

# FIRE DETECTION SYSTEM IN PYTHON USING OPENCV

M.Mohamed Ismail<sup>1</sup>, B Chouthri<sup>2</sup>, M Chandru<sup>3</sup>, Mr.V.Maheskumar, M.E.<sup>4</sup>

<sup>1,2,3</sup>UG Scholar Department Of Computer Science and Engineering,  
Paavai Engineering College, Namakkal, Tamilnadu

<sup>4</sup>Assistant Professor Department Of Computer Science and Engineering, Paavai Engineering  
College, Namakkal, Tamilnadu

**ABSTRACT:** Our project aimed to detect fire by using the image processing technology that will alert people by early detection of fire. As there are many automatic fire alarm systems already existed like the sensor method, that has some limitations and designed to sense fire with the smoke, limited areas. To reduce limitations and to optimize with new technology, the project is proposed. The project is implemented by using Raspberry Pi3 Model B as a central processing unit and to connect the webcam as hardware. Webcam is taken as an input source, which captures the video feed from the surrounding and feeds into the Raspberry Pi. The entire code is written in pure python language using the open CV library for image processing. The theoretical parts emphasize more in computer vision, machine learning, image processing, color model, and the working algorithm of the project to detect the fire. The project gives a better understanding of object detection with the computer and the use of these technologies in different forms and uses.

## 1.INTRODUCTION

Fire is very dangerous that brings great loss of life and properties. Yearly thousand of accidents related to fire happen all over the world due to power failure, accidental fire, natural lightning. So to control fire, various system is developed and being developed. The existing systems are the smoke sensors types and sprinkler type systems that detect fire from smoke and designed to activate after reaching the threshold set temperature. Still, with this kind of system, there are many disadvantages like a false alarm, space coverage, signal transmission, and also the delay in a fire alarm. As smoke detectors are placed in the ceiling, smokes take time to reach up to the ceiling, which results in time delay. And another problem of the existed system is tough to implement in the open environment and large infrastructures like stadiums, aircraft hangers due to the vast area covered by these infrastructures. Using the image processing technology in fire detection opens many possibilities. The technology can implement in hazardous areas where the heat and temperature are very high, and there is always a chance of getting fire. Those places should be monitored continuously because of the high-risk zone with this technology. The system can give the right information about the site. The project is designed with the Raspberry Pi as the central processing unit to obtain the cost-effectiveness, low power consumption, and portability. Raspberry Pi also offers many other advantages like the coupling of additional hardware, gives many opportunities to use software and networking, which helps this project to use in the future with other projects as well. The importance of the proposed thesis is to make a reliable, safe, and smart system to reduce limitations and faults like false alarms, which cause panic among the people and even the loss of money with the use of new technology. And make the places safe from the hazardous fire.

## 2.COMPUTER VISION

This is a challenging task for the computer to identify and recognize the image from surrounding because humans have a better understanding of an object around their surroundings. From the beginning of human evolution, human has huge memory and data of an environment and surroundings from the thousands of years which computer don't have. But due to advancements in computer technologies, there are billions of stored data in a computer, which helps to make computers advanced on recognizing the images. Usually, computer vision refers to artificial intelligence as a computer must compare and interpret the surrounding what it sees around it and must perform the appropriate tasks of analyzing the object, process it, and compare from the memory of the computer and give the result. For an example of a jigsaw puzzle to make clear how computer vision works. At first, while solving the puzzle first, human has to make a mind or visualize how the final result is going to look like, so that's how the neural network in the computer takes action. Firstly, the computer analyzes different pieces of images and then identifies the edges and corner of the puzzle and tries to organize the pieces through the deep network layers as humans do. Even the computer doesn't have the final result of the



puzzle. It uses the information and the images, which is fed by the many available related images and information; thus computer recognizes the object and gives the result. Computer vision technology is developing more and more due to the rapid rise of development in machine learning and artificial intelligence. Due to the development of object detection technology, it is used in a wide range. From daily use of face detection to unlock the mobile device or to open the door to the research in the cell division, DNA structures also in the space exploration, for the weather forecasting everywhere there is the use of this technology.

