

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO  
BỘ MÔN TỰ ĐỘNG ĐIỀU KHIỂN



**HCMUTE**

**ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI: THIẾT KẾ MÔ HÌNH ROBOT DELTA ỨNG  
DỤNG PHÂN LOẠI RÁC THẢI**

**GVHD: TS. Trần Đức Thiện**

**Sinh viên thực hiện:**

**Nguyễn Nhật Trường 19151187**

**Nguyễn Trần Bửu Thạch 19151171**

**Khóa 2019 – 2023**

**Ngành Công Nghệ Kỹ Thuật Điều Khiển và Tự Động Hóa**

**TP. HỒ CHÍ MINH, tháng 07 năm 2023**

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO  
BỘ MÔN TỰ ĐỘNG ĐIỀU KHIỂN



**HCMUTE**

**ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI: THIẾT KẾ MÔ HÌNH ROBOT DELTA ỨNG  
DỤNG PHÂN LOẠI RÁC THẢI**

**GVHD: TS. Trần Đức Thiện**

**Sinh viên thực hiện:**

**Nguyễn Nhật Trường 19151187**

**Nguyễn Trần Bửu Thạch 19151171**

**Khóa 2019 – 2023**

**Ngành Công Nghệ Kỹ Thuật Điều Khiển và Tự Động Hóa**

**TP. HỒ CHÍ MINH, tháng 07 năm 2023**

Tp. Hồ Chí Minh, ngày ... tháng ... năm 2023

## NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Sinh viên thực hiện đề tài:

- |                          |                |                |
|--------------------------|----------------|----------------|
| 1. Nguyễn Nhật Trường    | MSSV: 19151187 | Lớp: 19151CL3B |
| 2. Nguyễn Trần Bửu Thạch | MSSV: 19151171 | Lớp: 19151CL3B |

Ngành: Công Nghệ Kỹ Thuật Điều Khiển và Tự Động Hóa

Giảng viên hướng dẫn: TS. Trần Đức Thiện ĐT: 0988862588

Ngày nhận đề tài: 06/02/2023

Ngày nộp đề tài: 30/06/2023

### 1. *Tên đề tài:*

Thiết kế mô hình Robot Delta ứng dụng phân loại rác thải.

### 2. *Các số liệu, tài liệu ban đầu:*

Tìm hiểu thiết kế mô hình Robot Delta.

Định hình đối tượng phân loại: chai nhựa và lon.

Tìm hiểu về máy học, YOLO.

### 3. *Nội dung thực hiện đề tài*

- Thiết kế mô hình Robot Delta trên phần mềm Solidworks 2021.
- Tính toán động học cho robot, khảo sát không gian làm việc.
- Ứng dụng thuật toán YOLOv5 trong hệ thống phân loại rác (chai nhựa và lon) theo thời gian thực.
- Thi công mô hình Robot Delta theo bản vẽ đã thiết kế trên phần mềm Solidworks, lắp đặt tủ điện gọn gàng và an toàn theo quy chuẩn, hệ thống camera và hệ thống cơ cấu chấp hành hút thả vật.
- Lập trình điều khiển động cơ cho Robot và các thiết bị bằng phần mềm Arduino IDE, thiết kế giao diện người dùng và xử lý thuật toán phân loại trên phần mềm Visual Studio Code.

- Kiểm chứng động học, thống kê số liệu đánh giá tính chính xác của hệ thống Robot Delta phân loại rác (chai nhựa và lon) trong nhiều lần.

#### **4. Các sản phẩm dự kiến:**

- Hệ thống phân loại rác tự động cho 2 đối tượng (Robot Delta, thị giác máy tính, băng tải).

- Tủ điện thiết bị.
- Chương trình, giao diện điều khiển.
- Quyền bảo cáo đồ án tốt nghiệp.
- Poster giới thiệu đồ án.

**5. Ngôn ngữ trình bày: Bản báo cáo: Tiếng Anh  Tiếng Việt**   
**Trình bày bảo vệ: Tiếng Anh  Tiếng Việt**

**TRƯỞNG NGÀNH**

(Ký, ghi rõ họ tên)

**GIẢNG VIÊN HƯỚNG DẪN**

(ký, ghi rõ họ tên)

Tp. Hồ Chí Minh, ngày ... tháng ... năm 2023

## LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN TỐT NGHIỆP

Sinh viên thực hiện đề tài:

- |                          |                |                |
|--------------------------|----------------|----------------|
| 1. Nguyễn Nhật Trường    | MSSV: 19151187 | Lớp: 19151CL3B |
| 2. Nguyễn Trần Bảo Thạch | MSSV: 19151171 | Lớp: 19151CL3B |

## Ngành: Công Nghệ Kỹ Thuật Điều Khiển và Tự Động Hóa

Giảng viên hướng dẫn: TS. Trần Đức Thiện ĐT: 0988862588

### 1. *Tên đê tài:*

## Thiết kế mô hình Robot Delta ứng dụng phân loại rác thải

## 2. Kế hoạch thực hiện đồ án

Tuần/ ngày	Nội dung	Xác nhận GVHD
06/02/2023 (1 tuần)	<ul style="list-style-type: none"> <li>- Tìm hiểu tổng quan đề tài.</li> <li>- Tìm hiểu về các Robot phân loại rác, ưu và nhược điểm của Robot Delta trong phân loại rác.</li> <li>- Tìm hiểu về động cơ Step và Arduino.</li> </ul>	
13/02/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Thiết kế mô hình Robot Delta trên phần mềm Solidworks.</li> <li>- Lựa chọn động cơ.</li> <li>- Tìm hiểu về động học của Robot Delta.</li> <li>- Tìm hiểu về xử lý ảnh, computer vision.</li> </ul>	
27/02/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Tiếp tục hoàn thiện thiết kế mô hình trên phần mềm Solidworks.</li> <li>- Lập trình động học thuận nghịch trên phần mềm Arduino IDE.</li> <li>- Khảo sát mô phỏng không gian làm việc trên Maltab.</li> </ul>	

13/03/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Thi công mô hình.</li> <li>- Tìm hiểu, lựa chọn thiết bị cho mô hình Robot.</li> </ul>	
27/03/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Thi công mô hình.</li> <li>- Điều khiển động học thuận nghịch Robot.</li> <li>- Tìm hiểu về mạng thần kinh nơ ron và thư viện YOLOv5.</li> <li>- Tìm hiểu phần mềm huấn luyện (Roboflow, Google colab).</li> </ul>	
10/04/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Xây dựng tập dữ liệu huấn luyện trên Roboflow.</li> <li>- Huấn luyện mạng YOLOv5 trên Google Colab.</li> </ul>	
24/04/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Kiểm chứng kết quả huấn luyện với ảnh đầu vào và camera.</li> <li>- Nhúng thư viện đã huấn luyện về máy tính.</li> </ul>	
08/05/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Xác định vị trí của vật trên hệ tọa độ ảnh.</li> <li>- Chuyển đổi vị trí của vật từ hệ tọa ảnh về hệ tọa độ Robot.</li> <li>- Xây dựng giải thuật điều khiển hành trình Robot phân loại.</li> <li>- Lắp đặt cơ cấu khí nén.</li> </ul>	
22/05/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Chạy thực nghiệm.</li> <li>- Đì dây hệ thống và lắp đặt tủ điện.</li> <li>- Thiết kế giao diện người dùng trên phần mềm Visual Studio Code.</li> </ul>	
5/06/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Đánh giá, kiểm chứng động học thuận, nghịch.</li> <li>- Đánh giá kết quả phân loại rác với hai trường hợp ánh sáng.</li> <li>- Cải thiện chất lượng mạng (Tăng số lượng ảnh đầu vào).</li> </ul>	
19/06/2023 (2 tuần)	<ul style="list-style-type: none"> <li>- Viết quyển báo cáo đồ án.</li> <li>- Hoàn thành Poster, video giới thiệu đồ án và PowerPoint trình bày.</li> </ul>	

## GIẢNG VIÊN HƯỚNG DẪN

(ký, ghi rõ họ tên)

**TRƯỜNG ĐH SPKT TP. HỒ CHÍ MINH**  
**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**  
**BỘ MÔN TỰ ĐỘNG ĐIỀU KHIỂN**

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  
ĐỘC LẬP - TỰ DO - HẠNH PHÚC**

Tp. Hồ Chí Minh, ngày ... tháng ... năm 2023

# PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

Sinh viên thực hiện đề tài:

1. Nguyễn Nhật Trường      MSSV: 19151187      Lớp: 19151CL3B  
2. Nguyễn Trần Bảo Thạch      MSSV: 19151171      Lớp: 19151CL3B

## Ngành: Công Nghệ Kỹ Thuật Điều Khiển và Tự Động Hóa

## Tên đề tài: Thiết kế mô hình Robot Delta ứng dụng phân loại rác thải

Họ và tên giảng viên hướng dẫn: TS. Trần Đức Thiện

## Ý KIẾN NHẬN XÉT

- ## 1. Về nội dung đề tài & khối lượng thực hiện

---

---

- ## 2. Ưu điểm

---

---

- ### 3. Khuyết điểm

---

---

- #### 4. Đề nghị cho bảo vệ hay không?

5. Đánh giá loại: \_\_\_\_\_  
6. Điểm: \_\_\_\_\_ (Bằng chữ \_\_\_\_\_)

Tp. Hồ Chí Minh, ngày tháng năm 2023

Giảng viên hướng dẫn

(ky ghi rõ ho tên)

TRƯỜNG ĐH SPKT TP. HỒ CHÍ MINH  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO  
BỘ MÔN TỰ ĐỘNG ĐIỀU KHIỂN

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  
ĐỘC LẬP - TỰ DO - HẠNH PHÚC  
----000----

Tp. Hồ Chí Minh, ngày ... tháng ... năm 2023

### **LỜI CAM ĐOAN**

Nhóm sinh viên cam kết đây tài này là do nhóm thực hiện dựa vào một số tài liệu trước đó và không sao chép từ tài liệu hay công trình đã có trước đó.

Tp.HCM, ngày ... tháng ... năm 2023  
Sinh viên thực hiện  
(Ký và ghi rõ họ tên)

## LỜI CẢM ƠN



Sau khi trải qua quá trình thực hiện khóa luận tốt nghiệp trong 5 tháng, nhóm đã nhận được nhiều sự giúp đỡ và hỗ trợ của các giảng viên trong trường Đại Học Sư Phạm Kỹ Thuật Thành Phố Hồ Chí Minh, đặc biệt là TS. Trần Đức Thiện đã tận tình hướng dẫn, góp ý và tạo mọi điều kiện cho nhóm hoàn thành khóa luận tốt nghiệp một cách thuận lợi.

Tiếp đến nhóm gửi lời cảm ơn đến các thành viên trong RIC LAB đã nhiệt tình hỗ trợ và động viên trong quá trình thực hiện khóa luận tốt nghiệp này.

Với thời gian cũng như sự hiểu biết còn hạn chế, nhóm gặp không ít khó khăn trong việc tiếp cận vào việc nghiên cứu để thực hiện đề tài nên đồ án này không tránh khỏi những thiếu sót không mong muốn, nhóm rất mong nhận được sự góp ý của các thầy cô để đề tài của nhóm ngày càng hoàn thiện hơn nữa.

Cuối cùng, nhóm xin chúc cho thầy cô thật nhiều sức khỏe để tiếp tục trên con đường giảng dạy.

Nhóm xin chân thành cảm ơn!

Tp.HCM, ngày ... tháng ... năm 2023

Sinh viên thực hiện

(Ký và ghi rõ họ tên)

## TÓM TẮT ĐỒ ÁN

Sau quá trình thực hiện khóa luận tốt nghiệp với đề tài “Thiết kế mô hình Robot Delta ứng dụng phân loại rác thải” nhóm đã học hỏi được thêm nhiều kiến thức bổ ích về Robot cũng như về xử lý ảnh và rút ra được nhiều kinh nghiệm. Trong báo cáo này nhóm sẽ trình bày từng bước nhóm đã thực hiện để hoàn thành khóa luận tốt nghiệp. Đầu tiên nhóm đã tìm hiểu về cấu trúc và thiết kế mô hình trên phần mềm Solidworks, sau khi thiết kế hoàn tất nhóm tiến hành thi công phần cứng. Tiếp đến nhóm đã có những bước đầu điều khiển Robot và các thiết bị liên quan. Bên cạnh đó nhóm cũng thiết kế giao diện người dùng trên phần mềm Visual Studio Code để có thể giám sát và điều khiển hệ thống dễ dàng hơn. Để phát hiện, phân loại hai đối tượng rác thải (chai nhựa và lon) nhóm đã xây dựng và huấn luyện mô hình mạng YOLOv5 sau đó tiến hành tính toán vị trí để điều khiển Robot Delta thực hiện phân loại. Cuối cùng nhóm tiến hành đánh giá kết quả phân loại hai đối tượng (chai nhựa và lon) với số lượng 900 mẫu trong nhiều điều kiện ánh sáng khác nhau và đưa ra kết luận về độ chính xác, tốc độ xử lý của mô hình.

## MỤC LỤC

<b>NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP</b> .....	i
<b>LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN TỐT NGHIỆP</b> .....	iii
<b>PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN</b> .....	v
<b>LỜI CAM ĐOAN</b> .....	vi
<b>LỜI CẢM ƠN</b> .....	vii
<b>TÓM TẮT ĐỒ ÁN</b> .....	viii
<b>MỤC LỤC</b> .....	ix
<b>DANH MỤC BẢNG</b> .....	xiii
<b>DANH MỤC HÌNH ẢNH</b> .....	xv
<b>DANH MỤC CÁC TỪ VIẾT TẮT</b> .....	xviii
<b>CHƯƠNG 1: TỔNG QUAN</b> .....	1
<b>1.1 Đặt vấn đề</b> .....	1
<b>1.2 Tình hình nghiên cứu</b> .....	1
1.2.1 Nghiên cứu nước ngoài .....	1
1.2.2 Nghiên cứu trong nước .....	2
1.2.3 Nghiên cứu tại trường Đại học Sư Phạm Kỹ Thuật TP. Hồ Chí Minh .....	4
<b>1.3 Mục tiêu đề tài</b> .....	5
<b>1.4 Nội dung nghiên cứu</b> .....	5
<b>1.5 Giới hạn đề tài</b> .....	5
<b>1.6 Phương pháp đánh giá kết quả</b> .....	5
<b>CHƯƠNG 2: CƠ SỞ LÝ THUYẾT</b> .....	6
<b>2.1 Giải thuật phân loại rác và quy trình thực hiện</b> .....	6
2.1.1 Giải thuật phân loại rác thải .....	6
2.1.2 Quy trình thực hiện phân loại rác thải .....	7
<b>2.2 Giới thiệu Robot Delta</b> .....	8
2.2.1 Cấu tạo Robot Delta .....	8
2.2.2 Nguyên lý hoạt động Robot Delta .....	8

<b>2.3 Giới thiệu chung về các thiết bị.....</b>	<b>9</b>
2.3.1 Giới thiệu chung về Arduino .....	9
2.3.2 Giới thiệu chung về động cơ bước và bộ điều khiển động cơ.....	10
<b>2.4 Tổng quan thị giác máy tính, xử lý ảnh và các phần mềm.....</b>	<b>11</b>
2.4.1 Giới thiệu về thị giác máy tính .....	11
2.4.2 Giới thiệu về xử lý ảnh .....	11
2.4.3 Giới thiệu phần mềm sử dụng trong quá trình huấn luyện.....	12
<b>2.5 Giới thiệu mạng thần kinh CNN và mô hình YOLO.....</b>	<b>13</b>
2.5.1 Cấu trúc mạng thần kinh CNN .....	13
2.5.2 Mô hình YOLO (You Only Look Once).....	15
2.5.3 Phương pháp đánh giá mô hình phát hiện đối tượng .....	21
<b>2.6 Phương pháp chuyển đổi hệ tọa độ xác định vị trí đối tượng.....</b>	<b>23</b>
<b>CHƯƠNG 3: THIẾT KẾ VÀ THI CÔNG PHẦN CỨNG .....</b>	<b>24</b>
<b>3.1 Tổng quan về hệ thống.....</b>	<b>24</b>
<b>3.2 Yêu cầu thiết kế phần cứng .....</b>	<b>25</b>
<b>3.3 Thiết kế mô hình trên phần mềm Solidworks .....</b>	<b>25</b>
3.3.1 Đế cố định Robot.....	26
3.3.2 Đế di động Robot .....	27
3.3.3 Khớp trên Robot .....	28
3.3.4 Khớp hình bình hành Robot .....	28
3.3.5 Khung Robot .....	29
3.3.6 Một số chi tiết khác .....	29
<b>3.4 Lựa chọn thiết bị .....</b>	<b>30</b>
3.4.1 Động cơ .....	30
3.4.2 Thiết bị điện.....	30
3.4.3 Thiết bị điều khiển.....	32
3.4.4 Camera Logitech .....	33
3.4.5 Cơ cấu phân loại .....	33
<b>3.5 Thi công phần cứng.....</b>	<b>35</b>

3.5.1 Các chi tiết của Robot .....	35
3.5.2 Băng tải.....	36
3.5.3 Sơ đồ hệ thống .....	37
3.5.4 Thi công tủ điện.....	39
<b>CHƯƠNG 4: TÍNH TOÁN ĐỘNG HỌC VÀ THIẾT KẾ PHẦN MỀM ĐIỀU KHIỂN.....</b>	<b>40</b>
4.1 Các thông số của Robot Delta .....	40
4.2 Động học nghịch .....	41
4.3 Động học thuận.....	47
4.4 Không gian làm việc của Robot Delta .....	52
4.5 Mô tả quy trình hoạt động và lưu đồ giải thuật của hệ thống .....	53
4.6 Lập trình, thiết kế phần mềm điều khiển và giao diện người dùng .....	57
4.6.1 Lập trình điều khiển Robot và các thiết bị .....	57
4.6.2 Lập trình, thiết kế phần mềm điều khiển và giao diện người dùng.....	57
<b>CHƯƠNG 5: PHÁT HIỆN, PHÂN LOẠI VÀ XÁC ĐỊNH VỊ TRÍ ĐỐI TƯỢNG .....</b>	<b>59</b>
5.1 Quy trình huấn luyện mô hình YOLO.....	59
5.2 Chuẩn bị tập dữ liệu .....	59
5.2.1 Thu thập dữ liệu.....	59
5.2.2 Dán nhãn dữ liệu .....	60
5.2.3 Phân chia tập dữ liệu .....	61
5.2.4 Tiền xử lý dữ liệu .....	61
5.2.5 Làm giàu dữ liệu.....	61
5.3 Quá trình huấn luyện.....	62
5.3.1 Quá trình huấn luyện trên phần mềm Google Colab.....	62
5.3.2 Kết quả huấn luyện.....	64
5.3.3 Đánh giá huấn luyện .....	67
5.4 Tính toán và xác định vị trí đối tượng .....	68
<b>CHƯƠNG 6: KIỂM CHỨNG VÀ ĐÁNH GIÁ KẾT QUẢ THỰC NGHIỆM .</b>	<b>72</b>

<b>6.1 Kiểm chứng động học .....</b>	<b>72</b>
6.1.1 Kiểm chứng động học trên Visual Studio Code .....	72
6.1.2 Kiểm chứng động học nghịch trên checkerboard.....	74
6.1.3 Kiểm chứng động học thực tế .....	76
<b>6.2 Kiểm chứng YOLO .....</b>	<b>78</b>
6.2.1 Kiểm chứng và thử nghiệm lần thứ nhất .....	79
6.2.2 Kiểm chứng và thử nghiệm lần thứ hai .....	80
<b>6.3 Đánh giá kết quả toàn hệ thống .....</b>	<b>80</b>
<b>CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>	<b>85</b>
<b>7.1 Kết luận .....</b>	<b>85</b>
<b>7.2 Hướng phát triển.....</b>	<b>85</b>
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>86</b>
<b>PHỤ LỤC .....</b>	<b>88</b>

## **DANH MỤC BẢNG**

Bảng 1. 1: Một số Robot phân loại rác trên thế giới .....	2
Bảng 1. 2: Một số Robot Delta ở Việt nam.....	3
Bảng 1. 3: Đề tài Robot phân loại tại trường Đại học Sư Phạm Kỹ Thuật TP.HCM.	4
Bảng 3. 1: Một số chi tiết khác của Robot Delta .....	29
Bảng 3. 2: Động cơ Nema 17 .....	30
Bảng 3. 3: Bảng danh sách các thiết bị điện .....	31
Bảng 3. 4: Bảng danh sách các thiết bị điều khiển .....	32
Bảng 3. 5: Thông tin về Camera Logitech C920 .....	33
Bảng 3. 6: Các thiết bị cơ cấu khí nén .....	34
Bảng 3. 7: Các chi tiết sau gia công .....	35
Bảng 3. 8: Thông số của băng tải .....	36
Bảng 3. 9: Hình ảnh trước và sau thi công tủ điện.....	39
Bảng 4. 1: Thông số các chi tiết Robot .....	40
Bảng 5. 1: Các bước huấn luyện mô hình YOLOv5 trên Google Colab .....	63
Bảng 5. 2: Chương trình cụ thể các bước huấn luyện.....	63
Bảng 6. 1: Nhập tọa độ để kiểm chứng động học TH1 .....	72
Bảng 6. 2: Nhập tọa độ để kiểm chứng động học TH2.....	73
Bảng 6. 3: Nhập góc và vị trí kiểm chứng động học trên checkerboard.....	75
Bảng 6. 4: Nhập tọa độ góc để kiểm chứng động học thực tế .....	76
Bảng 6. 5: Huấn luyện với dữ liệu 515 ảnh (ánh sáng bình thường) .....	79
Bảng 6. 6: Huấn luyện với dữ liệu 515 ảnh (ánh sáng tối) .....	79
Bảng 6. 7: Huấn luyện với dữ liệu 1026 ảnh (ánh sáng bình thường) .....	80
Bảng 6. 8: Huấn luyện với dữ liệu 1026 ảnh (ánh sáng tối) .....	80
Bảng 6. 9: Quy trình giai đoạn hoạt động của hệ thống.....	81
Bảng 6. 10: Bảng đánh giá hệ thống lần 1 .....	83
Bảng 6. 11: Bảng đánh giá hệ thống lần 2 .....	83



## **DANH MỤC HÌNH ẢNH**

Hình 2. 1: Quy trình phân loại rác.....	7
Hình 2. 2: Một số mô hình Robot Delta.....	8
Hình 2. 3: Arduino Mega 2560 .....	9
Hình 2. 4: Sơ đồ chân Arduino Mega 2560 .....	9
Hình 2. 5: Một số loại động cơ bước .....	10
Hình 2. 6: Driver A4988 .....	10
Hình 2. 7: Board shield V3 .....	11
Hình 2. 8: Phần mềm Roboflow.....	12
Hình 2. 9: Phần mềm Google Colab .....	12
Hình 2. 10: Cấu trúc mạng CNN.....	13
Hình 2. 11: Thuật toán của lớp tích chập .....	14
Hình 2. 12: So sánh giữa Max Pooling và Average Pooling .....	14
Hình 2. 13: Quá trình làm phẳng.....	14
Hình 2. 14: Cấu tạo của lớp kết nối đầy đủ.....	15
Hình 2. 15: Cấu trúc mô hình YOLOv5.....	16
Hình 2. 16: Cấu trúc trích xuất đặc trưng của YOLOv5 .....	16
Hình 2. 17: Các feature map của mô hình YOLOv5 với đầu vào ảnh 416x416.....	17
Hình 2. 18: Anchor box trong feature map .....	18
Hình 2. 19: Công thức dự đoán Bounding box từ Anchor box .....	19
Hình 2. 20: Thuật toán Non-max suppression .....	19
Hình 2. 21: Công thức tính IOU.....	21
Hình 2. 22: Confusion Matrix cho hai đối tượng.....	21
Hình 2. 23: Quy trình chuyển đổi hệ tọa độ .....	23
Hình 3. 1: Sơ đồ khối tổng quan hệ thống .....	24
Hình 3. 2: Mô hình 3D Robot Delta. a) Góc nhìn tổng quan; b) Góc nhìn từ ngoài vào; c) Góc nhìn từ trên xuống; d) Góc nhìn từ dưới lên.....	26

Hình 3. 3: Mô hình 3D để cố định Robot.....	27
Hình 3. 4: Mô hình 3D để di động Robot .....	27
Hình 3. 5: Mô hình 3D khớp trên Robot.....	28
Hình 3. 6: Mô hình 3D khớp hình bình hành Robot .....	28
Hình 3. 7: Khung Robot .....	29
Hình 3. 8: Mô hình thực tế sau khi thi công .....	36
Hình 3. 9: Sơ đồ mạch động lực.....	37
Hình 3. 10: Sơ đồ mạch điều khiển tổng quát.....	37
Hình 3. 11: Sơ đồ kết nối Arduino-Board Shield V3.....	38
Hình 3. 12: Sơ đồ kết nối Board Shield V3-Driver-Động cơ bước .....	38
Hình 3. 13: Các thiết bị bên trong tủ điện.....	39
Hình 4. 1: Mô hình động học của Robot Delta .....	40
Hình 4. 2: Điểm cuối cùng của Robot và các góc $\theta$ .....	41
Hình 4. 3: Giao tuyến của hình cầu và mặt phẳng yz .....	41
Hình 4. 4: Giao điểm của 2 đường tròn.....	42
Hình 4. 5: Hệ tọa độ mới $(x', y', z')$ .....	47
Hình 4. 6: Mô hình sau khi tính tiền $J_1, J_2, J_3$ .....	48
Hình 4. 7: Giao điểm 3 mặt cầu .....	48
Hình 4. 8: Hình chiếu trên mặt phẳng xy .....	49
Hình 4. 9: Không gian làm việc của Robot Delta trong. a) Không gian 3 chiều xyz; Không gian 2 chiều b) xy; c) xz; d) yz.....	52
Hình 4. 10: Lưu đồ tổng quan hệ thống .....	54
Hình 4. 11: Lưu đồ khói xử lý tín hiệu .....	55
Hình 4. 12: Lưu đồ khói điều khiển .....	55
Hình 4. 13: Lưu đồ dự đoán Bounding box của YOLOv5.....	56
Hình 4. 14: Lập trình điều khiển trên phần mềm Arduino IDE .....	57
Hình 4. 15: Giao diện giới thiệu trên Visual Studio Code .....	58

Hình 4. 16: Giao diện điều khiển trên Visual Studio Code.....	58
Hình 5. 1: Cấu trúc huấn luyện mô hình YOLO .....	59
Hình 5. 2: Tải dữ liệu lên phần mềm Roboflow .....	60
Hình 5. 3: Dán nhãn theo Bounding box .....	60
Hình 5. 4: Phân chia tập dữ liệu để huấn luyện .....	61
Hình 5. 5: Dữ liệu sau khi đã được làm giàu .....	62
Hình 5. 6: Các hình ảnh mới được tạo ra sau khi được làm giàu.....	62
Hình 5. 7: Quá trình huấn luyện trên Google Colab .....	64
Hình 5. 8: Kết quả huấn luyện với 185 ảnh đầu vào.....	65
Hình 5. 9: Kết quả huấn luyện với hình ảnh .....	66
Hình 5. 10: Kết quả huấn luyện với camera thời gian thực .....	66
Hình 5. 11: Kết quả huấn luyện trên phần mềm Google Colab .....	67
Hình 5. 12: Hệ tọa độ Robot Delta và không gian làm việc Camera.....	68
Hình 5. 13: Xác định tọa độ tâm của vật từ Bounding box.....	68
Hình 5. 14: Hệ tọa độ Pixel trong vùng làm việc của camera.....	69
Hình 5. 15: Hệ tọa độ Ảnh trong vùng làm việc của camera .....	70
Hình 5. 16: Tương quan giữa hệ tọa độ Ảnh và hệ tọa độ Robot .....	71
Hình 6. 1: Hệ tọa độ Robot so với checkerboard.....	74
Hình 6. 2: Camera với ánh sáng bình thường .....	78
Hình 6. 3: Camera với ánh sáng tối.....	78

## DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Tên tiếng Anh	Tên tiếng Việt
YOLO	You Only Look Once	
YOLOv5	You Only Look Once Version 5	
MATLAB	Matrix Laboratory	
GPS	Global Positioning System	Hệ thống định vị toàn cầu
CNN	Convolution Neural Network	Mạng nơ-ron tích chập
IOU	Intersection Over Union	
TP	True Positive	
FP	False Positive	
TN	True Negative	
FN	False Negative	
PLA	Polylactic Acid	
PC	Personal Computer	Máy tính cá nhân
HD	High Definition	Độ nét cao
DC	Direct Current	Dòng điện một chiều
AC	Alternating Current	Dòng điện xoay chiều
CB	Circuit Breaker	Cầu dao
NO	Normal Open	Thường mở
NC	Normal Close	Thường đóng
I/O	Input/ Output	Ngõ vào/ Ngõ ra
VCC	Voltage at the common collector	
GND	Ground	
EMG BT	Emergency Button	Nút nhấn khẩn cấp
UART	Universal Asynchronous Receiver-Transmitter	
GPU	Graphics Processing Unit	
TPU	Tensor Processing Unit	
CPU	Central Processing Unit	
TH		Trường Hợp

## CHƯƠNG 1: TỔNG QUAN

Trong chương đầu tiên nhóm sẽ đưa ra thực trạng vấn đề rác thải đang là một vấn đề lớn và cực kỳ cấp thiết trên thế giới. Từ những nghiên cứu trên thế giới, những nghiên cứu ở Việt Nam cũng như tại Trường Đại học Sư Phạm Kỹ Thuật TP.HCM, nhóm nhận thấy vấn đề áp dụng Robot trong phân loại rác thải vẫn chưa thực sự được áp dụng rộng rãi và nhóm quyết định chọn đề tài này. Nhóm sẽ trình bày mục tiêu nhóm đặt ra, nội dung nhóm đã thực hiện được cũng như những vấn đề còn hạn chế và đưa ra những phương pháp đánh giá đã sử dụng.

### 1.1 Đặt vấn đề

Trong thời điểm công nghiệp hóa, hiện đại hóa hiện nay cùng với sự gia tăng dân số nhanh chóng dẫn đến lượng rác thải công nghiệp và rác thải sinh hoạt cũng tăng lên rất nhanh. Trung bình mỗi năm con người tạo ra 1,3 tỷ tấn chất thải rắn đô thị và chỉ trong năm 2020 thế giới thải ra hơn hai tỷ tấn rác thải rắn. Với số lượng khổng lồ và chủng loại đa dạng, vấn đề xử lý rác thải đã và đang là một vấn đề lớn trên thế giới, đặc biệt là ở các nước đang phát triển. Vấn đề này bắt buộc phải được thực hiện vì nó không chỉ ảnh hưởng tới môi trường sống bị ô nhiễm mà còn ảnh hưởng trực tiếp tới đời sống và sức khỏe của con người. Việc thu gom, phân loại và xử lý rác thải công nghiệp mất rất nhiều thời gian và công sức cũng như ảnh hưởng đến sức khỏe trực tiếp của người thực hiện công việc. Chính vì thế với sự phát triển nhanh chóng và vượt bậc của quá trình công nghiệp hóa, hiện đại hóa trên thế giới, các nước lớn đang dần áp dụng quá trình tự động hóa vào trong công nghiệp và sinh hoạt. Các hệ thống, dây chuyền tự động và Robot đang được ứng dụng ngày càng nhiều với ưu điểm tốc độ xử lý và thực hiện công việc nhanh gấp nhiều lần cũng như có thể thay thế hoàn toàn con người trong các công việc nguy hiểm. Đây chính là giải pháp cho vấn đề phân loại rác thải.

