International Journal of Pure and Applied Mathematics
Volume 118 No. 16 2018, 213-230
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
Special Issue
ijpam.eu

Design, Implementation and Analysis of a Low Cost Drawing Bot for Educational Purpose

Rajesh Kannan Megalingam[1], Shreerajesh Raagul[2],
Sonu Dileep[3], Sarveswara Reddy Sathi[4],
Bhanu Teja Pula[5], Souraj Vishnu[6],
Vishnu Sasikumar[7] and Uppala Sai Chaitanya Gupta[8]
[1 2 3 4 5 6 7 8]Department of Electronics
and Communication Engineering,
Amrita University, Amritapuri, India
rajeshkannan@ieee.org, shreerajgul@gmail.com,
sonudileep@gmail.com, sarveswarareddysathi@gmail.com,
pulabhanuteja@gmail.com, sourajvishnup@gmail.com,
vishnusasikumarpp@gmail.com, uschaitanyagupta776@gmail.com

January 9, 2018

Abstract

This low cost drawing bot is an embedded system which works on the basis of Computer Numerical control. This paper deals with the design, implementation and analysis of a low cost drawing robot for educational purposes. With the aim of boosting the students robot making interests and their technical expertise to the next level, the research elaborates on the technology behind building this low cost bot. The hardware includes a simple embedded system with low cost microcontroller board. The bot has a 2 axis control using the stepper motor and the pen is controlled using a servo motor. It uses Inkscape software for drawing images and to create G-code files. Universal G-code sender sends these G-code files to microcontroller board for controlling the pen/pencil to move in x, y and z directions to draw the image. The experimental results show that how close the images drawn by the bot to that of the original one. In addition the details simple way of building this bot is also provided in this paper.

Key Words : drawing robot; education; Inkscape.

1 Introduction

In recent years we have witnessed a drastic influence of robotics in a lot of scientific and technological fields. Robotic technologies are being extensively used in the field of education and entertainment. This paper introduces a robot that has artistic abilities of drawing and writing. The robot is capable to draw sketches similar to how a human perform with greater precision and accuracy. It is a complex and challenging task to assign a robot with the task of performing drawing and writing which requires high degree of innovation and creativity. The drawing robot is designed to work with the help of open source software Inkscape and G- code sender. Inkscape provides a platform to input the sketches and then convert them into G-code. The robot has the capability to draw sketches in a workspace of area 40 cm x 30 cm. Mechanical parts were designed using CAD tools and printed using 3D printer. The main goal of this research is to develop a drawing robot which can draw complex sketches on a paper. The robot can make significant influence in training and other educational purposes. This robot will not only encourage people to involve in artistic activities but also infuse interest in people to actively participate in robotics and programming.

2 Motivation

With the progress in technologies, there has been a large advancement with respect to the accessibility of robots. Now-a-days robots can be easily designed and incorporated with multitude of sensors and motors as well as interface with a desktop computer for robust programming experience. Robotic parts can be easily procured at a low cost. Use of Robotic technologies is the best way to interact with the rapidly advancing technological world. A robot with drawing capabilities would definitely inculcate interest in people to experiment with robots. Apart from that, a drawing robot can be used for training students to draw complex images with perfect accuracy and precision. Practical applications of a drawing robot in education sector is numerous.

3 Related Works

Numerous methods exist for autonomous drawing and writing using robotic systems. Large number of attempts aimed at designing a drawing robot can be found in the robotics literature. However majority of these works uses a humanoid robot to make the drawing process look human like. One such fully automated humanoid system capable of drawing was developed in EFPL by using HOAP-2 robot[1].

