Fire Detection Use CCTV with Image Processing Based Raspberry Pi

H Pranamurti¹, A Murti¹ and C Setianingsih¹

¹ Telkom University, Indonesia

Abstract. Image Processing is a form of processing the signal with the input of image and transformed into another image as output with certain techniques. In this opportunity we will create a fire detector device using Raspberry Pi and Raspberry Pi Camera and apply the concept of Image Processing and Internet of Things (IoT) in order to get the real-time detection results. In this study Haar Cascade Classifier Algorithm is used for the preprocessing of image, to detect the correct image of fire. As long as the device in active condition, the camera will continue to monitor the conditions in the room. If there is a fire detected by the camera, then the fire image will be processed using Image Processing then the result is a data string sent to the server and forwarded to the user's smartphone (using the concept of IoT) as a warning (alert) there is fire in the room.

1. Introduction
As we know in a company or industrial plants are very prone to fires, whose name occurs whether caused by man or machine error in production and even not rarely also caused by short-circuit or shorted power.
Quoted from www.wartakota.tribunnews.com, according to the Chief of Department of Fire and rescue Countermeasures (Sudin Gulkarmat) South Jakarta, Irwan said, from January to December 2017, the total number of fires in South Jakarta reached 348 cases. That number has increased compared to year 2016 to reach as many as 244 cases.
Typically, these fires occur in an area that is less oversight such as warehouse storage. Originally probably only happen a small spark of fire on the room, but due to the lack of supervision and the slowness anticipation that is done then the sparks that burn the entire contents of the room so that the fire is widespread.
Growing technology, came the tools like camera surveillance CCTV or is usually called. CCTV is used to monitor or oversee the rooms there is less supervision, so if there are such incidents spark can be rapidly anticipation before the spark of the fire spread and cause a fire. However, even though it has CCTV surveillance with no fixed rarely happen that the name fire. It will be based in the background by the lack supervision off. Even though we've been keeping an eye on a room with CCTV but if the officer's supervisors off guard and not immediate do anticipation when the api recorded by CCTV and thus are not denied the fire kept coming.
Because the CCTV can only supervise, then to add a function or applied a method on CCTV feature named image processing or image processing. So, this image processing is any form of signal processing where the input is an image like a photo or video, while its output can be either a picture or a number of characteristics or parameters related to the image. By combining image processing in camera surveillance, then when the camera surveillance to detect or record the presence of sparks packed image processing or image processing will continue to cultivate the image that sparks and displayed on the monitor supervision.
But if only shown without any notice then it's just the same, therefore we can combine these image processing with a concept called IoT or Internet of Things. As quoted from www.id.techinasia.com, according to Rowan Trollope as Senior Vice President and General Manager of the field of application of the IoT from Cisco, the current internet network has been able to handle 340 trillion trillion devices thanks to the presence of protocols IPv6. This certainly must be supported by the device makers IoT so as not to create devices that send requests too often, which can overload the internet network. So IoT is able to connect something through the internet such as receive and send pictures or data string examples. With the IoT is string data in the form of notifications (alerts) the result of image processing image fire is detected by the camera we can send to the server with the help of the internet and the server then we submit to the supervisory officer with the aim of smartphone as a warning that there was a fire in the room. Not denied again on this day and age people could not be separated from the name of a smartphone, by utilizing this then it will be more effective if we send to fire warning smartphone supervisor with the objective information that we send it can directly viewed and acted upon.

2. Related Work

2.1. Localization of White Blood Cell Images using Haar Cascade Classifiers (Rezha Aditya Maulana Budiman, Balza Achmad, Faridah, Agus Arif, Nopriadi, Luthfi Zharif, 2016)

Rezha Aditya and his friends tried to locating the white blood position in an image use the Haar Cascade Classifier Algorithm on this papper. And the result showed that the Haar Cascade Classifier can locating white blood cells with precision and recall values of 95% and 74% respectively. With Haar Cascade Classifier Algorithm also able to differentitate the white blood cells from other objects that have resemble color.

