The Development of Image Capturing System Applied in Shooting Simulator using LabVIEW

SHU-HUANG SUN, CHIH-WEI CHIANG, BONILLA GRAMAJO JOSE MANUEL
Department of Mechanical Engineering
Kun Shan University
No.195, Kunda Rd., YongKang Dist. Tainan City
TAIWAN
sh-sun@mail.ksu.edu.tw

Abstract: - An electronically controlled M16 rifle simulator, integrated a control box and an infrared laser device had been successfully integrated to a shooting virtual reality as an all-in-one solution in this study. This system is shown to provide a flexible, cost-effective and safe environment. It is an effective educational tool used in defense industry to assist soldiers in shooting skill improvement, or provide players an immersive shooting experience in the virtual reality game industry. It is developed and tested for understanding its advantages and limitations. Interface image capturing and processing response time analysis has been performed. In the results of this study, the response time is one-tenth of a second. It is satisfactory in a real application.

1 Introduction
1.1 Background description
Firearms training simulators as shown in Figure 1 have been developed to cover different scope for training. They also were applied in different weapons and functions. The following are a list of projects related to firearms simulators worldwide [1] :

(1) Noptel marksmanship and analysis system (Finland) [2]
(2) Indra SAC Lite (Spain) [3]
(3) ZEN SATS® SL (India) [4]
(4) Laser Shot PSATS-MIL Portable Small Arms Training Simulator (United States) [5]
(5) FATS® 100LE firearms training simulator (United States) [6]
(6) MILES IWS2 (Multiple Integrated Laser Engagement System) (United States) [7]
(7) VirTra Small Arms Training Simulator (United

Fig 1. Finished virtual shooting facility built

Virtual reality technology not only enables interactions of users with educational-orientated virtual environments but also allows direct information exchanges among users. Through a computer, the technology can present a three-dimensional virtual environment completed with spatial, sensory, and audio simulation based on a real or imaginary scenario. It raises interest and motivation of trainees to improve knowledge and real experience transfer. It can provide a highly immersive, interactive, and imaginative training context that in real world would be too dangerous, too expensive, or simply impossible to access. Warwick, Gray, and Roberts [9] gathered a number of papers on the use of VR in engineering in their book. The successful applications of VR as well as new attempts can also be found in a very broad range of other domains. Military sector also is an important field of VR application for combat and fight training [10, 11]. Hughes etc. [12] transformed core Mixed Reality (MR) technology and methods into diverse urban terrain applications. Bowen etc. [13] described the application of virtual environment technology on a novel and complex military checkpoint. Battles are simulated and fought, using simulations to explore various maneuvers and possible outcomes as training and preparation for real confrontations [14-16]. A simulation system called Red Flag Measurement and Debriefing System (RMDS) [17] developed by

US air force is applied to reconstruct a mock air battle.

1.1.1 Common features found in virtual shooting simulator systems

Overall shooting simulation systems worldwide share similar features and techniques. The main objective is to determine whether the target was hit or missed by comparing the shooting coordinates and the target position. The most common techniques can be summarized as follows:

(1) Point of impact: fire directly at the target position to get the point of impact such as the VirTra Small Arms Training Simulator.

(2) Image type: Integrate a CCD camera to detect the laser beam projected position of the target surface image when the shots are fired and capture images to obtain the point of impact like the Laser Shot or Noptel systems.

(3) Supersonic acoustic beam sensors: Use laser or ultrasonic sensors arrangement to obtain the position of the point of impact.

(4) Array induction: Photo sensors are arranged on the target surface to detect the simulated firing bullets from the laser beam to spot and capture signals to determine the point of impact.

(5) Light pen: The screen display is the target surface and bullets are replaced by a light pen. These systems use the light pen to detect the position of the light beam at the moment of firing to receive the location of the point of impact.

(6) Mechanical: Shoot from a firearm mounted on a rod.While firing, the difference between the current aiming location and the simulated origin position is detected.By this way, the impact point locationis obtained.

Based onthe comparison results above, the use of image shooting training simulators can have many advantages due to the low CCD camera cost, working easily on a two-dimensional coordinate system and reducing main program development and system maintenance. Therefore, this technique is selected for this study.

This study applies image processing techniques to a firearms training simulator. Itused a CCD camera as a sensor to capture images and then employed image processing to obtain the laser coordinate location.The necessary analysis is then executedand this point location is transferred to the main virtual scene software for impact point display/animation. In order to avoid the visibility of the laser diode emission and to favor impact simulation authenticity, an infrared laser diode will be used in this study so operators will not be able to see the laser light spot formed on the target surface/screen. A camera with wavelength range 700nm and above is used since this wavelength range is different from the visibility of human eyes'.The infrared laser beam then can't be seen by operators but still can be observed by the camera. The aiming point detection design for this study will be based on this principle.

1.2 Motivation

The goal of this study is to integrate two standalone projects. The first project developed an electronically controlled M16 rifle and integrated a control box for controlling the weapon's different operation modes and an infrared laser device as targeting aiming. The second project developed a virtual shooting simulator with three scene types. Currently, the shooting simulator works only with mouse events, the firearm trigger is simulated with "on mouse click" event. So, one of the main tasks is to replace the "on mouse click" event by an interface which is capable of acquiring the firearm shooting coordinates from an external application.But before attempting to develop the simulator blocks logic, the interface type and tools availabilities have to be defined to check the compatibility with Virtools software, the virtual scenes developing tool used in this study.

1.2.1 Laser firearms and fire control box

The developed shooting simulators include rifles, machine guns, mortars and other hardware as shown in Figure 2, such as T91 (Figure2-a), 65K2 (Figure2-b), 50 machine guns (Figure2-c), 40 machine guns (Figure2-d), T74 guns (Figure2-e), T75 machine guns (Figure2-f), 60 mortars (Figure2-g), 81 mortars (Figure2-h), 120 mortars (Figure2-i), etc. With the total weight exactly the same as the real firearms and same physical characteristics, combined with a virtual environment including visual and sound effects,the simulations results can give trainees analmost 100% experience just like to operator a real firearm. The shooting simulation tests are shown in Figure 3.To reduce the load on the host computer, each firearm will be controlled with a separate control box. The main control device is an 8051 microcontroller.