### 1.2 Tình hình nghiên cứu

#### 1.2.1 Nghiên cứu nước ngoài

Trước thực trạng trên, nhiều quốc gia đang phát triển, tổ chức quốc tế, các nhóm nghiên cứu và doanh nghiệp đang đi tiên phong nỗ lực tìm kiếm những giải pháp mới hiệu quả nhằm hạn chế, thu thập và xử lý triệt tiêu lượng rác thải. Một số hãng sản xuất Robot lớn trên thế giới như ABB, AMP, Machinex,... đã thiết kế và cho ra mắt các loại Robot với các tính năng và công việc khác nhau nhằm giải quyết các vấn đề rác thải. Có thể nói ứng dụng Robot vào công việc xử lý rác thải sẽ là xu thế toàn cầu trong nhiều năm tiếp theo.

## Chương 1: Tổng quan

Bảng 1. 1: Một số Robot phân loại rác trên thế giới

STT	Hình ảnh	Thông tin
1		<ul style="list-style-type: none"><li>+ Robot 3 bậc</li><li>+ Công ty FANUC</li><li>+ Sản xuất năm 2021</li><li>+ Nước Nhật Bản</li><li>+ Công suất: 55 lần nhặt/ phút</li><li>+ Loại rác: rác thải nhựa</li><li>+ Cơ cấu hút</li></ul>
2		<ul style="list-style-type: none"><li>+ Robot 6 bậc</li><li>+ Công ty CSAIL</li><li>+ Sản xuất năm 2019</li><li>+ Nước Mỹ</li><li>+ Công suất: 10-15 lần nhặt/ phút</li><li>+ Loại rác: rác thải nhựa</li><li>+ Cơ cấu gấp</li></ul>
3		<ul style="list-style-type: none"><li>+ Robot Delta</li><li>+ Công ty AMP</li><li>+ Sản xuất năm 2022</li><li>+ Nước Mỹ</li><li>+ Công suất: 80 lần nhặt/ phút</li><li>+ Loại rác: rác thải sinh hoạt</li><li>+ Cơ cấu hút</li></ul>

### 1.2.2 Nghiên cứu trong nước

Tại Việt Nam, việc nghiên cứu phát triển Robot đã có những bước tiến đáng kể trong nhiều năm qua. Nhiều đơn vị trên toàn quốc thực hiện thành công các nghiên cứu về ứng dụng của Robot như: Trung tâm Tự động hóa thuộc Bộ Công nghiệp, Đại học Bách khoa Thành phố Hồ Chí Minh, Đại học Sư Phạm Kỹ Thuật Thành phố Hồ Chí Minh, Viện Khoa học và Công nghệ quân sự, Học viện Kỹ thuật Quân sự, Viện Cơ học,... Việt Nam đã có các công ty lớn, nhỏ chuyên thiết kế, sản xuất và xuất khẩu Robot có thể kể đến nổi bật như: Next Robotics, Tosy,... Ngoài ra các công ty, xí nghiệp lớn cũng đang từng bước áp dụng Robot vào trong các hệ thống, dây chuyền sản xuất tự động của họ nhằm cải thiện năng suất cũng như giảm thiểu chi phí nhân

lực. Có thể kể đến một số tập đoàn lớn đã ứng dụng Robot vào trong quy trình sản xuất của họ: Vinfast với hơn 1000 Robot phục vụ cho việc sản xuất ô tô, SamSung sử dụng các tay Robot trong sản xuất, lắp ráp linh kiện,... Điều đó cho thấy Việt Nam cũng đã và đang từng bước làm chủ công nghệ Robot, không để bị bỏ xa trong lĩnh vực công nghệ này. Tuy nhiên hầu hết các ứng dụng về Robot ở nước ta chủ yếu được sử dụng trong các nhà máy sản xuất, vận chuyển hàng hóa hoặc y tế. Rất ít người nghĩ đến việc ứng dụng Robot cho vấn đề phân loại rác thải, hầu hết chỉ dừng lại ở những nghiên cứu tại các trường đại học chứ chưa ứng dụng vào các nhà máy công nghiệp. Đối với các ứng dụng yêu cầu về độ chính xác, tốc độ xử lý nhanh và hoạt động với công suất lớn như: phân loại rác thải, đính linh kiện,... Thì các loại Robot nối tiếp thông thường không thể đáp ứng được. Chính vì vậy các loại Robot song song, Robot Delta đã được thiết kế ra nhằm giải quyết các vấn đề đó. Với thiết kế nhỏ gọn, tốc độ xử lý nhanh với các đối tượng nhẹ cũng như dễ dàng lắp ráp và tích hợp vào các hệ thống khác nhau, Robot Delta chính là loại Robot phù hợp nhất trong vấn đề xử lý rác thải.

Bảng 1. 2: Một số Robot Delta ở Việt nam

STT	Hình ảnh	Thông tin
1		<ul style="list-style-type: none"> <li>+ Robot Delta công ty TOPSTAR</li> <li>+ Ứng dụng rộng rãi tại các nhà máy sản xuất trong các công đoạn: xử lý, phân loại và đóng gói sản phẩm với tốc độ và độ chính xác cao</li> </ul>
2		<ul style="list-style-type: none"> <li>+ Robot Delta công ty IMWI</li> <li>+ Được dùng phổ biến trong các tác vụ gấp thả và sắp xếp sản phẩm trên băng chuyền sản xuất công nghiệp</li> <li>+ Mẫu Delta X phù hợp cho công việc nghiên cứu và thử nghiệm của các phòng Lab về Robotics</li> </ul>

### 1.2.3 Nghiên cứu tại trường Đại học Sư Phạm Kỹ Thuật TP. Hồ Chí Minh

Đã có rất nhiều nghiên cứu về Robot tại trường Đại học Sư Phạm Kỹ Thuật TP. Hồ Chí Minh trong những năm gần đây tuy nhiên chủ yếu những nghiên cứu chỉ dừng lại ở phòng thí nghiệm chứ chưa có tính ứng dụng vào môi trường công nghiệp. Các đề tài về Robot chủ yếu liên quan đến phân loại sản phẩm, dò tìm đường đi, xác định vị trí,... Nhìn chung vẫn chưa có nhiều nghiên cứu về vấn đề ứng dụng Robot trong việc phân loại và xử lý rác thải.

Bảng 1. 3: Đề tài Robot phân loại tại trường Đại học Sư Phạm Kỹ Thuật TP.HCM

Tên đề tài	Tác giả	Kết quả đạt được	Hạn chế
Ứng dụng Robot tích hợp thị giác máy tính phân loại rác	Lê Tiến Thành Nguyễn Thế Hùng	Robot Scara 3 bậc tự do, sử dụng hệ thống thị giác máy tính phân loại rác thải nhựa	+ Còn hạn chế số lượng đối tượng phân loại, về vị trí phân loại
Thiết kế và chế tạo hệ thống phân loại rác tự động	Nguyễn Đức Lợi Lương Công Hiệu Võ Nhựt Minh	Thiết kế mô hình Robot Scara phân loại rác thải	+ Tốc độ phân loại chưa cao
Phân loại sản phẩm linh hoạt ứng dụng Robot và xử lý ảnh	Lê Đức Kông Nguyễn Văn Thiên	Thiết kế mô hình Robot Delta phân loại sản phẩm theo màu sắc	+ Tốc độ và độ chính xác chưa cao

Sau khi tham khảo các đề tài ứng dụng Robot phân loại của một vài nhóm nghiên cứu tại trường Đại học Sư Phạm Kỹ thuật, nhóm nhận thấy nhìn chung các đề tài đã có thể thiết kế và thi công các hệ thống Robot phân loại khác nhau nhưng vẫn còn rất nhiều hạn chế như: tốc độ xử lý vẫn còn chưa cao, chỉ có thể phân loại với một số ít các đối tượng khác nhau chưa thể mở rộng thêm. Bên cạnh đó, các đề tài Robot Delta ở trường cũng chỉ mới dừng lại trong việc phân loại sản phẩm. Mà xu thế toàn cầu hiện nay là ứng dụng xử lý ảnh và trí tuệ nhân tạo vào trong các hệ thống tự động hóa vẫn chưa được áp dụng nhiều dẫn đến những đề tài này vẫn chưa thể sử dụng vào trong công nghiệp. Đối với vấn đề phân loại rác thải thì chỉ mới có một vài nhóm nghĩ đến và thực hiện những bước đầu tiên trong khi vấn đề rác thải trong công nghiệp không đơn giản như vậy. Chính vì những yếu tố trên cũng như muôn tiếp cận thế giới trong lĩnh vực Robot, tự động hóa nhóm quyết định chọn đề tài “THIẾT KẾ MÔ HÌNH ROBOT DELTA ỨNG DỤNG PHÂN LOẠI RÁC THẢI”.

### 1.3 Mục tiêu đề tài

Thiết kế, thi công và điều khiển mô hình Robot Delta. Xây dựng và huấn luyện mô hình YOLOv5 phân loại 2 loại rác thải (chai nhựa và lon). Kết hợp giải thuật phân loại và xác định vị trí điều khiển Robot phân loại rác thải.

### 1.4 Nội dung nghiên cứu

Chương 1 nhóm đã đưa ra các vấn đề liên quan, lý do chọn đề tài, tình hình nghiên cứu trong và ngoài nước cũng như mục tiêu, giới hạn của đề tài. Tiếp theo nhóm trình bày các cơ sở lý thuyết đã áp dụng để thực hiện đề tài về Robot Delta và mô hình YOLOv5 trong chương 2. Ở chương 3 sẽ trình bày cách thiết kế mô hình, lựa chọn thiết bị và các công đoạn thi công. Chương 4 nhóm sẽ tính toán động học, xây dựng chương trình điều khiển Robot và trình bày lưu đồ giải thuật hệ thống. Tiếp theo sẽ trình bày các bước huấn luyện mô hình YOLOv5 cho 2 đối tượng rác thải ở chương 5. Chương 6 sau khi đã huấn luyện mạng thành công sẽ áp dụng vào mô hình, tiến hành chạy thực nghiệm và đánh giá kết quả phân loại. Cuối cùng sẽ đưa ra kết luận và hướng phát triển trong tương lai ở chương 7.

### 1.5 Giới hạn đề tài

Trong đề tài này vì nhóm tham khảo một số mô hình Robot Delta nhưng vì kiến thức về cơ khí còn hạn chế nên quá trình thiết kế và thi công Robot vẫn chưa tối ưu, Robot vẫn còn bị rung lắc khi hoạt động với tốc độ cao.

Ngoài ra nhóm chỉ huấn luyện mô hình YOLOv5 phân loại với 2 đối tượng (chai nhựa, lon) chưa thể phân loại với nhiều đối tượng hơn. Tốc độ phân loại đối tượng chưa cao.

### 1.6 Phương pháp đánh giá kết quả

Trong quá trình thực hiện có nhiều bước cần phải đánh giá và kiểm chứng lại quá trình thực hiện để đảm bảo sai số phải ở mức tối thiểu và tốc độ phải tối ưu. Một số cách và ứng dụng mà nhóm đã thực hiện để đánh giá lại quá trình thực hiện đề tài:

- Đánh giá tính toán động học và quy hoạch quỹ đạo: nhóm đã tiến hành lập trình điều khiển thuận nghịch cho Robot, sau đó tổng hợp và đưa ra sai số cho phép từ đó điều chỉnh phần lập trình và phần cứng.
- Đánh giá quá trình huấn luyện mô hình YOLO: tiến hành ghi nhận lại số lượng phân loại chính xác và chưa chính xác, lập ra ma trận Confusion Matrix và đánh giá mô hình huấn luyện qua các thông số Recall, Precision và F1-score.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

Chương này trình bày tổng quan và khái quát cơ sở lý thuyết các nội dung liên quan được thực hiện trong đề tài. Trình bày về các phương pháp phân loại rác. Giới thiệu về cách tạo và nguyên lý hoạt động Robot Delta. Giới thiệu chung về các thiết bị sử dụng trong đề tài. Giới thiệu tổng quan về thị giác máy tính và xử lý ảnh. Giới thiệu cấu trúc mạng thần kinh CNN, mô hình YOLO và các phương pháp đánh giá. Cuối cùng nói về phương pháp chuyển đổi hệ tọa độ xác định vị trí đối tượng.

### 2.1 Giải thuật phân loại rác và quy trình thực hiện

Để có thể thiết kế một hệ thống phân loại và xử lý rác thải tự động ta sẽ phải quan tâm đến hai yếu tố là giải thuật phân loại và cơ cấu thực hiện phân loại. Cả hai yếu tố này đều ảnh hưởng đến độ chính xác cũng như tốc độ phân loại rác của hệ thống.

#### 2.1.1 Giải thuật phân loại rác thải

Giải thuật phân loại là phương pháp xác định ra và kiểm tra xem đó là loại rác thải nào cũng như cho biết vị trí chính xác của loại rác đó đang ở đâu trong một vùng không gian xác định.

Đối với giải thuật phân loại thì có rất nhiều phương pháp khác nhau tùy thuộc vào độ sáng tạo cũng như kiến thức của người thiết kế hệ thống. Chúng ta có thể sử dụng, kết hợp các công cụ khác nhau để kiểm tra, phân loại và đưa ra vị trí của loại rác thải. Hiện nay có rất nhiều phương pháp để phân loại rác thải như: Sử dụng các loại cảm biến khác nhau, sử dụng camera quang phổ, huấn luyện mạng thần kinh nhân tạo.

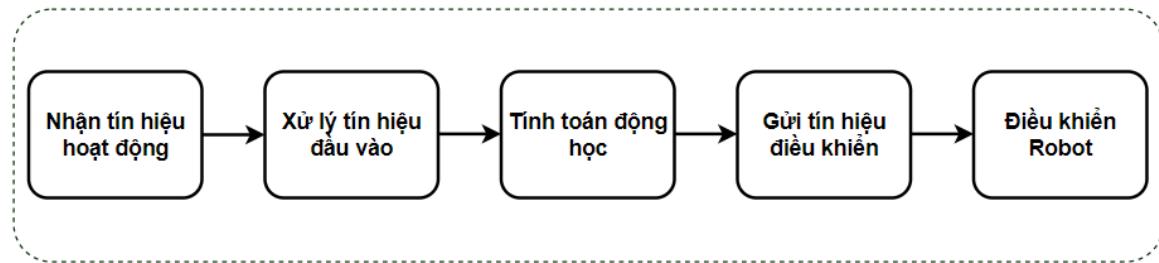
- ❖ Phương pháp đầu tiên cũng là phương pháp phổ biến nhất trên thế giới hiện nay là **sử dụng các loại cảm biến** để xác định. Ta có thể sử dụng các loại cảm biến khác nhau để phân loại ra các loại rác thải khác nhau như: cảm biến kim loại với rác thải kim loại, cảm biến độ ẩm với rác thải hữu cơ,...
  - **Ưu điểm:** tốc độ xác định, phân loại nhanh và độ chính xác cũng khá cao. Ngoài ra không phải sử dụng thêm các thiết bị khác để xác định vị trí như camera.
  - **Nhược điểm:** chỉ phân loại được một số loại rác cố định, không thể mở rộng đối tượng phân loại.
- ❖ Một phương pháp nữa cũng được dùng để phân loại rác thải trên thế giới đó là **sử dụng camera quang phổ**. Phương pháp là đo sự phản xạ ánh sáng của

các loại rác thải khác nhau để phân loại và thường được ứng dụng trong các ứng dụng phân loại rác thải trong xây dựng.

- **Ưu điểm:** Có thể ứng dụng vào trong nhiều ứng dụng khác nhau đặc biệt là phân loại rác thải công nghiệp và xây dựng.
  - **Nhược điểm:** Phải sử dụng song song 2 camera và phải có thêm những khâu tiền xử lý rác thải vì không thể phân biệt được những đối tượng có chỉ số phản xạ gần bằng nhau.
- ❖ Phương pháp cuối cùng được cho là có tính ứng dụng cao nhất là **huấn luyện mạng thần kinh nhân tạo**. Với tập dữ liệu đã được chuẩn bị trước để tiến hành huấn luyện.
- **Ưu điểm:** Có thể phân loại rất nhiều loại rác thải khác nhau và có thể mở rộng số lượng rác phân loại tùy thuộc vào số lượng và chất lượng của đầu vào huấn luyện ngoài ra tốc độ phân loại gần như cũng bằng thời gian thực.
  - **Nhược điểm:** Người huấn luyện mạng phải có một số kiến thức nhất định về xử lý ảnh và mạng thần kinh nhân tạo. Trong quá trình huấn luyện mạng công đoạn chuẩn bị tập dữ liệu đầu vào sẽ tốn khoảng thời gian rất lớn và có thể ảnh hưởng đến chất lượng và tốc độ phân loại.

Mỗi phương pháp sẽ có các ưu nhược điểm khác nhau tùy vào mục đích của người sử dụng phương pháp. Vì có nhiều ưu điểm như: tính ứng dụng, linh hoạt cao, ứng dụng các kiến thức về xử lý ảnh và mạng thần kinh nhân tạo phù hợp với đê tài nêu nhóm đã lựa chọn tiến hành sử dụng phương pháp huấn luyện mạng thần kinh nhân tạo để thực hiện trong đê tài phân loại rác này.

### 2.1.2 Quy trình thực hiện phân loại rác thải



Hình 2. 1: Quy trình phân loại rác

Quy trình thực hiện phân loại rác thải được thực hiện qua 5 bước:

**Bước 1 - Nhận tín hiệu hoạt động:** Người dùng thao tác trực tiếp trên giao diện điều khiển để điều khiển Robot và các thiết bị hoạt động.

**Bước 2 - Xử lý tín hiệu đầu vào:** Đưa rác thải vào băng tải, khi đối tượng đi qua vùng làm việc của camera thì mô hình sẽ phân loại và tính toán được vị trí của đối tượng.

**Bước 3 - Tính toán động học:** Khi có được vị trí đối tượng thì tiếp tục tính toán động học nghịch để gửi tín hiệu điều khiển cho Robot.

**Bước 4 - Gửi tín hiệu điều khiển:** Bộ điều khiển sẽ tiến hành cấp xung cho các động cơ.

**Bước 5 - Điều khiển Robot:** Khi nhận được tín hiệu xung điều khiển từ động cơ thì Robot sẽ thực thi việc phân loại.

## 2.2 Giới thiệu Robot Delta

### 2.2.1 Cấu tạo Robot Delta

Robot Delta có cấu tạo bao gồm 2 phần để song song: để cố định ở trên để đặt các động cơ và để di động ở dưới đặt cơ cấu chấp hành, 2 phần để này được kết nối qua các cánh tay có dạng hình bình hành được điều khiển qua các động cơ được gắn trên để cố định, đây chính là ưu điểm của Robot Delta so với các loại Robot thông thường khác. Robot Delta chủ yếu làm việc với những ứng dụng ở hai mặt phẳng song song như đính lính kiện, phân loại sản phẩm,... Vì có kết cấu nhỏ gọn với chất liệu nhẹ, tốc độ làm việc của Robot Delta rất nhanh nhưng bù lại không thể làm việc với những ứng dụng yêu cầu chịu tải cao.



Hình 2. 2: Một số mô hình Robot Delta

### 2.2.2 Nguyên lý hoạt động Robot Delta

Nguyên lý hoạt động của Robot Delta là sử dụng các hình bình hành để tạo chuyển động đầu cuối cùng mà vẫn giữ nguyên hướng. Tức là chỉ chuyển động theo

hướng X, Y và Z mà không bị quay hướng. Để có định được đặt ở phía trên chứa các động cơ điều khiển các khớp trên và khớp hình bình hành.

Số bậc của Robot theo số khớp của Robot, trên thế giới đã sản xuất ra các mô hình Robot Delta 4,5,6 bậc nhưng phổ biến và ứng dụng vào công nghiệp nhiều nhất vẫn là Robot Delta 3 bậc.

### 2.3 Giới thiệu chung về các thiết bị

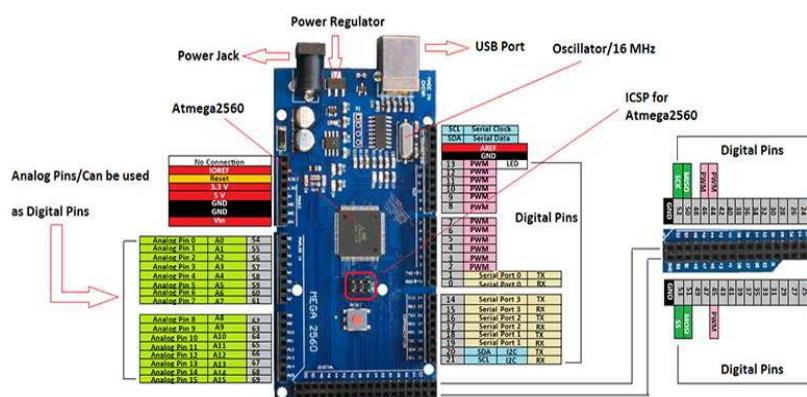
#### 2.3.1 Giới thiệu chung về Arduino

Arduino là một board mạch vi điều khiển do một nhóm giáo sư và sinh viên nước Ý thiết kế và đưa ra sử dụng đầu tiên vào năm 2005. Arduino có thể thực hiện nhiều nhiệm vụ như lấy tín hiệu từ cảm biến đến điều khiển đèn, động cơ, và nhiều đối tượng khác. Ngoài ra Arduino còn có khả năng kết nối, tương tác với các thiết bị GPS, Internet, Camera, điện thoại di động,...

Trên thế giới hiện nay có rất nhiều loại Arduino với các kích thước và cấu hình khác nhau nhưng trong đề tài lần này nhóm sử dụng Arduino Mega 2560 vì yêu cầu kết nối với số lượng thiết bị điều khiển khá lớn và cũng cần một tốc độ xử lý nhất định.



Hình 2. 3: Arduino Mega 2560



Hình 2. 4: Sơ đồ chân Arduino Mega 2560

### 2.3.2 Giới thiệu chung về động cơ bước và bộ điều khiển động cơ

#### 2.3.2.1 Giới thiệu về động cơ bước

Động cơ bước (còn gọi là Step Motor hay Stepping Motor) là một loại động cơ sử dụng điện. Động cơ bước quay theo từng bước một tùy theo số xung được cấp vào sẽ tạo thành một góc nào đó gọi là góc bước, nên có độ chính xác rất cao, đặc biệt là về mặt điều khiển. Nếu góc bước càng nhỏ thì số bước trên mỗi vòng quay của động cơ càng lớn và độ chính xác của vị trí càng lớn.

Các góc bước của động cơ thường được sử dụng phổ biến nhất là góc 1,8 độ, góc 2,5 độ, góc 7,5 độ và góc 15 độ vì đây là các góc thuận lợi cho việc tính toán nhất.



Hình 2. 5: Một số loại động cơ bước

Hiện nay, có 4 phương pháp để điều khiển động cơ bước được sử dụng phổ biến nhất, đó là:

- Điều khiển động cơ bước dạng sóng (Wave).
- Điều khiển động cơ bước đầy (Full step).
- Điều khiển động cơ nửa bước (Half step).
- Điều khiển động cơ vi bước (Microstep).

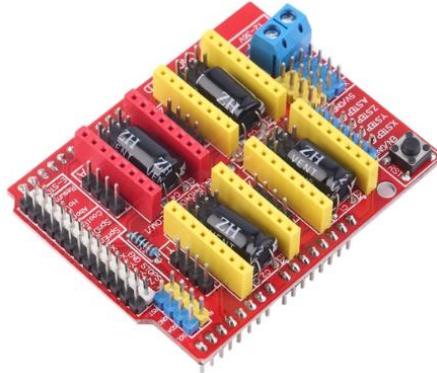
#### 2.3.2.2 Bộ điều khiển động cơ

Bộ điều khiển động cơ bước hay driver là thiết bị không thể thiếu khi sử dụng động cơ bước. Tùy vào loại động cơ, tốc độ và ứng dụng sẽ có các driver phù hợp. Trong đề tài lần này nhóm chọn driver A4988 vì kết cấu nhỏ gọn, dễ dàng kết nối lắp đặt cũng như đáp ứng khá tốt trong ứng dụng cấp xung.



Hình 2. 6: Driver A4988

Ngoài ra nhóm còn sử dụng thêm một thiết bị tích hợp cho Arduino và driver A4988 là Board Shield V3 nhằm giảm số lượng đầu dây cũng như có thể điều khiển cùng lúc cả 3 động cơ.



Hình 2. 7: Board shield V3

### 2.4 Tổng quan thị giác máy tính, xử lý ảnh và các phần mềm

Thị giác máy tính và xử lý ảnh là hai lĩnh vực liên quan mật thiết với nhau, áp dụng một số kỹ thuật trí tuệ nhân tạo AI và nhận dạng mẫu để lấy thông tin có ý nghĩa từ hình ảnh, video và một vài đầu vào trực quan khác.

#### 2.4.1 Giới thiệu về thị giác máy tính

Thị giác máy tính sử dụng xử lý hình ảnh để nhận dạng và phân loại dữ liệu hình ảnh. Trên thực tế, thị giác máy tính đang trở nên phổ biến hơn trong việc xác định các đối tượng từ hình ảnh so với hệ thống nhận thức thị giác con người. Công nghệ này ngày càng được sử dụng rộng rãi trong nhiều ngành nghề khác nhau như: công nghiệp, chăm sóc sức khỏe, giải trí, ...

#### 2.4.2 Giới thiệu về xử lý ảnh

Xử lý ảnh là đầu tàu trong các lĩnh vực 4.0 và là nền tảng của lĩnh vực thị giác máy tính và là quá trình biến đổi từ một ảnh ban đầu sang ảnh khác tùy theo nhu cầu và mong muốn của người sử dụng. Các thuật toán xử lý hình ảnh được sử dụng để trích xuất thông tin từ hình ảnh, khôi phục và nén dữ liệu hình ảnh và video.

Mục đích trong việc sử dụng xử lý ảnh bao gồm:

- Sử dụng một số kỹ thuật khác nhau để phân loại đối tượng.
- Phát triển các thuật toán trong xử lý ảnh để tăng hoặc giảm chất lượng ảnh đầu vào, lọc nhiễu, làm sắc nét, ...

### 2.4.3 Giới thiệu phần mềm sử dụng trong quá trình huấn luyện

Nhóm sử dụng phần mềm Roboflow cho bước chuẩn bị tập dữ liệu đầu vào, Roboflow là một công cụ cho phép người dùng xây dựng các tập dữ liệu hình ảnh hoặc truy cập các tập dữ liệu đã có sẵn trong các ứng dụng đào tạo mô hình xử lý ảnh. Roboflow có đầy đủ các chức năng cần thiết để chuyển đổi, xử lý hình ảnh đầu vào thành mô hình thị giác máy tính. Được tích hợp rất nhiều thư viện kinh điển như: EfficientNet, MobileNet, YOLO, TensorFlow, Pytorch,... cho phép người dùng có thể triển khai dự án trên bất kỳ nền tảng nào.



Hình 2. 8: Phần mềm Roboflow

Trong quá trình đào tạo, nhóm đã sử dụng ứng dụng Colaboratory hay Google Colab là một sản phẩm của Google Research. Colab cho phép chạy chương trình trực tiếp trên trình duyệt mà không cần cài đặt hoặc yêu cầu về phần cứng của máy tính để đáp ứng các yêu cầu về khối lượng công việc nặng về CPU/GPU. Hơn nữa, Colab cung cấp quyền truy cập miễn phí vào cơ sở hạ tầng điện toán như bộ lưu trữ, bộ nhớ, khả năng xử lý, đơn vị xử lý đồ họa (GPU) và đơn vị xử lý tensor (TPU).

Colab cung cấp nhiều loại GPU, thường là Nvidia K80s, T4s, P4s và P100s, tuy nhiên người dùng không thể chọn loại GPU trong Colab, GPU trong Colab thay đổi theo thời gian. Vì là dịch vụ miễn phí nên Colab sẽ có những ưu tiên thứ nhất trong việc sử dụng tài nguyên hệ thống, cũng như giới hạn thời gian sử dụng, thời gian sử dụng tối đa lên tới 12 giờ.



Hình 2. 9: Phần mềm Google Colab

❖ *Ưu điểm của Google Colab:*

- Google Colab miễn phí.
- Cung cấp phần cứng mạnh mẽ cho việc chạy thử chương trình từ CPU, GPU đến TPU.
- Dễ dàng sử dụng và truy cập, kết nối lưu trữ trực tiếp trên Google Drive, Github.

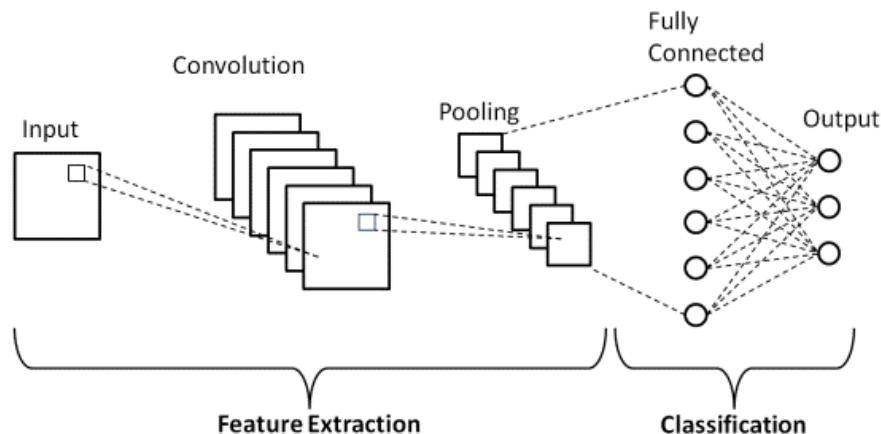
- Sử dụng được nhiều thư viện phổ biến dành cho DeepLearning như Keras, TensorFlow, PyTorch và OpenCV.

❖ *Nhược điểm của Google Colab:*

- Thời gian sử dụng có giới hạn, sau 12 giờ dữ liệu của colab sẽ bị đặt lại.
- Không thể tùy chọn cấu hình phần cứng, với tài khoản miễn phí.

## 2.5 Giới thiệu mạng thần kinh CNN và mô hình YOLO

### 2.5.1 Cấu trúc mạng thần kinh CNN



Hình 2. 10: Cấu trúc mạng CNN

CNN (Convolution Neural Network) hay còn gọi là mạng nơ-ron tích chập là một trong những mô hình Deep Learning vô cùng tiên tiến ứng dụng mạng thần kinh Nơ-ron. CNN cho phép xây dựng các hệ thống thông minh với độ chính xác vô cùng cao.

Mạng CNN được xây dựng theo một kiến trúc tuần tự. Cấu trúc mạng CNN là một chuỗi các lớp có nhiệm vụ khác nhau, bao gồm:

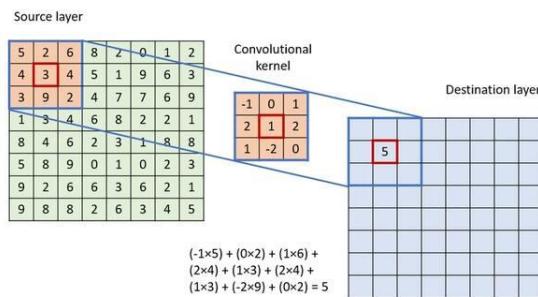
❖ *Lớp tích chập (Convolutional Layer):*

Đây là lớp quan trọng nhất của CNN, là lớp đầu tiên để trích xuất các tính năng, đặc trưng từ hình ảnh đầu vào. Ma trận điểm ảnh đầu vào sẽ được nhân tích chập với các ma trận lọc (thường có kích thước 3x3, 5x5, 7x7 và phải nhỏ hơn kích thước ma trận đầu vào) để tạo ra bản đồ đặc trưng hay feature map.

