People’s Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University M’Hamed BOUGARA – Boumerdes

Institute of Electrical and Electronic Engineering
Department of Power and Control

Final Year Project Report Presented in Partial Fulfilment of the Requirements for the Degree of

MASTER

In Electrical and Electronic Engineering
Option: Control

Title:

Sensor-based Approach for Manipulation Tasks Execution with a Collaborative Robot

Presented by:
- Ishak BEY
- Seif Eddine SERAY

Supervisor:
- Mrs Nabila DERRAGUI

Co-supervisor:
- Dr Isma AKLI

Academic Year : 2023/2024

Dedications

In the name of Allah, the Most Merciful, the Most Compassionate. All praise be to Allah, the Lord of the worlds. May prayers and peace be upon the Prophet Muhammad, His servant and messenger.

This study is dedicated wholeheartedly to our beloved parents, who have been a constant source of inspiration, providing us with hope and encouragement when we faced moments of doubt and despair. They have always been there, guiding and supporting us.

Lastly, we dedicate this thesis to everyone who has helped us along the path, whether it has been through nice deeds, encouraging words, or times of laughing. Your presence in our life has enhanced our encounters and given our journey a deeper sense of significance. This thesis is an appreciation of everyone's combined efforts who have helped and encouraged us over this journey. We are grateful that you are our sources of support.

Acknowledgments

First and foremost, we would like to extend our deepest gratitude to God Almighty who provided us with his blessing and the opportunity to successfully conclude our project.
We would like to express our deepest gratitude to our supervisors Mrs Akli Isma and Mrs Derragui Nabila, We are immensely grateful for their valuable advice regarding our research and future career prospects. Their willingness to assist us, steadfast encouragement and continuous support were instrumental in making this project a reality. We deeply appreciate their patience, dedication, and unwavering belief in our abilities.
We extend our gratitude to the CDTA research center for their invaluable support and resources throughout the duration of our internship. The guidance and expertise provided by the researchers and staff at the CDTA have been instrumental in the successful completion of this project.

Abstract

The quantity of domestic waste is constantly increasing, resulting in an overload of wastes. Recycling became imperative for ecologic and economical considerations in order to apply circular economy rules. In general, humans are recruited in waste sorting plants for performing the dangerous and tedious sorting tasks. Automation is thus necessary.
This report deals with the problem of waste sorting automation. In this work, we propose a robotised automated waste sorting system with the ability to detect human presence and cooperate with the human operator when needed. In order to accomplish the solution, We used computer vision based waste recognition and classification using YOLOv8, human presence detection using RFID (Radio Frequency Identification) tags which allows human-robot collaboration and a robotic arm to pick and place waste objects. The system was built in ROS (Robot Operating System), starting the work with a virtual environment for validation before implementing the system on a real collaborative robotic arm. The system' implementation led to good results concerning the waste detection part. Objects classes were identified accurately with acceptable precision in object positioning. The human detection was accurate within the range of RFID modules and robot movements were executed without any issues.

List of Figures

List of Tables

List of Abbreviations

RFID Radio Frequency Identification

ROS Robot Operating System

PLC Programmable Logic Controller

SCARA Selective Compliance Articulated Robot Arm

AGV Automated Guided Vehicle

YOLO You Only Look Once

SCARA Selective Compliance Articulated Robot Arm

AI Artificial Intelligence

URDF Unified Robot Description Format

UI User Interface

RSS Received Signal Strength

API Application Programming Interface

Contents

General introduction

Robotics became a very important discipline combining electronics, mechanics and computer science. Robots are widely used in industry for process automation. Furthermore, Artificial intelligence technologies are in constant development. This introduces new aspects for combining robotic systems with artificial intelligence to have a more sophisticated and more intelligent solutions. Moreover, collaborative robots (cobots) field of study holds considerable interest for researchers, who are actively exploring their possibilities of integration in industrial worksites in different scenarios such as assembly, welding, or sorting facilities.

Meanwhile, the entire globe is facing difficulties with waste management problems. The surge in pollution rates has driven to the transfer to a circular economy. This type of economy focuses on recycling the products in order to produce less waste. As a result, greater attention is being given to the development of advanced recycling facilities. Many of these facilities rely on human operators to perform the waste sorting tasks, while some others have opted for automatic systems to do the job.

The aim of this work is to automate the waste sorting task by designing an intelligent system based on the collaborative robot arm Kuka LBR iwaa r800 that is able to manipulate different waste objects, after acquiring their type using a visual sensor and an object detection algorithm. In addition, The system allows for cooperation between human operators and the robot.

An artificial intelligence object detection model was trained on a dataset curated for the waste sorting purpose. In addition, A software system was developed for the waste sorting purpose. First, the system was validated in simulation in a dedicated simulation platform. After that, the system was implemented using the actual KUKA lbr robot.

This report is divided into three chapters:

- Chapter One: An overview about robotics is presented. Then, the chapter introduces the literature review, where previous works and projects in the field of waste sorting are mentioned.

- Chapter Two: It states the problematic of our project and the proposed solution. It also includes the general architecture of the system with its different parts, along with definitions and explanations for each part.

- Chapter Three: The implementation steps of the system are shown in details. The results are then mentioned and discussed.

The report finishes with a general conclusion and further-work propositions.

Chapter 1

Robotics for Solid Waste Sorting

1.1 Introduction

In this section, we start first by giving definitions about robotics in general, then defining robotic arms specifically. Next, we explore previous works and contributions to the field of waste sorting concerning automatic systems, and explaining the various techniques used in different categories.

1.2 Robotics

1.2.1 Definition

Robotics is a technical science that deals with the conception, construction and operation of robots. A robot can be defined as a machine capable of executing a set of tasks in order to achieve a defined goal with a certain degree of autonomy and robustness [1]. The robots could be programmed using computers or special controllers. Throughout the years, robots have been in use in multiple fields especially in the industry and manufacturing. In addition, researchers and development firms were pushing the boundaries trying to improve the overall technology characteristics and also trying to find new applications for the robots. This resulted in a wide range of solutions with various types of robots capable of many tasks such as exploring, manufacturing and transportation.

1.2.2 History of Industrial Robots

The word ”robot” originates from the Czech word ”robota” that means ”Forced labour”. First mentioned in the novel ”RUR” (Rossum’s Universal Robots 1920) written by Czech novelist Karel Apek. The concept of the robots was then brought to live in the years after. The first real industrial robot was created by Unimation in 1961 under the name ”Unimate”. It was quickly adopted in automotive by many firms for welding and pick and place applications. First generations of robots had very limited capabilities and relied mainly on pneumatic actuators. The next generation of industrial robots used microprocessors and PLCs (Programmable Logic Controller), this gave the opportunity to program the robot into doing more complicated maneuvers with higher precision. During this robot generation, the use of hydraulic actuators was explored; however, manufacturers preferred electric actuators due to their high precision and efficiency which resulted in more powerful and cos-effective solutions. Following 1978, advancement in electronics and instrumentation led to more sophisticated robots equipped with sensors to better perceive the surroundings[2].

1.2.3 Common Robot Types

Some of the common robot types are:

- **Robotic Arms:** This type of robot is defined according to the standard ISO 8373:2021 as ”interconnected set of links and powered joints of the manipulator, between the base and the wrist”[3]. This type of robots have a fixed base and manipulate the objects on its surroundings within its range[2].

- **Cartesian Robots:** Defined in the standard ISO 8373:2021 as ”manipulator which has three prismatic joints, whose axes form a Cartesian coordinate system”[3]. This type of robots is well suited for applications requiring extensive horizontal movements.

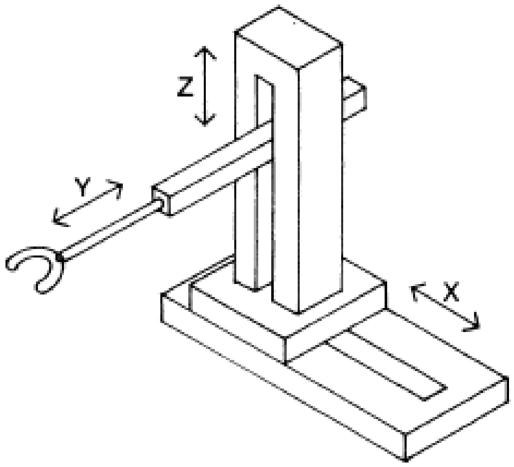


Figure 1.1: Cartesian robot[2]

- **SCARA (Selective Compliance Articulated Robot Arm):** Defined in the standard ISO 8373:2021 as "manipulator which has two parallel rotary joints to provide compliance in a selected plane"[3]. First created in Japan in 1979. This type of robot revolutionized assembly tasks at that time[2]. Figure 1.2 shows an example of a SCARA robot.

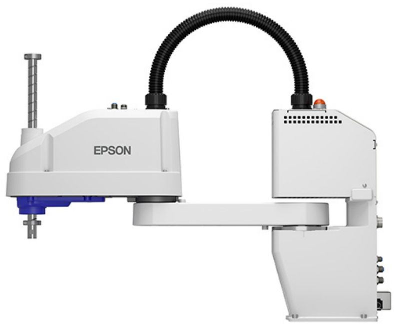


Figure 1.2: SCARA robot[4]

- **Humanoid Robots:** Defined in the standard ISO 8373:2021 as "robot with body, head and limbs, looking and moving like a human"[3].

- **Gantry:** The base of this type of robot moves on rails mounted on the ceiling of the cluttered environment. The arm then is lowered to manipulate object and executed the required tasks[2]. Figure 1.3 shows an example of a Gantry robot.

Figure 1.3: Two axis Gantry robot[5]

- **AGVs (Automated Guided Vehicles) :** As its name suggests, AGVs are autonomous vehicles used usually for transportation of heavy items. It travels around predefined paths marked in the floor. Most AGVs stop when it encounters an obstacle in its path[2].
- **Mobile Robots :** Also known as AMRs (Autonomous Mobile Robots). This type of robots is not limited to certain paths but it is able to move freely in the working area and re-route its path around obstacles autonomously[2].

1.2.4 Robotic Arms

Robot arms (manipulators) are electro-mechanichal structures composed of a set of links that are connected with joints. The main types of joints are linear joints and rotational joints. The first one makes a link move in one Cartesian axis according to the parent link. Whereas, the latter vary the angle between the parent and child links creating rotational movements. Generally, the last link is connected with an end-tool that fits the purpose of the robot.

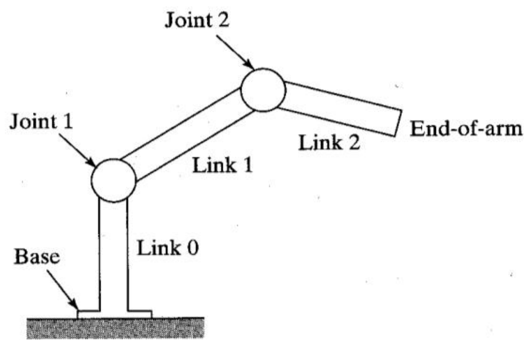


Figure 1.4: Robotic arm with rotational joints[6]

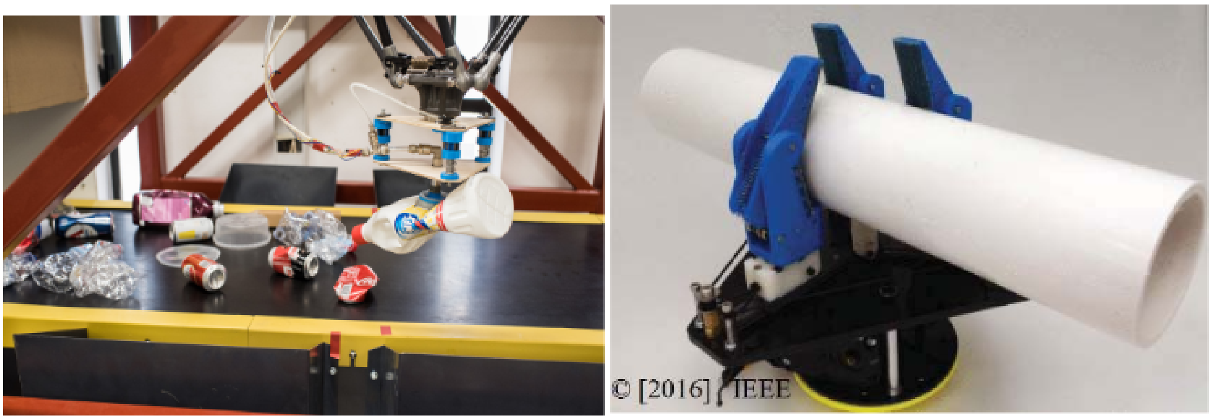
1.2.4.1 Degree of Freedom

It is important to differentiate between the degree of freedom of the system and the degree of freedom of the end tool. The degree of freedom of the end tool (end-effector or gripper) of the robot is the number of independent coordinates needed to express the position of the tool with all its parts. Therefore, more joints that make the robot move in independent coordinates will result in a higher degree of freedom. The position of the tool center point could be described by three coordinates, one for each Cartesian axis, another three coordinates can describe the orientation of the end tool. Hence, a maximum of 6 DOF is enough to place the end tool in any possible position and orientation within the workspace of the robot. On the other hand, the degree of freedom of the whole system is the total number of joints, even if some added joints does not change the DOF of the end-tool. Some authors refer to the number of these added joints as the the degree of mobility. The added joints give more flexibility to the robot and could be really beneficial if the application requires complicated manoeuvres or obstacles are present inside the workspace of the robot[2].

1.2.4.2 End Effectors

Grippers are a type of end effectors of robotic arms. They are used to perform pick and place operations on targeted objects. There exist many types of grippers, and each one is designed to handle a specific spectrum of object features. In [7], many configurations were compared. For waste sorting applications, we will be interested in two types of grippers: grippers with fingers and vacuum suction grippers.

