IAES International Journal of Robotics and Automation (IJRA)
Vol. 99, No. 1, Month 2099, pp. 1~1x
ISSN: 2722-2586, DOI: 10.11591/ijra.v99i1.pp1-1x

Object Detection

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

Article Info

Article history:

Received December 1, 2023
Revised December 1, 2023

Keywords:

Pymycobot
Atom
YOLO
Object Detection

ABSTRACT

The myCobot 280 m5 is a cobot which I use in this lab to demonstrate its object detection capabilities. Finding and locating things in pictures or videos is known as object detection in computer vision. The objective is to encircle interesting items with bounding boxes and perhaps categorise them. Applications for object detection are numerous and include medical image processing, surveillance, and self-driving cars. Object detection is an essential technology. You Only Look Once (YOLO) is an object recognition algorithm that works in real-time, splitting an image into grids and projecting probabilities of classes and bounding boxes into each grid cell.

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, object detection will be accomplished via the object detection method of OpenCV combined with Python. In essence, it detects the objects that are in the camera's field of vision. Once the objects under the camera area have been determined, put the object into the matching containers. In this particular case, the eye-to-hand mode is utilised in addition to the camera for taking pictures, OpenCV is employed to load the Yolov5 model data, image block recognition, and image block position within the recognition area. The space coordinate location of the object with regard to the robot arm is determined through the relevant points. Set the robot arm to a series of related duties, and put the determined object in the right spot.

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 object detecting field, the camera needs to be adjusted. It's crucial to keep in mind, though, that the camera needs to record the whole aruco detection square. You then need to be able to launch the application.

The main objective is that to identify the different objects and put them into separate bin. So, in real time, if I need to separate the objects based on their identity. Segregate them and place them in separate places.

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.

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 yolov5_img.py file after installing the necessary packages.
5. Put the object “Vase” under the camera area and then take the picture and crop the image and the algorithm detects the image as “Vase”, picks up the “Vase” and keep it in the mentioned place (in our case it is bin). This goes on with the other two objects as well. The same process gets repeated and puts the respective object in the concerned bin. This process keeps on repeating for “n” times, and the different objects are thus segregated into their respective bins.
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 programme is operating appropriately if the camera window opens successfully. As of right now, you can follow the instructions on the terminal to position an identifiable object in the detection region, snap a picture of it, and intercept it. The robot arm is prepared when you see "OK" in the command console, which indicates that yolov5 has found it and grabbed it. To identify the following image, press the spacebar after taking the previous one. Once that is done, the robot advances to the object and picks it up and puts it in separate bin and this process repeats for different objects and for multiple iterations. So, I scripted the small part of the code in such a way that the each of the different object gets sucked up by the suction cup and the respective objects will be placed in respective bin. 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, and picks up the object and place it in bins. I understood the object detection technique and how OpenCV works better.
Cobot Serial Number : ER28001202200478
Video Link : (https://www.youtube.com/watch?v=xcu1wwhQOMk**)**
Code : yolov5_img.py

4. CONCLUSION

The robot advances to each of the object after identifying and picks up the respective block and put it in the respective bin. This demonstrates the cobot’s capabilities by using the object detection. Thus the capabilities of mycobot can be used to the complete extent in the real time by using this capability.

ACKNOWLEDGEMENTS

I thank Prof Redkar, teaching assistants for taking the demonstrations.

REFERENCES

The main references are
[1]https://docs.elephantrobotics.com/docs/gitbook-en/2-serialproduct/2.10-AIkit2023en_320/7-yolov5_recognition.html
[2] https://docs.elephantrobotics.com/docs/gitbook-en/

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