Công thức toán học của lớp tích chập:

$$(I * K)_{xy} = \sum_{i=1}^h \sum_{j=1}^w K_{ij} \times I_{x+i-1, y+j-1} \quad (2.1)$$

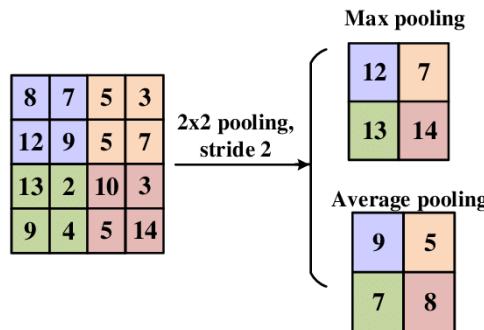
Với  $I$  là ma trận ảnh đầu vào,  $K$  là kích thước của ma trận lọc có chiều rộng  $w$  và chiều cao  $h$ .



Hình 2. 11: Thuật toán của lớp tích chập

❖ *Lớp gộp (Pooling Layer)*

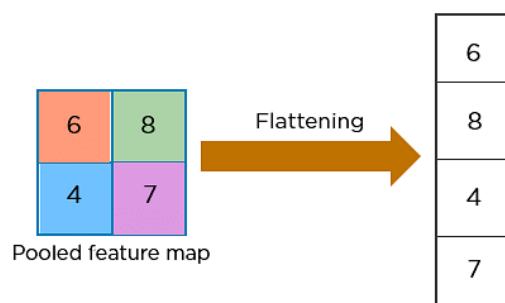
Sau khi có được bản đồ đặc trưng từ lớp tích chập ta cần thực hiện gộp để giảm số lượng thông số, giảm khối lượng tính toán. Có hai phương pháp gộp là: Max Pooling và Average Pooling.



Hình 2. 12: So sánh giữa Max Pooling và Average Pooling

❖ *Lớp làm phẳng (Flattening Layer)*

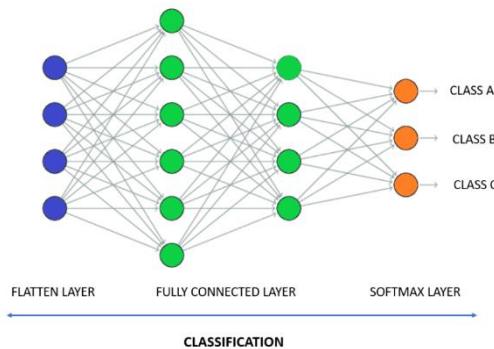
Đây là một lớp đơn giản được sử dụng để chuẩn hóa dữ liệu từ lớp trước đó làm đầu vào cho lớp cuối cùng – lớp kết nối đầy đủ (Fully Connected Layer).



Hình 2. 13: Quá trình làm phẳng

### ❖ Lớp kết nối đầy đủ (Fully Connected Layer)

Lớp cuối cùng thực hiện quá trình phân loại là lớp kết nối đầy đủ (Fully Connected Layer). Các giá trị từ lớp làm phẳng sẽ được đưa qua các lớp của mạng nơ-ron, mạng sẽ dự đoán ra xác suất và vị trí của từng đối tượng. Cuối cùng mô hình tính toán hàm loss function và sử dụng thuật toán lan truyền ngược để cập nhật trọng số với mục tiêu hàm loss function sẽ tiến về 0. Quá trình này được thực hiện đến khi hết chù kỳ.



Hình 2. 14: Cấu tạo của lớp kết nối đầy đủ

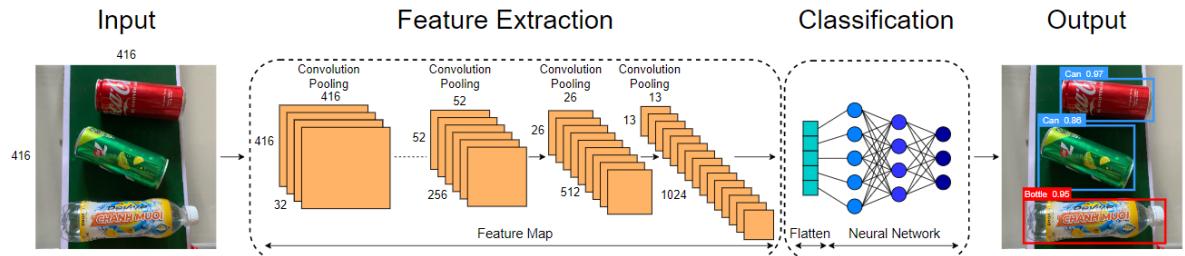
### 2.5.2 Mô hình YOLO (You Only Look Once)

Mô hình YOLO đã được giới thiệu bởi Joseph Redmon vào năm 2016 là một mô hình mạng CNN ứng dụng nhận dạng và phân loại đối tượng dựa vào dự đoán các bounding box. Về độ chính xác thì YOLO có thể không phải là thuật toán chính xác nhất nhưng là thuật toán nhanh nhất trong các mô hình phát hiện đối tượng. YOLO không chỉ là thuật toán phát hiện và dán nhãn cho bài toán phân loại mà còn xác định được vị trí của vật thể trong không gian.

Mô hình YOLO có rất nhiều phiên bản khác nhau và được cập nhật liên tục nhưng trong đề tài này nhóm sử dụng phiên bản YOLOv5 trong việc huấn luyện phân loại vì tính chính xác cũng như tốc độ xử lý của nó.

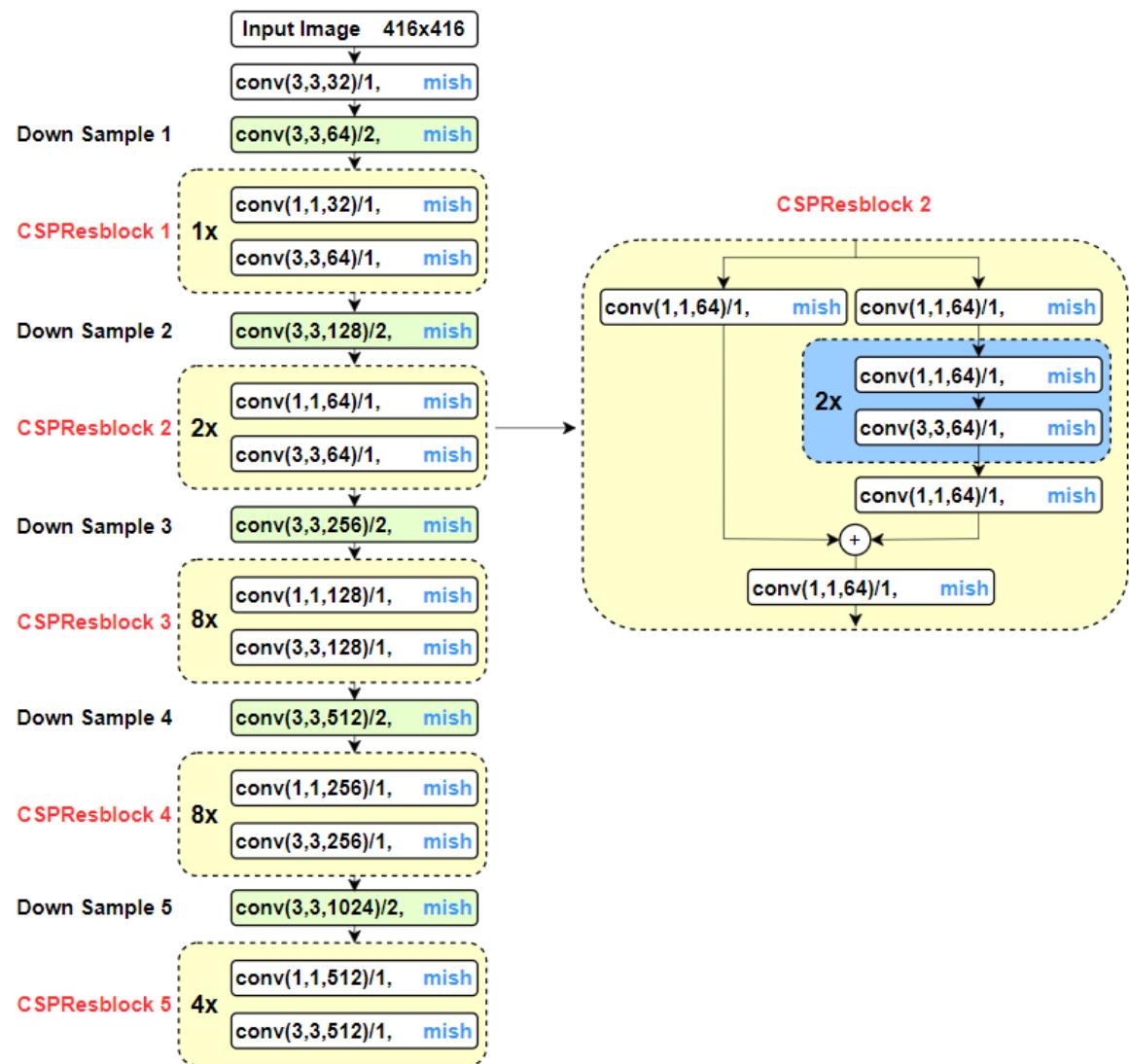
#### 2.5.2.1 Kiến trúc mô hình YOLO

Kiến trúc mô hình YOLOv5 cũng giống như kiến trúc mạng CNN bao gồm các lớp tích chập giúp trích xuất các feature map, các lớp gộp để giảm thông số tính toán, lớp làm phẳng để chuẩn bị đầu vào cho mạng nơ-ron và lớp kết nối đầy đủ sẽ dự đoán xác suất và vị trí của từng đối tượng.



Hình 2. 15: Cấu trúc mô hình YOLOv5

Cấu trúc trích xuất đặc trưng của mô hình YOLOv5 còn gọi là cấu trúc CSPDarknet53 gồm 6 lớp convolution, 5 lớp pooling và 5 lớp CSPResblock.



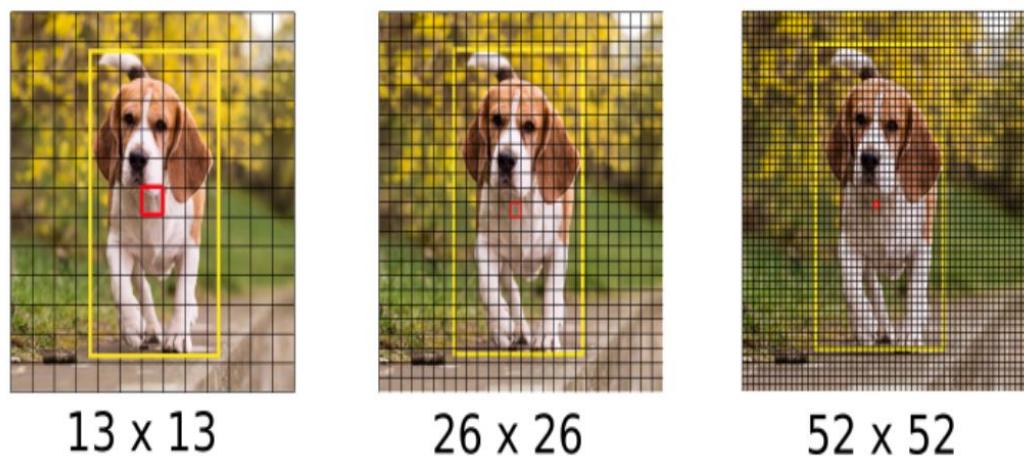
Hình 2. 16: Cấu trúc trích xuất đặc trưng của YOLOv5

### 2.5.2.2 Đầu vào của YOLOv5

YOLOv5 hỗ trợ các đầu vào ảnh với 2 kích thước 416x416 và 608x608. Khi ảnh được đưa vào sẽ thu được các feature map với kích thước nhỏ hơn để dự đoán vật thể trên từng ô. Với đầu vào 416x416 thì mô hình sẽ dự đoán trên 3 feature map 13x13, 26x26 và 52x52, còn với đầu vào 608x608 thì mô hình sẽ dự đoán trên 3 feature map 19x19, 38x38 và 72x72.

### 2.5.2.3 Feature map

Mô hình YOLOv5 phát hiện vật thể dựa trên 3 bản đồ đặc trưng hay feature map khác nhau từ lớn tới nhỏ. Những feature map có kích thước nhỏ giúp dự báo được các đối tượng có kích thước lớn. Ngược lại những feature map có kích thước lớn sẽ giúp dự báo các đối tượng kích thước nhỏ.

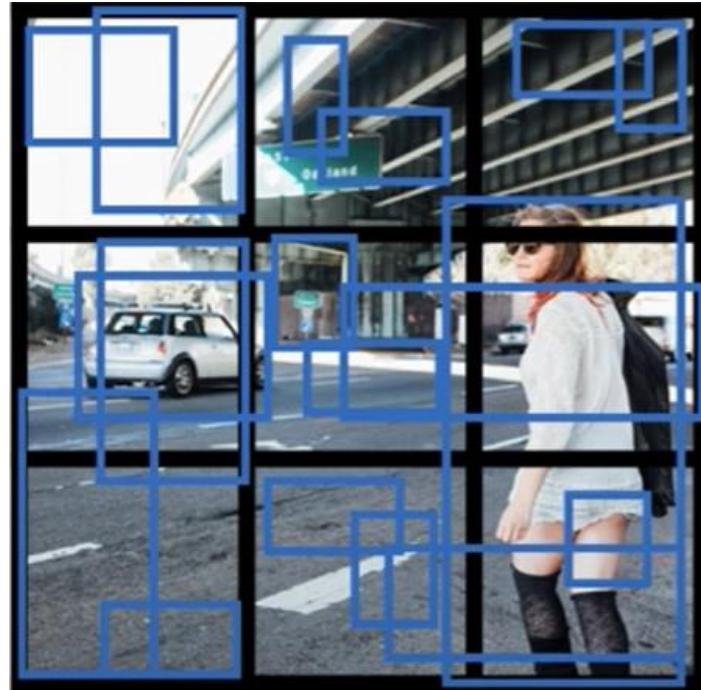


Hình 2.17: Các feature map của mô hình YOLOv5 với đầu vào ảnh 416x416

#### 2.5.2.4 Anchor box

Sau khi có được các feature map với các kích thước khác nhau YOLOv5 sẽ tạo ra các anchor box trong từng ô của feature map. Dựa vào các anchor box này thuật toán regression bounding box sẽ dự đoán ra vị trí đối tượng và đóng khung lại, các khung này còn gọi là bounding box của vật thể.

Số lượng anchor box sẽ được quy định, nếu số lượng anchor box trong từng ô của feature map càng nhiều thì sẽ giúp cho việc phân loại càng chính xác tuy nhiên sẽ làm giảm tốc độ phân loại. Thường thì số lượng anchor box được chọn cho mỗi ô sẽ là 3 anchor box với kích thước khác nhau.



Hình 2. 18: Anchor box trong feature map

#### 2.5.2.5 Bounding box

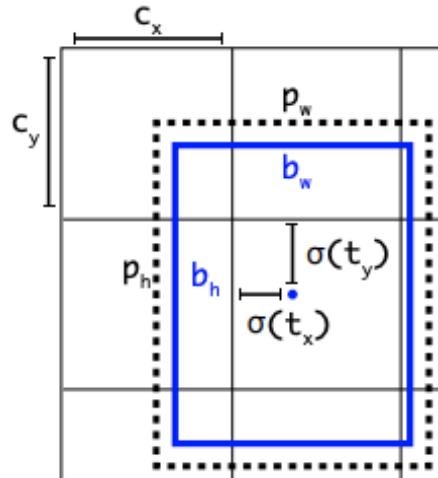
Sau khi đã chọn được các anchor box cho từng ô, thuật toán sẽ dự đoán ra các bounding box cho mỗi ô dựa trên những anchor box. Với anchor box có kích thước  $(p_w, p_h)$  tại ô  $(c_x, c_y)$  thuật toán sẽ dự đoán ra bounding box có kích thước  $(b_w, b_h)$  và tâm  $(b_x, b_y)$  dựa vào công thức:

$$\begin{aligned}
 b_x &= 2\sigma(t_x) - 0.5 + c_x \\
 b_y &= 2\sigma(t_y) - 0.5 + c_y \\
 b_w &= p_w (2\sigma(t_w))^2 \\
 b_h &= p_h (2\sigma(t_h))^2
 \end{aligned} \tag{2.2}$$

Trong đó:

$(t_x, t_y)$ : là khoảng cách từ tâm anchor box đến góc trên bên trái của ô

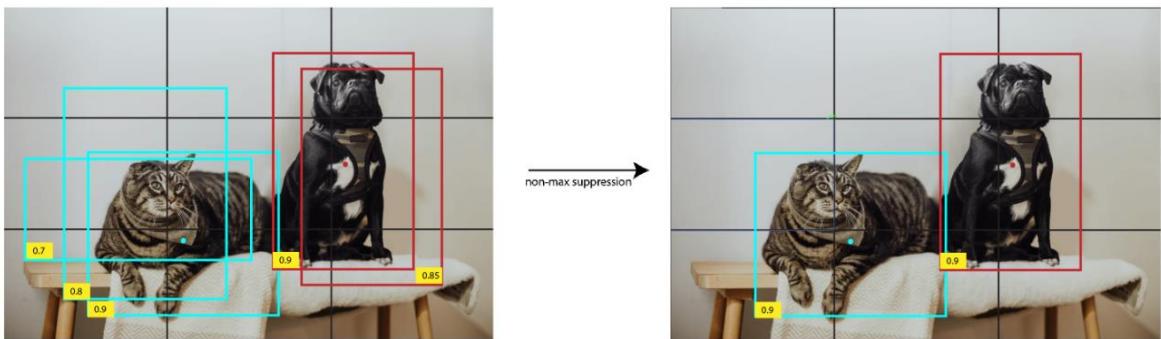
$(t_w, t_h)$ : là tỉ lệ với anchor box



Hình 2. 19: Công thức dự đoán Bounding box từ Anchor box

#### 2.5.2.6 Non-max suppression

Sau khi đã dự đoán thành công các bounding box của vật thể cho từng ô thì lúc này sẽ có rất nhiều bounding box bị trùng lặp hoặc không liên quan. Trong trường hợp đó thuật toán Non-max suppression sẽ được thực hiện để loại bỏ những bounding box với xác suất thấp hoặc trùng lặp chỉ giữ lại duy nhất 1 bounding box.



Hình 2. 20: Thuật toán Non-max suppression

#### 2.5.2.7 Loss Function

Hàm loss function là hàm đo độ sai số của mô hình trong quá trình huấn luyện. Mục tiêu của việc huấn luyện là tìm ra bộ thông số phù hợp nhất để hàm loss function tiến về 0 nghĩa là mô hình đã được huấn luyện tốt.

Hàm loss function của YOLO chia thành 2 thành phần:  $L_{loc}$  (localization loss) đo sai số vị trí của bounding box dự đoán và  $L_{cls}$  (confidence loss) đo sai số của phân phối xác suất.

$$L_{loc} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B l_{ij}^{obj} \left[ (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \right] \quad (2.3)$$

$$L_{cls} = \sum_{i=0}^{S^2} \sum_{j=0}^B \left[ l_{ij}^{obj} + \lambda_{noobj} (1 - l_{ij}^{obj}) \right] \left[ C_{ij} - \hat{C}_{ij} \right]^2 + \sum_{i=0}^{S^2} \sum_{c \in C} l_i^{obj} \left[ p_i(c) - \hat{p}_i(c) \right]^2 \quad (2.4)$$

$$L = L_{loc} + L_{cls} \quad (2.5)$$

Trong đó:

- $l_{ij}^{obj}$ : Hàm chỉ số cho biết bounding box thứ  $j$  của ô  $i$  có phải bounding box của vật thể được dự đoán.
- $l_i^{obj}$ : Hàm chỉ số có giá trị 0 hoặc 1, nhằm xác định ô  $i$  có chứa vật thể hay không, bằng 1 nếu chứa vật thể và ngược lại.
- $C_{ij}$ : Điểm tin cậy của ô  $i$ .
- $\hat{C}_{ij}$ : Điểm tự tin dự đoán.
- $p_i(c)$ : Xác suất của ô  $i$  có chứa đối tượng của lớp  $c$  trong tập  $C$ .
- $\hat{p}_i(c)$ : Xác suất có điều kiện dự đoán.

#### 2.5.2.8 Đầu ra của YOLOv5

Đầu ra mô hình YOLO là một vector:

$$y^T = [p_0, (t_x, t_y, t_w, t_h), (p_1, p_2, \dots, p_n)] \quad (2.6)$$

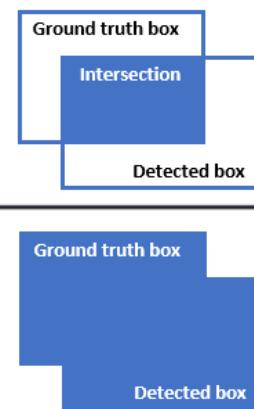
Trong đó:

- $p_0$  là tỉ lệ có vật thể trong bounding box.
- $(t_x, t_y, t_w, t_h)$  lần lượt là vị trí tọa độ tâm, chiều rộng, chiều cao của bounding box.
- $(p_1, p_2, \dots, p_n)$  là vector dự đoán sẽ có đối tượng nào trong bounding box.
- Cứ mỗi 1 anchor box sẽ cho ra  $(n + 5)$  đầu ra ( $n$  là số đối tượng huấn luyện). Nếu ta sử dụng 3 anchor box cho mỗi ô và với feature map có kích thước  $13 \times 13$  thì số lượng output sẽ là:  $13 \times 13 \times 3 \times (n+5)$ .

### 2.5.3 Phương pháp đánh giá mô hình phát hiện đối tượng

#### 2.5.3.1 Intersection Over Union (IOU)

Intersection Over Union (IOU) là chỉ số đánh giá được sử dụng để đo độ chính xác của mô hình phân loại đối tượng. IOU là tỷ lệ giữa giao và hợp của Ground truth box hay bounding box thực tế và Detected box hay bounding box dự đoán. IOU càng lớn thì chứng tỏ mô hình phân loại càng tốt, đồng nghĩa với việc bounding box dự đoán gần giống với bounding box thực tế.

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} = \frac{\text{Intersection}}{\text{Union}}$$


Hình 2. 21: Công thức tính IOU

Với:

- Ground truth box là bounding box thực tế đã được dán nhãn.
- Detected box là bounding box được mô hình dự đoán ra.

#### 2.5.3.2 Confusion Matrix

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Hình 2. 22: Confusion Matrix cho hai đối tượng

Confusion Matrix là một phương pháp đánh giá kết quả của những bài toán phân loại với việc xem xét cả những chỉ số về độ chính xác và độ bao quát của các dự đoán cho từng đối tượng. Với bài toán nhận dạng với 2 đối tượng là Positive và Negative, Confusion Matrix có 4 thành phần:

- True Positive (TP): các điểm positive được nhận đúng là positive
- False Positive (FP): các điểm negative thực được nhận sai là positive
- False Negative (FN): các điểm positive được phát hiện sai là negative
- True Negative (TN): các điểm negative được nhận đúng là negative

Theo 4 thông số này được dùng để tính toán các chỉ số đánh giá phân loại như Precision và Recall. Cả 2 thông số này đều quan trọng trong mô hình đánh giá phân loại, tùy thuộc vào mục đích ứng dụng của người huấn luyện mà ưu tiên thông số nào hơn.

### 2.5.3.3 Precision, Recall và F1-score

#### Precision:

Precision được định nghĩa là tỉ lệ số điểm true positive (TP) trong số những điểm được phân loại là positive (TP+FP). Được dùng để đánh giá xem trong những dự đoán positive mà mô hình đã thực hiện có bao nhiêu là đúng.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2.7)$$

#### Recall:

Recall được định nghĩa là tỉ lệ số điểm true positive (TP) trong số những điểm thực sự là positive (TP+FN). Được dùng để đánh giá trong những điểm thực sự là positive thì mô hình dự đoán đúng bao nhiêu. Đây là chỉ số cho thấy độ bao quát của mô hình.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2.8)$$

#### F1-score:

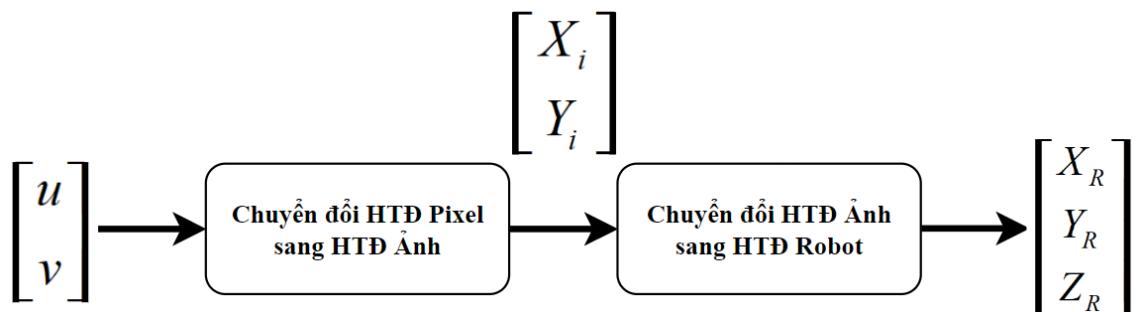
Precision và Recall càng cao thì mô hình huấn luyện càng tốt, tuy nhiên trong thực tế 2 chỉ số này thường đối lập nhau không thể cùng tăng hoặc cùng giảm. Vì vậy ta cần 1 chỉ số thứ 3 để đánh giá tổng quan mô hình là F1-score. F1-score càng cao thì mô hình càng tốt, lý tưởng nhất là F1-score = 1 (khi Precision = 1 và Recall = 1).

$$F1 = 2 \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.9)$$

## 2.6 Phương pháp chuyển đổi hệ tọa độ xác định vị trí đối tượng

Mục đích của việc chuyển đổi hệ tọa độ là lấy được vị trí của vật thể từ tọa độ camera đem chuyển sang tọa độ Robot để điều khiển Robot di chuyển đến tiếp cận đối tượng và xử lý.

Có rất nhiều phương pháp chuyển đổi hệ tọa độ camera sang hệ tọa độ Robot. Nhóm đã chọn phương pháp chuyển đổi dựa theo mô hình thực tế, đây là phương pháp chuyển đổi dựa trên vị trí các thiết bị trong thực tế chứ không dựa vào các thông số có sẵn của camera.



Hình 2. 23: Quy trình chuyển đổi hệ tọa độ

❖ *Ưu điểm của phương pháp:*

- Phương trình chuyển đổi đơn giản, có thể dễ dàng chỉnh sửa khi muốn thay đổi vị trí camera và Robot.
- Không cần người lập trình phải có kiến thức về xử lý các ma trận thông số của camera.
- Vì không phải dựa vào thông số của camera nên độ chính xác tương đối cao.

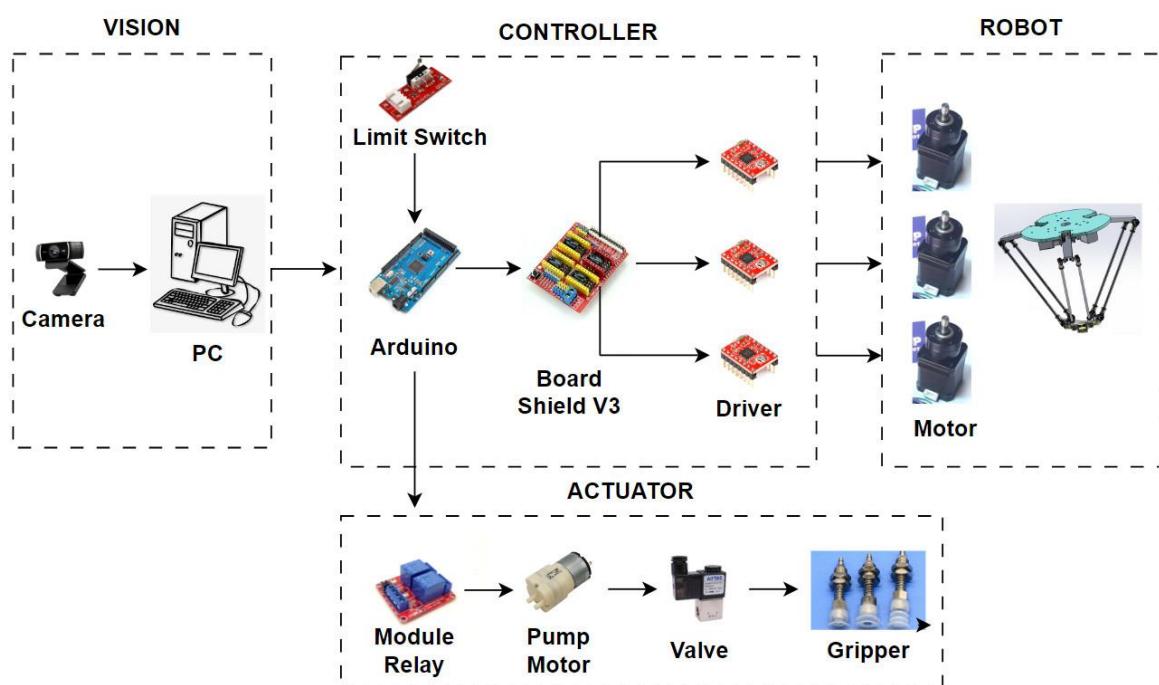
❖ *Nhược điểm của phương pháp:*

- Vì là phương pháp chuyển đổi dựa theo mô hình thực tế nên không thể áp dụng phương pháp cho các mô hình khác. Đây là phương pháp riêng biệt.
- Khi muốn sử dụng phương pháp cho mô hình khác phải tiến hành lấy thông số và tính toán lại từ đầu.

## CHƯƠNG 3: THIẾT KẾ VÀ THI CÔNG PHẦN CỨNG

Chương này trình bày tổng quan về hệ thống, yêu cầu thiết kế mô hình và các chi tiết của mô hình Robot Delta được thiết kế trên phần mềm Solidworks. Tiếp đến nhóm tiến hành thi công lắp ráp các chi tiết lại với nhau thành một mô hình hoàn chỉnh, lắp ráp tủ điện đảm bảo an toàn và thẩm mỹ. Liệt kê các thiết bị điện và cơ cấu phân loại được sử dụng trong đề tài.

### 3.1 Tổng quan về hệ thống



Hình 3. 1: Sơ đồ khái quát tổng quan hệ thống

**Khối Vision:** Thực hiện nhiệm vụ lấy thông tin đầu vào dưới dạng hình ảnh thông qua camera sau đó gửi đến trung tâm xử lý tín hiệu đầu tiên là máy tính. Mô hình YOLO sẽ phân loại và trích xuất ra vị trí đối tượng gửi cho khối điều khiển.

**Khối điều khiển:** Sau khi có được vị trí đối tượng khối điều khiển ở đây là Arduino sẽ tính toán và cấp xung để điều khiển động cơ và các thiết bị khác phối hợp theo quy trình để thực hiện phân loại.

