

# Aryu\_Proceeding\_2014-1.pdf

*by*

---

**Submission date:** 07-Feb-2021 09:58AM (UTC+0700)

**Submission ID:** 1503341285

**File name:** Aryu\_Proceeding\_2014-1.pdf (815.81K)

**Word count:** 2928

**Character count:** 15400

# Raspberry Pi Based Laser Spot Detection

2  
Aryuanto Soetedjo

Department of Electrical Engineering  
National Institute of Technology (ITN)  
Malang, Indonesia  
aryuanto@gmail.com

M. Ibrahim Ashari

Department of Electrical Engineering  
National Institute of Technology (ITN)  
Malang, Indonesia

Ali Mahmudi

Department of Informatics Engineering  
National Institute of Technology (ITN)  
Malang, Indonesia

Yusuf Ismail Nakhoda

Department of Electrical Engineering  
National Institute of Technology (ITN)  
Malang, Indonesia

**Abstract**—This paper presents an application of the Raspberry Pi for detecting laser spot using the OpenCV library. To capture the image, a Web-Camera connected to the USB port of the Raspberry Pi is employed. Since the Raspberry Pi is small in size, the module could be mounted on the gun for laser shooting. Compared to the existing laser shooting, our method has benefit in the flexibility of laser shooting mechanism, where the shooter could move the gun freely without the restriction of the camera's cable connection. The image processing tasks and interfacing to the camera could be accomplished by the Raspberry Pi mounted on the gun. Experimental results show that the proposed system could detect the laser spots under the varying backgrounds, lighting conditions, and shooting distances effectively. The processing speed of 18 fps could be achieved for real time implementation.

**Keywords**—Raspberry Pi; laser shooting; OpenCV; image processing.

## 3 I. INTRODUCTION

The Raspberry Pi is a credit card size single board computer. Due to the small size and low cost, there are many applications employed it as the embedded components. The Raspberry Pi was incorporated with GPS and GSM modules for enhancing the navigation system which is integrated with the traffic lights [1]. In the telemedical application [2], the Raspberry Pi was employed as the main processor to monitor the cardiac function. The performance of monitoring system was increased by introducing the digital signal processing implemented on the low-cost Raspberry Pi module.

The Raspberry Pi was used to perform the image processing tasks in ASV (Autonomous Surface Vehicle) [3]. Two webcams are connected to the USB ports to accomplish the stereo vision system. The GPIO port of the Raspberry Pi is connected to the ultrasonic sensor for calculating the distance to the obstacle.

In the wireless sensor network systems, the Raspberry Pi module was used as the nodes [4],[5]. In [4], the Raspberry Pi

was used as the web server for allowing the monitoring process via the Internet. In [5], the Raspberry Pi was used as the coordinator node for monitoring the pH and the temperature in the wine making process.

The works presented above take the advantages of Raspberry Pi such as: low cost with higher performances of processing, memory and connectivity [4]. In this work, we propose to employ the Raspberry Pi for detecting the laser spot.

The systems to detect laser spot have been proposed for several applications. In [6], they proposed a system to detect the laser spot for interpreting its behaviour. The program translates the behaviour of detected laser spot into specific commands. The laser spot detection was employed for replacing the mouse events applied in the projected screen [7].

The laser spot detection was also employed in the laser shooting applications [8]-[12]. Basically, there are two types of the laser shooting detection systems, i.e. when the camera is installed in a fixed position in front of the target screen [8], [9], and when the camera is mounted on the gun [10] – [12].

In the laser shooting applications, the camera systems play an important role. Thus the laser shooting simulator system employ a computer to interface with the camera and to perform the image processing tasks. The common approach uses the USB camera to capture the image. Then the image processing tasks, such as laser color detection, laser spot location, and the homography transform are carried out by the computer. This system is suitable for the fixed camera system [8], [9]. However in the camera on gun system, the USB camera's cable restricts the movement of gun [10], [11]. To overcome the limitation, our previous work [12] proposed an embedded camera (CMUCam4) to avoid the cable connection between camera and computer.

The CMUCam4 is an embedded camera consists of the OmniVision 9665 CMOS camera and the Parallax P8X32A processor [12]. It could track the colored objects of 160 x 120 image at 30fps. Unfortunately, it could not perform the

complex image processing tasks. <sup>1</sup> this paper, we extend our previous work by employing the Raspberry Pi module as the image processor. Since the Raspberry Pi is a small size computer equipped with the USB ports, we could mount the USB camera and the Raspberry Pi on the gun. It offers an efficient camera on the gun system. Further the Raspberry Pi is able to handle the image processing tasks, especially using the OpenCV library.

