

Low Cost Shooting Simulator Based on a Single Board Computer

¹Aryunto Soetedjo, ²Ali Mahmudi, ³Mochammad Ibrahim Ashari and ⁴Yusuf Ismail Nakhoda

^{1,3,4}Department of Electrical Engineering,
Faculty of Industrial Technology, National Institute of Technology, Malang, Indonesia

²Department of Informatics Engineering,
Faculty of Industrial Technology, National Institute of Technology, Malang, Indonesia

Article history

Received: 11-11-2014

Revised: 11-02-2015

Accepted: 04-04-2015

Corresponding Author:
Aryunto Soetedjo
Department of Electrical
Engineering, Faculty of
Industrial Technology, National
Institute of Technology,
Malang, Indonesia
Email: aryunto@gmail.com

Abstract: This paper proposes the camera on weapon system used as the shooting simulator. The single board computer (Raspberry Pi module) is used as the main processor. The OpenCV is implemented on the embedded system to perform the image processing task. The aim of the proposed camera on weapon system is to detect the position of the markers on the target and the laser spot. The detected coordinates are sent to the personal computer using a wireless communication provided by a low power Zigbee protocol. The performance of the proposed system using two camera systems (the Web-Camera and the Raspberry Pi-Camera) are evaluated. Both systems achieve the higher performances in the shooting detection which is greater than 85%, the target detection error which is lower than 0.8% and the small displacement on the arrangement of the camera and the laser pointer which is lower than one pixel. The Raspberry Pi-Camera system has a fast processing speed of 27 fps, while the processing speed of the Web-Camera system is 17 fps.

Keywords: Single Board Computer, Raspberry Pi, Web-Cam, Raspberry Pi-Camera, Zigbee, Camera on Weapon, Laser Spot

Introduction

Shooting simulator is a common instrument used for shooting practice. The most popular one employs the vision system to capture the laser spot emitted by the shooter on the target. There are two techniques called single fixed stationary system and camera on the weapon system (Ladha *et al.*, 2010). In the first technique, a single camera is placed in front of the target. The camera is connected to the computer for further processes. While in the second method, the camera is mounted on the weapon (Soetedjo *et al.*, 2013; 2014; Ladha *et al.*, 2010; 2013). The latter one has an advantage i.e., allowing the multi shooters shot on a single target.

On both systems, the most challenging task is to detect the laser spot and the target by the camera. To detect the laser spot, (Lapointe and Godin, 2005) employed the thresholding method in the intensity images. The RGB color thresholding methods were employed in (Liang and Kong, 2006; Soetedjo *et al.*, 2013). While the HSV color thresholding techniques were employed in (Kim *et al.*, 2007; Widodo *et al.*, 2012).

The previous works employed the Web-Camera or USB camera connected to the computer. Then the image processing techniques are implemented on the computer. This arrangement is suitable for the single fixed camera system. However, it becomes impractical to be implemented on the camera on weapon system, due to the USB cable connection between the camera on the weapon and the computer. To overcome the limitation, our previous work (Soetedjo *et al.*, 2014) proposed the embedded camera system (CMUCam4) equipped with the wireless system. However, since the embedded microcontroller is employed, it could do a simple color thresholding technique only.

In this study, we extend our previous work by adopting a single computer board (Raspberry Pi) as the embedded system. Using this approach, the performance of image processing task could be increased, while maintaining the simplicity of the hardware system. To provide the flexibility of the weapon against the wired connection, a wireless communication is introduced between the embedded system and a personal computer. Two types of the camera systems, i.e., the Web-Camera and the Raspberry Pi-Camera are tested in the experiments.

Proposed System

System Architecture

Figure 1 illustrates the architecture of proposed system. The main components are: (a) The shooting target, (b) the camera on weapon system, (c) the computer system. The shooting target consists of a target screen where the image is projected on it by a LCD projector. The camera on weapon system is a gun (artificial gun) equipped by the camera and the single board computer system. The computer system is a personal computer used to monitor and calculate the shooting score. The wireless modules are employed on the camera on weapon system and the computer system for providing the wireless communication.

The camera on weapon system as illustrated in Fig. 1 consists of five components, i.e., single board computer, camera, laser pointer, wireless module and battery. The operation principle is described in the following. When the shooter aims the gun into the target, a camera installed on the gun captures the image of target. Once the shooter shoots, the laser beam is emitted from the gun to the target.

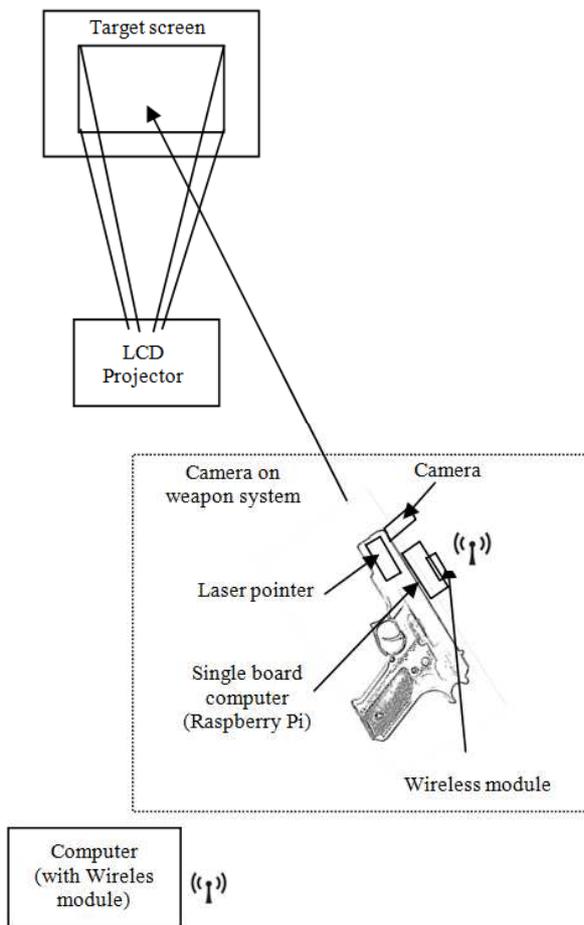


Fig. 1. System architecture

The captured image is processed by a single board computer for locating the target and laser beam on the image. Then the coordinates of detected target are sent to the computer wirelessly. The personal computer processes the detected coordinates by computing the homography transform and locating the laser hit point.