**Khối Robot:** Gồm có các công tắc hành trình dùng để đặt vị trí home và Robot Delta sẽ di chuyển tiếp cận đối tượng thực hiện phân loại.

**Khối cơ cấu chấp hành:** Gồm Module Relay điều khiển bật tắt động cơ bơm hút chân không và van 2 chiều. Động cơ bơm hút chân không sẽ tạo ra khí hút để giữ

chặt vật đồng thời khi Robot di chuyển đến vị trí cuối thì van sẽ cho khí thoát ra và vật sẽ rơi xuống.

#### 3.2 Yêu cầu thiết kế phần cứng

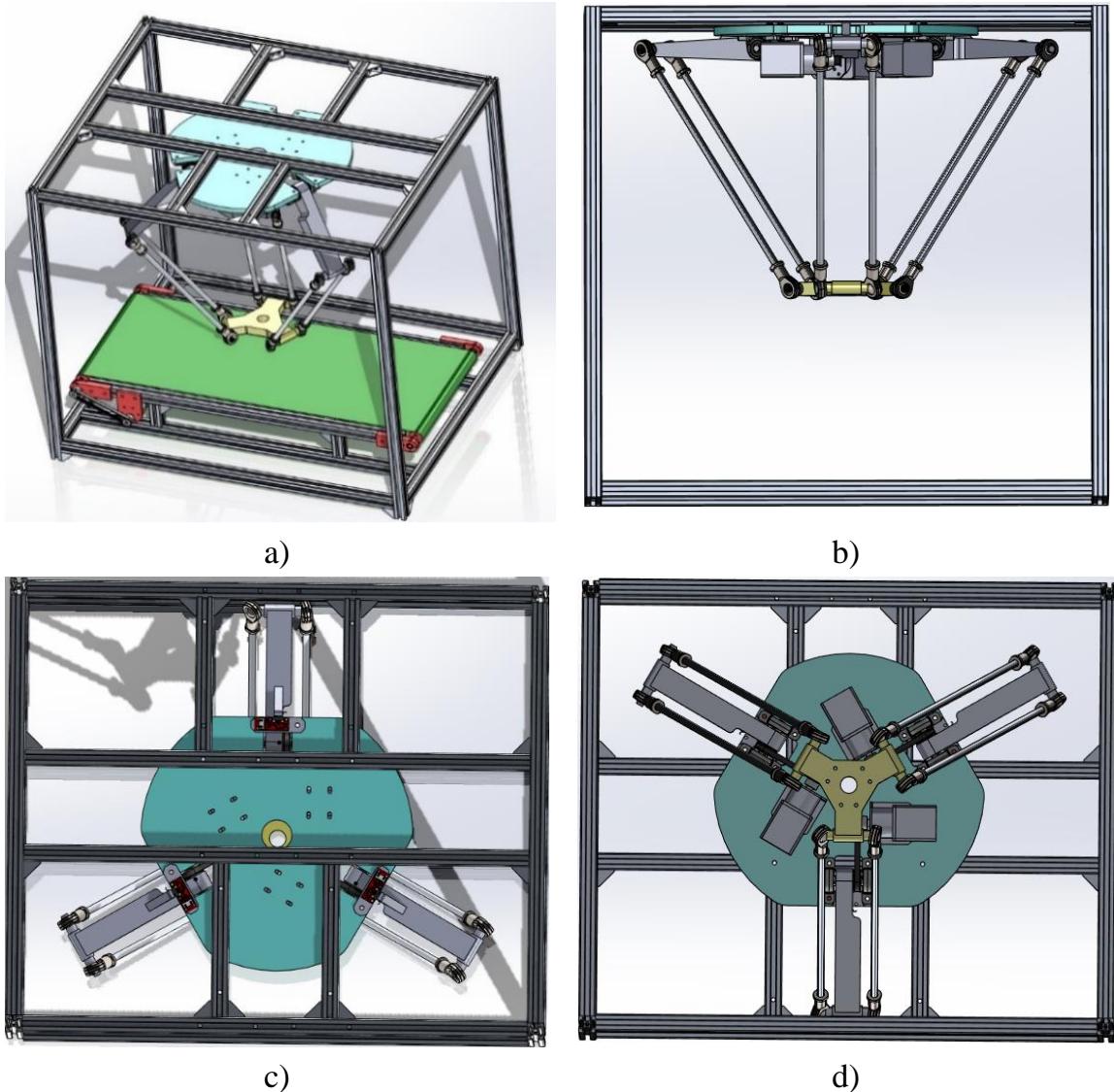
Mục tiêu ban đầu của hệ thống mà nhóm đề ra là xây dựng một mô hình phân loại cho hai đối tượng chai nhựa và lon. Dựa vào mô hình đã được thiết kế trên Solidworks, nhóm đã tiến hành gia công và lắp ráp các bộ phận robot lại với nhau. Dưới đây là một số yêu cầu khi thiết kế mô hình Robot Delta phân loại rác thải:

- Thiết kế mô hình Robot Delta hoạt động trong không gian làm việc của khung 600x500x600 (mm) chính xác, ổn định và đáp ứng được yêu cầu phân loại.
- Bố trí các thiết bị hợp lý (băng tải, camera) cho ứng dụng phân loại rác thải.
- Đầu nối đi dây khí nén, cơ cấu động cơ bơm hút chân không, van 2 chiều để hút thả vật.
- Lắp đặt tủ điện một cách hợp lý và mang tính thẩm mỹ, đảm bảo tính an toàn và thuận tiện trong việc sửa chữa, thay đổi thiết bị.

#### 3.3 Thiết kế mô hình trên phần mềm Solidworks

Nhóm đã tham khảo một số mô hình Robot Delta kinh điển trên thế giới cũng như của các đề tài đồ án tốt nghiệp trong trường Đại Học Sư Phạm Kỹ Thuật TP.Hồ Chí Minh. Nhóm đã tiến hành nghiên cứu, tính toán và đưa ra các thông số phù hợp cho mô hình Robot Delta ứng dụng trong việc phân loại rác thải. Vì đây là một ứng dụng yêu cầu về tốc độ và độ chính xác cao mà Robot là bộ phận chịu trách nhiệm hoạt động chủ đạo nên cần được thiết kế tỉ mỉ, chính xác, hợp lý và chắc chắn, đảm bảo hiệu suất làm việc trong thời gian dài.

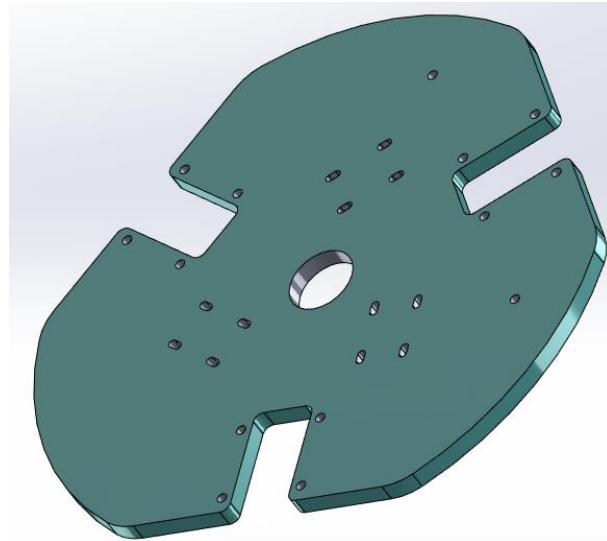
Mô hình Robot Delta bao gồm các bộ phận chính như: đế cố định, đế di động, khớp trên, khớp hình bình hành, khung Robot và một số chi tiết khác để kết hợp hình thành một mô hình Robot hoàn chỉnh. Tất cả các bộ phận phải được thiết kế và thi công chính xác mới đảm bảo mô hình Robot hoạt động ổn định và an toàn.



Hình 3. 2: Mô hình 3D Robot Delta. a) Góc nhìn tổng quan; b) Góc nhìn từ ngoài vào; c) Góc nhìn từ trên xuống; d) Góc nhìn từ dưới lên

### 3.3.1 Đế cố định Robot

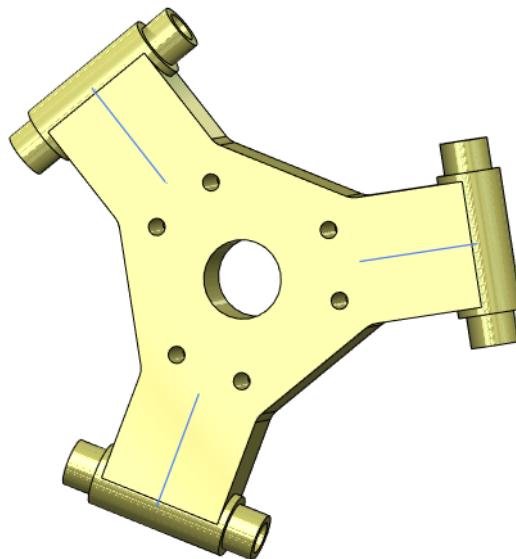
Đế cố định được dùng để kết nối với phần khung và để gắn 3 động cơ tương ứng với 3 chuỗi động học của Robot. Khi Robot vào trạng thái hoạt động, khối lượng của đối tượng sẽ được chia đều cho 3 cánh tay. Do đó, đế cố định phải chịu trọng lượng khá lớn của đối tượng và của các khớp nên phải thiết kế cứng cáp và dày với chất liệu nhôm và độ dày 10mm.



Hình 3. 3: Mô hình 3D đế cố định Robot

### 3.3.2 Đế di động Robot

Đế di động là chi tiết gắn cơ cấu chấp hành ti giác hút để thực hiện việc phân loại. Vì mục tiêu đề tài của nhóm là phân loại chai nhựa và lon, đối tượng cần phân loại có tải khá nhẹ nên yêu cầu thiết kế của đế chuyển động thì có độ cứng vừa phải và nhẹ để giảm tải cho cả Robot nên nhóm thiết kế và chọn gia công chất liệu nhựa PLA, với độ dày 16mm.

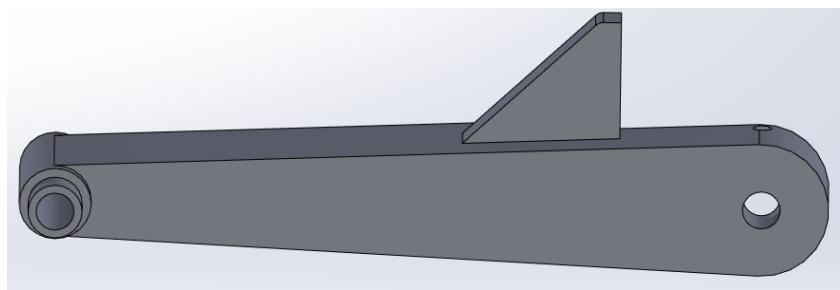


Hình 3. 4: Mô hình 3D đế di động Robot

### 3.3.3 Khớp trên Robot

Khớp trên Robot được sử dụng làm khâu truyền động, được điều khiển trực tiếp từ động cơ thông qua dây đai. Do đó, yêu cầu thiết kế của khớp trên này là trọng lượng nhẹ để giảm tải bót mô-men quán tính, bên cạnh đó thì vẫn đủ độ cứng để giảm độ đàn hồi và rung lắc trong quá trình phân loại. Với mong muốn trọng lượng nhẹ nên nhóm đã với chất liệu nhựa PLA.

Thêm vào đó, nhóm đã thiết kế thêm cho khớp trên Robot một khối hình tam giác để khi Robot quay lên điểm cao nhất là 3 góc về 0 (vị trí home) thì đụng vào công tắc hành trình, giúp cho Robot dừng lại và không bị quay quá khớp, tránh hư hỏng cho Robot.



Hình 3. 5: Mô hình 3D khớp trên Robot

### 3.3.4 Khớp hình bình hành Robot

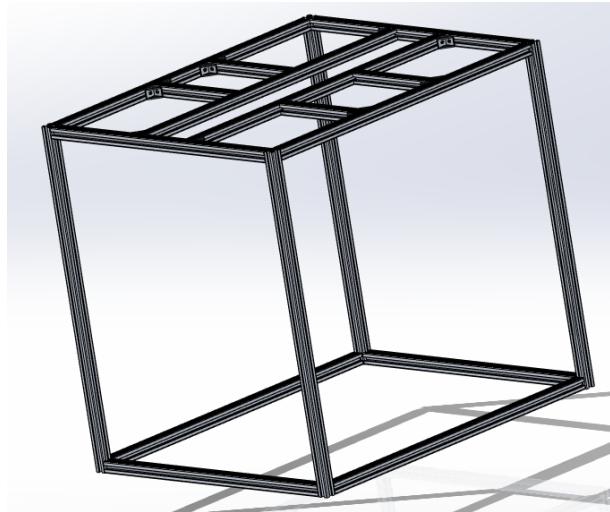
Khớp hình bình hành là chi tiết để nối cánh tay Robot với để chuyển động thông qua vòng bi mắt trâu SI8T. Vì thanh trực là khâu thụ động, chịu tác động lực từ cánh tay Robot và để chuyển động nên cần thiết kế và gia công chất liệu inox để đảm bảo độ cứng cáp và với chất liệu này thì thanh sẽ khó bị gỉ khi để ngoài môi trường.



Hình 3. 6: Mô hình 3D khớp hình bình hành Robot

### 3.3.5 Khung Robot

Robot Delta được gắn lên khung, do đó yêu cầu thiết kế khung phải đảm bảo độ cứng, vững và cân bằng để giúp robot hoạt động ổn định trong suốt quá trình phân loại. Khung Robot nhôm đã thiết kế với kích thước 600 x 500 x 600 mm với chất liệu nhôm định hình.



Hình 3. 7: Khung Robot

### 3.3.6 Một số chi tiết khác

Để hoàn thiện một mô hình Robot Delta, ngoài bốn chi tiết chính như: Đế cố định, khớp trên, khớp hình bình hành và đế di động. Bên cạnh đó còn có thêm một số linh kiện chi tiết khác được liệt kê theo bảng bên dưới.

Bảng 3. 1: Một số chi tiết khác của Robot Delta

STT	Tên chi tiết	Hình ảnh	Thông tin chi tiết
1	Vòng bi mắt trâu SI8T		+ Ren trong M8, trục 8mm
2	Gối đỡ vòng bi KP08		+ Trục: 8mm

3	Puly GT2-24mm		+ Trục 8mm + Số răng: 40 răng + Đường kính bánh răng: 24.8mm
4	Puly GT2-12mm		+ Trục 8mm + Số răng: 20 răng + Đường kính bánh răng: 12mm
5	Dây đai 2GT-200		+ Số răng: 100 + Rộng: 6mm + Kích thước: 200mm

### 3.4 Lựa chọn thiết bị

#### 3.4.1 Động cơ

Mục tiêu đề tài thực hiện điều khiển Robot Delta phân loại rác, nhóm đã lựa chọn động cơ bước giảm tốc Nema 17 tích hợp hộp số giảm tốc với tỉ số truyền 57/11 giúp tăng lực kéo và độ chính xác của động cơ. Sử dụng thích hợp cho các Robot chịu tải lớn và có độ chính xác cao như cánh tay Robot 3 bậc, Robot Scara,...động cơ có độ bền và độ ổn định cao.

Bảng 3. 2: Động cơ Nema 17

	+ Tỉ số giảm tốc qua hộp số: 57/11 + Điện áp cấp tối đa: 24VDC + Dòng điện tối đa 1.7A + Momen xoắn: 30kgf.cm + Độ phân giải bước: 1.8 độ
--	---

#### 3.4.2 Thiết bị điện

Nhằm bảo vệ các thiết bị điện trong quá trình hoạt động để tránh việc ngắn mạch hay chập điện, nhóm đã sử dụng CB để đóng ngắt an toàn mạch.

Trong đề tài nhóm sử dụng bộ nguồn 24V 10A và bộ nguồn 12V 5A để điều khiển các thiết bị khác nhau như: băng tải, động cơ bơm hút chân không, van 2 chiều, board shield.

### Chương 3: Thiết kế và thi công phần cứng

Bảng 3. 3: Bảng danh sách các thiết bị điện

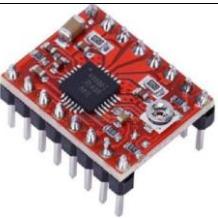
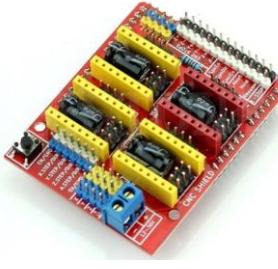
STT	Tên gọi	Hình ảnh	Thông tin
1	Mitsubishi CP30-BA 2P 5A		<ul style="list-style-type: none"> <li>+ Số cực: 2P</li> <li>+ Dòng định mức: 5A</li> <li>+ Điện áp định mức: 250VAC/125VDC</li> <li>+ Ngắt dòng ngắn mạch 2.5KA</li> </ul>
2	Nguồn 24V 10A		<ul style="list-style-type: none"> <li>+ Điện áp vào: 110VAC-220VAC</li> <li>+ Điện áp ra: 24VDC</li> <li>+ Đầu ra: 3 cặp</li> <li>+ Dòng out Max 10A</li> </ul>
3	Nguồn 12V 5A		<ul style="list-style-type: none"> <li>+ Điện áp ngõ ra: 185-260VAC</li> <li>+ Điện áp ra: 12VDC</li> <li>+ Đầu ra: 1 cặp</li> <li>+ Dòng ngõ ra 5A</li> </ul>
4	Đèn báo AD16-22DS 220VAC		<ul style="list-style-type: none"> <li>+ Điện áp: 220VAC</li> <li>+ Dòng điện max: 20mA</li> <li>+ Đường kính: 22mm</li> <li>+ Màu sắc: vàng</li> </ul>
5	Đèn báo AD16-22DS 24V		<ul style="list-style-type: none"> <li>+ Điện áp: 24VDC</li> <li>+ Dòng điện max: 20mA</li> <li>+ Đường kính: 22mm</li> <li>+ Màu sắc: xanh lá</li> </ul>
6	Nút dừng khẩn cấp LAY37-11ZS 1NO-1NC		<ul style="list-style-type: none"> <li>+ Điện áp: 660V</li> <li>+ Dòng điện: 10A</li> <li>+ Đường kính ren: 22mm</li> <li>+ 2 tiếp điểm 1NO và 1NC</li> </ul>

### 3.4.3 Thiết bị điều khiển

Để điều khiển động cơ bước và các thiết bị nhóm sử dụng: Arduino Mega 2560, Driver A4988 và board shield V3.

Công tắc hành trình giúp Robot Set home mà không phải quá khó khăn. Relay điều khiển: Van, băng tải và động cơ bơm hút chân không.

Bảng 3.4: Bảng danh sách các thiết bị điều khiển

STT	Tên gọi	Hình ảnh	Thông tin
1	Driver A4988		<ul style="list-style-type: none"> <li>+ Điện áp cấp cực đại 35V</li> <li>+ Dòng cấp liên tục cho mỗi pha cực đại 2A</li> <li>+ Độ phân giải: full, 1/2, 1/4, 1/8 và 1/16</li> </ul>
2	Arduino Mega 2560		<ul style="list-style-type: none"> <li>+ Điện áp hoạt động 5V</li> <li>+ Số lượng chân I/O: 54</li> <li>+ Điện áp đầu vào: 7-12V</li> </ul>
3	Module Công tắc hành trình		<ul style="list-style-type: none"> <li>+ Điện áp cấp: 3-12V</li> <li>+ Đầu ra: VCC, GND, OUT</li> </ul>
4	Module Relay với Opto cách ly kích H/L		<ul style="list-style-type: none"> <li>+ Sử dụng điện áp 5VDC</li> <li>+ Điện thế đóng ngắt tối đa: 30VDC-10A hoặc 250VAC-10A</li> </ul>
5	Board CNC Shield V3		<ul style="list-style-type: none"> <li>+ Tương thích với arudino UNO R3 và Mega 2560</li> <li>+ Hỗ trợ tối 4 động cơ, cắm trực tiếp dây động cơ.</li> <li>+ Hỗ trợ gắn các công tắc hành trình.</li> <li>+ Hỗ trợ thiết lập độ phân giải vi bước động cơ bằng jump</li> </ul>

#### 3.4.4 Camera Logitech

Để xác định và phát hiện vật nhom lựa chọn sử dụng Camera webcam của Logitech bởi vì tính thông dụng trong việc kết nối với máy tính và đặc biệt là chi phí phải chăng. Bên cạnh đó, cũng sẽ bị hạn chế về chất lượng ảnh xử lý.

Bảng 3. 5: Thông tin về Camera Logitech C920

Hình ảnh	Thông tin	
	Tên thiết bị	Logitech C920
	Độ phân giải	1920x1080p
	Tốc độ khung hình	30fps
	Loại lấy nét	Tự động loại nét
	Góc xoay	78 độ
	Trọng lượng	162g
	Kích thước	43.3x94x71 mm

#### 3.4.5 Cơ cấu phân loại

Ngày nay chủ đề về phân loại các đối tượng đang được quan tâm và ứng dụng rộng rãi trong các nhà máy xí nghiệp. Bên cạnh đó các cơ cấu phân loại cũng được lưu ý nhiều bởi sự đa dạng về hình dạng và biên dạng của đối tượng. Có hai loại cơ cấu phân loại chính là hút thả và gấp vật.

Nhóm đã chọn lựa cơ cấu phân loại hút thả vật với các thiết bị được lựa chọn như sau:

### Chương 3: Thiết kế và thi công phần cứng

Bảng 3. 6: Các thiết bị cơ cấu khí nén

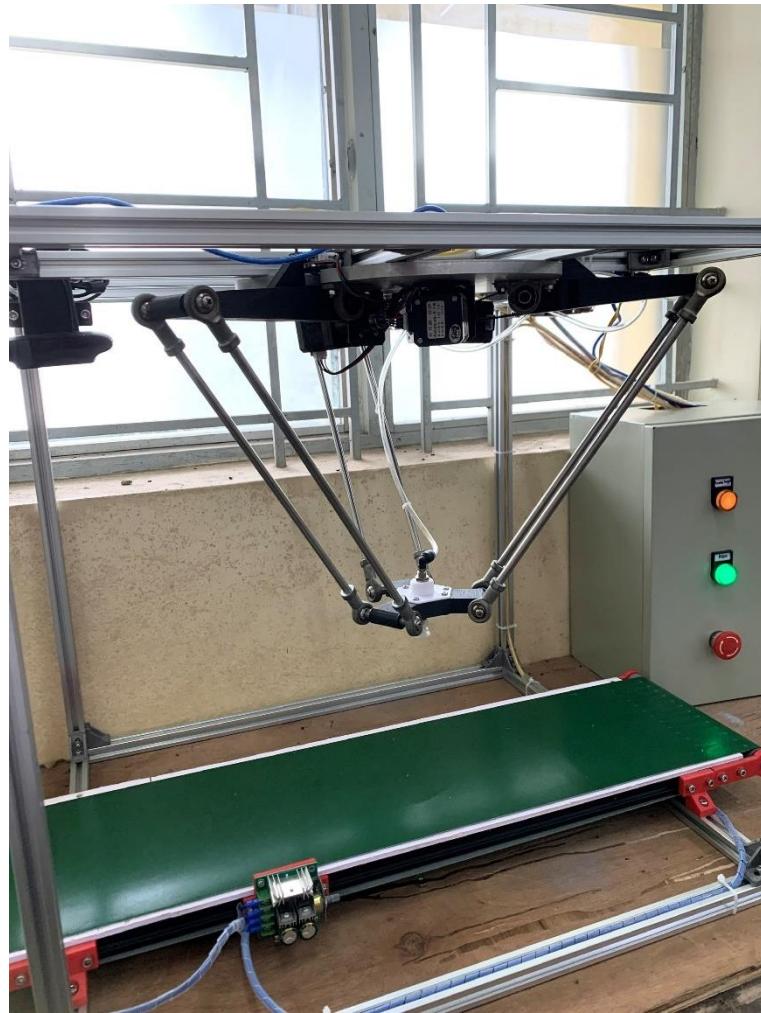
STT	Tên thiết bị	Hình ảnh	Thông tin
1	Động cơ bơm hút chân không YYP528-12A		<ul style="list-style-type: none"> <li>+ Điện áp hoạt động: 12VDC</li> <li>+ Dòng không tải: ~0.2A</li> <li>+ Dòng khí nén: &gt;0.4A</li> <li>+ Áp suất: 600~850 mmHg</li> <li>+ Dòng khí: 2.5~7L/phút (Lpm)</li> </ul>
2	Ti giác hút, đầu giác hút		<ul style="list-style-type: none"> <li>+ Dài: 70mm</li> <li>+ Đường kính giác hút: 20mm</li> <li>+ Ống đầu vào: 6mm</li> </ul>
3	Van 2 chiều Airtac 2V025-08		<ul style="list-style-type: none"> <li>+ Điện áp đầu vào: 24VDC</li> <li>+ Áp suất: 0-10kgf/cm2</li> </ul>
4	Ống dẫn khí		<ul style="list-style-type: none"> <li>+ Chất liệu PU</li> <li>+ Đường kính: 6mm</li> </ul>
5	Co nối ống khí		<ul style="list-style-type: none"> <li>+ Chất liệu nhựa</li> <li>+ Đường kính: 6mm</li> </ul>

### 3.5 Thi công phần cứng

#### 3.5.1 Các chi tiết của Robot

Bảng 3. 7: Các chi tiết sau gia công

STT	Hình ảnh thực tế sau gia công	Thông tin chi tiết
1		+ Đè có định + Kích thước 110mm + Chất liệu nhôm
2		+ Khớp trên + Kích thước 157mm + Chất liệu nhựa
3		+ Khớp hình bình hành + Kích thước 360mm + Chất liệu inox
4		+ Đè di động + Kích thước 60mm + Chất liệu nhựa



Hình 3. 8: Mô hình thực tế sau khi thi công

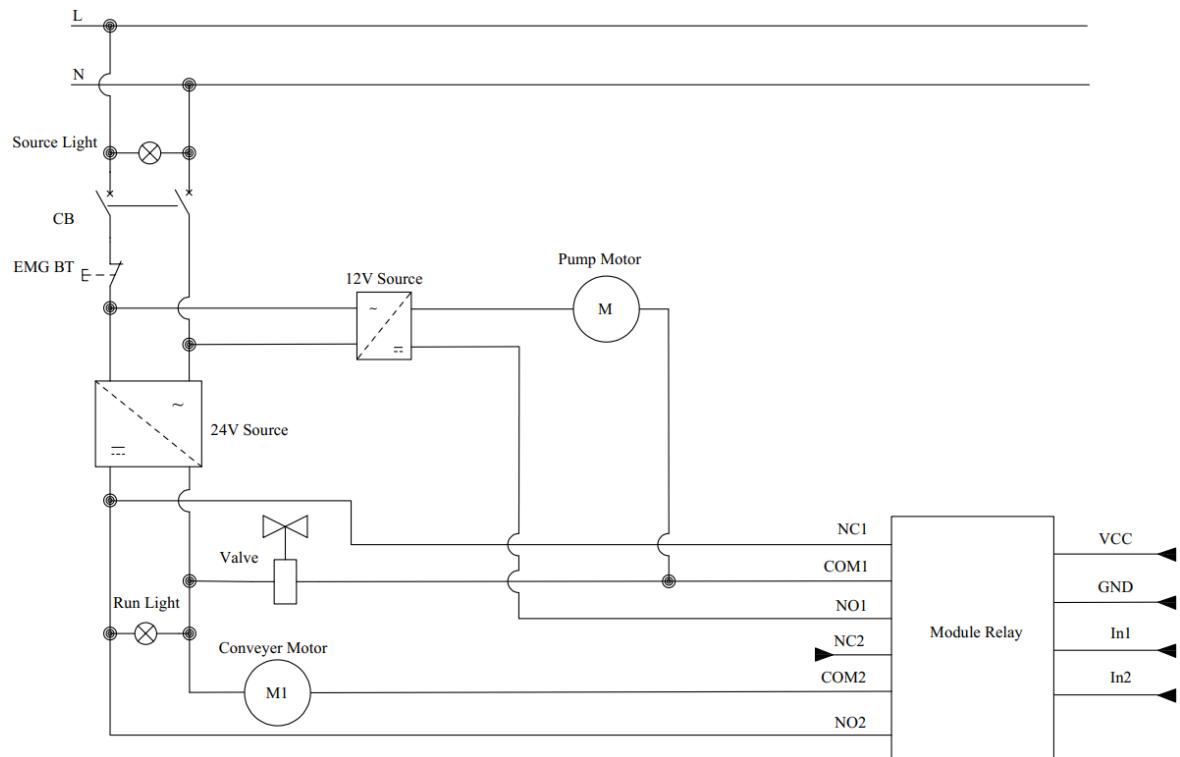
### 3.5.2 Băng tải

Băng tải có nhiệm vụ đưa đổi tượng cần phân loại là chai nhựa và lon vào vùng hoạt động và phát hiện của Robot.

