A comparative study of using colour sensors and Raspberry Pi camera to track colour detection

Vella Roderick
MCAST, IICT

Malta College of Arts, Science and Techonology, ICT Department
Paola, Malta

roderick.vella@mcast.edu.mt

Tonna Clive
MCAST, IICT

Malta College of Arts, Science and Techonology, ICT Department
Paola, Malta

clive.tonna.a100148@mcast.edu.mt

Micallef Lucienne
MCAST, IICT

Malta College of Arts, Science and Techonology, ICT Department
Paola, Malta

lucienne.micallef@mcast.edu.mt

Xuereb Owen
MCAST, IICT

Malta College of Arts, Science and Techonology, ICT Department
Paola, Malta

owen.xuereb.a100078@mcast.edu.mt

Abela Andrea
MCAST, IICT

Malta College of Arts, Science and Techonology, ICT Department
Paola, Malta

andrea.abela.a100229@mcast.edu.mt

Magro Brandon
MCAST, IICT

Malta College of Arts, Science and Techonology, ICT Department
Paola, Malta

brandon.magro.a100253@mcast.edu.mt

Abstract— **A light sensor works by reflecting a bright light onto the object and through IR it reflects the brightness of the colour which in turn is converted to a colour, based on its RGB values.**

This main aim of this paper is to compare results when detecting colours using the colour sensor (rev-31-1154) and the raspberry pi Camera v2.

The colour sensor (rev-31-1154) has a built-in IR (optical) and its main feature is that it allows the user to return all the colour and status data in one read command, which makes the process faster and more accurate. The PICamera is a powerful tool which not only takes still and moving images but also has the capability to detect colour based on RGB filters. To be able to detect colours using the latter, openCV library will be installed and its functions will be called and modified using python.

In order to detect the colours, the same experiment will be set up and tested using the colour sensor and the PI Camera module. To test this experiment various coloured cubes 3cms x 3cms x 3 cms will be placed in front of different coloured squares 5cms x 5cms.

The result of this experiment is to identify the best option which yields the optimum results vis-a-vis accuracy and efficiency will be integrated into our robot and used for the tournament.

Keywords—Raspberry PI 2 Model B, REV robotics, OpenCV, PI Camera, colour sensors

I. INTRODUCTION

This paper is written as part of the ECER Botball competition 2018. Part of the tournament the robot has to detect coloured pom poms, mainly red, green, yellow and blue and sort them in the respective "bins".

Computer vision and real-time image processing are needed for automated systems. Applications are being developed and implemented to include detection software, some examples include; facial recognition apps, object and colour tracking and a lot more.

The aim of this paper is to detect and identify colours and compare results when detecting colours the raspberry pi Camera v2 *(Experiment 1)* and using the colour sensor (rev-31-1154) *(Experiment 2)*.

The experiment used for this paper is to test colour detection in the "Red, Green, Yellow Zones". Testing will be carried out using the PICam and the colour sensor. Data of both components will be analysed and the most accurate and efficient component will be used for this competition.

II. LITERATURE REVIEW

The purpose of colour sensors is to record the intensity of the brightness of the colour through RGB filters. The photodiode converts the amount of light to current and the robot will in turn perform the requested instructions, such as following a particular colour. [3] carried out an experiment were they used coloured lines and instructed the robot to move along a particular line and detect other colours as "obstacles". Through continuous data from sensor the robot is capable of correcting itself accordingly to follow the "target" colour.

Given that the PI camera can capture HD video it can be easily used with OpenCV to detect colours and objects. An experiment to interpret the real world visually through computers and detect objects has been carried out by [1] through OpenCV and Raspberry PI based applications. As a result, by reading pixel values of frames captured by the RPI camera, objects were detected and traced. Similar hardware and software specifications were used in an experiment carried out by [2] the authors used RPI camera to detect skin colour,

store the colour in RGB and convert to HSI. An algorithm was developed to enhance the image in different enhancement degree.

A. *Abbreviations and Acronyms*

API - Application programming interface
Apps - Applications
GUI - graphical user interface
GPU - graphics processing unit
HD - High-definition
OpenCV - Open-Source Computer vision library
RPi - Raspberry PI
RGB - Red, green, blue
HSI - Hue, Saturation, Intensity

III. CONCEPT / DESIGN

In preparation for *Experiment 1*, the PI Camera was set up as shown in Fig 1. OpenCV library includes various methods and algorithms. One of the most useful function in this library is the *inRange(image, lower, upper)* which accepts three arguments and the range of colours to detect, usually set in RGB.

