



جامعة النجاح الوطنية

An-Najah National University

كلية الهندسة | Faculty of Engineering

وحدة الجودة والاعتماد - مركز ABET

Quality and Accreditation Unit - ABET Center



An-Najah National University

Faculty of Engineering & Information Technology

Department of Computer Engineering

Intelligent Multi-Mode Robotic Platform

Autonomous Pick-and-Place • AI Tic-Tac-Toe • Mobile Control • Xbox Control

Presented in partial fulfilment of the requirements for Bachelor degree in Computer Engineering.

Accomplished By:

Ali Odeh

Izzeddin Asmar

Supervisor:

Dr. Emad Natsheh

June 9, 2026



جامعة النجاح الوطنية

An-Najah National University

كلية الهندسة | Faculty of Engineering

وحدة الجودة والاعتماد - مركز ABET

Quality and Accreditation Unit - ABET Center



Acknowledgements

This project would not have been possible without the support, guidance, and encouragement of many individuals.

First and foremost, we express our sincere gratitude to Almighty God for granting us the strength, patience, and perseverance needed to successfully complete this project.

We would also like to extend our deepest appreciation to our families for their unwavering support, understanding, and encouragement throughout this journey. Their belief in us and constant motivation played a significant role in helping us achieve our goals.

A special thanks goes to Dr. Emad Natsheh for his invaluable guidance, expertise, and continuous support throughout the project. His insightful advice and constructive feedback greatly contributed to the successful completion of this work.

We are also grateful to our friends and everyone who contributed directly or indirectly to this project. Their encouragement and assistance are sincerely appreciated.

To all those who supported us along the way, we offer our heartfelt thanks.



جامعة النجاح الوطنية

An-Najah National University

كلية الهندسة | Faculty of Engineering

وحدة الجودة والاعتماد - مركز ABET

Quality and Accreditation Unit - ABET Center



Disclaimer Statement

DISCLAIMER

This report was written by students at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.



Table of Contents

Contents

Acknowledgements.....	i
Disclaimer Statement	ii
Table of Contents.....	iii
List of Abbreviations.....	iv
Abstract.....	v
Chapter 1: Introduction.....	1
1.1 General Background.....	1
1.2 Objectives of the Work.....	1
1.3 Significance of the Work	1
1.4 Organization of the Report.....	2
Chapter 2: Theoretical Background and Previous Work	3
2.1 Background.....	3
2.2 Previous and Related Work.....	3
Chapter 3: Methodology.....	5
3.1 Standards and Specifications	5
3.2 Design Constraints	5
3.3 Design Overview	6
3.4 Control and Automation	7
3.5 Programming and Circuit Integration	8
3.5.1 Vision-Guided Autonomous Pick-and-Place Mode.....	8
3.5.2 OpenAI-Based Robotic Tic-Tac-Toe Mode	9
3.5.3 Mobile Application Module	10
3.5.4 Xbox-Based Robot Control Module.....	11
3.6 Safety Measures	12
3.7 Component List and Description	12
Chapter 4: Results and Analysis	18
4.1 The Design.....	18
4.2 Final System	18



4.3 Project Outcomes.....	19
Chapter 5: Discussion.....	21
Chapter 6: Conclusions and Recommendations	22
6.1 Conclusions.....	22
6.2 Recommendations.....	22
6.3 Future Work.....	23
References	24

List of Abbreviations

YOLOv8 – You Only Look Once version 8 (real-time object detection model)

OAK-D – OpenCV AI Kit with Depth (stereo depth camera)

UART – Universal Asynchronous Receiver Transmitter

SPP – Serial Port Profile (Bluetooth standard)

PWM – Pulse Width Modulation

HC-05 – Bluetooth Serial Module

SDK – Software Development Kit

AI – Artificial Intelligence

ML – Machine Learning

IoT – Internet of Things

DC – Direct Current

IDE – Integrated Development Environment

API – Application Programming Interface



Abstract

Robotics and computer vision are increasingly being applied to create intelligent systems that bridge the gap between digital perception and physical interaction. Toy sorting and object manipulation, tasks that traditionally require direct human involvement, present a compelling opportunity to demonstrate how automation can be made interactive, engaging, and accessible. This project is significant because it explores that intersection — combining real-time object detection, autonomous robotic manipulation, and interactive gameplay into a single low-cost platform, demonstrating that intelligent robotic systems are not limited to industrial settings but can be designed for interactive and educational purposes as well.

The most important aspects of this project are the vision-based perception system, the autonomous decision-making pipeline, the mechanical manipulation capability, and the multi-modal control architecture. Together, these aspects form a complete end-to-end system that takes a visual input, processes it intelligently, and produces a meaningful physical action — which is the fundamental challenge at the heart of modern robotics.