While grippers with fingers can achieve higher payloads and a more powerful grip on the objects, The suction grippers are more suitable for handling unknown objects with arbitrary dimensions and orientations. In [8], suction grippers were examined in waste sorting tasks, and it was reported that they scored high accuracies.



(a) suction gripper[8]

(b) gripper with fingers[7]

Figure 1.5: Suction and finger grippers

1.2.4.3 Collaborative Robots

Typically, industrial robots should be isolated from human workers to ensure the safety of the operators. In contrast, collaborative robots or co-bots are a new trend of robots designed with improved and sophisticated safety measures allowing the human operator and the robot to work together in a shared workspace. In addition, Cobots are lighter, more flexible and very cheap compared to the industrial robots. These features make the Cobots a good option for exploring the use of robots in previously ignored tasks. Another key feature that makes it easy to implement in various scenarios is that it could be programmed by just teaching the required motions manually by moving it through the desired path. The main firms that produce cobots are: Universal Robotics, ABB, Kuka, Omron and etc[9].

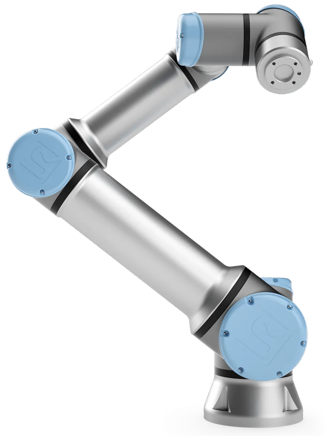


Figure 1.6: UR16e collaborative robot [10]

1.3 Automated Solid Waste Sorting Mechanisms

Pollution has become a growing concern throughout the globe. It causes many environmental issues such as global warming, air pollution and deforestation. A major factor for these issues is bad waste management. One of the proposed solutions to reduce the effect of poor waste management is recycling. Therefore, many recycling plants were deployed all around the world. In order to perform the recycling more conveniently, the wastes are sorted to treat each type of material apart (plastics, glass, cardboard, metals...). Classical sorting methods involve human manipulators which is unhealthy and dangerous in some applications. Hence, automated solutions that minimize human interactions are safer since it will reduce human contact with dangerous items. For some scenarios where the sorting is straightforward, the automated solutions are more robust because they operate in higher speeds and for longer hours; However, In some cases where there are some nuances in the sorting phase, Collaboration between the human operator and the robot is the best combination to achieve high speed and precision. The sorting process have been evolving for a long time and many techniques were used along the way. These techniques can be divided into two main categories:[11]

- **Direct Sorting :** Direct sorting techniques are the ones where objects are sorted with the help of their physical properties.
- **Indirect Sorting :** In this category, the sorting phase is divided into two steps: Identification of the type of object, and moving it to the right place or though the right path of the specific category of the item. [12]

1.3.1 Direct Sorting

The main used methods can be classified as:

- **Magnetic sorting :** magnets are used to separate ferromagnetic objects.

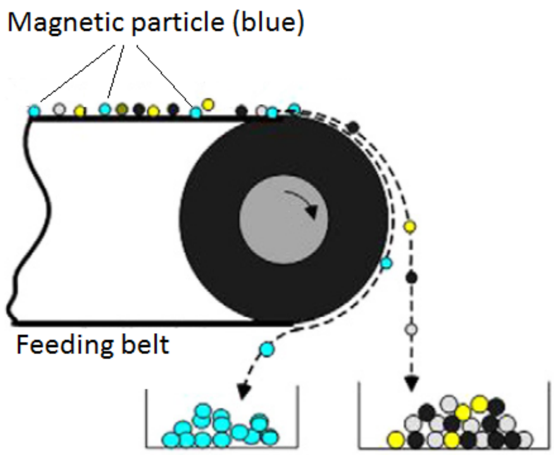


Figure 1.7: Schematic diagram of magnetic separation

- **Gravity based sorting :** objects can be separated depending on their densities. an effective method is Sink-float sorting.
- **Electric conductivity based sorting :** an example method is Eddy current sorting. Where non-ferrous metals are separated from insulators, by applying a varying magnetic field to objects moving in a conveyor. This induces eddy currents in metals which cause them to be carried away further than other materials.

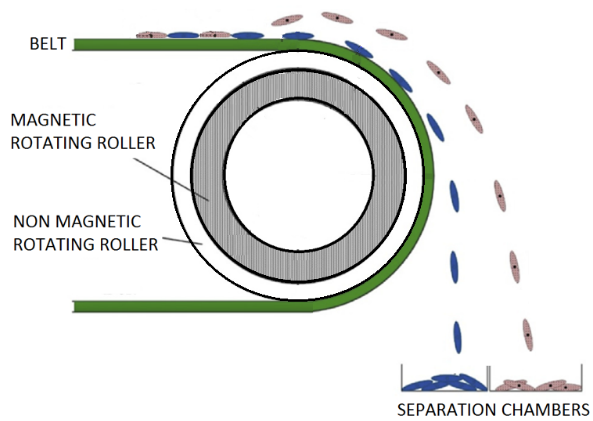


Figure 1.8: Eddy current sorting[13]

1.3.2 Indirect Sorting

1.3.2.1 Objects Identification Methods

Several approaches might be used to detect the type of items to be sorted which are: sensor based approach or vision based approach or a combination of the two.

Sensor based object detection

In the sensor based approach the most used sensors are: Capacitive proximity sensors, Ultrasonic sensors, Infrared (IR) sensors, Moisture sensors, Voltage sensors, Inductive proximity sensors, Humidity sensors, Gas sensors. Some systems start by isolating metallic items using inductive sensors and electromagnetic sensors. In some cases, a voltage sensor is used to differentiate between organic and plastic waste. Capacitive proximity sensors are used for non-contact detection. Ultra-sonic sensors help in level indication. Even if several sensors are used for item detection and classification, visual methods proves to be more efficient and accurate.

Computer Vision Based Object Detection

When it comes to computer vision detection, the sorting of objects is done by analysing images using artificial intelligence algorithms. These algorithms can be divided into two main categories: Conventional machine learning and deep learning. Many algorithms were used for object detection purposes in waste sorting applications such as: LSTM[14], KNN[15], SVM[16], YOLOv3[17], Mask-RCNN, YOLOv5, YOLOv7...[18]. These algorithms had different performances depending on the objects to be detected, the size of the database that the module is trained on, the architecture of the model and other parameters. But, regression-based Deep learning algorithms such as YOLO are the most suitable for real time detection because of their high speed and performance[19]. This made our work focus on deep learning methods rather than other classical ones. In a comparative research, three deep learning algorithms (Faster-RCNN, YOLOv4, and SSD) were compared. Their task was to classify vehicles based on their type. The results showed that YOLO was the best among the tested algorithms, proving its high accuracy and efficiency [20]. Furthermore, in a test between different versions of YOLO, it is shown that the latest version YOLOv8 is both faster, and more accurate compared to its predecessors[21].

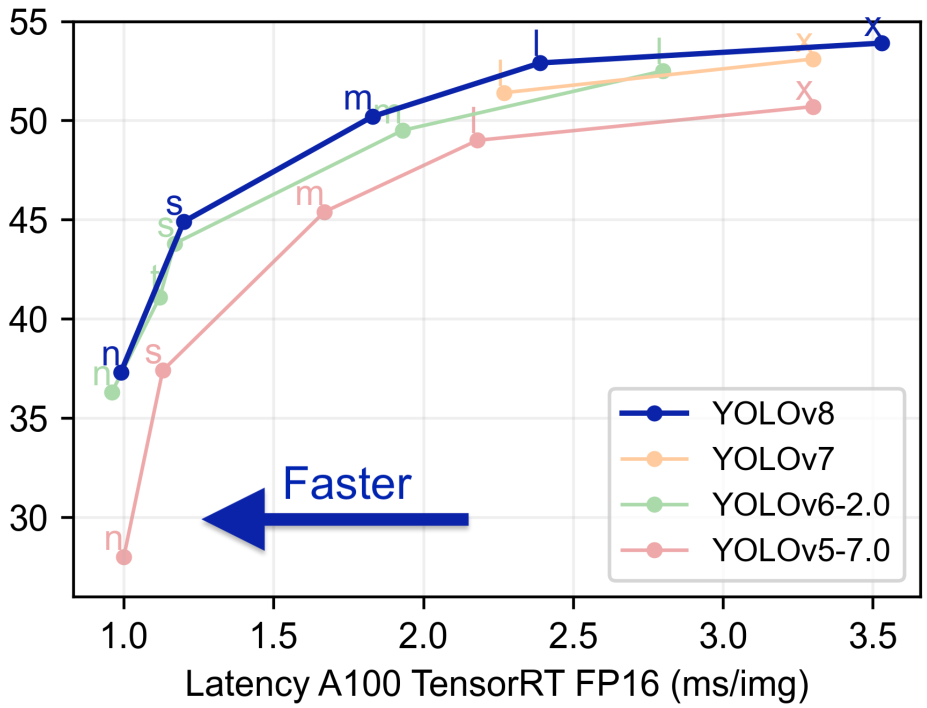


Figure 1.9: YOLO versions comparison[21]

Figure 1.9 shows the accuracy (y-axis) and latency (x-axis) of different variants of different YOLO versions.

Hybrid Object Detection Methods

Some researches explored the possibility to use both visual and sensor based sensing like [22] where a combination of visual detection in addition to tactile sensing in order to achieve better precision in determining the dimensions and position of the objects.

1.3.2.2 Segregation Methods

After Objects are identified, they are physically sorted depending on their material type. One of the popular techniques is using an air nozzle to push specific objects to their corresponding container.

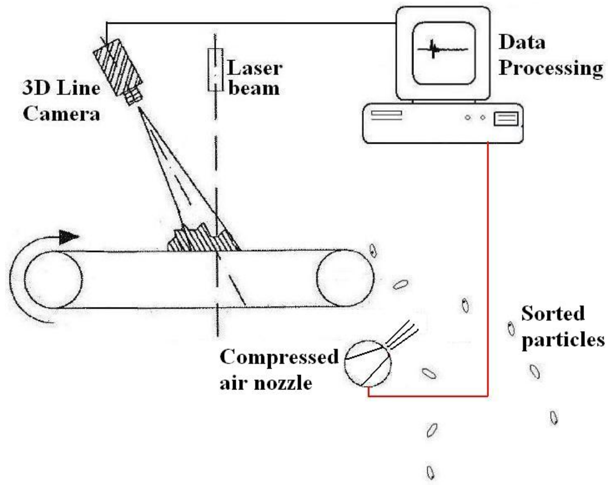


Figure 1.10: Air nozzle segregation[23]

In recent years, robotic systems have been developed to become a reliable option when it comes to waste sorting.

1.3.3 Robotized Waste Sorting Systems

Many works have implemented robotic systems for waste sorting, most of these projects used robotics arms, mobile robots or mobile manipulators.

In [24], a system which consists of a mobile robot equipped with a robotic arm was developed. It is supplied with sensors to detect biodegradable and non-biodegradable items to sort them in their respective bins. The system is shown in figure 1.11.

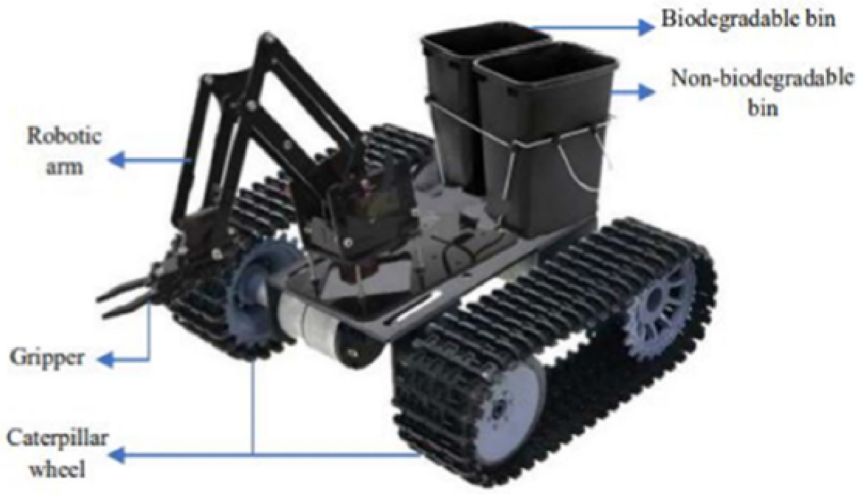


Figure 1.11: Mobile robot that sorts objects into biodegradable and non-biodegradable[24]

In an another work, a robotic arm was used to sort items moving on a conveyor belt using a set of sensors (IR sensor, Voltage sensor, Inductive sensor), with an Arduino UNO as the controller for decision making. The overall accuracy was 82%.

[25].

In [26], a robotic arm equipped with a vacuum gripper was used to pick and place waste objects moving on a conveyor after they are detected by a camera.

Bobulski et al [27] proposed using a mobile robot with caterpillar drive equipped with a conveyor belt to pick plastic items. This robot is tolerated and controlled via WIFI.

In [28], a robotic system for waste sorting was developed to pick and toss recyclable objects moving a conveyor belt. They used vision-based object detection, with the deep learning model Mask R-CNN. The gripper used was a vacuum gripper.