## 1. Machine Learning

Machine learning is the study of the algorithms and the statistical models that computer system uses for the specific task without the use of instruction or relying on some patterns and 13 interference instead. A machine learning algorithm is created by collecting the data and then represent in the mathematical model, which helps to make the predictions and make the decision referencing from the mathematical model. Machine Learning is used in a wide variety of field like in object detection, sorting of packages, filtering of documents, predicting the patterns which help perform the task very fast and more precisely than human. Machine learning is the science of making computers learn and act like humans by feeding the data and information without being explicitly programmed. Generally, machine learning has a different way of learning.

### • Supervised learning

Supervised learning works under the supervision in which machine learns and trains data with the labeled dataset. Data that knows the target answer is called the label data, suppose if there is a dog that is labeled as a dog, then it is feed in the machine, which analyzes and learns the associations of the images based upon the features like color, shape, and sizes. So now, a machine can recognize the dog if the new unknown image without a label is fed in a machine, which is known as supervised learning. Supervised learning has a feedback mechanism. Mostly supervised algorithms use a decision tree, logistic regression, support vector machine.

### • Unsupervised learning

In unsupervised learning, the algorithm is trained with the data which is unlabeled; hence the machine tries to identify the patterns and give the response. Suppose if the data is given, which is not labeled, then the machine finds the structure and pattern in the data and groups them accordingly to the similarities, patterns. Unsupervised learning algorithms use k-means clustering, hierarchical clustering, apriori algorithm.

### • Reinforcement learning

Reinforcement learning is adapted from the supervised learning. But there is certain difference between these two method of machine learning. In supervised learning the output result is already known but in reinforcement learning the output is unknown but here to retain output the best action is measured. Reinforcement learning is totally based upon reward based learning system. In Reinforcement learning, an agent interacts with its environment by producing 14 actions and discovers errors or rewards. There is no predefined data the input depends upon the actions taken by the agent. The action is then saved in the form of some data that helps as a memory for the agent. As the agent explores the environment, it will collect data, which is then used to get the output. Reinforcement learning algorithm use Q-Learning, SARSA.

## 2. Object Detection

Object detection is the computer vision that deals with the identifying of the object and locating the object with certain classes in it with the help of the pictures, videos, or the camera feed. Object detection technology is used almost every where these days to make works easy, safe, and fast. The use of this technology is endless, like video surveillance, face detection, self driving vehicles, cell count, and the list goes on. Object detection before the advancement in deep learning was very lengthy and had several steps processes to execute. The first time face detection work in 2001 with the help of a violaones algorithm. They introduce to detect face by the help of webcam their algorithm only detect the face suppose if the face is in a different angle or downwards, then it would not detect. For this project, the primary process of detecting fire is inspired by the same algorithm proposed by the Viola-Jones face detecting algorithm



### 3.METHODOLOGYAND ALGORITHM

Here in fire detection, we are using the Haar Cascade classifier, which is very popular in object detection through the image or any other video feeds. The algorithm is also known as the ViolaJones algorithm because Paul Viola and Michael Jones developed it. The algorithm to detect fire is based on the machine learning concept using the Haar-like features, which is combined with the cascade classifier. To create a cascade classifier, many images so-called positive and negative images were uploaded for training into the computer. Positive images refer to the pictures in which fire is included, and negative images are just the background images that mean those images, which doesn't contain a fire in the images. For this training, more than 150 images were uploaded in which 75% of the images were positive, and 30% were negatives. In order to train the cascade classifier, almost double the amount of negative images must be uploaded because the cascade classifier is based upon appearance-based methodology. After uploading of images, there are two stages in the algorithm for detecting fire one is the detection and another is training algorithm which is explained below with the steps included in it: 5.1 Detection The procedure for detecting fire classifies images based on the value of simple features from an image. While detecting the fire, first of all, image is converted into grayscale since it is easy to work with, and it has less data rather than RGB color images. The algorithm outlines a box and searches for the fire in the image the box shows searching of fire mainly, the box is searching for Haar-like features that are explained in further paragraphs. Along with small steps, the box detects the features like edges, brightness level from the image of fire, and then data collected by boxes are put together, which helps to determine where the fire is located.

