Low-Cost Movable Robot Arm to Play Chess

by

Hemaharshini Ragupathy

A project submitted in Partial Fulfillment
of the Requirements for the Degree of

Master of Science in Computer Science

SUPERVISED BY

DR. SABAH MOHAMMED

Department of Computer Science

Lakehead University

Thunder Bay, Ontario, Canada.

June 2019

Abstract

Arduino UNO is a more convenient computer for projects of IoT. The use of this Robotic Arm makes life easier. Nowadays, Technology has developed in the same direction in line with rapidly increasing human needs, and Robots are an excellent tool to help humankind one of the best innovations in history. Among the different types of robots, the robot arm is one of a kind. Game playing robots are also popular among them. Arduino board is programmed to control the servo motors, and Arduino's analog input is given to the Servo Motors. The Arduino has been programmed to provide rotation to each servo motor corresponding to the amount of rotation. Using the developed Android application, the user can control the Robot arm to pick and move the chess piece. A pixy2 camera is used for object detection and tracking. Real-time object detection based on the Internet of things is more suitable as it is small. This Robotic Arm is low-cost, and feasible in case of personal and educational purpose.

Table of Contents

List of Tables

List of Figures

Acknowledgments

Firstly, I would like to show my sincerest gratitude to my supervisor, Dr. Sabah Mohammed, a decent and professional scholar. It is my honour to work with him. He provided insightful guidance for my academic project. When I encountered obstacles, he encouraged me with great enthusiasm, impressive kindness and extraordinary patience. I appreciate Dr. Sabah Mohammed for providing me with the opportunity to improve and express my knowledge. I would also like to express my special thanks to the Department of Computer Science Graduate Studies at Lakehead University for allowing me to attest myself.

Finally, I would like to thank my parents for their help and support.

Chapter 1

1.1 Introduction

Nowadays, Technology has developed in the same direction in line with rapidly increasing human needs, and Robots are an excellent tool to help humankind one of the best innovations in history. They can be customized according to ones need. Robots are widely used in almost all the applications of the world, such as; Aerospace, Disaster response, Drones, Education, Entertainment, Industrial and Medical, the most common is the industrial and medical sector. Every robot is different in its features; they vary in size, shape and skills. Among the different types of robots, the robot arm is one of a kind.

The mechanical arm is the most common component of the new industrial robot. These industrial robots are mainly used to replace human workers performing dangerous operations. The robot arm in most industries performs useful tasks such as loading, unloading and many more.

(a)

(b)

Figure 1.1: Examples of industrial robot applications (a) Pick and place robot[1] (b) Welding robot[2]

1.2 Robotics and Automation

"Automation technology has matured to a point where several other technologies have developed from it and have achieved a recognition and status of their own" [3]. Generally, robots are known to perform tasks automatically without human intervention. Automation is a technology that performs a process by programming commands which can be combined with automatic feedback control for the proper execution of instructions; thus, the system will be able to conduct operations without human intervention [3].

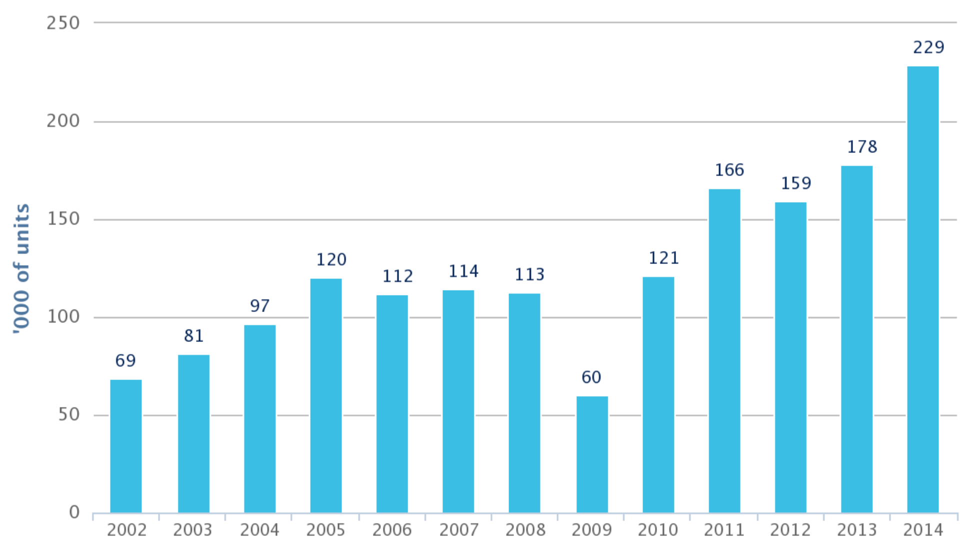

Figure 1.2 Estimated worldwide annual supply of industrial robots [4]

1.3 Motivation

The primary motivation behind my project is based on the industrial applications of lifting, moving the heavy objects and performing dangerous operations without any human intervention. It has proven that industrial robots can save money and time while increasing production and quality [5]. They provide top quality if instructions are good enough, there can be no mistakes. When it comes to safety, some applications require working in hazardous environments such as nuclear power plants which would be impossible for humans to work, industrial robots keep the working environment safer.

1.4 Problem Statement

This project aims to build a simple four-axis robot arm to move objects which are controlled by Arduino UNO, four Servo Motors and a Bluetooth module which therefore builds a connection between the Android application and the circuit. Further, Android applications are used to control the movement of each servo motors. The design of the robot arm is cost-effective and straightforward. In addition to this, a pixy2 camera has been used for object detection.

Figure 1.3 Gambit an autonomous robot playing chess against a child [27]

All the equipment is programmed in such a way that the robot arm picks and moves any desired object. And for the object detection, the pixy2 cam is trained, programmed and attached to Arduino UNO in such a way to fix any object into the frame and to the set the signatures for the respective objects.

Chapter 2

This section will describe the literature review of the essential elements that are to be considered while developing the game playing robot arm. It consists of the explanations about the related research and introduction of the essential terms and their definitions.

2.1 Overview

The history of industrial automation is characterized by periods of rapid change in popular methods. Either as a cause or, perhaps, an effect, such periods of change in automation techniques seem closely tied to world economics. Use of the industrial robot, which became identifiable as a unique device in the 1960s, along with computer-aided design systems and computer-aided manufacturing systems, characterizes the latest trends in the automation of the manufacturing process. These technologies are leading industrial automation through another transition, the scope of which is still unknown.

IoT is one of the promising technologies for automation and control of robotics. Internet of things mainly communicates between "things" and transfer network protocols. Combining the Internet-of-Robotic-Things and autonomous robot systems through the sensors and smart objects. Building a robot that can recognize and manipulate objects in a physical environment is a complex and challenging problem. While basic robot kinematics is well understood, the ability to recognize and manipulate arbitrary objects in a complex the environment remains an active area of research [36].

Game playing robots are popular nowadays; they are the most exciting and challenging to develop. Robotic game boards are among the applications in human-robot interaction.

Among the various game boards, the popular one is chess. Robots are designed to play autonomously with board games. Playing board games like chess presents several challenges that include observation of the board and game pieces, reasoning about the game and the game states, handling of the chess pieces, as well as coordinating with the human opponent [35]. The above challenges can be solved by autonomous robots as they are developed in such a way to understand the rules of games and learn to play without human intervention.

2.2 Mind map of keywords

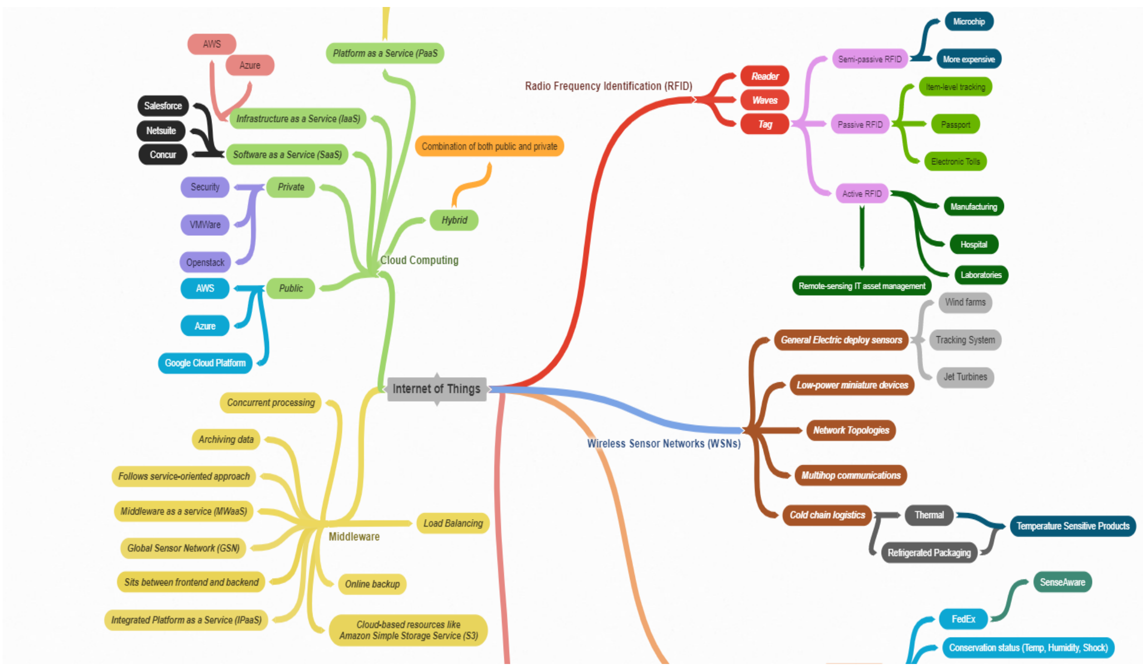

Figure 2.1 Mind map for IoT (first section)

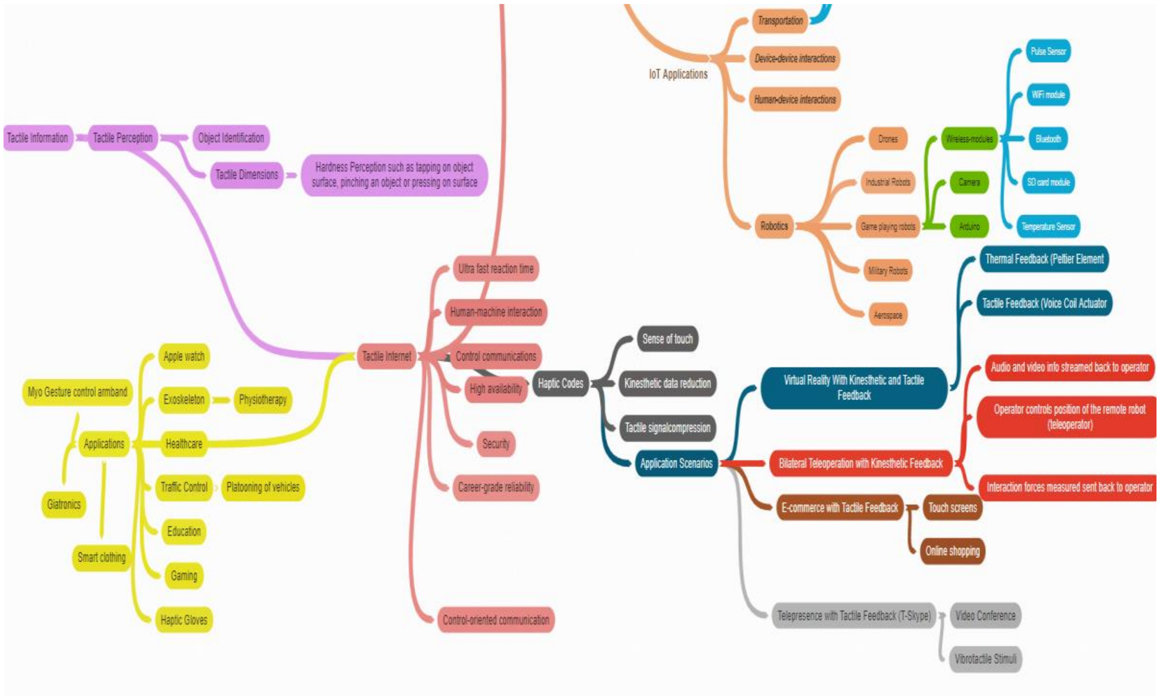

Figure 2.2 Mind map for IoT (second section)

2.3 Critical Review

- **A Smartphone Connected Software Updating Framework for IoT Devices:** The exchange of binaries and profiles between smartphone and IoT devices in a wired or wireless environment can be used to implement a practical updater. By having a software framework and hardware, architecture helps to achieve the result. This paper thus proposes a software framework to abstract the communication of an IoT device and a hardware board to update the application further, which is connected to a smartphone. By having a software framework helps to resolve issues such as OS-independent updates, smartphone ecosystem, which enables users to update the IoT devices. The communication module can support Bluetooth and Wi-Fi. Finally, the architecture here carries an Arduino Board [6].

- **Design of Humanoid Symbiotic Robot TWENDY-ONE:** Authors in this paper developed a new generation symbiotic human-robot named TWENDY ONE that has a head, trunk, two arms, skin and an Omni-wheeled vehicle which provides physical support to the elderly or disabled people in daily life needs. TWENDY ONE offers friendly communication, personal safety assistance and dexterous manipulation. Figure 2.1 shows TWENDY-ONE picking an object from the floor by making joints of waist bend cooperatively [7].
- **A Low-cost Compliant 7-DOF Robotic Manipulator:** This paper presents the design of a low-cost elastic robot arm. As described, it is a safe human design; the arm uses stepper motors for four degrees of freedom which offers high torque at low speeds and robotic servos for three degrees of freedom. The primary aim of this research is to design a robot with great features at a lower price, which requires design trade-offs and compromises [8]. "These design trade-offs were chosen for the envisioned target application of robots interacting with unstructured environments such as typical home or workplace, where the safety of intrinsic mechanical compliance is an important design consideration"[8].
- **A Hybrid Actuation Approach for Human-Friendly Robot Design:** This paper primarily focuses on the safety of robots to operate in human environments. For the development of human-friendly concepts, the authors introduced the concept of hybrid actuation, "which uses artificial muscles augmented with small electrical actuators, human-bone-inspired robotic links"[9] therein uses two degrees of freedom. Therefore, there is only a low complexity in design and assembly. "The artificial pneumatic muscles are to be light, compact and compliant due to air

compressibility"[9]. However, specific issues address this concept, such as range of motion, delay in response and portability [9].

- **Development of a Movable Service Robot with Double working arms for the Elderly and the Disabled:** This paper showcases a service robot that aids the elderly, and the disable with performing simple daily housework. The robot here seems to have a simple structure, low cost with two working arms and also a wheel-based motion method. As per this paper, the robot can perform actions like object tracking, fetching, and transferring [10].
- **Research on Kinematics and Dynamics of a 7-DOF Arm of Humanoid Robot:** The authors in this paper have developed a robot arm which is flexible anthropomorphic seven degrees of freedom. For the self-motion of the arm, the concepts and methodologies of the inverse kinematics base have been followed. "By this method the task and motion sensing can be easily done"[11]. A comparison is made between two previous humanoid success robots WABIAN, ARMRA, and the arm configuration in this paper found that it has dynamic properties [11].
- **Arduino Based Foot Pressure Sensitive Smart Safety System for Industrial Robots:** The primary focus of this paper is having safety while working with a robot in an industrial environment, mainly to avoid industrial mishaps. The authors proposed an approach to shut the robot system or slowing down upon human entry by adding a real-time MEMS sensor. They also focus on any fault during the operation is detected the cabin operator would be informed through Human Machine Interface (HMI). The Arduino micro-controller will trip the robot when

the operator steps on the piezoelectric mat. "It is programmed with a calculated threshold value of the foot pressure corresponding to 60 kg of body mass for the circuit tripping to avoid false triggering"[12]. This safety system can be used in manufacturing units, assembly units, inspection units and packaging units [12].

- **Design and Implementation of Robot-Assisted Surgery based on Internet of Things (IoT):** The authors in this paper have implemented a robot arm that can replace nurses in the hospital. This robot arm can perform tasks such as picking, placing and handing objects to the doctor while performing the surgery. The movements of the robot arm are controlled by an Android application and the signals of the accelerometer; the gyroscope will be captured and sent to the Raspberry Pi, which in turn controls the robot arm. Besides controlling the movements, gestures of the smartphone can also control the movements of the robot arm. "This robot arm can also be controlled from any part of the world as long as it has Internet access"[13]. Besides its features, the limitation is its complexity and efficiency [13].
- **Multi-Sensor Based Glove Control of An Industrial Mobile Robot Arm:** Authors in this paper designed a multi-sensor glove controller controls a six-axis robot arm and a mobile robot. They mainly focus on implementing in a safe working environment. The glove controller is controlled wirelessly to perform human tasks. The primary aim is to make people work and execute their task without entering any dangerous workplaces. The data glove comprises of various sensors to control modes. The results have proven that the robot arm is efficient in

performing human tasks. The mobile robot arm performs tasks such as placing objects in one place to another, which is the job of a typical industrial robot [14].