2.2. Eye-Gaze Tracking System by Haar Cascade Classifier (Yunyang Li, Xin Xu, Nan Mu1, Li Chen, 2016)

Like the title this paper discuss about a real-time eye gaze tracking system using Haar Cascade Classifier Algorithm. Yunyang Li and teams want to calculate the position of eye gaze with Haar Cascade Classifier Algorithm based on the rectangular features of human eye. The features of rectangular of human eye is adopted to match the coordinates of represented where an object is looking. And the results show that Haar Cascade Classifier Algorithm can validate the effectiveness of the system.

2.3. A Human Face Detection Algorithm Via Haar Cascade Classifier Combined with Three Additional Classifiers (Li Cuimei, Qi Zhiliang, Jia Nan, Wu Jianhua, 2017)

In this paper Li Cuimei and teams want to discuss about detection the human face used Haar Cascade Classifier Algorithm and also combined the algorithm with a new three additional weak classifiers algorithm. The three additional classifiers are first skin hue histogram matching, and second eyes detection and also the last are mouth detection. The mouth detection is used to utilized the remaining of non-human faces and to make the false positive rate is further decreased.

3. Research Method

3.1. Image processing

In image processing method we can get the information about the object by a few things such as image processing, image models, and RGB [1].
3.1.1. Image Processing

Image Processing is a form of processing the signal with the input of image and transformed into another image as output with certain techniques. In the image processing there are techniques and processes to reduce or eliminate the effects of degradation on image includes the techniques of repair or improvement of image (image enhancement), the restoration of the image (image restoration) and the special transformation, another subject of digital image processing such as image encoding, image segmentation (image segmentation), representation and the task image (image representation and description).

3.1.2. Model Image

Example of Model Image is Conversion of discrete image coordinate system shown in Figure 1.

The image seen is the reflected light of everyday objects. The function $f(x, y)$ can be seen as a function with two elements, the first is the magnitude of the light source that complements our view of objects (illumination) or can be written $i(x, y)$, and the second is the quantity of light that reflected by object in our view (reflectance component) or can be written $r(x, y)$, written with the following equation 1.

$$f(x,y)=i(x,y)r(x,y)$$
$$0 < f(x,y) < \infty$$
$$0 < r(x,y) < 1$$

(1)

The above equation indicates that the value of the reflected is limited by the value 0 (total absorption) and a value of 1 (total reflectance) of the function $f(x, y)$ coordinate special good discrete as well as the level of its brightness. We can consider a digital image (next will be abbreviated with the image) as the matrix size the rows and columns is MxN shows the dots shown in the following equation 2.

$$f(x,y) = \begin{bmatrix}
    f(0,0) & f(0,1) & f(0,N-1) \\
    f(1,0) & f(1,1) & f(1,N-1) \\
    \vdots & \vdots & \vdots \\
    f(M-1,0) & f(M-1,1) & f(M-1,N-1)
\end{bmatrix}$$

(2)

3.2. RGB Method

Color references in RGB is the color white which is combined (tristimulus) from the three main colours with values of $RN = GN = BN = 1$. The following table shows the combined value of the three colours are for some of the major colors of the color coordinates of the NTSC [2].
Table 1. RGB table.

The Figure 2 below shown a picture of the Tristimulus Color Diagram of the System Krominan and the main recipient of NTSC.

![Tristimulus color diagram of the system krominan and the receiver main.](image)

**Figure 2.** Trimulus color diagram of the system krominan and the receiver main.

3.3. HSV Method
Beside the RGB method, there is a supporting method of image processing called HSV color space (Hue, Saturation, Value) Method. [2]. First, Hue (H) is kinds of colors (e.g. red, green, or yellow). Hue is represented as well as the degree of the angle whose value range from 0° to 360° (although for some applications the normalized from 0% to 100%). Second, Saturation (S) is represented as the distance from the axis of the light black-and-white. The value ranges from 0% to 100%. And the last is Value (V), which is on the shaft height is represented as black and white. The possibility of distance values ranges from 0% to 100%. A value of 0 is always black. Based on saturation, 100 can be white or the level of saturation is more even less. [2].