Hardware Module

The proposed camera on weapon system is implemented on a low cost hardware as described in the following. Two things are considered for implementing the hardware, i.e., low cost and the components are available commercially.

The hardware module consists of the following components:

- Camera: Web-Camera and Raspberry Pi-Camera
- Laser Pointer: Red laser pointer which is usually used for aiming on the air gun
- Single Board Computer: Raspberry Pi module
- Wireless Module: Xbee module
- Battery: Power bank for smart phones

Fortunately, those components and modules could be mounted easily on PS3 pistol which used as the artificial gun. Therefore the camera on weapon system could be assembled easily.

Camera and Laser Pointer

Two types of camera systems are developed. The first one employs the Logitech HD Pro Webcam C920 (Fig. 2a) and the second one employs the Raspberry Pi-Camera module (Fig. 2b). The Logitech Webcam is equipped with USB interface and the popular device driver. The important feature of the camera is the Carl Zeiss lens which is adopted for providing the brighter image. The other consideration is the dimension and physical arrangement which are suitable for mounting on the gun. The Webcam provides a still image with resolution of 4492×3328 pixels (15 megapixel).

The Raspberry Pi-Camera is a camera module designed for connecting to the Raspberry Pi using CSI (Camera Serial Interface) connector. The CSI bus offers the high data rates. The camera sensor has a still image resolution of 2592×1944 pixel (5 megapixel).

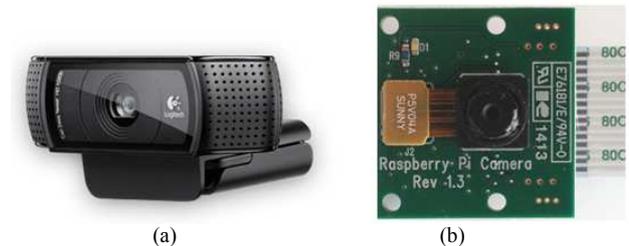


Fig. 2. (a) Logitech Webcam; (b) Raspberry Pi camera module

Many laser pointers are available in the market. The common laser pointer is the one used for presentation purpose. Another type of laser pointer is the one used in the shooting sport or hobby (assembled on the air gun). In this research, we employed the second one. It has the wavelength of 532 nm and the power of 5 mW.

Raspberry Pi

The Raspberry Pi is a credit-card sized computer based on ARM processor. The recent model (Raspberry Pi Model B) has 512 MB RAM and an ARM1176JZF-S 700 MHz processor. There are two USB ports and one Ethernet port. The module is equipped with a RCA video out and HDMI out for interfacing to display monitor, audio out, SDCARD slot, General Purpose Input Output (GPIO) and micro USB power input. The module is suitable to be mounted on the gun.

One USB port is used for connecting to the USB camera, while the another one is used for connecting to the USB Wifi dongle. It is noted here that the wifi module is used for debugging only. The wireless connection between the Raspberry Pi and a personal computer employs the Xbee module as described in the next section.

Wireless Module

One benefit of our proposed camera on weapon system is the wireless connection between the gun and a personal computer. This wireless connection provides the flexible handling of the gun. The shooter is able to hold the gun freely. There is no restriction caused by the cable connection. There are two wireless modules employed in the system, i.e., Xbee module and USB Wifi module. The Xbee module is the main wireless connection used by the system. The USB Wifi module is used for monitoring the Raspberry Pi remotely. While the Xbee module is employed to transfer the data from the gun to the personal computer.

The Xbee Pro Series 2 as illustrated in Fig. 3a is employed. It provides the Zigbee protocol which allows a very simple and reliable wireless communication. The module offers the low power consumption.

The Xbee extension shield as illustrated in Fig. 3b is adopted for attaching the Xbee module to the Raspberry Pi via GPIO pins. The extension shield is equipped with an RGB LED, which is utilized for shooting purpose as explained in the next section.

Software Module

Raspberry Pi requires the Linux based operating system. Fortunately, there are many packages available on Internet. The most popular one is Raspbian, which is adopted in the proposed system. The program is written with C++ language for accessing the camera, the Xbee module and performing the image processing tasks.