The control box PLC (RIO-47100 with Ethernet/RS232 interface) connects to the Ethernet port of the host computer through Galil Suite/Galil Toolssoftware (Galil Suite/GalilTools Software is used in the controller for communication utilities) and shows a complete list of I/O ports along with its current logical value 0 or 1. The control box will inform the host computer all sensor signals, operation status, and errors through a PLC so the

main program can read all data output from the firearm in real time.The control box can be connected also to the air compressor to provide gas when firing instantly to simulate realistically the real firing shot recoil.
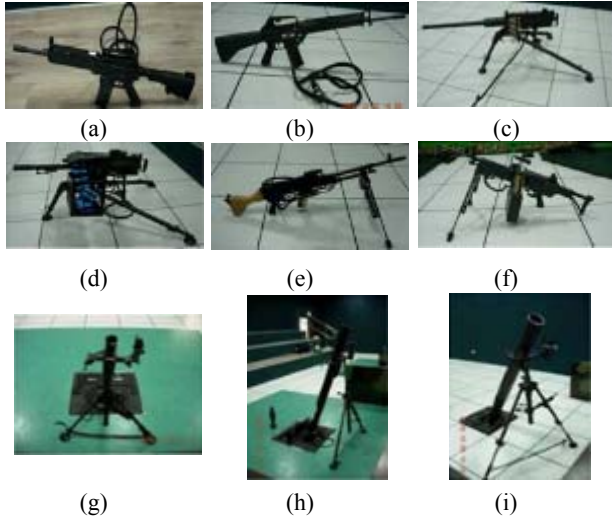

(a)　　　　　　(b)　　　　　　(c)

(d)　　　　　　(e)　　　　　　(f)

(g)　　　　　　(h)　　　　　　(i)

Fig 2. Hardware count for shooting simulators

(a)　　　　　　　　　　(b)
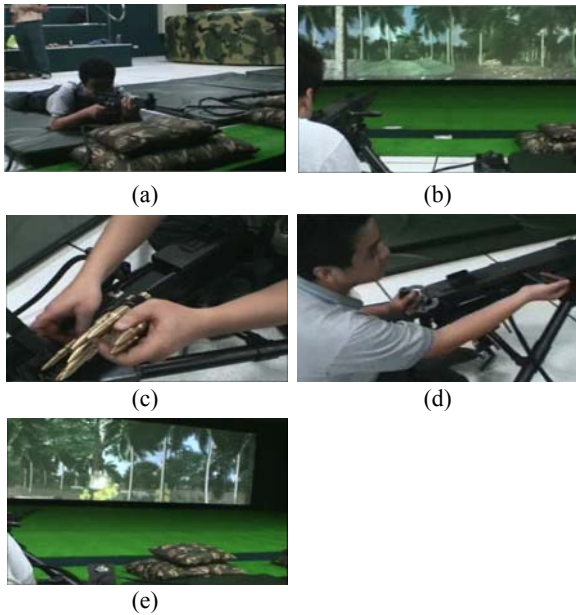
(c)　　　　　　　　　　(d)

(e)

Fig 3. Shooting simulation tests

1.2.2 Virtual scene

Owing to the lack of domestic software for virtual scenes and technical aspect related, it is necessary to invest in this research field so that the shooting simulators technology developed by the hardware manufacturer can be integrated gradually to build an integrated all-in-one virtual shooting simulator system. Therefore,a project had developed to complete the construction of a virtual scene and interactive human-machine interface systems as

shown in Figure 4.The virtual scene is divided into four parts in total, including:

(1) Single target shooting scene/calibration scene
(2) Multiple target shooting scenes
(3) In motion vehicle shooting scene:
　　(3-1) Jeep mobile shooting scene
　　(3-2) Helicopters mobile shooting scene

The item (1) and (2) referred to static scenes and item (3-1) and (3-2) referred to dynamic scenes. They are all developed by 3DVia Virtools- Dassault Systèmes version 4.0.
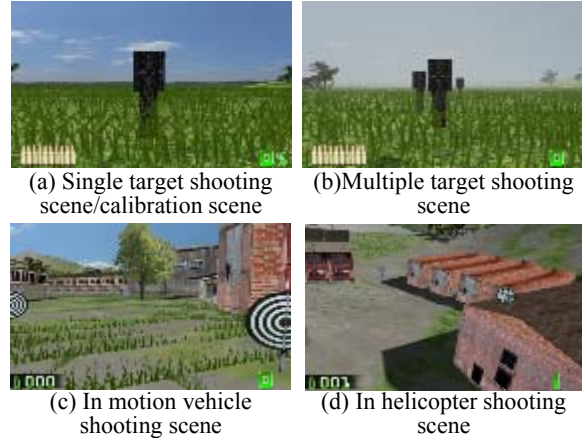

(a) Single target shooting　　(b)Multiple target shooting
scene/calibration scene　　　　　　scene

(c) In motion vehicle　　　(d) In helicopter shooting
shooting scene　　　　　　　　scene

Fig 4. Completed shooting simulation virtual scenes

1.2.3 Image capturing part

From allmajor project components, the only remaining component is the integration of the firearm hardware including control box and virtual scene. When this component is completed, the system can turn into a fully integrated virtual reality shooting simulator system. The image capturing component is the main part of the scope of this study. The detailed system architecture will be shown and described in the next section.

2 Research Method
2.1 System architecture

Shooting simulators are the future of military training; the importance is to increase the technical skill of the army in a fast, effective and convenient way with the possibility of extending the equipment to a variety of weapons and even vehicles. Figure 5 presents the shooting simulator architecture design of this study. Generally shooting simulators can be roughly divided into five parts, which include:

(1) Laser firearm & recoil mechanism: The actual firearm has the exact same weight and operating procedures including safety, trigger, and recoil resembling to a real firearm. There are no real bullets. Instead, an IR laser device is attached to the front of the cannon. This laser emits an

infrared beam to virtual scenery screen. The firearm reproduces recoil in the same way of a real firearm so that the operator can have a realistic experience.

(2) Fire control system: Some parts of the system will be integrated into the main program, but a large amount of processing and computation are released from the main program for the reason of avoiding sluggish response. The fire control system can be built into a separate fire control component. A general fire control component can be divided into fully automatic fire, burst fire, single shot, etc by sending a logical 1 or 0 to the specified control box PLC outputs. Its operating principle is based on the compressed air transferred via control box to simulate the feel and recoil of a real firearm on firing.
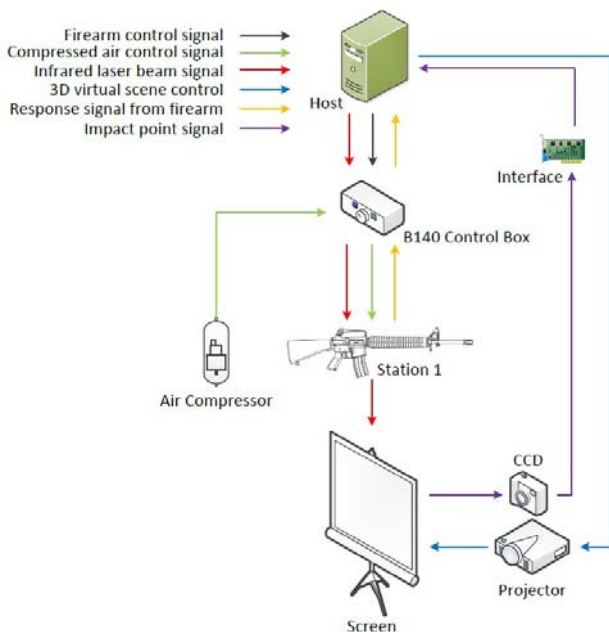