In *Experiment 2,* the robot was built using the First Global REV Robotics Kit and camera sensors were fixed to the frontal area, as shown in Fig 7. The robot awaits start instructions from a tablet and follows a set of commands to detect colour objects. Commands were implemented using Google's Blockly software.

TABLE I. PI CAM HARDWARE SPECIFICATIONS:

	PI Camera V2
	Specifications
Still Resolution	8 MP
Video Modes	1080p30, 720p60 and 640 × 480p60/90
Sensor	Sony IMX219
Sensor resolution	3280 × 2464 pixels
Sensor image area	3.68 x 2.76 mm (4.6 mm diagonal)

TABLE II. COLOUR SENSOR (REV-31-1154) HARDWARE SPECIFICATIONS::

Colour Sensor
Specifications
Supports Standard (100kHz) or High Speed (400kHz) I2C
Measures: Alpha, Red, Green, Blue, and Proximity
Auto Increment Read: Read All Color and Status Registers with One Call
Built-in IR Proximity Emitter and Detector - 5 -25cm Range

IV. IMPLENTATION

A game board similar to the one used for the ECER competition has been built. Zones have been set up for testing purposes. Fig 1 illustrates the area where testing was carried out and Fig 2 displays the 3 cubes used for testing with a size of 3 cms x 3 cms x 3cms

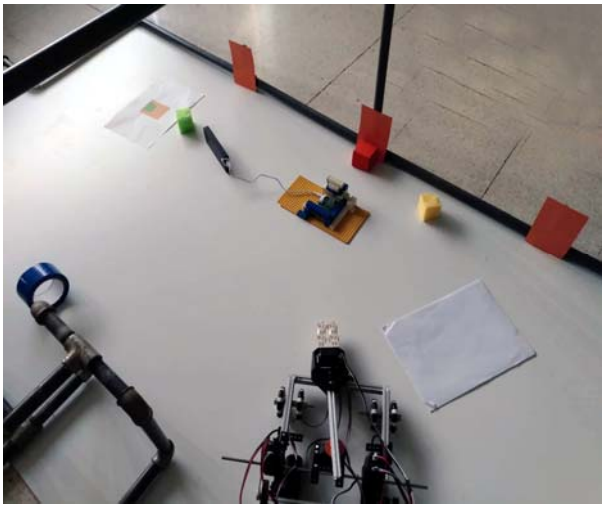

Fig. 1. The game board during testing. An orange background set up against the a vertical pole and coloured cube infront of the background.

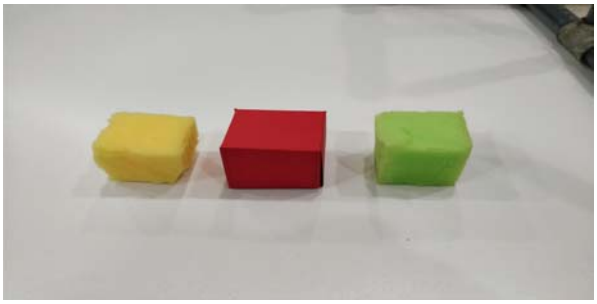

Fig. 2. Yellow, Red, Green cubes 3cms x 3cms x 3 cms

A. *Experiment 1 - RPi and PICam:*

The PICam is attached to the Raspberry PI 3 Model B as shown in the image below:

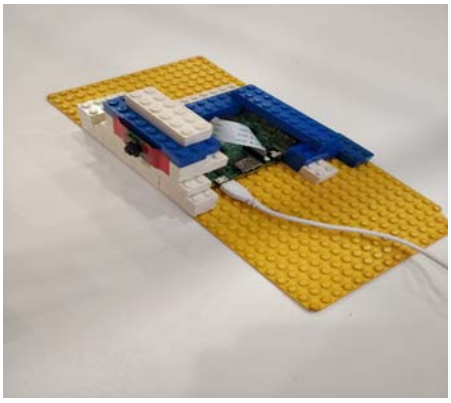

Fig. 3. The PI Camera and RPi set up in a Lego Structure to keep a steady position

PI cam has been set on the game board at a distance of 21 cms and 31 cms from the object. This distance should allow the camera to focus on the object, as shown in Fig 4.