The robot is capable to draw the portrait of the person sitting in front of it. The robot makes use of image processing techniques and classical trajectory planning algorithms. Design of an interactive educational drawing system using a humanoid robot is presented in the research paper [2]. The system uses light polarization property to draw shapes appearing on the computer screen. Paper [3] presents a human portrait generation system using PICA, a two armed humanoid robot. The robot converts the image of the face into line segments and execute drawing within short span of time. The research work presented in [4] aims to develop a painting robot with multi-fingered hands and stereo vision. After observing the object, the system obtains a three dimensional model of the image and the paint it on a canvas using a painting brush. The authors in [5] presents a 3 DOF robotic arm which can be used for drawing images on a paper canvas. The robot is built using LEGO NXT bricks. Inverse kinematics is calculated on a PC connected to the NXT via Bluetooth. The artist robot in [6] makes use of a KUKA KR6 robot. The end-effector is specialized to serve dual purpose. The effector not only holds the pen used for drawing but also the eraser used for wiping operation. Betty, a 12 DOF humanoid robot is a portrait drawing artist. The robot was developed using modified Theta-graph, called Furthest Neighbor Theta-graph [7]. The research work in paper [8] introduces the design and implementation of a drawing robot which makes use of MATLAB and Arduino Mega.Implementation of a low cost 3D printed robotic hand is discussed in [9]. Bluetooth control of robotic arm discussed in paper [10] introduces the scope of wireless interface and interactions.

4 System Architecture

The system architecture for the drawing robot is depicted in Fig1.1. The system architecture mainly consists of 6 parts. The image given as input to the system is first converted into G-code. The G-code is then sent to the microcontroller which facilitates the interfacing of the CNC shield. CNC shield converts the G-code into stepper motor signals. The motors will cause the movement of the belt attached to it. Synchronized movement of the belt will then guide the pen to draw sketches in the drawing pad.

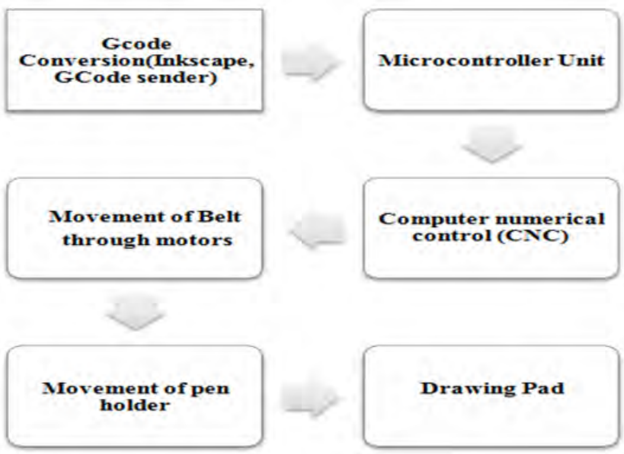

Figure 1: Block diagram of drawing robot

4.1 G code conversion

Inkscape is a vector graphic software .We are using Inkscape for converting image into XYZ prospective so that we can point out each and every pixel by XYZ axis.

To incorporate Inkscape to drawing robot we have added some software libraries so that we can control z axis by servo motor. Without these libraries the Inkscape can control only stepper motors and z axis commands get converted into stepper motor signals. Z-axis control using servo is essential to lift the drawing pen up and down where ever necessary while drawing. Instead of servo we can also use stepper motor but it is not required in this case. Inkscape takes the image of the drawing as input to it and generates G-code of the drawings. These G-codes are then given as input to the software Gcode Sender as shown in Fig1.2.

The software UGCS (Universal G-code Sender) was used for this research . It is a java based GRBL compatible cross platform G-code sender. It helps in running GRBL controlled CNC machines. It enables sending of converted images to Arduino to facilitate the working of CNC machine.

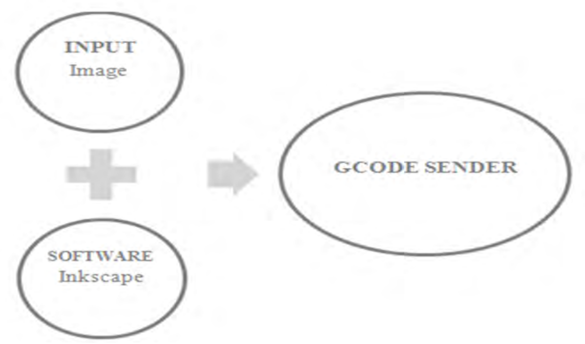

Figure 2: Generation of G-code

4.2 Micro Controller Unit - Arduino Uno

We are controlling XYZ movement of the robot by Arduino Uno and CNC shield. To facilitate communication of the CNC machine with the G-code sender there is a need to upload the GRBL program to the Arduino. This is done by adding extra software libraries to Arduino. We have used grbl-Core XY-Servo-Master library.

