International Conference on Mechanical, Industrial and Energy Engineering 2014
25-26 December, 2014, Khulna, BANGLADESH

ICMIEE-PI-140101

An Android Controlled Mobile Robot for Stereo Vision and Live Streaming with Robotic Arm

Md. Muhaimin Rahman, Ashik-E-Rasul, Nowab Md. Aminul Haq, Md. Mehedi Hassan
Department of Mechanical Engineering, Bangladesh University of Engineering & Technology, Dhaka-1000, BANGLADESH

ABSTRACT
Mobile robotics and Android Operating System are two of the most advancing technologies at present. On the other hand, stereo vision technology is becoming more and more popular among the present youth. This project combines all of these advancing and developing technologies for humane purpose. The robot uses two smart phones with android operating system to send videos to the control server continuously and that will generate a stereoscopic image, a 3D live video from those images. In addition to that, a robotic arm is placed in front of that robot to pick any desired object . The whole process is be controlled by another smart phone with android operating system. The robot also can be controlled by the server computer only. Its usage are many. First of all, it can be sent to any place which is humanly impossible and do any task needed. For being small in size, it can be sent to any dangerous places like caves or underground operations also can be done by this robot. We hope this project can reduce the cost, human error, promote human trust and enable a robotic application which never has been possible before .

Keywords: Stereo Vision, Image Processing, Android OS, Mobile Robot.

1. Introduction

This paper focuses on a multipurpose mobile robot for stereo vision, live streaming controlled manually by an Android OS operated cell phone. It has a robotic arm to pick and place objects. The robot will have another two cell phones with Android OS used as their robotic eyes. An android application will transmit images captured by the cell phones continuously over wireless network to the server computer. The server computer will process the images using to generate stereo images out of them. Using those images, it will map any unknown area.

This paper is segmented into three parts. First portion details the architectural and mechanical construction. In the second portion, electrical design of the system is discussed. The final portion covers the software development and the algorithm of the robot.

2. Mechanical Construction

The mechanical body the robot consists of three main parts. Chassis with four wheel drive, pan and tilt mechanism and a robotic arm.

2.1 Chassis with four wheel drive:
The chassis of the robot body is made by bending aluminum sheets and acrylic sheet. Four wheels are connected to four high torque gear motors to facilitate a four wheel drive.

2.2 Pan and Tilt mechanism:
The android phones can be panned 180 degrees at different speed with a 180 degree high torque servo motor. However they can also be tilted with another 180 degree servo motor.

2.3 Robotic arm:
The robotic arm is at the front side of the robot. It has two degrees of freedom. The gripping mechanism used here is "Four Bar Linkage End Effector."[1]. In this mechanism, the gripper is operated with a gear drive connected to a servo motor. The up-down motion of the arm can be controlled by another servo. Gears involved in the gripping mechanism were made from acrylic sheet with precise laser cutting.

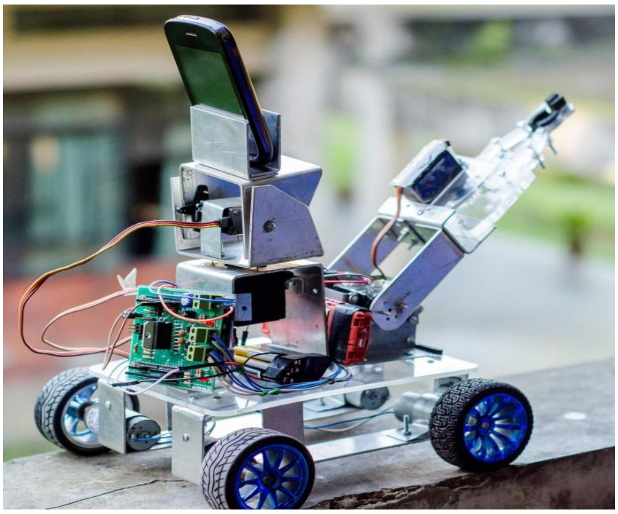

Fig. 1 Prototype

2.4 Steering:
For steering, "Skid Steering Method" [2] is used in this robot. This method engages one side of the tracks or wheels and turning is done by generating differential velocity at opposite side of a vehicle as the wheels or tracks in the vehicle are non-steerable.