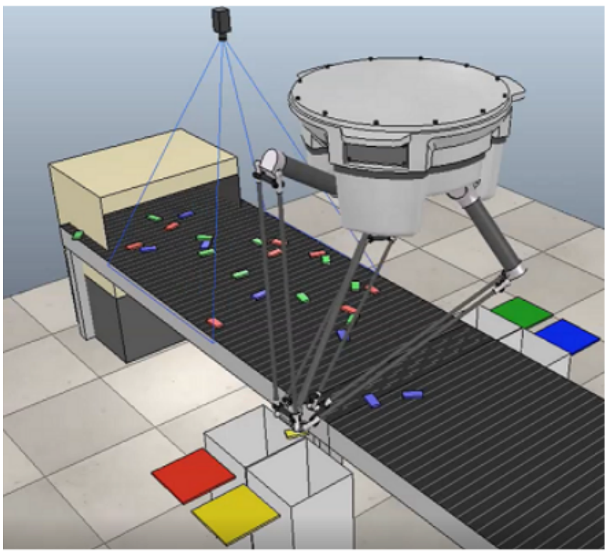


Figure 1.12: Simulation environment for the Pick-and-Toss system
[28]

Also, various robotic system were developed by companies specializing in waste sorting.

Chapter 2

Problem formulation and Proposed Solution

2.1 Introduction

In this Chapter we we state the problematic of our project and our proposed solution. We first describe the global architecture of the system, which contains multiple modules. We then explain how they work together and define each one individually.

2.2 Problem formulation

The conventional waste sorting systems involves calling on human operators in waste sorting facilities to do the tasks of picking objects (generally moving on a conveyor) and placing them accordingly Fig. 2.1. However, given that these objects might not be pre-processed, they may be dangerous to handle with bare hands. Sharp objects can cause wounds and injuries. Also, since waste objects are likely to carry bacteria or microbes, they can cause infections and disease. Human operators are facing many dangers . Some of these dangers are the risks of getting injured when manipulating dangerous items and exposition to toxic objects. In [33], narratives were collected from Brazilian people working in waste sorting facilities. They described the dangerousness of their working environment. Also, the quantity of generated domestic bio medical wastes increasing after the covid pandemy[34] in india; the human operators became confronted even more to bio medical wastes considered as toxic, harmful, infectious, and hazardous to humans.

The well being and the health preservation of human operators, in the context of waste sorting are major. The automation of the process is, thus, mandatory. The automated recycling systems require, object recognition module allowing the recognition of objects in order to take the right decision that depends on the material of the object (plastic, metal, glass...), and its dangerousness degree(broken glass, syringe, aerosol...).

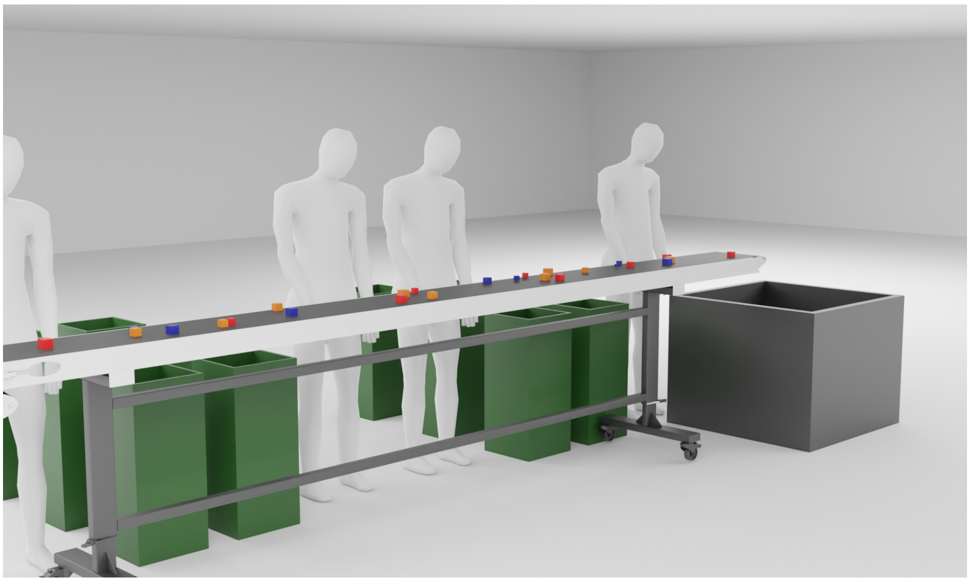


Figure 2.1: Waste sorting process before the automation

Passing from a waste sorting operated by humans to automated process necessitates certain requirements. We opted in our work to use a robot for performing the action of sorting wastes.

The main needs and constraints for designing robotized waste sorting system are described below:

- The prescribed task consists of picking a recognised object and place it in a right recycling bin.
- The object recognition process must be executed in real-time. The objects are supposed moving on a conveyor.
- The object recognition process must be executed with a relatively high accuracy. The objects can be dangerous to the human operator. Thus a high accuracy is required to reduce dangers on the human.
- The presence of the human should be detected in real-time. This allows to adjust the behavior of the robot rapidly.
- Collaborative robot is required to accomplish a prescribed task in an autonomous way. This requires ingenuity, dexterity and intelligence. The human can collaborate for certain tasks with the robot

We propose in the following section the proposed solution for automating waste sorting with a robot.

2.3 Proposed Solution

We propose in an automatic system that identifies the wastes using Artificial Intelligence, takes the right decision depending on the material of the object, and then uses a co-bot (collaborative robot) to place the detected waste items into the appropriate bins designated for their respective categories. Human detection module is integrated into the system to detect human presence, allowing it to decide if it should cooperate with the human operator or not. During cooperative sorting, waste is divided into human-handled (e.g., fragile, needing disassembly or upcycling items) and robot-handled (e.g., dangerous items).

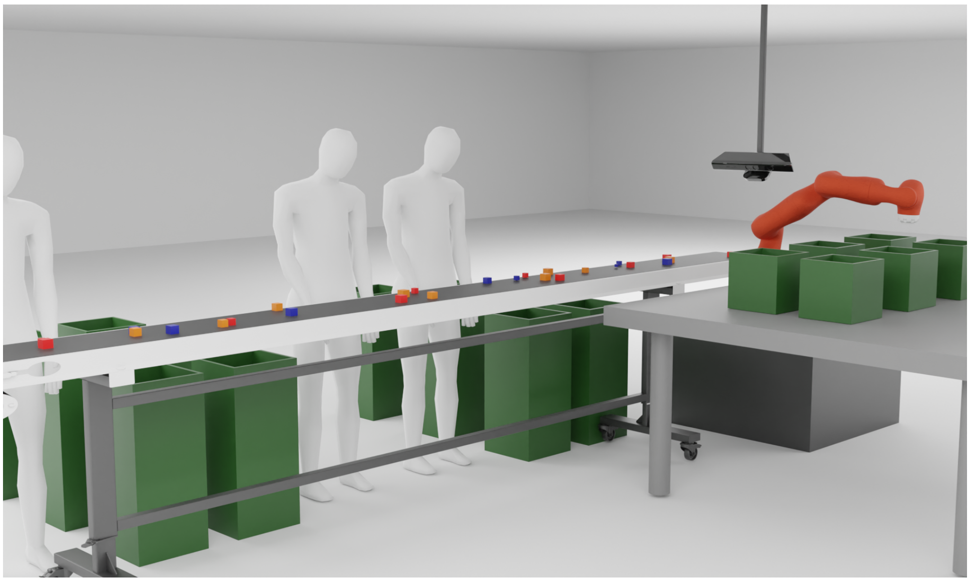


Figure 2.2: Waste sorting process after the automation

2.3.1 System Requirements

In Fig 2.3, we define the main modules of our system. The Object Detection and recognition module is used to detect objects on the workspace of the robotic arm, and get their corresponding positions and material type. This module can use a camera with an artificial intelligence model to detect the objects to be sorted.

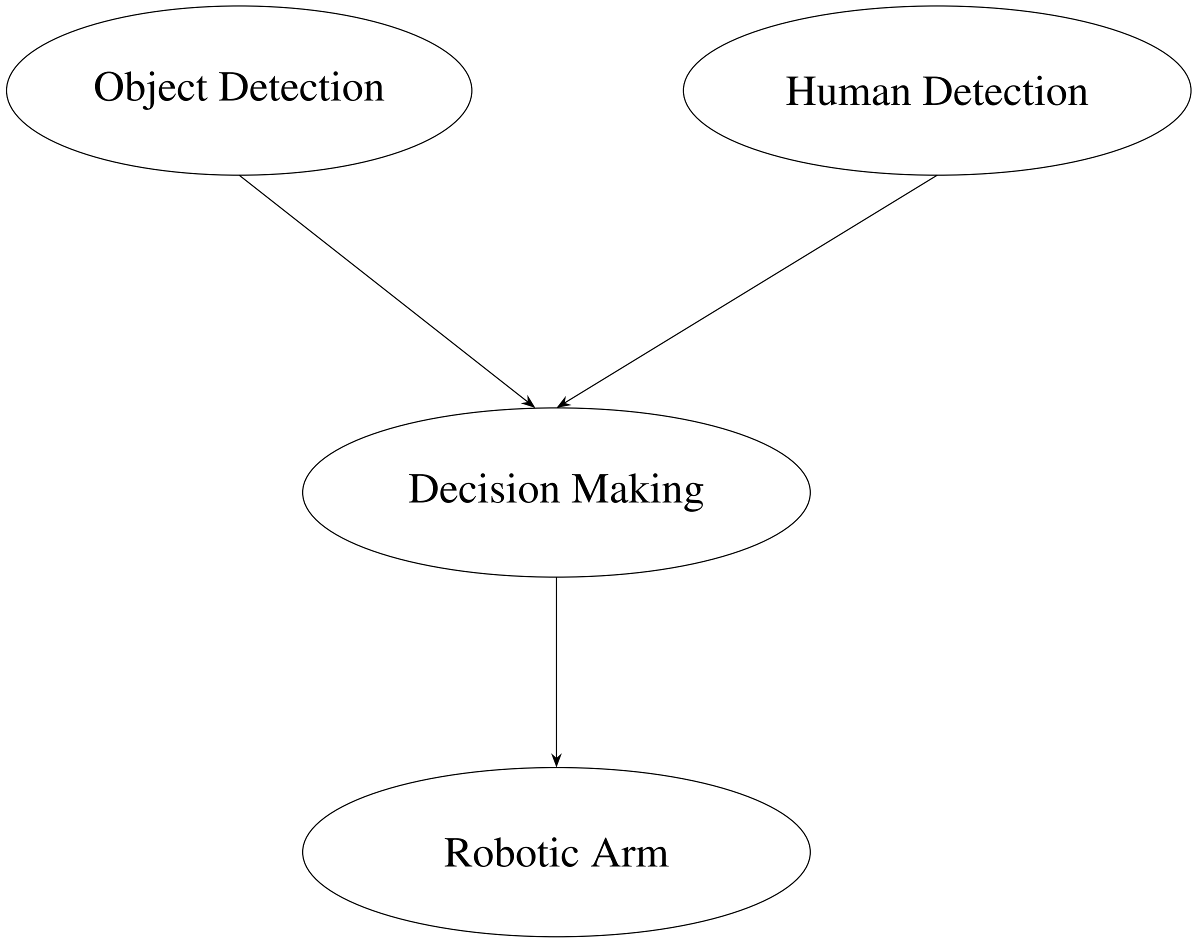


Figure 2.3: Global architecture of robotized waste sorting system

Human detection module can use different kinds of sensors. RFID sensors can be efficient for human presence detection in the vicinity of the cobot.

The decision making module is an important module allowing the analysis of information coming from different modules cited previously, and taking decisions about the robot's behavior. The robot controller module gathers information from the decision making module. it then moves the cobot accordingly.

2.3.2 Object Detection and Recognition

2.3.2.1 Definition

The process of localizing certain objects on an image, and identifying their classes is called Object detection and recognition. The algorithms used in object detection and recognition use Artificial intelligence, which includes both machine learning and deep learning. Thus, they require training a model on datasets of images before they can be used. Many algorithms have been developed, and each one performs differently in different scenarios. These algorithms are usually compared based on their accuracy, their speed and the size of the training dataset needed [35].

2.3.2.2 Main Types and Comparison

Object detection algorithms can be classified into two main categories: Machine learning based algorithms and Deep learning based algorithms. The main differences between them lies in:

- **Pattern recognition:** in deep learning patterns and features are detected automatically, while machine learning requires a manual features extraction step before training [16].
- **Needed dataset size:** Unlike deep leaning, machine learning does not require large amounts of samples to achieve good results [36].
- **Accuracy:** Generally, Deep learning algorithms proved to be more accurate and robust, as they are less affected by noises and variations in data [36].

Moreover, there exist different deep learning algorithms with different specifications. We can find two main categories: The first category includes fast models which support real-time detection, and the other one includes slower models that are more accurate. In [37], a comparison was done between YOLO model which represents the first category, and RCNN and Fast RCNN from the second category.

2.3.2.3 Choosing the AI Model

Based on the the comparative studies and the algorithms mentioned in the state of the art, we decided to use a deep learning model, specifically YOLOv8. It is also worth mentioning that YOLOv8 has five variants as shown in Fig 1.9. Which are n,s,m,l and x, with n being the fastest and x being the most accurate.

After experimenting we decided to choose YOLOv8x (the slowest but the most accurate), since we can work with a low frequency of detection. On the other hand, we would have a more accurate result of object detection.

2.3.2.4 YOLO

YOLO (You Only Look Once) is a deep learning model based on Convolutional neural networks, which can do object detection tasks. The YOLO model introduced a new algorithm balancing between accuracy and speed. This algorithm achieves higher speeds by doing the whole detection process in one neural network evaluation. Unlike other algorithms which features a two stage detection process. YOLO has some limitations when it comes to detecting multiple small adjacent objects. [38]

Our System will detect waste products, and determine their material type and position using a camera. For this we will use an object detection AI model.