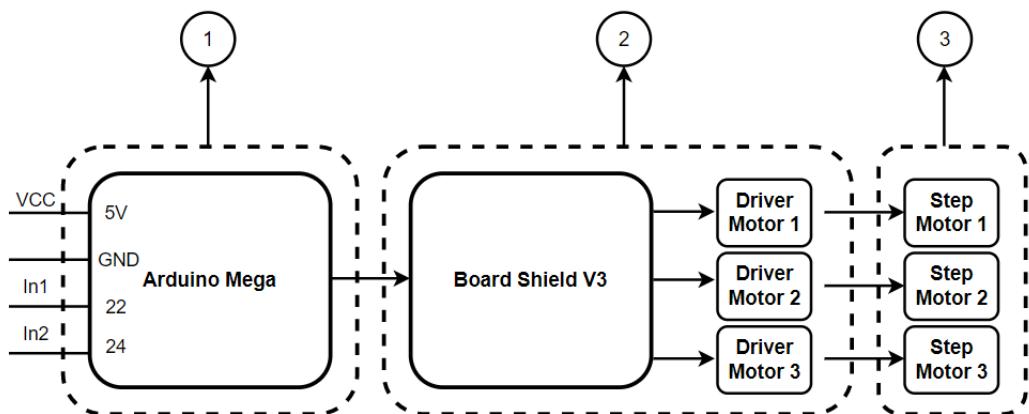
Bảng 3. 8: Thông số của băng tải

	<ul style="list-style-type: none"><li>+ Chiều dài băng tải: 800mm</li><li>+ Chiều rộng băng tải: 200mm</li><li>+ Sử dụng động cơ DC</li><li>+ Biến trờ thay đổi tốc độ băng tải</li></ul>
--	---

### 3.5.3 Sơ đồ hệ thống

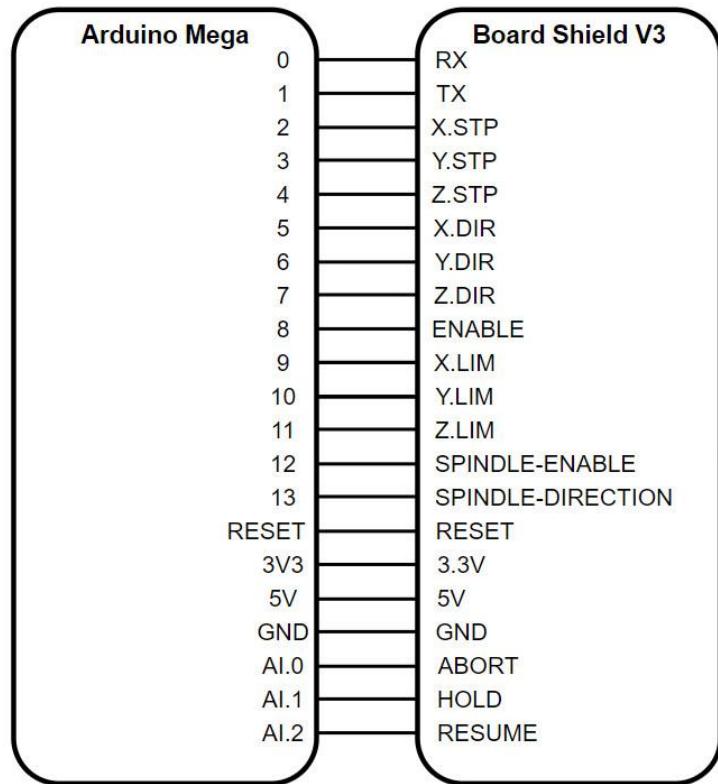


Hình 3. 9: Sơ đồ mạch động lực

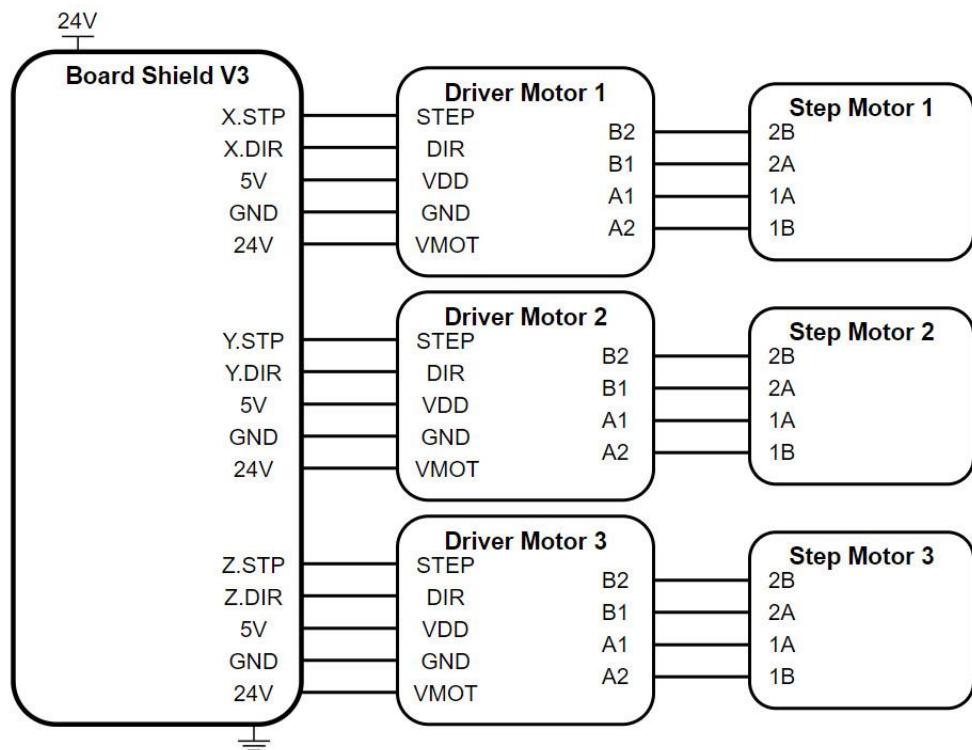


Hình 3. 10: Sơ đồ mạch điều khiển tổng quát

- Khối đầu tiên là: khối Arduino Mega điều khiển các thiết bị như: Valve 2 chiều, băng tải và động cơ bơm. Arduino sẽ cấp xung điều khiển cho 3 động cơ bước.
- Khối thứ hai mô tả sơ đồ kết nối giữa hai thiết bị điều khiển Board Shield V3 và Driver A4988 sẽ giúp điều khiển tốc độ và chiều quay của động cơ.
- Khối thứ ba khống động cơ bước.



Hình 3. 11: Sơ đồ kết nối Arduino-Board Shield V3



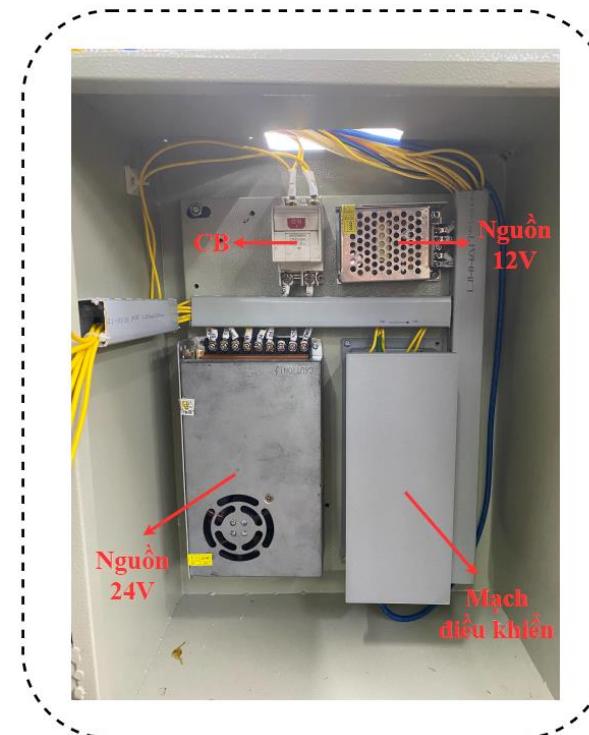
Hình 3. 12: Sơ đồ kết nối Board Shield V3-Driver-Động cơ bước

### 3.5.4 Thi công tủ điện

Bảng 3. 9: Hình ảnh trước và sau thi công tủ điện

Trước khi thi công	Sau khi thi công
	

Với các thiết bị được liệt kê nhóm đã tiến hành sắp xếp và lắp ráp vào tủ điện và đi dây kết nối các thiết bị một cách gọn gàng, đúng sơ đồ đấu nối. Kết quả thu được một tủ điện mang tính thẩm mỹ, an toàn và dễ dàng sửa chữa khi gặp sự cố.



Hình 3. 13: Các thiết bị bên trong tủ điện

## CHƯƠNG 4: TÍNH TOÁN ĐỘNG HỌC VÀ THIẾT KẾ PHẦN MỀM ĐIỀU KHIỂN

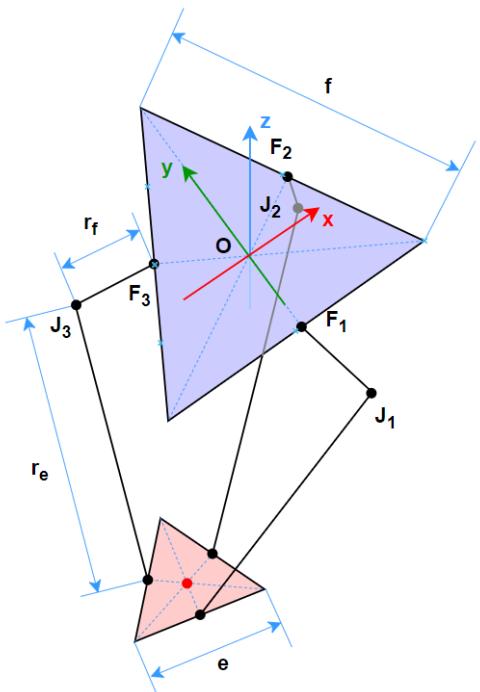
Trong chương này trình bày về các thông số của Robot Delta, tính toán động học thuận nghịch và tìm không gian làm việc trên phần mềm Matlab. Tiếp đến là thiết kế phần mềm điều khiển và giao diện người dùng trên phần mềm Visual Studio Code. Cuối cùng trong chương là mô tả quy trình hoạt động và lưu đồ giải thuật hệ thống.

### 4.1 Các thông số của Robot Delta

Trước khi tính toán động học ta phải xác định 1 số thông số cố định của Robot Delta bao gồm:  $f$ ,  $e$ ,  $r_f$ ,  $r_e$ . Đây là 4 thông số chính của các bộ phận Robot Delta.

Bảng 4. 1: Thông số các chi tiết Robot

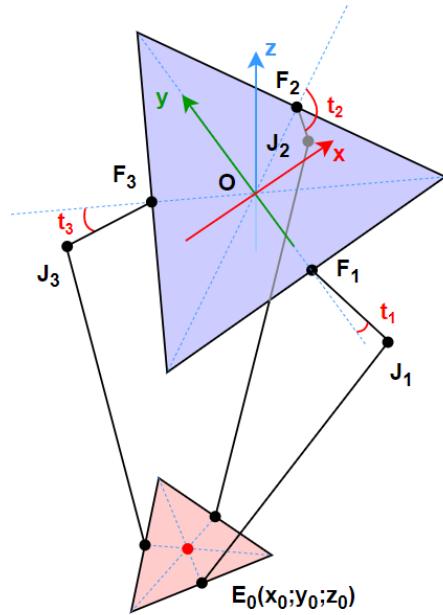
Tên bộ phận	Ký hiệu	Kích thước (mm)
Cạnh tam giác để cố định	$f$	110
Cạnh tam giác để di động	$e$	60
Khớp trên	$r_f$	157
Khớp hình bình hành	$r_e$	360



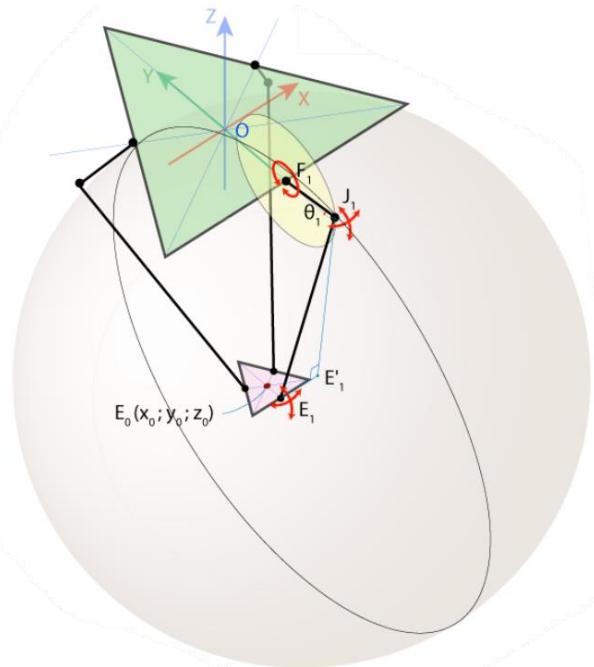
Hình 4. 1: Mô hình động học của Robot Delta

## 4.2 Động học nghịch

Với bài toán động học nghịch chúng ta sẽ biết được tọa độ điểm cuối cùng của Robot  $E_0(x_0, y_0, z_0)$  là trọng tâm của tam giác để di động và các thông số cần xác định là 3 góc  $\theta$  được tạo bởi các khớp trên và mặt phẳng của đế cố định. Với hệ quy chiếu sẽ được chọn có gốc tọa là trọng tâm của tam giác đế cố định.

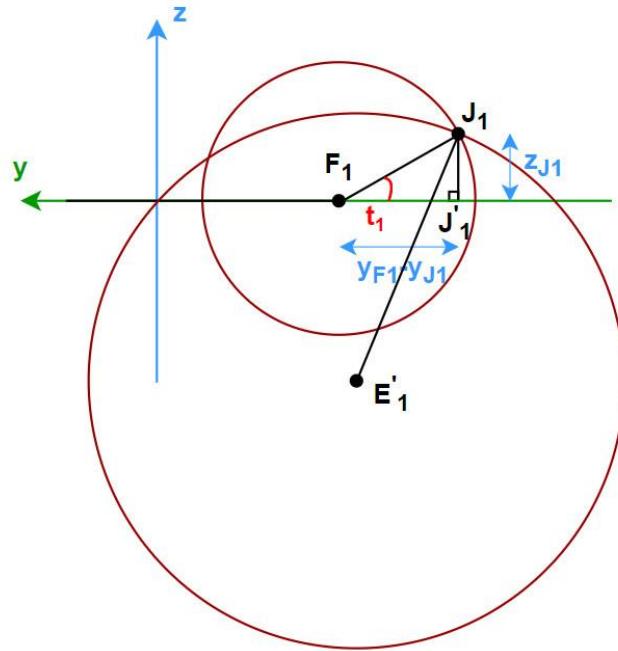


Hình 4. 2: Điểm cuối cùng của Robot và các góc  $\theta$



Hình 4. 3: Giao tuyến của hình cầu và mặt phẳng  $yz$

Vì thiết kế của Robot nên khớp trên  $F_1J_1$  chỉ có thể xoay trong mặt phẳng  $yz$ , tạo thành đường tròn có tâm  $F_1$  và bán kính  $r_f$  đồng thời khớp hình bình hành  $J_1E_1$  có thể xoay tự do tạo thành hình cầu có tâm  $E_1$  và bán kính  $r_e$ . Giao tuyến của hình cầu này và mặt phẳng  $yz$  là đường tròn có tâm  $E'_1$  và bán kính  $E'_1J_1$ , trong đó  $E'_1$  là hình chiếu của  $E_1$  trên mặt phẳng  $yz$ .



Hình 4. 4: Giao điểm của 2 đường tròn

Theo hình học:

$$EE_1 = \frac{e}{2} \tan 30^\circ = \frac{e}{2\sqrt{3}} \quad (4.1)$$

Suy ra:

$$E_1 = (x_0, y_0 - \frac{e}{2\sqrt{3}}, z_0) \Rightarrow E'_1(0, y_0 - \frac{e}{2\sqrt{3}}, z_0) \quad (4.2)$$

Và:

$$E_1E'_1 = x_0 \Rightarrow E'_1J_1 = \sqrt{E_1J_1^2 - E_1E'_1^2} = \sqrt{r_e^2 - x_0^2} \quad (4.3)$$

Lại có:

$$OF_1 = \frac{f}{2} \tan 30^\circ = \frac{f}{2\sqrt{3}} \quad (4.4)$$

Suy ra:

$$F_1(0, -\frac{f}{2\sqrt{3}}, 0) \quad (4.5)$$

Áp dụng định lý Pytago cho 2 tam giác vuông  $F_1J_1J_1'$  và  $J_1E_1E_1'$ :

$$\begin{cases} (y_{J_1} - y_{F_1})^2 + (z_{J_1} - z_{F_1})^2 = r_f^2 \\ (y_{J_1} - y_{E_1})^2 + (z_{J_1} - z_{E_1})^2 = r_e^2 - x_0^2 \end{cases} \quad (4.6)$$

Từ (4.6) suy ra được như sau:

$$\Rightarrow \begin{cases} (y_{J_1} + \frac{f}{2\sqrt{3}})^2 + z_{J_1}^2 = r_f^2 \\ (y_{J_1} - y_0 + \frac{e}{2\sqrt{3}})^2 + (z_{J_1} - z_0)^2 = r_e^2 - x_0^2 \end{cases} \Rightarrow J_1(0, y_{J_1}, z_{J_1}) \quad (4.7)$$

Từ hệ phương trình (4.7), nhóm lấy phương trình trên trừ đi phương trình dưới:

$$2y_{j1} \frac{f}{2\sqrt{3}} + 2y_{j1} \left( y_0 - \frac{e}{2\sqrt{3}} \right) + \left( \frac{f}{2\sqrt{3}} \right)^2 - \left( y_0 - \frac{e}{2\sqrt{3}} \right)^2 + 2z_{j1}z_0 - z_0^2 = r_f^2 - (r_e^2 - x_0^2) \quad (4.8)$$

Từ phương trình (4.8) suy ra được phương trình như sau:

$$2y_{j1} \left( \frac{f}{2\sqrt{3}} + y_0 - \frac{e}{2\sqrt{3}} \right) + \left( \frac{f}{2\sqrt{3}} \right)^2 - \left( y_0 - \frac{e}{2\sqrt{3}} \right)^2 + 2z_{j1}z_0 - z_0^2 - r_f^2 + (r_e^2 - x_0^2) = 0 \quad (4.9)$$

Nhóm đặt biến như sau:

$$m_1 = \frac{f}{2\sqrt{3}} + y_0 - \frac{e}{2\sqrt{3}} \quad (4.10)$$

$$m_2 = \left( \frac{f}{2\sqrt{3}} \right)^2 - \left( y_0 - \frac{e}{2\sqrt{3}} \right)^2 - z_0^2 - r_f^2 + (r_e^2 - x_0^2) \quad (4.11)$$

Thay 2 biến vừa đặt (4.10) và (4.11) vào phương trình (4.9), nhóm thu được:

$$2y_{j1}m_1 + m_2 + 2z_{j1}z_0 = 0 \quad (4.12)$$

Từ phương trình (4.12) suy ra được phương trình:

$$z_{j1} = \frac{-2y_{j1}m_1 - m_2}{2z_0} \quad (4.13)$$

Thay phương trình (4.13) vào phương trình trên của hệ phương trình (4.7) sẽ thu được phương trình bậc 2 theo ẩn  $y_{j1}$ :

$$y_{j1}^2 + 2y_{j1} \frac{f}{2\sqrt{3}} + \left( \frac{f}{2\sqrt{3}} \right)^2 + \left( \frac{-2y_{j1}m_1 - m_2}{2z_0} \right)^2 = r_f^2 \quad (4.14)$$

Khai triển phương trình (4.14) thu được phương trình sau:

$$y_{j1}^2 + 2y_{j1} \frac{f}{2\sqrt{3}} + \left( \frac{f}{2\sqrt{3}} \right)^2 + \left( \frac{y_{j1}m_1}{z_0} \right)^2 + \frac{y_{j1}m_1m_2}{z_0^2} + \left( \frac{m_2}{2z_0} \right)^2 - r_f^2 = 0 \quad (4.15)$$

$$y_{j1}^2 \left( 1 + \frac{m_1^2}{z_0^2} \right) + y_{j1} \left( \frac{f}{\sqrt{3}} + \frac{m_1m_2}{z_0^2} \right) + \left( \frac{f}{2\sqrt{3}} \right)^2 + \left( \frac{m_2}{2z_0} \right)^2 - r_f^2 = 0 \quad (4.16)$$

Nhóm xét công thức tính  $\Delta$  cho phương trình (4.16):

$$\Delta = b^2 - 4ac \quad (4.17)$$

Với:

$$a = 1 + \frac{m_1^2}{z_0^2} \quad (4.18)$$

$$b = \frac{f}{\sqrt{3}} + \frac{m_1m_2}{z_0^2} \quad (4.19)$$

$$c = \left( \frac{f}{2\sqrt{3}} \right)^2 + \left( \frac{m_2}{2z_0} \right)^2 - r_f^2 \quad (4.20)$$

Từ phương trình (4.16) suy ra được 2 nghiệm  $y_{j1}$  như sau:

$$y_{j1} = \frac{-b \pm \sqrt{\Delta}}{2a} \quad (4.21)$$

Điểm  $J_1$  luôn ở vị trí âm của trục y nên nhận được nghiệm:

$$y_{j1} = \frac{-b - \sqrt{\Delta}}{2a} \quad (4.22)$$

Thay nghiệm phương trình (4.22) vào phương trình (4.13) thu được:

$$z_{j1} = \frac{-2 \left( \frac{-b - \sqrt{\Delta}}{2a} \right) m_1 - m_2}{2z_0} \quad (4.23)$$

Góc  $\theta_1$  là góc hợp bởi khớp trên  $F_1J_1$  và mặt phẳng để cố định.

Từ (4.22) và (4.23) suy ra:

$$\theta_1 = \arctan \left( \frac{z_{J_1}}{y_{F_1} - y_{J_1}} \right) \quad (4.24)$$

Khi đã tính được góc  $\theta_1$  nhóm lợi dụng tính đối xứng của Robot Delta để tính toán góc  $\theta_2$  và  $\theta_3$ . Bằng cách xoay hệ trực xyz 120° với tọa độ điểm cuối cùng của Robot mới  $E'_0(x'_0, y'_0, z'_0)$  và sử dụng cùng những thuật toán ở trên để tìm góc  $\theta_2$ . Và để tìm góc  $\theta_3$  chỉ cần xoay hệ trực xyz 1 góc  $-120^\circ$ .

Nhóm đặt hệ tọa độ ban đầu là A thì điểm  $E_0$  nhìn từ hệ tọa độ A sẽ là  $A_{E_0} = (x_0, y_0, z_0)$ . Khi xoay từ hệ trực tọa độ A một góc theo trục z thì sẽ nhận được hệ tọa độ mới B với tọa độ điểm cuối cùng  $E_0$  nhìn từ hệ tọa độ B là  $B_{E_0} = (x'_0, y'_0, z'_0)$ . Ta có công thức của phép xoay từ hệ tọa độ A qua hệ tọa độ B như sau:

$$A_{E_0} = {}^A_B R \cdot B_{E_0} \quad (4.25)$$

Để tìm được  $B_{E_0} = (x'_0, y'_0, z'_0)$  ta cần nhân 2 vế của phương trình (4.9) cho  $({}^A_B R)^T$ .

$$A_{E_0} \cdot ({}^A_B R)^T = {}^A_B R \cdot ({}^A_B R)^T \cdot B_{E_0} \quad (4.26)$$

Với:

$$({}^A_B R) \cdot ({}^A_B R)^T = 1 \quad (4.27)$$

Từ (4.26) và (4.27) suy ra được:

$$B_{E_0} = ({}^A_B R)^T \cdot A_{E_0} \quad (4.28)$$

Bằng cách xoay từ hệ tọa độ A sang hệ tọa độ B quanh trục z với góc  $120^\circ$  ta suy ra được phương trình:

$$({}^A_B R)^T = (R_z(120^\circ))^T = \begin{bmatrix} \cos(120^\circ) & \sin(120^\circ) & 0 \\ -\sin(120^\circ) & \cos(120^\circ) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.29)$$

Ngoài ra, ta có ma trận chuyển đổi hệ tọa độ tổng quát từ hệ tọa độ A sang hệ tọa độ B. Với lần lượt là phép xoay quanh trục z 1 góc  $120^\circ$  và vì chỉ xoay tại chỗ mà không tịnh tiến. Từ phương trình (4.28) suy ra phương trình như sau:

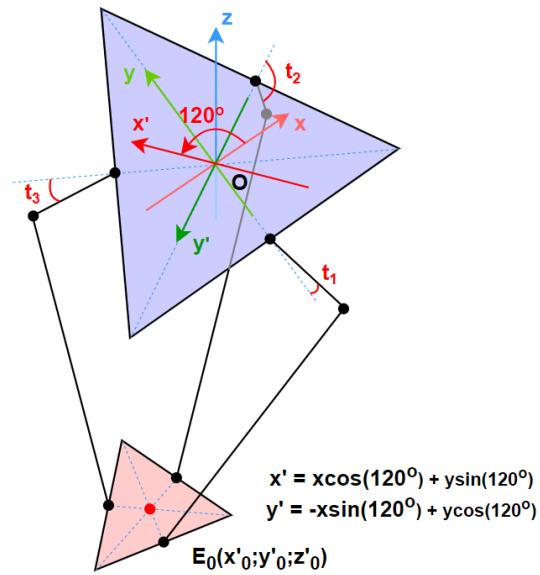
$$B_{E_0} = R_z(120^\circ)^T T \cdot A_{E_0} \quad (4.30)$$

Tiếp tục thay phương trình (4.29) vào phương trình (4.30) ta được phương trình:

$$B_{E_0} = (R_z(120^\circ))^T T \cdot A_{E_0} = \begin{bmatrix} x_0 \cos(120^\circ) + y_0 \sin(120^\circ) \\ -x_0 \sin(120^\circ) + y_0 \cos(120^\circ) \\ z_0 \\ 1 \end{bmatrix} \quad (4.31)$$

Vậy nhóm đã tìm được hệ phương trình để tính góc  $\theta_2$  khi xoay quanh hệ trục tọa độ 1 góc  $120^\circ$  theo trục z:

$$\begin{cases} x_0' = x_0 \cos(120^\circ) + y_0 \sin(120^\circ) \\ y_0' = -x_0 \sin(120^\circ) + y_0 \cos(120^\circ) \\ z_0' = z_0 \end{cases} \quad (4.32)$$



Hình 4. 5: Hệ tọa độ mới  $(x', y', z')$

Ngược lại với góc  $\theta_2$ , tiến hành xoay hệ tọa độ A quanh trục z 1 góc  $-120^\circ$  tính toán tương tự cho góc  $\theta_3$ , ta sẽ thu được hệ phương trình như sau:

$$\begin{cases} x_0' = x_0 \cos(120^\circ) - y_0 \sin(120^\circ) \\ y_0' = x_0 \sin(120^\circ) + y_0 \cos(120^\circ) \\ z_0' = z_0 \end{cases} \quad (4.33)$$

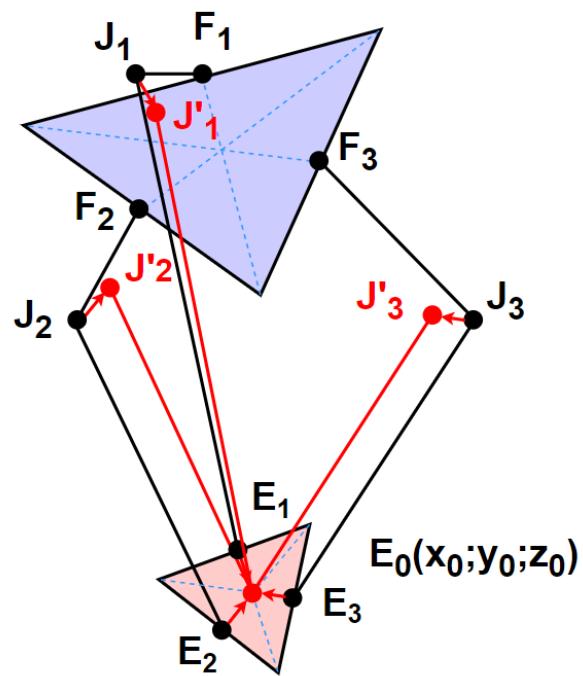
### 4.3 Động học thuận

Bài toán động học thuận là bài toán ngược lại là đi tính tọa độ điểm cuối cùng của Robot  $E_0(x_0, y_0, z_0)$  khi đã biết được các góc  $\theta$ .

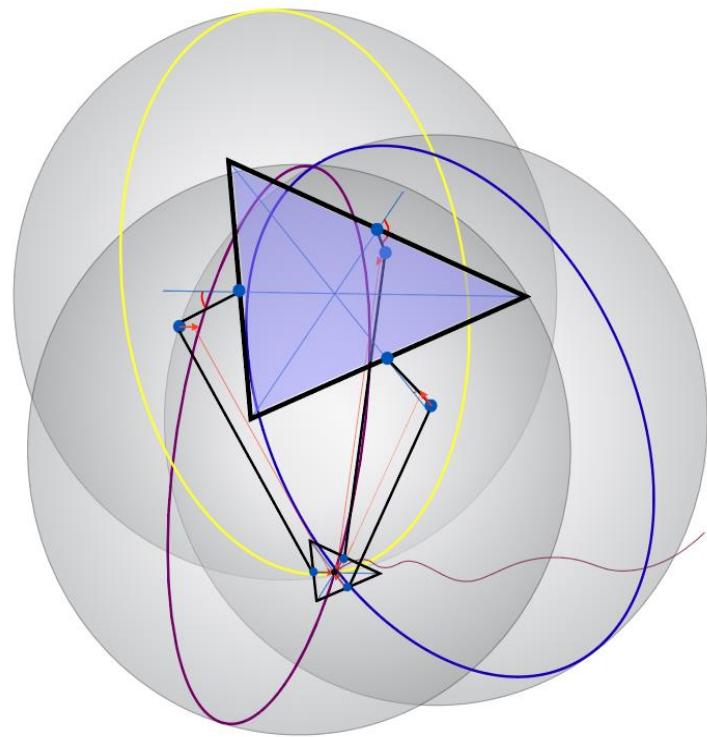
Như đã tính toán ở trên các khớp hình bình hành  $J_1E_1, J_2E_2, J_3E_3$  có thể xoay tự do quanh các điểm  $J_1, J_2, J_3$  (có thể tính được nếu biết góc  $\theta$ ) tạo thành các mặt cầu có bán kính  $r_e$ . Nếu tịnh tiến lần lượt các điểm  $J_1, J_2, J_3$  theo các vector  $\overrightarrow{E_1E_0}, \overrightarrow{E_2E_0}, \overrightarrow{E_3E_0}$  thì các mặt cầu sẽ cắt nhau tại điểm  $E_0$ .

Lúc này chỉ cần giải 3 phương trình mặt cầu có tâm lần lượt là  $J_1', J_2', J_3'$  và bán kính  $r_e$  để tìm giao điểm  $E_0$ .

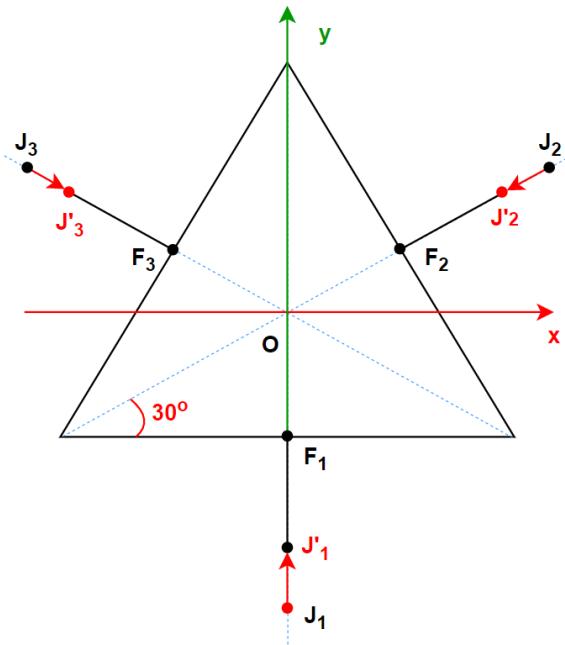
$$(x - x_j)^2 + (y - y_j)^2 + (z - z_j)^2 = r_e^2 \quad (4.34)$$



Hình 4. 6: Mô hình sau khi tịnh tiến  $J_1, J_2, J_3$



Hình 4. 7: Giao điểm 3 mặt cầu



Hình 4. 8: Hình chiếu trên mặt phẳng xy

Theo hình học:

$$OF_1 = OF_2 = OF_3 = \frac{f}{2} \tan(30^\circ) = \frac{f}{2\sqrt{3}} \quad (4.35)$$

Và:

$$J_1 J'_1 = J_2 J'_2 = J_3 J'_3 = \frac{e}{2} \tan(30^\circ) = \frac{e}{2\sqrt{3}} \quad (4.36)$$

$$\begin{cases} F_1 J_1 = r_f \cos(\theta_1) \\ F_2 J_2 = r_f \cos(\theta_2) \\ F_3 J_3 = r_f \cos(\theta_3) \end{cases} \quad (4.37)$$

Từ (4.35), (4.36) và (4.37) suy ra:

$$\begin{cases} J'_1 \left( 0, -\frac{(f-e)}{2\sqrt{3}} - r_f \cos(\theta_1), -r_f \sin(\theta_1) \right) \\ J'_2 \left( \left[ \frac{(f-e)}{2\sqrt{3}} + r_f \cos(\theta_2) \right] \cos(30^\circ), \left[ \frac{(f-e)}{2\sqrt{3}} + r_f \cos(\theta_2) \right] \sin(30^\circ), -r_f \sin(\theta_2) \right) \\ J'_3 \left( \left[ \frac{(f-e)}{2\sqrt{3}} + r_f \cos(\theta_3) \right] \cos(30^\circ), \left[ \frac{(f-e)}{2\sqrt{3}} + r_f \cos(\theta_3) \right] \sin(30^\circ), -r_f \sin(\theta_3) \right) \end{cases} \quad (4.38)$$

Nhóm sẽ có hệ phương trình của 3 mặt cầu với giao điểm là  $E_0(x_0, y_0, z_0)$

$$\begin{cases} x^2 + (y - y_1)^2 + (z - z_1)^2 = r_e^2 \\ (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = r_e^2 \\ (x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 = r_e^2 \end{cases} \quad (4.39)$$

Từ (4.39) suy ra:

$$\begin{cases} x^2 + y^2 + z^2 - 2y_1y - 2z_1z = r_e^2 - y_1^2 - z_1^2 \\ x^2 + y^2 + z^2 - 2x_2x - 2y_2y = r_e^2 - x_2^2 - y_2^2 - z_2^2 \\ x^2 + y^2 + z^2 - 2x_3x - 2y_3y - 2z_3z = r_e^2 - x_3^2 - y_3^2 - z_3^2 \end{cases} \quad (4.40)$$

Với:  $(x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3)$  lần lượt là tọa độ của các điểm  $J_1', J_2', J_3'$ .