Fig 5.Shooting simulator system architecture

(3) Virtual scenery: Shooting simulation software uses different graphics virtual scenes software such as 3Dmax, 3DVia Virtools or EON, to render. Using any of the previous or similar rendering software can generate a virtual scene projected onto a projection screen so that the operator can experience a more realistic scenario depending on the rendered scene.

(4) Image capturing system: The image capture system contains a CCD camera, a video capturing card and related software. The principle of operation is the following: for every shot fired from the firearm, the CCD captures an image; each image is then transferred to the main program (LabVIEW) and processed. The main program decomposes the image and applies image processing techniques to calculate the shot

coordinates of the laser beam in real time. Then, after the (X,Y) coordinates being extracted, the values are sent to the virtual simulator to determine if the shot fired hit the target or not. At the same time, the main program produces visual and sound effects to improve the realistic experiences.

(5) Host computer: It is the responsibility for all scenarios, virtual scenery, fire control, image capturing and score calculation. If the amounts of processing consume resources excessively, sharing the load between two or more computers would be possible also; one computer is responsible for virtual scenery and 3D rendering and the other one is responsible for fire control, image capturing, score calculation and after training reports.

As Figure 5 shown, the shooting simulator system should consist of these five parts. Based on the recoil shooting simulation, control box, laser device and virtual scene, this study focuses on image capturing techniques and main interface program. After the completion of this study, the system will be able to join all five parts and become an integrated shooting simulator system.

2.2 Operation mode

When operating, the projected virtual scene will be first shown on the projection screen. The firearm barrel will be equipped with the infrared laser and emit a signal point to the projection screen. Next, for every firearm trigger executed by users, the main program (LabVIEW) will detect the output status change of the PLC and trigger the CCD camera to capture an image for processing and finding the laser shot coordinate. Also, at the same time, the main program sends a signal to virtual scene software (Virtools) to activate a sound effect and a virtual effect on the projection screen. After shot coordinate detection, the coordinate values will be stored into a text file for the virtual scene to read it. If the coordinate shot hits a target, the corresponding animation will be displayed at the point of impact to complete the process.

For the image capture system, a video capture card, CCD software, and drivers are also installed on the computer. The LabVIEW program has to be able to discover and operate camera functions for image capturing. These drivers and software are also going to be installed on the main computer.

After the LabVIEW program is completed and successfully tested, the program can be exported to an executable file for easier transportation and installation on other computers.

2.3 Interface development

This study mainly employed LabVIEW as a development tool to enable hardware and software integration. It can be broken down into three phases. The first stage develops a laser shot detection program, the second stage captures the laser shot image by CCD device and the third phase is the integration. They are described in detail as following.

(1) Development of shot coordinate detection program: A LabVIEW VI program will be written to read image files (BMP or JPEG). The image file will be decomposed as a 2D data array, the array has to be iterated and each color pixel value is to be compared to find which the represented point of impact is. After the program reads the images and iterates it, each pixel value is extracted to identify the shot point. If remarkable values are found, the program will return the corresponding coordinate value.

(2) Using video capture device: the LabVIEW program must be able to control the image capturing and camera triggering. So whenever the firearm trigger is pressed, the program has to trigger the CCD camera to capture an image for processing. Image files can be captured and stored as BMP image files. Following the program flow, the image must be sent to the shot coordinate detection program for further processing.

(3) Shooting image capture integration: A trigger action will be implemented to generate information to a shooting event [18]. After the first two phases are carried out, the next phase of the study will work to integrate the hardware (firearm) and software (main LabVIEW program). By the advantage of LabVIEW hardware integration, the firearm PLC controller can be controlled and read to obtain current I/O ports logical status (0,1). A LabVIEW block is written to accomplish three functions, (i) connect the control box PLC, (ii) set the laser input to "1", and (iii) set the program to wait for the trigger event to occur. When the firearm trigger event occurs, the LabVIEW program must read the change in the PLC output to detect the value change and execute the image capturing block to capture and process the image data for shot coordinate detection on this value change.

2.4 Basic shooting simulator program flow chart

Figure 6 shows the basic flow diagram suggested for the development of the interface between the M16 firearm control box and the 3D virtual scene developed using Virtools 4.0.

3 Experiments and Results

3.1 Laser point acquisition

Before the laser point acquisition program is developed, a more extensive image analysis is required to determine if the correct threshold range is suitable for the laser detection algorithm. It is very different from our imagination that the shot point of scenery screen emitted by the infrared laser is a red spot on the projection screen. Furthermore, the same laser shot in different locations of images possess variable RGB pixel values as Figure 7 shown (also note that the change in contrast/background changes the RGB values).
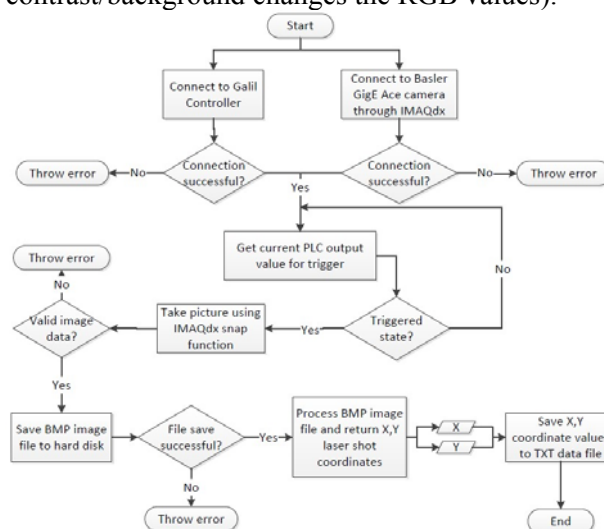

