ISSN (e): 2250 – 3005 || Volume, 07 || Issue, 07|| July – 2017 ||
International Journal of Computational Engineering Research (IJCER)

An Autonomous Tennis Ball Picking Robot

Prof. Dr. Rambabu Vatti, Prof. Milind Patwardhan, Prof. V.D.Pawar, Kaiwalya Belsare, Sarang Badgujar, Avinash Kangare and Sumedh Pawar
*Corresponding Author:*Prof.Dr. RambabuVatti*

ABSTRACT
Lawn Tennis often involves players hitting balls all over the tennis court. This results in accumulation of balls on the other side of court, especially during practice sessions. It is a tiresome job for the players to stop play and manually pick the balls. This paper emphasizes the development of a mobile robot that autonomously picks up the tennis balls on a lawn tennis field. Relative localization of a two wheeled differential drive robot is achieved by the means of odometry. The closed loop system features feedback from wheel encoders by taking into consideration the parameters such as payload, velocity, diameter of wheel and distance between the wheels. Implementation of advanced Image Processing techniques to find and trace tennis balls has been showcased. It also involves development of a DC motor driver with n-channel enhancement type MOSFETs for driving the motors. A robust mechanical design suited to the requirements facilitates picking the tennis balls off the field.

Index Terms: Odometry, Haar Cascading, PID Controller, PWM, Bootstrapping.

Date of Submission: 11-07-2017 Date of acceptance: 26-07-2017

I. INTRODUCTION

THErecent studies regarding the mobile robots are focusing on their traversal in partially known dynamic environments. An error in orientation or during the traversal of the robot may lead to deviations. If such deviations are hampering the capacity of the system to realize its task, then the error becomes a serious fault. An integral part of this system is the localization module. It is prone to constant source of faults that comes from odometry errors. Therefore it is mandatory to maintain the error coefficient to the minimum extent.Moreover, precise detection of tennis balls on a tennis field is equally important. The color, shadow and lighting conditions affect the detection of the ball. Image Processing algorithms such as detection through color and through shape are erroneous especially when the size of the object to be detected is small, as is observed from experimental results.This paper presents a solution to increase the availability of tennis ball picking mobile robot through odometry correction mechanisms as well as through advanced image processing techniques required for ball detection. It also presents a detailed analysis of development of customized motor driver that is able to cater high current requirement.The paper is divided into six parts. The first part being: a study of closed loop control with integrated wheel encoders for accurate position control, second part is a presentation of the 'Pixsee' mobile robot used in the application. The third part is about customised motor driver schematic developed and mathematical calculations involved. The fourth presents the image processing technique and its analysis. Thefifth part shows a practical experiment illustrating the robot's movement which involves integration of odometry and image processing algorithms, finally the Conclusion section.

II. WHEEL VELOCITY CONTROL

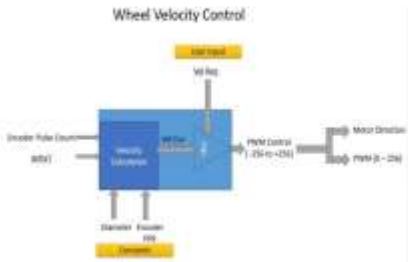

Fig. 1 Wheel Velocity Control Block Diagram.

Odometry is the mostly used navigation method for relative positioning of mobile robot because it provides good accuracy, is inexpensive, and allows very high sampling rates (Borenstein, 1998). Optical encoders mounted on both wheels feed discretized wheel increment information to the controller, which in turn used to calculate the robot's state using geometric equations. The input to the robot is the linear velocity (V_{input}), command along X, Y or angular rotation velocity and the robot drives around at the requested velocity (See Fig 1). This is possible through the odometrical mechanism showcased above. The sampled encoder counts are fed to the microcontroller along with diameter and Encoder PPR (Pulses per Revolution) as constants. A mathematical conversion from Pulse Count to Velocity is executed and instantaneous velocity (V_{abs}) is calculated. V_{abs} and V_{input} are processed in the PID controller. The output of PID controller is framed into PWM control and is applied as a correction factor to individual motors. Along the course of traversal the motor returns displacement which is responsible for finding the position of the bot from the reference position. Thus, velocity control and the localization module are realized.

