❏   1

Color Recognition

Karteek Menda
School Of Manufacturing Systems and Network, Ira A Fulton School, Arizona State University

Article Info	ABSTRACT
we	The myCobot 280 m5 is a cobot which I use in this lab to demonstrate its colour recognition capabilities. Color recognition is a process of identifying and segregating the colors under a focus of a camera (in this case). This cobot uses the HSV approach for color recognition. HSV stands for Hue, Saturation, and Value. These can be very much intuitive for particular applications, such as recognizing colors regardless of variations in brightness.

Corresponding Author:

Karteek Menda
School Of Manufacturing Systems and Network, Arizona State University
Tempe, Arizona, United States
Email: kmenda@asu.edu

1. INTRODUCTION

In this instance, colour localization will be accomplished via the eye-to-hand method of OpenCV combined with Python. In essence, it recognises the colours of blocks that are in the camera's field of vision. Once the colour of the blocks has been determined, put the matching-hued blocks into the matching containers.

A myCobot 280 m5, a camera module, a suction device, a USB cable, a few jumper cables, and a camera square are needed for this experiment. To focus on the colour detecting field, the camera needs to be adjusted. It's crucial to keep in mind, though, that the camera needs to record the whole colour detection square. You then need to be able to launch the application.

2. METHOD

Setup:

Installing Pymycobot and the necessary Python software is the first step. You must install myCobot for linux after installing Python. Next, you must download and burn the transponder while the base is linked to the display. Next, open myStudio, download Atom, and flash it. To stop the robot from hitting the camera module, simply install the required software and then set all motor settings to 0 degrees. After that, use the calibration code to calibrate.

Steps:

1. You must first install and dual-boot Ubuntu on your laptop. Furthermore, enough storage space must be available for the necessary software to be installed and run.
2. Install all necessary programmes, including Visual Studio Code, Python, and OpenCV.
3. Launch the programme in Visual Studio Code after installing the prerequisites. It is now necessary to install additional packages, which can be done with 'pip'.
4. Run the code in the aikit_color.py file after installing the necessary packages.
5. The Red colour block and Yellow colour block gets identified and their bounding boxes are taken into considerations and the centroids are calculated for those two blocks. So, the objective of this exercise is to identify the Yellow colour block and picking it up and place it on the Red block.

6. Once the end goal is achieved, the exercise is demonstrated to the concerned person in the lab.

3.   RESULTS AND DISCUSSION

Please be aware that running this programme on Ubuntu may cause you to encounter issues. Thus, to launch the programme in the terminal, type "sudo". Furthermore, if the programme is running, the front camera may open. This is as a result of the cameras having pre-numbered labels to guarantee module usage. The number of sections can be changed as necessary. Especially the cap number of the camera (in my case it is set to -1).

Once my code is run, a new pop-up windows will appear with the camera on. The blocks (Red, Yellow) needs to be kept on the board under the camera. The camera module gets opened and starts detecting the block colors. The red block and yellow block both gets detected. The red block gets detected first and the bounding boxes coordinates gets identified. Then the yellow block bounding boxes can be detected and the coordinates can be detected.

So, I scripted the small part of the code in such a way that the yellow block gets sucked up by the suction cup and the yellow block will be placed on red block. That brings a close to the task.

To sum up, this experiment clarifies how the camera module enables the robot to independently travel to the block. I understood the color identification technique and how OpenCV works better.
Cobot Serial Number : ER28001202200478
Video Link : (https://www.youtube.com/watch?v=dc25zBDw63k)
Code : aikit_color.py

4.   CONCLUSION

The robot advances to the yellow block and picks up the yellow block and put it on red block. This demonstrates the cobot's capabilities by using the color recognition. Thus the capabilities of mycobot can be used to the complete extent in the real time by using this capability.

REFERENCES

The main references are
[1] https://www.hackster.io/Elephant-Robotics-Official/newly-upgraded-and-user-friendly-ai-kit-2023-a23669
[2] https://docs.elephantrobotics.com/docs/gitbook-en/2-serialproduct/2.10-AIkit2023en_320/3-color_recognition.html
[3] https://docs.elephantrobotics.com/docs/gitbook-en/2-serialproduct/2.10-AIkit2023en_320/2-knowledge_preparation.html
[4] https://docs.elephantrobotics.com/docs/gitbook-en/2-serialproduct/2.10-AIkit2023en_320/introduce.html

BIOGRAPHY OF AUTHOR

Karteek Menda is a Robotics GRAD Student at ASU, Ex - Machine Learning Engineer, and a Machine Learning Blogger. Experienced data science professional with extensive knowledge of building data-intensive applications and overcoming complex architectural and scalability challenges. Proficient in predictive modelling, data processing, and data mining algorithms, as well as scripting languages, including Python and C++. A passion for Artificial Intelligence coupled with a comprehensive understanding of machine learning concepts and other related technologies. Through the design, development, testing, and deployment of highly adaptive diverse services, able to transform business and functional qualifications into substantial deliverables. I can be contacted at email: kmenda@asu.edu