Wireless Visual-Servoing Using Primary Color Space in MATLAB

Syed Tahir Hussain Rizvi[1], Muhammad Yaqoob Javed[2], Kamal Shahid[3], and Tasweer Ahmad[4]
[1]Dipartimento di Automatica e Informatica, Politecnico di Torino, Turin, Italy
[2]Department of Automation, University of Science and Technology of China, Hefei, China
[3]Department of Electrical Engineering, (CEET), University of the Punjab, Lahore, Pakistan
[4]Department of Electrical Engineering, Government College University, Lahore, Pakistan
Email: syed.rizvi@polito.it, yaqoob@ustc.edu.cn, kamal.ee@pu.edu.pk, tasveer.ahmad@gcu.edu.pk

Abstract—**In this paper, Wireless Visual Servoing using MATLAB is implemented. MATLAB is used as Digital Image Processing tool for Objects and their Position Detection on the basis of Primary Color Space and then used as Primary Controller to navigate the robot about these positions. While Arduino is used as secondary controller to drive motors of robot to move towards the provided values of locations. Developed image processing algorithm, interfacing of Arduino and Wireless Camera with MATLAB are discussed in detail. Complete Control system is implemented and experimental results are obtained to verify Image Processing Algorithm.**

Index Terms—**object detection, primary color space, digital image processing, visual servoing, MATLAB, Arduino**

I. Introduction

Visual Servoing is to navigate a robot based on a visual sensor [1], [2]. An image or video captured from a camera can be used to detect objects and their positions in a scenario. Detected position of an object is always in terms of Image Coordinate system (row and column number of a pixel in digital image) [3]. This has to be translated for navigation of a robot, so it can be used in real environment.

Localization of robot can also be performed using Ultrasonic Sensors, Contact type Position Sensors and Compass Sensor etc. Contact type Position Sensors are avoided due to problem of inaccuracy [4]. Ultrasonic Sensor cannot cover a complete field or specific area, they can be used with other sensors to increase accuracy of algorithms, but cannot perform the operation of localization standalone [5]. Compass Sensor is affected by magnetic interference [6]. Hawk-eye based localization of robot is good for the applications where a specific area is to be covered [7]. So in this work, top view of field is used to perform localization of robot.

Vision based control system can be used as both open and closed loop control system according to requirement. In open loop control system, a single image is to be captured from a camera as shown in Fig. 1 [8], [9]. In second step, this image is brought into MATLAB for detection of objects and their positions using Image Processing. Then, these positions are transferred to a controller like Arduino to drive motors of robotic structure to perform desired task. This process is less accurate due to lack of feedback image but is more faster and simpler than closed loop control system where acquisition of image is required again and again as feedback to ensure accurate navigation of robot as shown in Fig. 2 [10].

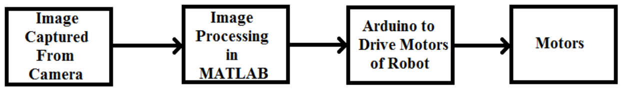

Figure 1. Open loop vision based control system.

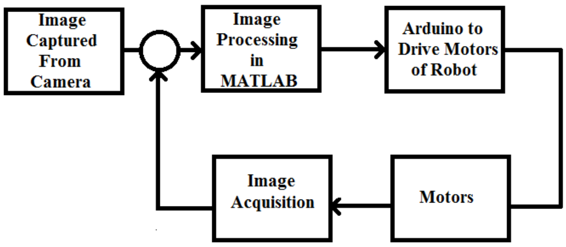

Figure 2. Closed loop vision based control system.

Basic purpose of this research work is to develop an understanding to perform Wireless Visual Servoing easily in MATLAB. Same setup can be used in different scenarios of Robotic competitions. In this work, image (top view) of an arena having black colored robotic structure, blue colored target and red colored obstacle is captured using Wireless IP (Internet Protocol) camera. Then this image is processed in MATLAB to find locations of colored objects. These locations in terms of pixel count are transferred serially to Arduino. Pixel count describes number of pixels to be travelled horizontally (x-axis) and number of pixels to be travelled vertically (y-axis) to reach the target, where Arduino translated these pixel counts in terms of signals to drive motors of robotic structure in a particular direction.

This paper is organized as follows: Section II describes the hardware components of control unit. System implementation is presented in Section III. Experimental results are presented in Section IV and last section concludes the results.

Manuscript received March 25, 2015; revised August 31, 2015.

doi: 10.18178/joig.3.2.132-137

II. Control Unit

This control structure is combination of different hardware components. All components play important role in implementation of Wireless Visual Servoing.