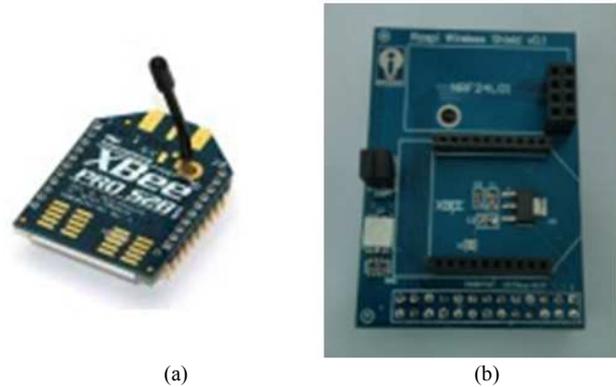


Fig. 3. (a) Xbee Pro Series 2 (b) Xbee extension shield

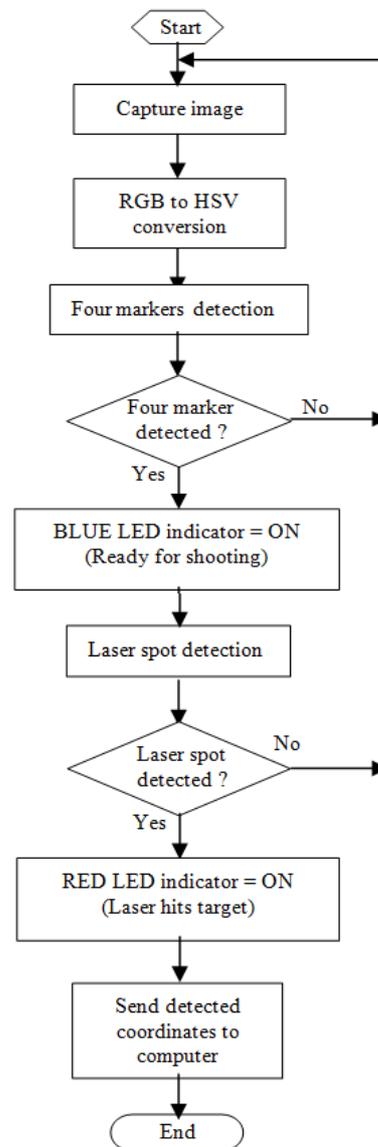


Fig. 4. Shooting algorithm

Figure 4 illustrates the proposed shooting algorithm. It starts by capturing the target image. Then it converts RGB color image to HSV color image. After color conversion, the red color thresholding is applied to detect the markers. The markers are four red solid circles painted on the border of target image. The markers are used to calculate the homography transform for correcting the camera view, due the LCD projection and the camera position on the gun.

When the four markers are detected by the camera, it will inform the shooter that he/she could start to fire the laser on the gun. This information is indicated by turning-on the Blue LED on the gun. If the four markers are not detected, the Blue LED is turned-off, then the camera will capture the image again until the markers are detected.

Once the shooter fires the laser, the algorithm will detect the laser spot on the target using the red color thresholding similar to the marker detection. If it verifies that the laser spot is detected, then the Red LED is turned-on. Finally the coordinates of four markers and laser spot are sent to the personal computer wirelessly. It is worthy to note that by sending the coordinates only, instead of a whole image, the high speed communication is not required. Therefore the wireless communication between the camera on weapon system and the personal computer could be developed easily using the Zigbee protocol.

Marker and Laser Detection

When capturing a rectangular target image using a camera, the captured image might be distorted by the LCD projection or the camera rotation. In our application where the camera is mounted on the weapon, the camera moves along the gun. Therefore the above distortion could not be avoided.

To provide the correct image projection, the homography transform should be computed. Our approach employs four known points for computing the homography matrix as employed in (Widodo *et al.*, 2012; Ladha *et al.*, 2013). Four points are taken from the center coordinates of four red markers as illustrated in Fig. 5. To speed up the process, the homography transform is computed on the personal computer. While the camera on weapon system detects the coordinates of four markers and sends them wirelessly.

In this research, the most popular computer vision library called Open CV is employed to perform the image processing tasks. Fortunately, the OpenCV is also available for the Raspberry Pi running under Raspian operating system.

Detecting the markers involves two steps i.e., the color thresholding and the contour finding. The color thresholding is used to find the binary objects (blobs) according to the certain color criteria. In this case the hue, saturation and value of the HSV color space. If the color of pixels satisfy the criteria, they are painted in the white color, otherwise the pixels are the black color.



Fig. 5. Target and four red markers

After the black and white image is obtained, the contours of detected objects (white objects) are found. To verify that the contours are the red markers, the numbers and the areas of contours are checked. Namely, the numbers of contours should be four and the areas of contours should fall in a certain ranges defined empirically.

To detect the laser spot, the similar red color thresholding is employed. Since the area of laser spot is smaller compared to red marker, the laser spot could be distinguished easily. Further verification is done by calculating the position of laser spot in the image, i.e., the position of laser spot should fall inside the bounding box determined by four red markers.

Monitoring Software

In our proposed system, although the single board computer is attached on the weapon, the shooting monitoring is carried out by a personal computer. This approach has several benefits such as: (a) The coach could monitor or examine the trainee's shooting performance effectively; (b) The shooting hit points could be recorded for further analysis; (c) The GUI application software could be developed attractively on the personal computer; (d) Since data is sent wirelessly, the monitoring computer could be placed far away from the shooting room, e.g., in the coach room.

In our research, the monitoring software is developed to display the detected target and the laser spot and to compute the homography transform. The goal is to find the normalized target image, i.e., projecting the detected markers and laser spot into the reference target image as illustrated in Fig. 6.

The reference target image is an image of the shooting target. The image contains the target picture such as human, animal, circle, etc. and four red markers as illustrated in Fig. 6a. This image is projected by the LCD projector into the target screen.

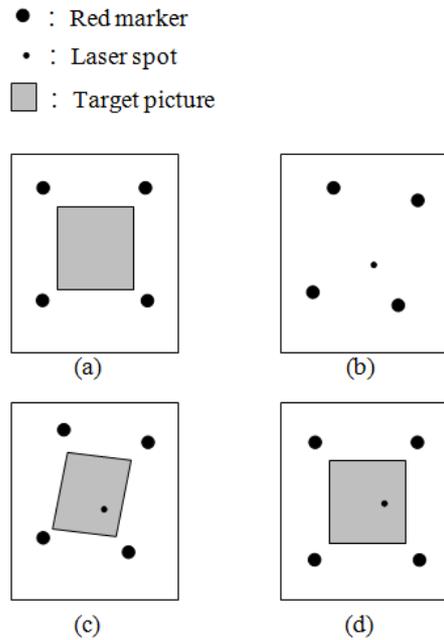


Fig. 6. (a) Reference target image (b) Detected marker and laser spot (c) Reconstructed target image (d) Normalized target image

As described previously, the camera on weapon system does not send the whole captured image to the personal computer. Instead, only the detected coordinates of the four markers and the laser spot are sent. The coordinates are received by the personal computer and displayed as illustrated in Fig. 6b. To reconstruct the detected target image, the homography transform is employed. Let x is the coordinates of point in the reference target image. The coordinates of point in the reconstructed image (x') is expressed as:

$$x' = Hx \quad (1)$$

where, H is the homography matrix, which is solved using four points of the markers. All points in the reference image are mapped to the new image using Equation 1. The resulted image called as the reconstructed target image is illustrated in Fig. 6c, where the target picture is not a rectangular, but it appears as a parallelogram conforms with the detected markers.