3.4. Haar Cascade Classifier
The Algorithm we use in this project is Haar Cascade Classifier, which is a method for detecting objects in an image, and the method of Haar Cascade Classifier is an object detection method developed by Viola Jones. This method is based on Haar-like features, combined with the classifier cascade are strengthened. Haar-like features are features that are widely used in detection of objects, offering rapid extraction process and are able to represent a lower resolution image. This method has been successfully applied in many object detection [8].

The classifier usually done with training from some of the examples of drawings of simple and positive examples of negative images, which have the same size. The area is marked with 1 Classifier for rated similar to object or 0 to be assessed are not similar. After the training, the Classifier can be found all across the entire object with a region of the image. And to detect the target area more accurately, the scanning window size changed adaptively by Classifier. During the process of
classification, the model features the optimal rectangle are selected in accordance with the objects and the scanning window [9].

Figure 3. Haar-like features representation.

4. System Design and Overview

4.1. System Overview
The system will be built using the fire detection system is an image processing-based image processing or Raspberry Pi and IoT. This system uses a Haar Classifier Cascade Algorithm. The system will be described in Figure 4.

In this system the camera functions as a Catcher that would later become the image input (input) Raspberry Pi. Raspberry Pi used are the Raspberry Pi 3 Model B which will be the brains of the system. All the image data obtained will be in sports in Raspberry Pi 3 Model B by using Haar Classifier Cascade algorithm.

The image data that is already in the sport in Raspberry Pi 3 Model B will then be sent to the server in the form of string data in the form of warning (alert), here I use the platform server Antares is a Platform of IoT, and subsequently from the server will send data warning (alert) and display it on the user's smartphone.

4.2. Flowchart System
The flowchart system from this work show in Figure 5 below. First the camera is in ready status and show all condition of the room. Image from camera will process in Raspberry Pi with Haar Cascade Classifier Algorithm, and if the image is fire, system will recognition the fire and send the string data
to sever and also send it to user (smartphone) for the notification to make the user know the condition about the room.

![Flowchart system](image)

**Figure 5.** Flowchart system.

4.3. Preparation Haar Cascade Classifier Training

Before use the Haar Cascade Classifier, we need to prepare all of the needed to create the good Haar Cascade Classifier. This preparation steps are used for prepare all of the training data. For training a boosted cascade of weak classifiers we need a bunch set of positive samples (containing actual objects you want to detect) and a bunch set of negative images (containing everything you do not want to detect). The set of negative samples must be prepared manually, whereas set of positive samples is created using the opencv_createsamples application.

1. Negative Samples

First we need to prepare the negative samples. The negative samples are taken from any images that not contain the objects that you want to detect. These images are generated with some method and called the background samples or background images.

The image that used in the negative samples should be equal or larger than the size of the object image that you want to detect, because these negative samples are used to subsample a negative image into several image samples which has the same training window size.

2. Positive Samples

After we done prepare the negative samples, now we need to prepare the positive samples. Positive samples are created by the opencv_createsamples command on the Raspberry Pi. The command are used the boosting process to define what the model should actually look for when trying to find the objects that you want to detect. And before that we need prepare more than a single positive samples to make the positive samples with these command.

The bunch of images on positive samples are created to a text file that similar to the background description file.

3. Cascade Training

The last step we need to do after we finish prepare negative and positive samples, is the training of the cascade classifiers step. This step is processed the positive and negative samples that was prepared beforehand. The process of cascade training is using the command opencv_trainscascade to make the file cascade.
After the cascade training process finished, the trained cascade will be saved in xml file type. Training is finished and cascade classifier are ready to use. it might be necessary to place some before their text citation. Figures should never appear within or after the reference list.