Đặt:

$$w_i = x_i^2 + y_i^2 + z_i^2 \quad (4.41)$$

Từ (4.40) và (4.41) suy ra:

$$\begin{cases} x_2x + (y_1 - y_2)y + (z_1 - z_2)z = \frac{w_1 - w_2}{2} \\ x_3x + (y_1 - y_3)y + (z_1 - z_3)z = \frac{w_1 - w_3}{2} \\ (x_2 - x_3)x + (y_2 - y_3)y + (z_2 - z_3)z = \frac{w_2 - w_3}{2} \end{cases} \quad (4.42)$$

Lại đặt:

$$d = (y_2 - y_1)x_3 - (y_3 - y_1)x_2 \quad (4.43)$$

$$a_1 = \frac{1}{d} [(z_2 - z_1)(y_3 - y_1) - (z_3 - z_1)(y_2 - y_1)] \quad (4.44)$$

$$a_2 = -\frac{1}{d} [(z_2 - z_1)x_3 - (z_3 - z_1)x_2] \quad (4.45)$$

$$b_1 = -\frac{1}{2d} [(w_2 - w_1)(y_3 - y_1) - (w_3 - w_1)(y_2 - y_1)] \quad (4.46)$$

$$b_2 = \frac{1}{2d} \left[ (w_2 - w_1)x_3 - (w_3 - w_1)x_2 \right] \quad (4.47)$$

Từ (4.26), (4.27), (4.28), (4.29), (4.30) và (4.31) ta được:

$$x = a_1 z + b_1 \quad (4.48)$$

$$y = a_2 z + b_2 \quad (4.49)$$

Thay (4.32) và (4.33) vào (4.24) ta thu được phương trình bậc 2 ẩn  $z_0$ :

$$(a_1^2 + a_2^2 + 1)z^2 + 2(a_1 + a_2(b_2 - y_1) - z_1)z + (b_1^2 + (b_2 - y_1)^2 + z_1^2 - r_e^2) = 0 \quad (4.50)$$

Sau khi giải phương trình tìm được  $z_0$  và tiếp tục ta sẽ tìm được  $x_0$  và  $y_0$  từ (4.32) và (4.33).

Phương trình bậc 2 ẩn  $z_0$  có dạng:

$$az_0^2 + bz_0 + c = 0 \quad (4.51)$$

Với:

$$a = a_1^2 + a_2^2 + 1 \quad (4.52)$$

$$b = 2(a_1 + a_2(b_2 - y_1) - z_1) \quad (4.53)$$

$$c = b_1^2 + (b_2 - y_1)^2 + z_1^2 - r_e^2 \quad (4.54)$$

Nhóm xét:

$$\Delta = b^2 - 4ac \quad (4.55)$$

Với:

$\Delta < 0$  phương trình sẽ không có nghiệm

$\Delta = 0$  phương trình sẽ có nghiệm  $z_0 = -\frac{b}{2a}$

$\Delta > 0$  phương trình sẽ có 2 nghiệm  $z_0 = \frac{-b \pm \sqrt{\Delta}}{2a}$

Tuy nhiên ta chỉ lấy nghiệm:  $z_0 = \frac{-b - \sqrt{\Delta}}{2a}$  vì đối với mô hình Robot Delta, điểm cuối cùng của robot luôn luôn ở vị trí âm của trục z.

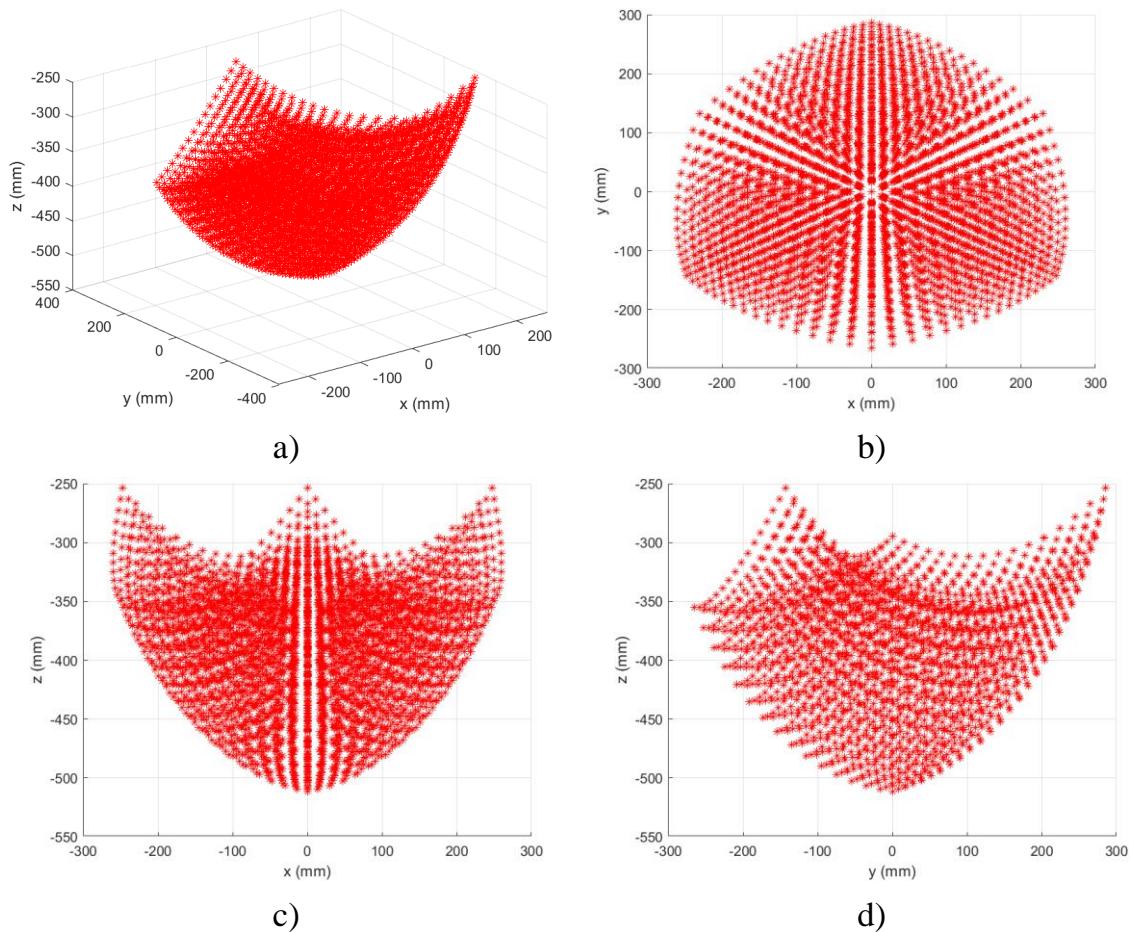
#### 4.4 Không gian làm việc của Robot Delta

Không gian làm việc của Robot Delta là toàn bộ thể tích mà cơ cấu chấp hành cuối có thể hoạt động khi Robot thực hiện tất cả các chuyển động. Thể tích không gian làm việc phụ thuộc vào kết cấu của khớp trên và khớp hình bình hành của Robot. Bên cạnh đó không gian làm việc cũng phụ thuộc vào miền giá trị của các biến khớp. Biết được vùng làm việc của Robot giúp nhóm dễ dàng và thuận tiện trong việc điều khiển và phân loại.

Nhóm đã sử dụng phần mềm Matlab và kết quả của bài toán động học thuận, tiến hành cho các điểm tọa độ của cơ cấu chấp hành cuối chạy trong không gian theo các góc  $\theta_1, \theta_2, \theta_3$  đã giới hạn.

Trong trường hợp này, nhóm sinh viên đã giới hạn các góc  $\theta_1, \theta_2, \theta_3$  như sau:

$$0^\circ < \theta_1, \theta_2, \theta_3 < 90^\circ \quad (4.56)$$



Hình 4. 9: Không gian làm việc của Robot Delta trong. a) Không gian 3 chiều xyz; Không gian 2 chiều b) xy; c) xz; d) yz

Tính toán không gian làm việc của Robot Delta là việc rất quan trọng trong việc nghiên cứu, thiết kế và chế tạo Robot. Với việc đã xác định được không gian làm việc của Robot, ta có thể đáp ứng được các yêu cầu làm việc thực tế, cũng như tiện trong việc cải tiến và nâng cấp trong tương lai.

### 4.5 Mô tả quy trình hoạt động và lưu đồ giải thuật của hệ thống

Quy trình hoạt động của hệ thống bao gồm 5 bước:

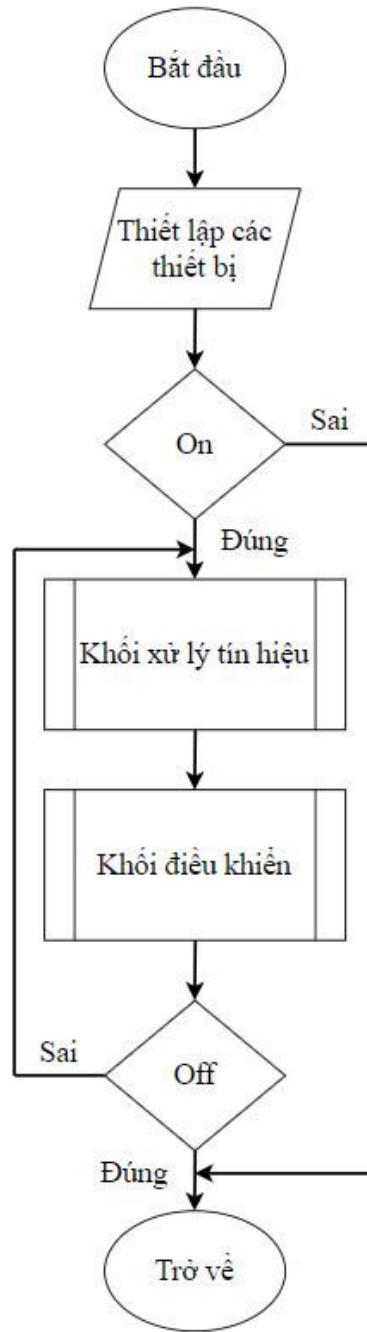
**Bước 1** - Nhận tín hiệu hoạt động: Người dùng sẽ điều khiển hệ thống hoạt động thông qua giao diện điều khiển được thiết kế trên Visual Studio Code. Giao diện này sẽ cho phép điều khiển các thiết bị trong hệ thống (bật tắt băng tải, bật tắt bơm hút, điều khiển robot chạy động học thuận nghịch). Và khi người dùng bật chế độ phân loại thì camera sẽ hoạt động.

**Bước 2** - Xử lý tín hiệu đầu vào: Khi camera và băng tải đã trong chế độ sẵn sàng thì rác thải có thể được đưa vào. Khi đi qua vùng làm việc của camera, mô hình YOLO sẽ ghi nhận, phân loại và xuất giá trị vị trí của bounding box chứa loại rác thải đó.

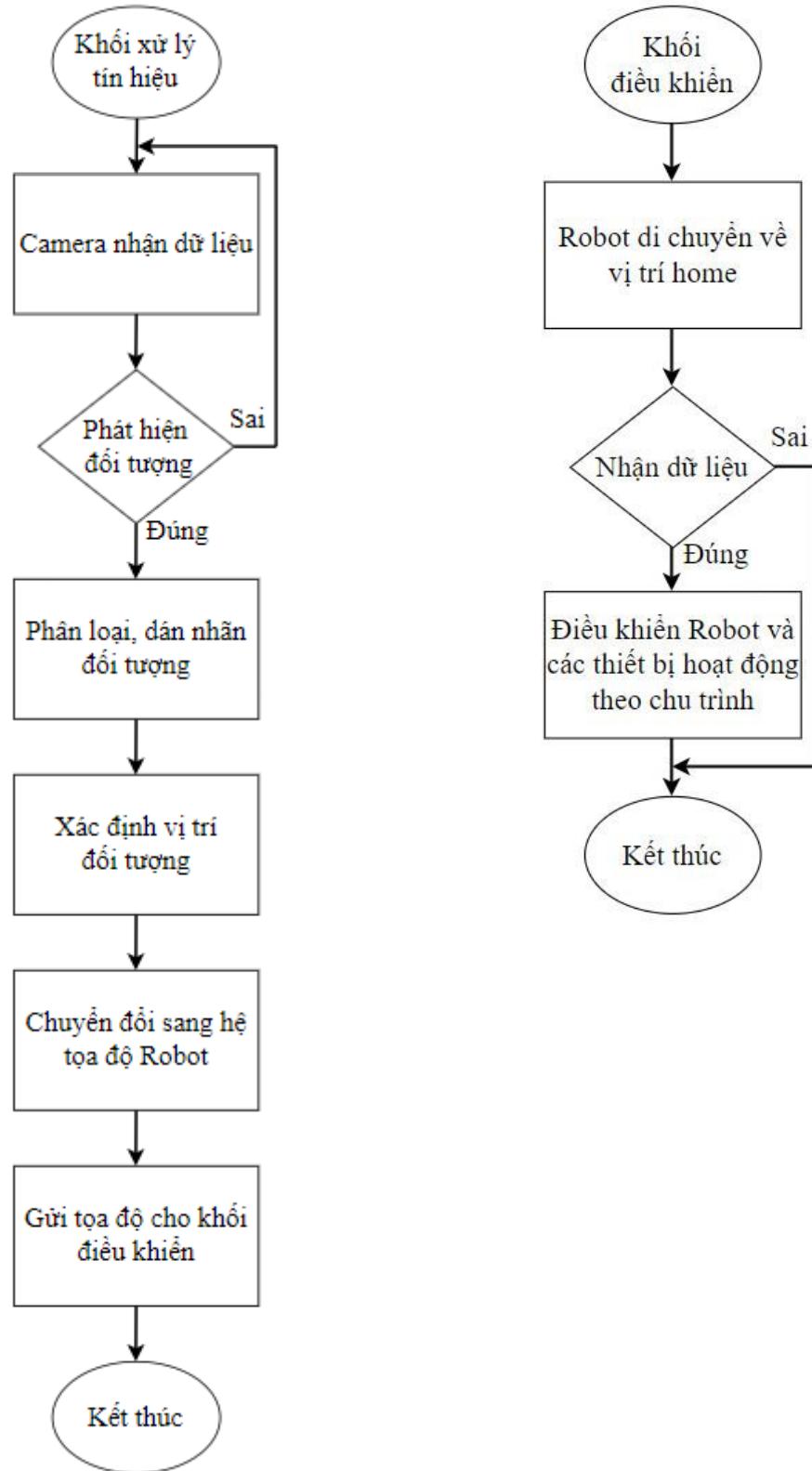
**Bước 3** - Tính toán động học: Khi có được vị trí của rác thải hệ thống sẽ tính toán động học nghịch và chuyển đổi giá trị vị trí đó thành các góc của Robot. Từ các góc này sẽ tiếp tục tính toán ra số xung cần cấp để điều khiển từng động cơ và gửi về bộ điều khiển.

**Bước 4** - Gửi tín hiệu điều khiển: Khi nhận được tín hiệu, bộ điều khiển sẽ tiến hành cấp xung cho từng động cơ đồng thời điều khiển các thiết bị khác (băng tải, động cơ bơm hút) để phối hợp với Robot thực hiện chu trình.

**Bước 5** - Điều khiển Robot: Khi có tín hiệu xung điều khiển các cánh tay Robot sẽ di chuyển thông qua các động cơ tiếp cận đối tượng và tiến hành thực hiện chu trình phân loại. Khi kết thúc 1 chu trình phân loại Robot sẽ di chuyển về vị trí để đợi đối tượng rác thải tiếp theo vào.

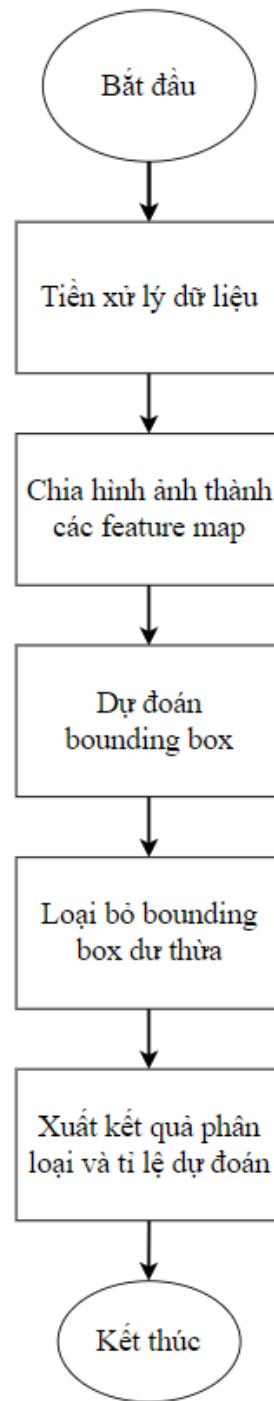


Hình 4. 10: Lưu đồ tổng quan hệ thống



Hình 4.11: Lưu đồ khối xử lý tín hiệu

Hình 4.12: Lưu đồ khối điều khiển

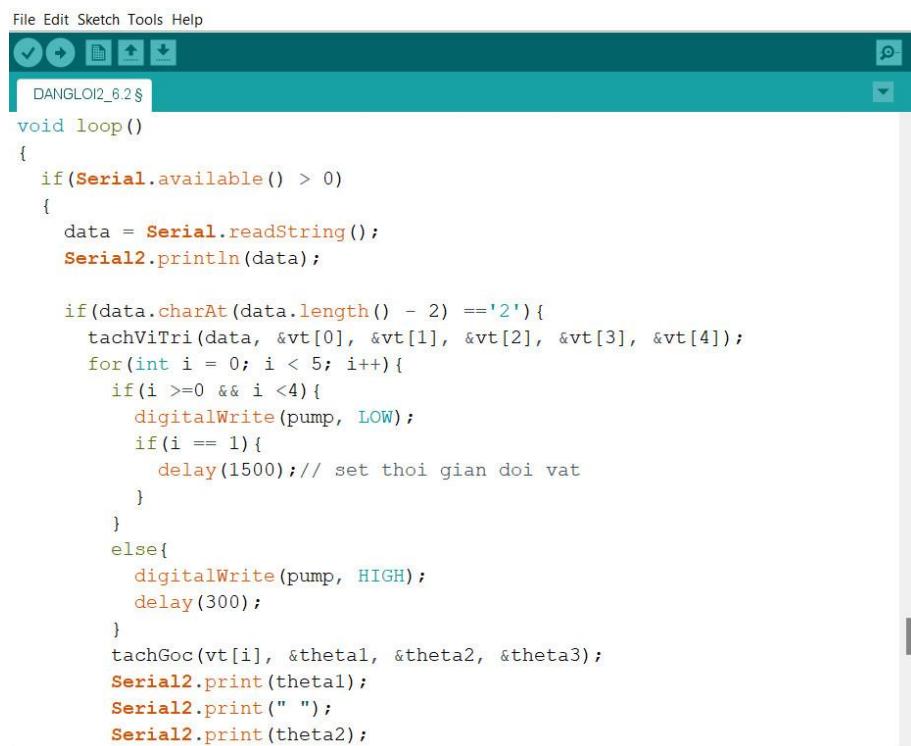


Hình 4. 13: Lưu đồ dự đoán Bounding box của YOLOv5

## 4.6 Lập trình, thiết kế phần mềm điều khiển và giao diện người dùng

### 4.6.1 Lập trình điều khiển Robot và các thiết bị

Nhóm sử dụng phần mềm Arduino IDE để lập trình điều khiển cho động cơ và các thiết bị. Trong đề tài này, nhóm sử dụng Arduino Mega 2560 điều khiển Robot thông qua 3 động cơ bước và các driver. Ngoài ra nhóm cũng điều khiển các thiết bị khác như: băng tải, bơm hút chân không, van 2 chiều để phối hợp với Robot trong quy trình phân loại rác thải.



```
File Edit Sketch Tools Help
DANGLOI2_6.2 §
void loop()
{
    if(Serial.available() > 0)
    {
        data = Serial.readString();
        Serial2.println(data);

        if(data.charAt(data.length() - 2) == '2'){
            tachViTri(data, &vt[0], &vt[1], &vt[2], &vt[3], &vt[4]);
            for(int i = 0; i < 5; i++){
                if(i >=0 && i <4){
                    digitalWrite(pump, LOW);
                    if(i == 1){
                        delay(1500); // set thời gian đợi vắt
                    }
                }
                else{
                    digitalWrite(pump, HIGH);
                    delay(300);
                }
                tachGoc(vt[i], &thetal, &theta2, &theta3);
                Serial2.print(thetal);
                Serial2.print(" ");
                Serial2.print(theta2);
            }
        }
    }
}
```

Hình 4. 14: Lập trình điều khiển trên phần mềm Arduino IDE

### 4.6.2 Lập trình, thiết kế phần mềm điều khiển và giao diện người dùng

Python là một ngôn ngữ lập trình rất phổ biến được nhiều người sử dụng rộng rãi hiện nay. Python hỗ trợ rất nhiều chức năng cho người lập trình, tùy vào mục đích của người lập trình.

Yolov5 chạy chương trình bằng ngôn ngữ Python. Sau khi Training xong trên phần mềm Google Colab, ta sẽ nhúng về máy tính và thu được một file detect.py, một file trọng số weight.pt và tiếp tục lập trình trên file đã được nhúng xuống.

Bên dưới là màn hình giao diện giới thiệu của nhóm. Trên màn hình sẽ mô tả, giới thiệu về các thông tin về tên đề tài, giảng viên hướng dẫn, người thực hiện và tên trường Đại Học Sư Phạm Kỹ Thuật TP. Hồ Chí Minh, Khoa Đào Tạo Chất Lượng

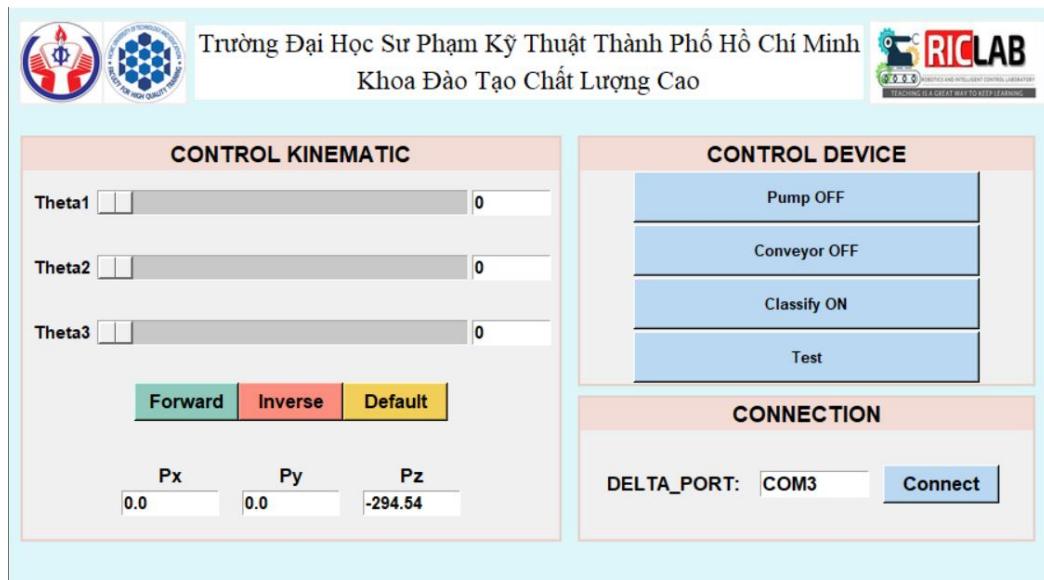
## Chương 4: Tính toán động học và thiết kế phần mềm điều khiển

Cao. Ngoài ra còn có các hình ảnh liên quan về đề tài, hình ảnh logo trường, logo khoa, logo của RIC LAB. Có một nút nhấn Start để chuyển giao diện giới thiệu sang giao diện điều khiển.



Hình 4. 15: Giao diện giới thiệu trên Visual Studio Code

Khi nhấn vào nút Start ở màn hình giới thiệu thì sẽ chuyển tiếp đến màn hình giao diện điều khiển, cho phép người dùng thao tác trực tiếp trên màn hình. Trên màn hình điều khiển có 3 khu vực lân lượt là khu vực điều khiển động học thuận nghịch, tiếp đến là khu vực kết nối các thiết bị điều khiển và cuối cùng là khu vực kết nối port để giao tiếp UART giữa Visual Studio Code và Arduino.



Hình 4. 16: Giao diện điều khiển trên Visual Studio Code

## CHƯƠNG 5: PHÁT HIỆN, PHÂN LOẠI VÀ XÁC ĐỊNH VỊ TRÍ ĐỐI TƯỢNG

Quá trình phát hiện, phân loại và xác định vị trí đối tượng được chia thành 2 giai đoạn chính. Đầu tiên là huấn luyện mô hình YOLOv5 đối với 2 loại rác thải (chai nhựa và lon) trên Google Colab, sau đó tiến hành nhúng mô hình đã huấn luyện về máy tính để phân loại trực tiếp bằng camera. Tiếp theo để xác định được vị trí của rác thải nhóm đã tính toán dựa vào vị trí thực tế của Robot và camera từ đó điều khiển Robot đến tiếp cận.

### 5.1 Quy trình huấn luyện mô hình YOLO

Nhóm sử dụng mô hình YOLO (You Only Look Once) là một trong những mô hình phát hiện và phân loại đối tượng nhanh nhất dựa trên mạng nơ-ron tích chập CNN.



Hình 5. 1: Cáu trúc huấn luyện mô hình YOLO

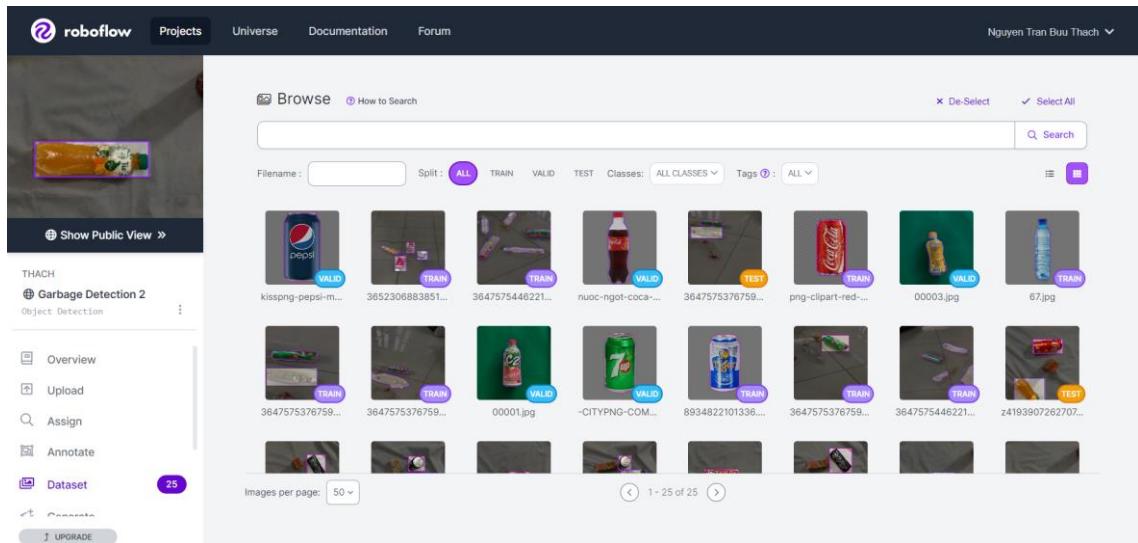
### 5.2 Chuẩn bị tập dữ liệu

#### 5.2.1 Thu thập dữ liệu

Tập dữ liệu cụ thể là các hình ảnh dùng để huấn luyện mạng nơ-ron sẽ được chụp trực tiếp từ máy ảnh, đây là các hình ảnh đầu vào chưa qua xử lí và định dạng. Tùy vào chất lượng của thiết bị chụp và điều kiện ánh sáng sẽ ảnh hưởng đến chất lượng ảnh đầu vào qua đó có thể ảnh hưởng phần nào đến quá trình huấn luyện.

Sau khi có ảnh đầu vào nhóm đã tạo một tập dữ liệu riêng trên phần mềm Roboflow và tải tất cả ảnh lên đó.

## Chương 5: Phát hiện, phân loại và xác định vị trí đối tượng

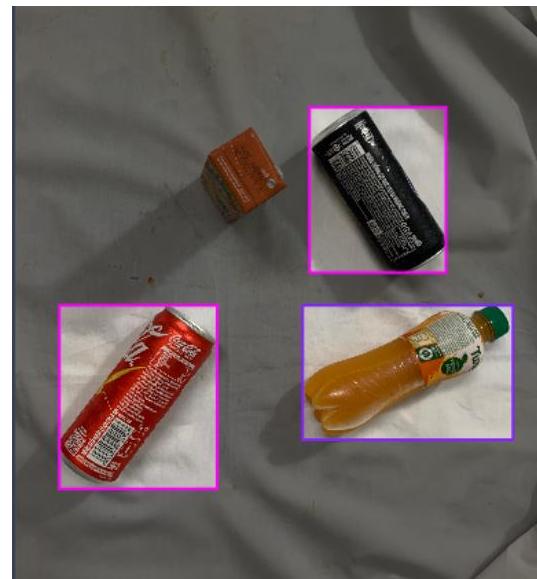


Hình 5. 2: Tải dữ liệu lên phần mềm Roboflow

### 5.2.2 Dán nhãn dữ liệu

Sau khi đã tải tập dữ liệu lên nhóm sẽ tiến hành bước đầu tiên của xử lý ảnh là dán nhãn dữ liệu. Đây sẽ là quá trình vẽ các đường bounding box bao quanh vật thể và chú thích vật thể sẽ thuộc nhóm đối tượng cần phân loại nào. Tùy vào số lượng đối tượng mong muốn phân loại mà quá trình này sẽ mất nhiều thời gian hơn. Ở đề tài này nhóm chỉ phân loại với 2 đối tượng đầu vào là: Can (lon) và Bottle (chai nhựa).