Using the reconstructed target image, we could display the laser spot on the target picture as the real captured image, although the whole captured image does not sent to the personal computer. However to locate the laser spot or score the hit point accurately, the laser spot should be mapped onto the reference target image. Fortunately, it could be done easily using Equation 1 by inverting the homography matrix H . The resulted image is called the normalized target image as illustrated in Fig. 6d.

Experimental Results

To verify the proposed system, several experiments are conducted. Two types of camera systems are examined during experiments. The first type employs the Logitech Web-Camera, while the second one employs the Raspberry Pi-Camera as described previously. During the experiments, the Raspberry Pi is operated remotely through a Putty software using the Wifi connection. This method allows the image captured by the Raspberry Pi to be sent to the personal computer for comparison purposes.

The objectives of experiments are to measure the performance of proposed techniques, which comprise of the following three factors. First is the success rate of shooting detection. It is measured by firing the weapon several times until ten shots are detected. The Success Rate (SR) is defined as:

$$SR = \frac{10}{\text{numbers of shooting}} \times 100\% \quad (2)$$

The shooting is considered as detected when the Blue LED and the Red LED are turned-on.

Second is the target detection error, which is defined as the discrepancy between the reference target image and the normalized target image. It measures how accurate does the system detect the target and perform the projection into the reference target. The Target detection Error (TE) is computed using the following Equations 3-7:

$$TE_{TL} = 0.5 \times \left(\frac{|TLXn - TLXr|}{TLXr} + \frac{|TLYn - TLYr|}{TLYr} \right) \quad (3)$$

$$TE_{TR} = 0.5 \times \left(\frac{|TRXn - TRXr|}{TRXr} + \frac{|TRYn - TRYr|}{TRYr} \right) \quad (4)$$

$$TE_{BL} = 0.5 \times \left(\frac{|BLXn - BLXr|}{BLXr} + \frac{|BLYn - BLYr|}{BLYr} \right) \quad (5)$$

$$TE_{BR} = 0.5 \times \left(\frac{|BRXn - BRXr|}{BRXr} + \frac{|BRYn - BRYr|}{BRYr} \right) \quad (6)$$

$$TE = 0.25 \times (TE_{TL} + TE_{TR} + TE_{BL} + TE_{BR}) \quad (7)$$

where, TE_{TL} , TE_{TR} , TE_{BL} , TE_{BR} are the target detection errors calculated from top-left, top-right, bottom-left, bottom-right markers respectively; $TLX(Y)n$, $TRX(Y)n$, $BLX(Y)n$, $BRX(Y)n$ are x(y)-coordinates of top-left, top-right, bottom-left, bottom-right markers of the normalized target image respectively; and $TLX(Y)r$, $TRX(Y)r$, $BLX(Y)r$, $BRX(Y)r$ are x(y)-coordinates of

top-left, top-right, bottom-left, bottom-right markers of the the reference target image respectively.

Third is the deviation of the camera displacement. Since the camera and the laser pointer are installed on the weapon in a fixed position, the position of detected laser spot on the camera's coordinate should be constant. This arrangement could be used to detect the laser spot directly. Although in our approach we don't use this approach, the measurement of camera displacement is conducted to verify the laser spot detection as follow. In the experiments, we record the coordinates of detected laser spots using the techniques described before, for all weapon shots. Then the mean and the standard deviation are calculated for all collecting data. By assuming that there are no displacement between the camera and the laser pointer during the experiments, the standard deviation could be used to measure the accuracy of laser detection.

Ten reference target images as illustrated in Fig. 7 are tested during the experiments. For every target, two shooting scenario are examined. In the first scenario, the shooter holds the weapon in a fixed position and shots a fixed hit point on the target several times. While in the second scenario, the shooters moves the weapon to shot the random hit points on the target.

Detection Using the Web-Camera

Figure 8 illustrates the shooting detection results using the Web-Camera for fixed hit point and random hit points scenarios. The average Succes Rates (SRs) are given in Table 1. The average succes rate is calculated using Equation 2 for all ten reference target images. From the table, it is obtained that SR of the fixed hit point scenario is better than the one of the random hit points scenario. In the random hit points scenario, when the shooter moves the weapon to shot the different hit point, the shooter must wait until the Blue LED is ON before firing. It ensures that the four markers are being detected. Thus the lower SR of the random hit points scenario is caused by the laser spot detection error, not by the marker detection. From the observations, this error is affected by the different position of the laser spot on the target images that yields the different color of laser spot. For instance, the color of laser spot captured by the camera varies when the laser hits the black, blue, white colors of the background.

The experimental results of target detection errors using the Web-Camera for fixed hit point scenario and random hit points scenario are illustrated in Fig. 9 and 10 respectively. In the figures, TE-TL, TE-TR, TE-BL, TE-BR and TE are the detection errors as defined by Equation 3-7. The target detection errors for both scenarios are plotted again in a graph as illustrated in Fig. 11. While the average target detection errors for both scenario are given in Table 2.