III. TENNIS BALL PICKING ROBOT 'PIXSEE'

The robot 'Pixsee' is used to test the velocity control algorithms integrated with image processing algorithms in dynamic circumstances. The main components of the robot are: STM32 Nucleo F446RE (ARM Cortex M4) microcontroller development board, Encoders (for odometrical control), Bluetooth Module (for wireless communication between the robot and PC), Customized Motor Driver PCB (to drive dc motors) and 16x2 LCD (to display information). An overhead camera located on the top (to scan the field and will determine the position of the balls if present and to orient the bot towards the selected ball).

The conceived algorithm for the robot's movement has been illustrated with the help of a process flow diagram (Fig. 2)

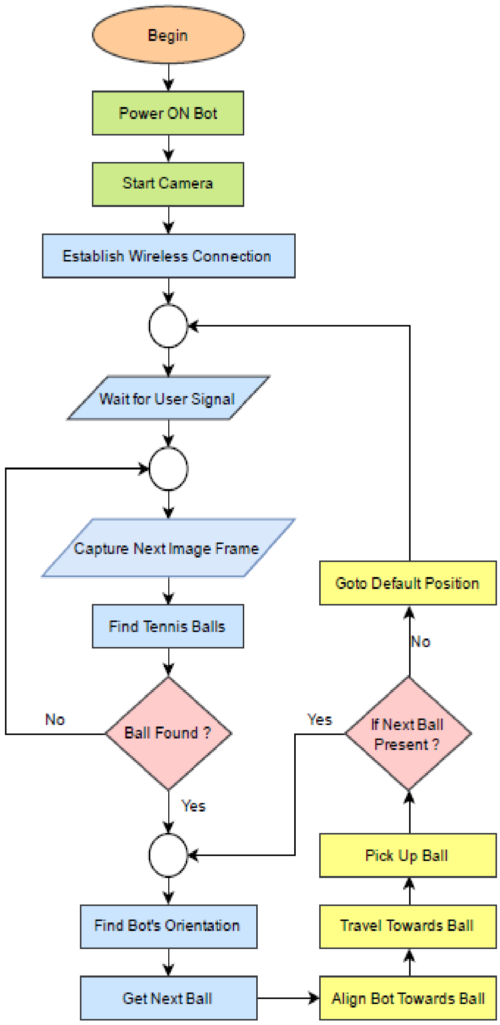

Fig. 2 Process Flow Diagram.

The system comprises of essentially two sub-systems working in synchronized manner to efficiently carry out the task of picking up tennis balls. The sub-systems namely, the robot and the Central Processor. The overhead camera is interfaced to the Central Processor, which performs image processing and sends out necessary commands to the robot. The process starts when the robot is powered on and the image processing algorithm is run on Central Processor. The robot establishes a wireless connection (Bluetooth) with the processor once it is powered on. After the system is up and robot has established connection with Central Processor, the Processor will wait for a signal from user after which the robot shall perform the ball picking activity. Once the signal is received, an image will be captured from the overhead camera of the field. Central Processor will then find tennis balls in the captured image. If no tennis ball is found then next image will be captured and the process will repeat, else the identified balls coordinates will be used for further actions.

The ball picking process will be carried out for every ball found. First and foremost, the robot's orientation with respect to x-axis of image will be calculated. Next, a ball from the list of identified balls will be considered and its orientation also will be calculated by the Central Processor. The Central Processor will then send appropriate rotation signals to the robot so that it will align itself towards the ball. After the robot is properly aligned towards the ball, the Processor will send control signals for robot's traversal towards the ball. Once robot reaches the ball's position, the ball will be picked up by the ball picking mechanism on the robot. If all identified balls are not processed, then this procedure will repeat for next ball, else Central Processor will instruct robot to go to a defined default position and will wait for next start signal from the user.

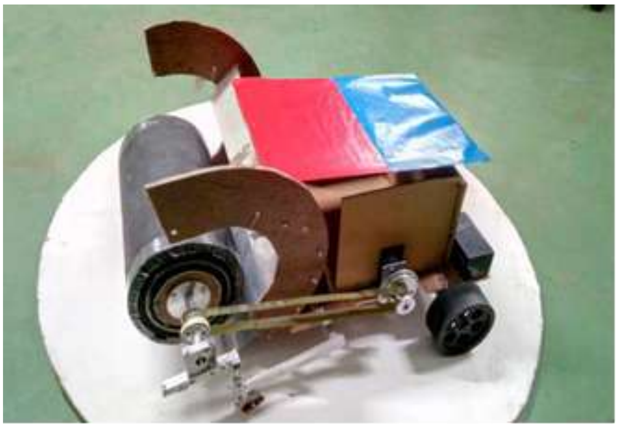

Fig. 3 Image of the Robot 'Pixsee'.