Generally, differential drive concept is used in making robots which are two wheel drive robots. But that method has some disadvantages. In differential drive method, the robot does not drive as expected. It neither drives along a straight line nor turn exactly at expected

* Corresponding author. Tel.: +88-01557335807
E-mail address: sezan92@gmail.com

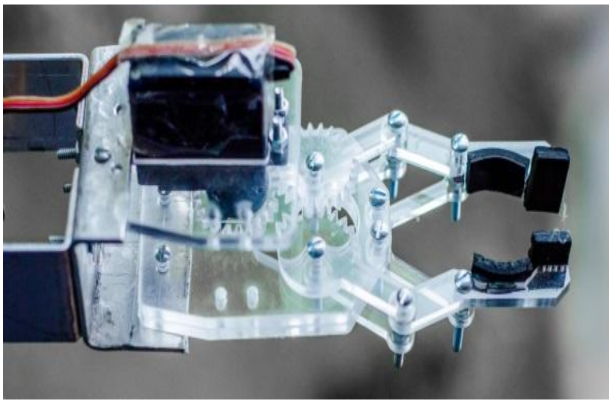

Fig. 2 Gripping Mechanism of the Claw

angles, especially when we use DC motors. So to overcome these problems, we used the skid-steering method.

3. Electrical Design

For locomotion, four wheels with four gear motors are used. For camera movement two servo motors are used and for the arm another two servo motors are used. These motors are controlled remotely from an android operated cell phone. A Bluetooth module 'HC-05'[3] is used to connect the system to the cell phone wirelessly. Whole Electrical System is described below,

3.1 Arduino

Arduino is a single-board microcontroller, intended to make the application of interactive objects of environments more accessible. The Hardware consists of an open source hardware board designed around an 8-bit Atmel AVR microcontroller. Here we've used Arduino Uno R3.[4] It has six analog output pins or "Pulse Width Modulation" or "PWM" pins. These pins

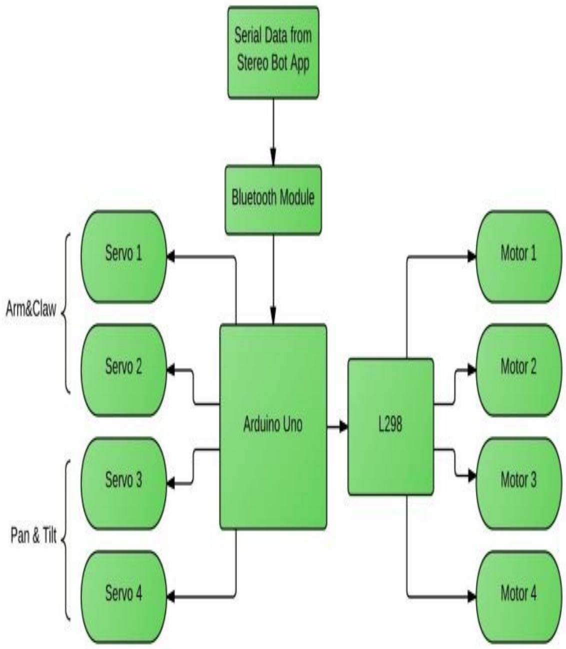

Fig. 3 System Block Diagram

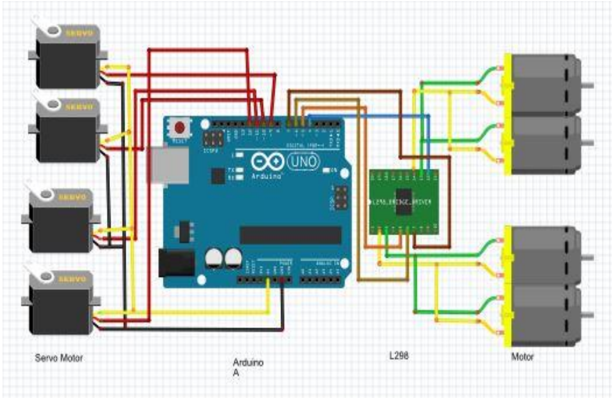

Fig. 4 Circuit Diagram using fritzing®

are enough for controlling the gear motors and servo motors.

