Robothon 2023- Team RoboTechX Middlesex University Dubai

Judhi Prasetyo, Ziad Burhani, James Brazier, Sameer Kishore

Introduction

Dubai has the largest e-waste recycling facility with processing capacity of 100,000 tonnes of total integrated waste per year, of which 39,000 tonnes is e-waste. However, as reported by CNN (https://edition.cnn.com/2021/10/07/middleeast/enviroserve-e-waste-dubai-spc-intl/index.html) some of the process in disassembling the e-waste are still done manually by human.

We aim to automate the disassembling process with robot to save the labor from being exposed to toxic and hazardous materials.

Solution Overview

The solution provided by Team RoboTechX features simplicity, cost-effective and high accuracy in handling the e-waste handling automation. We use EPSON VT6 6-axis robot, controlled by Python script over a TCP/IP network. Using a commonly found USB Webcam attached to the robot's arm, the Python script locates the world coordinate points on the task board and map it to local coordinate of the board. Subsequently the script will coordinate the motion of EPSON VT-6 robot and its custom 3D printed gripper to perform the tasks by sending the coordinate points and gripper opening control. The camera is also used to capture the display on the M5 device and determine the amount of distance the slider needs to travel.

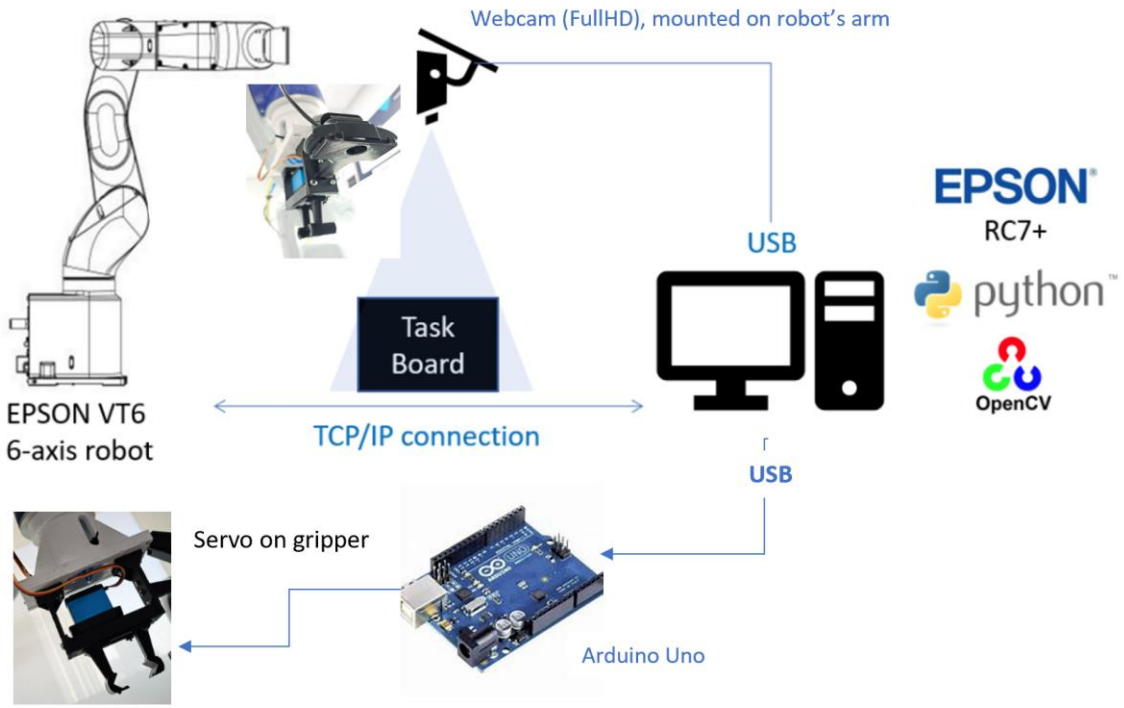

Figure-1: system connectivity

The Computer Vision System (CVS) is written in Python and utilizes OpenCV, an open source computer vision and machine learning software library. The CVS is used for two purposes: task board localization and determining slider position from the M5 display.

Localization: the image of the e-waste item is taken using a web camera with Full High Definition (1290x1080 pixel) resolution. The camera is mounted on the robot's arm above the gripper. Based on the image taken, the CVS will calculate the orientation of the e-waste and translates the pixels into real world coordinate to guide the robot manipulator. On the task board given for Robothon Grand Challenge, there are several features on the board that can be used to mark the position and orientation. The blue button and the door knob are chosen as the markers as they are unique and far apart. Once the markers' position is obtained, the values will be passed to the robot via TCP/IP network.