IV. MOTOR DRIVER ANALYSIS AND SCHEMATIC

The DC motors used to drive the Pixsee robot run on Full-Bridge 3.3V compatible MOSFET driver. The switching of the bridge is done by two International Rectifier IR2104 Half-Bridge driver chips. A Full-Bridge driver is created by connecting the output terminals to one side of the DC motor and using identical cicuit on the other terminal (Fig 4). The driver has complete control over the Direction and Duty Cycle of the output. IC IR2104 requires a 12V DC supply and it has the capability to drive two N -channel Enhancement type MOSFETS. Inputs to the IC include logic signal (IN) and Shutdown signal ($\overline{SD}$). The chip supplies enough drive voltage to the gates of the MOSFET, thereby ensuring both MOSFETS are never ON at the same time as this would short the high side rail to ground. The IR2104 chip has internally set dead time while providing the signals HO and LO to the high side and low side MOSFET respectively. HO and LO are Out of Phase signals.

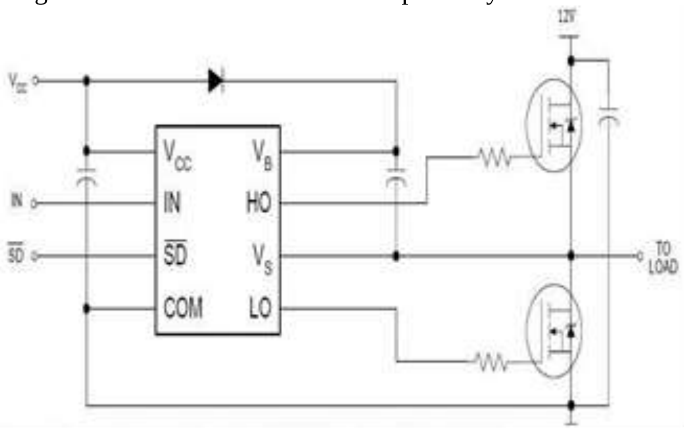

Fig. 4

Bootstrapping is required to drive the high side MOSFET. The Bootstrap circuit consists of a capacitor and diode. The diode must have fast recovery time and low voltage drop in forward bias. The value of the capacitor is calculated with the formula (eqn. 1).

$C \geq 2\,[2Q_g + I_{qbs}(max) + Q_{ls} + I_{cbs}(leak))\,] / (V_{cc} - V_f - V_{ls} - V_{min})$……..(eqn. 1)

Q_g : Gate charge of High side MOSFET
f : Frequency of operation
I_{cbs} (leak) : Bootstrap capacitor leakage current
I_{qbs} (max) : Maximum V_{bs} quiescent current
V_{cc}: Logic section voltage source
V_f: Forward voltage drop across the bootstrap diode
V_{ls} : Voltage drop across the low-side FET or load
V_{min}: Minimum voltage between V_b and V_s
Q_{ls} : Level shift charge required per cycle (Typically 5nC for 500V/600V MGDs)
I_{cbs} : 0.3*sqrt (C*V)

The high current and power rating characteristics made the IRFZ44 the initial choice for the switching device. The motors have a maximum current rating of 10A. Since drive voltage being comparatively low, MOSFET IRFZ44 can handle the power requirements of the system.

MOSFET	VTH	RDS	VDS	ID
IRFZ44	10V	0.028	60V	10A

PCB contains two driver ICs to drive the four MOSFETS along with the Bootstrapping circuit. Digital AND gate IC 7408 is used to give the PWM logic to the driver (See Fig. 4).

Fig. 5 AND gate logic.

IN1 will be driving signal to first Half-Bridge and IN2 will the driving signal for the second half bridge. PWM signal will be common to both of the signal. The drive table for MOSFET driven DC Motor Driver is given below:

Table 1

Enable	Enable2	PWM	State
1	0	1	Clockwise
0	1	1	AntiClockwise
1	1	0	Brake
0	0	0	Brake

V. IMAGE PROCESSING

An integral part which defines this system is the detection of tennis balls. The process involves detection and tracing of balls via Haar Cascading technique. When Haar Training on the balls is done, an XML file is generated which is then used to detect balls in any image.