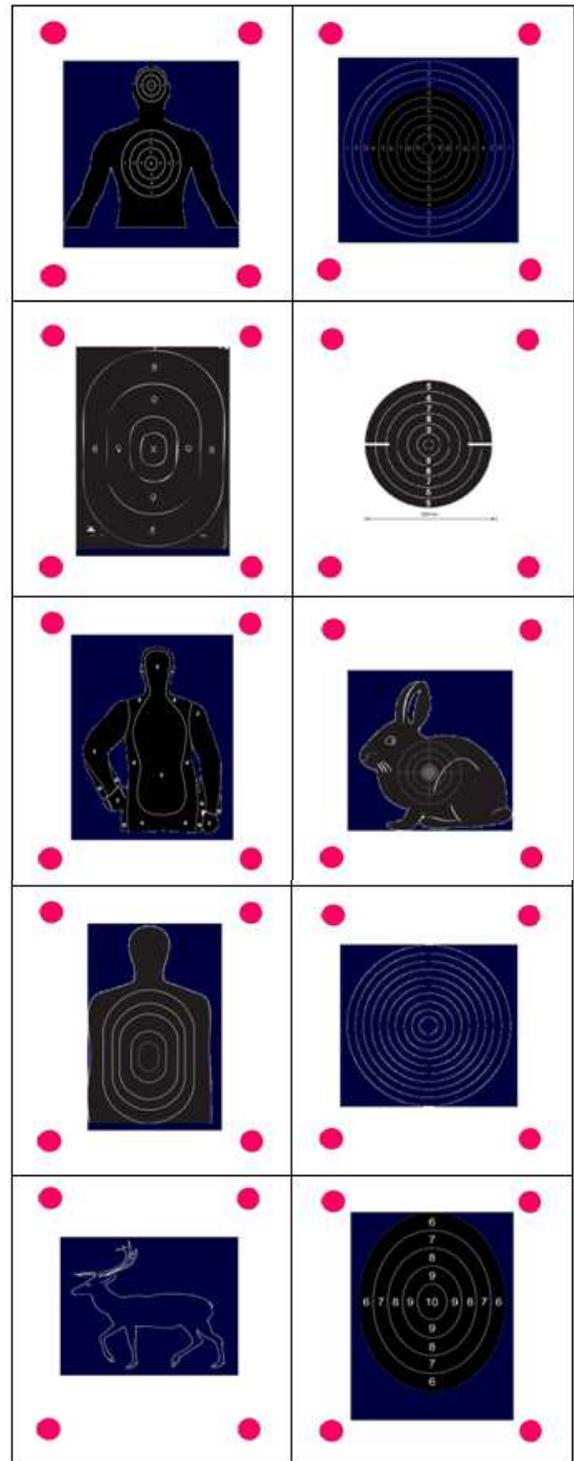


Fig. 7. Reference target images

Table 1. Average SR using the Web-Camera

Scenario	Average SR (Succes Rate)
Fixed hit point	95.12%
Random hit points	85.71%

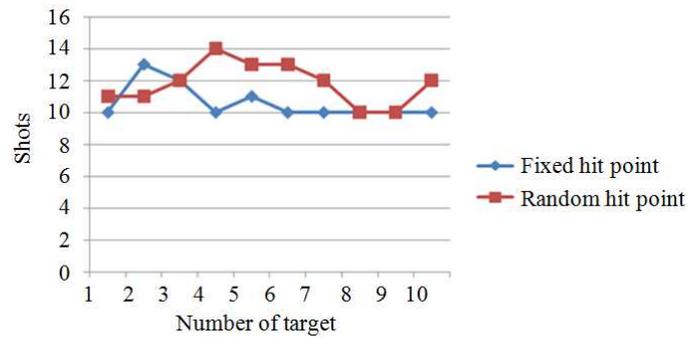


Fig. 8. Shooting detection results using the Web-Camera

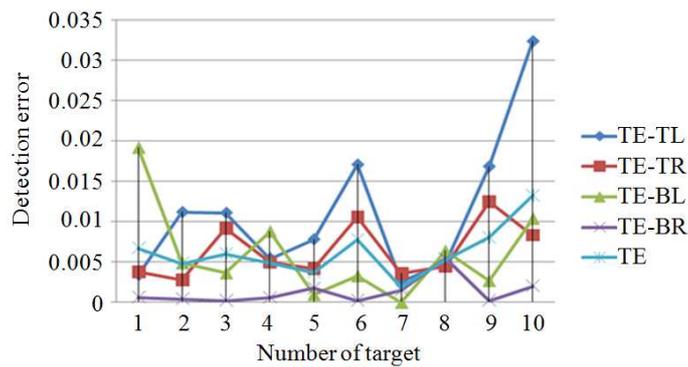


Fig. 9. Target detection errors using the Web-Camera for fixed hit point scenario

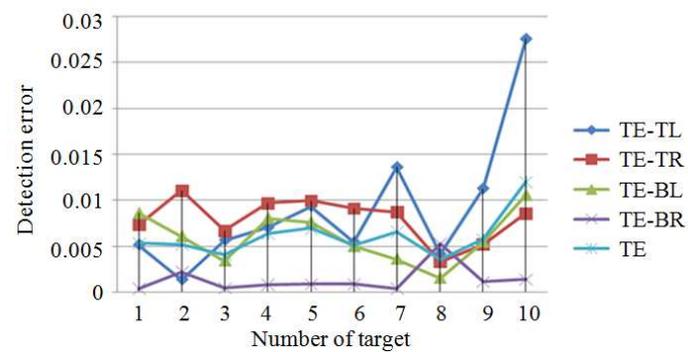


Fig. 10. Target detection errors using the Web-Camera for random hit points scenario

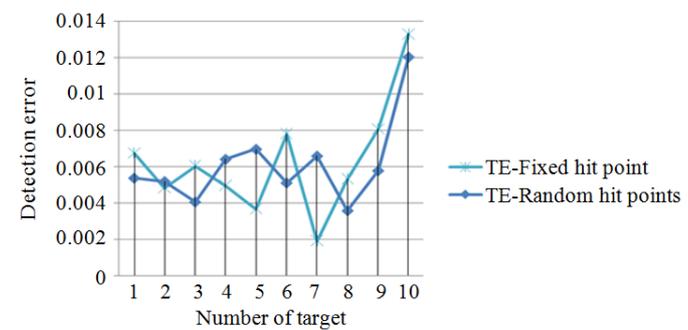


Fig. 11. Target detection errors using the Web-Camera for fixed hit point and random hit points scenarios

From Fig. 11 and Table 2, it is clearly shown that the target detection errors for both scenarios are almost the same. Compared to the previous laser detection error, where the random hit points scenario yields the greater error, the target detection error is not affected by the shooting scenarios. It could be realized from the fact that in the target detection method, both scenarios use the fixed markers. Further it could be stated that the target detection method achieves a very small error about 0.6%.