A. Controlled Robotic Structure

Three wheeler robotic platform is used for hardware implementation where a laptop can easily be placed to acquire image wirelessly and to find locations of colored objects using MATLAB. This structure has a lead screw mechanism to pick and place the objects as shown in Fig. 3.

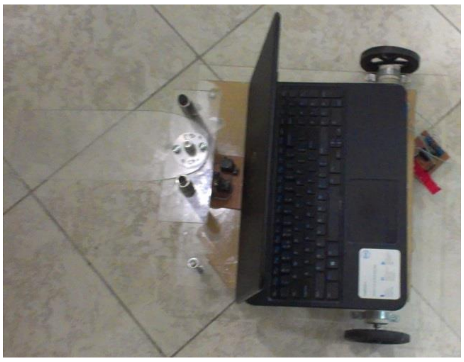

Figure 3. Used robotic platform.

B. Laptop as Primary Controller

A laptop is used as primary controller and a communication channel between camera and Arduino. Matlab is installed in this laptop for image acquisition and processing. Positions of robotic structure, target and obstacle are transferred to Arduino serially using USB cable to initiate the process.

C. Arduino as Secondary Controller

Arduino UNO having ATmega 328 is selected as secondary controller. Purpose of Arduino is to receive position of target serially and according to these values Arduino drives the motors to perform desired task.

D. Wireless IP Camera

Wireless Internet protocol Camera is used as visual sensor as shown in Fig. 4. A Power supply is needed for this camera. It provides audio, image and video footage of scenario. These facilities can be accessed using provided software or URLs (Uniform Resource Locators). By simply opening unique URL in web browser, respected service can be used. This camera is placed at top of arena to capture the scenario perfectly.

Figure 4. Wireless IP camera.

III. System Implementation

All the hardware components are connected as described in Section II. Camera is mounted above the arena to take image of target, obstacle and robotic structure. Designed algorithm is executed in MATLAB. Proposed algorithm performs three operations in MATLAB: i) Image Acquisition from Camera, ii) Image Processing for Detection of objects and their positions on the basis of primary color space and iii) Communication with Arduino.

A. Image Acquisition from Wireless Camera in MATLAB

All Wireless IP cameras provide their services (Audio, Image and Video footage) through graphical software or URLs. These URLs are unique addresses provided with every IP camera which can be used in any browser to access its services. For example all services of an IP camera having Ip address and port number http://192.168.1.4:8080 can be accessed using a browser as shown in Fig. 5.

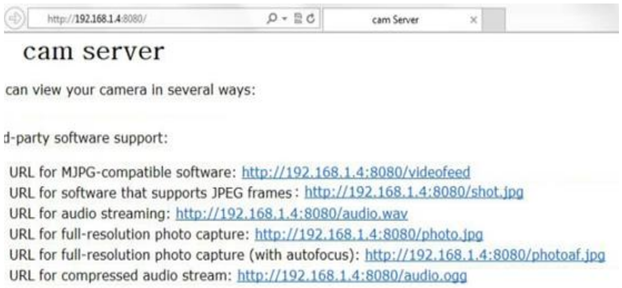

Figure 5. URLs to access services of an IP camera.

If it is required to capture an image, so it has a unique allocated URL.

In this particular case, http://192.168.1.4:8080/shot.jpg can be used to capture a single frame of image as shown in Fig. 6.

Figure 6. URL to access single frame of image.

In MATLAB, “imread” command is used to read an image from any URL/drive of hard disk. Syntax of this command is

A=imread(“Path or URL”);

If it is required to read an image named “new.jpg” present in Images directory in D:\ drive of hard disk, so command would be:

A=imread(“D:\Images\new.jpg”);

Similarly, provided URL is used in imread command to capture image of arena/workspace on which image processing has to be performed.

A=imread(“http://192.168.1.4:8080/shot.jpg”);

B. Image Processing in MATLAB

Image captured in previous step is a colored Image (RGB Image)having black colored robotic structure, blue colored target and red colored obstacle as shown in Fig. 7. This Image is used for detection of objects having primary colors and their positions in term of pixel count. For implementation of this algorithm, initially a sampled image is prepared in Microsoft Paint and later realized on real time image of scenario.