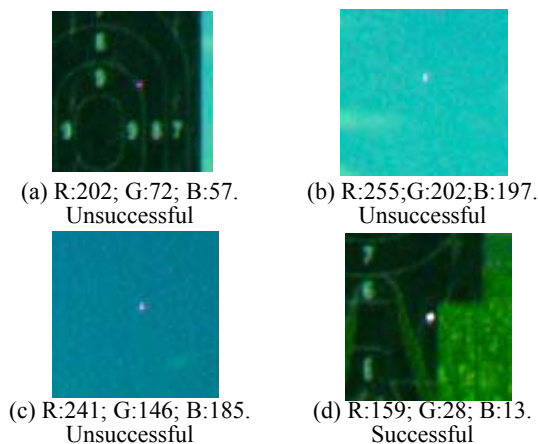Fig 6. Basic virtual shooting simulator program flow chart

(a) R:202; G:72; B:57. Unsuccessful

(b) R:255;G:202;B:197. Unsuccessful

(c) R:241; G:146; B:185. Unsuccessful

(d) R:159; G:28; B:13. Successful

Fig 7. Image processing tests with 24 bit sample images

3.1.1 Analyzing RGB layers sample images data

By examining further samples images, the next step is to analyze RGB layers data. For this case, we took a sample image and decomposed it to extract all

pixel values information. Using Matlab, a program was written to read image files and plot in XY coordinates corresponding to width and height, extract each pixel data value and then plot a three-dimensional diagram as shown in Figure 8.

From analyzing layers composition, pixel values in RBG and Gray color map are observed that the region with most pixel density is between 50 and 100 across all layers. At this point, it is a difficult task to iterate and identify the laser point location. Because all pixel values are close, the laser point detection program fails to find accurately point location.
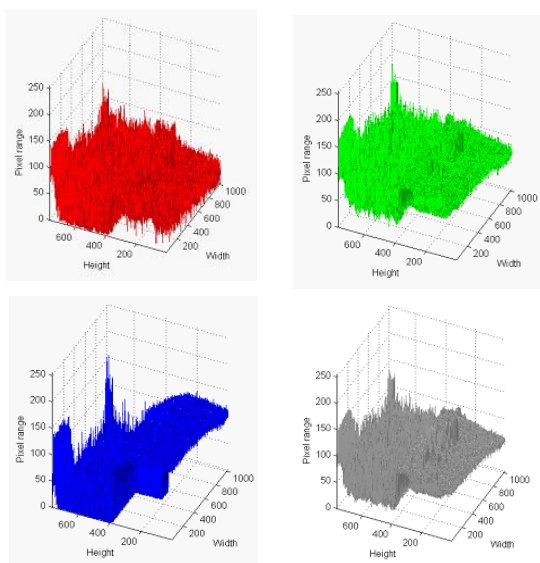

Fig 8. Sample image layer decomposition, RGB and gray scale pixel values shown

3.1.2 Filtering visible spectrum

We had already the experience that in real experiments laser point does not show constant pixel values across samples. Reviewing component specifications and system requirements, we defined the following statements:

(1) The laser device integrated into the firearm is an infrared laser; therefore its wavelength is above 700nm (above visible light spectrum).

(2) Because CCD sensor is more sensitive in the Near-Infrared (NIR), cameras partially detect near infrared spectrum, above 700nm up to 1000nm

(3) Basler cameras cover a spectral range from 400 to 1000nm. Human eyes roughly detect about 400 to 800nm [19].

For this application, it is not required to analyze the complete image data from acquired images. Our focus of interest is to find the pixel values in the NIR section of the spectrum. Even though CCD sensor is able to capture values in the NIR, those

values are transformed to be shown into the RBG color map. Pixel values are represented between 0 to 255 intensity values. In other words, by filtering the visible image spectrum, it is possible to integrate an IR filter with the CCD camera to exclude all visible light and allow only the infrared section of the spectrum to go through.

For Basler cameras, the blue channel is set for the range of 400 to 500nm, the green is for 500 to 600nm and the red is for more than 600nm. But, the NIR opens all three channels for higher than 700nm [19]. To avoid capturing colors in the visible spectrum, an IR long wave pass filter is required. By using IR cut-on filter, all visible light except the IR section of the spectrum is excluded. Figure 9(a) shows these devices used in this study to capture an image file. Except a CCD and a lens, a filter is also installed in the front of the lens. Figure 9(b) shows a sample image taken with IR long wave pass filter. From this image, we can easily find the location of the laser point.

(a) CCD + lens + filter   (b) Sample image

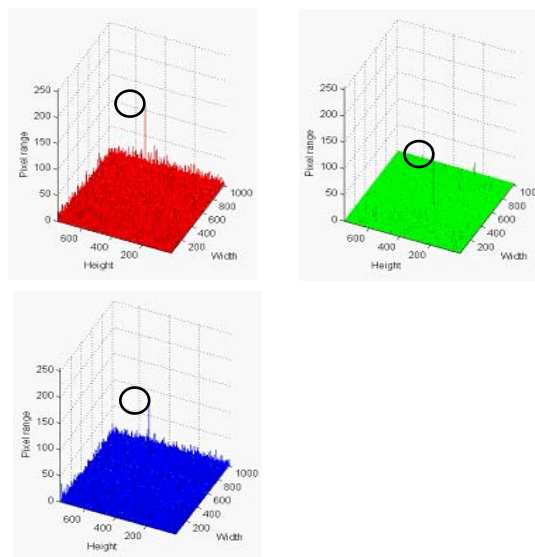Fig 9. Image capturing device and sample image taken with IR cut-on long wave pass filter

Fig 10. Sample image with IR cut-on long wave pass filter

3.2 Analyzing sample image with IR cut-on long wave pass filter RGB layer decomposition

As shown in Figure 10, IR filter provides a more suitable solution for filtering all visible image data

and lowering noise to the minimum. Contrary to sample images taken without IR filter, it is noticed that Red, Green, and Blue pixel values all get a high-intensity scale, while noise values are lower than 50 on the intensity scale. Using IR-cut filter improves the margin for the laser shot detection algorithm and achieves a satisfactory result.

Also, for the improvement of laser point calculation and detection algorithm, it is recommended to apply a threshold method for image segmentation [20]. By applying a threshold, the image can be divided into 2 sections: (i) for all pixel values below 150 can be set up to be 0 on the intensity scale and (ii) pixel values above 150 can be set up to be 255. This would result in an even more suitable way for the laser point detection program to successfully locate the laser shot coordinates. LabVIEW's built-in threshold function is recommended for improving the algorithm.

3.3 An Experimental infrared laser point acquisition with IR cut-on long wave pass filter

3.3.1 F-number and working distance calibration

Best image results will appear with f-number (focal ratio or relative aperture) = 4, 5.6 and 8. With lower f-numbers the image is blurred by aberrations (spherical aberrations, astigmatism, field curvature, coma, and distortion), and for high f-numbers it is blurred by diffraction [21]. Many tests are achieved and described as follows:

(1) When setting up with f-number= 16, no color pixel values are found as shown in the 1st row of Figure 11.