#### 1. Haar-like Features

Haar features are similar to convolutional kernels, which are used to detect the presence of those features in the given image. Some features some of the regions are black, some regions are white, and some region is slightly darker or vice-versa from each other which helps the machine to understand an image. Due to this difference in brightness of an image, the Haar features like edge features, line features, center-surround features rectangle is moved around the image. The rectangle box of the color white and black is decided according to the region. Which compares the brightness and extracts features by assigning some values as it goes and inspects through that region in an image. Applying Haar-features The white and black rectangular Haar-feature is explained by assuming the rectangle into a grid of size  $4 \times 4$  each square represents as a pixel. In every pixel is assigned with the intensity of white and black color in grayscale in which 0 represents a white pixel, and 1 represents a black pixel. But in a real scenario, the values of the pixel are different because the image doesn't have perfect white or perfect black intensities. After this, the algorithm will compare the real and ideal scenario by using the mathematical equation, as mentioned below, which will give a value of the particular feature. So here for the case, values 0.52 is the value of the real image. The closer the value to 1, the more likely to found a Haarfeature. Since this calculation is done only for a small amount of pixel, but in the real scenario, this type of intense calculation is being carried out several times, so to make it fast and reliable, another important part-integral image is carried out.

#### 2. Integral Image

The integral image plays an important role. It gives a boost to the calculation, which makes the approach as fast as possible because there are thousands of pixels that need to be calculated. Haar features the sum of pixel. This is to large number of features the algorithm requires iterating overall features, the features must be computed efficiently. So to solve this issue integral image is introduced. The sum of the pixel in rectangle D can be calculated with reference to four arrays. The value of the image at location 1 is the sum of pixels in a rectangle. Respectively value at position 2 is  $A+B$ , value at position 3 is  $A+C$ , and value at position 4 is  $A+B+C+D$ . So the sum within pixel D can be calculated as  $4+1-(2+3)$ . This calculation gives the actual value of the region, which is needed so that every time algorithm doesn't have carried out the repetitive calculation for the process. Integral image cuts out the time that will take to calculate the pixels. The integral image part is used to sum all the pixels of a particular box to its left and above ones. The four corner values of the area are to be calcued. This makes the summing of each pixel in the region.

#### 3. Training and Learning

After extracting features from an image to detection. The machine is now trained to identify the features from the input. So it is fed with lots of information so that it can predict objects from the information. For these, the image is shrunk into size  $24 \times 24$  from this; the algorithm looks for the features. To train the machine, It needs a lot of pictures of fire, different fireplaces, conditions of fire, also known as a positive image. Concerning that, many negative images were uploaded. The negative image refers to pictures in which fire is not included in an image. Negative images are uploaded so it can

differentiate between the two classes, which helps to an algorithm which features are more likely to be a fire and which are not expected to be fire. The next step is to use the AdaBoost technique and cascading technique to make powerful classification and accuracy in recognizing the flame.

### 3.1 Adaptive Boosting (Ada Boost)

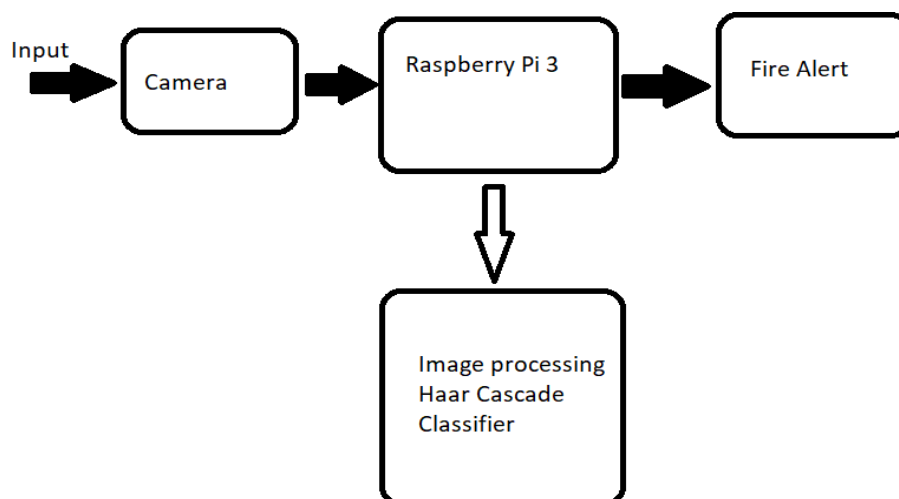
Adaboost algorithm is a learning algorithm used to train the classifier and selects the best subset of features. The algorithm learns from the data that is given and determine the false positive and true negatives in data. For an equation:  $F(x) = a_1f_1(x) + a_2f_2(x) + \dots$ . Here  $F(x)$  is the strong classifier, and  $a_1f_1(x)$ ,  $a_2f_2(x)$  are weak classifiers where  $a_1$ ,  $a_2$  are the weights, and  $f_1$ ,  $f_2$  are the features. So the strong classifier is made up of many weak classifiers since one weak classifier is not good as adding more classifiers make the algorithm stronger and accurate, which is called an ensemble.