2.3.2.5 Collecting Data and Training the Model

To train an ai model, datasets are needed. The dataset size must correspond to the complexity of the problem to be solved. In our case we are trying to make the model detect objects' types given a picture. So we need a dataset containing multiple images for each class of objects, so we used the Roboflow framework to collect the needed images. This framework offers many features including:

- Uploading and labeling images.
- Collecting images from other public datasets, and merging them in one dataset.
- Generating pre-processed and augmented datasets.
- Training a model using the user's dataset to check for its accuracy.
- Generating a download code that can be used in a Google Colab notebook to do a cloud-based training.

We aimed to collect images for 6 classes: Metal, electronic, glass, broken glass, plastic and syringe. We collected images from other open source datasets to form a total of 1864 image.

Figure 2.4: Samples from the collected dataset

To experiment with different parameters, we also collected our own dataset from images captured in the working site.

Figure 2.5: Personal dataset samples

Before exporting the dataset, we used the RoboFlow interface tools to add pre-processing and augmentation adjustments. We added resize and auto-orient pre-processings to make the images of the dataset uniform in size and orientation. Then we added rotations and croppings to the images as dataset augmentations. The data augmentation step creates more images from existing ones by applying modifications to them. This introduces variety and randomness to the dataset, which help making the model more robust and less affected by changes that may occur in real world scenarios.

With the dataset ready we started the training in a Google colab notebook. When we finish training the YOLO model, we can use it on a video stream to get results of detected objects each frame. The position of the recognised objects is then calculated and sent to the decision making module which makes the right decision about the behavior of the robot.

2.3.3 Robot Motion

After the Objects are detected along with their position and material type, The robotic arm should reach to the object to pick it and place it in its corresponding container. The robot should be collaborative, as we mentioned in the problem description section, since human operators can be required to share the workspace with the robot, or can be very close to it. The robot should intrinsically be predisposed to allow the presence of a human. Other sensors and devices should be used to detect the presence of humans and warn them if any injury can occur.

The work presented in this document is presented in simulation and experimented on real robot. the proposed architecture was implemented using the Kuka LBR iiwa 7 R800 cobot.

The Kuka LBR iiwa 7 R800 is part of the LBR iwaa series, first HRC (Human Robot Collaboration) compatible industrial robot series. The light weight robot series open new possibilities in the area of collaboration in work sites.

The figure 2.6 shows the robot used and the figure **??** shows additional characteristics concerning the robot.

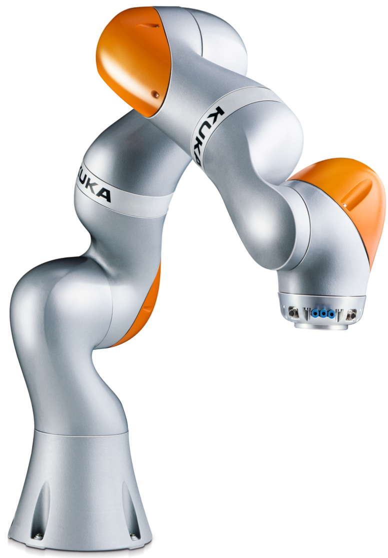


Figure 2.6: LBR iwaa 7 R800[39]

2.3.4 Decision Making Module

The module behavior could be described by the flowchart presented in figure 2.7.

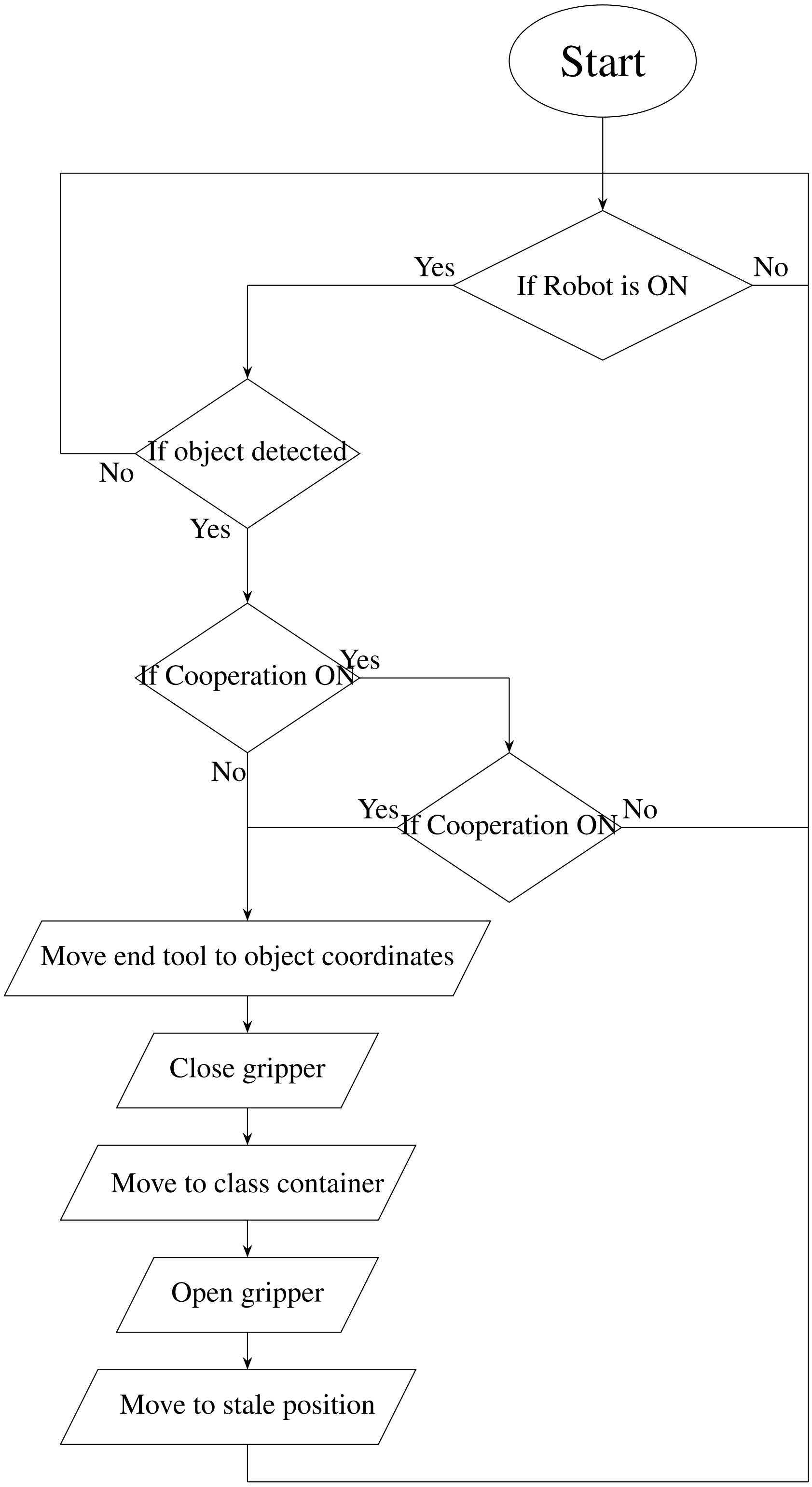


Figure 2.7: Program flow chart

2.3.5 Human Detection

Human presence should be detected. In this system we chose to use RFID technology to achieve this purpose.

2.3.5.1 Overview of RFID Technology

Radio frequency identification (RFID) is a wireless communication technology with a main feature of identifying objects by unique tags. As is seen in Figure 2.8 Three main components consists the RFID system[40].

- **The Tag:** It is a chip equipped with an antenna. It is glued or injected to the target object that needs to be identified. It has also a memory.
- **The Reader:** It is the device that detects the tag and collect all the information from it such as the content of its memory.

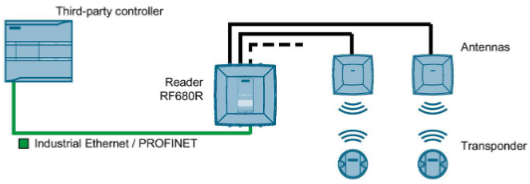


Figure 2.8: RFID architecture [40]

The key concept of the working principle of the RFID technology is that the reader emits electromagnetic signals which induces a current within the electronics circuity inside the tag. The tag then responds by communicating the information in its chip via a modulated signal (PM or AM). The reader receives and demodulates the signal to extract the data sent by the tag[40]. The RFID Tags might be classified into three categories according to their power source. Passive Tags receive power from the electromagnetic signal coming from the reader. However, Active Tags use a battery to power their circuitry. Active tags can achieve higher communication distances but are more expensive. The last class is the Semi Active RFID. This type of RFID systems have batteries to power the tags but also draw power from the electromagnetic signal[40]. The RFID operate in different ranges of frequencies, from Low Frequencies such as 125kHz and 134.2KHz to High Frequencies (13.56 MHz) and Ultra High Frequencies (433MHz, 2.45GHz). The choice of operation frequency depends on the trade-offs that each range have. For example, low frequencies work seamlessly on water and metal mediums but they are expensive and subjected to noise. Whereas, ultra high frequencies have cheaper tags and higher reading rates but they are vulnerable to water and metal[40]. The figure 2.9 shows the result of a study made in 2007 on 2066 project that incorporate use of RFID. This study yields that HF is the most common in RFID systems.

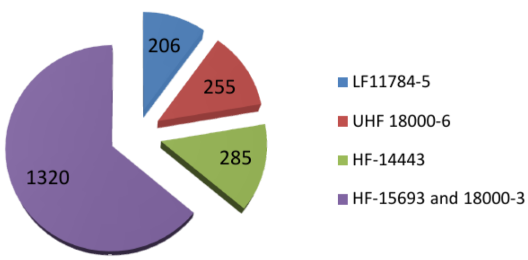


Figure 2.9: RFID frequency use [41]

2.3.5.2 Uses of RFID

The major field consuming RFID technologies is inventory management and logistics[40]. It is mainly used to track the items that are equipped with tags through the different facilities. It also has several other applications such as transportation, manufacturing, health care, security and access control, Contactless Payment, ID cards and Passport, RTLS and sensor network, military, animal farming, Libraries and archives.

In order to get the information about the presence of the human operator in the worksite, The SIMATIC RF600 RFID module is used. As stated earlier, each member of the staff is equipped with an RFID Tag with a special ID in order to give the right authorities for each worker. This setup opens possibilities for different applications, but for our case we consider only one scenario for demonstration purposes. There are 2 classes of workers, experts and operators. Both type of users have access to an interface that shows the following data:

- The real time camera stream with bounding boxes around the detected objects.
- The real time visualisation of the robot movement.

2.4 Conclusion

Throughout this Chapter, we presented the general architecture and some details about the modules. By the end of this chapter, we have a clearer view about the structure of our project, which prepares for the implementation stage in the next chapter

Chapter 3

Implementation, Results and Discussion

3.1 Introduction

We present in this section the implementation process and main results of our experiments. We start with describing the software and hardware tools used to realize our solution. Then, we present a general architecture of the system implemented under ROS Middleware. We describe mainly the communication process via topics. we present finally the simulated and real results.

3.2 Software Tools

3.2.1 Robot Operating System

The Robot Operating System (ROS) is an open source software ecosystem that provides tools to manage the codes of robotics projects properly. It allows for easier code maintenance and extensibility, easier code sharing and implementation of commonly used functionalities. At the same time, it gives the user flexibility since it supports many programming languages with keeping the possibility for low-level device control. In addition, ROS enables the programs to be executed in processes and supports message-passing between multiple processes.[42]

The processes of ROS are called nodes. systems built in ROS usually contain multiple nodes which communicate with each other using a specific protocol that allows passing messages through topics. Messages can be standard data types: integer, float, string, boolean ...[43] We used in our implementations ROS Noetic under linux ubuntu 20.04.

3.2.2 Python

Python is a powerful high level language created by Guido van Rossum. It is designed to be simple and very practical at the same time. It features many programming styles such as object-oriented programming making it very dynamic[44]. With python being the most used programming language in computer vision applications and the full support of python in ROS, it is the best option to maintain both efficiency and simplicity in the project. We used python to program nodes in ROS.

3.2.3 Google Colab

Colab is a free online tool provided by google that give access to computing resources such as TPUs and GPUs. The use of these resources is very helpful when performing heavy computational processes in the model training phase. We used Google Colab to train our AI model.

3.2.4 OpenCV

The OpenCV library is crucial for processing and handling images[45]. We used it in many occasions as:

- Opening the video stream from the camera and capturing images from the stream.
- Showing images of object detection.
- Preprocessing images and doing perspective corrections.

We used OpenCV for pre processing steps.

3.2.5 QT

QT is a library supported by many programming languages including Python and C++. It is used to create Graphical user interfaces. This Library was used to create the User Interface node

3.2.6 RVIZ

RVIZ is a 3D visualisation tool built in ROS. It allows graphical visualisation of the robots in real time. We used it alongside with MoveIt tool to simulate the system before working with the actual robot. In order to visualise the robot, a URDF file containing the robot model specifications is needed.

3.2.6.1 URDF Files

URDF (Unified Robot Description Format) files are XML specifications that model multi-body systems. This format is widely used in robotics for simulating the actual robots by importing the corresponding URDF files especially in ROS. URDF files contain the visual representation, collision representation and inertial properties of the different parts of the robot (links). It also contain information about the joints connecting the links.[46]

For this simulation, we downloaded an open source URDF file containing the visual representation of the robotic arm Kuka LBR iwaa r800 from the GitHub repository[47]. We then modified it to add a gripper to the arm, and simulation environment objects represented in: a surface where non-sorted elements are placed, and 6 boxes one for each object class.

3.2.6.2 MoveIt

MoveIt is an open source software framework used for motion planning, manipulation, inverse kinematics, control, 3D perception and collision checking for robotics applications. The setup assistant is a graphical tool provided by MoveIt to generate ROS packages[48]. In our application we loaded the URDF file downloaded earlier and went through the wizard steps:

- Defined the self collision constraints of the robot.
- Defined the planning groups.
- Defined robot poses for the gripper: close, open
- Defined end effectors.