The rest of paper is organized as follows. Section 2 describes the hardware configuration. Section 3 describes the laser spot detection. The software implementation is described in section 4. Section 5 presents the experimental results. Conclusions are covered in section 6.

## II. HARDWARE CONFIGURATION

Fig. 1 depicts the hardware configuration of our proposed laser shooting detection system. It consists of three main components: a) the Web camera; b) the Raspberry Pi; c) the laser pointer. Those three components are mounted on the gun. The laser shooting detection system works as follows. At first, the shooter aims the gun to the shooting target. The shooter fires the gun by pressing the trigger which emits the laser beam from the laser pointer. The camera will capture the target image and locating the position of laser spot by the image processing techniques.

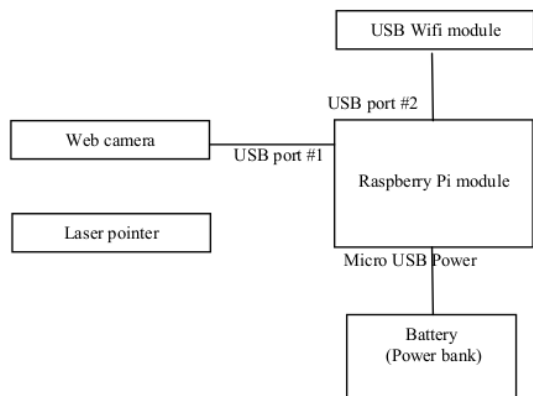


Fig. 1. Hardware configuration.

The Web camera used in the system is a Logitech HD Pro Webcam C920. The maximum image resolution is 1920 x 1080 pixels. This camera is equipped with the Carl Zeiss lens for providing the brighter image. The Red LaserScope laser pointer is adopted for shooting the target. The laser wavelength is 532nm, and the power is 5 mW. This laser pointer is usually used for aiming purpose in the air gun shooting sport.

The Raspberry Pi model B is employed as the main processor. It has 512 MB RAM and an ARM1176JZF-S 700 MHz processor. It is equipped with two USB ports, where the first port is connected to the Web camera, and the second one

is connected to the USB Wifi module. The Wifi module is used to communicate with the computer for remote monitoring and control. Since the module is mounted on the gun, it should be powered from the battery. Fortunately, the Raspberry Pi could be powered by the popular power banks which are usually used for smartphones.

## III. ALGORITHM

The laser spot detection algorithm is illustrated in Fig. 2. The algorithm starts by capturing the color images. Then the simple color thresholding similar to our previous work [12] is employed. The color thresholding is expressed as [12]

$$\begin{aligned} & \text{If } (R_{\min} \leq R(x,y) \leq R_{\max}) \text{ AND } (G_{\min} \leq G(x,y) \leq G_{\max}) \\ & \text{AND } (B_{\min} \leq B(x,y) \leq B_{\max}) \text{ THEN pixel}(x,y) \text{ is RED} \end{aligned} \quad (1)$$

where  $R(x,y)$ ,  $G(x,y)$ ,  $B(x,y)$  are the red, green, blue components of  $\text{pixel}(x,y)$  respectively;  $R_{\min}$ ,  $G_{\min}$ ,  $B_{\min}$ ,  $R_{\max}$ ,  $G_{\max}$ ,  $B_{\max}$  are the thresholds.

After color thresholding, the blob image (black and white image) are obtained. In our previous work [12], due to the limitation of the embedded camera (CMUcam4), the detected blobs are processed directly. In some cases, it may cause the false detection. To overcome this limitation, we introduce the blob checking as illustrated in the following. At first, it finds the contour of detected blobs. After contours are found, the contour areas are calculated. The blob is considered as the laser spot if the contour area satisfies the certain threshold.

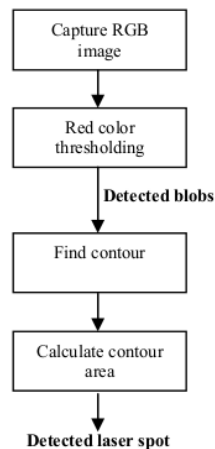


Fig. 2. Detection algorithm.