- **Toward Intelligent Security Robots: A Survey:** This paper surveys on the four-class taxonomy related security robots over the past decades mostly investigated on the crime-fighting robots. The authors "found that an ISR (Intelligent Security Robots) is indeed an artificial embodied agent comprised with advanced perception, capable of taking an intuitive decision and confronting crime by utilizing versatile scenarios"[15]. The security robots that can employ both advanced perception and confrontation can be qualified as ISR [15].
- **Active People Recognition using Thermal and Grey Images on a Mobile Security Robot:** The primary focus of this paper is to identify people on a mobile robot using thermal images and a particle filter. With this method, the person can be identified irrespective of skin colour and light conditions. The thermal images can also estimate the position of the person. The developers of this paper have also used the pan-tilt camera to track the face region of the person with a closer look. "This is a vision-based approach to track and identify people. The authors have proposed the usage of particle filtering in combination with a fast face classifier to accumulate evidence about the identity over time, instead of scanning each image independently from the previous one" [16]. One of the limitations of this development is that the robot can only lock up a single person [16].
- **Development of Industrial Dual Arm Robot for Precision Assembly of Mechanical Parts for Automobiles:** The authors in this paper have developed a dual robot arm for industrial application to perform precision assembly of

mechanical parts for automobiles with six degrees of freedom and three degrees of freedom in the torso. The authors in this paper explain that instead of two arms performing two tasks separately, dual industrial robot arms can perform tasks with more cooperation and precision [17].

- **Case studies of an industrial dual-arm robot applications:** The primary goal of this paper is to replace human workers with industrial robots. The dual-robot arm in this paper can be used for manufacturing IT products such as mobile phones and TV. The robot arm in this paper is used in cosmetic packaging and IT part packaging [18].
- **Small sized industrial dual-arm robot with convenient program interface:** In order to solve the problems in modern manufacturing systems, the authors in this paper have developed a dual-arm robot which can replace human workers. In this paper, the dual-arm robot is used for manufacturing mobile phones and TV. The real-time control capability has been used in the software platform for the robot arm for the precise motion command and user convenience [19].
- **Dual-Arm Robot Box Taping with Kinesthetic Teaching:** The primary focus on this paper is in developing a robot with a dual arm that can perform tasks equal to humans, such as domestic and industrial tasks. However, this requires high-level planning for the robot to operate, say the authors in this paper. "A bimanual manipulation and application of kinesthetic teaching in box taping is described [20]". Three types of kinesthetic teaching modes are introduced to the robots: Firstly, a task is given, and which is further analyzed and finally divided into

segments. The authors explain that if each teaching modes are applied in segments properly, then the dual-robot arm could perform efficiently [20].

- **Analysis of Human Arm Motions at Assembly Work as a Basic of Designing Dual Robot Arm System:** This paper is an analysis of human arm motions to the double-arm robot. Taking into consideration the behaviour of the human arm in the real-time, the model to follow based on human behaviours such as the holding objects differently irrespective of their weight and size. For setting the robot's parameters, human anthropomorphic is required. "A forward kinematic model of the arms using Denavit-Hartenberg approach has been done for modelling purpose. Simulations of the model’s behaviour for each REACH, MOVE, and RELEASE Therbligh motions have been performed. The simulation shows that the REACH motion depends on the 2nd and fourth joint, the MOVE motion depends on the 3rd and fourth joint, and the RELEASE depends on the 4th joint" [21]. One of the limitations in this analysis is stroke patients where the weight of arms should have taken into consideration as masses [21].
- **Wireless teleoperation of an industrial robot by using Myo arm band:** Myo armband has inbuilt sensors such as a gyroscope and electromyography sensors which can perform tasks such as picking and placing the objects. The myo arm controls six degrees of freedom industrial robot in this paper. " The movement of a robot is obtained by using the measured roll, pitch and yaw angle of wearing Myo band" [22]. Wireless communication acts as a medium between the armband and the robot controller. A Bluetooth module is used for this communication. This

study proved that human-robot collaboration is achieved using wearable technology and the position, gripper control performs precisely [22].

- **7 DOF Industrial Robot Controlled by Hand Gestures Using Microsoft Kinect v2:** Kinect is a product developed by Microsoft for the line of motion sensing input devices. The primary aim of this project is to provide a safer environment while working with industrial robots specially to make it easier for the physically disabled people to control the robots by gesture recognition by the usage of hands. One of the limitations in this project is Kinect v2 as Microsoft has discontinued the product, which would question the whole project if it would perform the operations without the Kinect v2 [23].

- **Markerless Gesture-Based Motion Control and Programming of Industrial Robots:** The authors in this paper presented their work on gesture-based control. As these systems are expensive, there are specific requirements to be kept in mind while working on this method such as safety, robustness, gesture recognition and applying this to the industrial environment. "Besides standard online programming methods, they introduce a programming technique, which enables the worker to define a movement by his body parts, e.g. the hand" [24]. They also define a natural way of human-robot communication and intuitive way to define trajectories by the movement of human body parts. Developing suitable gestures to realize the programming tasks is one of the challenges faced [24].

- **Image recognition application for robotics manipulation of moving objects:** Kinematic analysis is followed in this paper to recognize the object using image recognition to position the object for the robot to pick. This is an integration between the computer vision and an industrial robot. The geometric shapes are on a moving conveyer belt for the robot to position using image recognition in which the " code's main features is the possibility to choose among different shapes like triangles, rectangles, and circles; and that the found object's centroid is presented in world units" [25]. The authors in the paper explained that the vision system can detect an object as long as the invariant moment of the object is known. A path was computed between the centroids and the object selected, which is further manipulated for the robot to pick and place the objects at a different point [25].

- **Kinect Controlled Chess Playing Robot:** Based on the Kinect vision sensor, a chess-playing system was developed by the authors in this paper. It can recognize different hand gestures when the commands are sent to the robot simulation tool called V-REP. According to the commands sent to the vision software, the delta type robot will play the chess [26].
- **Gambit: An Autonomous Chess-Playing Robotic System:** The authors in this developed a customized, low-cost autonomous degree of a freedom robot arm which can play board games against human opponents. As board games have always been challenging, this Gambit includes a Kinect-style visual sensor for automatic detection and recognition of boards and objects. "As a use-case, the authors describe playing chess quickly and accurately with arbitrary, boards and pieces, demonstrating that Gambit’s engineering and the design represents a new state-of-the-art in fast, robust tabletop manipulation" [27]. One of the main advantages in this project is that the Gambit does not require any computer display

or input devices to play against human as it uses speech synthesis, own body gestures and board's perception. Moreover, there is also no chess timer for taking turns [27].

- **Gyro-Accelerometer based control of a robotic arm using AVR microcontroller:** The primary focus on this paper is the usage of the accelerometer sensor on a robotic arm to perform a human-like operation such as the capability to grab an object. There is a gesture-based recognition for the six degrees of the freedom robot arm. "An artificial algorithm used to evaluate all gesture data which helps to train the robotic arm. The most popular Kalman filter used to find out the exact position of the human arm more accurately" [28]. The authors in this paper say that the control strategy is a lot easier than a joystick. One of the main advantages is that this system makes it more stable and synchronous response [28].
- **Human hand tracking using MATLAB to control Arduino based robotic arm:** In this paper, the human hand is tracked using the robotic hand. The robotic hand is controlled by the human hand, which is done by using an image processing technique to recognize different colours with the different axis of the hand. "The main aim behind this approach to program a robotic arm, so that it should be controlled by human hand and will reach the locations where human will not be able to reach and do the given task by direct interfacing with human hand" [29]. This robotic hand is also useful for the disable people as it can listen to the instructions of the human hand. For the hand detection database is not required as it is a real-time process [29].

- **Managing Robot Kinematics Based on Arduino Controllers Using a Unity System:** This paper investigates on managing robot kinematics based on Arduino controllers using a Unity system [30]. "To realize these goals, the appropriate mathematical tools for its management was applied, and the software to control the system operation was written using C# programming language. The software solution on the operation of the robot arm can be adapted for warehouse operations" [30].
- **Computer Vision Based Object Grasping 6DoF Robotic Arm using Picamera:** This paper aims to design a robust robotic arm that can perform multiple tasks which can further be controlled by Arduino mega 2560 microcontroller [31]. "The aim of the project is to focus all axes of the manipulator to lift, carry and unload the objects at the desired location. This requires a precise drive motion control that incorporates electric motors as a drive system. Further experiments are done to implement a camera-based 3D vision system integrated with a computer vision algorithm to recognize object deformation and spatial coordination to control the deviation from the initial training" [31].
- **A Hybrid Switched Reactive-Based Visual Servo Control of 5-DOF Robot Manipulators for Pick-and-Place Tasks:** This paper addresses the problem of visual servo control of a robotic servo arm which can pick and place objects. A new hybrid switched reactive-based visual servo controller is used to handle this problem efficiently. "The proposed structure is like the logic-based approaches, but the requirements of the fuzzy modelling and the fuzzy-rule based learning are omitted. To achieve this, a novel hardware-aware reactive function approach is

presented to directly map an image position error vector to a desired end-effector velocity vector under the consideration of hardware limitations" [32]. One of the main advantages of using this system is having robustness and effectiveness of the system. The authors presented five degrees of freedom for robot manipulator for pick and place objects. For grasping and picking the objects, a novel image-based object orientation estimation algorithm that estimates the object with the real-time performance [32].

- **Arduino based Battlefield Assistive Robot:** This paper starts with a discussion on the military field on how robots are even trained to defeat enemies. The radio frequency modules were used for wireless remote signals to control motors and actuators of the robot system. In addition to this, night vision monitoring is also added. A GSM module and metal detector also added to detect any bombs. The robot can also pick and drop objects. Despite its amazing features, there are certain drawbacks such as RF communication link can work up to 8 meters, which is not safe for a blast radius [33].
- **Bluetooth Communication Controlled Robot Based on Gesture Recognition:** This paper primarily focuses on gesture recognition tool that could control the robot movement and gripper to grip the object using a Bluetooth module through an Android application where the user can tilt the smartphone which then provides the gesture to control the robot, Which is achieved by using the smartphone's accelerometer, processor and Bluetooth. One of the limitations that the robot face is working on the rough surface and that might not provide the smooth results [34].

- **Wizard Chess: An Autonomous Chess Playing Robot:** The authors in this paper developed an autonomous chess-playing robot that could recognize all the chess boards, generating and executing moves. Wizard chess is low-cost and feasible. The problems faced are caused by the off-centre placement of the piece by the mechatronic system. Moreover, this error was solved by using a PD controller. There were also errors encountered during the complete game, which were caused by serial communication between the PC and Arduino [35].

Chapter 3

This section will describe the methodology that has been adopted to implement the Reactive Chess Robot Arm. This chapter will cover the detailed explanation of the method that was carried out for the implementation of the project.

3.1 System Architecture

The main objective of this project is to design a robot arm that can pick and place objects without using the Internet. Figure. 3.1 Provides a better understanding of the proposed system. The 3-layered architectural framework in IOT provides efficient and effective functionalities for the system, which leads to the developed structure of the system. Each layer performs some set of interrelated tasks. The 3-layers are perception layer, transport layer, and application layer. The first layer, called the perception layer, which also known as the sensing layer, is responsible for the integration of hardware and the collection of data from the sensors. It supports sensing technologies like WSN (Wireless Sensor Network) and RFID. The second layer called the network layer, which implements the support services for securing the data transfer over the sensor networks. It uses network technologies like 3G, Wi-Fi, Bluetooth, etc. It handles the functionalities like data aggregation and addressing schemes. The last layer called the application layer, which is responsible for the delivery of services. It also provides intelligent tools that are necessary for processing the information.

3.2 System Visualization

The visualization of the system has been explained with the following diagrams.

3.2.1 Flow Chart for Arduino

Flowchart diagram depicts a process of the proposed system and communicates complex processes in clear, easy-to-understand. The flowchart diagram is shown below in Fig. 3.2 illustrates the process of collecting data from the sensor and storing it using Arduino microcontroller. The HC-05 Bluetooth modules are used to establish communication between the robot arm and the Android application. The data sensed are read from the analog pins, and the baud rate is set to 9600 for serial communication. Once the Bluetooth communication is established, the user can control the servo motors of the robot arm to control its movement to pick and place chess pieces.

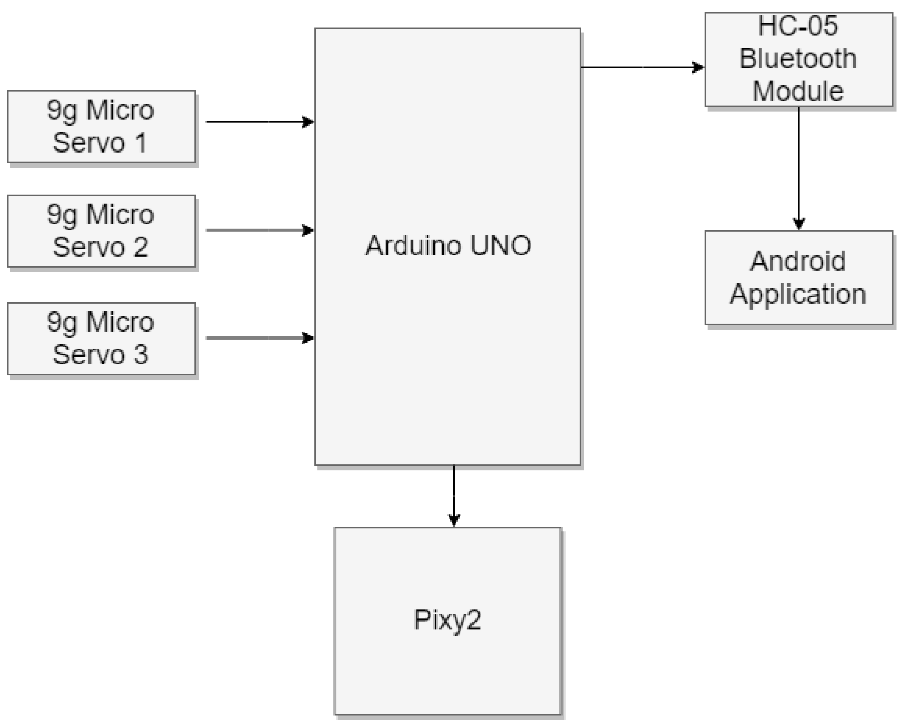

Figure 3.1 Block diagram of the proposed system

3.2.2 Flow Chart for Android application

The flowchart diagram shown in Fig. 3.3 depicts the working of the Android application in controlling the robot arm. Once the application is started, it asks the users permission to start Bluetooth. All the available Bluetooth devices are shown in the list when the desired device is selected, the connection is established. Now, the user can control servo motors of the robot arm using the slider in the Android application. There are three sliders to control the respective three servo motors: Base movement (Left/Right), Arm movement (Up/Down) and Gripper movement (Open/Close).

3.2.3 Flow Chart for Pixy2 Camera

The flowchart diagram shown in Fig. 3.3 depicts the working of the pixy2 camera concerning robot arm. The pixy2 camera is mainly used for object tracking and detection. The connection is established using the Arduino microcontroller. The baud rate here is set to 115200 for serial communication. Using the PixyMon UI, the signatures can be set for the unique objects; once the object is detected, the Pixy2 can track the object.