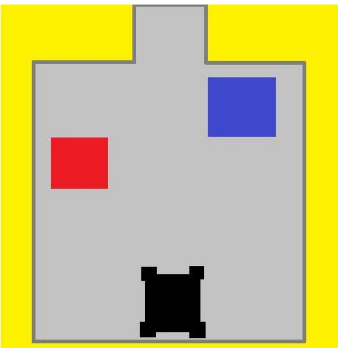

Figure 7. RGB image acquired from wireless camera.

A RGB image is a 3D Array defined as Rows x Columns x 3 [11]. Three Arrays/Layers of primary colors (Red, Green & Blue) participate to make a colored image as shown in Fig. 8. Combinations of different concentration of these colors define all shades of colors in an Image. Detection of a Primary color is an easiest task in an Image Processing Tool.

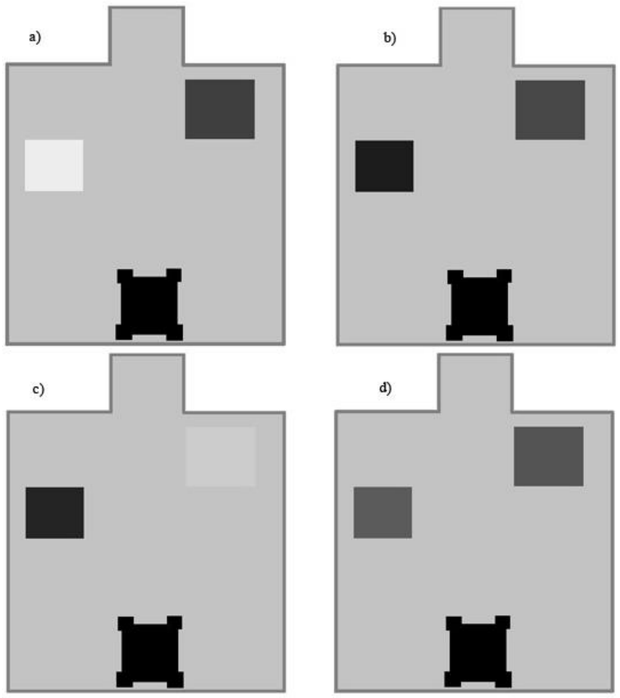

Figure 8. Sampled image of arena: a) red layer of image, b) green layer of image, c) blue layer of image, d) gray scale version of image.

All three primary layers can be separated easily in MATLAB. If acquired image is saved in variable “A”, so by writing A(:,:,1) red layer of image can be separated, similarly by writing 2 and 3 green and blue layers can also be extracted respectively as shown in Fig. 8(a).

Gray Scaled version of image plays an important role in Image Processing shown in Fig. 8(d). It can be obtained using “rgb2gray(A)” command in MATLAB.

As can be observed from Fig. 8, red layer of RGB image is representing the red object with brightest color (white) and all other objects in gray scale. Similarly, blue object has brightest pixel values in blue layer of Image.

1) Detection of colored object

If blue object has to be detected in a RGB image, so it can be achieved by subtracting gray scaled version of Image from Blue layer of RGB Image. MATLAB command is given as

Blue = imsubtract(A(:,:,3), rgb2gray(A));

This results in extraction of blue objects from scene as shown in Fig. 9.

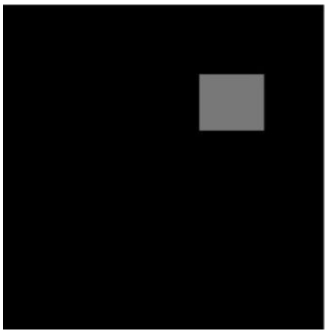

Figure 9. Extracted blue object from scene.

This is not a real time image, so factors like noise; room lighting and original shade of colored object can affect the detection of objects [9]. So, some type of filtration and way of defining required shade of color is an essential part in this algorithm.

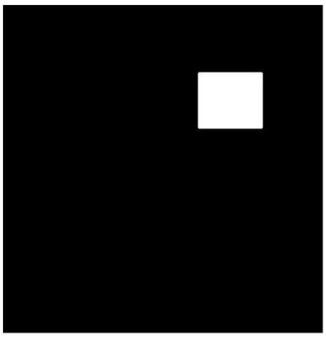

Figure 10. Detected object as white block.

So, median filter is used to remove the real time noise. This image is still a gray scale image; it has to be converted into black and white version having detected blue object as white and remaining objects as black as shown in Fig. 10. In MATLAB, “im2bw” command is used to convert an image into its black and white version (having only two colors) and its used value of threshold level 0.26 in this particular case defines the shade of color has to be detected. This value is decided on the hit and trial basis of analysis of achieved results. Similarly other objects having different colors can be detected in Image. This can be performed in MATLAB as follows

Blue = medfilt2(Blue, [3 3]);
BlueDetect = im2bw(Blue, 0.26);

2) To find location of detected target

A number of operations can be performed on a Black and White (BW) Image. Bwmorph command of MATLAB provides a way to manipulate a BW image to achieve desired results like unconnected pixels can be joined, holes inside a shape can be filled or size of object can be changed etc.