## IV. SOFTWARE IMPLEMENTATION

The Raspberry Pi runs under the Raspbian, a free operating system based on Debian optimized for the Raspberry Pi. The algorithm described above is implemented on the Raspberry Pi using the OpenCV [13]. Thank to the OpenCV, a free and useful computer vision library. Using the OpenCV, the image

processing tasks could be implemented easily as described in the following.

The first step is to capture the color image from the Webcam. Fortunately, the device driver of Logitech Webcam C920 is easily found for Raspbian operating system. After the driver is installed properly, the next step is to write the C code using OpenCV library. To capture the image from camera, `cvCreateCameraCapture` is used. It is needed to set the property of image, such as the image width and height using

```
cvSetCaptureProperty(capture,
CV_CAP_PROP_FRAME_WIDTH,320);
```

and

```
cvSetCaptureProperty(capture,
CV_CAP_PROP_FRAME_HEIGHT,240);
```

The image processing algorithm is performed frame by frame. Therefore it needs to grab the image using the function `cvQueryFrame`. Then the red color thresholding is easily implemented using the function `cvinRange(image, Scalar(Bmin, Gmin, Rmin), Scalar(Bmax, Gmax, Rmax), thr_image)`, where `image` is the color image; `Scalar(Bmin, Gmin, Rmin)` are the lower bounds of blue, green and red components; `Scalar(Bmax, Gmax, Rmax)` are the upper bounds of blue, green and red components; and `thr_image` is the resulted blob image.

The function `findContours(thr_image, contours, hierarchy, CV_RETR_CCOMP, CV_CHAIN_APPROX_SIMPLE)` is used to find the contour of detected blobs. Finally the function `contourArea` is used to calculate the area of contour. The blob is considered as the laser spot when the contour area is greater than a threshold.

## V. EXPERIMENTAL RESULTS

We conducted several experiments to verify the reliability of proposed system. In the experiments, the Raspberry Pi module is operated remotely from the personal computer running Windows 7 using the Putty and Xming software via the wifi connection. Using this remote connection, the performance of proposed laser spot detection system could be monitored easily on the PC.

In the experiments, the thresholds of Eq. (1) are set as follows:  $R_{min}=200$ ,  $G_{min}=0$ ,  $B_{min}=0$ ,  $R_{max}=255$ ,  $G_{max}=180$ ,  $B_{max}=180$ . The resolution size of the camera is 320 x 240 pixels. This lower resolution is selected to speed up the computation process. Using this resolution, the algorithm is able to track the image in 3 fps, including the tasks to save and display the images, which are used for debugging process. In real applications, it is not necessary to save and display the images. Without saving and displaying images, the speed could be increased to 18 fps.

The proposed method is tested under three different schemes, i.e. a) varying backgrounds; b) varying lighting conditions; c) varying shooting distances. The results are discussed in the following.

Fig. 3 to Fig. 6 illustrate the results of detected laser spots under varying backgrounds. In the figures, the detected spots

are indicated by the blue rectangles drawn on the original images (first rows). The blob images are shown on the second rows. In Fig. 3, the laser spot is emitted on the white background. From the figure, it is clear that the laser spot could be detected properly. In Fig. 4, the image background contains the colourful objects. The figure shows that the laser spot could be detected properly. In Fig. 5, the background is the computer monitor. The blobs image shows that many small objects are detected. However since the contour areas are considered in the detection, only the biggest one is detected as the laser spot as indicated in the figure.

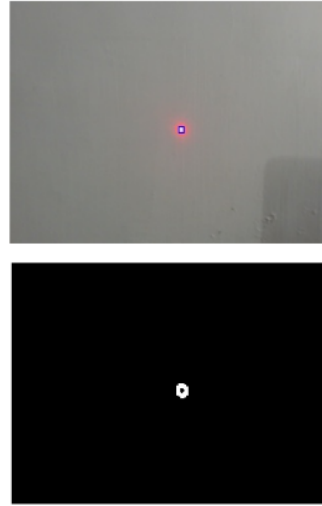


Fig. 3. Detected laser spot on white background.