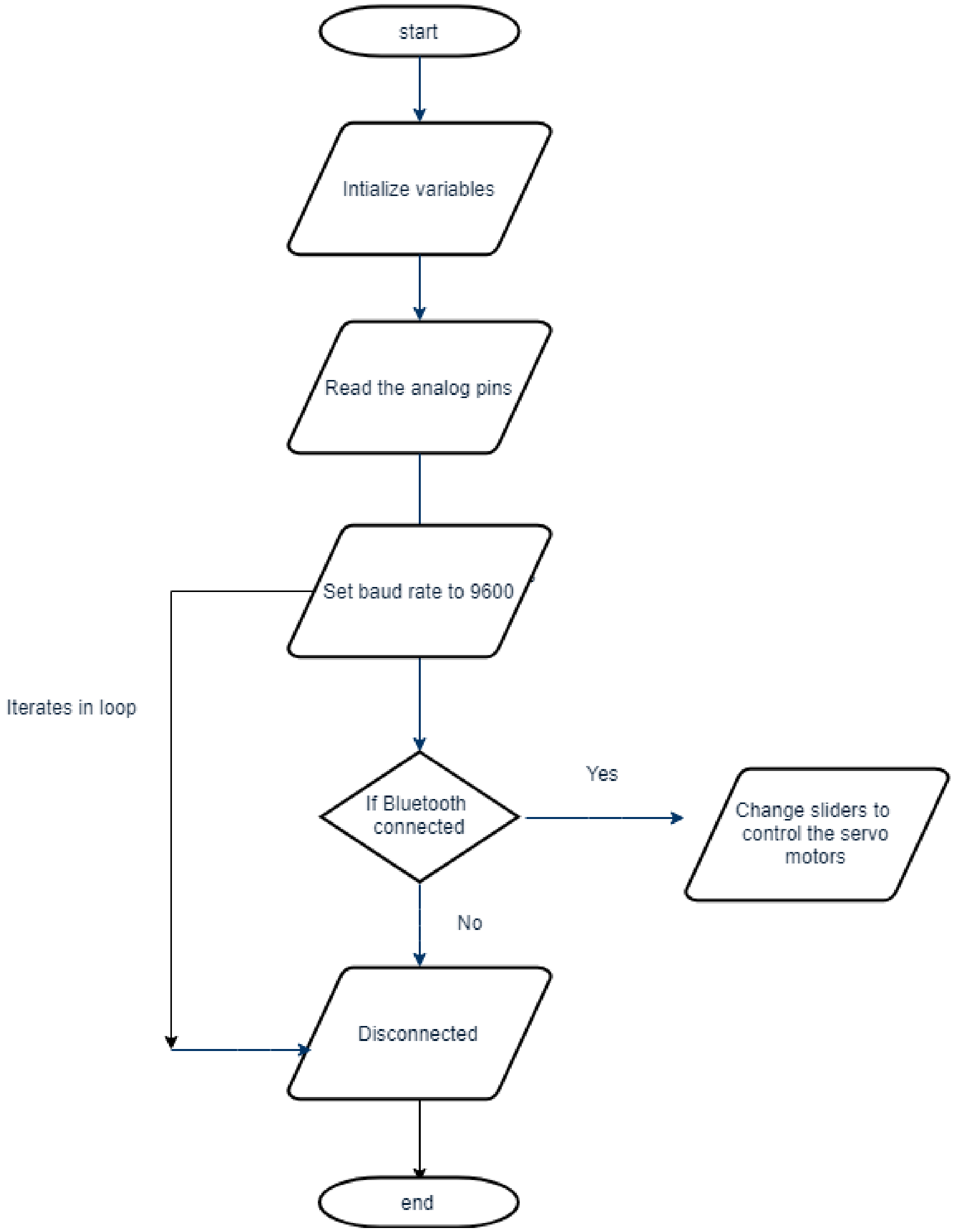

Figure 3.2 Flow chart for Arduino

3.2.4 Deployment Diagram

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to

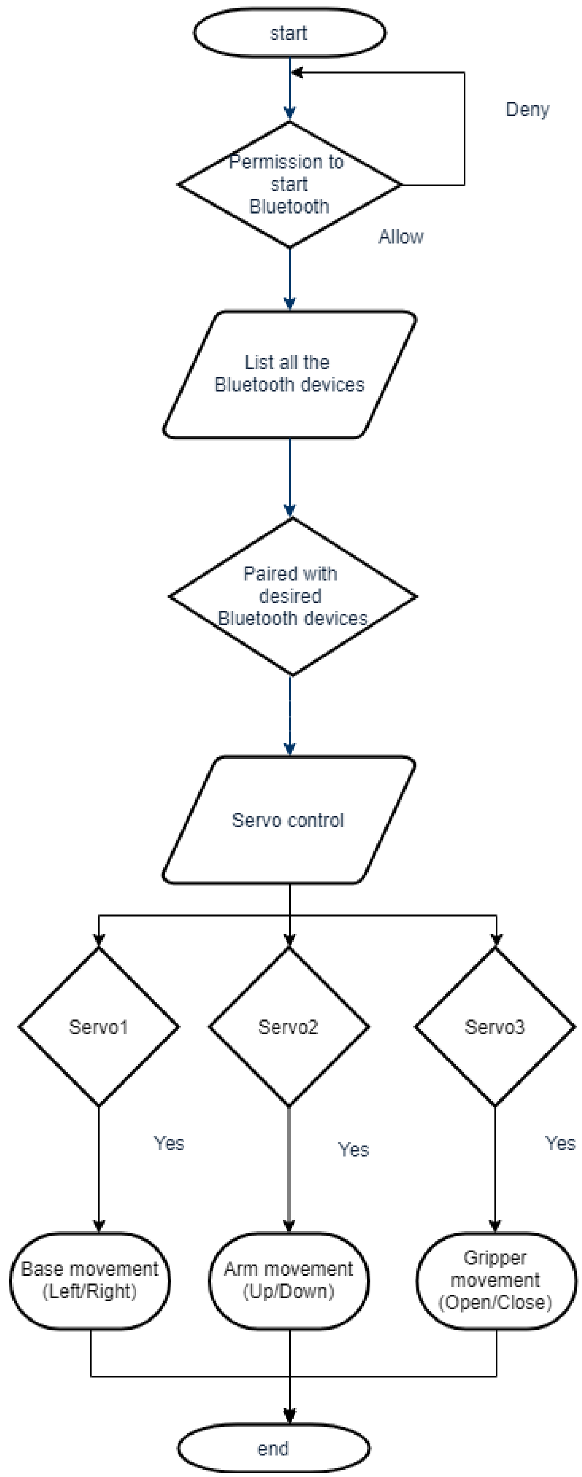

Figure 3.3 Flow Chart for Android application

describe the static deployment view of an order. Deployment diagrams consist of nodes and their relationships [37]. Here, the system can is deployed based upon the automation procedures.

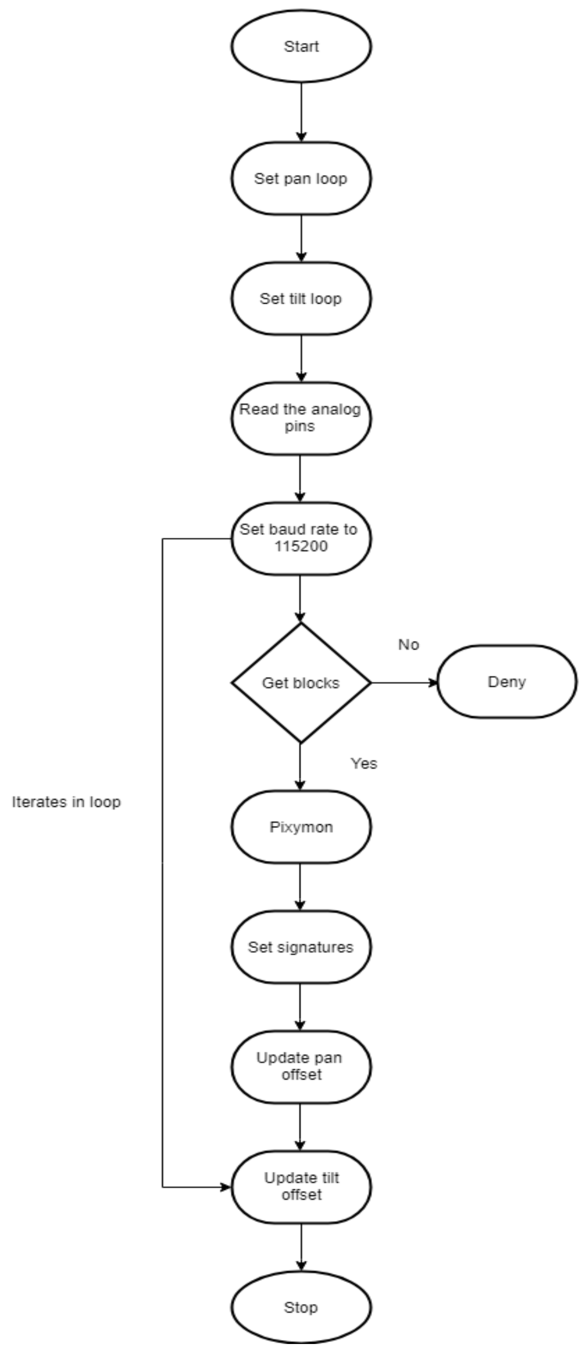

Figure 3.4 Flow chart for Pixy2

The sensors and servo motors are the nodes in the system, the Arduino, Bluetooth module and internet acts as the components of the system, and the Android application acts as the interface of the system.

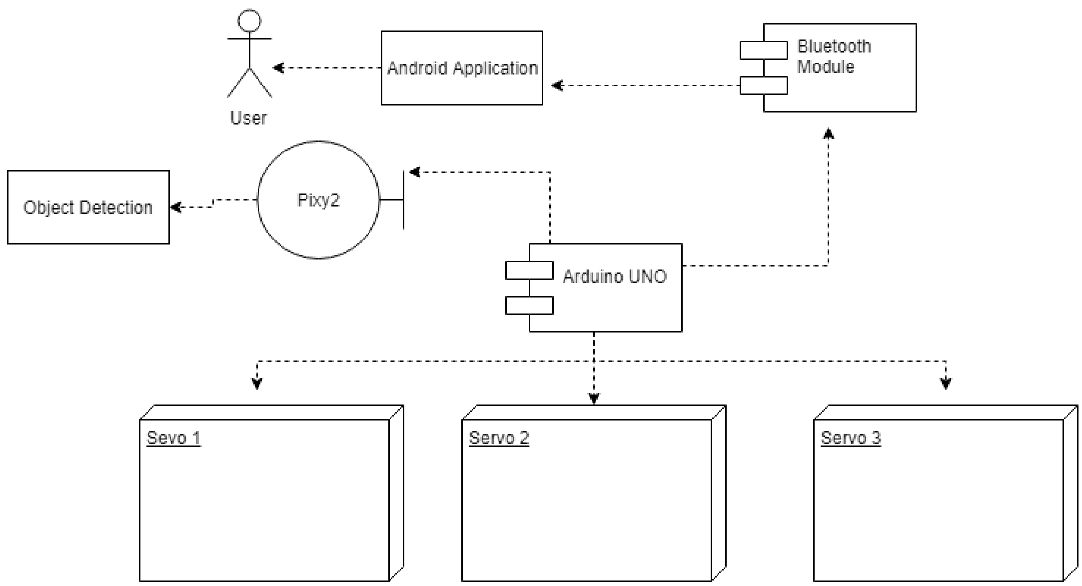

Figure 3.5 Deployment Diagram for the proposed system

3.3 UML Diagrams

Three Universal Modeling Language (UML) type diagrams that were used are the activity, sequence, and use case diagrams. UML's representation supersedes the flowchart because it makes provision for many different viewpoints on a system.

3.3.1 Component Diagram

UML Component diagrams are used in modeling the physical aspects of object-oriented systems that are used for visualizing, specifying, and documenting component-based systems and for constructing executable systems through forward and reverse engineering. Component diagrams are essentially class diagrams that focus on a system's components that often used to model the static implementation view of a system [38].

Arduino.ino: It contains the necessary coding required to control the servo motors for chess reactive robot arm which therein controls the movement of the robot arm to pick and place the chess pieces.

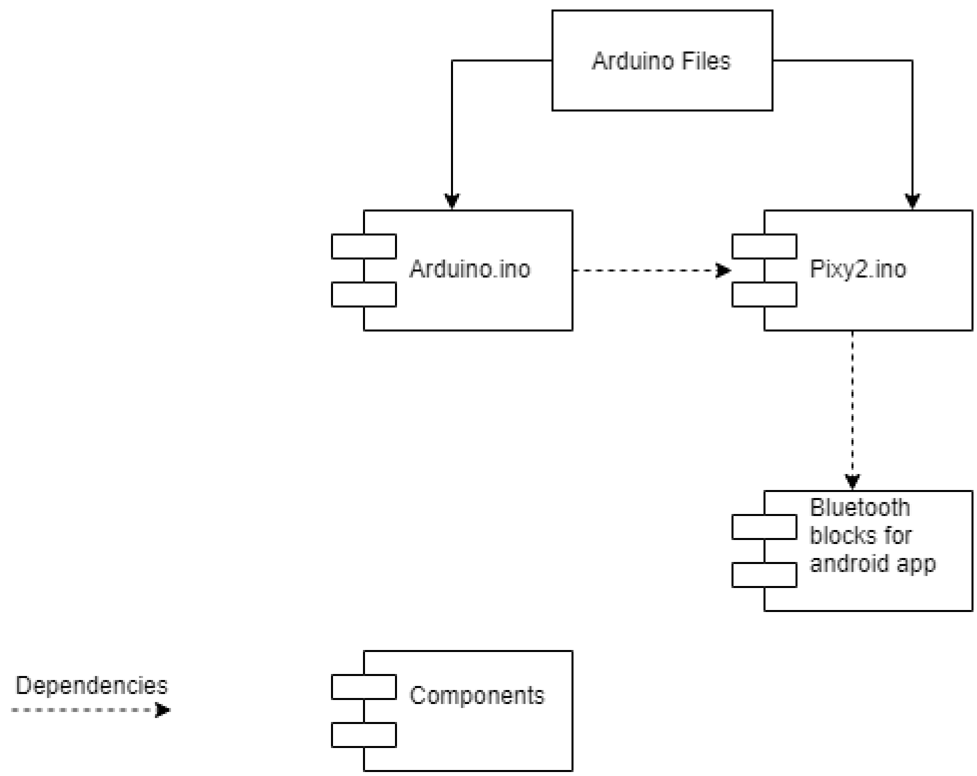

Figure 3.6 Component Diagram

Pixy2.ino: This Arduino file contains code to detect and track the object by setting the signature to the object. It uses the colour combination to track the objects.

Android App: This application is created using MIT app inventor drag and drop blocks. The app contains three sliders to control the movement of each servo motors, and it needs Bluetooth communication to establish the connection.

3.3.2 Use Case Diagram

The use case diagram is the description of the system's behaviour from a user's viewpoint. This diagram is a valuable aid during the analysis as developing use cases helps to understand requirements. Figure 3.7 is a Use Case diagram for the proposed system. There are four use cases for the robot arm: Bluetooth connects/disconnect, servo control, set signatures, detect objects and the user who controls the Robotic Arm.

Bluetooth Connection: This use case is for setting up the Bluetooth connection between the Android application and the Robotic Arm. The user can connect to the Bluetooth module and can select from the list of devices to establish a connection. Once the connection is made, the user can control the movement of the servos.

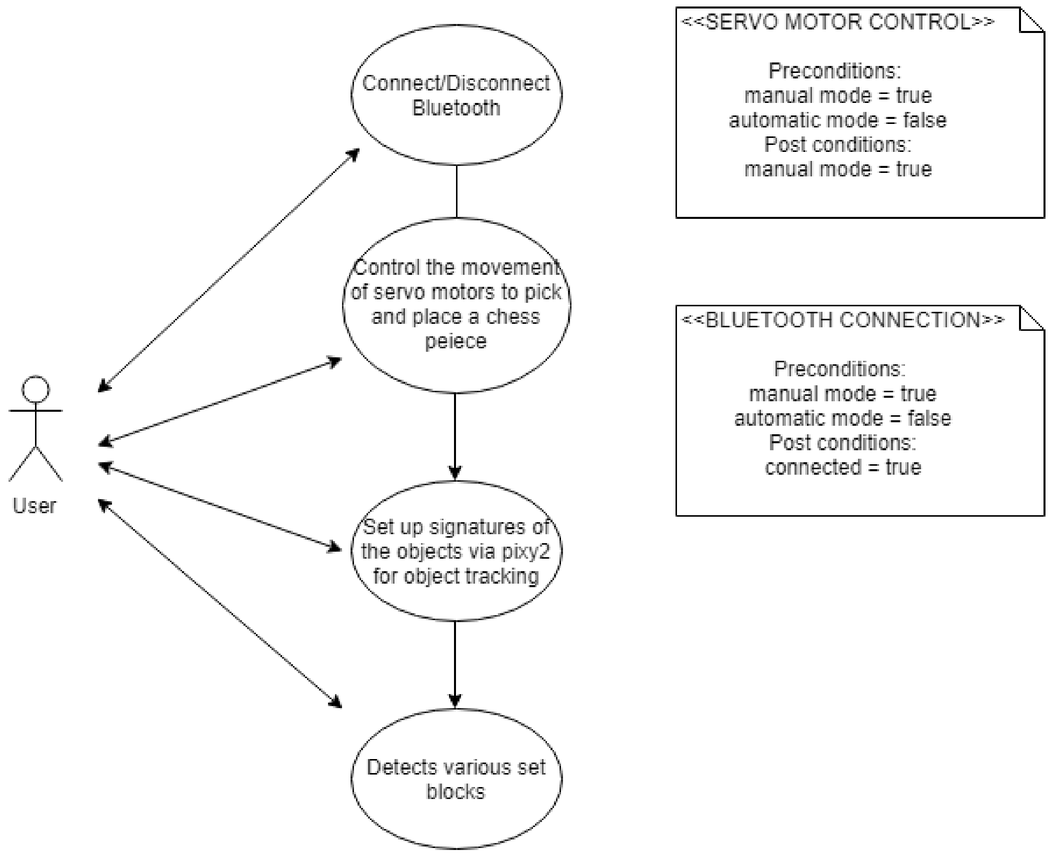

Figure 3.7 Use Case Diagram for the proposed system

Control Servo: From the data received from Arduino, the servo motors power up and be ready to provide the movement. Three micro 9g servos are used to control the movement of the Robotic Arm.