Arduino Uno is a Microcontroller based on ATMEGA328P. There are 14 digital pins(I/P,O/P) with 6-PWM output pins and 6 analog inputs,16MHZ quartz crystal, power jack, ICSP header, reset button. Arduino Uno is programmed with the Arduino Software IDE. It can be powered by USB or by external power supply. External power supply can be either battery or AC to DC adaptor. Positive and negative terminals of the source can be connected to GND pin and VIN pin of Arduino directly. In Arduino Uno there are ±5v and GND pins inorder to take power from Arduino. ATMEGA 328P has 32kb memory. It also has 2kb SRAM and 1kb EEPROM. In addition, there are few pins having special functions like rx(pin0), tx(pin1), external interrupt (pin2 and 3), PWM (pins 3,5,6,9,10,11), SPL (pins 10,11,12,13), Led (pin 13) and TWI A4 or SDA,A5 or SCL and pin AREF.

4.3 CNC Shield

CNC Shield is a open source firmware on Arduino which controls 4 stepper drivers using 4 different motor drivers. It is GRBL compatible and has 4-Axis support (2x end stops for each axis).It runs on 12-36v power supply. It runs on Arduino and converts G-code into stepper motor signal.

4.4 Stepper Motor NEMA 17

It is a Phases-2 motor with 200 steps/revolution and 5% step accuracy. Other specifications include shaft load-20,000 hours ay 1,000 rpm, Axial 25N push,65 N pull, IP rating-40, insulation resistance-100megohms,4-wire bipolar stepper has 1.8 degrees per step for smooth moment.

4.5 Servo Motor

It can rotate in 180 degrees(90 degrees in each direction) with torque-2.5kg-cm, speed-0.1sec, voltage 4.8 to 6v. In this robot servo helps in lifting pen holder so that we can lift pen whenever drawing on the paper is not needed.

5 Design and Implementation

5.1 'H' arrangement of the belt

The robot has been implemented using H arrangement as shown in Fig1.3. This H arrangement belt helps in positioning the object in x-y plane. It can point out every point in square without any overshoot or undershoot. In this type of arrangement each motion profile moves in each side of square which is defined as geometry of a square projected on X and Y axes and apply inverse kinematics to obtain position at motor shafts. This motion profile helps in creating perfect corners.

Mechanical design consists of base which can hold vertical and horizontal rods, Belt arrangement consists of 4 smooth idler pulleys in order to achieve H bridge arrangement using a belt. The belts run over two stepper motors. The ends of the belt are fixed to a stationary point. A connection between the belt and pulleys helps in attaining movements in x directions and y-directions. When two motors rotates in same direction, the belt move in x-direction (horizontal direction), left and right motion of the holder will be decided by the movement of the motors in same direction. Front and back motion of pen holder is decided by rotation of the motors in opposite directions (motor1-clockwise, motor2-anti-clockwise or vice versa).

Movement of the pen holder in Z axis is controlled by the servo connected to the pulley. Up and down motion of the penholder is controlled by the motion of the servo through an angle of +90 degrees and -90 degrees respectively. Two steppers are supported by two stands at the extreme end of the drawing robot. To get support between the motors and base, two ends of the robot is connected together using 4 rods (2-smooth rods, 2-screw rods). Smooth rods helps in movement of the base according to G-code and screw rod helps in overall support. In addition, there are two smooth rods to get movement of pen in y-direction.

Mechanical parts of the robot are manufactured using 3-D printer. Fig1.4 and Fig1.6 depicts the CAD generated models of the end part that holds the belt and the support for the stepper motors respectively while Fig 1.5 and Fig1.7 depicts the enclosure through which the belt runs to form the H arrangement. A prototype of the drawing robot is shown in Fig1.8.