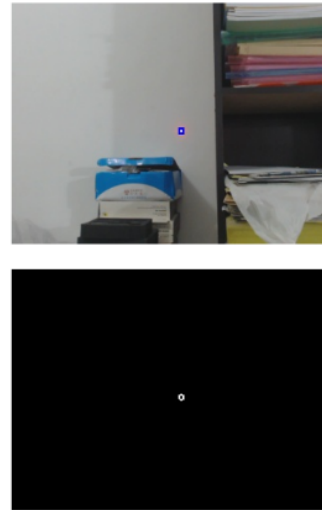


Fig. 4. Detected laser spot on colourful background.

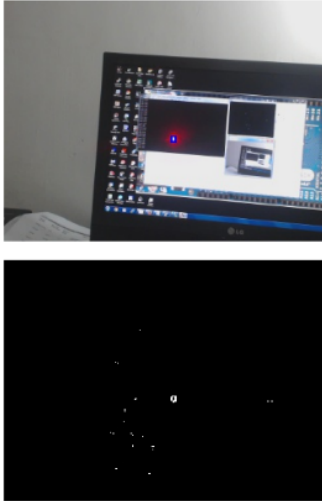


Fig. 5. Detected laser spot on the computer monitor.

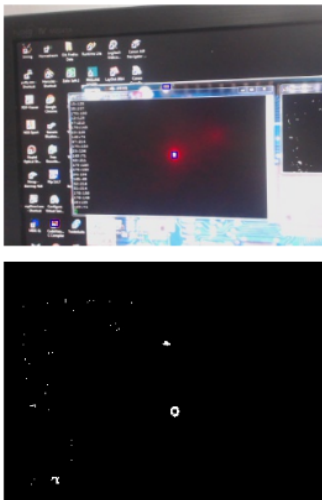


Fig. 6. False detection.

Fig. 6 illustrates the false detection. As shown in the figure, there are many blobs detected. Three of them are in larger sizes, which are drawn by the blue rectangles. Thus they are considered as the laser spots. These false detections are caused by the small red icons appearing on the computer monitor. This problem could be solved by utilizing the position of the camera and the laser pointer. Since they are installed on a fixed position, the detected laser blob could be determined appropriately.

TABLE I. DETECTION RESULTS UNDER THE DIFFERENT LIGHTING CONDITIONS AND SHOOTING DISTANCES.

No.	Lighting condition	Shooting distance	Detection result	Detected coordinate (x,y)
1.	Two lamps ON	0.5 m	Success	(170,158)
2.		1 m	Success	(170,150)
3.		2 m	Success	(170,146)
4.		3 m	Fail	NA
5.	One lamp near the target ON	0.5 m	Success	(170,159)
6.		1 m	Success	(170,151)
7.		2 m	Success	(169,142)
8.		3 m	Success	(169,140)
9.	One lamp near the shooter ON	0.5 m	Success	(170,159)
10.		1 m	Success	(170,149)
11.		2 m	Success	(169,144)
12.		3 m	Success	(169,141)

The experimental results under the varying lighting conditions and shooting distances are given in Table I. In the experiments, three lighting conditions are tested, i.e. : two lamps are switched-on, only one lamp near the target is switched-on, and only one lamp near the shooter is switched-on. While the shooting distances vary from 0.5 m, 1 m, 2 m, and 3 m.

From the table, it is obtained that the proposed system is able to detect the laser spot in eleven conditions, while one condition is failed. When the shooting distance is far (3 m) and the lighting condition is bright (two lamps ON), the system fails to detect the laser spot. It is caused by the limitation of the fixed threshold adopted in the color thresholding method.

Since the position of camera and laser pointer is fixed, the detected coordinate of the laser spot should be fixed. The results given in Table I verifies this condition, where the x-coordinates of detected laser spots are almost the same. However the y-coordinates vary in a small numbers of pixels. From observations, it is caused by the vertical alignment of the laser pointer, where the beam of laser pointer installed on the gun is not a straight line in the vertical direction.

## VI. CONCLUSIONS

In this work, the Raspberry Pi module is employed to perform the image processing task for detecting the laser spot. The proposed system takes the benefits of the Raspberry Pi, which is small in size while providing the ability to run the popular computer vision library called OpenCV. This approach work well in detecting the laser spot, and could be considered as the compromised choice between the small embedded camera system (microcontroller based) and the computer-based image processor.

In future, the system will be expanded to exploit the other features of the OpenCV implemented on the Raspberry Pi.

## Acknowledgment

<sup>1</sup> This work is supported by the “National Strategic Research Grant 2014” from Directorate General of Higher Education, Ministry of National Education and Culture, Republic of Indonesia, No. SP-DIPA-023.04.2.415015/2014.