The main objectives of this project are: to develop a real-time object detection system capable of identifying and classifying toy objects using a custom-trained YOLOv8 model; to enable a mobile robot to autonomously navigate toward detected objects, pick them up using a servo-driven gripper arm, and deliver them to their correct destination boxes; to implement an interactive Tic-Tac-Toe mode in which the robot plays against a human opponent by reading the board state and physically placing game pieces; and to support manual control through both a Flutter mobile application and an Xbox controller connected via Bluetooth.

The system was developed using a stand-mounted OAK-D depth camera positioned at a near-vertical angle above the play area to capture an overhead view of the workspace. A YOLOv8 object detection model was trained from scratch on a custom-labeled dataset covering seven classes, built and annotated using Roboflow and trained via Google Colab. The detection pipeline runs on a connected laptop and transmits movement commands wirelessly to the robot via an HC-05 Bluetooth module. The robot itself is built on a Mecanum-wheel platform controlled by an Arduino Mega, with a 3D-printed servo-driven arm handling object manipulation. The Flutter mobile application and Xbox controller provide additional manual control interfaces, both communicating through the same Bluetooth channel.

Similar robotic sorting systems have been explored in both academic and commercial contexts. Projects such as robotic arms guided by overhead cameras for bin sorting and warehouse automation share conceptual similarities with this work. Consumer products such as robotic vacuum cleaners also demonstrate environment-aware autonomous navigation. However, the specific combination of overhead vision-guided toy sorting, interactive Tic-Tac-Toe gameplay,



جامعة النجاح الوطنية

An-Najah National University

كلية الهندسة | Faculty of Engineering

وحدة الجودة والاعتماد - مركز ABET

Quality and Accreditation Unit - ABET Center



multi-modal manual control, and a fully custom-trained detection model in a single low-cost prototype represents a novel and distinct contribution that distinguishes this project from existing work.



Chapter 1: Introduction

1.1 General Background

The integration of robotics, computer vision, and artificial intelligence has opened new possibilities for creating systems that go beyond simple automation — systems capable of perceiving their environment, making decisions, and physically interacting with objects in real time. These capabilities, once confined to industrial settings, are increasingly being explored in interactive and educational contexts, where robotics can serve both functional and engaging purposes.

1.2 Objectives of the Work

The main objectives of this project are:

- To develop a real-time object detection system capable of identifying and classifying toy objects using a custom-trained YOLOv8 model.
- To enable a mobile robot to autonomously navigate toward detected objects, pick them up using a servo-driven gripper arm, and deliver them to their correct destination boxes.
- To implement an interactive Tic-Tac-Toe mode in which the robot plays against a human opponent by reading the board state and physically placing game pieces.
- To support manual control through both a Flutter mobile application and an Xbox controller connected via Bluetooth.

1.3 Significance of the Work

This project introduces the Smart Interactive Robotic System for Automated Toy Sorting, a robotic platform designed to autonomously detect, classify, and sort toy objects while also functioning as an interactive game-playing system. The project is significant because it explores the intersection of real-time object detection, autonomous robotic manipulation, and interactive gameplay into a single low-cost platform — demonstrating that intelligent robotic systems are not limited to industrial settings but can be designed for interactive and educational purposes as well.

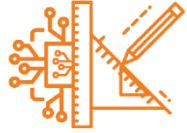


Similar robotic sorting systems have been explored in both academic and commercial contexts. Projects such as robotic arms guided by overhead cameras for bin sorting and warehouse automation share conceptual similarities with this work. However, the specific combination of overhead vision-guided toy sorting, interactive Tic-Tac-Toe gameplay, multi-modal manual control, and a fully custom-trained detection model in a single low-cost prototype represents a novel and distinct contribution.

1.4 Organization of the Report

The remainder of this report is organized as follows:

- Chapter 2 presents the theoretical background and an overview of related previous work.
- Chapter 3 outlines the constraints encountered during development.
- Chapter 4 details the system methodology, including design overview, hardware, software, standards, and communication architecture.
- Chapter 5 presents the results and discussion.
- Chapter 6 provides conclusions and recommendations for future work.
- References and appendices follow at the end of the report.



Chapter 2: Theoretical Background and Previous Work

2.1 Background

The Smart Interactive Robotic System for Automated Toy Sorting draws on several foundational areas of engineering and computer science, including computer vision, deep learning, embedded systems, and human-robot interaction.

Object detection using deep learning has advanced significantly in recent years. The YOLO (You Only Look Once) family of models, of which YOLOv8 is the latest generation, is widely used for real-time object detection due to its balance of speed and accuracy. YOLOv8 enables single-pass detection across an image and has demonstrated strong performance across a variety of custom datasets and deployment environments.