After that, a ROS package was generated. It could be used to simulate the movement of the robot by directly controlling the Cartesian coordinates of its end effector.

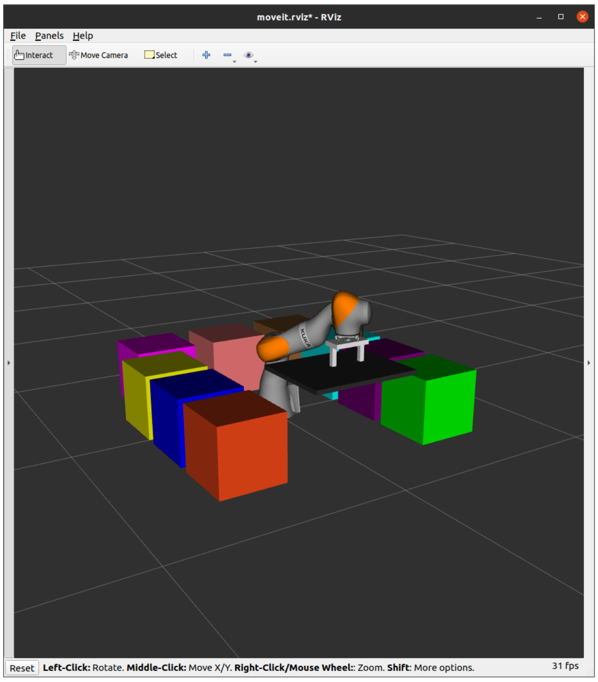


Figure 3.1: Simulation of our robot on Rviz window

3.3 Hardware Tools

In this section , we present the different hardware tools used in the implementation. All the algorithms were implemented on a host machine with the following specifications :

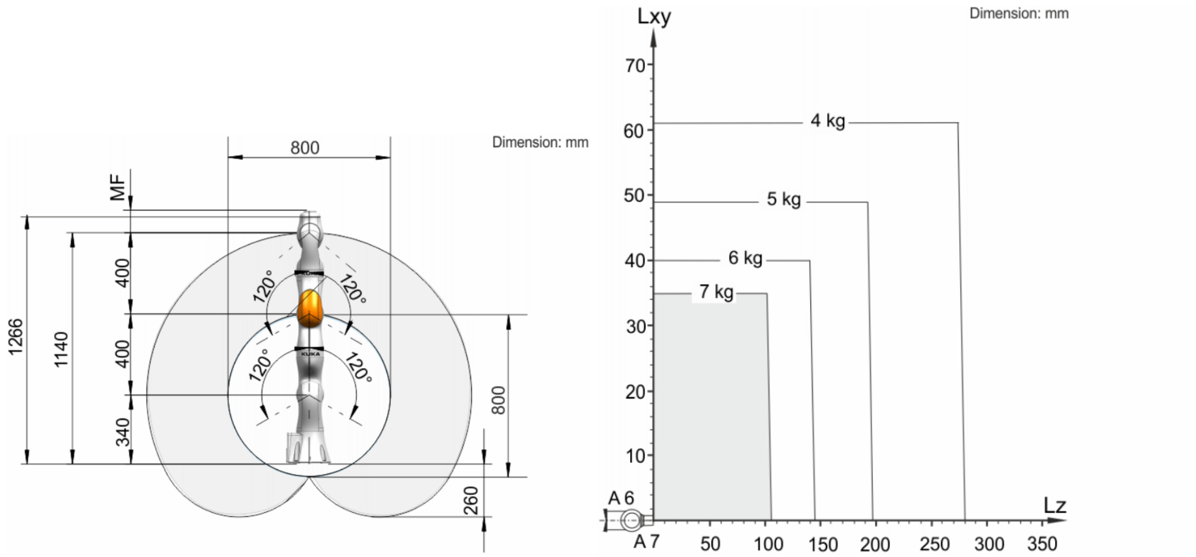
- Processor : Intel CORE i7 930
- Graphic card : NVIDIA GeForce GTX 560 Ti
- Ram : 8GB

3.3.1 Kuka LBR iwaa 7 R800

The following table showcases some characteristics of the robot.

Table 3.1: Specifications of the robotic arm [39]

Specification	**Value**
Number of axes	7
Rated payload	7 kg
Maximum Reach	800 mm
Repeatability	± 0.1 mm
Weight	23.9 kg
Protection rating	IP54



(a) Workspace (b) Payload Diagram

Figure 3.2: Workspace and payload diagram of the robot [39]

3.3.1.1 Interfacing the Robotic Arm

The arm is connected to the Kuka Sunrise Cabinet controller. It is a processing unit that runs a dedicated operating system called Sunrise OS . The arm is equipped with a smart pad that is used to manipulate the robot, execute applications and monitor all the robot information. The robot could be programmed in an external computer using the Sunrise Workbench software and upload the developed programs to the Sunrise Cabinet via Ethernet. The Workbench software supports java programs.

3.3.2 Gripper

A two-finger gripper is attached to the end of the robot. Due to the incompatibility of the robot controller with the used gripper, an extra PLC SIMATIC ET200SP is used for communication between them. A TCP/IP socket connection is established between the PLC and the Sunrise Cabinet. The PLC is programmed to close the gripper whenever it receives the message ”c” and opens it whenever it receives the message ”o”.

Figure 3.3: The used gripper

3.3.3 Kinect Camera

To capture images of the workspace for object detection, we used a Microsoft Kinect. It is equipped with an rgb camera with 640x480 resolution. It was connected to the host machine through usb, and it could be detected using the OpenCV library.

Figure 3.4: The used Kinect camera

3.3.4 RFID RF600

The SIMATIC RF600 is an RFID reader developed by Siemens. The reader is connected to the host machine via Ethernet, and connected to two RF650A antennas.

Figure 3.5: The used RFID system

Figure 3.5 shows the RF600 reader and the RF650A antenna used in our project

3.4 Implementation of the System in ROS

3.4.1 The System's Architecture in ROS

Using ROS, we implemented the diagram at Fig 2.3 using nodes and topic based communication.

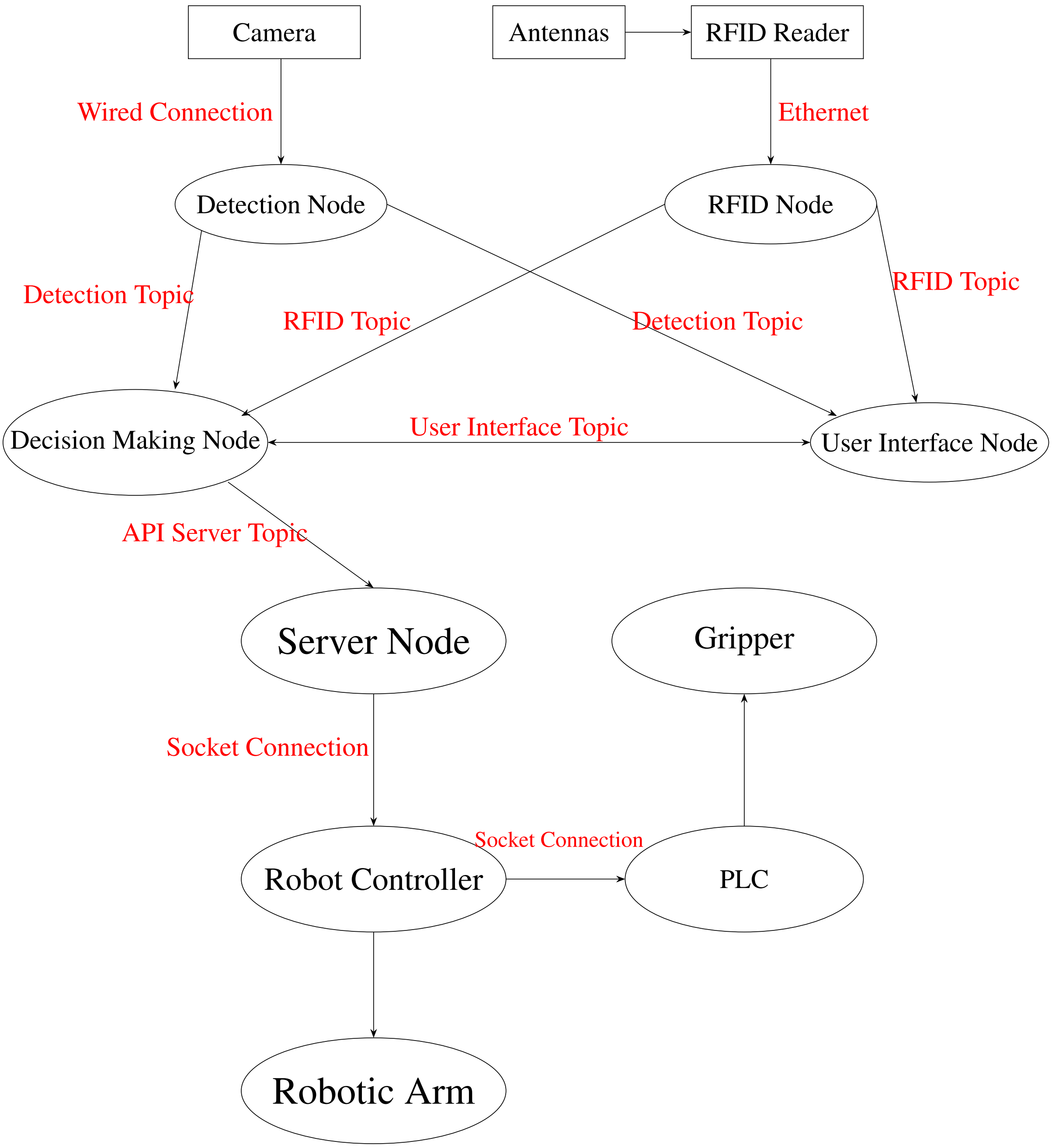


Figure 3.6: The Architecture of the system in ROS

3.4.2 Detection Node

In the detection node, we used the OpenCV and Ultralytics libraries. The program starts by opening the video stream from the camera, and loading the AI model. In a loop, an image is captured from the stream, perspective-corrected and then passed to the object detection function. The results of detection are represented by bounding boxes

3.4.2.1 Perspective Correction

Perspective correction is a pre-processing step, which makes a new image focusing only on the workspace rather than the whole picture. For this calibration process we used the function warpPerspective() from the OpenCV library

(a) Before (b) After

Figure 3.7: Before and after applying the calibration

This step was mainly introduced to fix the limitation of YOLO mentioned in Section 1.3.3 . By making objects occupying a bigger portion of the captured image.

3.4.2.2 Determination of Detected Object Position

After getting the results of detection, we calculate the central point of the bounding box, and apply a coordinates transformation to get the object's position in the XYZ coordinates with respect to the robotic arm. The results are then published in a topic we named ”detection results”.

3.4.3 RFID Node

A node responsible of the communication between the system and RFID reader was created. First the node is triggered by the decision making node through a rostopic. After that, the node keeps sending requests to the reader via TCP/IP and the reader responds with XML files containing all the information about the current readings. An example of the reader reply is presented in Appendix A. The node reads the content of the received files and use string manipulations to retrieve the important information from the replies such as the detected tag ID and the corresponding RSS. The RSS indicates the strenght of the signal received by the reader. In other words, the higher the RSS, The closer is the Tag to the reader. After collecting all the ID, attena name and RSS, the node concatinate these information into one string with the format: `(tagID+","+antennaName+","+RSS)` and broadcast this string to the RFID Topic. The work done in this section is inspired from previous works[49][50] done in the CDTA research center.

3.4.4 Decision Making Node

The decision making node is the brain of the system. It gets all the needed data from other components such as the detection, RFID and UI node, then process it to send commands to the server node that communicates with the robot.

This node is subscribed to the following topics: the object detection results topic and the rfid topic. When a message is published in one of them, a callback function is called in the subscribed node. The message to be sent is passed as an argument. The callback function for the object detection topic stores information about all objects in a list every time a detection is made, which can then be accessed by the main program for decision making. The RFID callback stores information about the human presence in a boolean variable. At the same time, the node is subscribed to the user interface topic and update the robot on and cooperation mode variables. The flowchart implemented in this node is presented in the figure 2.7.

3.4.5 Server Node

The server node is responsible of receiving the information from the robot and sending the commands to it. The server node creates a TCP/IP socket that the robot will connect to. In the robot controller, an API is installed to control the robot remotely from the ROS system. It is a modified version of the API created by the authors of this work [51], The work present a repository that includes a java file that should be installed in the robot controller and a ROS package containing the server node and a client python library. The API ensures making basic commands. Some of these commands are presented in the table.

Instruction	**Parameters**	**Description**
setPosition	A1 A2 A3 A4 A5 A6 A7	Move the arm according to the given joint angles
setPositionXYZABC	X Y Z A B C	Move the end-effector in a ptp motion
MoveXYZABC	X Y Z A B C	Move the end-effector in a linear motion
setCompliance	X Y Z A B C	Activate compliance mode with the defined stiffness
MoveCirc	X1 Y1 Z1 A1 B1 C1 X2 Y2 Z2 A2 B2 C2	Move the end-effector in a circular motion passing through the first point then the second

Table 3.2: API instructions [51]

In order to fit the use of our application, the following modifications on the API java file were performed.

- Added the gripper opening and closing API instruction.
- Added the status of gripper boolean variable and appended to the information string sent by the robot.
- Added a robot ready boolean variable (false when the robot is doing a movement)and appended to the information string sent by the robot.

- Changed the movement functions from blocking to unblocking to synchronise the arm movement with the gripper.

Once the socket connection with the robot controller is established, commands such as the ones presented in table 3.2 could be sent to the robot.

