose constant guidance and encouragement helped us succeed.

We thank Dr. Mohan Manghnani, Chairman of New Horizon Educational Institution, for providing necessary infrastructure and creating good environment.

We also record here the constant encouragement and facilities extended to us by Dr. Manjunath, Principal, NHCE and Dr. Sanjeev Sharma, head of the department of Electronics and Communication Engineering.

We extend sincere gratitude to them. We sincerely acknowledge the encouragement, timely help and guidance to us by our beloved guide Monika Gupta to complete the project within stipulated time successfully.

Finally, a note of thanks to the teaching and non-teaching staff of electronics and communication department for their co-operation extended to us, who helped us directly or indirectly in this successful completion of mini project.

**R. MERVIN FRANCIS 1NH18EC738**

**KIRAN JANGID 1NH18EC726**

**ABSTRACT**

The aim of this project is to develop a smoke detecting algorithm using digital image processing techniques on multi-spectral (visible & infrared) video. By using method of principal component analysis, spatial filtering of principal is followed by PCA component images ,the location of smoke can be detected over a period of exposure time with a given frame capture rate. The result can be further analysed with consideration of wind factor and fire detection range to determine if a fire is present within a scene.

Infrared spectral data will be shown to contribute little information concerning the smoke signature. Moreover, the final processing techniques are focused on the blue spectral band as it is furthest away from the infrared spectral bands and because it experimentally yields the largest footprint in the processed principal component images in comparison to other spectral bands. A frame rate of .5 images/sec (1 image every 2 seconds) is determined to be the maximum such that temporal variance of smoke can be captured. This study shows eigenvectors corresponding to the principal components that best represent smoke and are valuable indications of smoke temporal signature.

The Raw video of the data has been taken through rigorous pre-processing schemes to align the frames from their respective spectral band both spatially and temporally. MATLAB which is a multi paradigm numerical computing program which is used to match the field of view across five spectral bands: Red, Green, Blue, Long-Wave Infrared, and Mid-Wave Infrared. The Extracted frames will be aligned for a short period of time from key frames throughout the data capture. This alignment will be allowed for more accurate digital processing of smoke signature.

**TABLE OF CONTENTS**

**CHAPTER 1**

INTRODUCTION................................................................................................................4

**CHAPTER 2**

LITERATURE SURVEY……………………………………………………………………….…….....6

**CHAPTER 3**

MATLAB……………………………………………..………….9

**CHAPTER 4**

PROJECT DESCRIPTION …………………………………………………..……………16

PROPOSED METHODOLOGY………………………………………………….................23

**CHAPTER 5**

RESULT……………………….……………………………………………………………32

**CHAPTER 6**

CONCLUSIONANDFUTURESCOPE….............................................……………………34

REFERENCE……..…………………………………………………………………….….35

APPENDIX…………………………………………………………………………….…..36

**Keywords**

List of Keywords

* Image processing
* Feature space
* Principal component analysis
* Temporal variance
* Median filtering
* Multi spectral video
* MATLAB

**Chapter 1**

**INTRODUCTION**

The Detection of forest fire programs are having lots of demands of the forest fire and other forest accidents for both economic property and for the safety of public. Every year many forests are facing natural disasters such as forest fire which are also affecting wildlife. Forest fire can lead to the destruction of vast area of land, a total of 3.6 to 4.5 million km of land will be destroyed. The rapid increase in the detection of forest fire throughout the world has resulted in a greater motivation for the development of the forest fire detection system for the earliest detection of wild fires. The most wildly used technology for forest fire detection is the technology of sensor physical parameters such as change in pressure, temperature and humidity. There are various other parameters consisting of chemicals such as carbon dioxide, carbon monoxide and nitrogen dioxide. However, it has been hard to apply all these systems in large areas due to various reasons such as high cost, energy which will be used by the sensors, the proximity of the sensor to the fire for accurate sensing will result in physical damage to the sensors.

Further adding, the sensor techniques have a high falls alarm rates and their response time is very big. There are various motivating factors for the use of the image processing technique to detect forest fire. The first and foremost factor is the rapid development of digital camera method and CCD which gives a high performaces in image quality and has decreased the cost of cameras. The second factor being that the digital cameras can cover large areas with excellent results. The third factor is the response time of image processing model is better than that of existing sensor models. Finally, the overall cost of image processing technique is lower than the existing system.

There is various fire detection algorithm which have been proposed by various researchers. In this forest fire algorithm, it uses YCbCr colour space since it effectively separates luminance from chrominance and is able to separate high temperature fire pixels because the fire at high temperature region is white.

These codes mainly consists on smoke and fire colour properties to find out the fire. The final performance will show that the projected methodology is better towards the detection of forest fire system and has very less false alarm which being the main problems of the exisiting system.