3.2 Gear Motor Control

For locomotion, four high torque and high speed gear motors are used. Arduino boards cannot provide sufficient current and voltage for controlling these motors. So, a motor shield with dual bridge motor controller named L298 [5] is used. Four diodes are added to the motor shield to protect the circuit damage from back electromotive force generated from the motors. Two PWM pins D5 and D6 are connected to the input pins of the motor controller. The other pins are D4 and D7 respectively. PWM pins are used to control the speed of motors.

3.3 Servo Motor Control

A servo motor is a rotary actuator that allows for precise control of angular position, velocity and acceleration. In this particular robot, four high torque, coreless, metal gear dual bearing "Tower Pro MG995" - Standard Servo Motors are used. Two of them are used to control the arm. Remaining two of them are used to

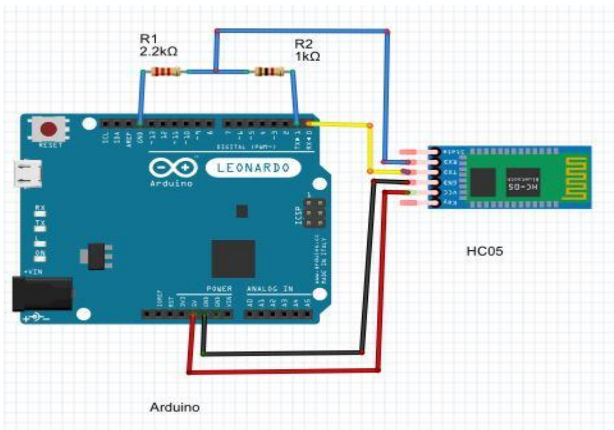

Fig.5 Arduino connection with HC05 using fritzing®

control the Camera pair. One servo motor rotates the cameras and the other one tilts the camera pair. Servo motors work on the pulses from microcontroller.

Changing the duty cycles of the pulses will change the angle of the servo motors. For zero degree position duty cycle must be 388 micro seconds, for ninety degree position duty cycle must be 1240 micro-seconds [6].

3.3 Bluetooth Communication

An Android Application will control the robot using Bluetooth connection. So, a Bluetooth module named HC-05 is used. This module works on 3.3 voltage level, while about all other electronic components work on 5 volt level. That means, to work with this module, we must make the signals voltage about 3.3 volt. To do that, a circuit connection like Fig.5 was made.

3.4 Software Development

3.4.1 Generating Stereoscopic Images

Stereo Vision: "Stereo vision is the extraction of 3D information from Digital Images." [7] At least two images of the same object is needed from two cameras. Here we used two cell phones as robotic eyes. The Distance between the two lenses is known as Baseline. The system will generate stereo images from those two images. There are some important steps to generate this image.

a) Stereo Image Rectification

First step for stereoscopic vision is to rectify stereo images. Image rectification is a transformation process used to project two-or-more images onto a common image plane. It corrects image distortion by transforming the image into a standard coordinate system.[8] We took help of MATLAB® documentation[9] regarding this very important process . The necessary steps are given below

Step 1: Reading the stereo image pair (Fig. 6)

Fig.6 Captured Images

Step 2: Generating points of correspondence. Points of interest are collected between two images and potential matches are found out using Speeded-Up Robust Features (SURF) algorithm. They will find blob-like features in both images [10].