3.4.6 User Interface Node

The user interface is a visual representation of the system made using QT library. It collects information from the detection and RFID nodes, gets notifications about the robot's actions, and transfers user commands to the decision node.

The interface by default shows the following:

- The detection data and feedback about the robot movements (object detected, object reached, object sorted ...)
- The number of objects sorted in each class.
- The state of different options: robot state, cooperation mode, compliant mode and classes options

the user can be an administrator or an operator. Both of them can have access to all the information cited above but only the administrator can make changes by turning on and off the robot and the cooperation mode, selecting the classes to be handled by the robot in cooperation mode.

3.5 Results

In this part, we show the findings of our experiments and the results of our implementation. We can divide this section into 2 parts corresponding to the two stages of our implementation: The object detection and the simulation.

3.5.1 Object Detection

After training the model, A confusion matrix is generated. It describes exactly how precise the model is on detecting elements from the dataset. For the dataset we collected from Roboflow, we had the following confusion matrix

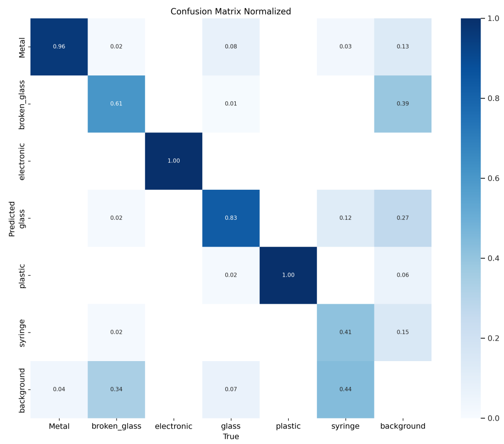


Figure 3.8: First confusion matrix

In this figure Fig. 3.8, the percentages shows how much images from a class in the columns are detected as a class from the rows. Of course each image from a certain class should be detected as its correct class, which will result in the highest percentages being represented in the diagonal. In our case, the model is not confusing between objects from certain classes, Although they are sometimes not detected at all in other classes. The final mean average precision (mAP50) of the model was 82.3%. However, some classes from this dataset were not uniform, and contained some noise factors which degraded the performance of the trained model. We could also notice that effect by looking at the "electronic" and "plastic" classes. They were both detected high accuracy, and both their datasets had images with a uniform background, and almost the same angle and distance from which images were taken.

To further test this, we collected a small dataset containing a few objects from the working site which had a uniform background and the camera was fixed. the results are shown in this confusion matrix

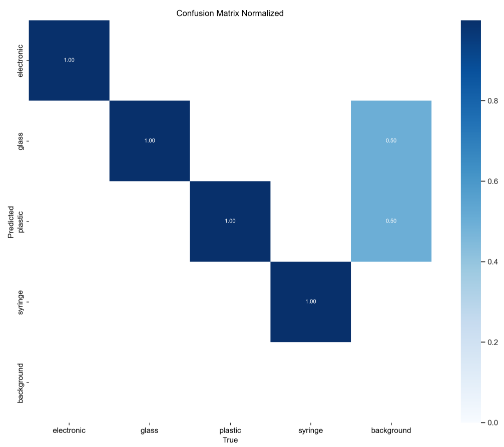


Figure 3.9: Second confusion matrix

Which shows that the accuracy of the model can be further improved if the whole dataset is collected from a recycling center.

We also tested the model with real images presented in Fig. 3.10.

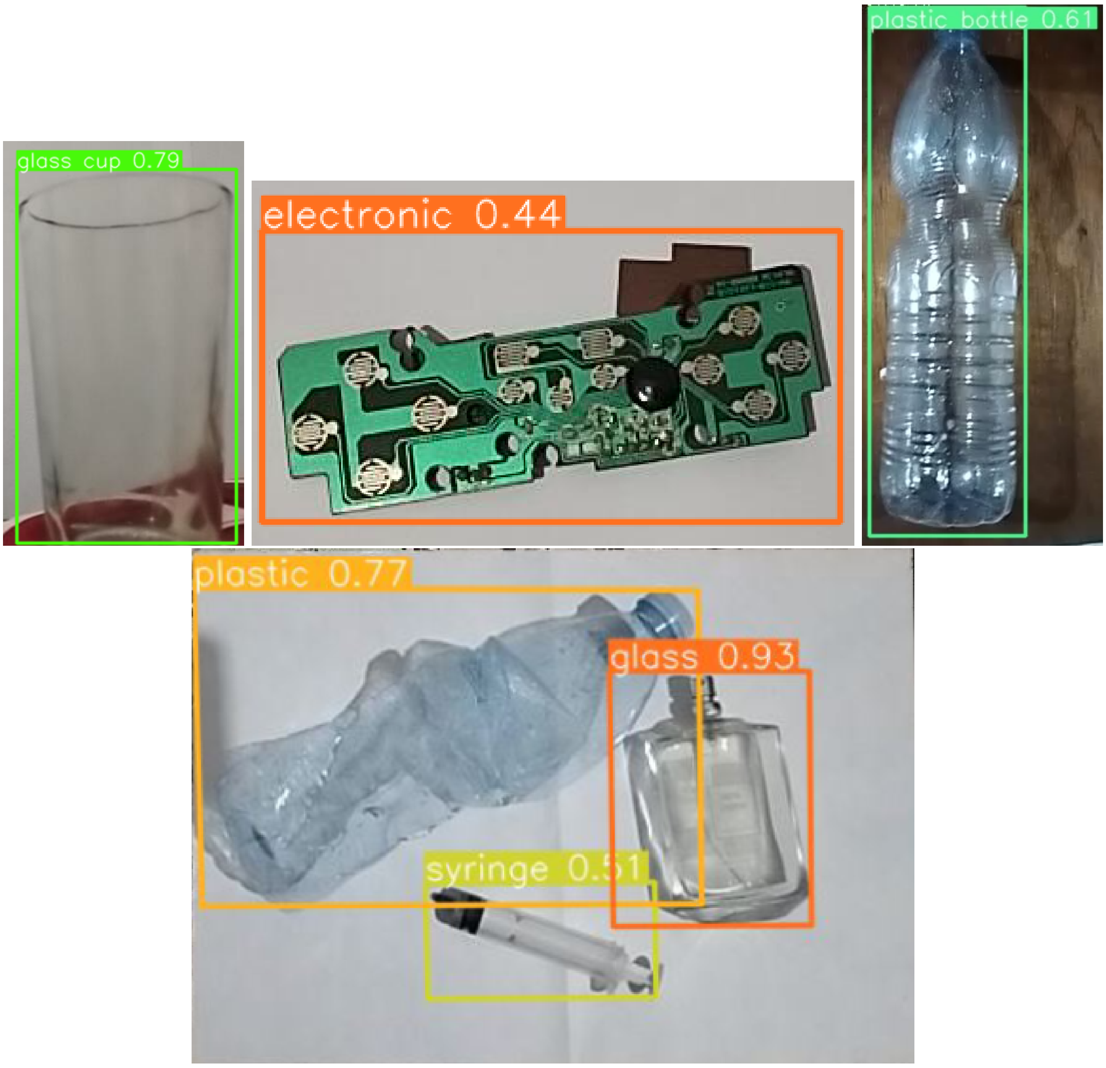


Figure 3.10: Examples of tests of objects recognition with YoloV8

The system contains multiple programs that run in different nodes in ROS. The Fig. 3.11 shows that the camera is placed on top of the table. The video stream is opened in the detection node and continuously being predicted by the trained YOLOv8 model. To test the working of each node, we checked the messages published in each topic by using the rostopic echo command. Once the model detects an object, we observe a message in the object detection topic containing both the coordinates of the object, and its class index.

3.5.2 Simulation

The simulation's purpose is to confirm the system is working concerning object detection, human detection and decision making nodes, and the communication between them.

The RFID node continuously checks for tags. So, whenever a tag is detected, the node sends a string that is a concatenation between the detected Tag ID and the Antenna name via a rostopic. The decision making node checks if the received ID corresponds to a human ID. If it does the "Human presence" state is set to True. When the "Human presence" state is True the robot will give priority to sort dangerous items like broken glass and syringes.

The detection node sends the type and position of detected objects each frame and the decision making node chooses what type of objects to handle.

We also used a command to listen to the rostopics that were published on, and both nodes worked as expected.

The decision making node then captures messages from both topics. the first step is to move the arm to the coordinates of the detected object. This is done by sending the corresponding coordinates to the moveit interface. Which then generates a path for the robotic arm to move along. By observing the simulation, the arm first moves to the object, closes its grip, then, it moves to the position of the box corresponding to the object type. It then opens its grip to simulate placing the object in its box and returns to the initial position.

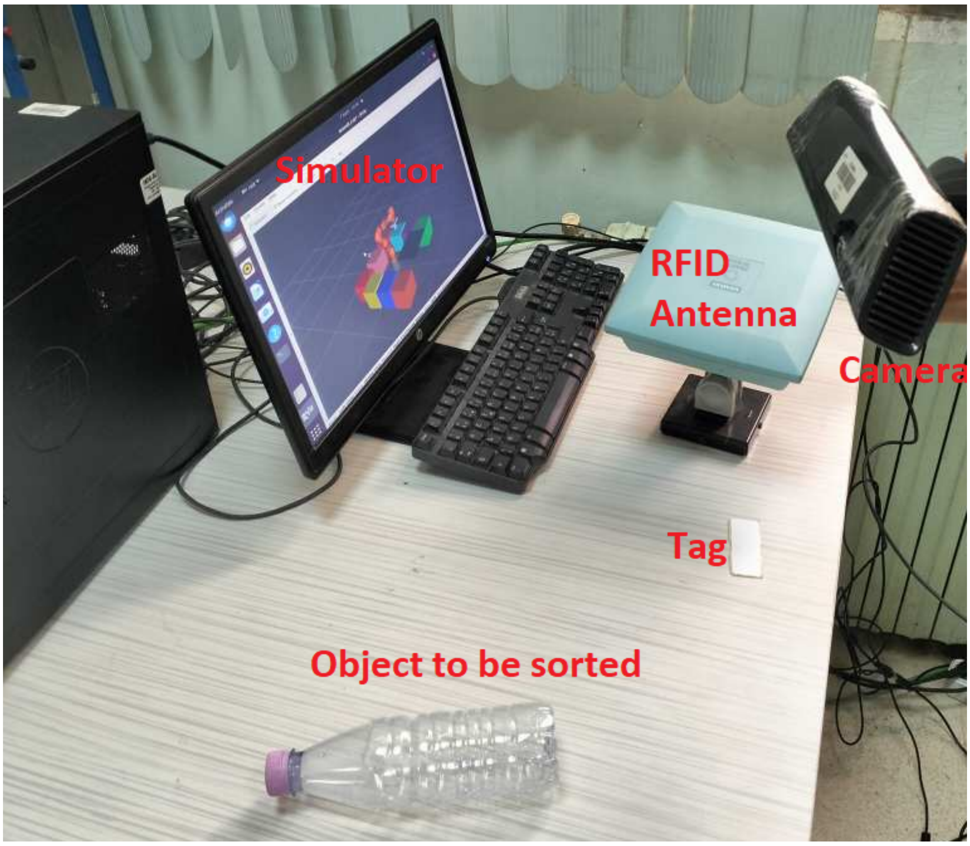


Figure 3.11: Testing setup

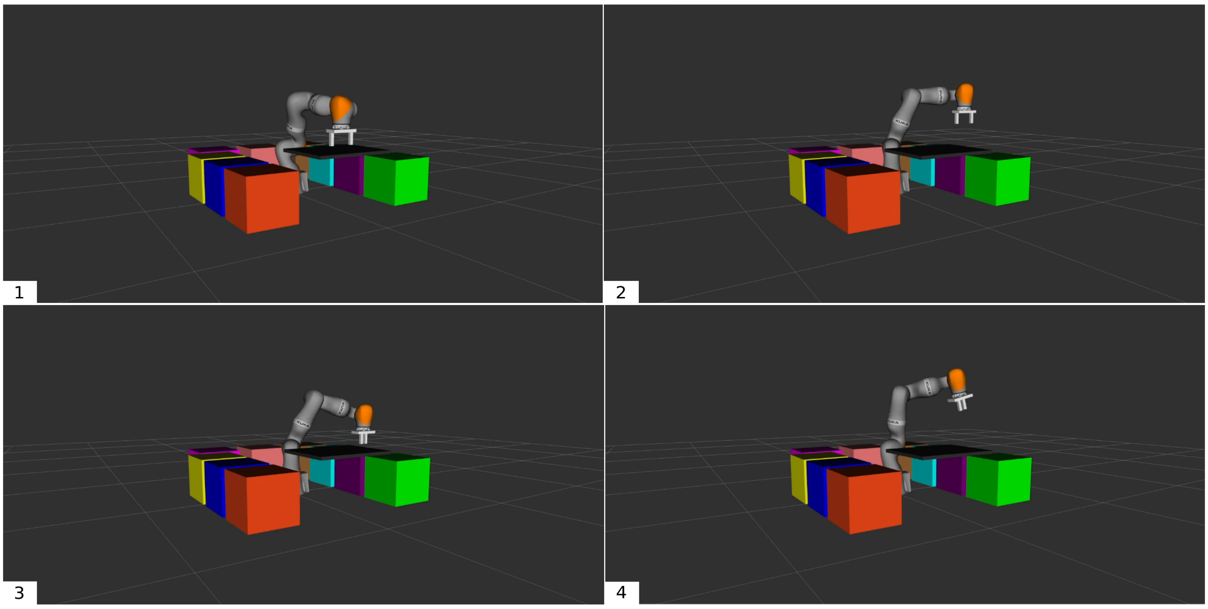


Figure 3.12: Robot in movement

Fig. 3.12 shows the simulated robot while moving an object to the corresponding box.