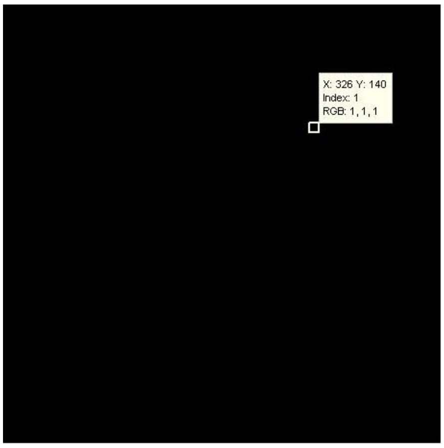

Figure 11. Detected center of blue object (target).

Using shrink operation of bwmorph command, size of detected solid object (box in this case) is reduced iteratively until only a single pixel which is centre of detected object is not detected as shown in Fig. 11.

detectedcenter= bwmorph(BlueDetect, 'shrink', Inf);

Now, using find command of MATLAB, all pixels having white value can be detected. These values are position of detected object. In this particular case, a single pixel is detected representing center of blue object in terms of values of x and y axis of Image coordinate system as shown in Fig. 11.

[yblue,xblue] = find(detectedcenter);

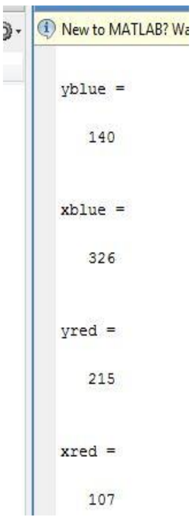

Figure 12. Detected locations of target (blue) and obstacle (red) in MATLAB.

Same steps can be repeated to detect remaining Primary colors (red and green objects) in a RGB image as shown in Fig. 12.

3) To find location of robotic structure

It is necessary to have robotic structure of such a color that can be distinguished easily from primary colors. For this purpose, robotic structure of black color is selected.

Black color can be detected easily in a gray scale image [12]. Subtraction of colored components is not required in this case. Only have to find zero valued pixels in an image using conditional operators to detect robotic structure in arena. MATLAB commands are as follows

Black = rgb2gray(A);
Black12 = [black==0];

Similarly, after detection of robotic structure, bwmorph and find command is used to obtain current location of robotic structure.

Now using difference of these horizontal, vertical distances between target and robotic structure, robot can be navigated to move to the desired target.

4) To find distance between target and robot

Distance between two dots/pixels can be calculated using Euclidean Distance formula. If “a” and “b” are two points having values of horizontal (x) and vertical (y) component, so distance between them can be find using (1).

$$D = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2} \quad (1)$$

p1=[xblueyblue];
p2=[xblackyblack];
d = sqrt((p1(1)-p2(1)).^2 + (p1(2)-p2(2)).^2);

These values can be used to obtain the distance between two points as shown in Fig. 13.

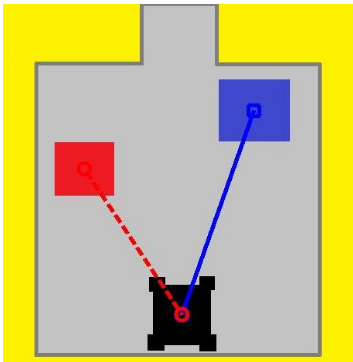

Figure 13. Euclidean distances between robot and other two objects.

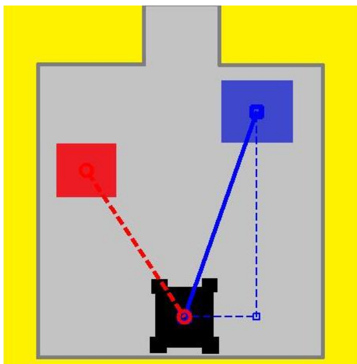

Figure 14. Distance to be covered in horizontal and vertical direction.

Using Detected locations, number of pixels to be travelled horizontally (x-axis) and number of pixels to be travelled vertically (y-axis) to reach the target can be found easily as shown in Fig. 14.