(2) When setting up with f-number= 8, laser point is only slightly visible on the blue layer color map as shown in the 2nd row ofFigure 11.

(3) When setting up with f-number= 4, laser point is clearly more visible in the red and blue layer color map as shown in the 3rd row ofFigure 11.

(4) When setting up with f-number= 2.8, the images are shown in the 4th row of Figure 11. In these images laser point is visible in the red and blue layer color maps, green layer is slightly presented. Color white starts to predominate in the center of the laser point.

(5) When setting up with f-number= 2, the images are shown in the 5th row ofFigure 11. In these images, laser point is visible in the red, green and blue layer color maps. As shown in the figure, the white color becomes more dominant in the middle section due to color saturation.

(6) When setting up with f-number= 1.4, the images are shown in the 6th row ofFigure 11. In these

images, laser point is visible in all layer color maps. As shown in the figure, white color is dominant in the middle section due to color saturation.

On these tests, laser point light is clearly visible after IR filtering is applied. Because laser device wavelength is 850nm and CCD sensor can capture imagery in the NIR, the white color indicates saturation in color intensity.Infrared color, therefore, can be seen as white. This poses a new point of view for the laser point detection program; it would be possible to search pixel values greater than 200.
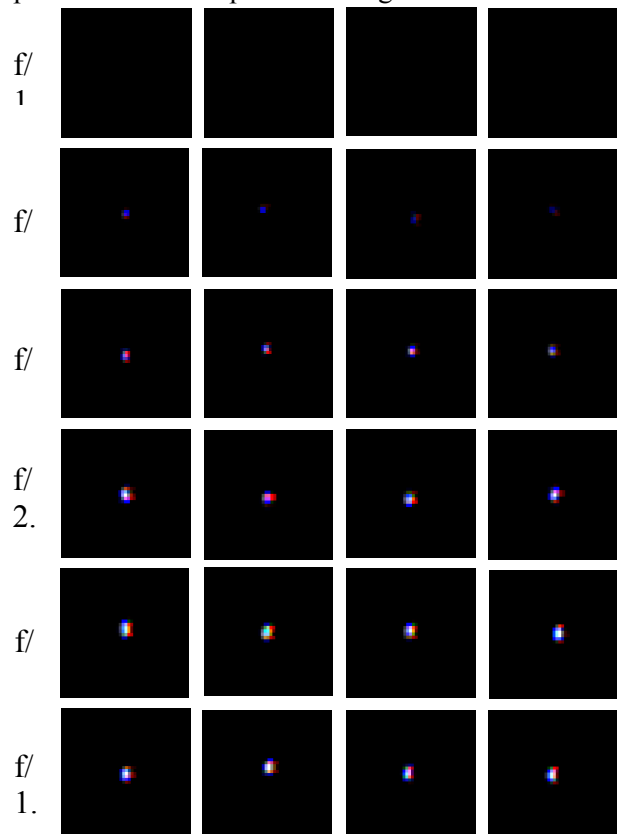

Fig 11. F-number calibration

3.3.2 Overexposure problem

There are some exceptions when acquiring images and moving the firearm rapidly while triggering. Overexposure can result in blurry laser point capturing due to the rapid firearm movement as shown in Figure 12.
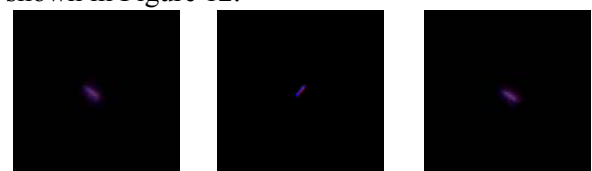

Fig 12. Overexposure, blurry laser point (Laser shot detection program unable to determine a shot coordinate location)

For minimizing abnormal laser shape, it is recommended to work with lower exposure and

aperture values. An aperture setting between f-number= 2.8to 4 with an exposure time of 15000μs yields satisfactory results for this study.

3.3.3Perspective distortion correction [21]

At this point, the camera position has to be centered and perpendicular to the projection screen for capturing images without perspective distortion. For an ease use, setup and accuracyimprovement, perspective distortion correction are integrated into the main program interface. The calibration
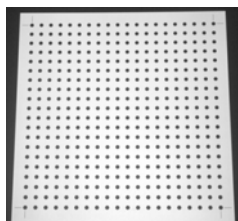
template,as shown in Figure 13, contains a grid of dots where the system can learn the calibration template and distortion model to perform an image distortion correction process when acquiring images on firearm trigger [22].

Fig 13. Image calibration template with perspective distortion

After the system learns the calibration template (this would be a process at system startup), the distortion correction model can be applied to every image acquired by the CCD camera on trigger [22]. The effect of distortion correction can be seen from Figure 14. Figure 14(a) is the original image captured by the camera and Figure 14(b) is the undistorted image after perspective correction applied.
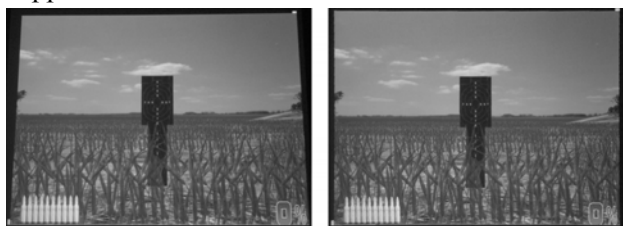
(a)   (b)

Fig 14. The effect of distortion correction

3.3.4Response time

After the complete system is setup and integrated, a response time analysis was required in this study due to a noticeable sluggish response when triggering, capturing and processing images for the main program. For a better understanding of where the problem arises, time intervals of the key points were put inside the main program to locate the bottleneck of the process.

3.3.4.1Response time in image acquisition and image processing

As the sluggish response was noticeable when triggering, acquiring and processing images for laser point detection, it was simpler to concentrate the efforts in the image capturing process and image processing part. In theexperimental test, the time response fluctuates around 0.35 seconds and occasionally up to 0.55 seconds (more than half a second). This, of course, is going to have a direct impact on the user experience as every shot triggered and processed will be out of time.It is not acceptable for the user to undergoa fast and smooth shooting experience.