In order to perform Haar Training, many sample images need to be gathered. Positive images are the ones on which training is performed. These images must be cropped images, containing only the desired object, in this case 'A Tennis Ball'. A total of 2000 positive images were taken for this purpose. Negative images are background images which does not contain the desired object. Similarly, 2000 negative images were taken for training. Thus in totality, 4000 images were gathered for Haar training of the tennis balls. Cropping of positive images is done, which means storage of pixel values of the tennis ball present in an image, in a text file.

Haar Training involves a set of positive and negative images which create a XML cascade file. The pixel values stored in the text file are used by the Haar Trainer Program to identify tennis balls in positive images for training them on background images. Thus, the XML file created at the end of this process can be used for detecting tennis balls in any test image. The more the number of different sample images used in the training, better are the results. The contour detection of the balls has been illustrated in Fig. 6.

Here are the steps to find the robot's orientation:

1. Obtain new image frame from top camera.
2. Apply red color detection on the original image and find red region's center point (x_1, y_1) (See Fig. 7).
3. Similarly apply blue color detection on the original image and find blue region's center point(x_2, y_2) (See Fig.7)
4. Find slope of the line joining these two points using two point formula for slope.
Slope = $(y_2-y_1)/(x_2-x_1)$.
5. Take arc tangent of the slope to get the angle of the line with respect to X-axis.

Angle = arc tan (Slope).

The obtained angle is in between -180 to 180 range which is converted to 0 to 359.

Fig. 6 Detection of balls using Haar Cascading

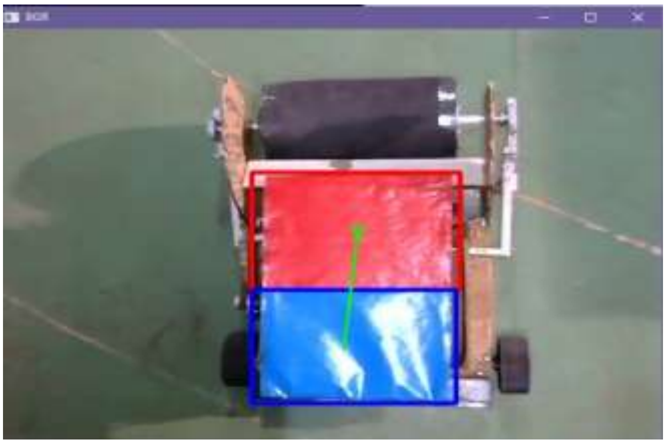

Fig. 7 Detection using robot's orientation using slope.

Fig. 8 Binary images of colors on the robot.

VI. PRACTICAL STUDY

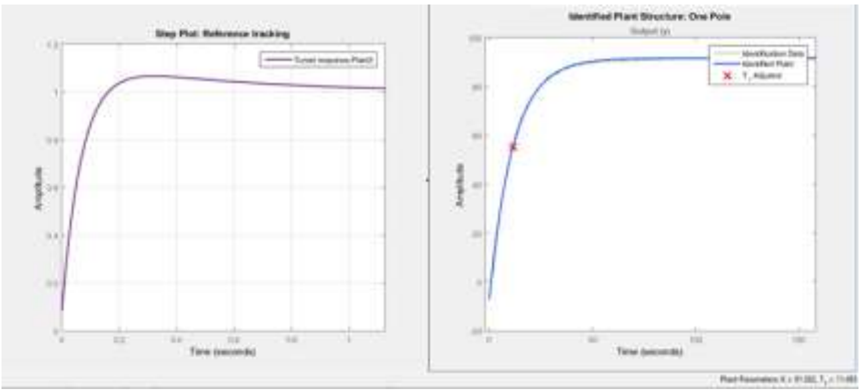

Fig. 9 Step Plot of encoder counts in MATLAB.

Results obtained from odometryare prone to get errors for uneven mechanical assembly and environment. Thus, to mitigate the errors in locomotion, encoder counts from both the wheels are analysed in MATLAB. Basis the plot,appropriate values of K_p, K_i and K_d are selected and are then fed to the PID controller. The PID controller computes the output value, which rectifies the error by introducing changes in relative velocities of wheels such that precise locomotion is achieved.

This section presents the results obtained from practical analysis of the tennis ball picking robot. At the beginning, the number of balls present in the frame of image will be detected as well the orientation of bot will be detected simultaneously. One of the ball from the detected balls will be selected at random. The bot will then orient itself towards selected ball and it will be instructed to go forward. The moment it inches closer to the ball, the roller mechanism would engulf the ball within the basket located inside the robot. Thus, this process will be repeated until all the balls are picked up.