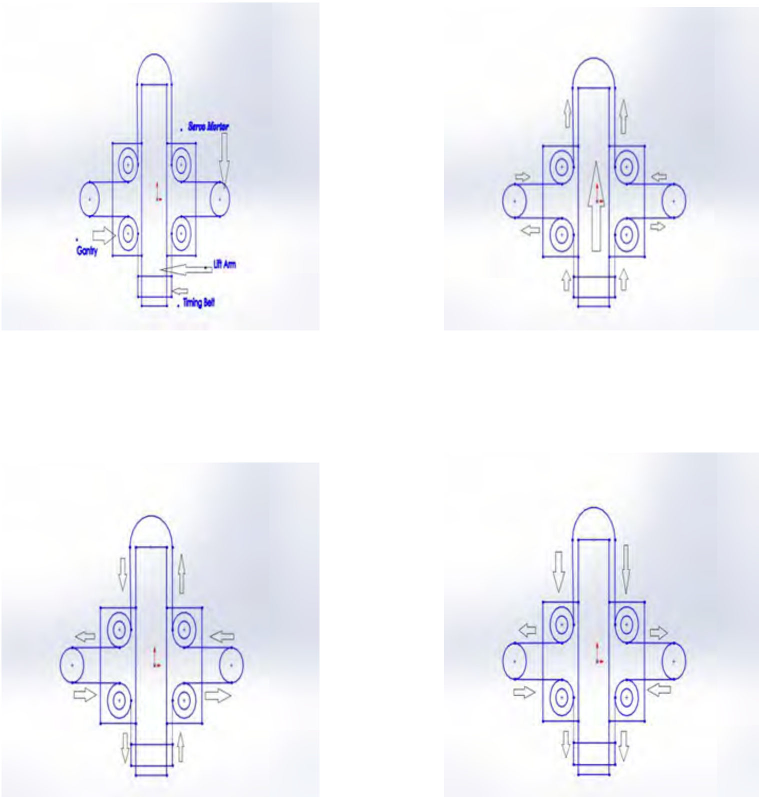

Figure 3: movement of the belt in accordance with the rotations of the motors

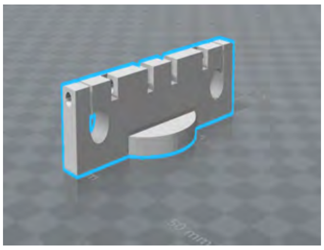

Figure 4: End part 1

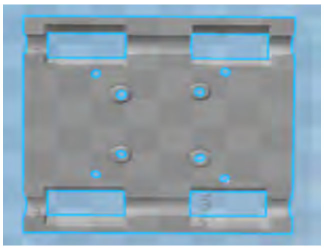

Figure 5: Centre piece 1

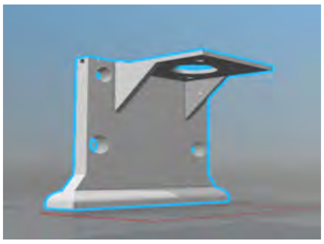

Figure 6: Stationary stand

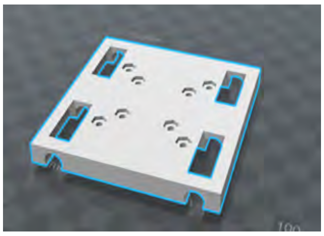

Figure 7: Centre piece 2

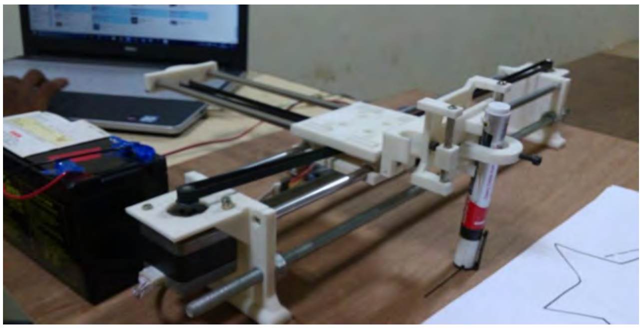

Figure 8: Drawing robot

5.2 Software Integration

We have used Inkscape to convert input image into appropriate G-code (xyz co-ordinates) and then with the help of g-code sender we send G-code to CNC machine so that motors will rotate on appropriate g-code signals. But Inkscape produces g-code appropriate for stepper motors rotation only. In order to lift pen using servo motor instead of stepper motor, we have to add extra servo library to the software extensions of Inkscape so that appropriate z-axis commands supporting servo motor will be enabled with respect to the input image. Drawing sketches in Inkscape and addition of software libraries is shown in Fig.1.9. Apart from Inkscape we have to add appropriate libraries in Arduino IDE to favor communication between Arduino IDE and g-code sender and with CNC Machine. Similarly, we have to include additional grbl settings in g-code sender to obtain the drawings. These settings can be used to control speed, acceleration, workspace per step, max travel, homing cycle etc. Fig. 1.10 shows our bot setup connected to Inkscape s/w via a laptop.

