RANGOLI
(project report)

CS 308
Embedded Real Time Systems LAB

Team 13

Rohith S - 09005052
Manohar T - 09005053
Anush V - 09005055
Manoj Azad - 09005074

Guided by
(Prof. Kavi Arya)

April 20, 2012

INTRODUCTION

- Rangoli is a folk art from India. These are decorative designs made on floors of living rooms and courtyards during Hindu festivals and are meant as sacred welcoming areas for the Hindu deities. We are trying to draw the basic rangoli with any three colours with negative feedback system (for accuracy and to be more realistic).

PROBLEM STATEMENT

- An image is given as input and the bot has to draw that image.

IMPLEMENTATION

- ***Input Image :*** A jpeg/bmp image with only red , green and blue colours are given to the scilab program (inputImageProcessing.sce)
- ***Image Processing :*** Scilab program process the pixels values for red , green and blue separately, in which it assigns value 1 to the pixel values near 255 and 0 to the remaining . It goes through the entire image and encodes the data in the format
  xaxaxaxa....byayayayay..brbc
  - X - 1(if pixel value of red is nearer to 255),
    2(if pixel value of green is nearer to 255),
    3(if pixel value of blue is nearer to 255),
    4(if no pixel values is nearer to 255)
  - Y Represents count (number of such continuous pixels) for the corresponding 'x'
    (for example x1ax2a..by1ay2a.. y1 is count for x1 and y2 is count for x2
  - R Number of rows for that image
  - C Number of columns for that image
- ***Sending the Image :*** The run-length encode image data is transferred into the bot using USB module . This data is saved in the bot untill further signal from negative feedback system is received.
- ***Drawing the image :*** Bot draws the image in to and from fashion . And if there are any part of the image with same pixel values (in a row) then it will draw a smooth line going through all of them.
- ***Ensuring Smooth flow of Colored powder :*** We have installed a vibrator motor on the body of the color funnel in order to ensure smooth flow of the colored powder whenever the lid of the funnel is opened. The vibrator motor requires a DC voltage of 5V which was taken from a power-slot provided on the bot itself just beside one of the IR sensors. We had to use a carbon resistor of 100 in series with it in order to reduce the vibration.
- ***Determination of co-ordinates of the bot :*** We have used a camera hanging on the top-centre of the arena looking for the exact position of the bot within the arena. There were two markers of colors respectively red and blue on top of the bot to detect its position as well as orientation. Depending on the resultant orientation the required movement signal is sent to the bot using a ZigBee module (refer to negativeFeedbackSystem.sce). This works as a virtual Global Positioning System (GPS) to guide the bot in the course of drawing. Different signals sent by the ZigBee module.