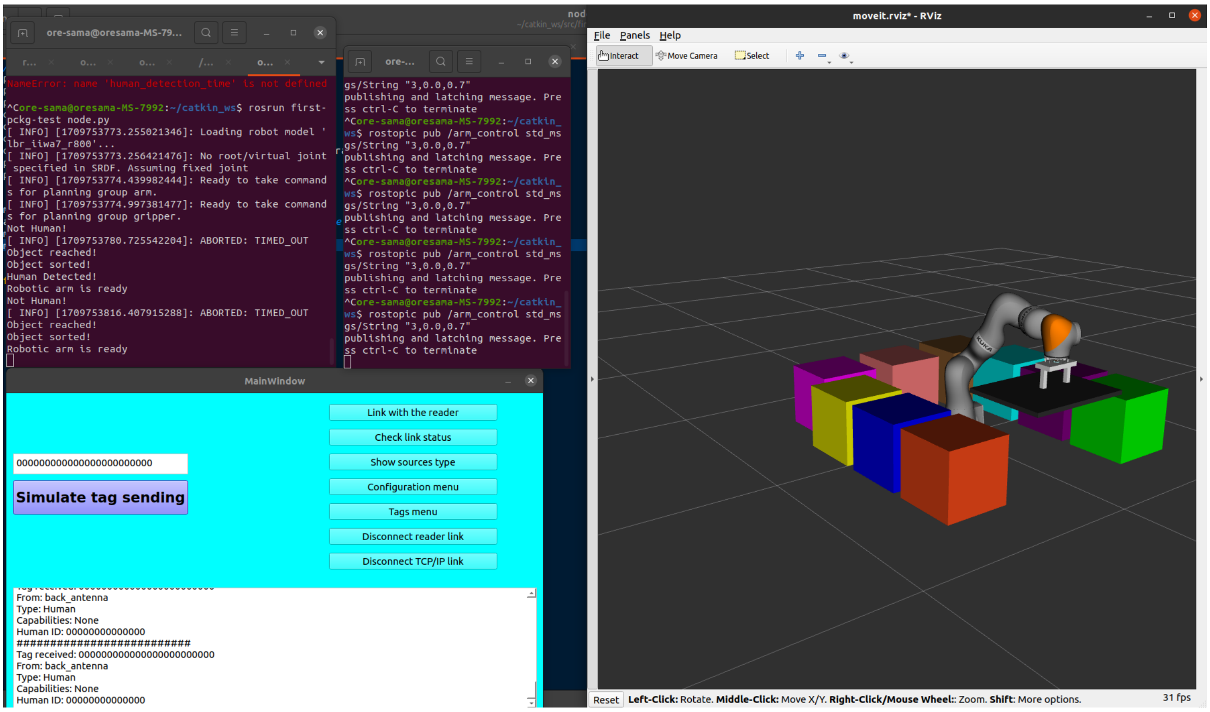


Figure 3.13: RFID detection test

As is seen in Fig. 3.13. A tag corresponding to human is put near the RFID antenna. The system detected the presence of the human as it is shown in the output of the control node in Fig. 3.14. This shows the system in action during the multiple scenarios it may face in operation.

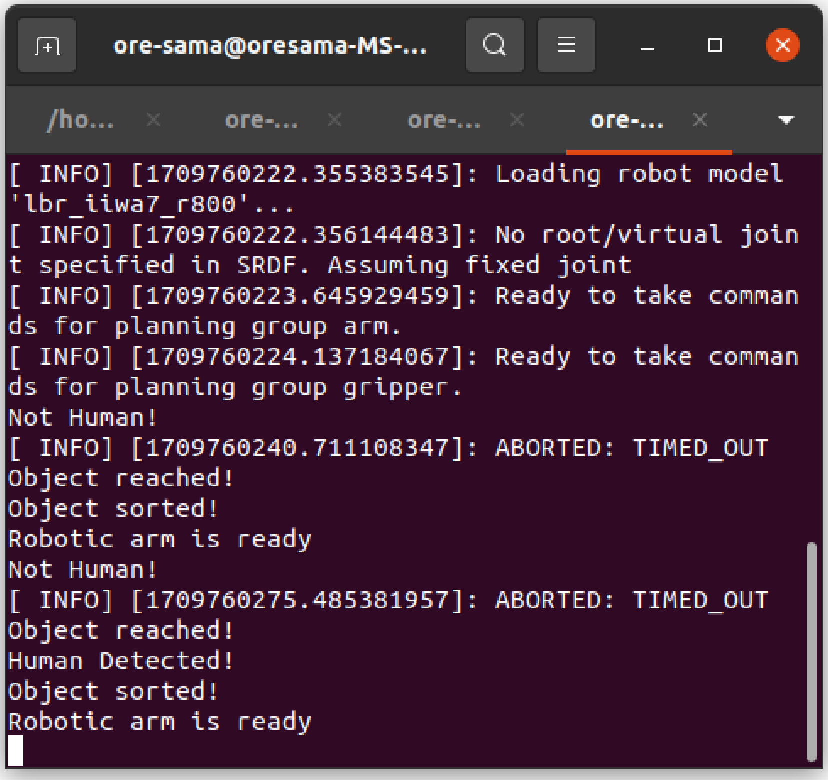


Figure 3.14: Terminal in RFID detection test

3.5.3 User Interface

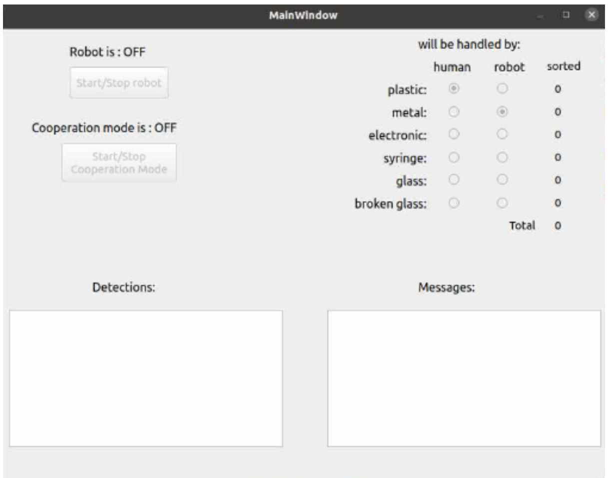


Figure 3.15: User interface

Fig 3.15, shows the final form of the user interface. After testing, the RFID authentication was working. All the states were being updated in the detection node, and the messages from the decision making node and detection node were showing.

3.5.4 The Setup with the Real Robot

After trying to simulate the system using the moveIt toolbox. We implemented the system described in the previous section in the Advanced Technology Development Centre (CDTA), in the Robotics and Industrial Automation Division, Robotized Production Systems “SRP” unit. The setup is shown in Fig3.16.

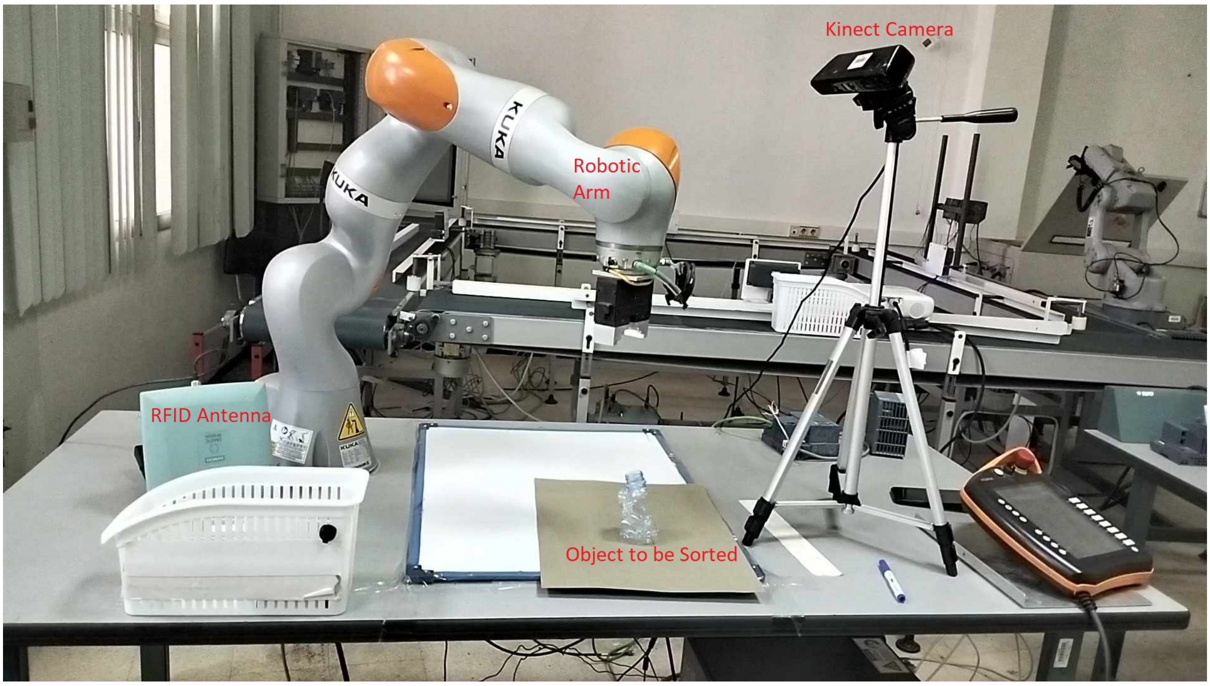


Figure 3.16: Waste sorting station

When the whole system was run, the camera detected objects on the workspace. Then, their positions were calculated with respect to the axis of the robotic arm. An instruction containing the coordinates of the detection was then sent to the robotic arm, which moved such that the end effector made contact with the object. However, the picking of objects was not very successful due to incompatibility of the gripper. To test the system’s performance without worrying about the gripper’s limitation, we decided to use legos instead of other objects because they were compatible with it. After training a model to detect legos, we did the operation again, and it was successful.

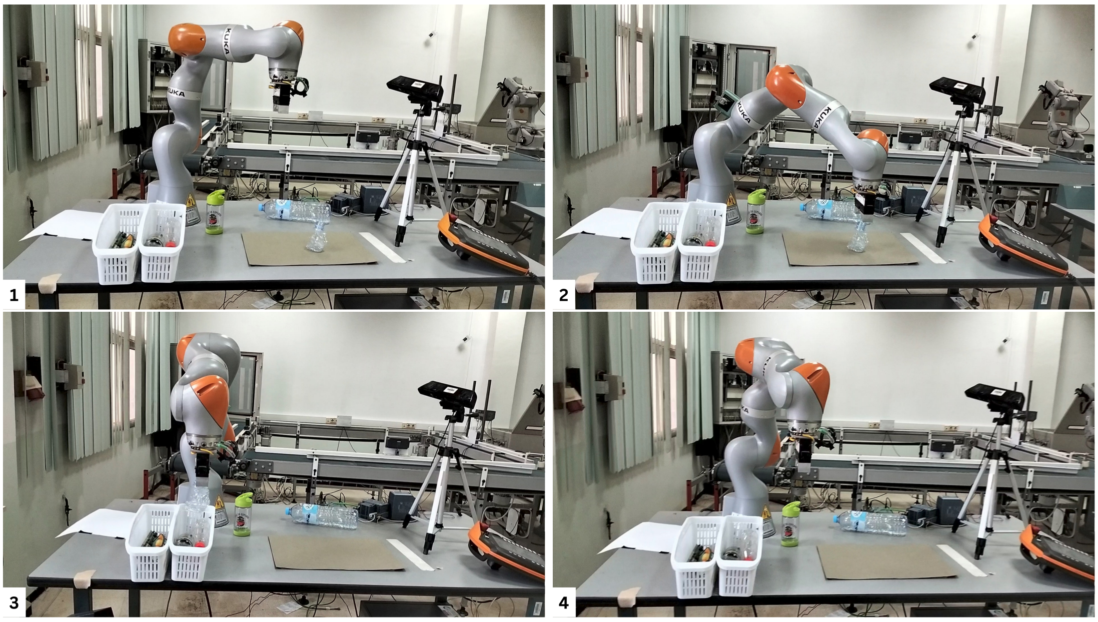


Figure 3.17: Frames of the sorting task

3.6 Advantages

The system main advantages are:

- Flexibility of waste types: If the user of the system needs different object classes, it is easier to reemployment the system, after training a new model based on a dataset adapter to the user needs.
- Use of other Robotic Arms: In order to use the system with another arm, it is sufficient to develop a new API for the corresponding robot keeping all the other components of the system.

3.7 Limitations

The main limitations of the system are:

- The cost of industrial robotic arms is higher than other waste sorting solutions.
- The speed of sorting might be a slow and cant keep up with the flow of wastes if the recycling line was loaded with a big quantities of wastes.
- The resolution of the camera was low, which limited the object detection process.
- The gripper we were using was not compatible with the objects we were handling.

3.8 Difficulties

Many difficulties were faced during the development of this project. Some of which are related the complexity of the network (many devices are connected to the same network and potential errors if any of the device is suddenly disconnected). Another main issue was due to the gripper equipped in the robotic arm not being suitable for this kind of application. The equipped gripper does not grab efficiently items having non-regular shapes. We noticed that it is efficient when manipulating rectangular objects such as Lego bricks, So we trained another detection model for Lego bricks and used to validate the system regardless of the gripper limitation and the system worked as expected with that type of objects. Another major difficulty was related to the development under the ROS middleware, especially the incompatibility of some ROS packages developed under ROS Melodic and Python 2 while we were working with ROS Noetic and Python 3. One more issue was the problem faced with the URDF file not working properly with some machines.

3.9 Conclusion

In this last chapter we finished our project by implementing the system and testing its different components. We then discussed the results and mentioned the difficulties we faced and our future aspects.