**Chapter 2**

LITERATURE SURVEY

1. Stipanicev, D.; Bugaric, M.; Bodrozic, L.; , "Integration of forest fire video monitoring system and Geographic Information System," ELMAR, 2009. ELMAR '09. International Symposium , vol., no., pp.49-52, 28-30 Sept. 2009
2. Changwoo Ha; Gwanggil Jeon; Jechang Jeong; , "Vision-Based Smoke Detection Algorithm for Early Fire Recognition in Digital Video Recording System," Signal-Image Technology and Internet-Based Systems (SITIS), 2011 Seventh International Conference on , vol., no., pp.209-212, Nov. 28 2011-Dec. 1 2011
3. Mustafa, M.; Taib, M.N.; Murat, Z.H.; Lias, S.; , "GLCM texture feature reduction for EEG spectrogram image using PCA," Research and Development (SCOReD), 2010 IEEE Student Conference on , vol., no., pp.426-429, 13-14 Dec. 2010
4. Jing Huang; Jianhui Zhao; Weiwei Gao; Chengjiang Long; Lu Xiong; Zhiyong Yuan; Shizhong Han; , "Local binary pattern based texture analysis for visual fire recognition," Image and Signal Processing (CISP), 2010 3rd International Congress on , vol.4, no., pp.1887-1891, 16-18 Oct. 2010
5. Vasyukov, V.; Kalennikova, E.; , "An adaptive procedure of smoke and background discrimination in the early fire detection video system," Strategic Technology (IFOST), 2011 6th International Forum on , vol.2, no., pp.844-847, 22-24 Aug. 2011
6. Schowengerdt, Robert A. (2007). Remote sensing: models and methods for image processing (3rd ed.). Academic Press. p. 2. ISBN 978-0-12-369407-2
7. Angayarkkani, K.; Radhakrishnan, N.; "Efficient Forst Fire Detection System: A Spatial Data Mining and Image Processing Based Approach," Computer Science and Network Security, 2009., IJCSNS International Journal of, vol. 9 , no. 9, pp. 100-106, Mar 2009
8. Calle, A.; Casanova, J.L.; Moclan, C.; Romo, A.J.; Cisbani, E.; Costantini, M.; Zavagli, M.; Greco, B.; , "Latest algorithms and scientific developments for forest fire detection and monitoring using MSG/SEVIRI and MODIS sensors," Recent Advances in Space Technologies, 2005. RAST 2005. Proceedings of 2nd International Conference on , vol., no., pp. 118- 123, 9-11 June 2005
9. Mazzeo, G.; Marchese, F.; Filizzola, C.; Pergola, N.; Tramutoli, V.; , "A Multitemporal Robust Satellite Technique (RST) for Forest Fire Detection," Analysis of Multi-temporal Remote Sensing Images, 2007. MultiTemp 2007. International Workshop on the , vol., no., pp.1-6, 18-20 July 2007
10. Krstinić, D.;, "Automatic Forest Fire detection in Visible Spectra, " Information and Intelligent Systems, 2010., Proceedings of the 21st Central European Conference on, vol., no.,pp.369-375, 22-24 Sept 2010
11. Cappellini, V.; Mattii, L.; Mecocci, A.; , "An intelligent system for automatic fire detection in forests," Image Processing and its Applications, 1989., Third International Conference on , vol., no., pp.563-570, 18-20 Jul 1989
12. Jones, T.A.; Christopher, S.A.; , "Multispectral Analysis of Aerosols Over Oceans Using Principal Components," Geoscience and Remote Sensing, IEEE Transactions on , vol.46, no.9, pp.2659-2665, Sept. 2008
13. Viktor Zubko; Yoram J. Kaufman; Richard I. Burg; J. Vanderlei Martins; , "Principal Component Analysis of Remote Sensing of Aerosols Over Oceans," Geoscience and Remote Sensing, IEEE Transactions on , vol.45, no.3, pp.730- 745, March 2007
14. Davenport Tim, “Early Forest Fire Detection Using Texture Analysis of Principal Components from Multispectral Video,” California Polytechnic State University, San Luis Obispo, CA, MS Thesis 2012.
15. Davenport Tim, “EFFD Summary,” unpublished.
16. Liu, H., Motoda H. (1998) Feature Selection for Knowledge Discovery and Data Mining., Kluwer Academic Publishers. Norwell, MA, USA. 1998.
17. Pearson, K. (1901). "On Lines and Planes of Closest Fit to Systems of Points in Space" (PDF). Philosophical Magazine 2 (11): 559–572.
18. Charles A. Poynton (2003). Digital Video and HDTV: Algorithms and Interfaces. Morgan Kaufmann. pp. 260, 630. ISBN 1-55860-792-7.
19. P. E. Trahanias and A. N. Venetsanopoulos, “Color image enhancement through 3-D histogram equalization,” in Proc. 15th IAPR Int. Conf. Pattern Recognition, vol. 1, pp. 545–548, Aug.-Sep. 1992.
20. M. Strianese (2015). Firefighting Applications [Online]. Available: http://pr- infrared.com/about-thermal-imaging/firefighting-applications/

**Chapter 3**

**MATLAB:**

The function [y1,...,yN] = myfun(x1,...,xM) declares a function named myfun that accepts inputs x1,...,xM and returns outputs y1,...,yN. This declaration statement must be the first executable line of the function. Valid function names begin with an alphabetic character, and can contain letters, numbers, or underscores.

You can save your function:

* In a function file which contains only function definitions. The name of the file must match the name of the first function in the file.
* In a script file which contains commands and function definitions. Functions must be at the end of the file. Script files cannot have the same name as a function in the file. Functions are supported in scripts in R2016b or later.

Files can include multiple local functions or nested functions. For readability, use the [end](https://www.mathworks.com/help/matlab/ref/end.html) keyword to indicate the end of each function in a file. The end keyword is required when;

* Any function in the file contains a nested function.
* The function is a local function within a function file, and any local function in the file uses the end keyword.
* The function is a local function within a script file.

MATLAB which is a high-performance programming language is used for basically technical computing. It integrates computation, visualization, and programming in an simpler way to use and is environment friendly where the problems and solutions can be expressed in a familiar mathematical notation. The typical uses of these include:

* Maths and computation
* The development of algorithm
* Modeling, simulation, and prototyping
* Data analysis, exploration, and visualization
* Scientific and engineering graphics
* Application development, including Graphical User Interface building

MATLAB :  is a programming platform designed specifically for engineers and scientists. The heart of MATLAB is the MATLAB language, a matrix-based language allowing the most natural expression of computational mathematics.

MATLAB is a highly interactive system where the basic data of elements in an array that has no need of dimensioning. This method will reduce the technical problems, mainly those with matrix and vector formulations, in a very soon amount of time it would take to develop a program in a scalar noninteractive language such as C.

The name MATLAB stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects, which together represent the state-of-the-art in software for matrix computation.