Set Signatures: Pixy2 camera is used for object tracking and detection. The signatures can be set for the objects using PixyMon. The set objects will be tracked by the Pixy2 Cam.

3.3.3 Sequence Diagram

The sequence diagram describes how the objects in the system interact over time. The objects used are, Sensors, Arduino UNO, Servo motors, HC-05 Bluetooth module, Android application, Pixy2 and PixyMon.

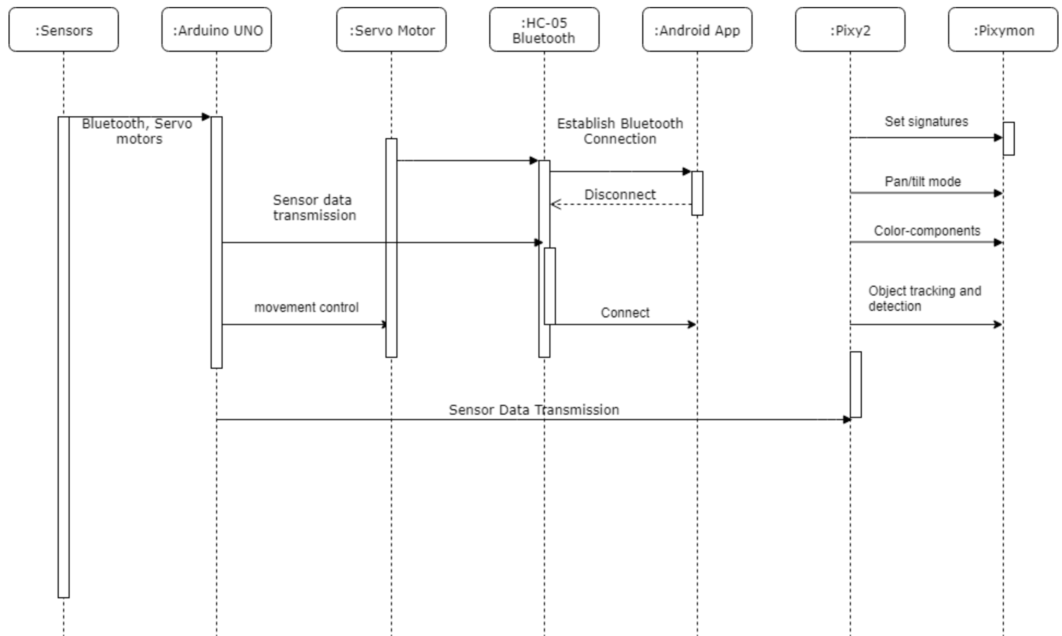

Figure 3.8 Sequence Diagram for the proposed system

The actions in chronological order are as shown in Figure 3.8. The data sent to the servo motors via Arduino and the servo motors can be controlled using an Android application. Android application is used to process and visualize sensed data. The communication between the Android application and the Servo motors (Robotic arm) is established with the help of the Bluetooth module. The following are the detailed explanation of the actions shown in the sequence diagram.

Servo Motors: The data is transmitted from the Arduino microcontroller to the servo motors to provide the movement of the servo motors. The 9g micro servos are used for the controlling of the robotic arm.

HC-05 Bluetooth: The Bluetooth module builds the connection between the servo motors and the android application.

Android application: This application lets the user control the movement of the servo motors. Therein the robot arm can pick up the desired objects.

Pixy2: Pixy2 camera here is used for object detection and tracking. The data from pixy2 is transmitted to the Pixy monitor.

PixyMon: The data is transmitted from the pixy2 tracks the object using corrected colour components.

3.3.4 Activity Diagram

The activity diagram shown in Figure 3.9 depicts the sequential flow of activities of the use case; it was used to model actions that are done when an operation is performed as well as the results of those actions.

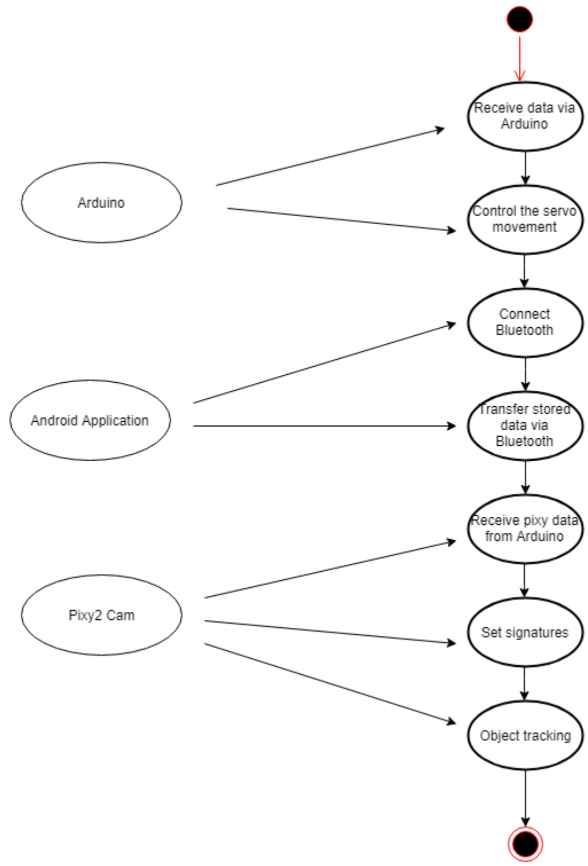

Figure 3.9 Activity Diagram of the proposed system

The model starts with the activity of the gathering the data from the Arduino. The collected data is then stored and transferred to the Android application, which is used to control the movement of the servo motors.

Then the data received from Pixy2 cam sets signatures to the objects, and the data is further sent to the PixyMon, which displays the result. Finally, the camera tracks the signature object.

Chapter 4

4.1 Overview

The proposed model has been implemented using Arduino UNO microcontroller. HC05 Bluetooth module has been used for Bluetooth communication. Android application has been developed using MIT App Inventor 2, which helps to control the movement of the Robotic Arm and therein picks and places the chess piece. A pixy2 camera is used to detect and track the objects.

4.2 Hardware Requirements

4.2.1 Arduino UNO

Arduino UNO microcontroller is based on ATmega328 datasheet, which has 14 digital input and output pins. Six pins are used for Pulse width modulation (PWM) outputs, and rest are used for analog inputs. It also has a 16 MHz resonator, a USB connection, a power jack, an ICSP (in-circuit serial programming) header, and a reset button. The microcontroller board can be connected to a computer with a USB cable or an adapter using a power jack. Arduino Uno board uses the Atmega16U2 programmed as a USB-to-serial converter. It helps to communicate directly with the PC or computer. Arduino microcontroller is used for this project since it has inbuilt ADC, which helps to give servo output in analog form. This board is also simple for programming; it does not need any external programmer or burner to burn the program in microcontroller since it has 32 kb flash memory so we can save our program as well as we can change the code according to our requirements.

Arduino pins operate at 5V, and specific functions of these pins are listed:

- Pins 0 and 1: Rx /Tx pins are used to receive and transmit serial data.
- Pins 2 and 3: These pins are for an external interrupt which will be triggered when there is a change in value.

Figure. 4.1 Arduino UNO microcontroller board [40]

- Pins 3, 5, 6, 9 and 11: These are PWM pins, and they give an 8-bit PWM output.
- Pins 10, 11, 12 and 13: These are used for Serial Peripheral Interface (SPI) communication.
- LED Pin 13: LED is on if the Pin 13 is high; otherwise, it will be off.
- AREF: provides a reference voltage for analog inputs.
- Reset Pin: it resets the microcontroller.

4.2.2 HC-05 Bluetooth Module

HC-05 Bluetooth module follows Serial Port Protocol (SPP) and easy to use. It is designed for wireless serial connection setup. It's easier to interface with controller or PC since it communicates via serial communication. It also provides a switching mode for master and slave modes. It operates on 5V DC voltage. There are various applications for the Bluetooth module, such as communication between laptops, smartphones, Robots, and Automation. Figure 4.2 shows the pin definition of the Bluetooth module.

Figure 4.2 HC-05 Bluetooth module [41]

Pin	**Description**	**Function**
VCC	+5V	Connect to +5V
GND	Ground	Connect to Ground
TXD	UART_TXD, Bluetooth serial signal sending PIN	Connect with the MCU's (Microcontroller etc.) RXD PIN.
RXD	UART_RXD, Bluetooth serial signal receiving PIN	Connect with the MCU's (Microcontroller etc.) TXD PIN.
KEY	Mode switch input	If it is input low level or connect to the air, the module is at paired or communication mode. If it's input high level, the module will enter to AT mode.

Table 4.1: Pin Definition of the HC-05 Bluetooth module

Hardware Connections:

Table 4.2 represents the hardware interfacing between the HC-05 Bluetooth module to the Arduino UNO. The Bluetooth module helps to establish a connection between the Arduino and the servo motors.

HC-05 Bluetooth module	**Arduino UNO**
VCC	3V3
GND	GND
TXD	A10
RXD	A11

Table 4.2 Hardware interfacing between HC-05 Bluetooth module and Arduino UNO

4.2.3 SG90 9g Micro Servo Motors

The next hardware, which is a drive system of a robotic arm is a servomotor. "Servo motors have been around for a long time and are utilized in many applications. They are small but pack a big punch and are very energy-efficient" [39]. Because of these features, they can be used to operate remote-controlled or radio-controlled toy cars, robots and airplanes. Servo motors are also used in industrial applications, robotics, in-line manufacturing, pharmaceutics and food services" [39].

Figure 4.3 SG90 9g Micro Servo [42]

To demonstrate the task, I have selected a small size servo named SG90 9g Micro Servo. It is a standard servo used in many small-scale applications; the detailed information is explained in the table 4.2. This Servo can rotate approximately 180 degrees (90 in each direction) and works just like the standard kinds but smaller.

Dimensions	**Specifications**
A (mm)	32
B (mm)	23
C (mm)	28.5
D (mm)	12
E (mm)	32
F (mm)	19.5
Speed (sec)	0.1
Torque (kg-cm)	2.5
Weight (g)	14.7
Voltage	4.8 - 6

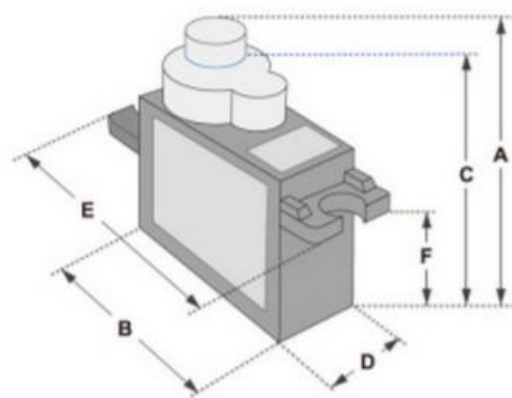

Table 4.3 Dimensions and specification of SG90 9g servo motor

Figure 4.4 SG90 Micro Servo Motor

Hardware Connections:

Relevant to the project, table 4.4 represents the hardware interfacing between the Servo 1 and Arduino UNO. Servo 1 represents the base servo on the chess board which provides the movement to right or left of 180 degrees approximately, 90 degrees in each direction.

Servo 1	**Arduino UNO**
Signal Input	A3
Red Input	VCC (Breadboard)
Brown Input	GND (Breadboard)

Table 4.4 Hardware interfacing between Servo 1 and Arduino UNO

Table 4.5 represents the hardware interfacing between servo2 and Arduino UNO. Here servo 2 depicts the arm movement, which can move up and down by using the corresponding slider in the Android application.

Servo 2	**Arduino UNO**
Signal Input	A4
Red Input	VCC (Breadboard)
Brown Input	GND (Breadboard)

Table 4.5 Hardware interfacing between Servo 2 and Arduino UNO

Similarly, table 4.6 also represents the hardware interfacing between servo3 and Arduino UNO. Here servo 3 depicts the gripper movement, which can open and close using the corresponding slider in the Android application. The gripper is used for grabbing the chess piece.

Servo 3	**Arduino UNO**
Signal Input	A5
Red Input	VCC (Breadboard)
Brown Input	GND (Breadboard)

Table 4.6 Hardware interfacing between Servo 3 and Arduino UNO

4.2.4 Pixy2 Cam

Pixy2 is an object detection and tracking camera module. This can be directly connected to the Arduino UNO microcontroller. The objects are set as a signature; the user can also train pixy2 to track an object. These objects can be set as signatures which, can be set using PixyMon which is an application that runs on PC. PixyMon can configure the pixy2 camera and allows the user to see what pixy cam sees. Table 4.2 shows the specifications of the camera. Pixy can send block information to Arduino at 1

Mbits/second, which means Pixy can send more than 6000 detected objects per second or 100 detected objects per frame (Pixy can process 60 frames per second).

Figure 4.5 Pixy2 [44]

Processor	NXP LPC4330, 204 MHz, dual core
Image sensor	Aptina MT9M114, 1296x976
Lens field of view	60 degrees horizontal, 40 degrees vertical
Power consumption	140 mA typical
Power input	USB input (5v)
RAM	264K bytes
Dimensions	1.5”x1.65”x0.6”
Flash	2M bytes

Table 4.7 Specifications of the Pixy2 [43]

Hardware Connections:

Connecting Pixy2 to Arduino is quite simple, use the supplied Arduino cable to connect Pixy to the Arduino.

4.2.5 Jumper Wires

Male to Male jumper wires have been used for connecting all the components explained above with the Arduino board. Figure 4.6 represents the jumper wires used in building this model.

Figure 4.6 Jumper wires (Male to Male)

4.2.6 9V Power supply

As Arduino UNO needs a continuous power supply to turn on, 9v batteries use circuits to control power supply. A normal USB cable is used for making a complete circuit between Arduino UNO and power supply. Thus, it provides 9V power for the servo motors.

Figure 4.7 9V Battery power supply

4.3 Software Requirements

4.3.1 Arduino IDE

Arduino IDE is a software application which helps computer programmers for software development by providing extensive facilities. It consists of:

- A source code editor
- A compiler or an interpreter
- Build automation tools
- A debugger

Arduino software has a text editor to write code, a text console, a series of menus, a toolbar with buttons to use standard functions and a message area. Arduino Uno board is used to upload programs and to communicate with them.

- Get an Arduino board and USB cable
- Download the Arduino environment
- Connect the board- Connect the Arduino board to your computer using the USB cable. The green LED labeled as PWR will be on.
- Install the drivers.
- Launch the Arduino application- Open any program from examples
- Select your board; Select your serial port and Select the serial device of the Arduino board from the Tools -> Serial Port menu
- Upload the program

Figure 4.8 shows how the sketch looks like in Arduino IDE. A sketch is a place where the coding takes place.