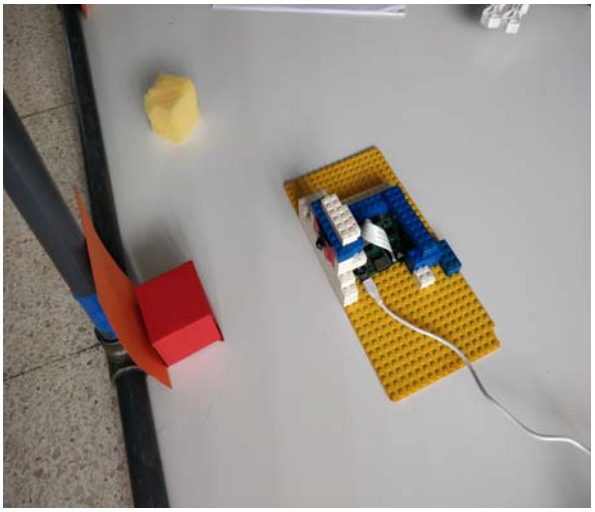

Fig. 4. The PI Camera and RPi set up at a distance from the red cube

The application was run and by using the OpenCV library, colours were detected from an image and a border was drawn around the object to identify intersection. The output of this experiment is similar to Fig 5.

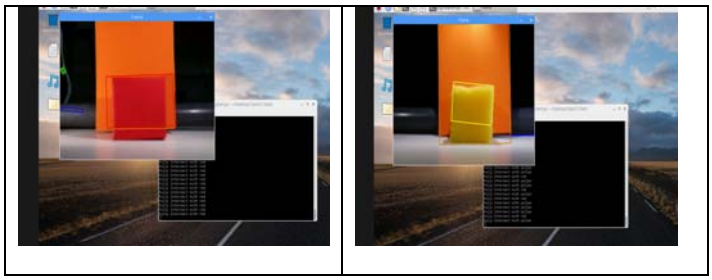

Fig. 5. Output of the *Experiment 1*

This experiment followed the algorithm as illustrated in Fig 6. The code starts by switching on the camera and allowing a few seconds to warm up. NumPy and OpenCV library are required to produce the optimum results for colour detection. NumPy creates arrays to hold the lower and upper range and a mask is created for the image. First the algorithm checks for the orange colour. If an orange colour is found, then the algorithm searches for green, red and yellow colours. Using the OpenCV function *rotatedRectangleIntersection(),* we print if there is any intersection between the two colours or not. This is done to minimise wrong readings. Through OpenCV a border around the detected colours is drawn for visual readings.

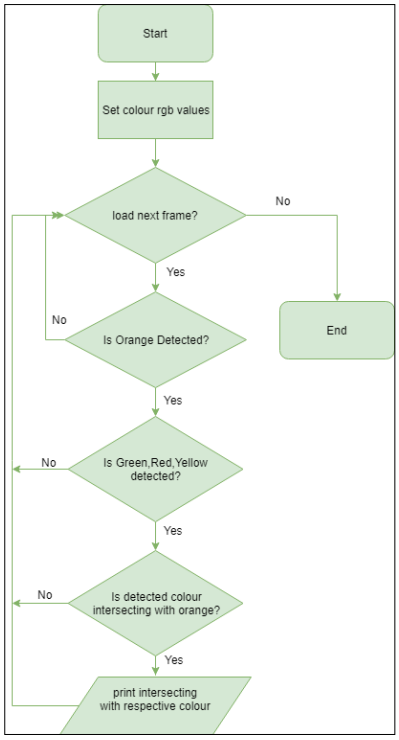

Fig. 6. Algorithm for *Experiment 1*

B. *Experiment 2- Rev Robotics Colour Sensor (rev-31-1154))*

The Rev Robotics colour sensor is an I2C sensor. It has two main purposes; to detect colour of an object and also to detect the distance or range from the object. The colour sensor is positioned at the front of the robot and connected to a port. In the configuration file of the Robot Controller adjust the color sensor range to read the colours. Fig 8 shows a screenshot of the code using Blockly.