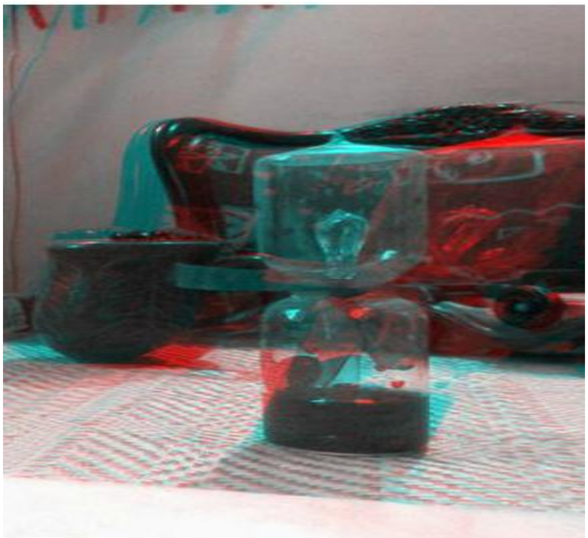

Fig. 7 Rectified Stereo Image

Step 3: Finding Putative Point Correspondences using Sum of Absolute Differences SAD. This algorithm takes absolute difference between each pixel in the original block and corresponding pixel in the block being used for comparison. These differences are summed to create a simple metric of block similarity. [11]

Step 4: Epipolar Constraints must be satisfied by the matched points. So, in this step , fundamental matrix was computed to check the inliers meet the epipolar constraint or not. [12] Then, outliers were removed using epipolar constraint.

Step 5: In this last step, the rectification transformation was computed. As a result, corresponding points appeared on the same rows. Finally, images were rectified using projective transformations and the overlapping area was cropped. (Fig.7)

b) Depth mapping

In 3D computer vision system, "Depth mapping" is referred to the process of measuring relative distances of surfaces of scene objects from a view point. At least two images are needed for depth mapping. Processing more images will result in more accurate depth map.[16]. It is worthy of mentioning that, we took a great help from MATLAB® documentation[9] in this regard. There are four steps for this particular process,

1) Basic Block Matching:
2) Sub-pixel Estimation
3) Dynamic Programming
4) Combined Pyramiding and Dynamic Programming.

The Block Matching block estimates motion between two images or two video frames using blocks of pixels. [13]. For every pixel in the right image, the 11-by-11-pixel block is extracted around it and searched along the same row in the left image for the block that best matches it.

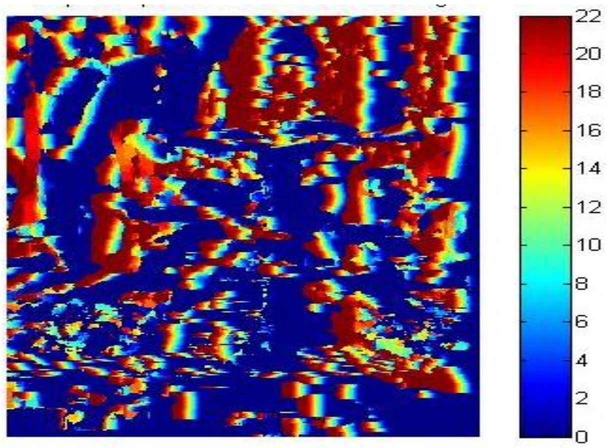

Fig. 8 Depth Map by Basic Block Matching

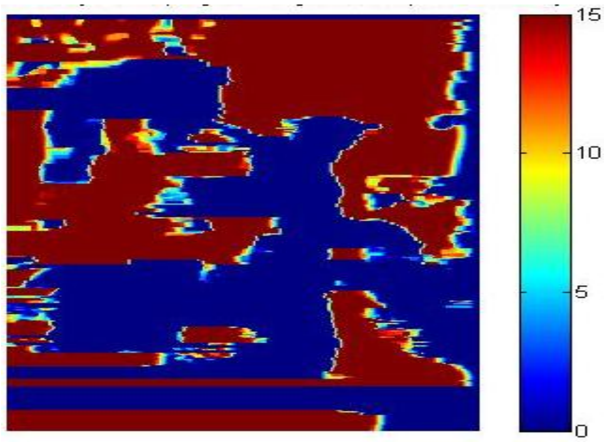

Fig.9 Depth Map for dynamic programming and sub-pixel accuracy.