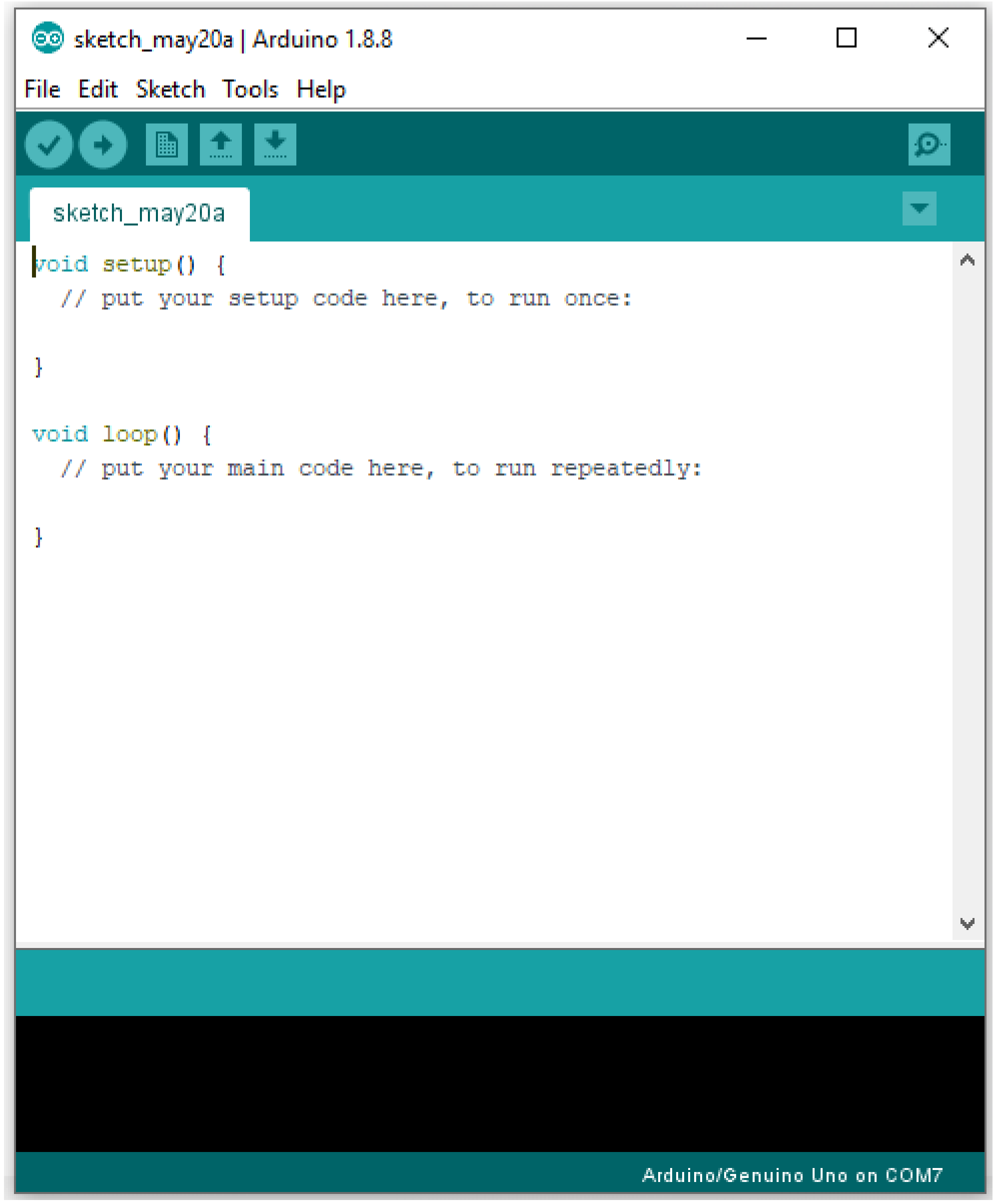

Figure 4.8 Arduino IDE

4.3.2 MIT App Inventor 2

Android applications can be built in the browser using App Inventor since it is a cloud-based tool. It allows you to make your own mobile apps using a blocks-based programming language, the blocks editor was implemented in Java on top of the OpenBlocks blocks framework. The Designer window is implemented in Java using the Google Web Toolkit (GWT), which is translated to JavaScript so that it, too, runs in the browser.

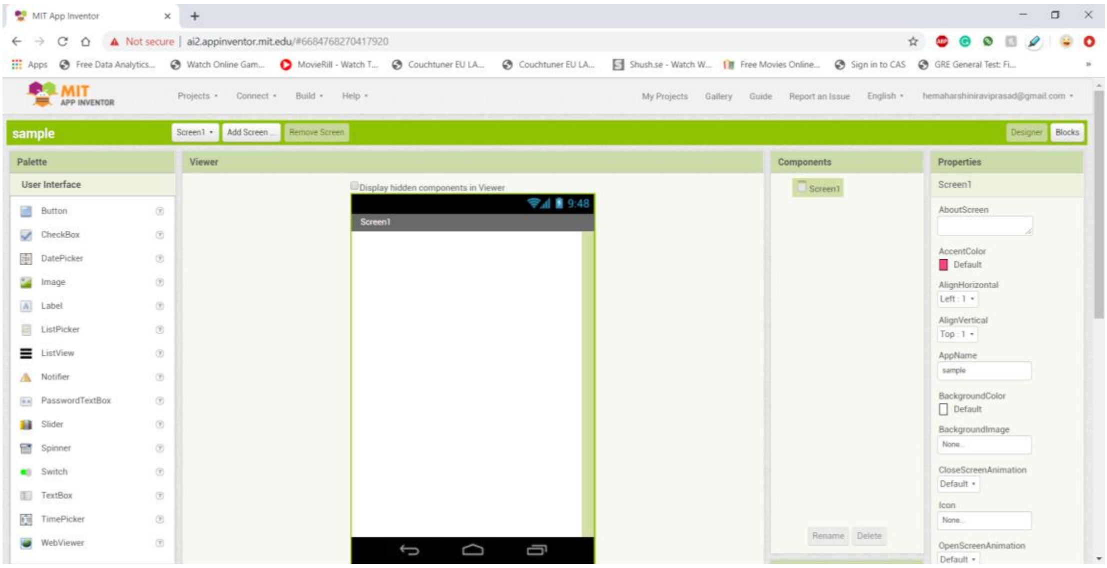

Figure 4.9 MIT App Inventor 2

4.3.3 PixyMon

"PixyMon is an application that allows you to configure Pixy and see what it sees. It runs on several different platforms including Windows, MacOS and Linux, as well as other smaller embedded systems like Arduino, Raspberry Pi and Beagle Bone Black" [46].

Buttons: These are the most common PixyMon actions, conveniently located at the top of the main window.

Video window: This is where PixyMon renders various types of raw or processed video.

Command/status: This is where status messages are displayed and where Pixy commands can be typed in.

Stop/resume: Pressing this button stops the video that's being rendered in the video window. This is useful when grabbing a frame or typing commands into the command/status window. Press this button again to resume video.

Default program: Pressing this button runs the default program, which is the program that executes when Pixy powers up. It's typically the program that does all of the

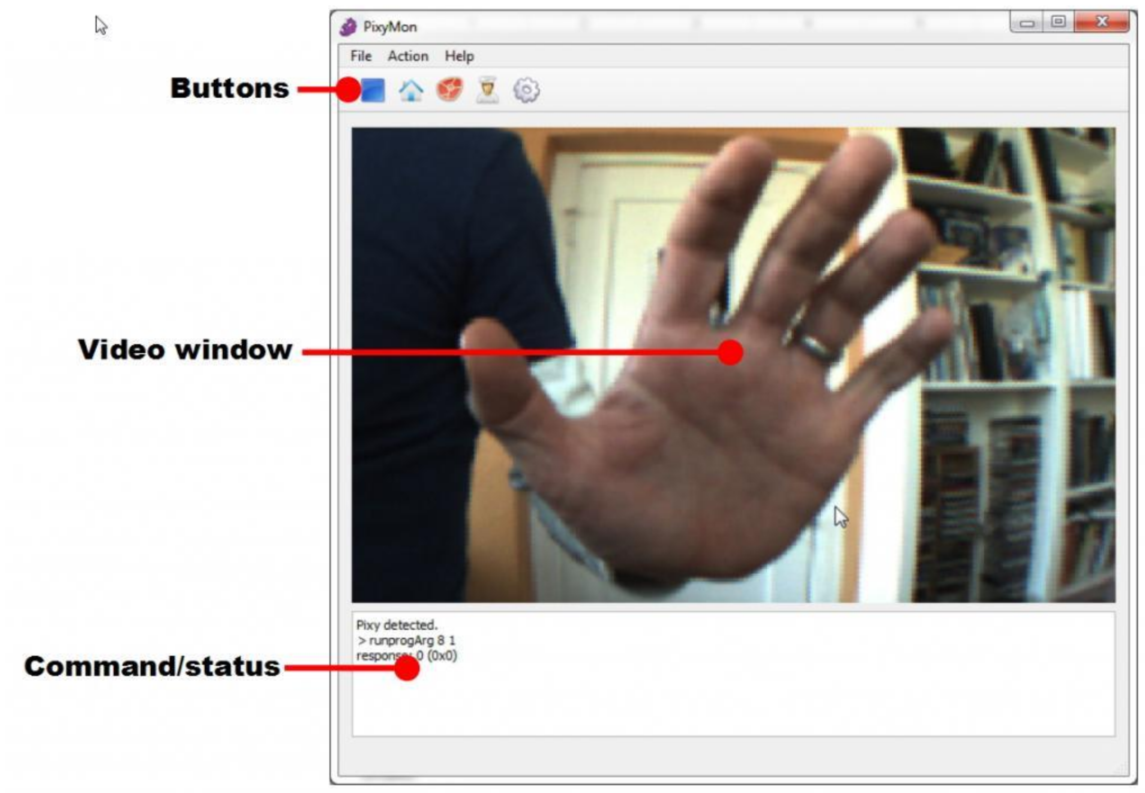

Figure 4.9 PixyMon [46]

processing on Pixy and outputs the results (e.g. detected objects) through one of Pixy's serial ports.

Raw video: Pressing this button displays raw, unprocessed video. This is useful for adjusting focus and camera brightness.

Configure: Pressing this brings up the Configure Dialog, which contains various configurable parameters for Pixy and PixyMon.

4.4 Experimental Setup

In this section, connections between the microcontroller Arduino Uno, HC-05 Bluetooth module, and the Pixy2 has been explained step by step.

4.4.1 HC-05 Bluetooth Module

The HC05 Bluetooth module can be powered from 3.6 to 6 volts since it has a breakout board with a voltage regulator. It has 6 pins, and four of them connected with Arduino to establish Bluetooth communication. The connections are shown below in a table 4.8 and a schematic diagram.

	TXD	RXD	GND	VCC
Arduino Uno	TX (10)	RX (11)	GND	5V

Table 4.8 Bluetooth module connections with Arduino

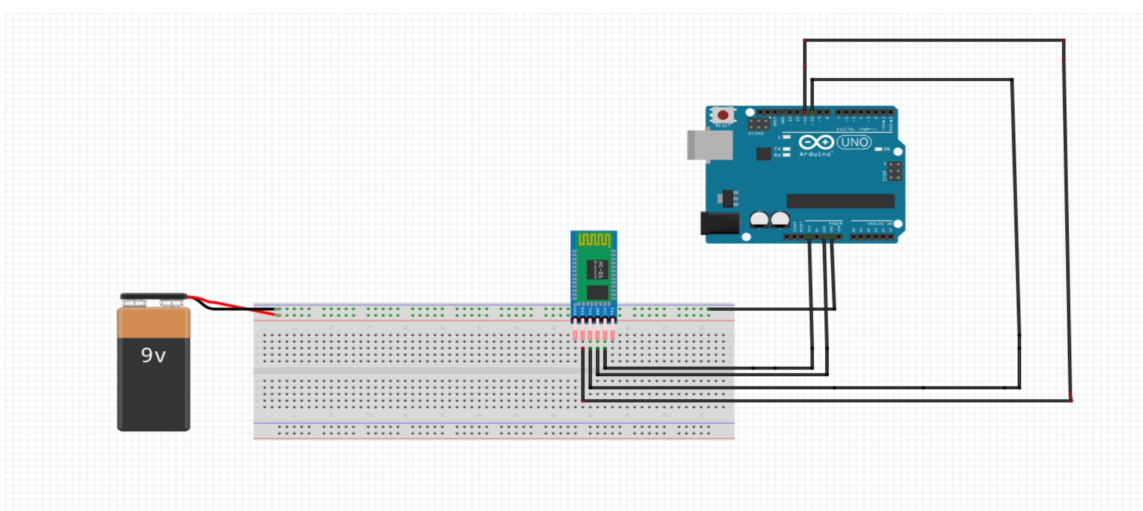

Figure 4.10 Fritzing representation of the Bluetooth module

4.4.2 SG90 9g Mini Servo Motors

Relevant to the project, table 4.9 represents the hardware interfacing between the Servo 1 and Arduino UNO. Servo 1 represents the base servo on the chess board which provides the movement to right or left of 180 degrees approximately, 90 degrees in each direction. Figure 4.11 represents the fritzing of the servo motors.

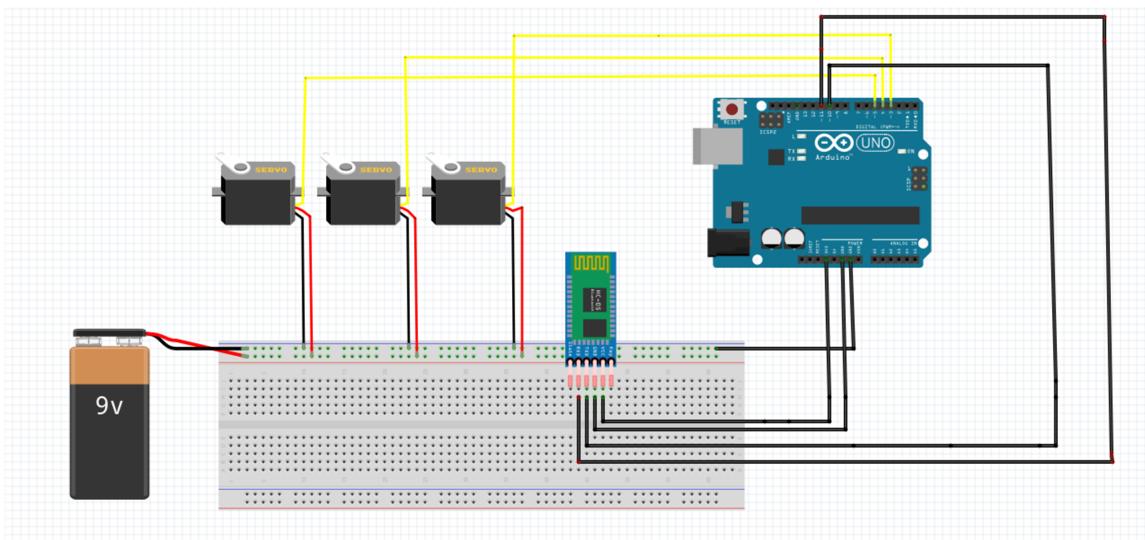

Figure 4.11 Fritzing representation of the Servo Motors

Servo 1	**Arduino UNO**
Signal Input (Orange)	A3
Red Input	VCC (Breadboard)
Brown Input	GND (Breadboard)

Table 4.9 Hardware interfacing between Servo 1 and Arduino UNO

Table 4.10 represents the hardware interfacing between servo2 and Arduino UNO. Here servo 2 depicts the arm movement, which can move up and down by using the corresponding slider in the Android application.

Servo 2	Arduino UNO
Signal Input	A4
Red Input	VCC (Breadboard)
Brown Input	GND (Breadboard)

Table 4.10 Hardware interfacing between Servo 2 and Arduino UNO

Similarly, table 4.11 also represents the hardware interfacing between servo3 and Arduino UNO. Here servo 3 depicts the gripper movement, which can open and close using the corresponding slider in the Android application. The gripper is used for grabbing the chess piece.

Servo 3	Arduino UNO
Signal Input	A5
Red Input	VCC (Breadboard)
Brown Input	GND (Breadboard)

Table 4.11 Hardware interfacing between Servo 3 and Arduino UNO

4.4.3 Pixy2 Cam

The objects are set as a signature; the user can also train pixy2 to track an object. These objects can be set as signatures which, can be set using PixyMon which is an application that runs on PC. PixyMon can configure the pixy2 camera and allows the user to see what pixy cam sees. Pixy can send block information to Arduino at 1 Mbits/second, which means Pixy can send more than 6000 detected objects per second or 100 detected objects per frame (Pixy can process 60 frames per second). Figure 4.12 shows the fritzing representation of the pixy2 camera.

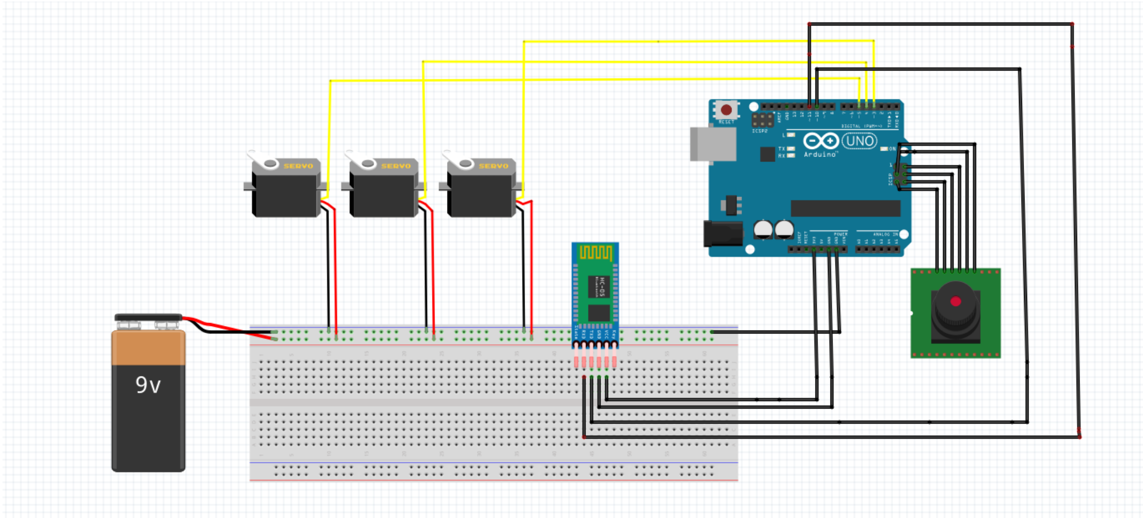

Figure 4.12 Fritzing representation of the Pixy2 Camera

4.5 Working of the Robotic Arm

4.5.1 Overview

In this section, the implementation of the Robotic Arm and the Android application has been explained. The mechanism for the robot arm using servo motors and Bluetooth is described along with the libraries used.