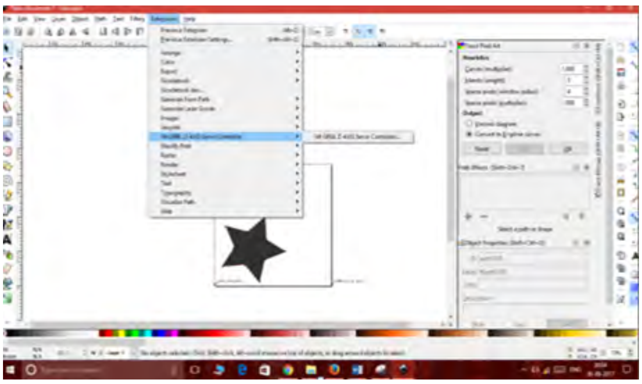

Figure 9: Inkscape Simulation Software

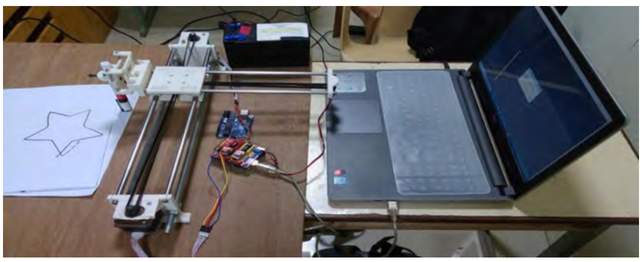

Figure 10: Drawing Robot

6 Experiment and Result

The images to be drawn by the bot is first drawn/imported using Inkscape software. Then it is converted G-code file. This can be done using the extension that we added to the Inkscape. Inkscape can also be used to control the servo motor for pen up and pen down motion. Here you can adjust the x axis and y axis speed. Also we can adjust the servo rotation angle. The grbl library generated is uploaded into Arduino. With the help of Universal G-code sender the COM port and appropriate baud rate can be selected. The drawing robot is adjusted to origin using machine control. We can also see how much time it will take to complete drawing our image as shown in Fig 1.11.

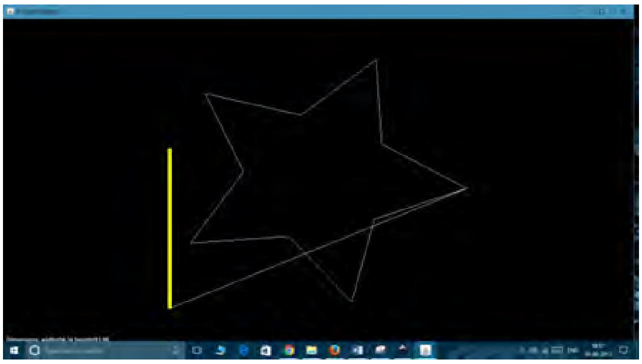

Figure 11: Visualization of drawing in UGCS

Fig 1.12 illustrates the process of drawing by the drawing robot. The image has been drawn repeatedly 6 times and response time has been noted. The response time depends on the step size and also on g-code sender. The response time showed only a slight variation for each trial. These steps has been repeated for 6 different images shown in Fig1.13.-Fig1.18 and response time shown in Table 1.