The pixel's location in the first image is searched in a range of pixels around, and the sum of absolute differences (SAD) is used to compare the image regions (Fig.7). "Templatematcher" system object was used to perform block matching. But in this step a noisy disparity image is created, which can be improved by introducing smoothness constraint. To remove these problems, "Subpixel Estimation" is used. At the end of this process, block matching was rerun .The result is given in the. The pseudo code is given below.

1) Set minimum and maximum row bounds for image block
2) Compute disparity bounds
3) Construct template and region of interest
4) Is there proper Template matcher object?
5) Run the" Templatematcher" object
6) Do "Subpixel Refinement"
7) Run Block Matching
8) Show the final figure

The final result with above process is shown in Fig.8. To improve the noise in the resulted figure, image pyramiding method and dynamic programming were used. [14][15] This improved the accuracy of the stereo image. The final result is shown in Fig. 9.

c)Backprojection:
Back Projection is the process of showing the position of every pixel in the three-dimensional image in a three-graph [9]. That is, it positions the objects in the plot and using that plot we can determine the exact position of every object. It's pseudo code is as following,

1) Declare Camera Intrinsic Matrix
2) Create a sub-sampled grid for back projection
3) Median Filter to smooth out noise
4) Derive conversion from disparity to depth with tie points
 4.1) Set Disparity values in pixels
 4.2) Set World "z" values in meters based on seen measurements.
5) Convert disparity to distances from camera using least square method
6) Set threshold
7) Remove near points
8) Show the 3-D plot.

To show this plot, intrinsic parameters of the cameras must be determined. Here, camera calibration toolbox[16] from California Institute of Technology was used to determine the intrinsic parameters . The parameters are shown in the following Table.

Table.1 Intrinsic Parameters of the Cameras

Parameters	x	Y
Focal Length	7305.527	-6009.122
Principal Point	239.5	319.5
Skew Factor	0	0

To measure the distances from the camera, an equation [17][18] is used in this process.

$$z_{world} = f + \frac{1+b}{d}$$

Here,

$z_{world} = Distance\ from\ the\ Camera$
$f = focal\ length\ of\ the\ Camera$
$b = stereo\ baseline\ of\ the\ cameras$
$d = disparity$

We know the focal length of the cameras. But we need to determine the baseline. That is done in MATLAB® . We gave input for known disparities the known distances of points from cameras. Doing simple regression analysis, the baseline can be measured. The final plot is shown in Fig. 10