Figure 3: Position of Servo motors and color dispenser

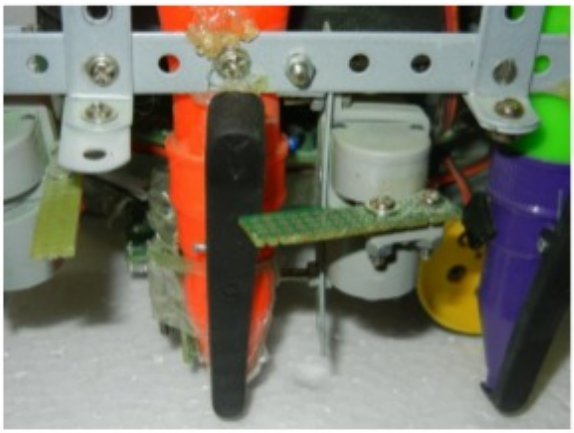

Figure 4: Funnel Blocking Mechanism using Servo Motor

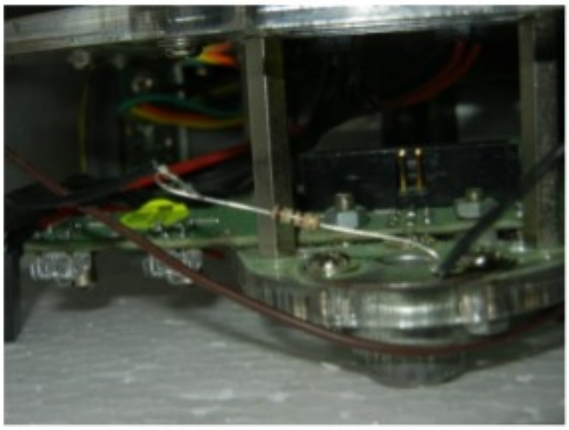

Figure 5: Resistance to control vibration of motor

Figure 6: Installation of Vibrator motor to ensure smooth flow of colored powder

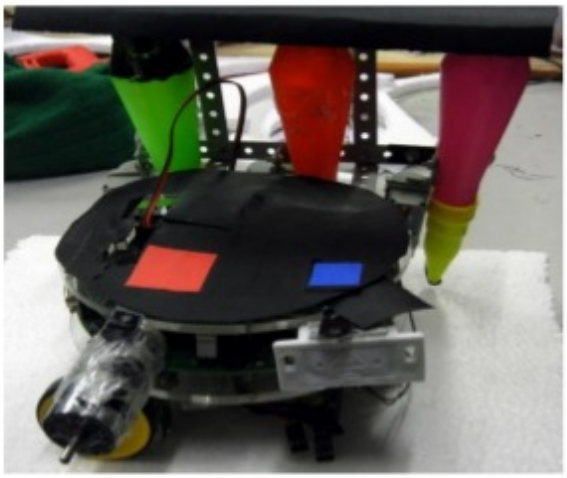

Figure 7: Position of colored markers on top of the bot

Figure 8: ZigBee module

Figure 1: Final System

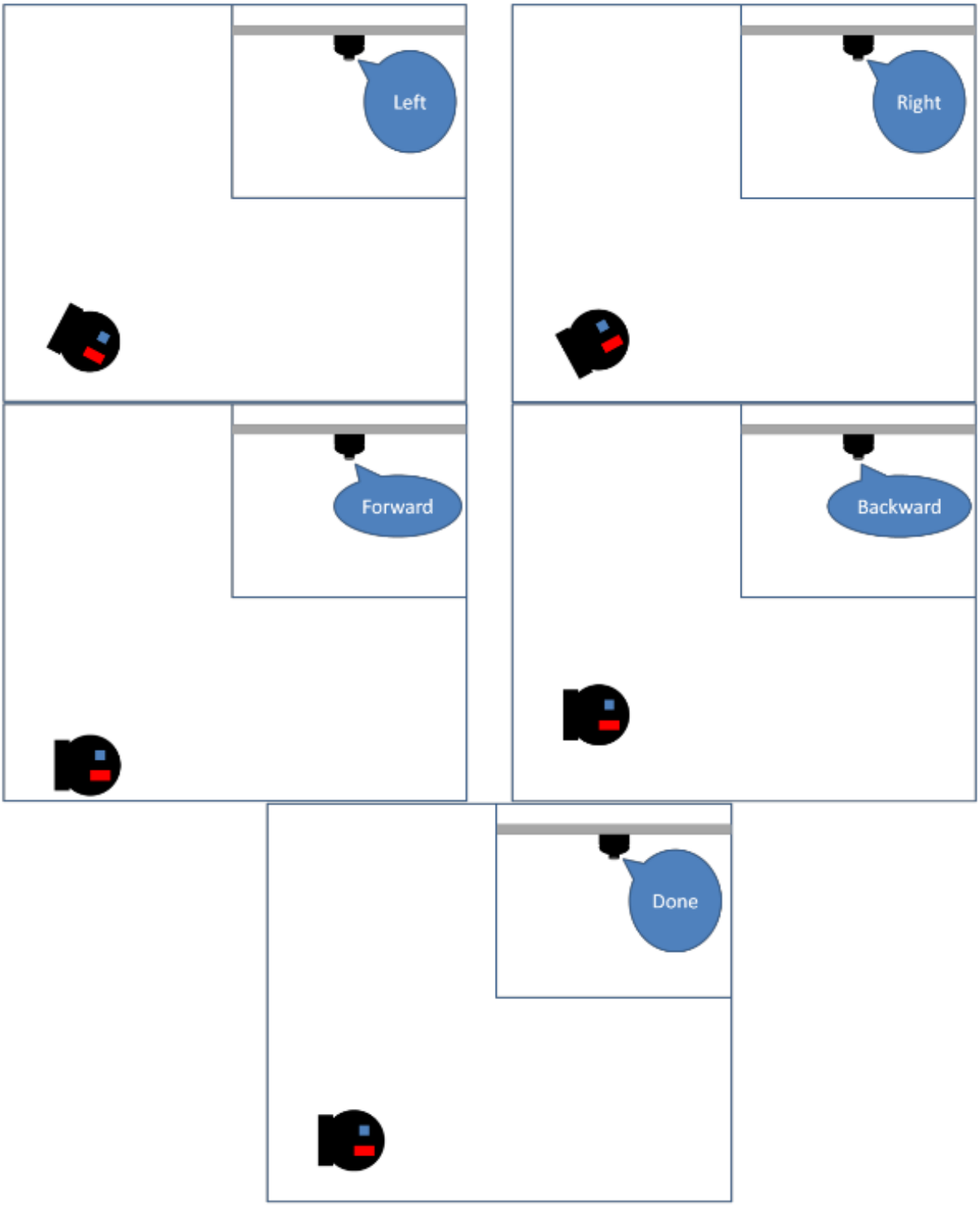

Figure 2: Negative Feedback System

PROBLEMS ENCOUNTERED

- ***Modifying Colouring device :*** the three pipes for the colouring device are separated by a distance, so the boundaries at which the three pipes can drop the colours will be different.We tried to remove this problem with the processing in negative feedback system but if we do so it will take some more time and the bot becomes more slow.So, we modified the colouring device in such a way that the three pipes point to the same point so that the boundaries of dropping the colour will be same for every pipe and so the image drawn will not be distorted.

- ***OpenCV :*** Installing the opencv and making the scilab use the libraries became difficult , and also the previous project uses the oldest version of OpenCV. OpenCV libraries are used to capture the image from the webcam . Since it works on the oldest OpenCV libraries , it was able to capture the video of only some webcam format.

- ***Scilab(4.0) and SIVP(2.0) :*** If else takes more time in scilab image processing so it wont be efficient for image processing (in which we have to keep the values near to 255 as 1 and all other values as 0).But initialisation of array is very fast in scilab so we intialized an array with zeroes and then we use only if to make the values at appropriate place 1. And there is also a bug in comparison of array values with integer values, it took time for us to figure it out. The previous project used imshow() function in the wrong way and so the SIVP is becoming unstable , we fixed that.