Omnidirectional robotic platforms using Mecanum wheels allow a mobile robot to move freely in any direction without requiring a change in orientation. This capability is particularly valuable in constrained environments where maneuverability is critical. Mecanum-wheel platforms are commonly used in research and industrial robotics.

The OAK-D (OpenCV AI Kit with Depth) camera integrates stereo depth sensing with an onboard visual processing pipeline, providing both RGB video and depth information. Its use in overhead and fixed-perspective configurations has been demonstrated in various vision-guided robotics projects.

2.2 Previous and Related Work

Robotic sorting systems guided by overhead cameras have been explored extensively in academic literature and industrial automation. Research projects have demonstrated pick-and-place robotic arms using convolutional neural networks for object recognition and localization, particularly in bin-picking and warehouse automation contexts.

Consumer products such as robotic vacuum cleaners represent commercial implementations of environment-aware autonomous navigation, demonstrating the maturity of perception-driven mobile robotics in accessible and interactive contexts.



جامعة النجاح الوطنية An-Najah National University

كلية الهندسة | Faculty of Engineering
وحدة الجودة والاعتماد - مركز ABET
Quality and Accreditation Unit - ABET Center



Human-robot interaction through game-playing — including chess, checkers, and Tic-Tac-Toe — has been studied as a means of making robotic systems more engaging and accessible to non-expert users. These systems typically rely on a combination of board state recognition via computer vision and a game-playing algorithm for move selection.

The present project draws from all of these areas and combines them into a single cohesive low-cost prototype, representing a novel integration not addressed by any single prior work.



Chapter 3: Methodology

The methodology for developing the Smart Interactive Robotic System for Automated Toy Sorting was structured to achieve accurate object detection, reliable robotic manipulation, and smooth multi-modal control. The system integrates a stand-mounted OAK-D depth camera for overhead visual perception, a laptop running a custom-trained YOLOv8 model for real-time object detection, an Arduino Mega for motor and gripper control, and an HC-05 Bluetooth module for wireless communication.

3.1 Standards and Specifications

In developing this system, established engineering and communication standards were followed to ensure reliable and safe operation across all subsystems.

The Bluetooth Serial Port Profile (SPP) standard, implemented through the HC-05 module, governed all wireless communication between the Arduino Mega and external control sources, including the laptop running the detection pipeline, the Flutter mobile application, and the Xbox controller. Communication between the HC-05 module and the Arduino Mega followed the UART (Universal Asynchronous Receiver Transmitter) protocol, ensuring consistent and real-time data transfer for robotic control.

The object detection pipeline was built on YOLOv8, a widely adopted deep learning framework for real-time object detection, trained from scratch on a custom-labeled dataset. All electrical components were designed to operate within standard low-voltage DC ranges, ensuring safe interfacing between the Arduino Mega, motor drivers, servo motors, and connected peripherals.

3.2 Design Constraints

The development of this system was shaped by several practical constraints that influenced design decisions and overall system performance:

Economy: Due to budget constraints, the system was powered through a wired power supply, replacing the planned battery-based approach. Available batteries provided only approximately 15 minutes of operation per charge, which was insufficient for reliable demonstration and testing.



Environment: The object detection model demonstrated reduced reliability under low or inconsistent lighting conditions. Since the OAK-D camera is mounted on a fixed stand at a near-vertical angle, the system is fully dependent on ambient lighting with no ability to compensate for shadows or dim environments.

Manufacturability: The structural parts of the robot, including the gripper and arm components, were fabricated using a 3D printer. While this made custom part manufacturing accessible and cost-effective, 3D-printed parts inherently carry dimensional tolerances and minor imperfections that reduced the mechanical precision of the gripper.

Health and Safety: All wiring and power rails were insulated and organized to prevent short circuits and accidental disconnections during operation. Appropriate motor drivers were used to safely control the NEMA 17 stepper motors without exceeding their rated current limits.

Sustainability: The fixed camera field of view limits the operational area of the system to the camera's coverage region. The Bluetooth range limitation also restricts the distance from which manual control can be reliably maintained.

Reliability: The gripper occasionally collided with the target object during the approach phase rather than cleanly enclosing it, requiring manual repositioning of the object to complete a successful grasp. This stemmed from the sub-centimeter precision required for reliable pick-up not always being achieved by the navigation system.

3.3 Design Overview

The system is divided into three main subsystems:

1. Vision and Detection Subsystem: A stand-mounted OAK-D depth camera is positioned at a near-vertical angle above the play area, providing a stable overhead view of the workspace. A laptop connected to the camera runs the custom-trained YOLOv8 model, performing real-time detection and classification of target objects as well as environmental landmarks. Detected positions are translated into movement commands and transmitted wirelessly via Bluetooth to the robot.