Simple mathematical operation of subtraction can be used to find horizontal and vertical distances in terms of pixels as calculated below and saved in variables named x_move and y_move.

```
x_move=p1(1)-p2(1)
y_move=p1(2)-p2(2)
```

C. Communication and Synchronization of MATLAB with Arduino

These values of horizontal and vertical distances are to be transferred to Arduino board. So, Arduino can translate these distances in terms of signals to drive motors of robotic structure to the targeted location. Arduino board is connected with Laptop through USB cable. A virtual COM port is created for communication between Laptop and Arduino Board. MATLAB uses this COM port to transfer values of locations/distances to Arduino Board. Given code is to transfer values of horizontal and vertical distances saved in “x_move” and “y_move” to Arduino using baud rate of 9600. In MATLAB, by using “% s” operator in fprintf command, string of characters can be transferred to other devices serially.

```
delete(instrfindall)
s=serial('COM9','BaudRate',9600);
s.terminator='LF/CR';
fopen(s);
pause(2);
fprintf (s,'%s\n',xmove)
fprintf (s,'%s\n',ymove)
fclose(s);
```

To drive motors according to received distances is a challenging task. Synchronization of motors’ movement and Distances in terms of Pixel count is achieved by repeating given three steps

First of all, robot is placed on a specified initial location, captured the picture and performed proposed image processing algorithm to find initial position of robot in terms of pixels.

In second step, robot is moved using Arduino. Delay of one second is generated using programming to drive the motors.

In third step, an image is captured again and used to find final position of robot. These steps are repeated again and again to find average distance covered in terms of pixels in one second by Robot. Now, using this calculated value, robot can be navigated to reach the target.

IV. Experimental Results

In this section, experimental results are presented. Same algorithm is used on real time image of arena and objects as shown in Fig. 15. Target Object of blue color is detected in image as sown in Fig. 16. But due to change in room lighting and real shade of colored object, more than one pixels are detected for a single object as shown in Fig. 17. While it is required to find only single pixel representing center of detected object. So average value of these pixels has to be calculated in MATLAB to get one pixel for one object. It solved the problem and perfectly detected the distance between target and robot as shown in Fig. 18.

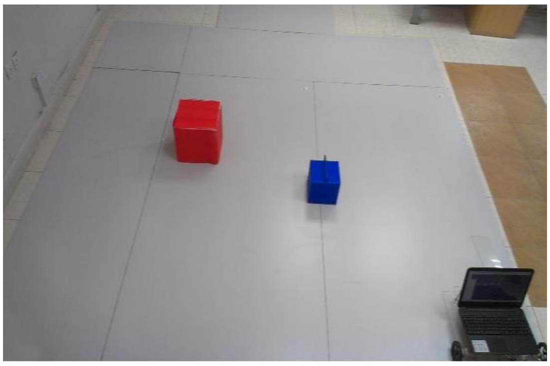

Figure 15. RGB image of arena.

Figure 16. Detected target of blue color.

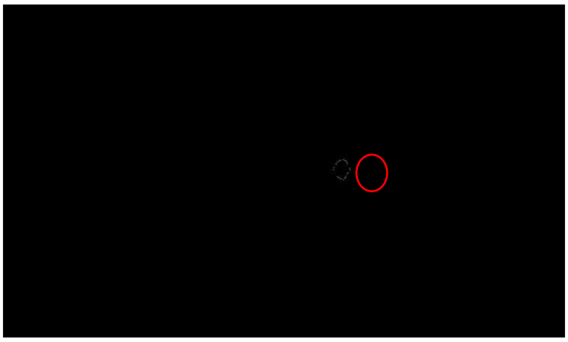

Figure 17. Detection of more than one pixel of target.

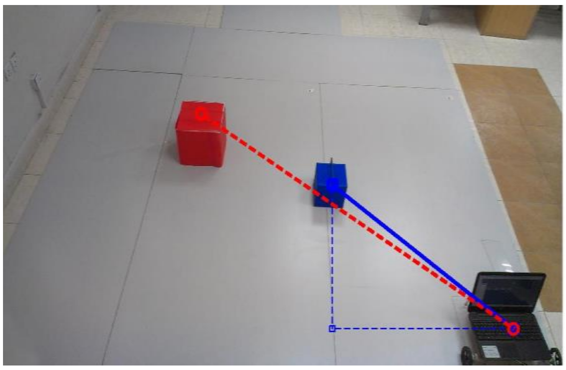

Figure 18. Euclidean, horizontal and vertical distances between target and robot.