Figure 12 and 13 illustrate the x-coordinate and y-coordinate of the detected laser spot using the Web-Camera respectively. From the figures, it is obtained that both scenarios yield the small deviation on detecting the laser spot. By observing the mean and standard deviation given in Table 3, our proposed Web-Camera on the weapon sytem shows a good accuracy on detecting the laser spot. Further, the results verify that the arrangement of the camera and the laser pointer provides a very small displacement error.

Detection Using the Raspberry Pi-Camera

The shooting detection results using the Raspberry Pi-Camera are illustrated in Fig. 14, while the average succes rates are given in Table 4. These results are similar to the system using the Web-Camera, where the average success rate of the fixed hit point scenario is better than the one of the random hit points. The average success rates for both systems are almost the same.

The target detection errors of the system using the Raspberry Pi-Camera are illustrated in Fig. 15, 16 and 17. Both scenarios show a small error as illustrated in Fig. 17 clearly. The average target detection errors are also very small (about 0.8%, Table 5). It is almost similar to the one using the Web-Camera.

Figure 18 and 19 illustrate the x-coordinate and y-coordinate of the detected laser spot using the Raspberry Pi-Camera respectively. The coordinates are different from the Web-Camera system. It is

caused by the different installations of the camera on the weapons. Observing the figures, both systems have a small deviation below than one pixel.

The means and the standard deviations given in Table 6 could be interpreted as the accuracy of the laser spot detection. Similar to the Web-Camera system, the Raspberry Pi-Camera system shows a good accuracy.

Table 2. Average TE using the Web-Camera

Scenario	Average Target detection Error (TE)
Fixed hit point	0.006271
Random hit points	0.006121

Table 3. Mean and standard deviation of detected laser spot using the Web-Camera

Scenario	Mean	Standard deviation
Fixed hit point: X-coord	170.76	0.363
Random hit points: X-coord	170.76	0.195
Fixed hit point: Y-coord	137.03	0.283
Random hit points: Y-coord	137.11	0.137

Table 4. Average SR using the Raspberry Pi-Camera

Scenario	Average Succes Rate (SR)
Fixed hit point	94.21%
Random hit points	89.40%

Table 5. Average TE using the Raspberry Pi-Camera

Scenario	Average Target detection Error (TE)
Fixed hit point	0.00731
Random hit points	0.00806

Table 6. Mean and standard deviation of detected laser spot using the Raspberry Pi-Camera

Scenario	Mean	Standard deviation
Fixed hit point: X-coord	173.00	0.000
Random hit points: X-coord	173.05	0.098
Fixed hit point: Y-coord	104.68	0.529
Random hit points: Y-coord	104.73	0.339

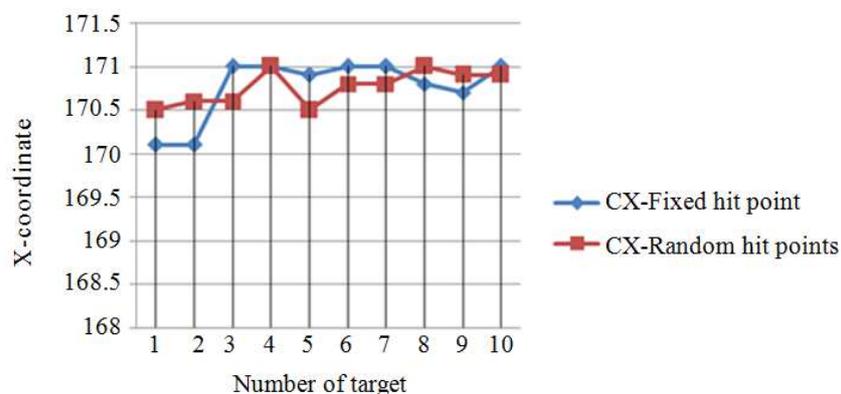


Fig. 12. X-coordinate of detected laser spot using the Web-Camera

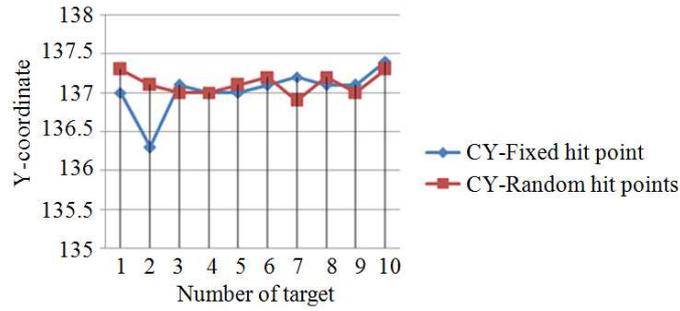


Fig. 13. Y-coordinate of detected laser spot using the Web-Camera

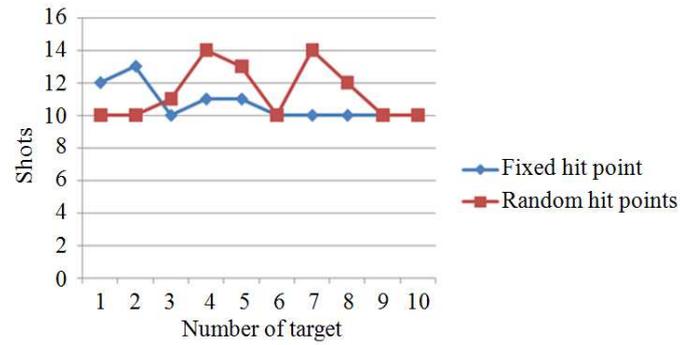


Fig. 14. Shooting detection results using the Raspberry Pi-Camera

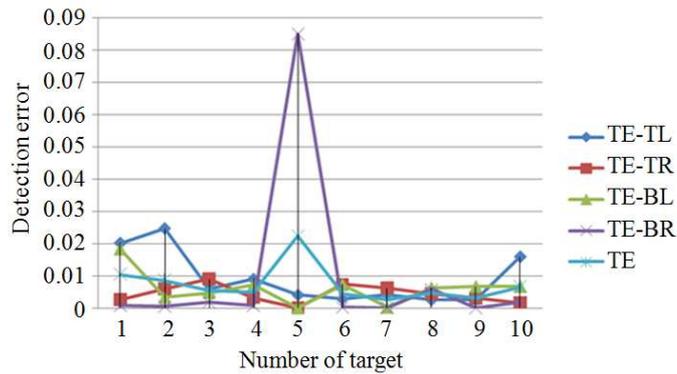


Fig. 15. Target detection errors using the Raspberry Pi-Camera for fixed hit point scenario