2. Robotic Platform and Arm Subsystem: The mobile base is built on a chassis fitted with Mecanum wheels, enabling omnidirectional movement with precise positional control. Mounted on the platform is a 3D-printed robotic arm with 180 degrees of freedom, driven by



MG996R and SG90 servo motors, responsible for picking up detected objects and placing them into their designated boxes.

3. Control and Communication Subsystem: All wireless communication is handled through the HC-05 Bluetooth module interfaced with the Arduino Mega via UART. The system supports three control sources — the laptop vision pipeline for autonomous operation, a Flutter mobile application for manual control, and an Xbox controller operated through the mobile app.

3.4 Control and Automation

The Arduino Mega serves as the central controller for both the robotic platform and the mounted arm, receiving all commands through the HC-05 Bluetooth module. The system supports two primary operation modes: autonomous and manual.

In autonomous mode, the laptop processes the OAK-D camera feed through the YOLOv8 detection model, identifies target objects and their positions within the play area, and transmits movement and sorting commands to the Arduino Mega via Bluetooth. The Arduino then drives the Mecanum wheels to navigate the robot toward the detected object, activates the arm, and commands the gripper to pick up and deliver the object to its designated box.

In manual mode, the user controls the robot through the Flutter mobile application, which communicates with the HC-05 Bluetooth module. The Xbox controller is also supported through the mobile app, relaying its inputs as Bluetooth commands to the robot. Only one control source can be active at a time, ensuring safe and conflict-free operation.



3.5 Programming and Circuit Integration

3.5.1 Vision-Guided Autonomous Pick-and-Place Mode

In this mode, the robot operates autonomously to detect, pick up, verify, and deliver a target object inside the arena. The system uses an OAK-D camera with a trained YOLO model to detect the target objects, the robot gripper, the pink alignment marker, and the destination boxes. An additional HSV-based color assist module is used to improve detection reliability when the YOLO model temporarily misses important visual elements.

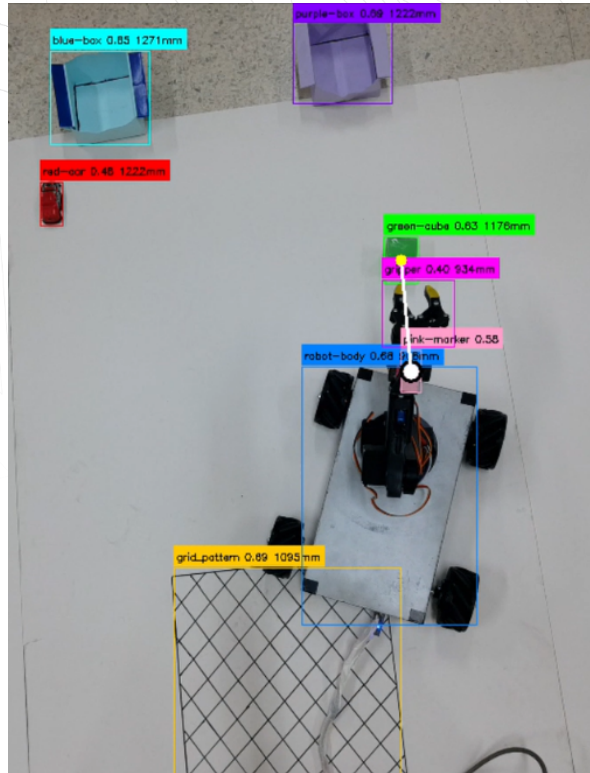
The robot supports two target-object mappings: the green-cube is delivered to the blue-box, while the red-car is delivered to the purple-box. After detecting the selected target, the control algorithm calculates the robot-relative alignment using the pink marker as a body reference and the gripper midpoint as the front direction. The controller estimates lateral error, forward distance, and angular error, then sends movement commands such as forward, backward, strafe, rotate, or diagonal motion until the object is centered in front of the gripper.

The robot also applies a nearest-object selection strategy. It first moves toward the closest detected target object, picks it up, delivers it to its assigned box, and then scans the arena again. If another object is detected, the robot repeats the same process. When all target objects are completed, the robot returns to its home position.

Once stable alignment is achieved, the system performs a final pre-pickup check before sending the pickup command to the Arduino through Bluetooth serial communication. The Arduino receives single-byte commands and controls the wheels, servos, gripper, pickup motion, and drop mechanism.

After pickup, the system captures several camera frames and uses visual verification to determine whether the object was successfully carried. The result can be classified as confirmed, failed, or uncertain. If the pickup is confirmed, the robot enters the delivery stage, aligns itself with the correct destination box, holds the alignment briefly, and then sends the drop command.

This mode combines object detection, geometric alignment, serial communication, robotic manipulation, and visual verification to create a complete autonomous pick-and-place workflow.