4.5.2 Prerequisites for Servo Motors and Bluetooth

The SoftwareSerial library is required to have multiple serial software ports to allow the serial communication on other digital pins of the Arduino, using software to replicate the functionality. It is possible to have multiple software serial ports with speeds up to 115200 bps. A parameter enables inverted signaling for devices which require that protocol.

The servo library allows an Arduino board to control servo motors. This library supports up to 12 motors.

4.5.3 Libraries used

Figure 4.13 represents the libraries, SoftwareSerial.h and Servo.h which used for setting up the Bluetooth communication and the functioning of servos.

```
#include <SoftwareSerial.h>

#include <Servo.h>

    SoftwareSerial bluetooth(bluetoothTx, bluetoothRx);
```

Figure 4.13 Libraries used for the Robot Arm implementation

4.5.4 Declaration and Initialization of variables

The sensor data is read from analog pins; two variables are used for temperature and pulse sensors. An instance for PulseSensorPlayground object has been created to use a pulse sensor. Threshold value to determine which signal to count as a beat and which to ignore has been set to 550. ****

```
//Variables
Servo myservo1, myservo2, myservo3;
byte serialA;

int bluetoothTx = 10;
int bluetoothRx = 11;
```

Figure 4.14 Code for variable Initialization

4.5.5 Setting up Serial Communication

The 'serial' variable is used to open serial communications and to wait for the port to open. The baud rate for serial has been set to 9600. SD card is initialized after checking whether the card is present in the adapter. Pulse sensor and 'rtc' for the timestamp, is initialized as shown in the code snippet below. The standard SD and SPI libraries are included, a ‘File’ object is created, and the chip select pin of SPI bus is set to 10. Initialization of SD card is done by calling SD.begin() function. ****

```
//Open Serial communications and wait for port to open:
Serial.begin(9600);
while(!Serial)
{
  ;//wait for serial port to connect. This is needed for activating USB port only
}

bluetooth.begin(9600);


if (bluetooth.available() > 2) {serialA = Serial.read();Serial.println(serialA);}
{
  //Configure the servopos by assigning variables to it
  unsigned int servopos = bluetooth.read();
  unsigned int servopos1 = bluetooth.read();
  unsigned int realservo = (servopos1 *256) + servopos;
  Serial.println(realservo);
```

Figure 4.15 Code for setting up serial communication

4.5.6 Reading Servo data

Reads the current angle of the servo. The position is set and can provide movement up to 180 degrees of freedom. Writes a value to the servo, controlling the shaft accordingly. On a standard servo, this will set the angle of the shaft (in degrees), moving the shaft to that orientation. On a continuous rotation servo, this will set the speed of the servo (with 0 being full speed in one direction, 180 being full speed in the other, and a value near 90 being no movement).The servo1 provides the base movement, left and right direction, servo2 provides the arm movement, up and down which helps the servo3, the gripper to open and close. The gripper helps to grip the chess piece.

```
if (realservo >= 1000 && realservo <1180){
int servo1 = realservo;
servo1 = map(servo1, 1000,1180,0,180);
myservo1.write(servo1);
Serial.println("servo 1 ON");
delay(10);

}

if (realservo >=2000 && realservo <2180){
  int servo2 = realservo;
  servo2 = map(servo2,2000,2180,0,180);
  myservo2.write(servo2);
  Serial.println("servo 2 On");
  delay(10);

}

if (realservo >=3000 && realservo < 3180){
  int servo3 = realservo;
  servo3 = map(servo3, 3000, 3180,0,180);
  myservo3.write(servo3);
  Serial.println("servo 3 On");
  delay(10);
}
```

Figure 4.16 Reading the servo data

4.5.7 Transfer Data via Bluetooth

The code snippet of the figure 4.17 shows that he bluetooth.begin command sets the data rate in bits per second. For communicating with the computer, 9600 baud rates has been used. Position of each servo is set after setting up the serial communication and reads the incoming serial data setting it to the initial position. If the data file is available, the contents of the file are written to the serial communication.

```
int bluetoothTx = 10;        //Bluetooth Transceiver intialization
int bluetoothRx = 11;        //Bluetooth Receiver initialization

SoftwareSerial bluetooth(bluetoothTx, bluetoothRx);


 bluetooth.begin(9600);

  if (bluetooth.available() > 2) {serialA = Serial.read();Serial.println(serialA);}
  {
    //Configure the servopos by assigning variables to it
    unsigned int servopos = bluetooth.read();
    unsigned int servopos1 = bluetooth.read();
    unsigned int realservo = (servopos1 *256) + servopos;
    Serial.println(realservo);
```

Figure 4.17 Transfer data via Bluetooth

4.5.8 Android Application Setup and Prerequisites

For android application, ‘BluetoothClient1’ and ‘Clock1’ libraries are used to setup Bluetooth communication and to get current date and time respectively. ‘Notifier1’ library is used to send a notice when Bluetooth is not enabled in the Android device. Bluetooth label status is initially set to ‘Disconnected’ when the screen is initialized. ‘Date’ and ‘Time’ labels are set with current date and time, as shown in the block’s snippet below.

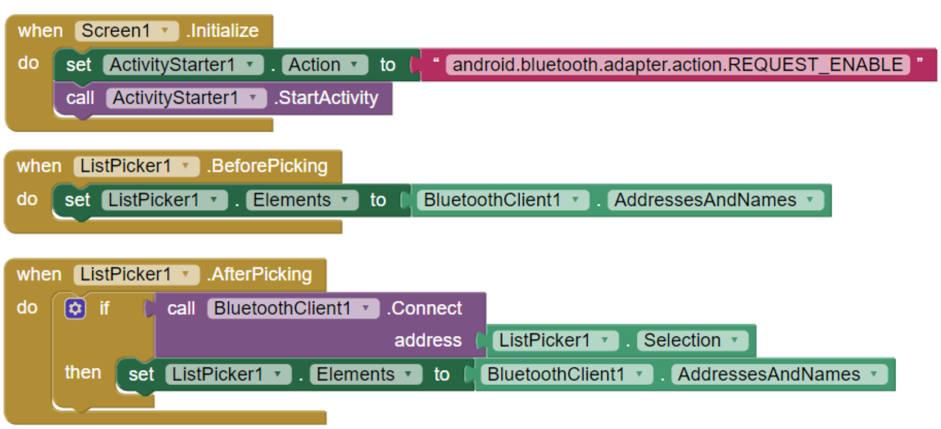

Figure 4.18 Initializing the Android application screen

4.5.9 Establishing Bluetooth Communication

All the Bluetooth devices visible are set as elements to a list picker. Once the device is selected, the Bluetooth connection is established, and the label status is set to connected. If the disconnect button is clicked, then the Bluetooth connection is disconnected, and label status is changed accordingly.

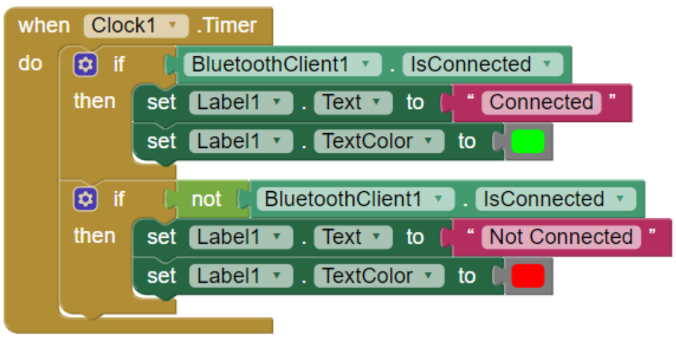

Figure 4.19 Establishing Bluetooth communication

4.5.10 Data communication via servo motors through Android application

The data is communicated to the servo motors through the Android application. The sliders in the Android application are used to control the movement of each servos. The maximum position for the servo1 is set to 1000, the maximum position of servo2 is 2000 and similarly the maximum position of sevo3 is 3000. Figure 4.20 represents the slider creation using the blocks in MIT app inventor.

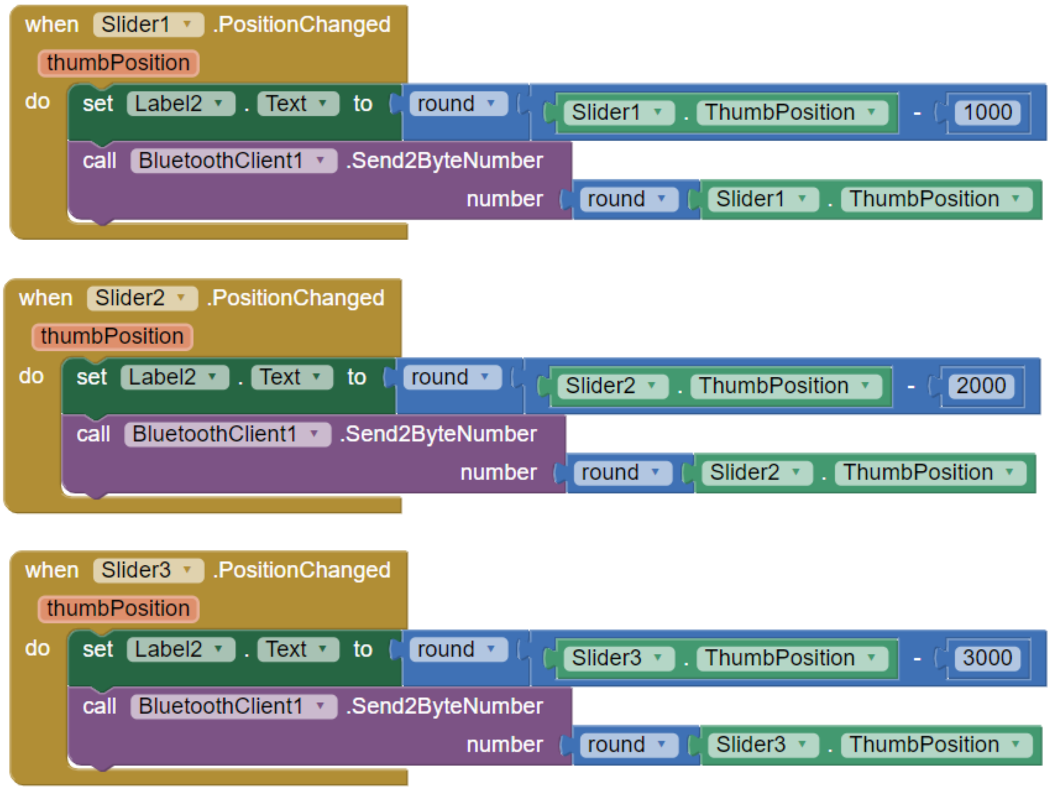

Figure 4.20 To control servo motors using the sliders in the Android application

4.6 Working of Pixy2

4.6.1 Overview

In this section, the implementation of the Pixy2 camera has been explained. The mechanism for the pixy2 while using Pixymon is described along with the libraries used.

4.6.2 Prerequisites for Pixy2

The most important method in the Arduino library is getBlocks(), which returns the number of objects Pixy has detected. The pixy2 library is required to have multiple serial software ports digital pins of the Arduino, using software to replicate the functionality. It is possible to have multiple software serial ports with speeds up to 115200 bps. A parameter enables inverted signaling for devices which require that protocol.

4.6.3 Libraries used

Figure 4.21 represents the libraries, pixy2 which used for setting up the the signatures of the object for tracking and detection.

```
#include <Pixy2.h>
#include <PIDLoop.h>

Pixy2 pixy;
```

Figure 4.21 Libraries used for Pixy2

4.6.4 Declaration and Intialization of variables

Figure 4.22 represents the declaration and initialization of the variables for Pixy2, init() should be called before your program communicates with Pixy, init() returns an error value (<0) if it fails and 0 (PIXY_RESULT_OK) if it succeeds. This variable contains the width and height of the frame in pixels. This field is updated upon calling init() and upon calling changeProg() as each program may have a unique frame size.

```
pixy.init();
```

```
static int i = 0;
int j;
char buf[64];
int32_t panOffset, tiltOffset;
```

Figure 4.22 Declaration and intialization of variables for Pixy2

4.6.5 Reading data

Pixy.cc is the instance of the Color Connected Components object associated with the color_connected_components program. The variable numBlocks is the number of blocks contained in the blocks member variable. The new data is then available in the blocks member variable. The returned blocks are sorted by area, with the largest blocks appearing first in the blocks array.

```
if (pixy.ccc.numBlocks)
{
  i++;

  if (i%60==0)
    Serial.println(i);

  // calculate pan and tilt "errors" with respect to first object (blocks[0]),
  // which is the biggest object (they are sorted by size).
  panOffset = (int32_t)pixy.frameWidth/2 - (int32_t)pixy.ccc.blocks[0].m_x;
  tiltOffset = (int32_t)pixy.ccc.blocks[0].m_y - (int32_t)pixy.frameHeight/2;

  // update loops
  panLoop.update(panOffset);
  tiltLoop.update(tiltOffset);

  // set pan and tilt servos
  pixy.setServos(panLoop.m_command, tiltLoop.m_command);
```

Figure 4.23 Reading data

4.6.6 PixyMon Pan and Tilt mode

setServos() sets the servo positions of servos plugged into Pixy2's two RC servo connectors. The servo values range between PIXY_RCS_MIN_POS (0) and PIXY_RCS_MAX_POS (1000). It returns an error value (<0) if it fails and 0 (PIXY_RESULT_OK) if it succeeds.

```
    // set pan and tilt servos
    pixy.setServos(panLoop.m_command, tiltLoop.m_command);

#if 0 // for debugging
    sprintf(buf, "%ld %ld %ld %ld", rotateLoop.m_command, translateLoop.m_command, left, right);
    Serial.println(buf);
#endif

  }
  else // no object detected, go into reset state
  {
    panLoop.reset();
    tiltLoop.reset();
    pixy.setServos(panLoop.m_command, tiltLoop.m_command);
  }
}
```

Figure 4.24 PixyMon Pan AND Tilt mode

4.7 Simulation and results with screenshots

Complete Setup:

Figure 4.25 represents the complete setup of the robot arm to play board games.

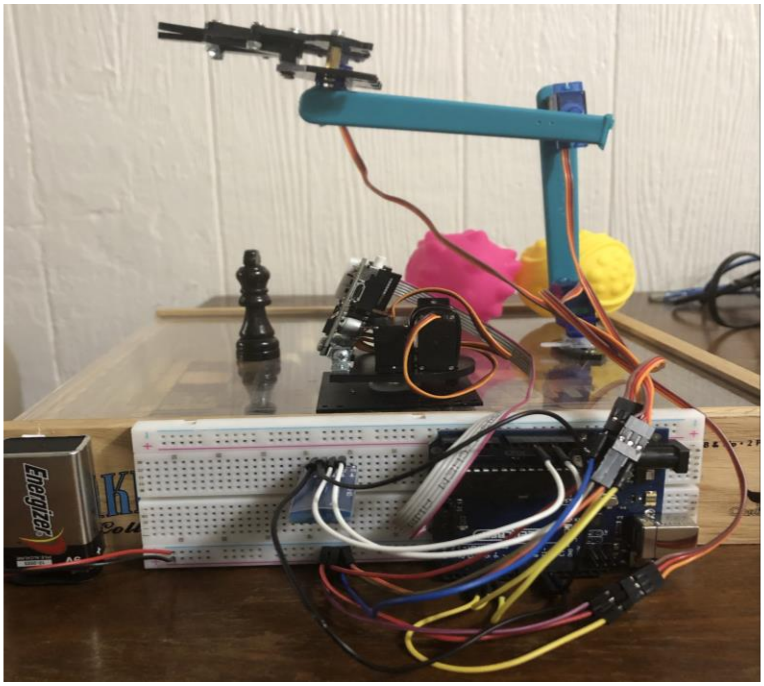

Figure 4.25 Complete Setup of the Robotic Arm

Servo base movement:

Figure 4.26 show the arm movement towards the left, which is controlled using the sliders in the Android application.