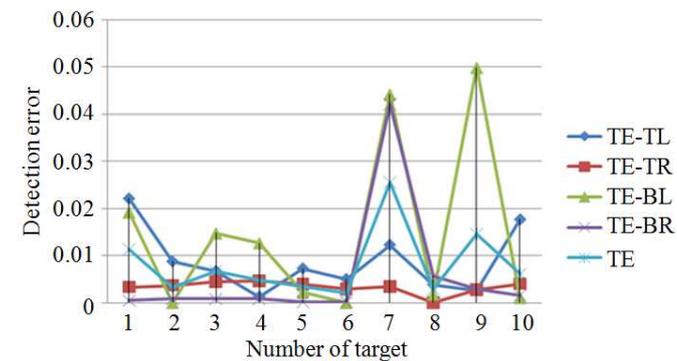


Fig. 16. Target detection errors using the Raspberry Pi-Camera for random hit points scenario

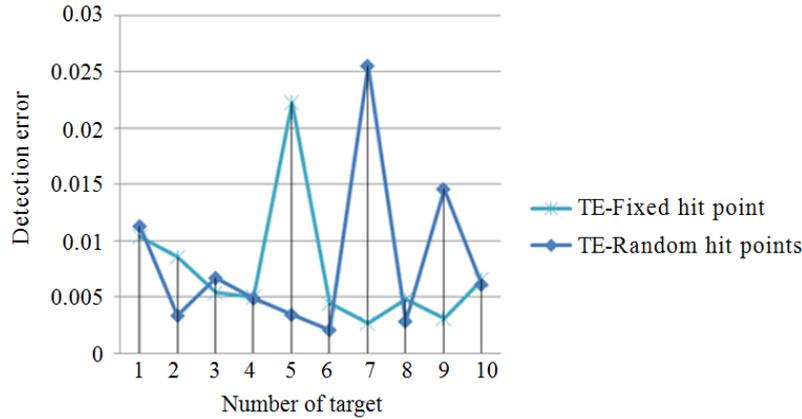


Fig. 17. Target detection errors using the Raspberry Pi-Camera for fixed hit point and random hit points scenario

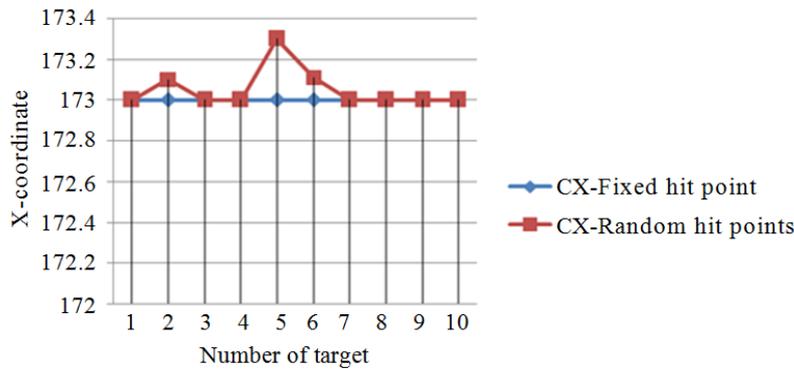


Fig. 18. X-coordinate of detected laser spot using the Raspberry Pi-Camera

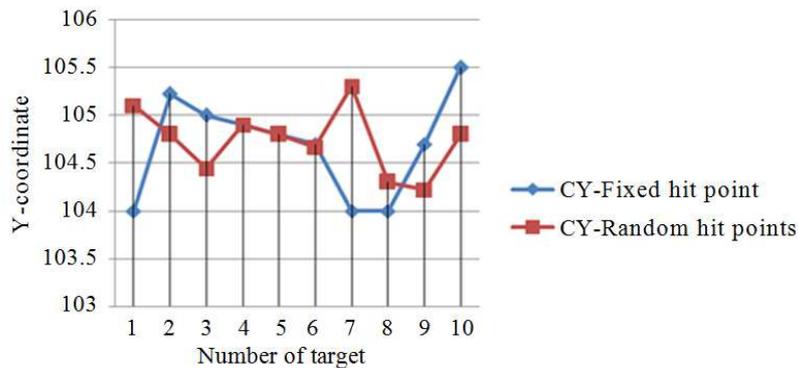


Fig. 19. Y-coordinate of detected laser spot using the Raspberry Pi-Camera

Results of Reconstructed Target Images

As discussed previously, our proposed technique is able to reconstruct the captured image by utilizing the detected coordinates of the markers and laser spot. Figure 20 and 21 illustrate some reconstructed target images of the Web-Camera and the Raspberry Pi-Camera systems respectively. The figures on the left columns are the reconstructed target images, while the ones on the right columns are the captured images by

the Raspberry Pi. The figures show that the reconstructed target images are almost similar to the captured images, in this case the detected positions of the markers and the laser spot.

To speed up the process, both camera systems use the resolution of 320×240 pixels. Using this resolution, the frame rates of 17 fps and 27 fps are achieved for the Web-Camera and the Raspberry Pi-Camera systems respectively.