**The MATLAB System:**

The MATLAB system consists of five main parts:

**The MATLAB language.**

This is a high-level matrix language which will control the flow of statements such as functions, data structures, input/output, and object-oriented programming features. It allows the access for both "programming in the small" to quickly create the dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.

**The MATLAB working environment. :**

This is the set of tools and facilities that you work with as the MATLAB user or programmer. It includes facilities for managing the variables in your workspace and importing and exporting data. It also includes tools for developing, managing, debugging, and profiling M-files, MATLAB's applications.

**Handle Graphics.**

This is the MATLAB graphics system. It includes high-level commands for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level commands that allow you to fully customize the appearance of graphics as well as to build complete Graphical User Interfaces on your MATLAB applications.

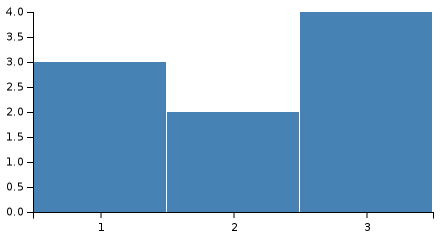
**The MATLAB mathematical function library.**

This is a vast collection of computational algorithms ranging from elementary functions like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

**The MATLAB Application Program Interface (API).**

This is a library that allows you to write C and Fortran programs that interact with MATLAB. It include facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

## Graphics and graphical user interface programming[[edit](https://en.wikipedia.org/w/index.php?title=MATLAB&action=edit&section=9)]

* 

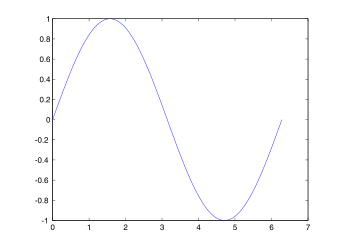
MATLAB has tightly integrated graph-plotting features. For example, the function *plot* can be used to produce a graph from two vectors *x* and *y*. The code:

x = 0:pi/100:2\*pi;

y = sin(x);

plot(x,y)

produces the following figure of the [sine function](https://en.wikipedia.org/wiki/Sine_wave):

[](https://en.wikipedia.org/wiki/File:Matlab_plot_sin.svg)

MATLAB supports three-dimensional graphics as well:

MATLAB will be able to call the functions and subroutines written in the programming languages C. A wrapper function will be created for allowing MATLAB data types which has to be passed and returned. The MATLAB executables are the dynamically loadable objected file which will be created by compiling. From the year 2014 increasing two-way interfacing along with python has been added.

The Libraries written in java Perl ActiveX can be directly called from MATLAB and a lot of MATLAB libraries like XML / SQL are implemented as wrappers around Java or ActiveX libraries. Calling MATLAB from Java is more complicated, but can be done with a MATLAB tool box which is sold separately by math works or using an undocumented mechanism called JMI Java-to-MATLAB Interface which should not be confused with the java Metadata interface that is also called JMI. The official script of MATLAB API for java programming was added in 2016.

As alternatives to the MuPad based Symbolic Math Toolbox available from MathWorks, MATLAB can be connected to maple or Mathematics.

Libraries also exist to import and export MathML.

**Functions :**

A function is a group of statements that together perform a task. In MATLAB, functions are defined in separate files. The file name and the function should be of the same type.

Functions can be operated on numerous variables within their own workspaces, which is also called the **local workspace**, separate from the workspace you access at the MATLAB command prompt which is called the **base workspace**.

Functions can accept more than one input arguments and may return more than one output arguments.

## Anonymous Functions :

An anonymous function is like an inline function in traditional programming languages, defined within a single MATLAB statement. It consists of a single MATLAB expression and any number of input and output arguments.

You can define an anonymous function right at the MATLAB command line or within a function or script.

This way you can create simple functions without having to create a file for them.

## Primary and Sub-Functions :

Any function other than an anonymous function must be defined within a file. Each function file contains a required primary function that appears first and any number of optional sub-functions that comes after the primary function and used by it.

Primary functions can be called from outside of the file that defines them, either from command line or from other functions, but sub-functions cannot be called from command line or other functions, outside the function file.

Sub-functions are visible only to the primary function and other sub-functions within the function file that defines them.

## Nested Functions :

You can define functions within the body of another function. These are called nested functions. A nested function contains any or all of the components of any other function.

Nested functions are defined within the scope of another function and they share access to the containing function's workspace.

## Private Functions :

A private function is a primary function that is visible only to a limited group of other functions. If you do not want to expose the implementation of a function(s), you can create them as private functions.

Private functions reside in **subfolders** with the special name **private**.

They are visible only to functions in the parent folder.

## Global Variables :

Global variables can be shared by more than one function. For this, you need to declare the variable as global in all the functions.

If you want to access that variable from the base workspace, then declare the variable at the command line.

The global declaration must occur before the variable is actually used in a function. It is a good practice to use capital letters for the names of global variables to distinguish them from other variables.