### 3.2 Cascading

Cascade classifier is used for the accuracy of identification. It is composed of several stages consisting of a strong classifier. Those strong classifiers are passed by So all the features are grouped in several stages where each stage has a certain number of features. The use of these several stages is used to determine whether the given input sub window has features of fire or not if there are no features of fire, then the given sub-window is discarded and fails to go for other stages. Here instead of using all the features more than 6000 in a window, the features are a group in different stages of classifiers, there are some stages stage 1 and stage 2. Usually, the first few stages will contain very less numbers of features. If the window fails, it is discarded if not apply the second stage of features and continue the process. The window or stage which passes all the features of fire, then it is detected as fire.

## 4. System Design

For this system fire detection using image processing, we are using Raspberry Pi 3 Model B as a processing device. It is the main computer for all the code and image processing. The USB camera is a primary source of input for processing the live video that fetches through the USB camera. After processing of video, which is done by using the Haar Cascade Classifier algorithm, the system continuously monitors the surroundings. If there is a fire in the surroundings, the webcam detects the fire or flame then the user gets fire alert. If there is no fire in the scene, then the video is continuously processed in the code until the fire is detected, so if there is fire, then the only user gets the alert of fire.



## 5. Algorithm

The first step is the training of the classifier, as mentioned early. Training for the highly accurate classifier needs lots of time and processing power, so here we only used a little number of pictures. After training the fire cascade classifier, the captured frame from the webcam is converted into grayscale. The reason for converting frame into grayscale is because the frame captured by webcam is in RGB color.

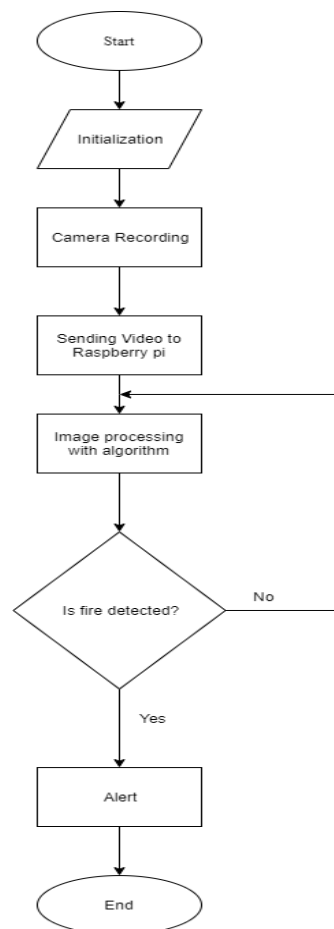
If the image is converted into grayscale, there will be only one channel, either black or white, which is easy to process. After the conversion, the fire classifier is used with inbuilt function detect Multiscale, which will help to find the features and location of images. Parameters like scale factor and min neighbor are passed. These factors are a nessential factor in



detecting fire. A scale factor is used for creating of scale pyramid because while training the classifier fixed size of the image is trained, so the scale factor will allow rescaling the size of an input frame to detect the fire. Another parameter min neighbor will determine the quality of an image here for this thesis min neighbor factor is given 7. . 21

## 6. Flowchart

Flowchart is the flowchart of the system that shows the working system. Firstly the video is fed through the webcam, and the video is then forwarded into Raspberry Pi for processing. It will use a cascade classifier to compare the image which is available in the classifier. If a fire is detected, the code will execute, and there will be the fire alert message, and if a fire is not detected, then the system will continue; there will be no alert of fire. The system stays in the same loop until the fire is detected in the aim place.