Roboflow sẽ hỗ trợ phương pháp dán nhãn bounding box đó là: dán nhãn theo hình chữ nhật.

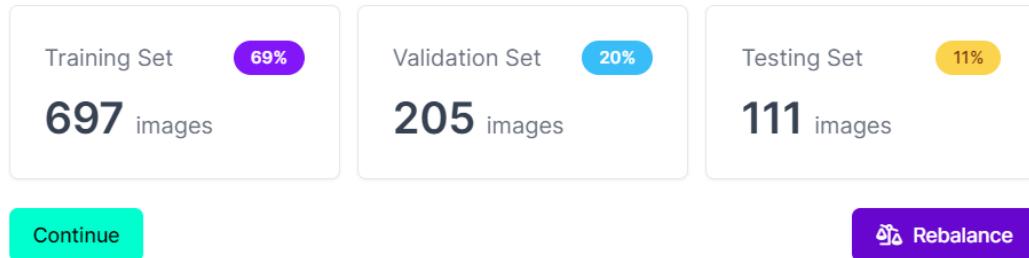


Hình 5. 3: Dán nhãn theo Bounding box

### 5.2.3 Phân chia tập dữ liệu

Ở bước này ta sẽ phân chia tập dữ liệu hình ảnh đã dán nhãn ra 3 mục:

- Training: sẽ được dùng làm đầu vào huấn luyện.
- Validation: sẽ được dùng để kiểm chứng qua đó tìm kiếm ra phương pháp huấn luyện tốt nhất cho mô hình.
- Test: sẽ được dùng để đánh giá kết quả huấn luyện.



Hình 5. 4: Phân chia tập dữ liệu để huấn luyện

### 5.2.4 Tiền xử lý dữ liệu

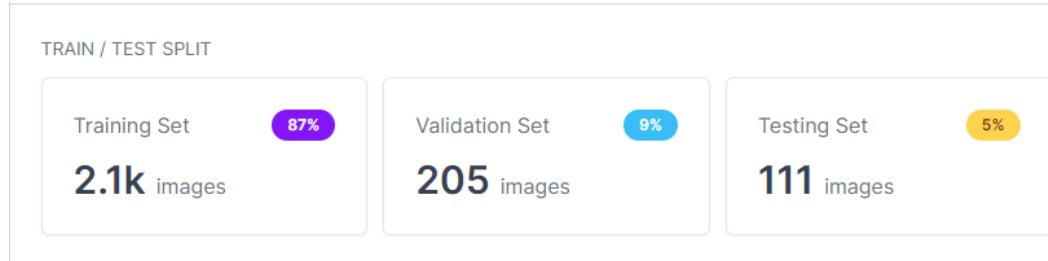
Các hình ảnh được đưa vào với các kích thước, định dạng khác nhau nên cần thêm một bước tiền xử lý dữ liệu để quy chuẩn các hình ảnh đầu vào. Vì đầu vào của mô hình YOLO chỉ có thể hỗ trợ với những ảnh có kích thước 416x416 hoặc 608x608 nên nhóm đã chọn định dạng 416x416 để quy chuẩn tất cả hình ảnh.

### 5.2.5 Làm giàu dữ liệu

Để có thể tiết kiệm thời gian trong việc thu thập dữ liệu cũng như đa dạng hóa dữ liệu đầu vào, phần mềm Roboflow hỗ trợ các chức năng như xoay ảnh, làm mờ, tăng, giảm độ sáng,... giúp tạo ra các ảnh mới hoàn toàn. Ở bước này ta có thể tăng số lượng ảnh của tập dữ liệu lên gấp 2 thậm chí gấp 3 tùy vào mong muốn của người dùng.

Nhóm đã sử dụng một số chức năng để làm giàu dữ liệu như:

- Xoay 90 độ.
- Tăng giảm độ phân giải 38%.
- Lật ảnh.
- Làm mờ ảnh.
- Tăng nhiễu 2%.



Hình 5. 5: Dữ liệu sau khi đã được làm giàu



Hình 5. 6: Các hình ảnh mới được tạo ra sau khi được làm giàu

Đây cũng là bước cuối cùng của chuẩn bị tập dữ liệu, sau khi đã hoàn thành người dùng sẽ có được tập dữ liệu dùng để huấn luyện. Roboflow cũng hỗ trợ tải tập dữ liệu này về máy tính dưới dạng nén zip, rar hoặc dưới dạng mã lệnh chương trình, người dùng có thể tải về và dùng trực tiếp trên máy tính hoặc nhúng vào các webserver có hỗ trợ lập trình trực tiếp.

### 5.3 Quá trình huấn luyện

#### 5.3.1 Quá trình huấn luyện trên phần mềm Google Colab

Bên dưới là mô tả chi tiết các bước huấn luyện mô hình trên phần mềm Google Colab và chương trình cụ thể cho các bước.

## Chương 5: Phát hiện, phân loại và xác định vị trí đối tượng

---

Bảng 5. 1: Các bước huấn luyện mô hình YOLOv5 trên Google Colab

Bước	Mục đích
Bước 1	Kết nối Google Colab với Google Drive nếu muốn lưu trữ thông tin huấn luyện.
Bước 2	Tạo bản sao YOLOv5 và cài đặt các thư viện cần thiết cho việc huấn luyện và lấy dữ liệu.
Bước 3	Cho phép Colab kết nối với Roboflow.
Bước 4	Lấy tập dữ liệu đã được chuẩn bị trên Roboflow.
Bước 5	Bắt đầu tiến hành huấn luyện với kích thước ảnh là 416x416, 150 chu kỳ và mỗi chu kỳ là 16 ảnh (Tùy vào số chu kỳ và số lượng ảnh đầu vào mà quá trình này sẽ nhanh hay chậm).
Bước 6	Chạy mô hình đã huấn luyện trên tập Test của Roboflow và xuất ra thư mục.
Bước 7	Tải thư mục trọng số về máy tính hoặc nhúng trực tiếp hình ảnh, video lên Google Colab để kiểm chứng.

Bảng 5. 2: Chương trình cụ thể các bước huấn luyện

Bước	Chương trình
Bước 1	From google.colab import drive drive.mount ('/content/drive')
Bước 2	#clone YOLOv5 and !git clone https://github.com/ultralytics/yolov5 # clone repo %cd yolov5 %pip install -qr requirements.txt # install dependencies %pip install -q roboflow import torch import os from IPython.display import Image, clear_output # to display images

	print(f"Setup complete. Using torch {torch.__version__}\n({{torch.cuda.get_device_properties(0).name if torch.cuda.is_available() else 'CPU'}})")
Bước 3	from roboflow import Roboflow rf = Roboflow(model_format="yolov5", notebook="ultralytics")
Bước 4	#after following the link above, recieve python code with these fields filled in  #from roboflow import Roboflow #rf = Roboflow(api_key="YOUR API KEY HERE") #project = rf.workspace().project("YOUR PROJECT") #dataset = project.version("YOUR VERSION").download("yolov5")
Bước 5	!python train.py --img 416 --batch 16 --epochs 150 --data {dataset.location}/data.yaml --weights yolov5s.pt --cache
Bước 6	!python detect.py --weights runs/train/exp/weights/best.pt --img 416 --conf 0.1 --source {dataset.location}/test/images
Bước 7	

```

Epoch GPU_mem box_loss obj_loss cls_loss Instances Size
145/149 1.716 0.03357 0.01891 0.002625 58 416: 100% 3/3 [00:00<00:00, 6.33it/s]
          Class Images Instances P R mAP50 mAP50-95: 100% 1/1 [00:00<00:00, 15.65it/s]
          all 7 9 0.778 0.617 0.677 0.292

Epoch GPU_mem box_loss obj_loss cls_loss Instances Size
146/149 1.716 0.02932 0.01885 0.001896 62 416: 100% 3/3 [00:00<00:00, 6.40it/s]
          Class Images Instances P R mAP50 mAP50-95: 100% 1/1 [00:00<00:00, 14.36it/s]
          all 7 9 0.777 0.62 0.676 0.303

Epoch GPU_mem box_loss obj_loss cls_loss Instances Size
147/149 1.716 0.0307 0.01705 0.001717 57 416: 100% 3/3 [00:00<00:00, 6.27it/s]
          Class Images Instances P R mAP50 mAP50-95: 100% 1/1 [00:00<00:00, 14.61it/s]
          all 7 9 0.773 0.62 0.677 0.318

Epoch GPU_mem box_loss obj_loss cls_loss Instances Size
148/149 1.716 0.02922 0.01818 0.001852 73 416: 100% 3/3 [00:00<00:00, 6.75it/s]
          Class Images Instances P R mAP50 mAP50-95: 100% 1/1 [00:00<00:00, 15.54it/s]
          all 7 9 0.773 0.62 0.677 0.318

Epoch GPU_mem box_loss obj_loss cls_loss Instances Size
149/149 1.716 0.02707 0.0173 0.002201 67 416: 100% 3/3 [00:00<00:00, 6.18it/s]
          Class Images Instances P R mAP50 mAP50-95: 100% 1/1 [00:00<00:00, 17.39it/s]
          all 7 9 0.771 0.62 0.677 0.315

```

Hình 5. 7: Quá trình huấn luyện trên Google Colab

### 5.3.2 Kết quả huấn luyện

Sau khi huấn luyện thành công Google Colab sẽ trả về kết quả huấn luyện là những hình ảnh ở thư mục Test trên Roboflow. Các hình ảnh này đã được phân loại và định vị bằng các bounding box bởi mô hình mạng đã được huấn luyện.

### Kết quả huấn luyện với 185 ảnh đầu vào:

Ban đầu nhóm kiểm tra kết quả khi mới chỉ huấn luyện 185 ảnh đầu vào. Kết quả trả về từ tập Test đa số không chính xác. Một số hạn chế mà nhóm nhận được:

- Tỉ lệ nhầm lẫn giữa 2 đối tượng vẫn còn khá cao và rất nhiều đối tượng vẫn không nhận dạng được.
- Khi có 2 hoặc nhiều đối tượng đặt gần nhau thì rất dễ nhầm lẫn đối tượng.
- Khi đã nhận dạng thành công thì xác suất dự đoán cũng không cao (khoảng 20%-85%).



Hình 5. 8: Kết quả huấn luyện với 185 ảnh đầu vào

### Kết quả huấn luyện với 1026 ảnh đầu vào:

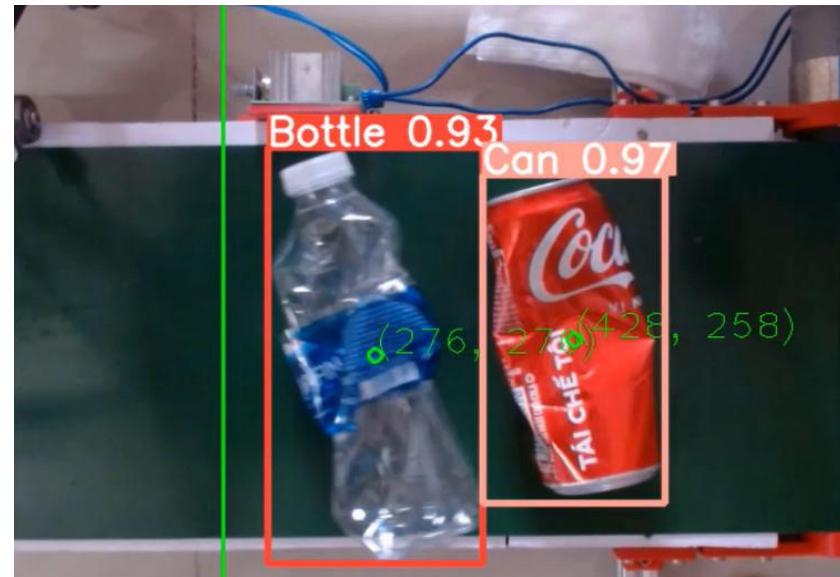
Nhận thấy kết quả huấn luyện chưa tốt nhóm đã tiến hành thu thập và tăng số lượng ảnh đầu vào lên 1026, gần gấp 10 lần so với lần thử nghiệm đầu tiên và nhóm nhận thấy được một số ưu điểm như:

- Tỉ lệ nhận dạng chính xác gần như là 100%.
- Nhận dạng vẫn chính xác ngay cả khi các đối tượng đặt gần nhau.
- Xác suất dự đoán khá cao (96%-100%).



Hình 5. 9: Kết quả huấn luyện với hình ảnh

Ngoài ra Colab còn trả về thư mục trọng số trong quá trình huấn luyện, người dùng có thể lấy nhúng về máy tính để thực hiện phân loại trực tiếp bằng camera hoặc phát triển các dự án khác.



Hình 5. 10: Kết quả huấn luyện với camera thời gian thực

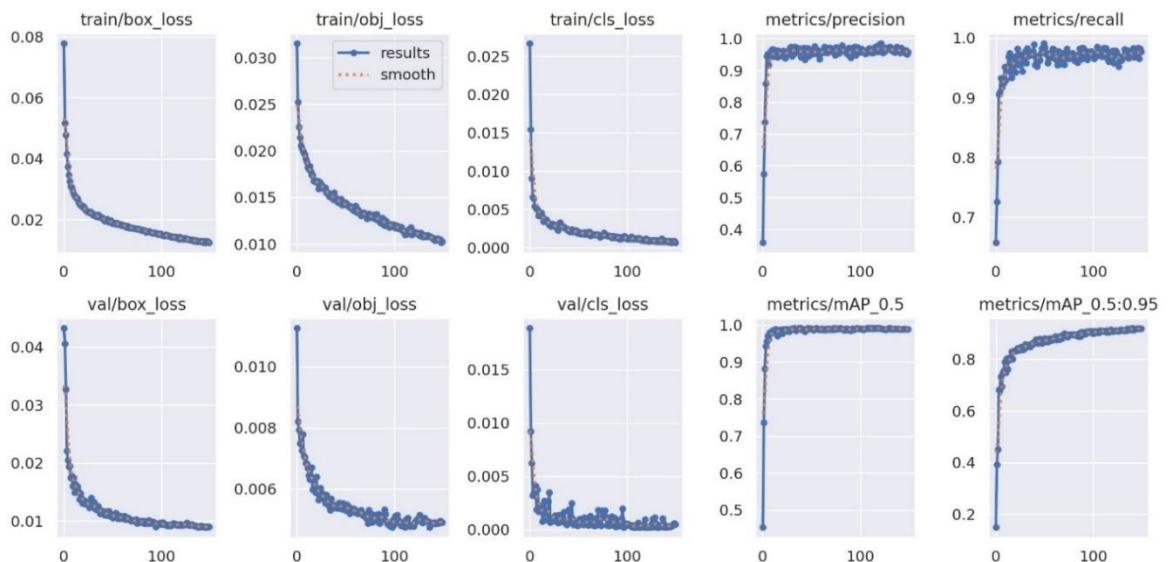
Tuy nhiên kết quả nhận dạng khi nhúng về máy tính để phân loại trực tiếp trên camera với thời gian thực hơi khác so với kết quả nhận dạng từ các hình ảnh của tập Test. Xác suất dự đoán khoảng 80%-98% ngoài ra nhóm còn nhận thấy một vấn đề là mật độ ánh sáng cũng ảnh hưởng một phần đến kết quả nhận dạng của mô hình. Nhưng nhìn chung khi nhúng mô hình mạng đã huấn luyện về máy tính thì kết quả vẫn đảm bảo về tính chính xác trên 90%.

### 5.3.3 Đánh giá huấn luyện

Khi đã huấn luyện thành công Google Colab cũng hỗ trợ đánh giá huấn luyện thông qua bảng các thông số thay đổi qua từng chu kỳ.

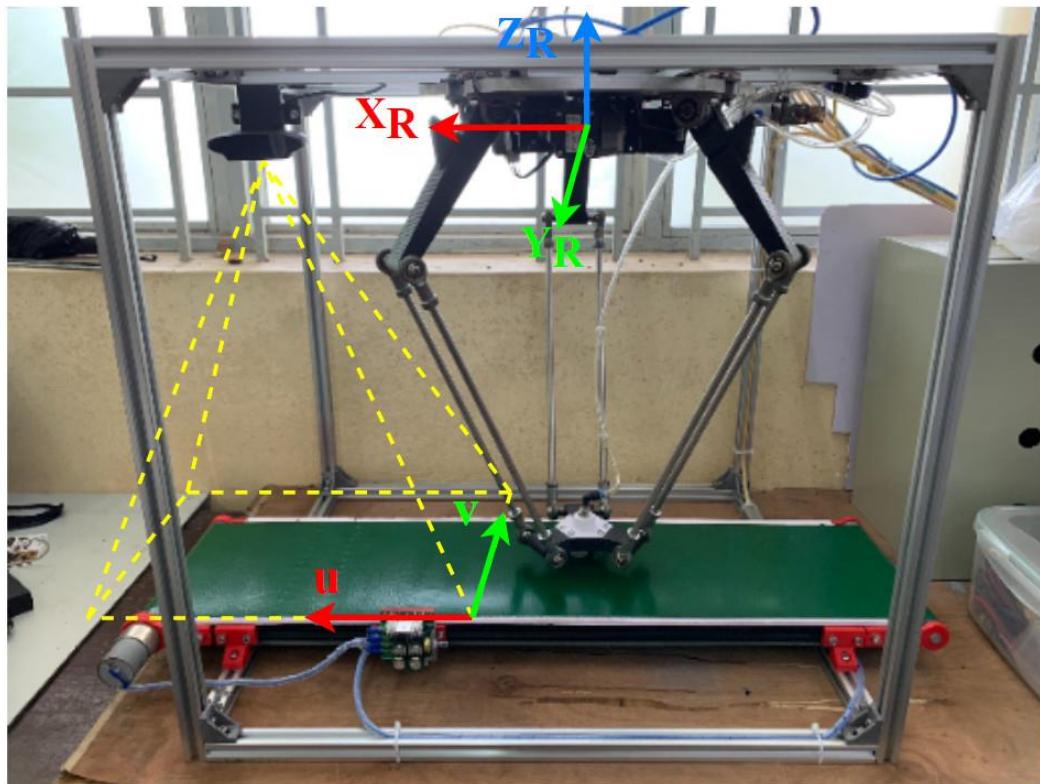
Có 4 thông số mà ta cần quan tâm:

- Loss: Sai số trong quá trình huấn luyện, sai số càng nhỏ chứng tỏ mô hình huấn luyện sẽ càng tốt.
- Precision: Độ chính xác là chỉ số cho thấy khả năng dự đoán chính xác của mô hình (Trong các đối tượng được dự đoán có bao nhiêu đối tượng chính xác).
- Recall: Độ bao phủ là chỉ số cho thấy khả năng tổng quát và toàn diện của mô hình (Trong các đối tượng không được dự đoán có bao nhiêu đối tượng bị bỏ sót).
- mAP: Độ chính xác trung bình là tổng hợp của 2 chỉ số trên.



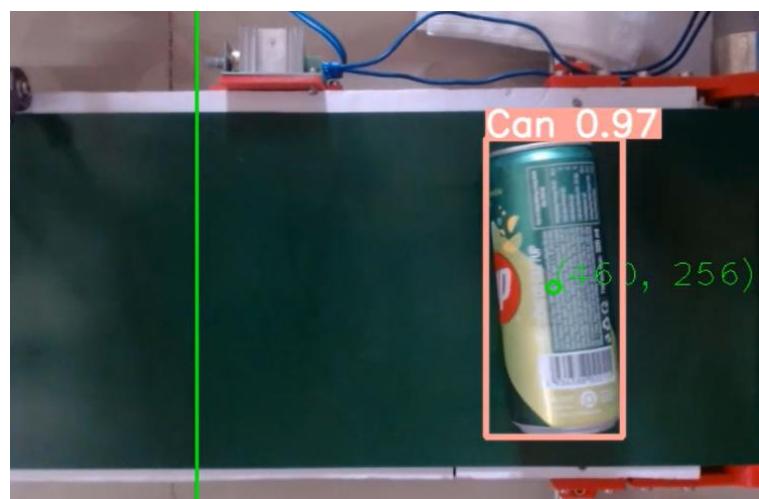
Hình 5. 11: Kết quả huấn luyện trên phần mềm Google Colab

#### 5.4 Tính toán và xác định vị trí đối tượng



Hình 5. 12: Hệ tọa độ Robot Delta và khung không gian làm việc Camera

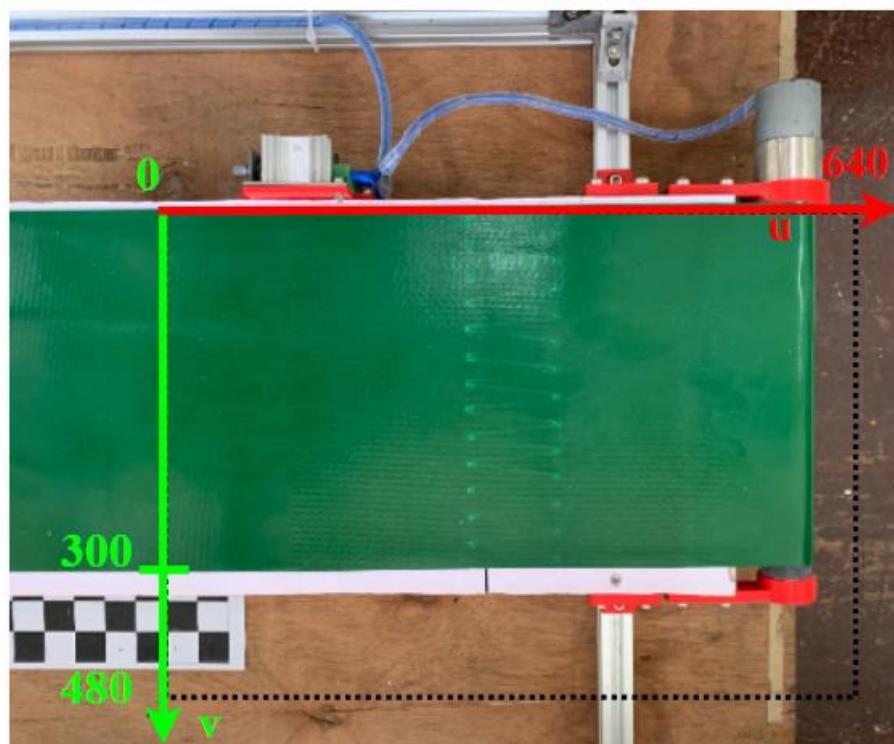
Sau khi phát hiện, phân loại và xác định được vị trí đối tượng ở trong khung hình, mô hình phân loại sẽ trả về vị trí của các bounding box của đối tượng trong hệ tọa độ pixel. Từ những vị trí này ta sẽ tính được vị trí tâm của đối tượng hay trọng tâm của bounding box để robot có thể làm việc dễ dàng hơn.



Hình 5. 13: Xác định tọa độ tâm của vật từ Bounding box

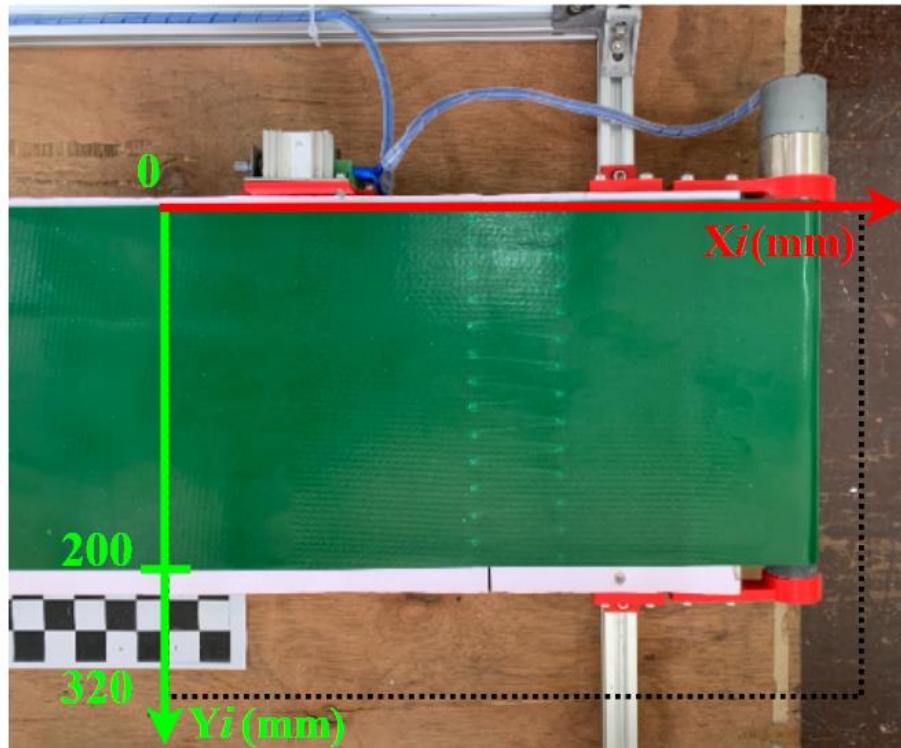
Hệ tọa độ pixel là hệ tọa độ chứa các điểm ảnh, tập hợp của các điểm ảnh này sẽ tạo nên một bức ảnh hoàn chỉnh. Mật độ điểm ảnh càng lớn thì bức ảnh càng sắc nét.

Với camera của nhóm sử dụng có độ phân giải là 1920x1080 nghĩa là trong một hình ảnh trả về từ camera sẽ có 1920x1080 điểm ảnh (pixel). Mô hình sẽ dựa vào tọa độ của điểm ảnh chứa đối tượng để xác định vị trí. Tuy camera nhóm sử dụng có thể đạt tới 1920x1080 pixel trên một khung hình nhưng nếu sử dụng độ phân giải lớn như vậy sẽ làm chậm quá trình xác định vị trí dẫn đến làm chậm quá trình phân loại. Nhóm quyết định giảm độ phân giải của camera xuống 640x480 để tăng tốc độ xử lý đồng thời cũng phù hợp với kích thước các ảnh mà nhóm đã tiến hành huấn luyện.



Hình 5. 14: Hệ toa độ Pixel trong vùng làm việc của camera

Khi có được vị trí tâm của vật trong hệ tọa độ Pixel ta phải dựa vào tương quan giữa 2 hệ tọa độ trong thực tế để chuyển vị trí đó về tọa độ Ảnh (mm) để có thể dễ dàng xử lý và tính toán.



Hình 5. 15: Hệ tọa độ Ảnh trong vùng làm việc của camera

Công thức chuyển đổi hệ trục tọa độ Pixel sang tọa độ Ảnh (mm):

$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \frac{S}{m} \cdot \begin{bmatrix} u \\ v \end{bmatrix} = \frac{320}{480} \cdot \begin{bmatrix} u \\ v \end{bmatrix} = \frac{2}{3} \cdot \begin{bmatrix} u \\ v \end{bmatrix} \quad (5.1)$$

Trong đó:

S: là khoảng cách thực tế (mm).

m: là mật độ pixel với khoảng cách đó.

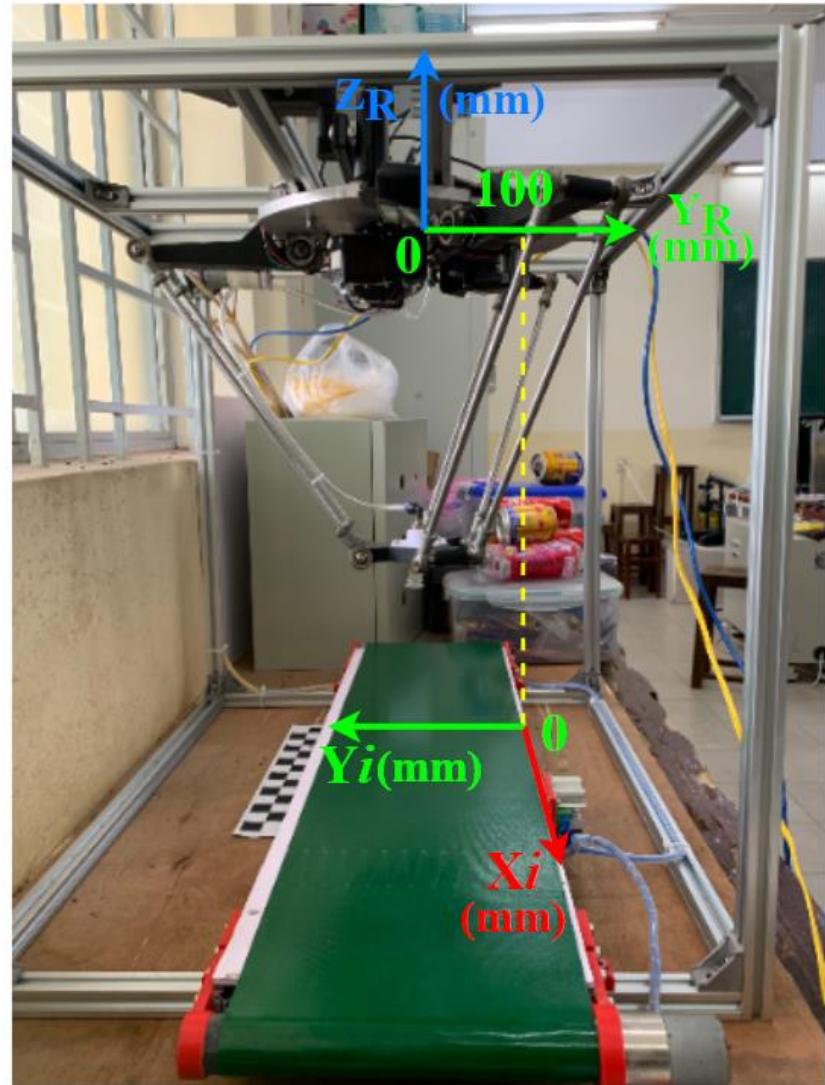
Ví dụ: Đối với camera nhóm sử dụng độ phân giải 640x480 thì 480pixel (m) tương ứng với 320mm (S).

Sau khi đã có được vị trí của vật trong tọa độ Ảnh (mm) ta sẽ tiến hành chuyển đổi sang tọa độ Robot để điều khiển robot tiếp cận đối tượng.

Ở đê tài lần này nhóm chỉ giới hạn điều khiển robot làm việc trên trục Y và trục Z không xét đến trục X nên giá trị  $X_R$  sẽ luôn bằng 0.

$$\begin{bmatrix} X_R \\ Y_R \\ Z_R \end{bmatrix} = \begin{bmatrix} 0 \\ Y_R \\ Z_R \end{bmatrix} \quad (5.2)$$

Vì đối tượng cần được phân loại chỉ nằm trên mặt phẳng băng tải nên nhóm cũng đã cố định các giá trị  $Z_R$  trong các bước hoạt động của robot. Vậy nhóm chỉ xét vị trí đối tượng trên trục Y của robot.