**Chapter 4**

**PROJECT DESCRIPTION**

**Proposed early forest fire detection**

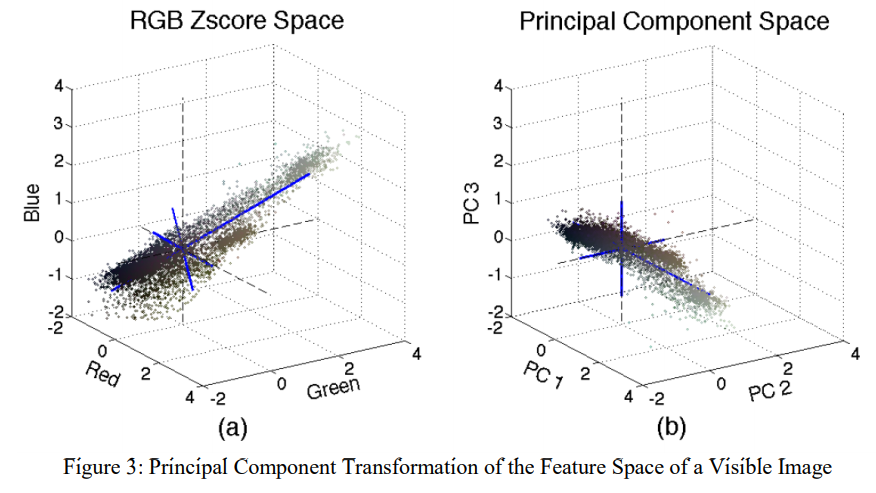
A land based multi spectral and multi temporal video has been proposed to determine whether smoke and fire is present in the scene. This method will use temporal, spatial and spectral information from multiple camera views to identify the smoke. The special feature of the smoke is compared to the special feature of the other objects in the same to identify the smoke’s spectral signature. Using various images through time, the temporal variance of the smoke will be identified with the simple difference in the image and principal component transportation, this is a complicated method which provides better results.

**Feature space**

A feature space is a numerical dimensional vector which will numerically represent some objects. In this project a feature vector can be constructed by using the pixel location of a particular image or a set of images. The feature space is associated with the vector space the encompasses image vectors. For example, a set of three images can be equally displayed in three-dimensional feature space. The pixel location of the image space has three coordinate values picked up from the red, green and blue images. Throughout this process the fire seen will be an analysed in both feature space and image space in order to identify the smoke.

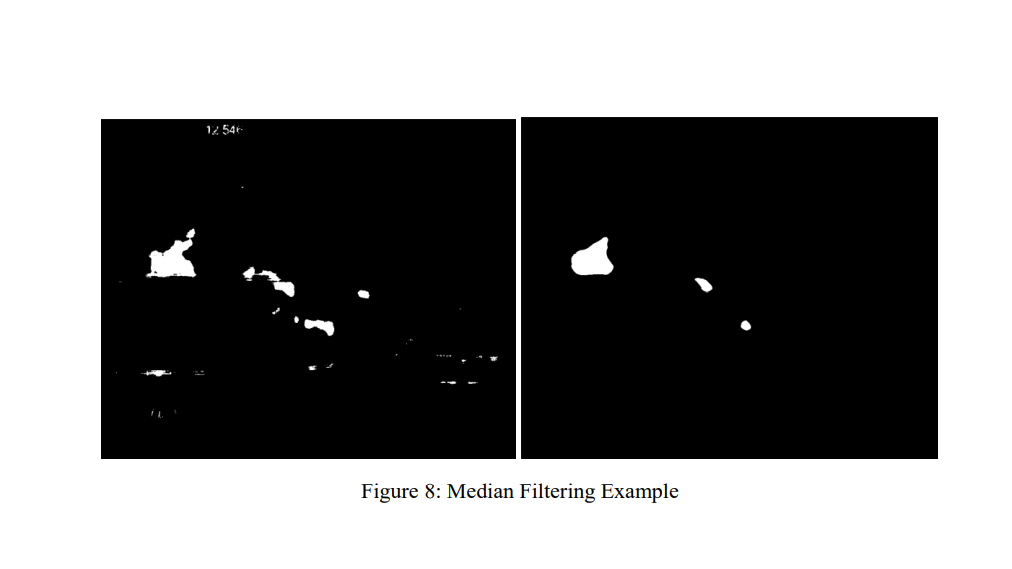
**Principal component transform**

The principal component analysis is a statistical process which results in decomposition of sample data. Since decomposition is linear transformation PCA will maintain a single integrity while revelling the internal structure of the data in a way that will explain the variance in data in the best way. Each PCA will account for more variabilities in the data as possible under the constrained that orthogonal and uncorrelated with preceding components. The number of the principal components will always be equal to or less than the number of dimensions that the data represents.



**Median filtering**

In image processing technique we can be able to reduce the amount of noise in the image. Median filtering process is a non-linear transformation technique that works well in eliminating salt and pepper while preserving edges. This filter will run through the image entry by entry and will replace annual replace each entry with median value of neighbouring entries. The median of list of numbers can be found by picking the middle index. Window, which is known as the size of the neighbourhood pixels, the size of this parameter can be chosen according to the size of the noise. MATLAB will provide two-dimensional median filtering function which by default will give an output image where each pixel will contain the median value in a three-by-three neighbourhood around the other pixels in the input image. Below we will be able to observe the effects od median filtering on noise removal in the smoke detection algorithm.



**Pre-processing**

In order to get a successful smoke detection algorithm a considerable amount of pre-processing will be done to extract the images from their respective video files and align them spatially and temporally. Due to the data corruption the pre-processing techniques will be revised and documented to help future fire fighters where relevant processing can take place.

**Working data**

Now, the end result of pre-processing data will give access to 100 consecutive instances in each time with five-dimensions corresponding to five channels. The visible camera will contribute colours (red, blue, green) and the IR camera will contribute the long wave and mid wave infrared channel. All the image frames will be having the same resolution and field of view which will allow for consistent and robust digital image processing techniques to identify the smoke in the fire seen successfully.