3.5.2 OpenAI-Based Robotic Tic-Tac-Toe Mode

In this mode, the robot plays a physical Tic-Tac-Toe game against a human player using AI from OpenAI. The human player uses `O`, while the robot/OpenAI uses `X`. A camera captures the numbered 3x3 board, where each cell has a fixed number from 1 to 9. The frame is sent to OpenAI to analyze the current board state and support the decision of where the next `X` should be placed.

At startup, the system waits 5 seconds and scans the board. If the player has already placed an `O`, OpenAI analyzes the image and the robot responds with an `X`. If the board is empty, the system waits another 10 seconds and scans again. If it is still empty, the robot/OpenAI starts first. During the game, the system keeps an internal board state and updates it after every scan.

After the next move is selected, the Python controller sends a command to the Arduino in the format `MOVE <pickup_index> <cell_number>`. The robotic arm first picks an X piece from a fixed stack. Since the X pieces are stacked on top of each other, each pickup index has a different predefined motion because the pickup height changes after each attempt. Then, the arm moves to the selected numbered cell using its corresponding predefined placement movement, releases the X, and returns to its home position.



After every robot move, the system waits for the human player, scans the board again, and checks whether a new `O` was placed. This process continues until the game ends with a win, loss, or draw.

3.5.3 Mobile Application Module

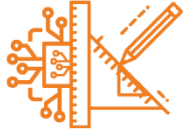
The project includes a mobile application developed using Flutter and Dart to provide a direct control interface for the robot. The application connects to the robot through Bluetooth Classic using the HC-05 Bluetooth module, which acts as a wireless serial bridge between the Android phone and the Arduino controller.

Before establishing the connection, the application requests the required Android Bluetooth and location permissions. After that, it displays the paired Bluetooth devices and allows the user to select the HC-05 module. Once connected, the application sends control commands to the Arduino using a one-byte command protocol that matches the robot firmware.

The application interface contains two main control sections. The first section controls the mecanum wheel base, allowing the robot to move forward, backward, sideways, diagonally, and rotate left or right. These controls use a press-and-hold mechanism, where pressing a button sends the movement command and releasing it immediately sends a stop command for safety.

The second section controls the robotic arm, including the waist, shoulder, elbow, wrist roll, wrist pitch, and gripper. Each arm joint has directional control buttons, and the application continuously sends servo commands while the button is pressed. The app also includes speed sliders for both platform movement and arm movement, allowing the user to adjust the robot's response speed during operation. With the ability to control the speed of the wheels or the arm

In addition, the application can receive angle feedback from the robot and display the current servo angles for the six joints. This helps the user monitor the arm position during manual control. Overall, the mobile application provides a wireless, user-friendly interface for operating the robot's movement system and robotic arm through Bluetooth communication.



3.5.4 Xbox-Based Robot Control Module

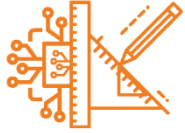
This control module allows the robot to be operated using an Xbox controller, with the mobile phone acting as an intermediate device between the controller and the robot. The Xbox controller is connected to the Android phone, while the phone remains connected to the robot through the HC-05 Bluetooth module. The Flutter application reads the Xbox inputs, converts them into the same one-byte robot commands used by the mobile touch interface, and sends them wirelessly to the Arduino.

In Xbox mode, the controller inputs are mapped to both the mecanum wheel base and the robotic arm. The left joystick controls platform movement, such as moving forward, backward, and rotating. The right joystick controls arm joints such as the shoulder and elbow. The bumpers control the waist, the triggers control the gripper, and the D-pad controls wrist movements. A dedicated button is also used as an emergency stop. With the ability to control the speed of the wheels or the arm

The module also includes a programmable movement feature. When the user enters program mode, the mobile application displays a notification asking the user to choose one of the Xbox buttons, such as A, B, X, or Y. After selecting the button, the user manually performs a movement sequence using the controller. The application records the sequence of commands, saves it to the selected button, and confirms the saved program through a mobile notification. Later, when the user presses that programmed button, the application replays the stored command sequence automatically and sends it to the robot.

To improve safety and stability, the application reads the Xbox controller at a fixed time interval and only sends new commands when the control state changes. A joystick dead zone is used to prevent small accidental movements from being interpreted as commands. If the controller is released, disconnected, or Xbox mode is exited, the application sends a stop command to prevent unintended robot movement.

Overall, this module provides a flexible manual control method that combines Xbox controller input, mobile-phone command processing, Bluetooth communication, and Arduino-based robot execution.