## 7. Result

The main goal of the thesis was to make a prototype fire detector system. After the completion of the code and installed in the Raspberry Pi, the fire was detected. It shows the detection of fire, fire is not detected because of the distance and the size of the flame. For the system the flame is look like a small led bulb so it is hard to detect. Finally, the result shows the fire is detected. In, there is a green rectangle box that appears when the flame has detected. The size of the rectangle is varied in the case how large the fire is. So while coding, some threshold values have been given for the rectangle to detect the fire region correctly. The threshold value is provided by setting the parameters for the value of scale factor in the case of this project 1.1 is the scale factor, and 7 is the min neighbor. The scale factor is used for the control of the image pyramid while detecting the object. If the scale factor is too big, then it will be difficult to identify the object. Hence it will be less accurate, and if the scale factor value is less than there will be false detection. Since the experiment was conducted in the room light condition, the scale factor is also calibrated according to the room light to get the accurate result





#### 4. EMBEDDED SYSTEM

An embedded system is the computer system with a combination of software and hardware that has the processing power, input-output functions with computer memory. Which is designed particular purposes to perform the specific tasks in the given time. They are used in the larger systems like in the industry, farming, automobiles, household appliances, etc. embedded systems can be both user interface UI or the complex graphical user interface GUI. An embedded system is based upon the microcontroller or the microprocessors. Here the microcontroller or microprocessors are used for processing, and there is a need for other components like memory chips, which are designed for the specific system. There are many characteristics of embedded systems like it has very sophisticated functionality for each appliance. It has a very low manufacturing cost. The embedded system is designed for specific tasks. It also has advantages of low power consumption because of the use of microcontrollers they don't take so much power to perform the tasks. Here the embedded system is designed for the application-dependent processor, not the general-purpose processor like in the computers. Embedded systems are used in the various field; it is the backbone of the automatic system. Embedded systems are used in the diverse field of engineering, like control engineering, operating systems, IC designing, Computer Architecture, Digital Electronics, Analog Electronics, Data Communication, Software Engineering.

##### 1. Raspberry Pi

Raspberry Pi is the single board computer sized of the credit card, which is inexpensive, and due to its size, it is good for the portability and usability so that it can be used and placed wherever desired. Raspberry Pi 3 is the third model of the raspberry family. Raspberry Pi is based upon the Linux-based operating system with the quad-core ARM CPU with the processor speed of 700MHz to 1.4 GHz. It has a 1GB of RAM with the 4 USB, 1 Ethernet, Headphone, one micro SD slots, and one HDMI ports. And the lower level output of 40 GPIO pin which allows connecting with everything like camera, motors whole other things. Raspberry is mainly designed for promoting the teaching of computers and technology using different programming languages. But due to its cost and modification, Raspberry Pi became very popular with all-around people who are interested in robotics and technology.

Software Requirements of the application is based upon the python language, which is high-level programming language easy to use, and with the python, the system uses the open CV library for all the image processing systems. Also, the system here uses Raspberry Pi as a central processing unit pi is more suitable for the python because it is preinstalled in the OS of the Raspberry Pi.

##### 2. Python

Python is considered as the general-purpose, high-level programming language. With the python, the readability of code is enhanced, compared to another language like C, C++, java. The use of many syntaxes and the procedure of writing code is way too difficult to read and lengthy. Whereas in python, there are fewer lines of code. The major point of using python is the simplicity and compatibility of the code. Python also supports multiple libraries and 26 frameworks, which makes easy and gives more space for developers to create and complete the project. Some of the libraries which are used in the system is NumPy. It is an important library for the multidimensional array function, matplotlib, which is the plotting library that gives the visual datasets and also helpful for the statistical analysis of the data. Due to simplicity, mostly python is used in R&D. Mostly now a days, python is used for Machine Learning and Artificial Intelligence.

##### 3. Open CV

OpenCV is an open-source package and machine learning library, which is designed for realtime computer vision applications. Open CV is the cross-platform library that supports many programming languages like Python, Java, C++, C, etc. This library is originally designed by the Intel corporation and free for use under the open-source BSD license. It is one of the most widely used libraries for implementing video detection, image detection, deep learning application, machine learning, human-computer interaction, 2D, and 3D feature toolkits. The library has more than 2500 algorithms, which include both computer vision and machine learning algorithms. Some of the syntaxes that are used in the program are: `Import cv2` # used for importing the open cv library `Import NumPy as np` # importing the NumPy array `Cv2.VideoCapture()` # used for capturing the video from the camera `Cv2.cvtColor()` # used for converting an image from one color space to another `gray = cv2.cvtColor(resized, cv2.COLOR_BGR2GRAY)` # used for converting RGB color to gray `fire = fire_detection.detectMultiscale(gray, 1.1, 7)` # using fire classifier in which objects are loaded with the `fire_detection.xml` # detect Multiscale A multi-scales detection algorithm is another important steps in the fire detection algorithm. This feature is used in order to detect the fire of any size because the size of the fire is unknown. since learning and testing are based on rectangular feature, those features should be calculated in any scale. In this project, we used the



scales by a factor of 1.1. In each iteration, the width and height of the rectangle will increase to 1.1 times of the previous one. Another parameter is min Neighbors, which is 7 in the project; this parameter specifies how many neighbors each candidate rectangle should retain it. The parameter directly affects the quality of detected fire, higher-value results in less detection of fire with a high-quality result.