**CODE USING MATLAB:**

img=imread('cadru230.jpg');

img=imgaussfilt(img,[3 3]);

tau=40;

subplot(2,2,1);

imshow(img);

red=img(:,:,1);

green=img(:,:,2);

blue=img(:,:,3);

%img=im2double(img);

hsv=rgb2hsv(img);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

ycbcrmap=rgb2ycbcr(img);

Y=ycbcrmap(:,:,1);

% Y=round(Y,2);

Cb=ycbcrmap(:,:,2);

% Cb=round(Cb,2);

Cr=ycbcrmap(:,:,3);

% Cr=round(Cr,2);

%%%%%%%%%%%%%calculare Ymean

[a,b]=size(Y);

SumaLinii=sum(Y);

total=sum(SumaLinii);

Ymean=total/(a\*b);

[a,b]=size(Cb);

SumaLinii=sum(Cb);

total=sum(SumaLinii);

CbMean=total/(a\*b);

% CbMean=round(CbMean,2);

%%%%%%%%%%%%%calculare CrMean

[a,b]=size(Cr);

SumaLinii=sum(Cr);

total=sum(SumaLinii);

CrMean=total/(a\*b);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

h=hsv(:,:,1);

h=round(h,2);

s=hsv(:,:,2);

s=round(s,2);

v=hsv(:,:,3);

v=round(v,2);

[lin,col]=size(s);

test=ones(lin,col)\*255;

for i=1:lin

for j=1:col

ss=1-3.0\*min(red(i,j),min(green(i,j),red(i,j)))/(red(i,j) +green(i,j)+blue(i,j));

if(~(abs(Cb(i,j)-Cr(i,j)<tau)&&red(i,j)>180 && red(i,j)>=green(i,j) && green(i,j)>blue(i,j) && ss>=((255-red(i,j))\*0.2/180) &&(v(i,j)>0.8 && v(i,j)<1)&& (Y(i,j)>=Ymean) && (Cb(i,j)<=CbMean) && (Cr(i,j)>=CrMean)))

%if(~(red(i,j)>=green(i,j) && green(i,j)>blue(i,j) && (red(i,j)>115 &&red(i,j)<135) && s(i,j)>(255-(red(i,j)\*60)/120)))

test(i,j)=0;

%contor=contor+1;

end

end

end

BW2 = bwareaopen(test,100);

test=imclearborder(test);

stats = struct2table(regionprops(test,{'Area','Solidity','PixelIdxList'}));

idx = stats.Solidity < 0.9 | stats.Area <350;

for kk = find(idx)'

test(stats.PixelIdxList{kk}) = true;

end

subplot(2,2,2);

imshow(test);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%aplic denoise

r=(7-1)/2;

densitate=zeros(lin,col);

densitate=calcDensitate(test,densitate,7);

for i=r:lin-r

for j=r:col-r

count=densitate(i,j);

if(count<5)

test(i,j)=0;

end

end

end

subplot(2,2,3);

imshow(test);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%fill

test=medfilt2(test);

sv=fspecial('sobel');

sx=sv';

Gy=imfilter(test,sv,'replicate');

Gx=imfilter(test,sx,'replicate');

TL=0.39; %prag mic si prag mare

TH=0.98;

% kk=edge(test,'sobel',0.39);

% ll=edge(test,'sobel',0.98);

kk=edge(test,'sobel',0.39);

ll=edge(test,'sobel',0.98);

lll=edge(test,'canny',[TL TH]);

final=imadd(kk,ll);

edgePixels=nnz(final);

area=bwarea(final);

subplot(2,2,4);

imshow(final);