- ***Previous Project limitations :*** Previous project is not working for the test inputs they gave, which are sufficiently large.It works only on image with pixels of 0 and 255

FUTURE WORK

- ***Vectorized Rangoli drawing :*** In our project the bot can only draw in a rasterized fashion. But it is possible to extend this project to make the bot draw vectorized components of the input image. The only constraint of doing that is to ensure that the bot doesnt tread on the portion already drawn. In order to do that one possibility is to use a strong net with sufficiently large holes in it. In that not only the bot can move freely on top of it but also the colored powder will fall freely through the holes of the net onto the ground to create the rangoli impression and there will be no chance of going over one portion.

- ***Entirely Negative feedback Controlled System :*** Currently the entire system is basically a hybrid control system i.e., part of it is positive feedback controlled and part of it is negative feedback controlled. The controlling through camera and image processing is negative feedback controlled whereas once the bot starts drawing the image on the floor it takes the control and acts as a positive feedback system. Due to the inefficiencies of Scilab we had decided to keep like that only as frame capture rate in Scilab is very low. As it improves with later versions we can control every movement of the bot through negative feedback system to make more accurate rangolis. In that case we will not have to depend on the accuracy of the bot itself.

- ***Usage of New versions of Scilab, SIVP, OpenCV :*** Usage of the new versions of those tools ensure the use of new funtions which are efficient and also the time taken to transfer the signals will highly be reduced