## 5.LIMITATIONS AND FUTURE IMPROVEMENTS

Due to safety reasons, the effectiveness of the system cannot be tested in the full phase, so here for the prototype demo, the lighter is used as the source of fire as it has the same features and characteristics of the fire. There are some limitations to the prototype. There is some error in detection when the fire is far away from the camera because of the size of the flame. The project was done with different methodology as well by using RGB color model concept, the RGB filter was used to extract Red, Green and Blue components for each pixel in the frame and then every pixel from the frame was passed through the condition:

- If  $R > G > B$
- If  $R > R_t$  (threshold value 0-255).

which is based upon the intensity of light in a particular environment. The result of this methodology was moderate and not so successful. Every time if there is a light source with the same color value of fire, the system detects it as a fire. The practical was conducted in the room. The light source has almost the same value as a color of fire, and the system detects it as fire. The system can be made more accurate and precise by using other machine learning algorithms like YOLO (You Only Look One)

## 6.CONCLUSION

The project aimed to detect fire with a different approach rather than using an existing system. Currently, systems like a smoke detector and sprinkler water discharge systems are used, which are very useful and work at its best. But there are certain limitations to these systems. The thesis is conducted to optimize the current system. As technology is getting better and better as to keep it up with the technology and to minimize the limitations also, the new system has been created. By using image processing technology for detecting the fire, these limitations can be reduced because in this system camera acts like a human eye, as it detects a fire, the video is captured, and the image is processed using the software alert user. Thanks to Raspberry Pi, because of the size, cost-effectiveness, simplicity, and portability, it can be used everywhere. The prototype successfully detects fire. The thesis gives the review analysis, designing system, and algorithm, test, and result.

## 7.REFERENCES

- [1] S. Y. Foo, "A fuzzy logic approach to fire detection in aircraft dry bays and engine compartments", IEEE Trans. Industrial Electronics, vol. 47, no. 5, pp. 1161-1171, 2000
- [2] B.U. Töreyn, Y. Dedeoglu, U. Gudukbay and A. E. Cetin, "Computer vision based method for realtime fire and flame detection", Patt. Recog. L., vol. 27, pp. 49-58, 2006.
- [3] C-C Ho, "Machine vision-based real-time early flame and smoke detection", Meas. Sci. Technol., vol. 20, pp. 1-13, 2009.
- [4] B.C. Ko, K-H Cheong and J-Y Nam, "Fire detection based on vision sensor and support vector machines", Fire Safety J., vol. 44, pp. 322-329, 2009.
- [5] M. Doostfateme and S. C. Kremer, "New directions in fuzzy automata", Int. J. of Approximate Reasoning, vol. 38, pp. 175-214, 2005.
- [6] F. Lin and H. Ying, "Modeling and control of fuzzy discrete event systems", IEEE Trans. On SMC-B, vol. 32, pp. 408-415, 2002
- [7] C. M. Bishop, "Pattern recognition and machine learning, " Springer Publisher, 2006.
- [8] A. Patcha and J. M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends, " Comput. Netw., vol. 51, no. 12, pp. 3448-3470, 2007. (Pubitemid 46921030)
- [9] W. Phillips III, M. Shah and N. V. Lobo, "Flame recognition in video, " Pattern Recognition Letters, vol. 23(1-3), pp. 319-327, 2002. (Pubitemid 33119526)
- [10] B. U. Toreyn, Y. Dedeoglu, and A. E. Petin, "Flame detection in video using hidden markov models, " ICIP '05, vol. 2, no. 2, pp. 1230-1233, 2005. (Pubitemid 44539168)
- [11] T. Chen, P. Wu and Y. Chiou, "An early fire-Detection method based on image processing, " ICIP '04, vol. 3, no. 24-27, pp. 1707-1710, 2004. (Pubitemid 40820559)



[12] X. L. Zhou, F. X. Yu, Y. C. Wen, Z. M. Lu and G. H. Song, "Early fire detection based on flame contours in video, " Information Technology Journal, 2010.