Because at this point, we don't know exactly which of the suspected processes is the one causing more time delay problems, it is necessary to take a second set of samples and analyze the block diagram execution time.For this case, timestamp elements were added in the main program to measure the response time for the two different processes under inspection: 1) Image capturing process and 2) Image processing (laser shot detection). Figure15 shows the time response for the second set of samples. It is observable the most time-consuming process at this point is the image processing response time. A more detailed inspection of the laser shot detection algorithm is necessary to determine whether the cause of the time delay is software misconfiguration, algorithm failure, hardware limits or other underlying processes.
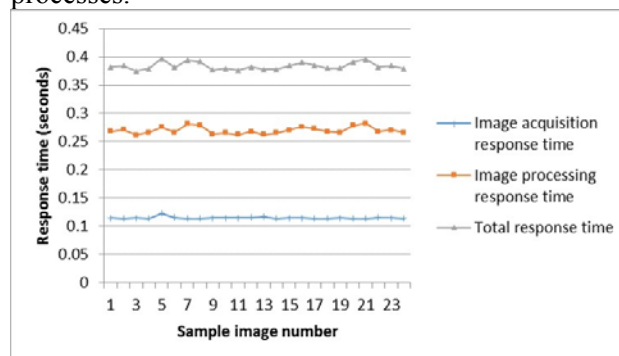
Fig 15. Response time measured for image acquisition and image processing blocks using 24 bit (RGB) images

3.3.4.224-bit images pixel iteration: a time consumption process

Because in every picture has todecompose and iterate each RGB value to compare each pixel value against pixel value intensity of interest.This process can be time-consuming since the laser detection program iterates each pixel coordinate and extracts RGB values for each pixel and then compares. A

solution is to modify image capturing format.Setting up CCD camera to capture grayscale monochrome images (U8) will drastically reduce the response time of iteration process. The laser shot detection program would have to iterate only through 1 layer data information and search for pixels value intensity greater than the value set.

3.3.4.3 Analyzing 8 bit (monochrome) image pixel values intensity

8-bit monochrome images maintain a large threshold for easily detecting laser shot coordinates. It is recommended that a 100-pixel value intensity or above searches for a shot location.Using 8-bit monochrome images, image processing response time drastically reduces to 0.03 seconds per sample. Image acquisition also reduces from 0.125 seconds to 0.09 seconds.

3.3.4.4 Response time improvement using 8-bit monochrome images

Figure16 shows the response time comparison between 24 bit (RGB) and 8-bit monochrome images processing. Also, the improvement percentage between 24 bit and 8 bit average response time comparison are shown in Table 1. From this table, using 8-bit monochrome format improves response time up to 70% than using 24 bit RGB image format.
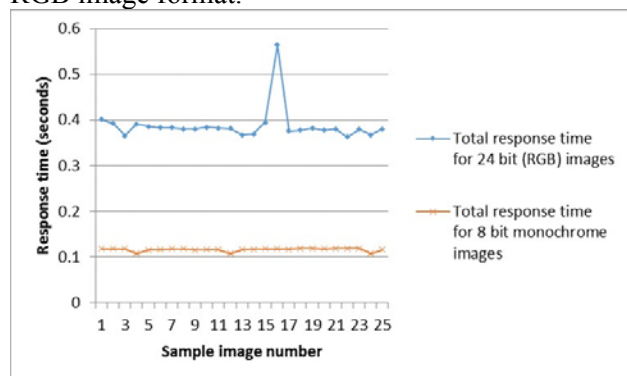

Fig 16. Response time comparison between 24 bit (RGB) and 8 bit monochrome images processing

Table 1. 24 bit and 8 bit average response time comparison

Average response & improvement	
24 bit (RGB)	8 bit (Monochrome)
0.38576 seconds	0.11616 seconds
Improvement	69.89%

Further analysis in this part is required for a better understanding of the cause. For the image capturing process, the Snap VI, a built-in function of IMAQdx is used to acquire images from the CCD camera. However, it is recommended to replace the current Snap function to IMAQdx Grab VI which is clearly defined as: "Use the grab VI for high-speed image acquisition" according to National Instruments IMAQdx VI Reference help.

3.3.4.5 High-speed image capturing and perspective distortion response time

Replacing the low-speed Snap function with the high-speed Grab function reduces image acquisition time to zero since images are already being buffered to the main program. Figure17 shows the response time analysis for the replacement of Snap VI function with the high-speed Grab VI function. This decreases the system response time from the best previous result of 0.09 seconds to zero. Nevertheless, as seen in Figure 17, the image processing time increases from the best previous result of 0.03 seconds to 0.14 seconds owing to the perspective distortion response time. Although the image processing response time has increased to one-tenth of a second, the result is considered satisfactory due to the mayor algorithm improvement achieved. At this point, the total system response time approximates to one-tenth of a second. For single shot mode, the system operation is satisfactory.But to integrate burst and fully automatic mode, the total system response time has to be improved to decrease time delay to a minimum.
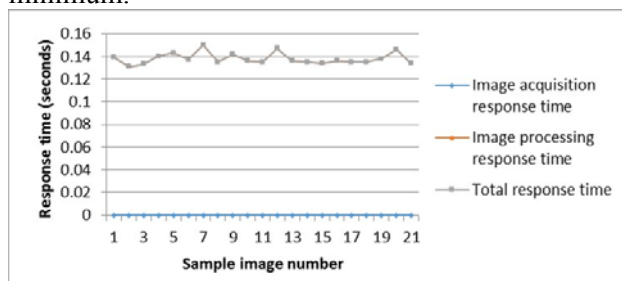

Fig 17. High-speed Grab VI and perspective distortion correction response time

3.3.5 System accuracy estimate

A total of 52 tested shots were performed for data collection analysis. Table 2 shows some of these data. Figure 18 shows the error estimation process description. Table 2 also shows an average absolute percentage error for x = 0.11% and y = 0.13%. The system resolution setup of this study is 1024 x 768 pixels. Transforming the percentage values to pixel values yields a system accuracy of ±1 pixel for x and y coordinates.