## References

- [1] D. Punetha, P. Jha, and S. Singh, “An Intelligent Transportation System with Computer System Architecture approaching Safe Passage in Adaptive Traffic Light,” *International Journal of Computer Networks and Wireless Communications*, Vol. 4, No. 2, pp. 143-149, 2014.
- [2] T. Nagy and Z. Gingly, “Low-cost photoplethysmograph solutions using the Raspberry Pi,” *Proceedings of 14th IEEE International Symposium on Computational Intelligence and Informatics*, Budapest, Hungary, 2013.
- [3] R. Neves and A.C. Matos, “Raspberry Pi Based Stereo Vision For Small Size ASVs,” *Proceedings of OCEANS 2013*, San Diego, USA, 2013.
- [4] V. Vujovic and M. Maksimovic, “Raspberry Pi as a Wireless Sensor Node : Performances and Constraints,” *Proceedings of International Convention on Information and Communication Technology, Electronics and Microelectronics*, Opatija, Croatia, 2014.
- [5] S.F.Di Gennaro, A. Matese, M. Mancin, J. Primicerio, “An Open-Source and Low-Cost Monitoring System for Precision Enology,” *International Electronic Conference on Sensors and Applications*, 2014.
- [6] R.B. Widodo, W. Chen, T. Matsumaru, “Laser Spotlight Detection and Interpretation of Its Movement Behavior in Laser Pointer Interface,” *Proceedings of IEEE/SICE International Symposium on System Integration*, pp. 780-785, December 16-18, 2012, Fukuoka, Japan.
- [7] N.W. Kim, S.J. Lee, B.G. Lee and J.J. Lee, “Vision Based Laser Pointer Interaction for Flexible Screens,” *Proceedings of the 12th International Conference on Human-computer Interaction: Interaction Platforms and Techniques*, Vol. 4551, 2007, pp. 845-853.
- [8] A.J.M.L. Junior, G.A.M. Gomez, N.A.C. Junior, A.D. Santos, C.A. Vidal, M. Gattas, “System Model for Shoting Training Based on Interactive Video, Three-dimensional Computer Graphics and Laser Ray Capture,” *Proceedings of 14th Symposium on Virtual and Augmented Reality*, May 28-31, 2012, Ri Janiero, Brazil, pp. 254-260.
- [9] H. W. Liang, B. Kong, “A Shooting Training and Instructing System Based on Image Analysis,” *Proceedings of the IEEE Conference on Information Acquisition*, August 20-23, 2006, Shandong, China, pp. 961-966.
- [10] S.L. Ladha, K.S. Miles, S. Chandran, “Multi-User Natural Interaction with Sensor on Activity,” *Proceedings of 1st IEEE Workshop on User-Centered Computer Vision*, January 15-17, 2013, Florida, USA, pp. 25-30.
- [11] A. Soetedjo, A. Mahmudi, M. I. Ashari, Y.I. Nakhoda, “Camera-Based Shooting Simulator using Color Thresholding Technique,” *Proceedings of 3rd International Conference on Instrumentation Control and Automation*, August 28-30, 2013, Bali, Indonesia, pp. 207-211.
- [12] A. Soetedjo, A. Mahmudi, M.I. Ashari, Y.I. Nakhoda, “Implementation of Sensor on the Gun System Using Embedded Camera for Shooting Training,” *Proceedings of International Conference on Technology, Informatics, Management, Engineering & Environment (TIME-E)*, August 19-21, 2014, Bandung, Indonesia, pp. 69-74.
- [13] <http://opencv.org/>

ORIGINALITY REPORT

---

<b>11</b> %	<b>12</b> %	<b>11</b> %	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

---

PRIMARY SOURCES

---

<b>1</b>	<b>thescipub.com</b> Internet Source	<b>5</b> %
<b>2</b>	Aryunto Soetedjo, M. Ibrahim Ashari, Ali Mahmudi, Yusuf Ismail Nakhoda. "Camera-based shooting simulator using color thresholding techniques", 2013 3rd International Conference on Instrumentation Control and Automation (ICA), 2013 Publication	<b>4</b> %
<b>3</b>	<b>www.ijetcse.com</b> Internet Source	<b>3</b> %

---

Exclude quotes On

Exclude matches < 2%

Exclude bibliography On