|  |
| --- |
|  |

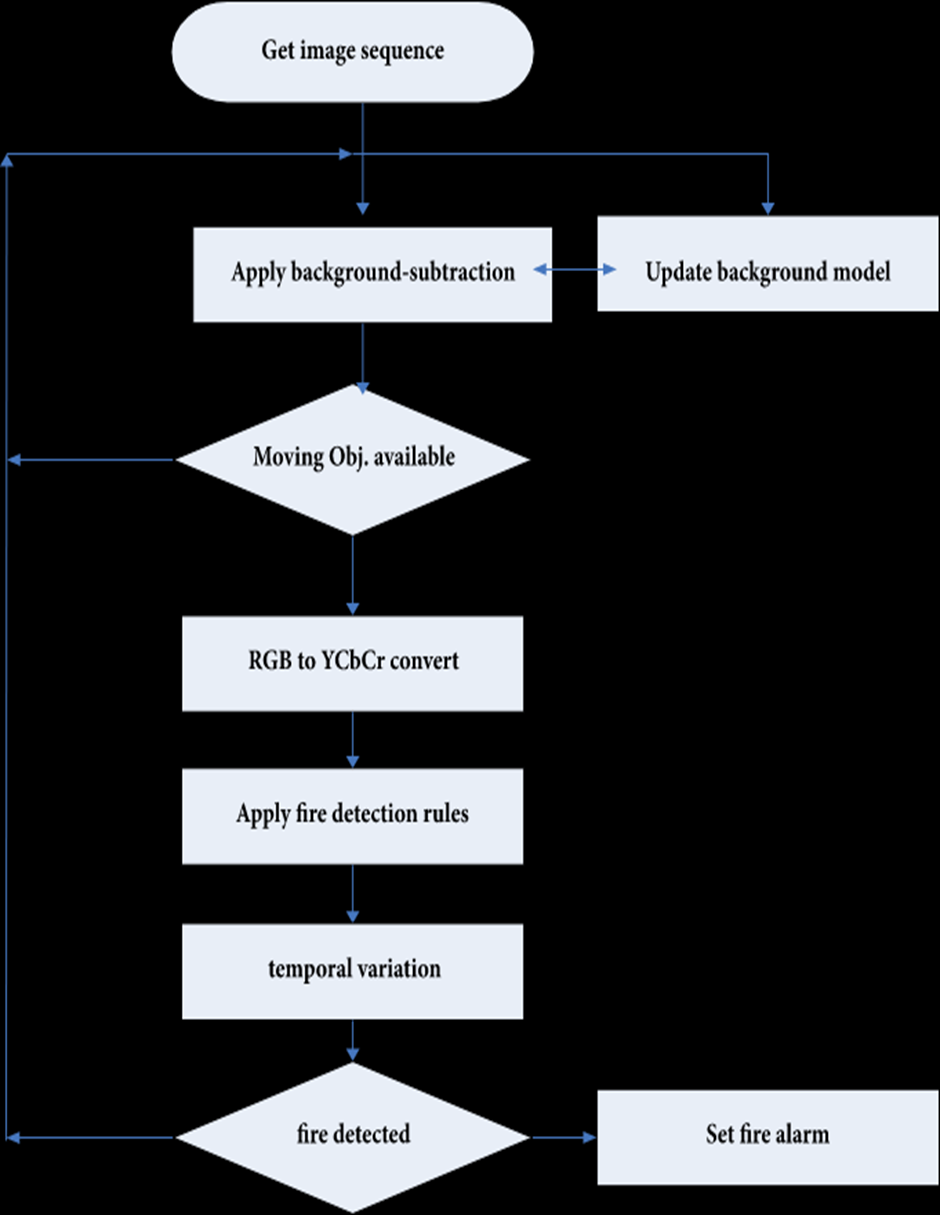
**PROPOSED METHODOLOGY**

The below mentioned section represents the proposed methodology of detection of forest fire. It consists of various stages: Getting the image sequence, applying the background subtraction, updating the background model, moving the objects that are there, convertion of RGB to YCbCr, applying forest fire detection rules, temporal variations, fire detection, setting fire alarms etc.

The foremost step is to receive the video or data from input device after that the next step involves Applying movements containing the region in which the detection based on background subtraction MRDB after that the following step is to convert the input image sequence from RGB to YCbCr colour space after that the final step is to apply the fire detection rules as well as temporal variation .

The fire alarms will be activated only when all the fire detection conditions are verified. The explanation of proposed algorithm stages will be discussed in detail.

The below diagrams will explain the exact methods of fire detection system.



Movement of Region Detection will be Based on the Background Subtraction MRDB.

The Detection of moving regions in a forest areas is a key factor in most of the video or picture based detection of fire systems because fire are boundaries continuously fluctuate. That’s why back ground subtraction is used for only to select the candidate regions of fire in the spot . A pixel located at (x, y) is supposed to be moving if the following condition is satisfied. Ex :

| In ( x , y) – Bn (x ,y) | > thr

(x, y) is represents the intensity value of the pixel at the exact location (x, y) in the nth gray-level for the current frame and Bn (x, y) represent the background intensity value at the same pixel location, and there is a threshold value experimentally set to 3. The background is continuously updated . ex:

Bn ( x , y) + 1 if In ( x ,y) > Bn (x , y)

B n+1(i,j) = Bn ( x , y) - 1 if In ( x ,y) < Bn (x , y)

Bn ( x , y) if In ( x ,y) = Bn (x , y)

:where (x, y) and (x, y) represent the intensity of the pixel value at location (x, y) for the current and previous backgrounds.

The below figure shows an example of MRDB .



This is the process of converting RGB to YCBCr images.

Hence in this situation the various kinds of moving objects which can be covered further applying the subtraction, like trees, animals birds and people. Hence images from the background subtraction stage are converted to YCbCr to select candidate fire regions using original RGB image and YCbCr component. The mean values of the YCbCr channel are then calculated using below methods :

Ex:

Y 0.2568 0.5098 0.0979 R 16

Cb = -0.1482 -0.2910 0.4392 \* G + 128

Cr 0.4392 -0.3678 -0.0714 B 128

After that we have to find Y mean , Cb mean , Cr mean by using the above calculations .

where , , and are the mean values for the YCbCr channels; Y (x, y), Cb (x, y), and Cr (x, y) are YCbCr channel values for pixel at specific location (x, y); and NM is the total number of pixels.

Fire Color Pixel Detection Rules are below :

In any fire image pixels, the red colour value is always larger than green and green is always larger than blue as illustrated is a fire image and is RGB channels histogram for the same image. This fact is represented in RGB colour space as R>G>B and can be converted to YCbCr using the flowing equations:

Y(x , y) > Cb ( x , y)

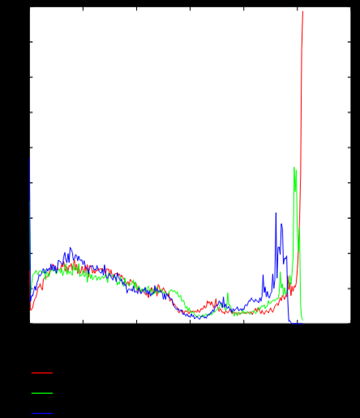
Cr (x ,y) >Cb (x ,y)

Also, the Y component value will be more than the mean Y component of that particular image and the Cb component will be small than the mean Cb of that particular image, while the Cr component is much more greater when compared to the mean Cr component. These can be represented in the following flow:

F(x, y) = 1 if Y(x ,y) >Ym; Cb (x ,y)Cr m

1. orelse

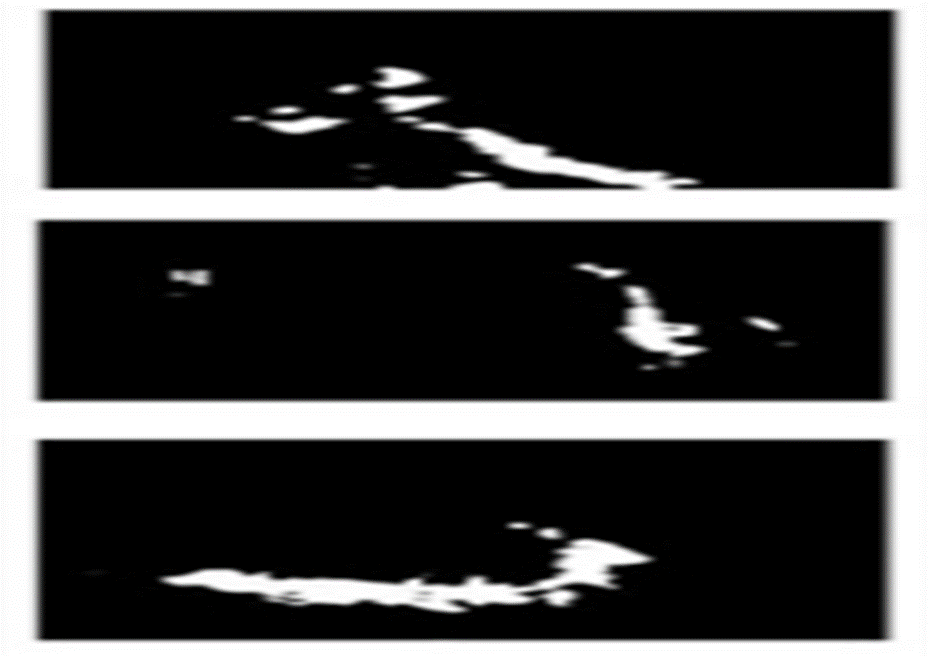
where F(x, y) can be any pixel on the image Ym , Crm and Cbm are the mean values for Y, Cb, and Cr, respectively.