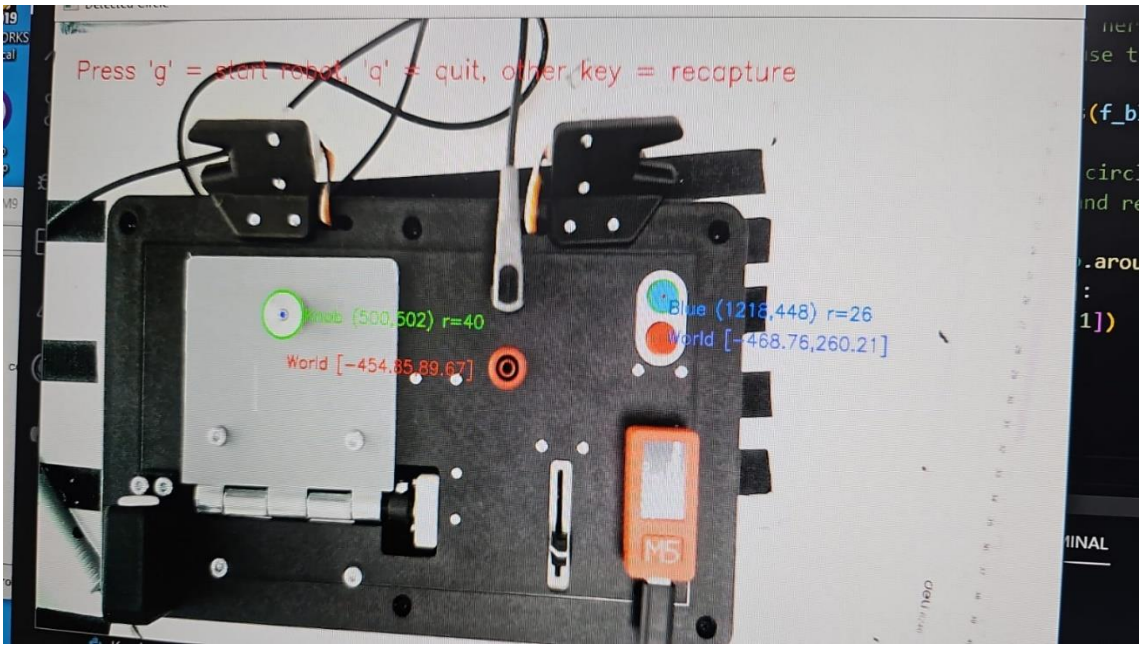

Figure-2: Localization of the task board

Slider (fader) position detection: first, the robot arm will move above the M5 device and take picture of its display. The Python script will determine the position of the arrow relative to the width of the screen and calculate the distance that the fader potentiometer has to travel from its original position. The position will be sent via TPC/IP network to the robot arm. Once the first process completed, the arm will move to the same position to take the second picture and determine the second arrow, hence the second distance can be calculated.

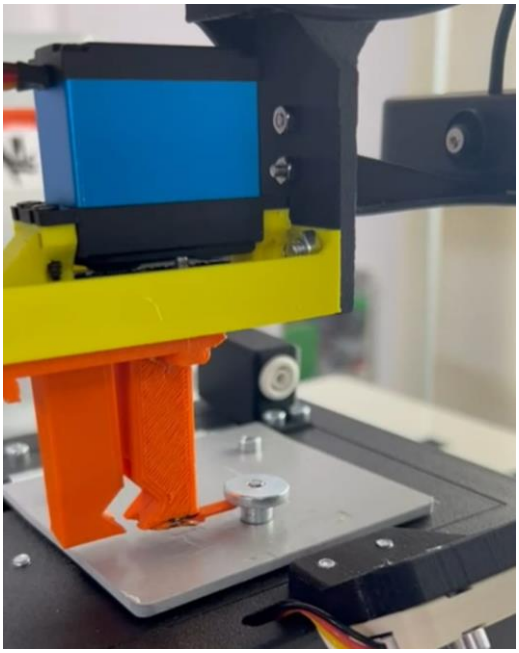

Figure-3: Custom gripper with side pick for opening door and web-camera mount

Equipment list

- EPSON VT6 6-axis robot (6Kg payload, 900mm reach)
- Arduino Uno for controlling Servo motor using PWM, controlled by Python via USB
- Servo motor (20kg) for gripper
- Custom 3D printed gripper
- Webcam Full HD

Total hardware cost of the solution including the robot and all components listed above is estimated at €15,750.

Software dependency list

- Python version 3 in VSCode IDE for writing Python code
- OpenCV Python
- EPSON RC7+ SPEL
- Arduino IDE, C++ for controlling gripper
- Github Desktop to synchronize the code to Github repository
- Custom Python scripts and SPEL program
    - R23_main.py - main python script
    - R23_Arduino.ino – Arduino sketch for controlling the gripper servo
    - main.prg – ESPON SPEL program to control the robot
    - Settings.py – module for settings and variables setup
    - Map_coord.py – module for coordinate translation
    - Slider_task.py – module for detecting target slider position
    - SendToEpson.py – module for communicating with EPSON robot
    - arduino_communication.py – module for communicating with Arduino

Quick start guide

- Place the task board on the table, make sure the Velcro tapes hold it firmly
- Run EPSON RC7+ SPEL program by pressing F5 and click on Start button
- Run the R23_main.py Python script by pressing “Play” button on VSCode
- Select the sequence of tasks to run
- Observe the captured image of the task board, make sure that the blue button and door knob are detected clearly. Press “g” to start the robot or any other key to recapture if better detection accuracy is required.
- Robot will go through the task sequence.