3.4.2 Android Application for robot control:

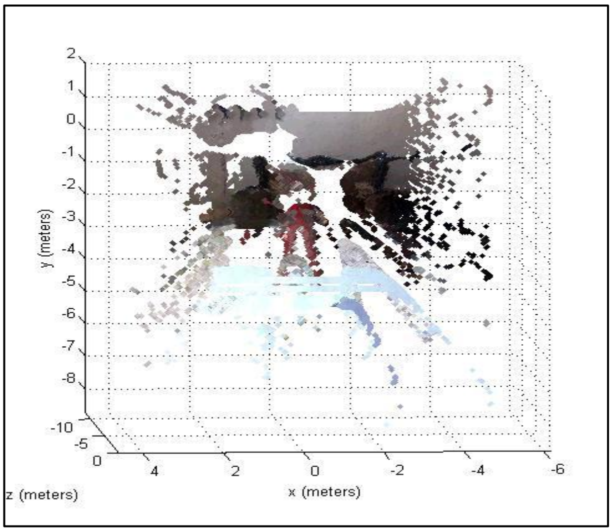

Fig. 10 Three dimensional plot after back projection

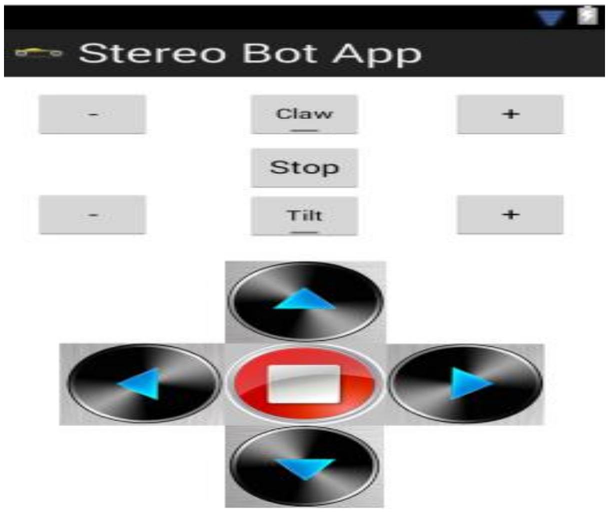

Fig. 12 Android Application interface

An android application named "Stereo Bot App" was developed to control the movement of the robot. Stereo bot connects HC-05 Bluetooth Module with the smart phone. While the connection is established serial data can be sent either from phone to Arduino or from Arduino to phone. The app works in two different control modes: Button control and accelerometer control. As described earlier, serial data can be sent to Arduino from two different user interfaces. The user Interface is given in the Fig.12. When an on screen button is touched a predetermined character assigned to that button is sent over Bluetooth network. For example, if forward button is pressed, 'w' character is sent. Other buttons work in a similar way.

Another control mode of Stereo Bot App is controlling using Accelerometer. Android's accelerometer can detect rotation of the phone around x, y and z axes and express the rotations in numerical values. These values are measured at an interval of 250 milliseconds. Working principle of accelerometer control is described in the table. The corresponding robot movements are illustrated in Table 2..

Table.2 Accelerometer Value and Robot Movement.

Axes value	Robot Movement	Assigned Character
y<-4	Forward	W
y>4	Backward	S
x<-4	Left	A
x>4	Right	D
-2<x<2	Stop	X
-2<y<2	Stop	X

A plot is generated based on accelerometer data provided by "SensoDuino®". (Fig.13) However, for arm-claw and pan-tilt control some on screen buttons are to be used.