a) image of the RGB

(b) same image channels of the histogram

The Cb component as shown will always predominantly “black” and the Cr component isalways “white”. where τ is the constant, specified by using receiver operating characteristic (ROC), by using various values of τ in the range . To measure the “true detection rate” and false detection, of the calculations data may consists of 500 imagesof forests and (example,300 of them being images of forest fire, 200 non fired images) collected them from Internet were used. Only 10 had been used with various values of τ in the range to find the binary images of the forest fire area . τ was chosen as τ = 70 which will give us an accurate detection of greater than 90 percent , the false detection will result is less than 40 perecent.



( methodology picture )

Rules for applying the input images like (a) an original RGB image, (b) by using rules of binary images for binary images .

Temporal Variation

Using only the colour models only is not enough to identify fire appeared in the forest accurately because there will be a lot of objects that are shares of the same fire colour such as red leaves, desert, and other red moving objects. The important difference between actual fire and the fire-colour objects is the motion of their nature in itself . The Shapes and size of the fire flame it may change, because of the burning materials and airflow thus, it produces a very high temporal variation. In contrast the rigid bodies’ movements will have lower temperature modulations. So that, it is possible to differentiate between the fire pixels and the fire colours and To detect a fire variations, the main variations between the successive frames was analysed. So for an example lets take a sequence of videos consisting of number of frames, and the moderate temporal variation is defined so where (x, y) is the average temporal variation, (x,y) is the intensity of a pixel at that particular spot x, y in the frame. If (x, y) are greater than thr (experimentally determined threshold), then the moving pixel is fire.

**Advantages And Disadvantages**

**Advantages**

* Reliability
* It is very compact
* It can easily detect the distance
* Response time is very fast
* Specially designed to detect the fire
* Detection is earlier compared to other methods of detection