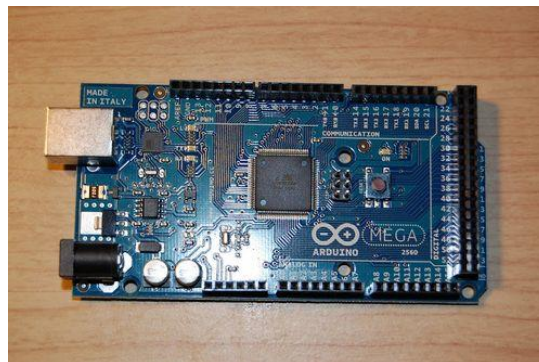
3.6 Safety Measures

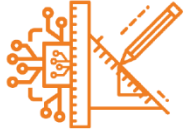
All wiring and power rails were insulated and organized to prevent short circuits and accidental disconnections during operation. Appropriate motor drivers were used to safely control the NEMA 17 stepper motors without exceeding their rated current limits, and heat management was considered during the placement of the A4988 drivers to prevent overheating during extended use. The robotic arm and gripper were tested under various load conditions to verify safe current consumption and reliable mechanical performance.

3.7 Component List and Description

The system is built using the following key hardware and software components:

- **Arduino Mega:** The central microcontroller responsible for managing all motor control, servo actuation, and Bluetooth communication logic.





- **OAK-D Depth Camera:** A stereo depth camera mounted on a stand at a near-vertical angle, used to capture the overhead view of the play area for real-time object detection.

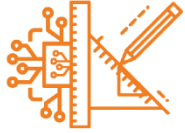


- **XL4015 Buck Converter:** A DC-DC step-down converter used to reduce the 12V output from the power supply to a stable 5V output required for powering the system's low-voltage electronic components.



- **NEMA 17 Stepper Motors:** Used to drive the Mecanum wheel platform, providing precise and controllable movement in all directions.





- **A4988 Motor Drivers:** Stepper motor driver modules used to control the NEMA 17 motors, with current limiting to ensure safe operation.

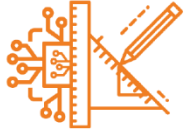


- **MG996R Servo Motor:** A high-torque servo motor used to drive the primary joints of the robotic arm.



- **SG90 Servo Motor:** A lightweight micro servo used for finer arm movements and gripper actuation.



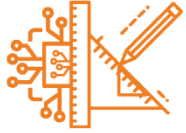


- **HC-05 Bluetooth Module:** Handles all wireless communication between the Arduino Mega and external control sources via the Bluetooth SPP standard.



- **Mecanum Wheels:** Four omnidirectional wheels that allow the robot to move forward, backward, laterally, and diagonally without rotating.



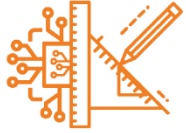


- **Wired Power Supply:** Provides stable and continuous power to all system components, replacing battery-based power due to budget and runtime constraints.

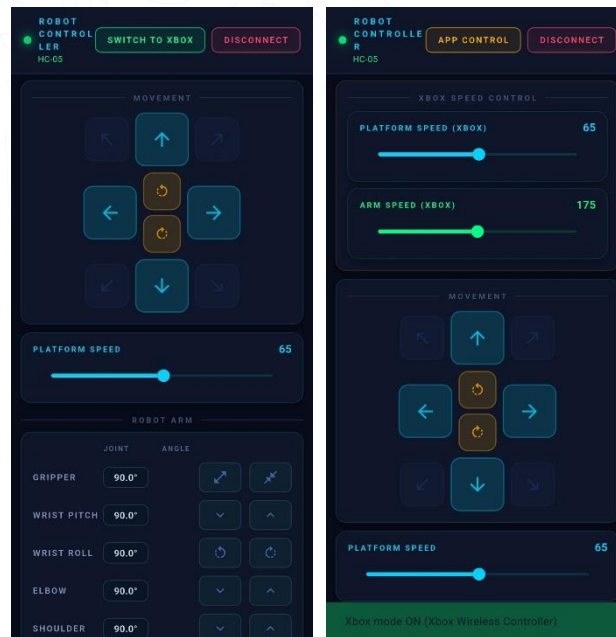


- **Laptop:** Runs the Python-based YOLOv8 detection pipeline and communicates commands to the robot via Bluetooth.





- **Flutter Mobile Application:** A cross-platform application providing manual control of the robot and relaying Xbox controller inputs via Bluetooth.



- **Xbox Controller:** Used as an alternative manual input device, operated through the Flutter mobile app.





Chapter 4: Results and Analysis

4.1 The Design

The Smart Interactive Robotic System for Automated Toy Sorting was designed with a focus on modularity, precision, and multi-modal interaction. The system brings together an overhead vision pipeline, a mobile omnidirectional platform, and a servo-driven robotic arm into a single cohesive interactive system.

The base of the robot is constructed around a chassis fitted with Mecanum wheels, enabling omnidirectional movement — forward, backward, lateral, and diagonal — without the need to rotate the platform. This wheel configuration was chosen specifically for its ability to make precise positional corrections during the object approach phase, which is critical for successful gripper alignment.