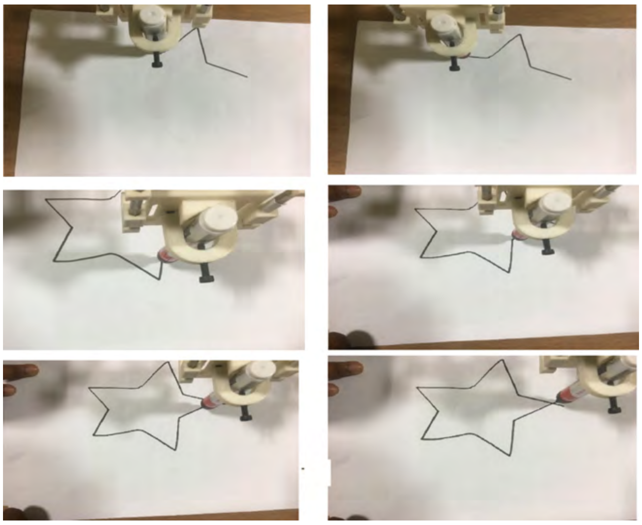

Figure 12: Drawing images on drawing pad

Figure 13: Sample-1

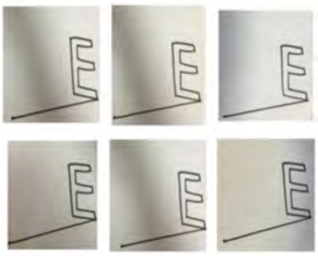

Figure 14: Sample-2

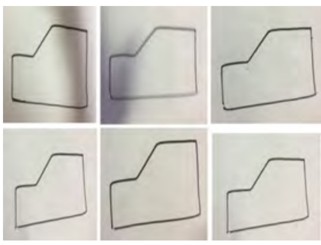

Figure 15: Sample-3

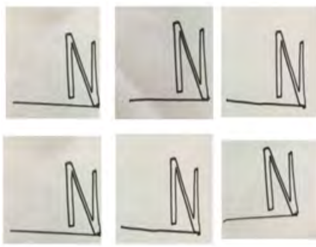

Figure 16: Sample-4

Figure 17: Sample-5

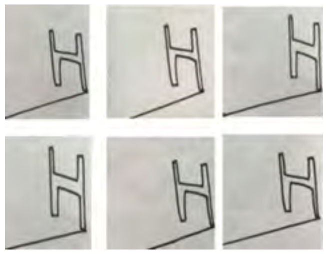

Figure 18: Sample-6

Table 1. Response Time Measurements

Trials	Response time(sec)					
	Sample 1	Sample 2	Sample 3	Sample 4	Sample 5	Sample 6
1	43	49	58	51	37	48
2	44	49	58	51	39	48
3	43	49	58	50	37	48
4	44	49	58	51	37	48
5	44	49	59	51	37	48
6	44	49	58	51	37	48

6.1 Mathematical modelling

The commandG00 directs the end effector to move to its assigned position at its maximum feed rate. Similarly, command G01 directs the end effector to the assigned position at a defined feed rate and the command G02 directs the end effector to move in clockwise direction to the assigned point at a defined feed rate. The command G03 directs it to move in anti-clockwise direction to the assigned point at defined feed rate.

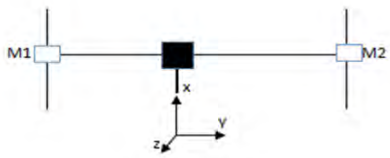

Figure 19: Coordinates of end-effector

1. $A_s = A_c + (D \times (\cos(\theta_1)))$
2. $B_s = B_c + (D \times (\sin(\theta_1)))$
3. $A_e = A_c + (D \times (\cos(\theta_2)))$
4. $B_e = B_c + (D \times (\sin(\theta_2)))$
5. $P = (A_c - (D \times (\cos(\theta_1))) - A_c$
6. $Q = (B_c - (D \times (\cos(\theta_1))) - B_c$

Here D is the radius of the arc and $_1$ and $_2$ are the angle of the start point relative to the X and Y axis respectively shown in Fig. 1.19 and Fig 1.20. A_c and B_c are the coordinates of the arc centre. A_s and B_s are the coordinate of the arc start point. A_e and B_e are the coordinate of the arc end point. P and Q are the incremental X and Y coordinates of the start point respectively. First the pen will be positioned at (A_s, B_s).Then it will draw an arc with an incremental value of P and Q.