Hình 5. 16: Tương quan giữa hệ tọa độ Ảnh và hệ tọa độ Robot

Ta có thể thấy tương quan giữa 2 hệ tọa độ Ảnh và tọa độ Robot theo trục Y. Vì khoảng cách giữa gốc của 2 hệ tọa độ là 100mm và 2 hệ tọa độ này cùng phương ngược chiều nhau nên ta có công thức chuyển đổi hệ trục tọa độ Ảnh sang hệ trục tọa độ Robot theo trục Y.

$$[Y_R] = (-1) \cdot [Y_i] + 100 \text{ (mm)} \quad (5.3)$$

## CHƯƠNG 6: KIỂM CHỨNG VÀ ĐÁNH GIÁ KẾT QUẢ THỰC NGHIỆM

Sau khi đã huấn luyện mạng phân loại và xác định vị trí rác thải thành công nhóm sẽ tiến hành áp dụng vào mô hình Robot sau đó tiến hành kiểm chứng đưa ra những đánh giá. Đầu tiên là kiểm chứng động học, vị trí Robot, tiếp theo là đánh giá kết quả phân loại với 900 mẫu cho 2 đối tượng chai nhựa và lon và đưa ra những kết luận.

### 6.1 Kiểm chứng động học

#### 6.1.1 Kiểm chứng động học trên Visual Studio Code

Chương trình kiểm chứng động học thuận nghịch được tính toán trên phần mềm Visual Studio Code. Đối với kiểm chứng động học nghịch, nhóm sẽ nhập lần lượt 3 điểm  $P_x, P_y, P_z$  bất kì sẽ tính toán cho ra các góc  $\theta_1, \theta_2, \theta_3$  tương ứng. Ngược lại với kiểm chứng động học thuận, nhóm sẽ nhập lần lượt 3 điểm  $\theta_1, \theta_2, \theta_3$  bất kì và nhận được vị trí điểm cơ cấu chấp hành cuối  $P_x, P_y, P_z$ . Dưới đây là 2 bảng số liệu kiểm chứng động học trên Visual Studio Code.

Bảng 6. 1: Nhập tọa độ để kiểm chứng động học TH1

Trường hợp 1	Kết quả mô phỏng
Tọa độ $(\theta_1, \theta_2, \theta_3) = (30, 45, 60)$ (độ)	<p style="text-align: center;"><b>CONTROL KINEMATIC</b></p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <input style="width: 100px; height: 20px; border: 1px solid #ccc; border-radius: 5px; margin-right: 10px;" type="text" value="30.0"/> <p>Theta1</p> </div> <div style="text-align: center;"> <input style="width: 100px; height: 20px; border: 1px solid #ccc; border-radius: 5px; margin-right: 10px;" type="text" value="45.0"/> <p>Theta2</p> </div> <div style="text-align: center;"> <input style="width: 100px; height: 20px; border: 1px solid #ccc; border-radius: 5px; margin-right: 10px;" type="text" value="60.0"/> <p>Theta3</p> </div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 10px;"> <span style="border: 1px solid #00aaff; padding: 2px 10px; border-radius: 5px; color: #00aaff;">Forward</span> <span style="border: 1px solid #ff0000; padding: 2px 10px; border-radius: 5px; color: #ff0000;">Inverse</span> <span style="border: 1px solid #ffd700; padding: 2px 10px; border-radius: 5px; color: #ffd700;">Default</span> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 20px;"> <span style="border: 1px solid #00aaff; padding: 2px 10px; border-radius: 5px; color: #00aaff;">Px</span> <span style="border: 1px solid #ff0000; padding: 2px 10px; border-radius: 5px; color: #ff0000;">Py</span> <span style="border: 1px solid #ffd700; padding: 2px 10px; border-radius: 5px; color: #ffd700;">Pz</span> </div> <div style="display: flex; justify-content: space-around; align-items: center;"> <span style="border: 1px solid #00aaff; padding: 2px 10px; border-radius: 5px; color: #00aaff;">53.77</span> <span style="border: 1px solid #ff0000; padding: 2px 10px; border-radius: 5px; color: #ff0000;">-84.12</span> <span style="border: 1px solid #ffd700; padding: 2px 10px; border-radius: 5px; color: #ffd700;">-420.41</span> </div>

Điểm $P_x, P_y, P_z$ = (53.77, -84.12, -420.41) (mm)	<b>CONTROL KINEMATIC</b> Theta1 <input type="range" value="30.3"/> Theta2 <input type="range" value="45.31"/> Theta3 <input type="range" value="60.31"/> <input type="button" value="Forward"/> <input type="button" value="Inverse"/> <input type="button" value="Default"/> <table border="1"> <tr> <td>Px 53.77</td> <td>Py -84.12</td> <td>Pz -420.41</td> </tr> </table>	Px 53.77	Py -84.12	Pz -420.41
Px 53.77	Py -84.12	Pz -420.41		

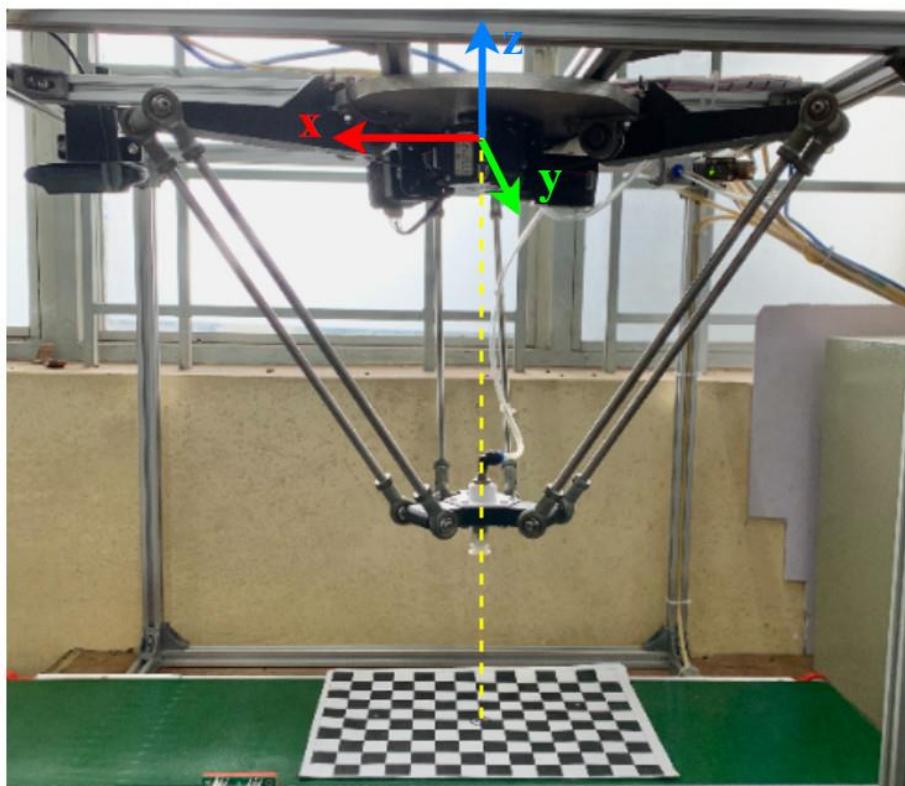
Bảng 6. 2: Nhập tọa độ để kiểm chứng động học TH2

Trường hợp 2	Kết quả mô phỏng			
Tọa độ $(\theta_1, \theta_2, \theta_3)$ = (29.58,43.75,16.67) (độ)	<b>CONTROL KINEMATIC</b> Theta1 <input type="range" value="29.58"/> Theta2 <input type="range" value="43.75"/> Theta3 <input type="range" value="16.67"/> <input type="button" value="Forward"/> <input type="button" value="Inverse"/> <input type="button" value="Default"/> <table border="1"> <tr> <td>Px -82.52</td> <td>Py -4.64</td> <td>Pz -378.78</td> </tr> </table>	Px -82.52	Py -4.64	Pz -378.78
Px -82.52	Py -4.64	Pz -378.78		
Điểm $P_x, P_y, P_z$ = (53.77, -84.12, -420.41) (mm)	<b>CONTROL KINEMATIC</b> Theta1 <input type="range" value="30.42"/> Theta2 <input type="range" value="44.17"/> Theta3 <input type="range" value="17.21"/> <input type="button" value="Forward"/> <input type="button" value="Inverse"/> <input type="button" value="Default"/> <table border="1"> <tr> <td>Px -82.52</td> <td>Py -4.64</td> <td>Pz -378.78</td> </tr> </table>	Px -82.52	Py -4.64	Pz -378.78
Px -82.52	Py -4.64	Pz -378.78		

Nhận xét: Kết quả khi nhóm kiểm chứng chương trình tính toán động học cho thấy, khi chuyển đổi động học thuận, nghịch trên giao diện điều khiển, sai số của các góc  $\theta_1, \theta_2, \theta_3$  không đáng kể (dưới  $1^\circ$ ).

### 6.1.2 Kiểm chứng động học nghịch trên checkerboard

Khoảng cách từ gốc tọa độ Robot Delta đến vị trí checkerboard nhóm chọn là tâm của checkerboard theo trục (x, y, z) lần lượt là (0, 0, -490) (mm) với mỗi ô của checkerboard với kích thước thực là 200x200 (mm). Đối với cách kiểm chứng động học thực tế trên checkerboard, nhóm sẽ tiến hành nhập lần lượt 2 vị trí bất kỳ và xem đổi chiều kết quả thực tế của cơ cấu chấp hành cuối so với checkerboard.



Hình 6. 1: Hệ tọa độ Robot so với checkerboard

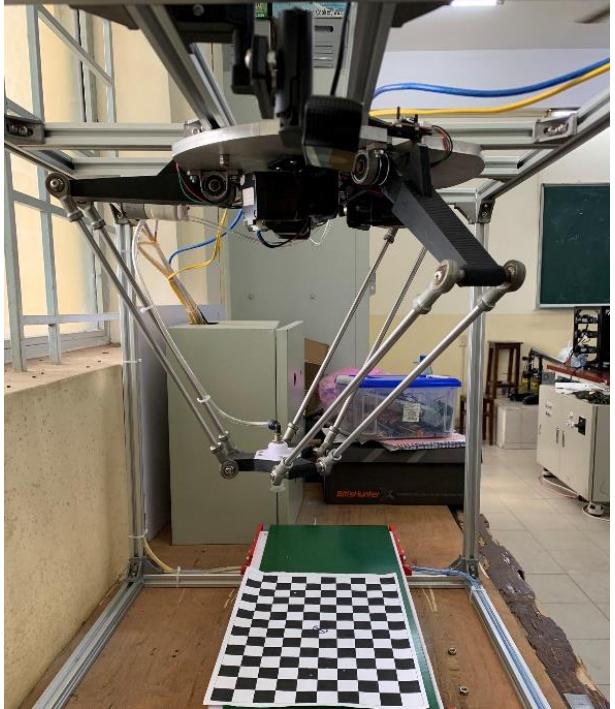
Bảng 6. 3: Nhập góc và vị trí kiểm chứng động học trên checkerboard

Trường hợp	Kết quả thực tế
Điểm $P_x, P_y, P_z =$ (0, -72.16 -405.72) (mm)	
Điểm $P_x, P_y, P_z =$ (46, 0, -440) (mm)	

Nhận xét: Kết quả khi nhóm kiểm chứng động học nghịch trên checkerboard cho thấy, khi nhập vị trí bất kỳ và cho tính toán động học nghịch thì vị trí cơ cấu cuối của Robot di chuyển đến không chênh lệch đáng kể với kết quả tính toán.

### 6.1.3 Kiểm chứng động học thực tế

Bảng 6. 4: Nhập tọa độ góc để kiểm chứng động học thực tế

Trường hợp	Kết quả thực tế
Tọa độ $(\theta_1, \theta_2, \theta_3) = (20, 0, 0)$ (độ)	
Tọa độ $(\theta_1, \theta_2, \theta_3) = (0, 40, 0)$ (độ)	

## Chương 6: Kiểm chứng và đánh giá kết quả thực nghiệm

Tọa độ $(\theta_1, \theta_2, \theta_3) = (0, 0, 30)$ (độ)	
Tọa độ $(\theta_1, \theta_2, \theta_3) = (20, 40, 30)$ (độ)	

## 6.2 Kiểm chứng YOLO

Để đánh giá kết quả của mô hình phát hiện rác thải chai nhựa và lon dựa trên YOLO, nhóm đã thực hiện huấn luyện và chạy thử nghiệm nhiều lần với số ảnh đầu vào khác nhau và hơn thế là trong các điều kiện ánh sáng khác nhau để khảo sát tính chính xác nhất về mức độ và khả năng phân loại của mô hình.



Hình 6. 2: Camera với ánh sáng bình thường



Hình 6. 3: Camera với ánh sáng tối

### 6.2.1 Kiểm chứng và thử nghiệm lần thứ nhất

Lần thứ nhất nhóm chỉ thực hiện huấn luyện tập dữ liệu với 515 ảnh đầu vào (280 ảnh là chai nhựa, 75 ảnh là lon và 160 ảnh có cả chai và lon) với cách dán nhãn phổ thông là dán bằng Bounding box và kiểm nghiệm trong hai điều kiện ánh sáng khác nhau (Ánh sáng bình thường và ánh sáng tối). Bên dưới nhóm sẽ trình bày một trong những ma trận confusion matrix đã được chọn lọc qua nhiều lần chạy thực nghiệm để kiểm chứng kết quả phân loại hai đối tượng chai nhựa và lon.

Bảng 6. 5: Huấn luyện với dữ liệu 515 ảnh (ánh sáng bình thường)

Thực tế Dự đoán	Chai	Lon
Chai	45	26
Lon	5	24

Kết quả thu được: Precision = 0.6338, Recall = 0.9, F1-score = 0.7438

Bảng 6. 6: Huấn luyện với dữ liệu 515 ảnh (ánh sáng tối)

Thực tế Dự đoán	Chai	Lon
Chai	46	8
Lon	4	42

Kết quả thu được: Precision = 0.8518, Recall = 0.92, F1-score = 0.8845

Nhận xét: Với trường hợp ánh sáng bình thường thì số lượng chai nhận dạng tương đối chính xác cao, số lượng lon bị nhận dạng sai thành chai tương đối cao (Vì nhóm huấn luyện ảnh đầu vào là chai nhiều hơn đáng kể). Với trường hợp bị hạn chế ánh sáng (ánh sáng tối) thì tỉ lệ lon và chai nhận dạng khá chính xác, đặc biệt hơn là tỉ lệ nhận dạng chính xác lon tăng lên đáng kể.

Kết luận: Qua nhận xét trên thấy rằng điều kiện ánh sáng ảnh hưởng đến độ chính xác của việc phân loại chai nhựa và lon. Trong điều kiện ánh sáng tối thì sẽ nhận dạng ổn định hơn, vì sẽ hạn chế được nhiều từ dữ liệu môi trường xung quanh.

### 6.2.2 Kiểm chứng và thử nghiệm lần thứ hai

Tiếp đến lần kiểm chứng thử nghiệm lần thứ hai, nhóm thực hiện làm giàu dữ liệu ảnh đầu vào hơn bằng cách tăng dữ liệu ảnh đầu vào từ 515 ảnh lên 1026 ảnh (Với khoảng 342 ảnh chai, 354 ảnh lon và 330 ảnh cả chai và lon) với cùng cách dán nhãn là Bounding box và cũng kiểm nghiệm hai điều kiện ánh sáng khác nhau (Ánh sáng bình thường và ánh sáng tối). Bên dưới nhóm sẽ trình bày một trong những trận confusion matrix đã được chọn lọc qua nhiều lần chạy thực nghiệm để kiểm chứng kết quả phân loại hai đối tượng chai nhựa và lon.

Bảng 6. 7: Huấn luyện với dữ liệu 1026 ảnh (ánh sáng bình thường)

Thực tế Dự đoán	Chai	Lon
Chai	48	1
Lon	2	49

Kết quả thu được: Precision = 0.9796, Recall = 0.96, F1-score = 0.9697

Bảng 6. 8: Huấn luyện với dữ liệu 1026 ảnh (ánh sáng tối)

Thực tế Dự đoán	Chai	Lon
Chai	47	1
Lon	3	49

Kết quả thu được: Precision = 0.9791, Recall = 0.94, F1-score = 0.9591

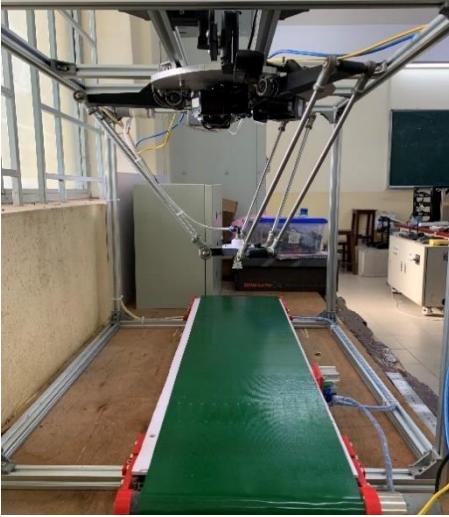
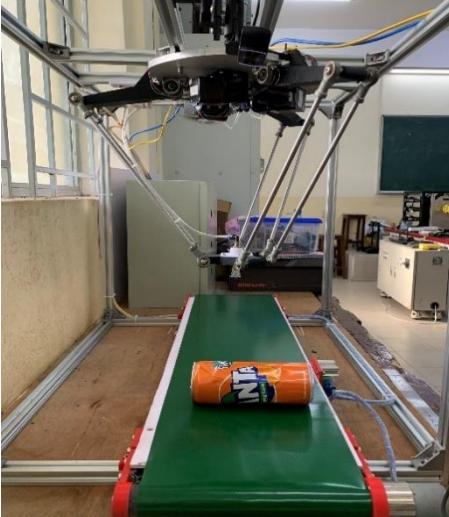
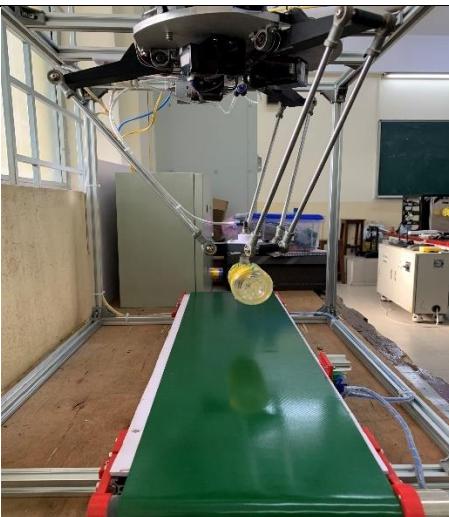
Nhận xét: Khi nhóm tiến hành tăng số ảnh đầu vào lên gấp đôi so với ban đầu thì kết quả nhận dạng được cải thiện đáng kể. Hầu như lon và chai được nhận dạng chính xác hoàn toàn. Chất lượng hệ thống phân loại có kết quả tốt với sai số tương đối nhỏ.

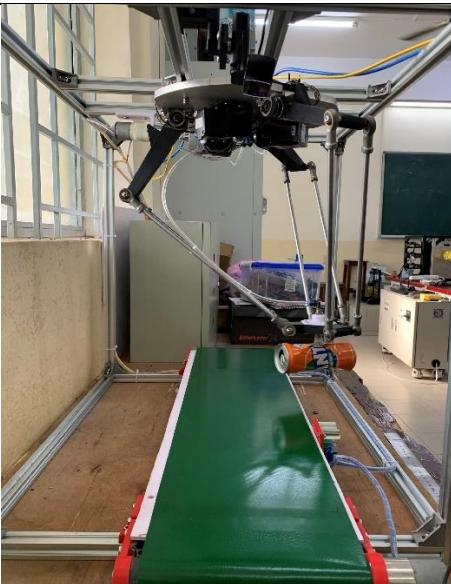
Kết luận: Qua 2 bảng so sánh phân loại với 1026 ảnh đầu vào ở trên với hai điều kiện ánh sáng bình thường và ánh sáng tối thì độ chính xác không chênh lệch nhiều.

### 6.3 Đánh giá kết quả toàn hệ thống

Nhóm tiến hành chạy thực nghiệm mô hình phân loại rác nhựa và lon để kiểm chứng tính liên tục của hệ thống, thả lần lượt chai nhựa (bottle) và lon (can) xen kẽ để mô hình Robot Delta phân loại. Bên dưới là bảng quy trình hoạt động của hệ thống.

Bảng 6. 9: Quy trình giai đoạn hoạt động của hệ thống

Giai đoạn trong hệ thống	Mô tả
	Ban đầu mở điện Robot Delta sẽ về vị trí Home với các góc $\theta_1, \theta_2, \theta_3 = 0$ (độ)
	Tiếp đến, thả chai hoặc lon lần lượt vào băng tải, camera sẽ phát hiện, nhận dạng và gửi vị trí đối tượng về Robot Delta
	Robot nhận tín hiệu và di chuyển đến vị trí chai (bottle) hoặc lon (can) và thực hiện thao tác hút vật lên khỏi băng tải

	Nếu tín hiệu từ khói thị giác máy tính xác định được là chai (bottle) thì Robot sẽ nhận tín hiệu rồi đưa sang bên trái và thả chai xuống
	Nếu tín hiệu từ khói thị giác máy tính xác định được là lon (can) thì Robot sẽ nhận tín hiệu rồi đưa sang bên phải và thả lon xuống
	Khi đã hoàn tất 1 chu trình thì Robot sẽ về vị trí đặt để đợi đối tượng tiếp theo

### Đánh giá hệ thống phân loại với cấu trúc mạng nơ-ron lần thứ nhất:

Với cách dán nhãn Bounding box, tập dữ liệu ảnh 515 ảnh (280 ảnh là chai nhựa, 75 ảnh là lon và 160 ảnh có cả chai và lon) được chia theo tỷ lệ (training set = 70%, valid set = 20%, test set = 10%). Chạy thực nghiệm trong điều kiện ánh sáng bình thường.

Bảng 6. 10: Bảng đánh giá hệ thống lần 1

Lần thử	Số lượng chai	Số lượng lon	Số lượng chai nhận đúng	Số lượng chai hút hụt	Số lượng lon nhận đúng	Số lượng lon hút hụt
1	50	50	45	5	20	6
2	50	50	43	6	23	8
3	50	50	40	5	27	7

Đánh giá trung bình cộng của thông số giữa các lần thử.

Precision = 0.6153, Recall = 0.8533, F1-score = 0.715

Nhận xét: Qua bảng đánh giá hệ thống lần thứ nhất với tập 515 ảnh dữ liệu đầu vào (Số lượng ảnh chai lớn hơn nhiều so với ảnh lon) thì tỉ lệ chai được nhận dạng chính xác khá cao còn tỉ lệ lon nhận dạng chính xác còn khá thấp. Tỉ lệ chai và lon hút không thành công có mà không đáng kể do hạn chế về cơ cấu chấp hành

Kết luận: Cần cải thiện mô hình mạng nơ-ron lần 1 bằng cách thử lọc bớt ảnh nhiễu và tăng thêm số lượng ảnh đầu vào.

### Đánh giá hệ thống phân loại với cấu trúc mạng nơ-ron lần thứ hai:

Với cách dán nhãn Bounding box, tập dữ liệu ảnh 1026 (Với khoảng 342 ảnh chai, 354 ảnh lon và 330 ảnh cả chai và lon) ảnh được chia theo tỷ lệ (training set = 70%, valid set = 20%, test set = 10%). Chạy thực nghiệm trong điều kiện ánh sáng bình thường.

Bảng 6. 11: Bảng đánh giá hệ thống lần 2

Lần thử	Số lượng chai	Số lượng lon	Số lượng chai nhận đúng	Số lượng chai hút hụt	Số lượng lon nhận đúng	Số lượng lon hút hụt
1	50	50	49	3	49	4
2	50	50	48	3	48	2
3	50	50	48	2	49	3

Đánh giá trung bình cộng của thông số giữa các lần thử

Precision = 0.9731, Recall = 0.9667, F1-score = 0.9698

Nhận xét: Qua bảng đánh giá hệ thống lần 2 khi tăng gấp đôi số lượng ảnh đầu vào (Số lượng ảnh lon với số lượng ảnh chai gần nhau) và lọc bớt ảnh nhiễu thì tỉ lệ chai và tỉ lệ lon được nhận dạng chính xác cao, gần như đúng hoàn toàn. Chỉ ra rằng tỉ lệ gán nhầm nhau cho hai đối tượng giảm rõ rệt. Vẫn xuất hiện 1 ít tỉ lệ hút không thành công do hạn chế về cơ cấu chấp hành đầu cuối.

Kết luận: Việc tăng số lượng ảnh đầu vào và lọc bớt ảnh nhiễu có độ chính xác cao. Vậy sử dụng mô hình mạng nơ-ron thứ 2 mang tính hiệu quả hơn.

### **Đánh giá hệ thống phân loại với cấu trúc mạng nơ-ron lần thứ ba:**

Với cách dán nhãn Bounding box, tập dữ liệu ảnh 1026 (Với khoảng 342 ảnh chai, 354 ảnh lon và 330 ảnh cả chai và lon) ảnh được chia theo tỷ lệ (training set = 70%, valid set = 20%, test set = 10%). Chạy thực nghiệm trong điều kiện ánh sáng tối.

Bảng 6.12: Bảng đánh giá hệ thống lần 3

Lần thử	Số lượng chai	Số lượng lon	Số lượng chai nhận đúng	Số lượng chai hút hụt	Số lượng lon nhận đúng	Số lượng lon hút hụt
1	50	50	43	7	45	6
2	50	50	45	7	48	4
3	50	50	47	6	47	5

Đánh giá trung bình cộng của thông số giữa các lần thử

Precision = 0.931, Recall = 0.9, F1-score = 0.9152

Nhận xét: Qua bảng đánh giá kết quả lần thứ 3, nhóm nhận thấy tỉ lệ nhận dạng chính xác của hai đối tượng chai nhựa và lon vẫn khá cao. Tuy nhiên tỉ lệ không cao bằng trong điều kiện ánh sáng bình thường. Bên cạnh đó thì xuất hiện trường hợp nhận dạng không thành công và bỏ qua đối tượng (không xác định được đối tượng).

Kết luận: Vậy ánh sáng có ảnh hưởng đến quá trình nhận dạng và phân loại. Và nên sử dụng mô hình mạng nơ-ron thứ 2 (trong điều kiện ánh sáng bình thường).

## CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 7.1 Kết luận

Sau quá trình nghiên cứu và thực hiện đề tài: “Thiết kế mô hình Robot Delta ứng dụng phân loại rác thải”, nhóm đã đạt được những kết quả như sau:

- Tìm hiểu cấu trúc và thiết kế mô hình Robot Delta hoàn chỉnh trên phần mềm Solidworks.
- Thi công phần cứng Robot Delta và tủ điện đảm bảo an toàn, thẩm mỹ.
- Tính toán động học thuận, nghịch và kiểm nghiệm động học trên phần mềm Visual Studio Code. Xác định không gian làm việc cho Robot trên phần mềm Matlab.
- Tính toán và điều khiển Robot trên phần mềm Arduino IDE và thiết kế giao diện người dùng trên phần mềm Visual Studio Code.
- Xây dựng và huấn luyện mô hình YOLOv5 để phân loại 2 loại rác thải (chai nhựa và lon) với độ chính xác cao.
- Kết hợp giải thuật phân loại điều khiển Robot Delta thực hiện chu trình phân loại.
- Hoàn thành luận văn.

Những hạn chế còn tồn tại trong quá trình thực hiện đề tài như sau:

- Tốc độ phân loại đối tượng chưa cao, khi phân loại xong 1 đối tượng thì mới bắt đầu tiếp nhận đối tượng tiếp theo.
- Độ chính xác của quá trình phân loại chịu ảnh hưởng từ ánh sáng và màu sắc của môi trường xung quanh
- Việc thiết kế và thi công phần cứng chưa tối ưu, khi Robot hoạt động ở tốc độ cao thì Robot có hiện tượng rung, tốc độ băng tải chưa nhanh.
- Cơ cấu chấp hành đầu cuối cũng hạn chế với một số đối tượng.

### 7.2 Hướng phát triển

- Điều chỉnh lại phần cứng (thiết kế lại một số chi tiết Robot) và thay thế một số thiết bị (động cơ, driver) để tốc độ hoạt động của Robot tối ưu.
- Có thể thiết kế thêm bộ điều khiển và quy hoạch quỹ đạo để tăng độ ổn định của Robot khi phân loại.
- Thay đổi cơ cấu chấp hành đầu cuối để cải thiện tính ứng dụng phân loại và tăng tỉ lệ hút thành công, điều tiết lượng khí tốt hơn.
- Tăng số lượng đối tượng huấn luyện để phân loại đa dạng hơn.

## TÀI LIỆU THAM KHẢO

- [1] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi "You Only Look Once:Unified, Real-Time Object Detection".
- [2] Do Thuan,"EVOLUTION OF YOLO ALGORITHM AND YOLOV5: THE STATE-OF-THE-ART OBJECT DETECTION ALGORITHM".
- [3] Mzavatsky,"Delta robot kinematics",<https://hypertriangle.com/~alex/delta-robot-tutorial/>.
- [4] Simplilearn,"What is a Confusion Matrix in Machine Learning?",<https://www.simplilearn.com/tutorials/machine-learning-tutorial/confusion-matrix-machine-learning>.
- [5] Nguyễn Thanh Tuán,"Convolutional neural network", 2019,<https://nttuan8.com/bai-6-convolutional-neural-network/>.
- [6] Andrew Ng,"YOLO Algorithm",[https://www.youtube.com/watch?v=9s\\_FpMpdYW8](https://www.youtube.com/watch?v=9s_FpMpdYW8).
- [7] LearnOpenCV,"YOLOV5: How to Train a Custom YOLOv5 Object Detector | Official YOLOv5",<https://www.youtube.com/watch?v=Ciy1J97dbY0&t=1724s>.
- [8] Nafiz Shahriar, "What is Convolutional Neural Network - CNN (Deep Learning)", <https://nafizshahriar.medium.com/what-is-convolutional-neural-network-cnn-deep-learning-b3921bdd82d5>.
- [9] <https://blog.roboflow.com/how-to-train-yolov5-on-a-custom-dataset/>.
- [10]<https://colab.research.google.com/github/ultralytics/yolov5/blob/master/tutorial.ipynb>.
- [11] [https://docs.ultralytics.com/yolov5/tutorials/train\\_custom\\_data/#train-on-custom-data](https://docs.ultralytics.com/yolov5/tutorials/train_custom_data/#train-on-custom-data).
- [12] Nguyễn Thế Hùng, Lê Tiến Thành,"Ứng dụng Robot tích hợp thị giác máy tính phân loại rác nhựa", Đồ án tốt nghiệp Đại Học Sư Phạm Kỹ Thuật TPHCM, 2022.

## Tài liệu tham khảo

---

- [13] Lê Đức Kông, Nguyễn Văn Thiên, "Phân loại sản phẩm linh hoạt ứng dụng Robot và xử lý ảnh", Đồ án tốt nghiệp Đại Học Sư Phạm Kỹ Thuật TPHCM, 2022.
- [14] Maker Store, "CNC Shield Guide", <https://www.makerstore.com.au/wp-content/uploads/firebase/publications/CNC-Shield-Guide-v1.0.pdf>.
- [15] Adrian Rosebrock, "Intersection over Union (IoU) for object detection", <https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>.
- [16] Nguyễn Thanh Tuấn, "Deep learning cơ bản version 2", 2020.
- [17] <https://www.youtube.com/watch?v=z7-JZMNIiSYY>.
- [18] <https://www.youtube.com/watch?v=l9wR7xoTys4>.
- [19] <https://www.youtube.com/watch?v=IV8v6G1wmsw>.

## **PHỤ LỤC**

Chương trình tính toán động học, điều khiển hệ thống và giao diện điều khiển:

<https://drive.google.com/drive/folders/1UAWp6Wc1GipJB-ZcPpAmCkJOPKoEVTCA?usp=sharing>