Mounted on the platform is a 3D-printed robotic arm with 180 degrees of motion, driven by MG996R and SG90 servo motors. The arm was custom-designed and fabricated in-house to fit the spatial requirements of the play area and the scale of the target objects. While 3D printing made the fabrication process accessible and cost-effective, it introduced mechanical tolerances that affected gripper rigidity and contributed to occasional object drops during transit.

The vision subsystem consists of a stand-mounted OAK-D depth camera positioned at a near-vertical angle approximately 110 cm above the play area. The camera feeds live video to a laptop running the custom-trained YOLOv8 detection model, which identifies objects and landmarks in real time and generates movement commands transmitted to the robot via Bluetooth.

4.2 Final System

The final outcome of the Smart Interactive Robotic System for Automated Toy Sorting consists of:

- A mobile robotic platform equipped with Mecanum wheels for omnidirectional movement.
- A 3D-printed robotic arm mounted on the platform, driven by MG996R and SG90 servo motors for object grasping and placement.
- An Arduino Mega controller managing all motor control, servo actuation, and Bluetooth communication.



- A stand-mounted OAK-D depth camera at approximately 110 cm height, providing a near-vertical overhead view of the play area.
- A custom YOLOv8 model trained from scratch at two heights (80 cm and 110 cm) for robust object detection across varying conditions.
- An HC-05 Bluetooth module handling all wireless communication between the laptop, mobile app, Xbox controller, and the robot.
- Three control modes: autonomous vision-guided sorting, manual control via a Flutter mobile application, and manual control via an Xbox controller relayed through the app.

4.3 Project Outcomes

Object Detection Accuracy: The YOLOv8 model achieved a detection accuracy exceeding 90% during testing at the trained operating height of 110 cm. Training the model at two different heights — 80 cm and 110 cm — improved its robustness and ensured reliable detection across a range of camera positions. Detection performance, however, was notably reduced under low lighting conditions, which remains an identified area for future improvement.

Autonomous Toy Sorting: The robot successfully navigated toward detected objects in the majority of test runs. In approximately half of the attempts, the gripper successfully grasped the object without requiring any manual adjustment. In the remaining cases, minor repositioning of the object was needed due to collision during the approach phase, a consequence of the sub-centimeter precision required for reliable gripper engagement.

Tic-Tac-Toe Mode: The interactive Tic-Tac-Toe mode was successfully implemented, with the overhead camera reading the board state in real time and the robot physically placing game pieces on the board in response to the human opponent's moves, demonstrating the system's ability to operate as an interactive game-playing platform.

Manual Control: Both manual control modes — the Flutter mobile application and the Xbox controller relayed through the app — performed with full reliability throughout all testing sessions. Navigation and arm control responses were smooth and consistent, with no connectivity or latency issues observed during operation.

Overall System Integration: The project successfully demonstrated the feasibility of combining real-time computer vision, machine learning, embedded control, and mechanical



جامعة النجاح الوطنية

An-Najah National University

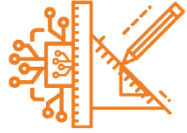
كلية الهندسة | Faculty of Engineering

وحدة الجودة والاعتماد - مركز ABET

Quality and Accreditation Unit - ABET Center



manipulation into a single interactive platform. Despite the mechanical limitations introduced by 3D-printed components and the gripper precision challenges, the system met its core objectives and laid a solid foundation for future refinement.



Chapter 5: Discussion

The results demonstrate that the Smart Interactive Robotic System for Automated Toy Sorting successfully achieved its core design objectives. The integration of a custom-trained YOLOv8 detection pipeline with an omnidirectional mobile platform and a servo-driven gripper arm produced a functional end-to-end robotic system capable of autonomous object sorting and interactive gameplay.

The object detection component performed reliably under controlled lighting conditions, achieving greater than 90% accuracy. However, performance degraded under low or inconsistent illumination, highlighting the system's current dependency on ambient lighting. This is a known limitation of camera-based perception systems and points toward the need for either software-based preprocessing or dedicated lighting infrastructure.

The gripper precision challenge — where objects were occasionally collided with during approach rather than cleanly enclosed — reveals an inherent tradeoff between the visual navigation accuracy achievable through Bluetooth-based command latency and the sub-centimeter mechanical precision required for reliable grasping. The 3D-printed construction of the gripper further compounded this issue through dimensional tolerances inherent to fused deposition manufacturing.

The Tic-Tac-Toe mode demonstrated that the same overhead vision pipeline could be adapted for a rule-based interactive task with minimal modification, confirming the architectural flexibility of the system.