5. Result

5.1. Result and testing at distance of 50 centimeters
For the first test we do it at distance of 50 centimeters. In this test we are measured the accuracy of the system used the light intensity and the distance of fire from the camera. For the test that was done we get each different accuracy depending on the level of intensity of light, we can see the result in the Table 2 below.

| Distance | Light | Fake Fire | Real Fire | Accuracy |
|----------|-------|-----------|-----------|----------|
| 50 cm    | 64 lx | 8         | 1         | 12.50%   |
|          | 56 lx | 6         | 1         | 16.70%   |
|          | 36 lx | 3         | 1         | 33.30%   |
|          | 28 lx | 2         | 1         | 50%      |
|          | 0 lx  | 1         | 1         | 100%     |

5.2. Result and testing at distance 100 centimeters
For the second test, we change the distance to 100 centimeters. And the result is same like the results obtained with a distance of 50 centimeters. The results can be seen in the Table 3 below.

| Distance | Light | Fake Fire | Real Fire | Accuracy |
|----------|-------|-----------|-----------|----------|
| 100 cm   | 64 lx | 8         | 1         | 12.50%   |
|          | 56 lx | 6         | 1         | 16.70%   |
|          | 36 lx | 3         | 1         | 33.30%   |
|          | 28 lx | 2         | 1         | 50%      |
|          | 0 lx  | 1         | 1         | 100%     |

5.3. Image of the test

![Figure 6. Test with 64 lx.](image)
Figure 7. Test with 56 1x.

Figure 8. Test with 36 1x.

Figure 9. Test with 28 1x.

Figure 10. Test with 0 1x.
6. Conclusion
Of the tests we have done, we can conclude that distance does not determine the level of accuracy, so that wherever the fire position accuracy measurement values will be the same. While the intensity of the light is very influential towards the accuracy of the measurements, the brighter of the light the smaller value of the accuracy of the measurements obtained. For the result we can see in the Table 4. Below.

Table 4. Conclusion of The Result.

| Indicator | Conclusion |
|-----------|------------|
| Distance  | The distance does not determine the level of accuracy, so that wherever the position of the fire the accuracy of the measurement values will be the same |
| Light     | The intensity of the light is very influential towards the accuracy of the measurements, the brighter of the light the smaller value of the accuracy of the measurements obtained. |

7. References
[1] Putri AR 2016 Pengolahan Citra dengan Menggunakan Webcam pada Kendaraan Bergerak di Jalan Raya JIPI (Jurnal Ilmiah Pendidikan Informatika) Vol 1 No 1 pp 1-6.
[2] Hutasoit SJ Susanto E Nugraha R 2016 Rancang Bangun dan Implementasi Prototipe Pendeteksi dan Pemadam Api Menggunakan Image Processing pada Quadcopter Universitas Telkom S1 Teknik Elektro Bandung
[3] Sharmila B Karalan N Neduraman D 2015 Image Processing on DSP Environment Using OpenCV Vol 5 Issue 2 ISSN: 2277 128X International Journal of Advanced Research in Computer Science and Software Engineering
[4] Palekar RR Parab SU Parikh DP and Kamble VN 2017 Real Time License Plate Detection Using OpenCV and Tesseract International Conference on Communication and Signal Processing India.
[5] Vankatesan S Jawahar A Varsha S Roshne N 2017 Design and Implementation of an Automated Security System using Twilio Messaging Service 2017 International Conference on Smart Cities, Automation & Intelligent Computing Systems Yogyakarta, Indonesia
[6] Abdalla GOE VeeramaniKandasamy T 2017 Implementation of Spy Robot for A Surveillance System using Internet Protocol of Raspberry Pi 2017 2nd IEEE International Conference On Recent Trends In Electronics Information & Communication Technology India.
[7] Lekić M Gardasević G 2018 IoT sensor integration to Node-RED platform 17th International Symposium INFOTEH-JAHORINA
[8] Budiman RAM Achmad B Faridah Arif A Nopriadi Zharif L 2016 Localization of White Blood Cell Images using Haar Cascade Classifiers 2016 1st International Conference on Biomedical Engineering (IBIOMED) Yogyakarta Indonesia
[9] Li Y Xu Xi Mu1 N Chen L 2016 Eye-Gaze Tracking System By Haar Cascade Classifier 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)
[10] Cuimei L Zhiliang Q Nan J Jianhua W 2017 Human face detection algorithm via Haar cascade classifier combined with three additional classifiers 2017 IEEE 13th International Conference on Electronic Measurement & Instruments