**Disadvantages**

* High cost
* Lot of energy will be used
* Debugging is not easier

**Chapter 5**

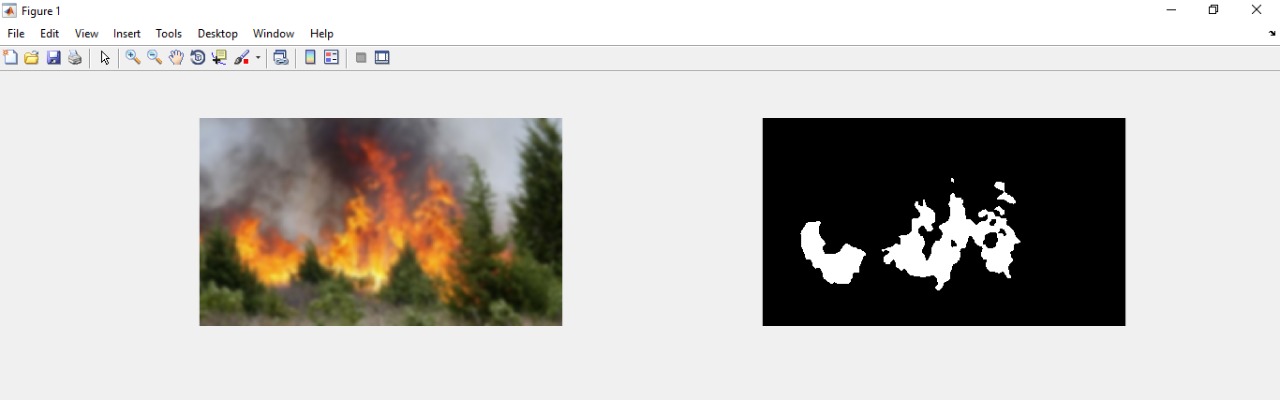
**RESULT**

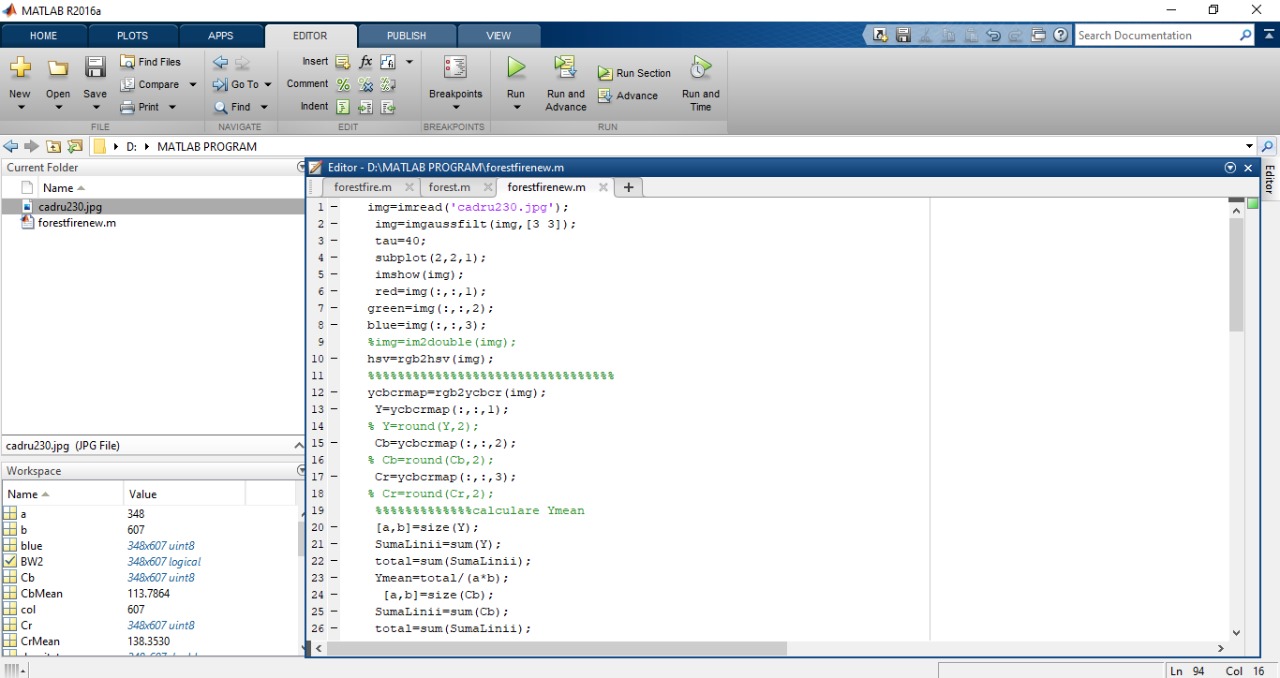
We have experimented the detection of forest fire practically with the help of the programming tool MATLAB. The experiment was performed the efficiency of the proposed system to detect the forest fire.

For comparison of images we have taken two images from the internet one which consist of fire and the other which doesn’t consist of fire. The image which consist of fire , shows a diversity in colour and environment. By doing this experiment we able to learn that this proposed method is one of the best methods to detect forest fire at a earlier stage.

This method also has a low effect of giving a false alarm and is very cost efficient. It is also easier than compared to using sensors which would be very difficult to place on a wide region of area and the cost price also would be significantly high.

The project is executed using the MATLAB code. Image processing technique has been used. Hence the project is carried out successfully and has been experimented carefully.





**Chapter 6**

**FUTURE SCOPE AND CONCLUSION**

Here in this project, a new image-based with real-time forest flame detection system has been projected but it is mainly based on different vision mathods. This projected method is mainly consisting of three various stages which are fire detection pixel using only colour, moving the pixel detection, and finally analysing the fire-coloured moving pixels in exact frames to raise an alarm. The main fire colours model will be got around 99.88% on the ten to nine tested video sequences with diverse imaging conditions. And more over the experiments on the benchmark fire of the databases of videos also show that the proposed method which achieves the comparable performance with respect to the state of art and fire detection methods. The total proposed system of these proposed forest fire detection system can also be improved by considering smoke at early stages of fire raising and more over detecting the smoke is a challenging task and different lighting conditions raised by nature and also other external optical effects of the nature. Such as high false detections also can be resolved by analysing every smoke regions of the forest. This yields a high computational load.

The motion information of fire appeared in the forest is also considered to characterize fire regions. The proposed system assumes that’s the fire definitely will grow gradually in a spatial domain. This might not be the case in some other situations. For instance, the system might not be able to detect a fire appeared in the forest caused by a sudden explosion. In order to activate such cases, the proposed system will be further improved to include different situations and moreover, shape information of fire regions of the forest will also be investigated to improve the system’s fire detection performance in future.

This project explains a useful method of detection of forest fire method using image processing technologies with the movements of the objects containing the region detection of the spot based on the used subtraction and colour segmentation. This process is mainly used in YCbCr colour space method which can be used for better separation of the luminance and also will have a greater detection rate , it would also be accurate and hence will provide us a proper detection rate. Detection of forest fire rules can be applied to find out the hearth.

The results and variations of the projected algorithm can be tested based on set of data consisting of 4-5 video collected from external source, 2-3 of which will be the actual forest fire detection video of the forest, tp-rate and tn-rate are well calculated. The performance will show that the projected algorithm which has achieved a good detection result. The overall performance indicates that the projected method is proper and well defined and is very well used for the purpose of the detection of forest fire alarms which will mainly be automatic.

Further the work for the system can also improve by using a mix of various rule of the different colour spaces; hence, the main challenge is choosing the exact rules from the various colour spaces which will be used by different methods.

For more accurate readings use of Neural Networks for decision making also can be made and GSM module and can also be implemented for sending SMS (text message) to nearby fire station in case of severe or high fire appears in forest. Water and Sprinklers can also be incorporated. By research and study, the use of these proposed system of detection of forest fire can be rapidly increased. The margin of the false alarms is often reduced even further by developing algorithms to eliminate the detection of red coloured cloth as fire. By exact analysis, suitable location height and length for camera instalment can be decided, in order to remove blind-spot areas of the forests

**REFERANCES**

* [www.nevonprojects.com](http://www.nevonprojects.com)
* [www.in.mathworks.com](http://www.in.mathworks.com)
* [www.ijitee.org](http://www.ijitee.org)
* [www.codeforge.com](http://www.codeforge.com)
* [www.ijarce.com](http://www.ijarce.com)
* [www.projectsinbangalore.com](http://www.projectsinbangalore.com)
* [www.ijecs.in](http://www.ijecs.in)
* [www.alldatasheets.com](http://www.alldatasheets.com/)
* [www.digikey.com](http://www.digikey.com)
* [www.wikipedia.com](http://www.wikipedia.com)
* [www.electronicslab4u.com](http://www.electronicslab4u.com)

**Appendix:**

* Forest fire detection
* Image processing
* Feature space
* Median filetring
* MATLAB