In Figure 18, a shot animation is drawn on x = 510 and y = 324. Laser detection program returned x = 510.321 and y = 324.227 for the corresponding real coordinates.
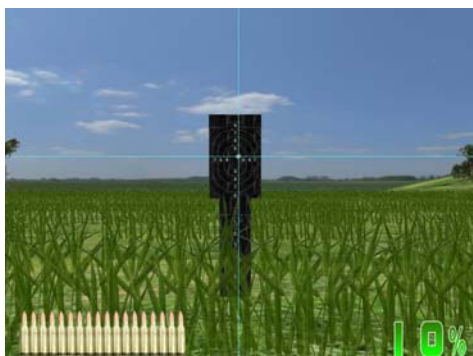
Fig 18. Sample image for accuracy estimation. Image size is 1024x768 pixels

Table 1. Mean absolute percentage error to estimate system accuracy

No	%x	%y	x(t)	y(t)	x(rt)	y(rt)	% err x	% err y
1	50	39	511.0	298.5	510	298	0.20	0.17
2	53	44	548.2	339.4	548	339	0.04	0.13
3	50	45	508.7	346.7	508	346	0.13	0.22
4	47	48	477.4	367.0	477	367	0.09	0.01
5	53	48	538.6	371.5	538	371	0.10	0.14
6	48	39	489.7	296.1	489	296	0.14	0.04
7	47	43	478.2	333.5	478	333	0.05	0.15
8	54	38	553.0	289.0	552	289	0.18	0.01
9	50	53	509.3	408.0	509	408	0.07	0.00
10	50	38	514.9	289.1	514	289	0.18	0.04
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
45	54	53	555.5	403.2	555	403	0.09	0.06
46	53	46	540.4	355.3	540	355	0.08	0.10
47	46	50	475.9	385.9	475	385	0.19	0.24
48	49	45	503.6	344.6	503	344	0.13	0.16
49	47	44	481.9	337.7	481	337	0.18	0.22
50	50	51	511.1	395.3	511	395	0.02	0.08
51	49	41	506.1	314.5	506	315	0.03	0.15
52	51	48	517.4	370.5	517	370	0.08	0.13
Mean absolute percentage error: x: 0.11%, y: 0.13%								

3.3.6 Main program GUI

For the usability and operation convenience, the main program GUI is developed as Figure 19 shows. The general user interface items are described in detail as follows:

(1) PLC RIO47100 IP address: A text control applied to input the IP address to connect to PLC through GalilTools COM library.

(2) Response value: Control box trigger output status. If response status data type is not numeric, the main program automatically parses the response field to obtain a boolean false or true or expressed as numeric values "0" or "1".

(3) Camera selection: A selection box lists all cameras connected to the host computer through any ports.

(4) Trigger status: A graphical indicator shows the current trigger status. The indicator light will turn on with the trigger.

(5) Acquired image: An image placeholder shows the total image captured by the CCD camera.

(6) The region of interest (ROI): An area of interest is delimited. It is going to be analyzed to obtain laser point location. This area is used to get an image subset.

(7) Image size (Width x Height): The image size is a camera dependent parameter; this value is used for calculating the laser shot location as percentage values.

(8) Stop VI button: Stop the current VI.

(9) Save images control: It is a special debug feature. If enabled, images are stored in the current VI's location under the ./images folder.

(10) X,Y coordinate indicator: Text indicators to display the laser coordinate values found by the Sub VI of getting shot coordinates.

(11) Error messages log: Error messages placeholder.

(12) ROI border size: An image placeholder that features for moving and resizing the current ROI to adjust camera's image in place.

(13) Laser status: A red indicator to show laser on/off status.

(14) Virtual scene resolution: Numeric inputs for determining the current virtual scene resolution. The default resolution is 1024x768 pixels.
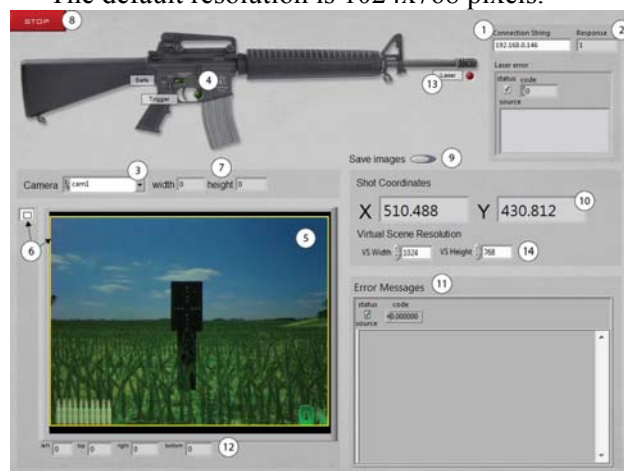

Fig 19. Main program General User Interface

3.3.7 Experimental setup

Figure 20 and Figure 21 shows the experimental setup. The main hardware components are the host computer, M16 rifle, control box, CCD camera, air compressor, projector and projection screen. In this setup, the CCD camera has been placed at ground level with a tripod, but in the ideal setup, the CCD camera would be placed on top. This arrangement would be recommended also for better system portability.

As shown in Figure 21, the distance from the projection screen to the user is independent of the

system working distance (projector + CCD camera to projection screen). This figure is also a demonstration of the usage of this achieved system. A detailed operation demonstration video can be referred as the Online Resource "Shooting Simulation.mp4" shown. (https://drive.google.com/file/d/1NIbEmnu4ic6Djf7f_MAjBAFX-oRsuBq7/view?usp=sharing)

4 Conclusions and Outlook

The two initial projects have been successfully integrated as an all-in-one solution using LabVIEW as the main program interface. Further, the system efficiency has been improved. It has been analyzed and special attention has been paid to image capturing and image processing time responses. High response times/delays have been analyzed and improved with different approaches and setups.

Time response analysis shows that setting up camera format to 8-bit monochrome drastically decreases image processing response time due to single layer iteration. Thus, combining 8-bit monochrome format plus real-time image acquisition lowered the response time considerably.
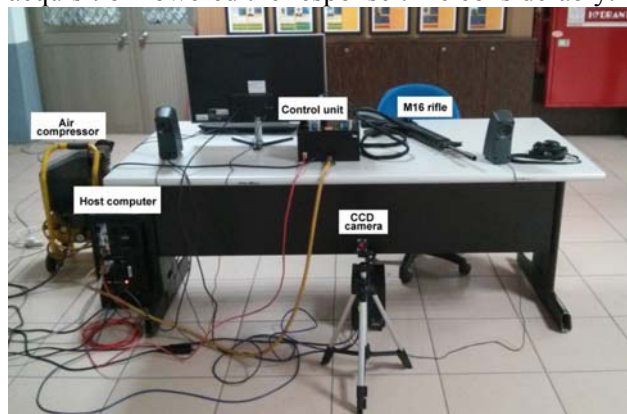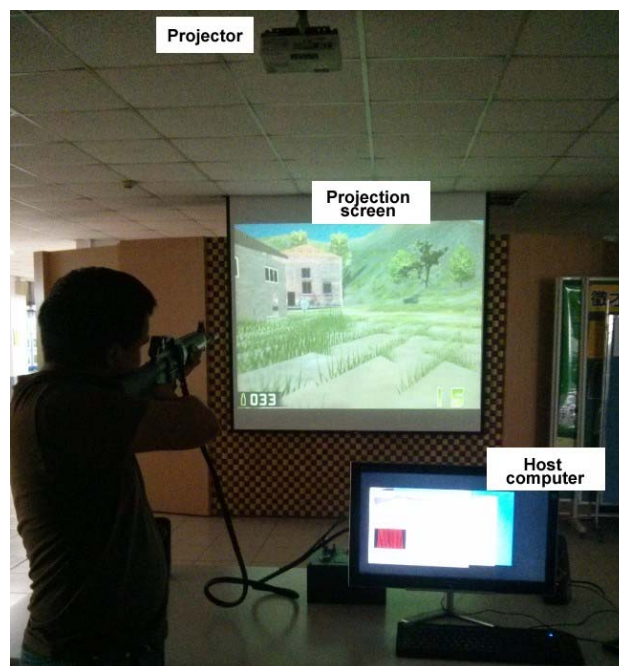