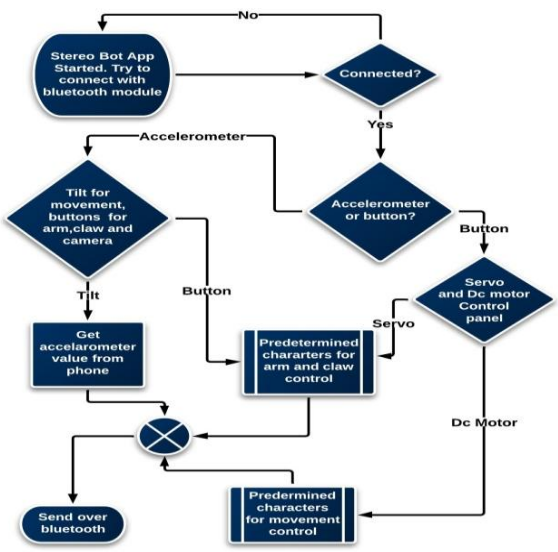

Fig. 11 Android Application Algorithm

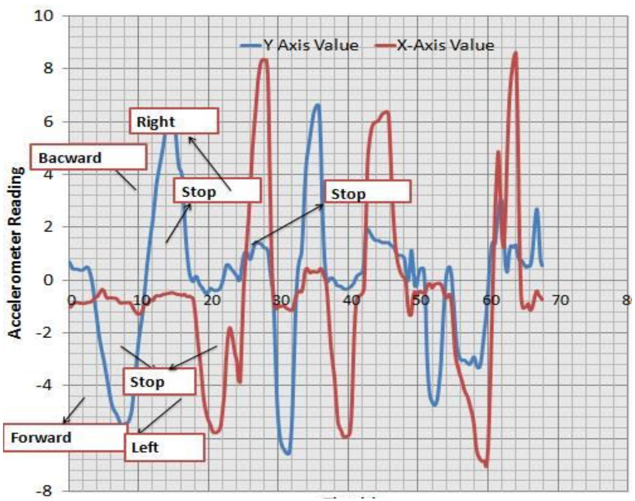

Fig. 13 Accelerometer reading vs time curve

REFERENCES

[1] Michael L. Turner, David A. Perkins, Andrew P. Murray,Pierre M. Larochelle "SYSTEMATIC PROCESS FOR CONSTRUCTING SPHERICAL FOUR-BAR MECHANISMS", ASME International Mechanical Engineering Congress and Exposition , Orlando, Florida USA. (2005)

[2] K. Kozlowsk, K. Pazderski, I. Rudas, J. Tar, "Modeling and Control of 4-Wheel Skid-Steering Mobile Robot: From Theory to Practice," in *International Journal of Applied Math and Computer Science*, Vol 14, n. 4, pp 477-496, (2004)

[3] HC-05 Bluetooth Module Datasheet. Available from http://www.scribd.com/doc/128811636/Datasheet-Bluetooth-Module-Hc05 . Accessed 17 November, 2014.

[4] Massimo Banzi, *Getting Started with Arduino*, O'Reilly Media Inc. (2009)

[5] L298 Datasheet. Available from https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf . Accessed 17 November , 2014

[6] Riazollah Firoozian, Electrical DC Servo Motors, *Servo Motors and Industrial Control Theory* , Springer Science & Business Media, Dec 4, 2008

[7] Richard Szelisk, *Computer Vision: Algorithms and Applications*, Springer Science & Business Media, Sep 30, 2010 .

[8] A. Fusiello, E. Trucco, and A. Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16-22, 2000.

[9] MATLAB and Computer Vision Toolbox Release 2012b, The MathWorks, Inc., Natick, Massachusetts, United States.

[10] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346—359. (2008)

[11] E. G. Richardson, Iain. *H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia*. Chichester: John Wiley & Sons Ltd. (2003)

[12] R A Hamzah, M. S. Hamid, H. N. Rosly, N. M. Z. Hashim and Z. A. F. M. Napiah ,"An Aligned Epipolar Line for Stereo Images with Multiple Sizes ROI in Depth Maps for Computer Vision Application .", International Journal of Information and Education Technology, Vol. 1, No. 1, April 2011.

[13] Aroh Barjatya, Block matching algorithms for motion estimation, 'Tech. Rep., Utah State University. (2004)

[14] Oliver Schreer, Peter Kauff, Thomas Sikora, Rendering with Geometry Compensation, *3D Videocommunication: Algorithms, concepts and real-time systems in human centred communication*, pp 167, John Wiley & Sons, (Nov 1, 2005).

[15] Burt, P. J. "Fast filter transform for image processing". *Computer Graphics and Image Processing* , (May 1981).

[16] Bouguet, JY. "Camera Calibration Toolbox for Matlab." Computational Vision at the California Institute of Technology. http://www.vision.caltech.edu/bouguetj/calib_doc/

[17] M. Okutomi , T. Kanade, A Multiple-Baseline Stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence, v.15 n.4, p.353-363, (April 1993)

[18] Md. Shafayat Hossain, Ahmedullah Aziz, and Mohammad Wahidur Rahman “RAFI - A STEREO VISION BASED AUTONOMOUS MOBILE AREA MAPPING ROBOT WITH 16 DOF”, International Journal of Artificial Intelligence & Applications (IJAIA), Vol.4, No.1, January 2013.

[19] Stephan Gobel, Ruben Jubeh, Simon-Lennert Raesch and Albert Zundorf: “Using the Android Platform to control Robots”. Software Engineering Research Group, Kassel University.