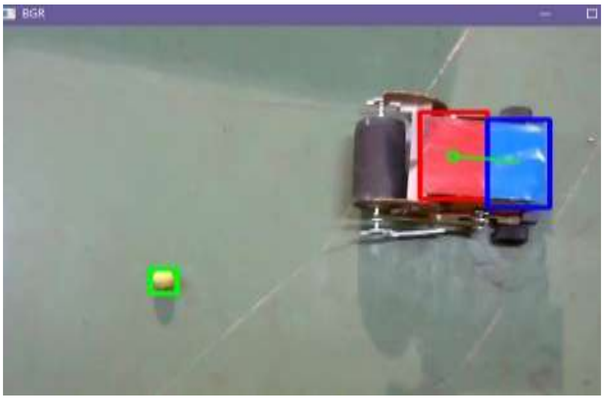

Fig. 10 Detection of ball and robot.

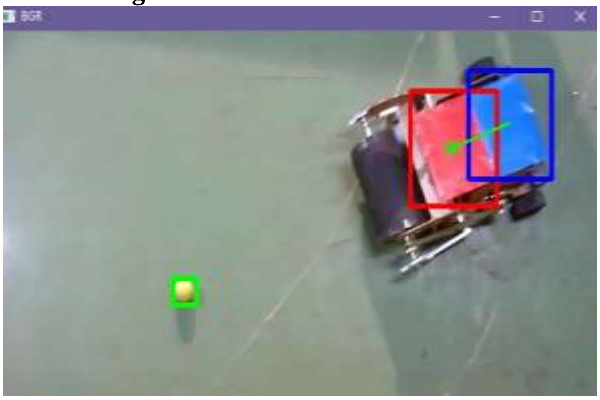

Fig. 11 Orientation of robot towards the ball.

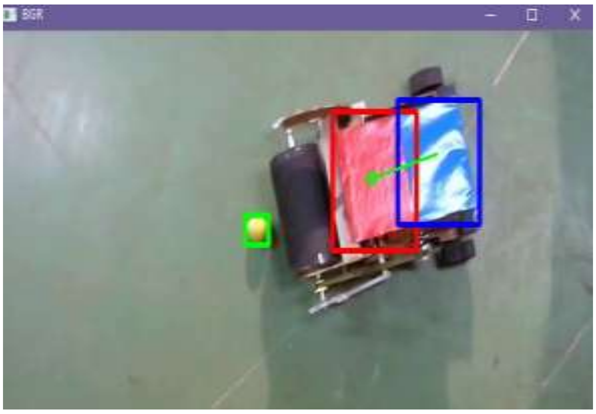

Fig. 12 Traversal of robot towards the ball.

Fig. 13 Roller mechanism engulfs the ball inside the robot.

VII. CONCLUSION

The paper presented a way to detect and pick up tennis balls autonomously with the help of a mobile robot. Availaibility of the robot was improved by implementation of algorithms such as odometry and Haar Cascading for traversal and ball detection respectively. It features successfulimplementation of proposed algorithms to increase accuracy for realizing the particular task of picking up tennis balls autonomously on a tennis field.

REFERENCES

[1] Borenstein, J., Feng, L., UMBmark: A Benchmark Test for Measuring Odometry Errors in Mobile Robots, Proceedings of the SPIE Conference on Mobile Robots, Philadelphia, October 22-26, 1995

[2] Carlson, J., Murphy, R. R., Reliability Analysis of Mobile Robots, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2003), pp. 274-281, 2003

[3] Application Note AN-6076 “Design and Application Guide of Bootstrap Circuit for. High-Voltage Gate-Drive IC”

[4] V. E.-C. Nathan Lovell, “Color Classification and Object Recognition for Robotic Soccer under Variable Illumination,” Griffith University

[5] OpenCV, “Cascade Classifier Training — OpenCV 2.4.9.0 documentation,” [Online]. Available: http://docs.opencv.org/doc/user_guide/ug_traincascade.html. [Accessed December 2014].

International Journal of Computational Engineering Research (IJCER) is UGC approved Journal with Sl. No. 4627, Journal no. 47631.

Prof.Dr. Rambabu Vatti. "An Autonomous Tennis Ball Picking Robot." International Journal of Computational Engineering Research (IJCER) 7.7 (2017): 42-49.