The multi-modal manual control architecture proved robust throughout testing. The Flutter application and Xbox controller relay operated without latency or connectivity issues, validating the Bluetooth SPP communication design.

Overall, the project validated its proof-of-concept goals and identified clear, actionable paths for improvement in gripper design, lighting robustness, and onboard processing capability.



Chapter 6: Conclusions and Recommendations

6.1 Conclusions

This project successfully demonstrated the development of the Smart Interactive Robotic System for Automated Toy Sorting, a robotic platform capable of autonomously detecting, classifying, and sorting toy objects while also functioning as an interactive Tic-Tac-Toe game-playing system. By integrating a custom-trained YOLOv8 vision pipeline, Mecanum-wheel omnidirectional mobility, a servo-driven robotic arm, and multi-modal wireless control, the prototype achieved its primary objectives and validated the feasibility of combining computer vision, machine learning, and robotic actuation into a single cohesive system.

The project fulfilled its core design goals by achieving the following:

- **Accurate Object Detection:** The custom YOLOv8 model, trained from scratch on a purpose-built dataset at two operating heights, achieved a detection accuracy exceeding 90%.
- **Autonomous Toy Sorting:** The robot successfully navigated toward detected objects and performed pick-up and delivery operations across the majority of test runs.
- **Interactive Gameplay:** The Tic-Tac-Toe mode was successfully implemented, with the robot reading the board state and physically placing game pieces in response to the human opponent's moves but there's some limitations is the accuracy due to the 3d printed components.
- **Reliable Manual Control:** Both the Flutter mobile application and the Xbox controller provided smooth, responsive, and fully reliable manual control throughout all testing sessions.
- **Interdisciplinary Integration:** The project brought together computer vision, embedded systems, mechanical design, and mobile application development into a single functional prototype.

6.2 Recommendations

While the project objectives were met, several improvements are recommended to enhance performance, reliability, and scalability:

- **Improve Gripper Design:** The 3D-printed gripper should be redesigned with tighter mechanical tolerances or fabricated using more rigid materials. Incorporating a



feedback mechanism to confirm successful grasps before the robot begins moving would further improve reliability.

- **Enhance Lighting Robustness:** The detection pipeline should be made more resilient to varying lighting conditions through software-based image preprocessing techniques such as histogram equalization and adaptive brightness correction, or by incorporating dedicated lighting fixtures.
- **Upgrade to Wireless Operation:** Replacing the wired power supply with a higher-capacity battery system would eliminate the mobility constraint and enable fully untethered autonomous operation.
- **Increase Detection Dataset Diversity:** Expanding the training dataset to include a wider variety of lighting conditions, object orientations, and camera heights would improve the model's generalization and robustness.
- **Upgrade Communication Protocol:** Replacing the HC-05 Bluetooth module with a higher-range and higher-bandwidth wireless protocol such as Wi-Fi or ESP-NOW would extend the operational range and support faster communication.

6.3 Future Work

The current prototype establishes a solid foundation for a more advanced system. Future development efforts could explore the following directions:

- **Onboard Processing:** Migrating the YOLOv8 detection pipeline from a connected laptop to an onboard computing unit such as a Jetson Nano or Raspberry Pi would enable fully self-contained autonomous operation.
- **Expanded Object Recognition:** Extending the detection model to classify a broader range of toy types and categories would increase the system's versatility.
- **Improved Arm Precision:** Redesigning the robotic arm with higher-precision servo motors and rigid non-printed structural components would significantly improve pick-up reliability.
- **Depth-Based Grasping:** Leveraging the OAK-D camera's depth sensing capabilities more extensively to calculate precise 3D object positions would enable more accurate gripper alignment.
- **Expanded Game Modes:** Adding additional interactive game modes beyond Tic-Tac-Toe would enhance the system's value as an interactive platform.
- **Full Wireless Mobility:** Developing a compact and long-lasting battery solution with proper power management circuitry would allow the robot to operate freely without a tethering power cable.



References

Jocher, G., Chaurasia, A., & Qiu, J. (2023). YOLO by Ultralytics (Version 8.0.0) [Computer software]. <https://github.com/ultralytics/ultralytics>

Luxonis. (2023). OAK-D depth camera documentation. Luxonis. <https://docs.luxonis.com>

Arduino. (2023). Arduino Mega 2560 microcontroller reference. Arduino. <https://docs.arduino.cc/hardware/mega-2560/>

Roboflow. (2023). Roboflow: Computer vision dataset management [Software]. <https://roboflow.com>

Flutter. (2023). Flutter documentation. Google. <https://flutter.dev/docs>

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 779-788.

Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. (2011). Introduction to autonomous mobile robots (2nd ed.). MIT Press.

Siciliano, B., & Khatib, O. (Eds.). (2016). Springer handbook of robotics (2nd ed.). Springer.