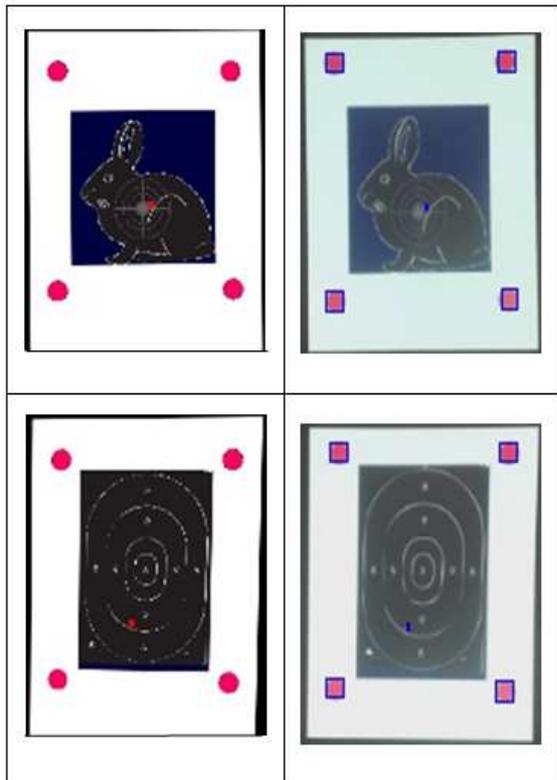


Fig. 20. Reconstructed target images of the Web-Camera system

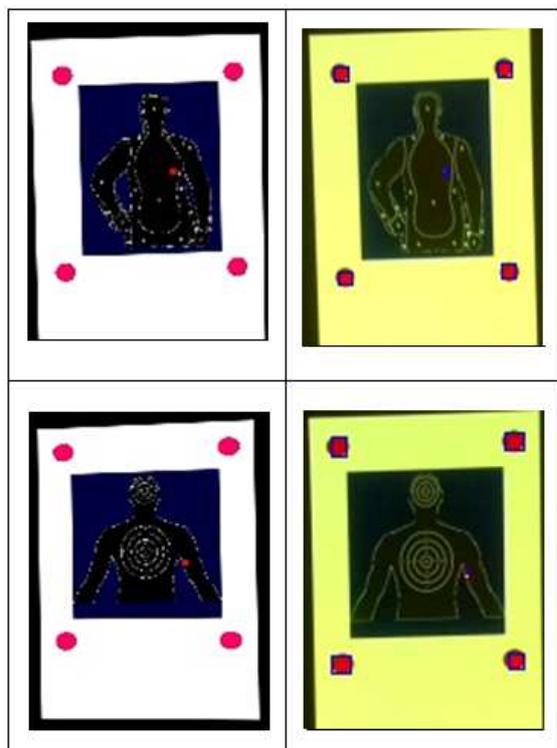


Fig. 21. Reconstructed target images of the Raspberry Pi-Camera system

Conclusion

The camera on weapon system was developed for detecting the laser spot on the shooting target. The Raspberry Pi module is employed as the embedded system to perform the image processing tasks. To avoid the cable connection between the gun and the personal computer, the wireless communication using Zigbee protocol is employed.

The experimental results show that the proposed system is effective to detect the shooting target and the laser spot with the high success rate, the low target detection error and the small displacement between the camera and the laser pointer.

In future, we will extend the research for handling the complex shooting scenario. Further the shooting analyzer software will be developed.

Acknowledgement

Authors would like to thank to National Institute of Technology, Malang, Indonesia for supporting this work. Students of Department of Electrical Engineering for helping us during the research.

Funding Information

This work is funded 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.

Author’s Contributions

Aryanto Soetedjo: Leader of the project, wrote the main text of manuscript, designed the whole system.

Ali Mahmudi: Participated in software design, collected data.

Mochammad Ibrahim Ashari: Participated in electronic design, conducted experiment.

Yusuf Ismail Nakhoda: Participated in mechanical design, edited manuscript.

Ethics

Authors confirm that this manuscript is original and has not been published elsewhere.

References

- Soetedjo, A., M.I. Ashari, A. Mahmudi and Y.I. Nakhoda, 2013. Camera-based shooting simulator using color thresholding techniques. Proceedings of 3rd International Conference on Instrumentation Control and Automation, Aug. 28-30, IEEE Xplore Press, Ungasan, pp: 207-211.
 DOI: 10.1109/ICA.2013.6734073

- Soetedjo, A., M.I. Ashari, A. Mahmudi and Y.I. Nakhoda, 2014. Implementation of sensor on the gun system using embedded camera for shooting training. Proceedings of 2nd International Conference on Technology, Informatics, Management, Engineering and Environment, Aug. 19-21, IEEE Xplore Press, Bandung, pp: 69-74. DOI: 10.1109/TIME-E.2014.7011594
- Liang, H.W. and B. Kong, 2006. A shooting training and instructing system based on image analysis. Proceedings of the IEEE Conference on Information Acquisition, Aug. 20-23, IEEE Xplore Press, Shandong, pp: 961-966. DOI: 10.1109/ICIA.2006.305866
- Lapointe, J.F. and G. Godin, 2005. On-screen laser spot detection for large display interaction. Proceedings of IEEE International Workshop on Haptic Audio Visual Environments and their Applications, Oct. 1-2, IEEE Xplore Press, Ottawa, pp: 72-76. DOI: 10.1109/HAVE.2005.1545655
- Kim, N.W., S.J. Lee, B.G. Lee and J.J. Lee, 2007. Vision based laser pointer interaction for flexible screens. Proceedings of the 12th International Conference on Human-computer Interaction: Interaction Platforms and Techniques, Jul. 22-27, Beijing, pp: 845-853. DOI: 10.1007/978-3-540-73107-8_93
- Widodo, R.B., W. Chen and T. Matsumaru, 2012. Laser spotlight detection and interpretation of its movement behavior in laser pointer interface. Proceedings of IEEE/SICE International Symposium on System Integration, Dec. 16-18, IEEE Xplore Press, Fukuoka, pp: 780-785. DOI: 10.1109/SII.2012.6427378
- Ladha, S., K.S. Miles and S. Chandran, 2013. Multi-user natural interaction with sensor on activity. Proceedings of 1st IEEE Workshop on User-Centered Computer Vision, Jan. 15-17, IEEE Xplore press, Tampa, FL, pp: 25-30. DOI: 10.1109/UCCV.2013.6530804
- Ladha, S., S. Chandran and K.T. Miles, 2010. Vision assisted safety enhanced shooting range simulator. Proceedings of National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics, (IPG' 10), Jaipur.