V. CONCLUSION

Wireless Visual Servoing in MATLAB with Simulation and Experimental results has been presented in this paper. Proposed Vision based Control System is implemented using Wireless IP Camera as Visual Sensor. Image Processing Algorithm is illustrated in detail. Real Hardware system is presented.

Same Vision based Control system can be used for picking and placing of heavy loads having distinguished Primary colors. By mounting camera on mechanical gripper (called eye-in-hand configuration), same image processing algorithm can be used for sorting of objects having colored markers.

REFERENCES

[1] F. Chaumette and S. Hutchinson, "Visual servo control, part ii: Advanced approaches," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 109-118, March 2007.

[2] S. Hutchinson, G. Hager, and P. Corke, "A tutorial on visual servo control," *IEEE Trans. on Robotics and Automation*, vol. 12, no. 5, pp. 651-670, 1996.

[3] I. Dinulescu, A. Predescu, G. Boccolato, R. T. Tanasie, and D. Cojocaru, "Control of a hyper-redundant robot," in *Proc. 19th IEEE International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD)*, 2010, pp. 435-440.

[4] A. M. H. S. Abeykoon, L. Udawatta, M. S. Dunuweera, R. T. Gunasekara, M. Fonseka, and S. P. Gunasekara, "Enhanced position sensing device for mobile robot applications using an optical sensor," in *Proc. IEEE International Conference on Mechatronics (ICM)*, 2011, pp. 597-602.

[5] D. Sanjay, T. S. Savithri, and P. R. Kumar, "Person follower robotic system," in *Proc. International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, 2014, pp. 1324-1327.

[6] T. Agrawal and D. Gopinath, "Localization using relative mapping technique for mobile soccer robots," in *Proc. International Conference on Communications and Signal Processing (ICCSP)*, 2013, pp. 265-269.

[7] L. Wang and G. Zhao, "A localization system for mobile robots based vision," in *Proc. 8th World Congress on Intelligent Control and Automation (WCICA)*, 2010, pp. 6671-6675.

[8] A. M. Al-Busaidi, "Development of an educational environment for online control of a biped robot using MATLAB and arduino," in *Proc. 13th Int'l Workshop on Mecatronics-REM*, 2012, pp. 337-344.

[9] K. Lutvica, N. Kadic, G. Dzampo, H. Muminovic, J. Velagic, and N. Osmic, "Remote position control of mobile robot based on visual feedback and ZigBee communication," in *ELMAR*, 2011, pp. 169-172.

[10] A. Burlacu, O. Mihai, and V. Tiganas, "An image based visual servoing scheme for wheeled mobile robots," in *Proc. 15th International Conference on System Theory, Control, and Computing (ICSTCC)*, 2011, pp. 1-5.

[11] C. Pop, S. M. Grigorescu, and A. Davidescu, "Colored object detection algorithm for visual-servoing application," in *Proc. 13th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*, 2012, pp. 1539-1544.

[12] J. A. Shajahan, J. S. Jain, C. Joseph, A. Amodkumar, V. Chauhan, G. Keerthipriya, and P. C. K. Raja, "Target detecting defence humanoid sniper," in *Proc. Third International Conference on Computing Communication & Networking Technologies (ICCCNT)*, 2012, pp. 1-6.

Syed Tahir Hussain Rizvi completed his BS Electrical Engineering from University of Central Punjab, Lahore, Pakistan and M.Sc. Computer Engineering from University of Engineering and Technology, Lahore, Pakistan. He is a PhD Scholar in Politecnico di Torino, Dipartimento di Automatica e Informatica, Turin, Italy. His area of research includes Embedded Systems, Image Processing and Artificial Intelligence.

Muhammad Yaqoob Javed completed his M.Sc. Computer Engineering from University of Engineering and Technology, Lahore, Pakistan and he is a PhD Scholar in University of Science and Technology of China, Hefei. His area of research includes Embedded Systems, Robotics and Power Electronics.

Kamal Shahid completed his BS Electrical Engineering from University of Central Punjab, Lahore, Pakistan and M.Sc. Computer Engineering from University of Engineering and Technology, Taxila, Pakistan. He is currently working as a Lecturer in the Department of Electrical Engineering, University of the Punjab, Lahore, Pakistan. His area of research includes Embedded Systems, Voice Processing and Control Systems.

Tasweer Ahmad received his B.Sc. Engineering from University of Engineering and Technology, Taxila, Pakistan and M.Sc. Engineering from University of Engineering and Technology, Lahore, Pakistan. His area of research includes Image Processing, Computer Vision and Machine Learning.