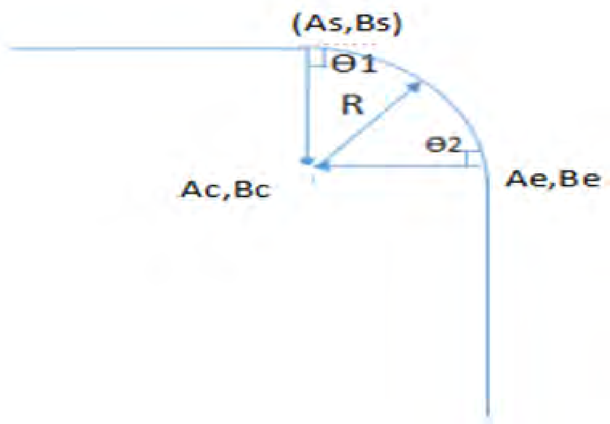

Figure 20: Coordinates of end-effector

7 Future Works

Our future work is aimed at increasing the efficiency of the robot by improving its performance. We wish to incorporate image processing tools into the robot to make the robot capable to draw photos by taking input from camera. We aim to investigate haptic technologies that could be incorporated into the robot so that the system could produce drawings simultaneously when the person makes it on a canvas. In addition, we would like to explore the capabilities of the system to function effectively to audio signals provided to it. This would enable the physical challenge students to make optimal use of this robot. The system has large scope for innovations.

8 Conclusion

This paper discusses about the design, implementation and analysis of a low cost drawing robot for educational purposes. In this research work we have presented an efficient robot with unique abilities of drawing and painting. The evaluation of the robot points to a promising end result. The outcome of the figure drawn by the robot is quite similar to that of the image/drawing given to it and the response time is quite same for every trials. The overall cost of the system excluding the microcontroller unit is about Rs 2500 only.

9 Acknowledgement

We wish to thank Almighty God for giving us the opportunity to successfully complete this venture. We would like to express our sincere gratitude to Amrita Vishwa Vidyapeetham and the Humanitarian Technology Lab for aiding us in our endeavor.

References

[1] Calinon, Sylvain, Julien Epiney, and Aude Billard. *A humanoid robot drawing human portraits*. Humanoid Robots, 2005 5th IEEE-RAS International Conference on. IEEE, 2005.

[2] Ahmed El-Barkouky, Ali Mahmoud, James Graham and Aly Farag. *An interactive educational drawing system using a humanoid robot and light polarization* .Image Processing (ICIP), 2013 20th IEEE International Conference on image processing.

[3] Chyi-Yeu Lin, Li-Wen Chuang, and Thi Thoa Mac. *Human Portrait Generation System for Robot Arm Drawing*. 2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics

[4] Shunsuke Kudoh, Shunsuke Kudoh, Miti Ruchanurucks, Katsushi Ikeuchi. *Painting Robot with Multi-Fingered Hands and Stereo Vision*. 2006 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems,Germany.

[5] A kos Hamori, Janos Lengyel, Barna Resko. *3DOF drawing robot using LEGO-NXT*. 15th International Conference on Intelligent Engineering Systems, Slovakia, 2011.

[6] Gazeau Jean-Pierre, Zeghloul SaId. *The Artist Robot: A robot drawing like a human artist*. 2012 IEEE International Conference on Industrial Technology. , vol., no., pp.486-491, 19-21 March 2012.

[7] Meng Cheng Lau, Jacky Baltes, John Anderson and Stephane Durocher, *A Portrait Drawing Robot Using a Geometric Graph Approach: Furthest Neighbour Theta-Graphs*.

[8] Mohammed Saifuddin Munna, Bijoy Kumar Tarafder, Md. Golal Robbani, Tuton Chandra Mallick, *Design and Implementation of a Drawbot Using Matlab and Arduino Mega*.International Conference on Electrical, Computer and Communication Engineering (ECCE).February 16-18,2017,Bangladesh

[9] Jeethesh Pai U, Sarath N P, Sidharth R, Anu P Kumar, Pramod S, Ganesh Udupa,*Design and Manufacture of 3D Printed Myoelectric Multi-Fingered Hand for Prosthetic Application*.2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA), Kerala, India.

[10] Rajesh Kannan Megalingam, Sricharan Boddupalli, K G S Apuroop, *Robotic Arm Control through Mimicking of Miniature Robotic arm*, 2017 International Conference on Advanced Computing and Communication System(ICACCS-2017), January 06-07,2017, Coimbatore, India.

[11] G. Gasper, M. Rahman, *Basic Hypergeometric Series*, Cambridge University Press, Cambridge (1990).

[12] M. Rosenblum, Generalized Hermite polynomials and the Bose-like oscillator calculus, In: *Operator Theory: Advances and Applications*, Birkhäuser, Basel (1994), 369-396.

[13] D.S. Moak, The q-analogue of the Laguerre polynomials, *J. Math. Anal. Appl.*, **81** (1981), 20-47.