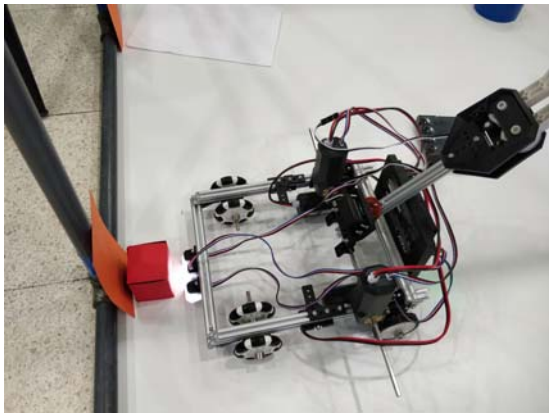

Fig 7. Robot moving closer to the object to detect colour.

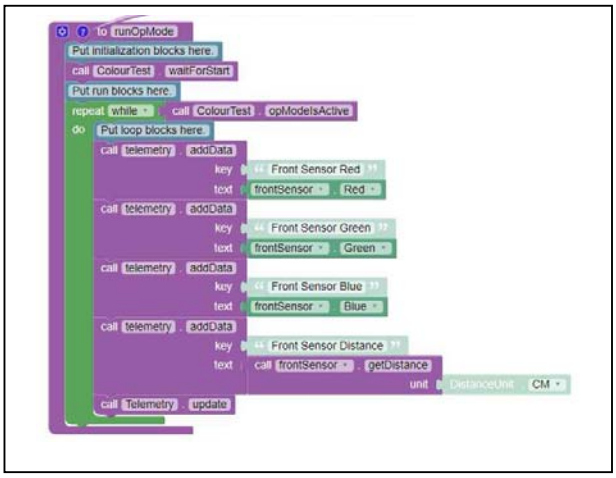

Fig 8 Screenshot for the code in Blockly to control the colour sensor

V. RESULTS

The PI Cam was placed at a distance of 21cms and 31 cms to the object and readings were taken, as shown in Fig 5. Results are depicted in Table III:

TABLE III. PI CAMERA RESULTS

	Focus Distance to object	
	21 cms	***31 cms***
Orange background & **green** cube	FAIL	FAIL
Orange background & **yellow** cube	PASS	PASS
Orange background & **red** cube	PASS	PASS

TABLE IV. COLOUR SENSOR RESULTS

	Distance sensor to object		
	1.5 cms	***2 cms***	***3 cms***
Orange background & **green** cube	PASS	PASS	PASS
Orange background & **yellow** cube	PASS	FAIL	FAIL
Orange background & **red** cube	PASS	PASS	FAIL

The colour sensor was placed at a distance of 1.5cms, 2 cms and 3 cms to the object. The reason for having the sensor so close to the object is because the light sensor is not designed for long distances.

Results show that when testing OpenCV, the closer the object is, the more difficult it is to detect the exact colour. OpenCV is very sensitive to light. For example, when intersecting green with an orange background, a yellow border is appearing and interfering the target colour. On the other hand, the outcome of the colour sensor is 100% accurate when the object is very close.

As a result, the colour sensor is proved to be more accurate and efficient in this experiment, hence, colour sensors will be used for the robot in the ECER competition.

REFERENCES

[1] C. Ivask (2015), Raspberry Pi based System for Visual Object Detection and Tracking (Bachelor's Dissertation)

[2] M. Sahani, S. K. Rout, A. K. Sharan and S. Dutta, "Real time color image enhancement with a high regard for restoration of skin color by using Raspberry Pi," 2014 International Conference on Communication and Signal Processing, Melmaruvathur, 2014, pp. 335-339.

[3] K. M. Hasan, A. Al-Nahid, K. J. Reza, S. Khatun and M. R. Basar, "Sensor based autonomous color line follower robot with obstacle avoidance," 2013 IEEE Business Engineering and Industrial Applications Colloquium (BEIAC), Langkawi, 2013, pp. 598-603.

[4] Pratyaksa D. (2015, May 24). Object Detection based on color with python 3, OpenCV 3, PI Camera and Raspberry PI 2. Retreived from: https://www.linuxcircle.com/2015/05/24/object-detection-based-on-color-with-python-3-opencv-3-and-raspberry-pi-2/ M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.