Fig 20. Experimental setup rear view

Fig 21. Experimental setup front view

With the low response time achieved, an algorithm improvement for image acquisition and processing has been also possible. Even perspective distortion integration has increased the image processing time, the result is still considered satisfactory as using for the single shot mode. From the system accuracy estimation, the errors in x and y coordinates are only 0.11% and 0.13%. It proves apparently the reliability of this system. Furthermore, thanks to the establishment of main program GUI, the complicated integration of hardware and software had been simplified greatly.

LabVIEW is a powerful high-level programming language but lacks flexibility for some specific tasks, especially with calculations and array iterations. An interesting development would be to integrate the system using a more flexible, fast and equally powerful programming language such as C++ or C# as the main program interface to decrease response time to a minimum and integrate more features in a more common use.

For future works, the virtual scene can be extended/modified/upgraded using any 3D scene render software. The system could be modified and tested for multiple player integrations. But in that case, the current data exchange process (.TXT data file storage) would have to be modified and further time response analysis would also be required to make the integration possible.

The virtual scene could be upgraded to integrate a ballistic model for adjusting shot locations variations due to distance and wind speed.

After-training reports could be integrated into the current virtual scene. An upgrade could store

recruits username at training startup and store all their training data such as shots fired, accuracy, shots missed, hit percentage up to training time and improvement rate.

References:

[1] Chen CC.*Implement a small arms simulator and training system*. Master Thesis, Department of Electrical and Electronic Engineering, Chung Cheng Institute of Technology, National Defense University, Taiwan, 2000. (Chinese Thesis)

[2] Noptel. *Training Systems* - Shooting Skills Training, http://www.noptel.fi/eng/nts/index.php.(2014, accessed 28 March 2014).

[3] Indra. SAC Lite,http://www.indracompany.com/sites/default/files/sac_0.pdf(2012, accessed 27 March 2014).

[4] Zen Technologies Limited.*Firearms Training Simulator - Zend Small Arms Training Simulator SL (Zen SATS® SL)*,http://www.zentechnologies.com/zen-small-firearms-training-simulator.php. (2014, accessed 15 February 2014).

[5] Laser Shot.*Portable Small Arms Training Simulator*,https://www.lasershot.com/government-military/marksmanship-systems/portable-small-arms-training-simulator. (2014, accessed 16 February 2014).

[6] Meggit.*FATS® 100LE Firearms Training Simulator, Military Training Solutions*,https://meggitttrainingsystems.com/simulation-training/live-fire-simulation/fats-100le. (2013, accessed 9 June 2013).

[7] Cubic Defense Applications.*Multiple Integrated Laser Engagement System*,https://www.cubic.com/Defense-Applications/Training-Systems/Multiple-Integrated-Laser-Engagement-System. (2014, accessed 6 April 2014).

[8] VirTra.*Overview - Training Products*,http://www.virtra.com/overview-mil/. (2013, accessed 11 August 2013).

[9] Warwick K, Gray J and Roberts D. *Virtual reality in engineering*, The Institution of Electrical Engineers, 1993. (ISBN: 0852968035)

[10] United States Congress, Office of Technology Assessment. *Virtual reality and technologies for combat simulation: backgroundpaper*, OTA-BP-ISS-136. US Government Printing Office,WashingtonDC, 1994. (Available online at: http://ota.fas.org/reports/9444.pdf)

[11] Rolfe JM and Staples KJ. *Flight simulation*, Cambridge: CambridgeUniversity Press, 1985. (ISBN: 0521357519)

[12] Hughes CE, Stapleton CB, Hughes DE, et al. Mixed reality in education, entertainment, and training. *IEEE Comput. Graphics Appl*. 2005,Vol.25, No.6, pp.24-30.

[13] Bowen Loftin R, Scerbo MW, McKenzie F D,et al. Training in peacekeeping operations using virtual environments. *IEEE Comput. Graphics Appl*.2004,Vol.24, No.4, pp.18-21.

[14] Calvin J, Dickens A, Gaines R, et al. The SIMNET virtual world architecture. *Proceedings of IEEE Virtual Reality Annual International Symposium*, Seattle, WA, USA, September 18-22 1993, pp.450-455.

[15] Macedonia MR, Zyda MJ, Pratt DR, et al. NPSNET: A networked software architecture for large-scale virtualenvironments.*Presence-Teleop. Virt*. 1994,Vol.3, No.4, pp.265-287.

[16] Stansfield S. Application of VR to nuclear safeguards, *JointESRADA/INMM Workshop on Science and Modern Technologiesfor Safeguards*. (Available online at: https://www.osti.gov/scitech/servlets/purl/1547)

[17] Gardner MT and Amburn P. Simulation-based remote debriefing for Red Flag missions. IEEE *Comput. Graphics Appl*. 1997,Vol.17, No.5, pp.30-39.

[18] Ossa DAH, Medina SAO, Rodríguez CF, et al. Immersive Simulator for Fluvial Combat Training, *Advances in Visual Computing. ISVC 2008. Lecture Notes in Computer Science*.(ed Bebis G. et al.), 2008; 5358:1018-1027. (DOI: https://doi.org/10.1007/978-3-540-89639-5_97)

[19] Basler AG. *Optics Recommendation*,http://www.baslerweb.com/. (2011, accessed 28 October 2013).

[20] Lin KC, Tsai MC. Image feedback path tracking control using an uncalibrated CCD camera. *Mach. Vis. Appl*. 2000,Vol.12, No.2, pp.53-58.

[21] Zhang Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell*. 2000,Vol.22, No.11, pp.1330-1334.

[22] Zhang Z, Matsushita Y and Ma Y. Camera Calibration with Lens Distortion from Low-rank Textures, *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, USA, June 20-25 2011, pp.2321-2328.