General Conclusion

The purpose of this work was to design and implement an automated waste sorting system based on a collaborative robotic arm with the ability to interact and cooperate with the human operator. The training of an Artificial Intelligence model was needed in order to identify waste objects classes and position to perform the sorting accurately. In addition, A software system was needed to handle the information and command the robot.

The report tackles generalities about robotics and the literature concerning actual waste sorting systems. After that, it describes the problematic and the proposed solution and finaly a description of the implemented solution and the findings.

For the object detection part several experiments were conducted using different datasets. Different accuracies were found when varying the classes in the datasets, number of images and the model architecture use. One of the key findings was that using images with a uniform background and preferably from the same workspace that the system would be deployed at will result in a much better precision in the detection.

Finally, a successful robotized waste sorting system with human-robot collaboration features was implemented through this project. With minor modifications, mentioned above, it would be ready for implementation in real life scenarios.

There is much room for improvements in this project. One of the main issues to tackle in further works is finding a way to better align the gripper with the objects to ensure good handling. The best option would be to change the gripper type to suction gripper because suction gripper does not require for the position to be very precise. If, due to some constraints, a finger gripper must be used then it would be preferable to improve the detection system into a more precise one. Some solutions would be to put the camera vertically and as high as possible, if the camera can not be put in a vertical position there is an option to use 2 cameras covering different angles. Another thing that could be improved is the detection model, there is a possibility to train a model with a larger variety of waste classes. This could help in creating a more universal solution if the model is able to detect every type of waste in different worksites. We are also planning to add object tracking algorithms to make sure the system keeps track of the objects position in real time.

At the end, we mention that we have submitted a conference paper entitled ”Waste Sorting Automation with Visual Sensor and RFID”. This paper covers the detection and simulation part of this work. It was accepted in ”International Conference on Advanced Electrical Engineering 2024”.

Appendix A

RFID Read Tag ID reply

```
<frame>
    <reply>
        <id>value_id</id>
        <resultCode> 0 </resultCode>
        <readTagIDs>
            <returnValue>
                <tag>
                    <tagID> value_tagID </tagID>
                    <tagPC> value_tagPC </tagPC> // opt
                    <utcTime> value_utcTime </utcTime> // opt
                    <antennaName> value_antennaName </antennaName> // opt
                    <rSSI> value_rSSI </rSSI> // opt
                    <channel> value_channel </channel> // opt
                    <power> value_power </power> // opt
                    <polarization> value_polarization </polarization> //
                    <inventoried> value_inventoried </inventoried> // opt
                    <filterDataAvailable> value_filterDataAvailable
                    </filterDataAvailable> // opt
                </tag>
                ...
                <tag> // opt
                     ...
                </tag> // opt
            </returnValue>
        </readTagIDs>
    </reply>
</frame>
```

Appendix B

Kinect Camera's Specifications

The kinect camera we used has the following specifications :

Table B.1: Kinect Specification [52]

Specification	**Details**
Model	Kinect I
Camera resolution (H x V)	640 x 480
Depth resolution (H x V)	320 x 240
Frame Rate	30 fps
Maximum depth range	6 m
Minimum depth range	40 cm

Appendix C

RFID Specifications

The RFID Reader used has the following specifications :

Table C.1: RF600 Specification

Specification	**Details**
Model	RF600
Read Range	Up to 8 meters
Frequency range (H x V)	865-868 MHz (Europe)
Data transfer rate	Max. 300 kbps
Power Supply	24VDC
Communication interface	Ethernet

Bibliography

[1] M. Ghallab and F. Ingrand, *Robotics and Artificial Intelligence*, pp. 389–436. Cham: Springer International Publishing, 2020.

[2] H. H. Poole, *Types of Robots*, pp. 27–51. Dordrecht: Springer Netherlands, 1989.

[3] 14:00-17:00, "ISO 8373:2021."

[4] "SCARA Robots | Explore by Product Series | Epson US." `https://epson.com/scara-robots-product-family`. [Accessed 11-06-2024].

[5] T. B. |. marchtwenty.com, "Cartesian, Rectilinear & Gantry Robots." `https://www.mwes.com/types-of-industrial-robots/cartesian-gantry-rectilinear-robots/`. [Accessed 11-06-2024].

[6] A. Dum and M. Savatekar, "Design of control system for articulated robot using leap motion sensor," *International research journal of engineering and technology*, vol. 03, pp. 1407–1417, 03 2016.

[7] Z. Samadikhoshkho, K. Zareinia, and F. Janabi-Sharifi, "A brief review on robotic grippers classifications," in *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, pp. 1–4, IEEE, 2019.

[8] E. Papadakis, F. Raptopoulos, M. Koskinopoulou, and M. Maniadakis, "On the use of vacuum technology for applied robotic systems," in *2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE)*, pp. 73–77, IEEE, 2020.

[9] F. Sherwani, M. M. Asad, and B. Ibrahim, "Collaborative robots and industrial revolution 4.0 (ir 4.0)," in *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*, pp. 1–5, 2020.

[10] "UR16e Heavy-duty, compact cobot."

[11] S. P. Gundupalli, S. Hait, and A. Thakur, "A review on automated sorting of source-separated municipal solid waste for recycling," *Waste management*, vol. 60, pp. 56–74, 2017.

[12] A. G. Satav, S. Kubade, C. Amrutkar, G. Arya, and A. Pawar, "A state-of-the-art review on robotics in waste sorting: scope and challenges," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 17, pp. 2789–2806, Dec. 2023.

[13] V. Lahtela and T. Kärki, "Mechanical sorting processing of waste material before composite manufacturing-a review.," *Journal of Engineering Science & Technology Review*, vol. 11, no. 6, 2018.

[14] Z. Xu, M. Chen, and C. Liu, "Object tactile character recognition model based on attention mechanism lstm," in *2020 Chinese Automation Congress (CAC)*, pp. 7095–7100, IEEE, 2020.

[15] M. O. Rahman, A. Hussain, E. Scavino, H. Basri, and M. Hannan, "Intelligent computer vision system for segregating recyclable waste papers," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10398–10407, 2011.

[16] G. E. Sakr, M. Mokbel, A. Darwich, M. N. Khneisser, and A. Hadi, "Comparing deep learning and support vector machines for autonomous waste sorting," in *2016 IEEE international multidisciplinary conference on engineering technology (IMCET)*, pp. 207–212, IEEE, 2016.

[17] S. Kumar, D. Yadav, H. Gupta, O. P. Verma, I. A. Ansari, and C. W. Ahn, "A novel yolov3 algorithm-based deep learning approach for waste segregation: Towards smart waste management," *Electronics*, vol. 10, no. 1, p. 14, 2020.

[18] M. R. Jabed and M. Shamsuzzaman, "Yolobin: non-decomposable garbage identification and classification based on yolov7," *Journal of Computer and Communications*, vol. 10, no. 10, pp. 104–121, 2022.

[19] K. Du and A. Bobkov, "An overview of object detection and tracking algorithms," *Engineering Proceedings*, vol. 33, no. 1, 2023.

[20] J.-a. Kim, J.-Y. Sung, and S.-h. Park, "Comparison of faster-rcnn, yolo, and ssd for real-time vehicle type recognition," in *2020 IEEE international conference on consumer electronics-Asia (ICCE-Asia)*, pp. 1–4, IEEE, 2020.

[21] "Ultralytics YOLOv8."

[22] D. Guo, H. Liu, B. Fang, F. Sun, and W. Yang, "Visual Affordance Guided Tactile Material Recognition for Waste Recycling," *IEEE Transactions on Automation Science and Engineering*, vol. 19, pp. 2656–2664, Oct. 2022. Conference Name: IEEE Transactions on Automation Science and Engineering.

[23] J. Huang, T. Pretz, and Z. Bian, "Intelligent solid waste processing using optical sensor based sorting technology," in *2010 3rd international congress on image and signal processing*, vol. 4, pp. 1657–1661, IEEE, 2010.

[24] S. Mondal, S. Das, S. Banerjee, and K. Pal, "A smart automated garbage management system to replace human labour," in *2022 6th International Conference on Devices, Circuits and Systems (ICDCS)*, pp. 1–5, IEEE, 2022.

[25] S. Z. Diya, R. A. Proma, M. N. Islam, T. T. Anannya, A. Al Mamun, R. Arefeen, S. Al Mamun, I. I. Rahman, and M. F. Rabbi, "Developing an intelligent waste sorting system with robotic arm: A step towards green environment," in *2018 International Conference on Innovation in Engineering and Technology (ICIET)*, pp. 1–6, IEEE, 2018.

[26] M. Koskinopoulou, F. Raptopoulos, G. Papadopoulos, N. Mavrakis, and M. Maniadakis, "Robotic waste sorting technology: Toward a vision-based categorization system for the industrial robotic separation of recyclable waste," *IEEE Robotics And Automation Magazine*, vol. 28, no. 2, pp. 50–60, 2021.

[27] J. Bobulski and M. Kubanek, "Vehicle for plastic garbage gathering," in *2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, pp. 1–5, IEEE, 2021.

[28] F. Raptopoulos, M. Koskinopoulou, and M. Maniadakis, "Robotic pick-and-toss facilitates urban waste sorting," in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, pp. 1149–1154, IEEE, 2020.

[29] R. Sarc, A. Curtis, L. Kandlbauer, K. Khodier, K. E. Lorber, and R. Pomberger, "Digitalisation and intelligent robotics in value chain of circular economy oriented waste management–a review," *Waste Management*, vol. 95, pp. 476–492, 2019.

[30] S. Ramadurai and H. Jeong, "Effect of human involvement on work performance and fluency in human-robot collaboration for recycling," in *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 1007–1011, 2022.

[31] K. Mahadevan, M. Sousa, A. Tang, and T. Grossman, ""grip-that-there": An investigation of explicit and implicit task allocation techniques for human-robot collaboration," *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, May 2021.

[32] R. Müller, M. Vette, and A. Geenen, "Skill-based dynamic task allocation in human-robot-cooperation with the example of welding application," *Procedia Manufacturing*, vol. 11, pp. 13–21, 2017.

[33] A. M. M. Moreira, W. M. R. Günther, and C. E. G. Siqueira, "Workers' perception of hazards on recycling sorting facilities in são paulo, brazil," *Ciencia & saude coletiva*, vol. 24, pp. 771–780, 2019.

[34] S. S. C S, S. C S, S. S, and U. S, "Automatic bio medical waste segregator," in *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, pp. 1124–1130, 2022.

[35] P. Malhotra and E. Garg, "Object detection techniques: A comparison," in *2020 7th International Conference on Smart Structures and Systems (ICSSS)*, pp. 1–4, 2020.

[36] J. Wang, S. Jiang, W. Song, and Y. Yang, "A comparative study of small object detection algorithms," in *2019 Chinese control conference (CCC)*, pp. 8507–8512, IEEE, 2019.

[37] P. Malhotra and E. Garg, "Object detection techniques: a comparison," in *2020 7th international conference on smart structures and systems (ICSSS)*, pp. 1–4, IEEE, 2020.

[38] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[39] "LBR iiwa — KUKA AG — kuka.com." `https://www.kuka.com/en-de/products/robot-systems/industrial-robots/lbr-iiwa`. [Accessed 25-05-2024].

[40] B. Fennani, H. Hamam, and A. O. Dahmane, "Rfid overview," in *ICM 2011 Proceeding*, pp. 1–5, 2011.

[41] R. Das and P. Harrop, "Rfid forecasts, players and opportunities 2009–2019," *IDTechEx report*, 2009.

[42] D. St-Onge and D. Herath, *The Robot Operating System (ROS1 &2): Programming Paradigms and Deployment*, pp. 105–126. Singapore: Springer Nature Singapore, 2022.

[43] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng, *et al.*, "Ros: an open-source robot operating system," in *ICRA workshop on open source software*, vol. 3, p. 5, Kobe, Japan, 2009.

[44] A. Sharma, F. Khan, D. Sharma, S. Gupta, and F. Student, "Python: the programming language of future," *Int. J. Innovative Res. Technol*, vol. 6, no. 2, pp. 115–118, 2020.

[45] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[46] "URDF Primer - MATLAB & Simulink."

[47] Rot_Tianers, "bhtxy0525/lbr_iiwa7_r800-urdf-package," Jan. 2024. original-date: 2020-03-16T03:17:56Z.

[48] D. Coleman, I. Sucan, S. Chitta, and N. Correll, "Reducing the barrier to entry of complex robotic software: a moveit! case study," *arXiv preprint arXiv:1404.3785*, 2014.

[49] A. Isma, A. Kamel, and S. Abderrahmane, "Hardware in the loop simulation for robot navigation with rfid," in *2022 7th International Conference on Image and Signal Processing and their Applications (ISPA)*, pp. 1–6, 2022.

[50] Z. Abdelmalek and C. Mohamed Rais, "Rfid-based mobile robot navigation," Master's thesis, Institute of Electrical and Electronic Engineering, University M'Hamed BOUGARA – Boumerdes, 2018.

[51] S. Mokaram, J. M. Aitken, U. Martinez-Hernandez, I. Eimontaite, D. Cameron, J. Rolph, I. Gwilt, O. McAree, and J. Law, "A ros-integrated api for the kuka lbr iiwa collaborative robot**the authors acknowledge support from the epsrc centre for innovative manufacturing in intelligent automation, in undertaking this research work under grant reference number ep/i033467/1, and the university of sheffield impact, innovation and knowledge exchange grant "human robot interaction development". equipment has been provided under the epsrc great technologies capital call: Robotics and autonomous systems.," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 15859–15864, 2017. 20th IFAC World Congress.

[52] H.-H. Wu and A. Bainbridge-Smith, "Advantages of using a kinect camera in various applications," *University of Canterbury*, pp. 1–4, 2011.