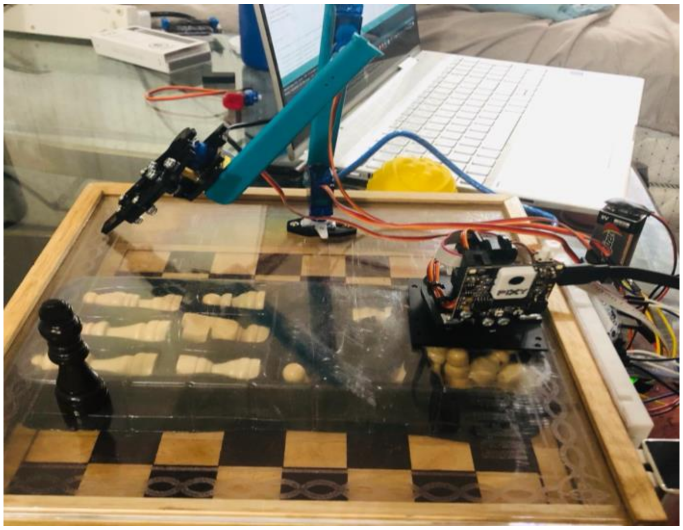

Figure 4.26 Servo base movement of the Arm

Servo arm movement:

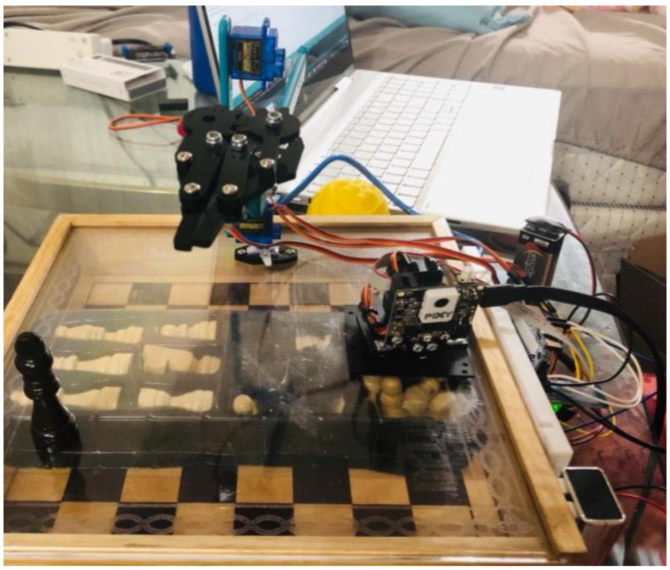

Figure 4.27 Servo arm movement of the robotic arm

Gripper Open:

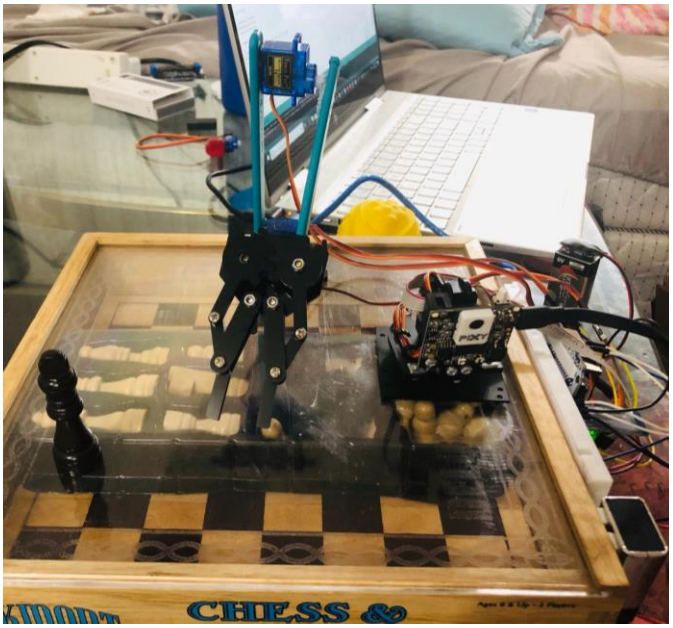

Figure 4.28 Gripper servo open

Gripper servo picking up a chess piece:

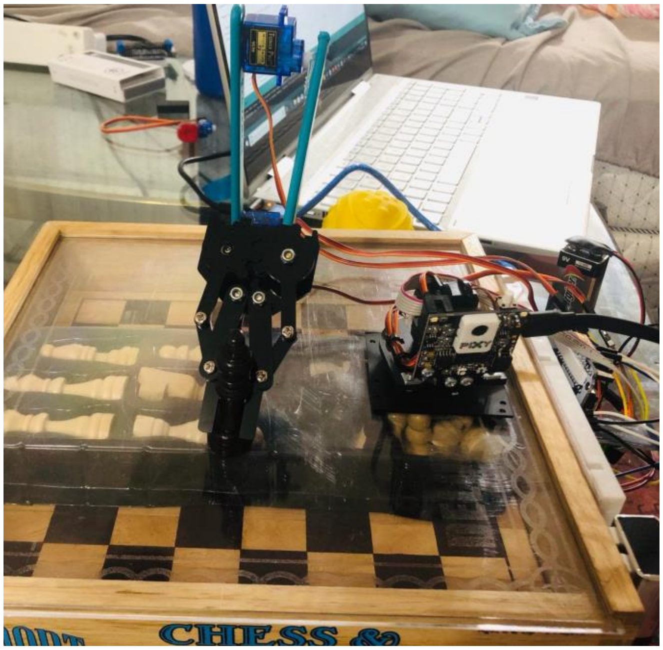

Figure 4.29 Gripper servo picking up a chess piece

Gripper servo moving a chess piece:

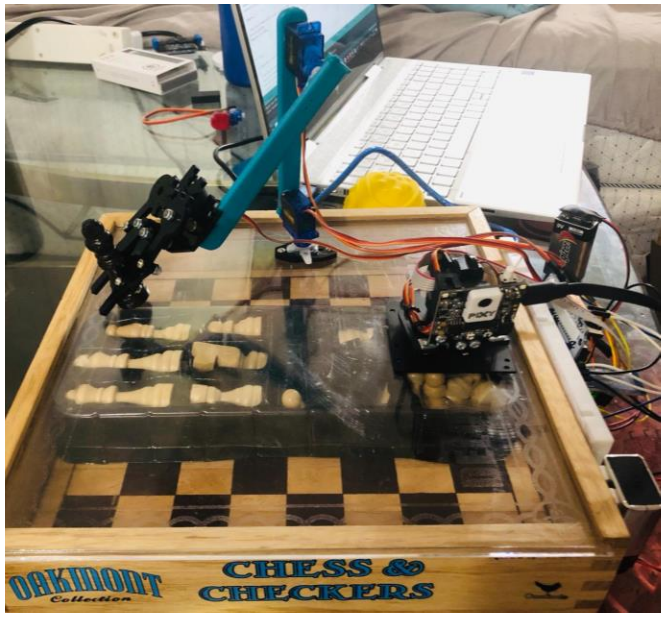

Figure 4.30 Moving the chess piece

Open page of android app:

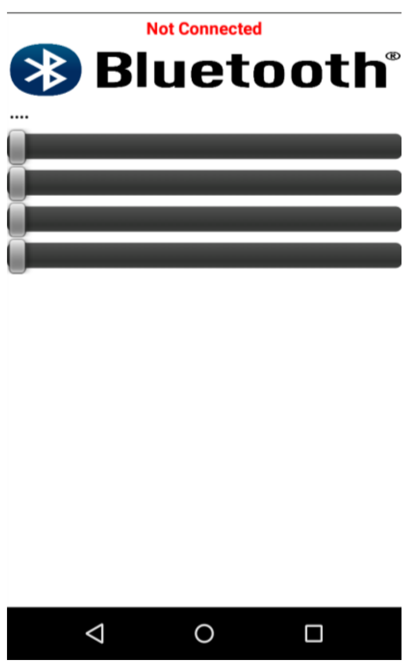

Figure 4.30 Android app to control the robot arm

List of devices:

Figure 4.31 List of devices available

Connection Establishment:

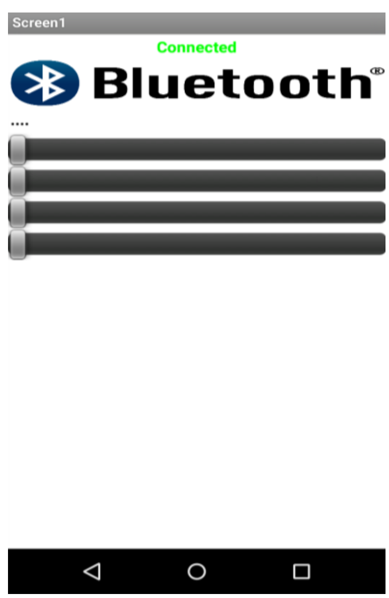

Figure 4.32 Connection Establishment

Slider control:

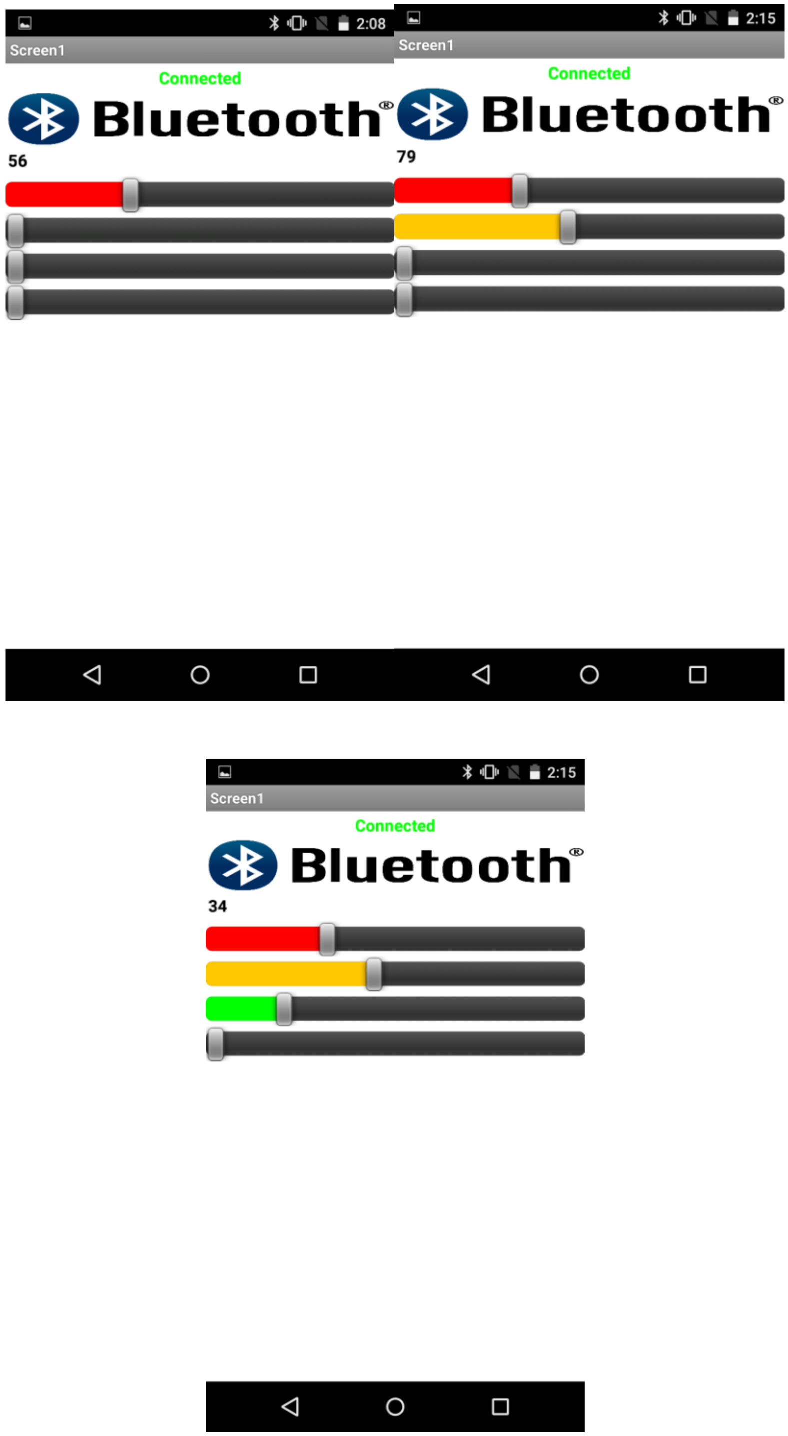

Figure 4.33 Sliding the slider for robotic arm control

PixyMon:

Figure 4.34 and 4.35 shows setting the signatures for object one and two respectively.

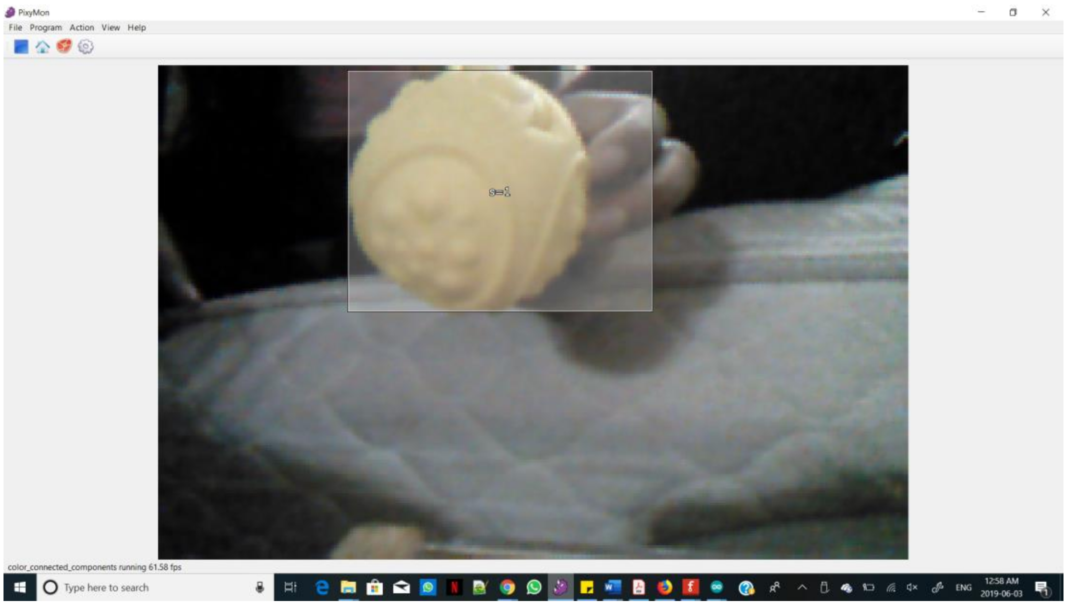

Figure 4.34 Set signatures for object one

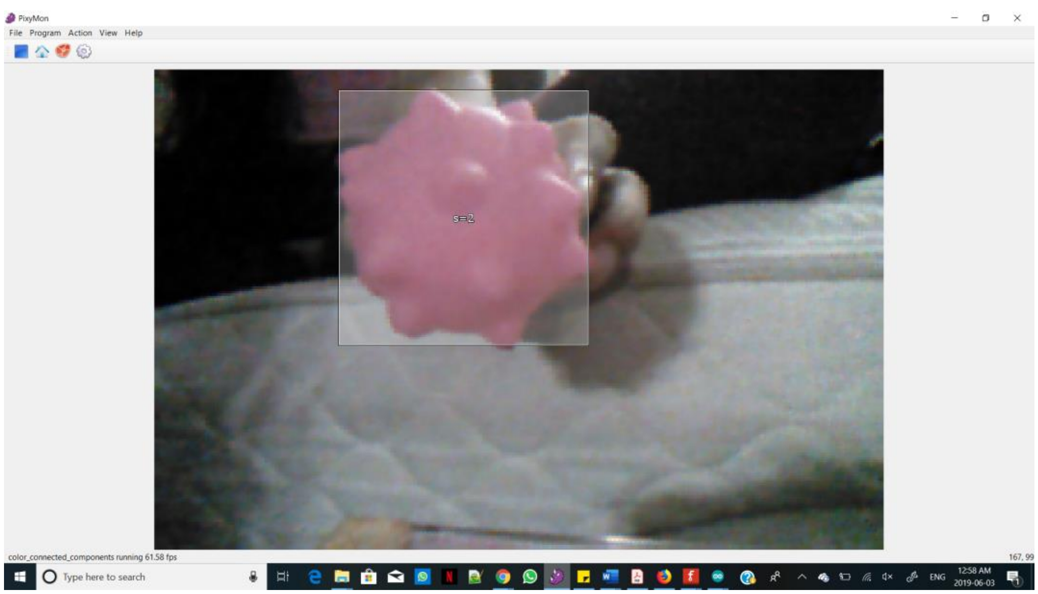

Figure 4.35 Set signatures for object two

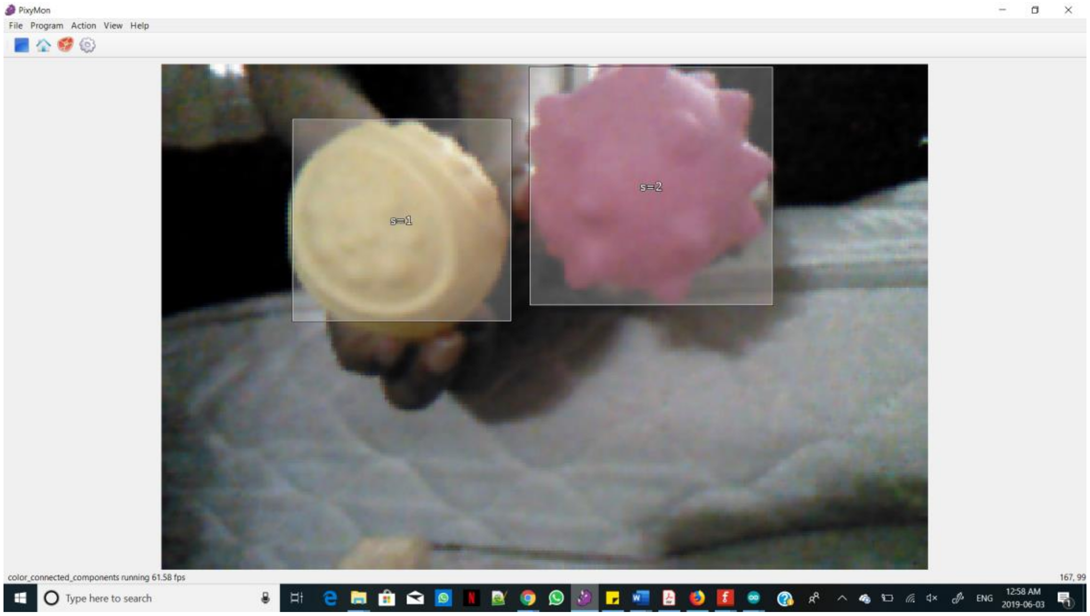

Figure 4.36 Two set objects together, target fixed

Pan/tilt:

-

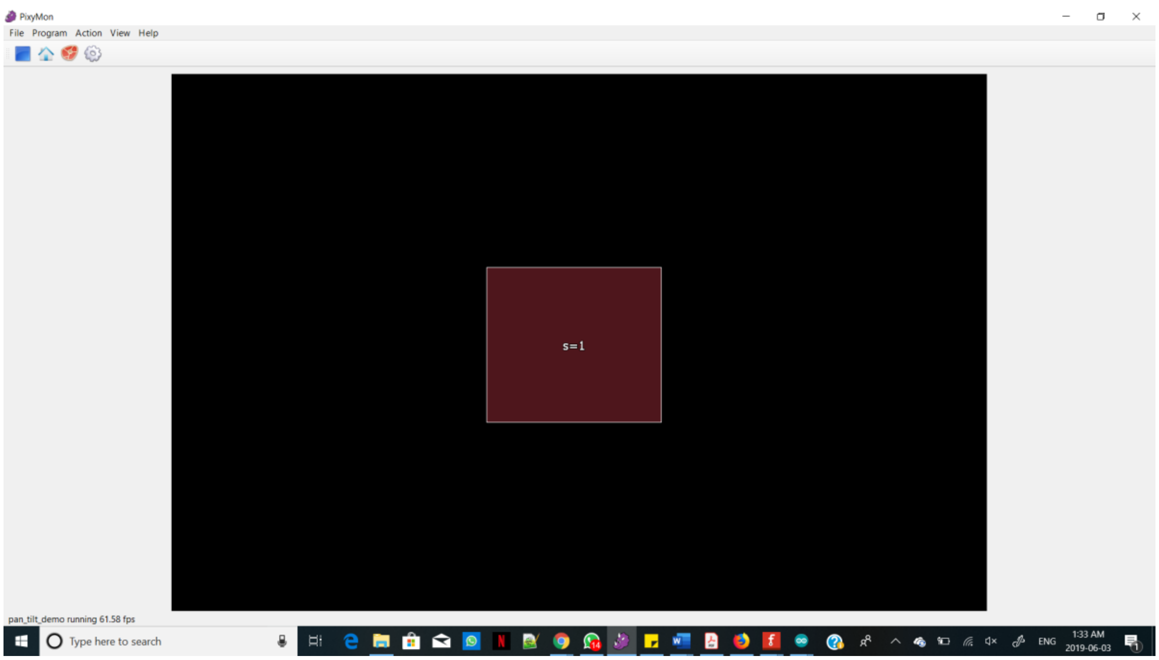

Figure 4.37 pan/tilt result

COM6 (Arduino/Genuino Uno)

```
Starting...
error: noStarting...
error: no response
60
120
180
240
300
360
420
480
540
600
660
```

Figure 4.38 Serial Monitor result for Object 1 (pan/tilt)

```
Detected 1
  block 0: sig: 1 x: 160 y: 111 width: 140 height: 135 index: 38 age: 42
Detected 1
  block 0: sig: 1 x: 160 y: 112 width: 140 height: 136 index: 38 age: 43
Detected 1
  block 0: sig: 1 x: 160 y: 111 width: 140 height: 136 index: 38 age: 44
Detected 1
  block 0: sig: 1 x: 160 y: 111 width: 140 height: 135 index: 38 age: 45
Detected 1
  block 0: sig: 1 x: 162 y: 109 width: 140 height: 133 index: 38 age: 46
Detected 1
  block 0: sig: 1 x: 162 y: 109 width: 140 height: 135 index: 38 age: 47
...
```

Figure 4.39 Serial Monitor result for object 2 (colour-components)

Chapter 5

5.1 Conclusion

The main objective of this project is to develop a Robot Arm to play board games such as chess. The development of this prototype is inexpensive and provides excellent performance. The developed system is user-friendly and is environmentally safe. These types of Robotic arms are helpful in industrial application. Producing them in mass can make the job easier, dangerous tasks such as dealing with nuclear parts or heavy machinery can be done quickly without any human intervention except for operating the arm when it comes to the pixy2 camera, object detection and tracking. It can be used for the surveillance of the area as it can be accessed remotely. It provides a live video stream to the available device. It also detects an object in the video frame and able to follow that object this functionality makes this detect more advance. It is a perfect example of a friendly robot which can be used for personal as well as for an industrial application. Although the robotic arm made by this project is of prototype quality, it has a condition that can be improved for more robotic systems. During the process of creating and developing the project, a lot of theoretical knowledge has been transferred to the practice, and it has been ensured that it is suitable for the project.

One of the main advantages is the low-cost, user-friendly Android application to control the robot arm — avoidance of human-intervention for performing a simple or complicated task, thus saving time consumption. The servo motors used in building this prototype is small and can provide high torque. The camera can detect an object and perform the moving function to follow that object and displays the accuracy of the prediction of that object, the robotic arm can work in offline mode requires only Bluetooth; the developed system can also work without the internet.

Despite having advantages, there are certain limitations for this prototype such as the camera is expensive, Arduino is the brain of this model, in comparison with the computer, it is bit slower as it has low processor than the computer, the servo motors are not reliable cannot accept high power. Using a USB connected camera reduces camera quality, which results in a lower frame rate per second (FPS).

5.2 Future Scope

The current model works on the operation of the Robotic arm and objects detection camera separately, combining both can provide better results such as hooking up the Pixy camera to the Robot Arm can detect the object first and sends the signal to the application after that picking the chess piece. The proposed model can also be expanded by following the tracked object and picking it once it comes to a stop. Using deep learning and face recognition algorithm can be implemented, which can allow the Robot arm to work by moving one place to another and picking up the object, and it will more appropriate in industrial applications.

Bibliography

[1] Vision online marketing team, (2019). [Online] Available: https://www.visiononline.org/blog-article.cfm/Vision-Guided-Robotics-for-Pick-and-%20Place-Applications/71.

[2] Jiangxi Ikv Robot Co., Ltd. (2019). [Online] Available: https://www.ecvv.com/product/4795703.html.

[3] Britannica.com, (2019). [online] Available: https://www.britannica.com/technology/automation.

[4] Ontario Canada, (2019). [Online] Available: https://www.investinontario.com/industrial-automation-and-robotics#robotics-intro

[5] Robots.com, (2019). [Online] Available: https://www.robots.com/faq/why-should-my-company-use-industrial-robots

[6] Hong, Sang Gi, Nae Soo Kim, and Taewook Heo. "A smartphone connected software updating framework for IoT devices." In *2015 International Symposium on Consumer Electronics (ISCE)*, pp. 1-2. IEEE, 2015.

[7] Iwata, Hiroyasu, and Shigeki Sugano. "Design of human symbiotic robot TWENDY-ONE." In *2009 IEEE International Conference on Robotics and Automation*, pp. 580-586. IEEE, 2009.

[8] Quigley, Morgan, Alan Asbeck, and Andrew Ng. "A low-cost compliant 7-DOF robotic manipulator." In *2011 IEEE International Conference on Robotics and Automation*, pp. 6051-6058. IEEE, 2011.

[9] Shin, Dongjun, Irene Sardellitti, and Oussama Khatib. "A hybrid actuation approach for human-friendly robot design." In *2008 IEEE international conference on robotics and automation*, pp. 1747-1752. IEEE, 2008.

[10] Zhang, Zhihua, Jianyong Zheng, Yang Liu, and Gang Lu. "Development of a Movable Service Robot with Double Working Arms for the Elderly and the Disabled." In *2009 International Conference on Electronic Computer Technology*, pp. 636-640. IEEE, 2009.

[11] Zhao, Tie-jun, Jing Yuan, Ming-yang Zhao, and Da-long Tan. "Research on the Kinematics and Dynamics of a 7-DOF Arm of Humanoid Robot." In *2006 IEEE International Conference on Robotics and Biomimetics*, pp. 1553-1558. IEEE, 2006.

[12] Sarkar, Sayan, Gautam Ghosh, Amitrakshar Mohanta, Atreye Ghosh, and Saptarshi Mitra. "Arduino based foot pressure sensitive smart safety system for industrial robots." *In 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pp. 1-6. IEEE, 2017.

[13] Ishak, Mohamad Khairi, and Ng Mun Kit. "Design and Implementation of Robot Assisted Surgery Based on Internet of Things (IoT)." In *2017 International Conference on Advanced Computing and Applications (ACOMP)*, pp. 65-70. IEEE, 2017.

[14] Jhang, Li-Hu, Carlo Santiago, and Chian-Song Chiu. "Multi-sensor-based glove control of an industrial mobile robot arm." In *2017 International Automatic Control Conference (CACS)*, pp. 1-6. IEEE, 2017.

[15] Theodoridis, Theodoros, and Huosheng Hu. "Toward intelligent security robots: A survey." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42*, no. 6 (2012): 1219-1230.

[16] Treptow, André, Grzegorz Cielniak, and Tom Duckett. "Active people recognition using thermal and grey images on a mobile security robot." In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2103-2108. IEEE, 2005.

[17] Park, Chan-Hun, Kyoung-Taik Park, and Dae-Gab Gweon. "Development of industrial dual arm robot for precision assembly of mechanical parts for automobiles." *In 2006 SICE-ICASE International Joint Conference*, pp. 3059-3062. IEEE, 2006.

[18] Kim, GukHwa, JooHan Park, TaeYong Choi, HyunMin Do, Dongll Park, and JinHo Kyung. "Case studies of a industrial dual-arm robot application." In *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pp. 301-302. IEEE, 2017.

[19] Choi, Taeyong, Hyunmin Do, Kyung Taik Park, Doohyung Kim, and Jinho Kyung. "Small sized industrial dual-arm robot with convenient program interface." In *IEEE ISR 2013*, pp. 1-5. IEEE, 2013.

[20] Kim, Peter Ki, Ji-Hun Bae, Hyeonjun Park, Dong-Huk Lee, Jae-Han Park, Moon-Hong Baeg, and Jaeheung Park. "Dual-arm robot box taping with kinesthetic teaching." In *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pp. 555-557. IEEE, 2016.

[21] Kristyanto, B., B. B. Nugraha, A. K. Pamosoaji, and K. A. Nugroho. "Analysis of human arm motions at assembly work as a basic of designing dual robot arm system." *In 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pp. 1316-1320. IEEE, 2017.

[22] Çoban, Mustafa, and Gökhan Gelen. "Wireless teleoperation of an industrial robot by using myo arm band." In *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, pp. 1-6. IEEE, 2018.

[23] Torres, S. Hugo M., and M. John Kern. "7 DOF industrial robot controlled by hand gestures using Microsoft Kinect v2." In *2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC)*, pp. 1-6. IEEE, 2017.

[24] Lambrecht, Jens, Martin Kleinsorge, and Jörg Krüger. "Markerless gesture-based motion control and programming of industrial robots." In *ETFA2011*, pp. 1-4. IEEE, 2011.

[25] Gómez, Andrea Builes, Julio César Correa Rodríguez, Jorge Mario Londoño, and Esteban Palacio Velásquez. "Image recognition application for robotic manipulation of moving objects." In *2015 IEEE 2nd Colombian Conference on Automatic Control (CCAC)*, pp. 1-5. IEEE, 2015.

[26] Tanberk, Senem, and Dilek Bilgin Tükel. "Kinect controlled chess playing robot." *In IEEE EUROCON 2017-17th International Conference on Smart Technologies*, pp. 594-598. IEEE, 2017.

[27] Matuszek, Cynthia, Brian Mayton, Roberto Aimi, Marc Peter Deisenroth, Liefeng Bo, Robert Chu, Mike Kung, Louis LeGrand, Joshua R. Smith, and Dieter Fox. "Gambit: An autonomous chess-playing robotic system." In *2011 IEEE International Conference on Robotics and Automation*, pp. 4291-4297. IEEE, 2011.

[28] Bhuyan, Ariful Islam, and Tuton Chandra Mallick. "Gyro-accelerometer based control of a robotic arm using AVR microcontroller." *In 2014 9th International Forum on Strategic Technology (IFOST)*, pp. 409-413. IEEE, 2014.

[29] Lengare, Pankaj S., and Milind E. Rane. "Human hand tracking using MATLAB to control Arduino based robotic arm." In *2015 International Conference on Pervasive Computing (ICPC)*, pp. 1-4. IEEE, 2015.

[30] Kolesnyk, Kostyantyn, Roman Panchak, Vitalii Pylypenko, Ismail Abliazizov, Oleksandr Fedosyeyev, and Roman Ferens. "Managing robot kinematics based on Arduino controllers using a Unity system." In *2017 XIIIth International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH)*, pp. 44-46. IEEE, 2017.

[31] Kumar, Vishal, Qiang Wang, Wang Minghua, Syed Rizwan, S. M. Shaikh, and Xuan Liu. "Computer vision-based object grasping 6DoF robotic arm using picamera." In

2018 4th International Conference on Control, Automation and Robotics (ICCAR), pp. 111-115. IEEE, 2018.

[32] Tsai, Chi-Yi, Ching-Chang Wong, Chia-Jun Yu, Chih-Cheng Liu, and Tsung-Yen Liu. "A hybrid switched reactive-based visual servo control of 5-DOF robot manipulators for pick-and-place tasks." *IEEE Systems Journal 9*, no. 1 (2014): 119-130.

[33] Hoque, Ahsanul, Md Baijid Hasan Shorif, Shekh Nuruzzaman, and Md Eftekhar Alam. "Arduino based battlefield assistive robot." In *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, pp. 304-309. IEEE, 2017.

[34] Singh, Rahul Kumar, Archisman Sarkar, Debashish Chakravarty, Paritosh Goyal, Vaibhav Lodhi, and Anurag Sharma. "Bluetooth communication-controlled robot based on gesture recognition." *In 2015 IEEE International Transportation Electrification Conference (ITEC)*, pp. 1-5. IEEE, 2015.

[35] Sarker, Sunandita. "Wizard chess: An autonomous chess playing robot." In *2015 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, pp. 475-478. IEEE, 2015.

[36] Kemp, Charles C., Aaron Edsinger, and Eduardo Torres-Jara. "Challenges for robot manipulation in human environments [grand challenges of robotics]." *IEEE Robotics & Automation Magazine 14*, no. 1 (2017): 20-29.

[37] UML - Deployment Diagrams - Tutorials Point

[38] Visual Paradigm, (2019) [Online] Available: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-component-diagram/;WWWSESSIONID=3E5D5B7E85EF401C16182EA407E2840C.www1.

[39] Jameco.com, (2019) [Online] Available: https://www.jameco.com/Jameco/workshop/howitworks/how-servo-motors-work.html.

[40] Module143.com, (2019). [Online] Available: https://www.module143.com/arduino/arduino-uno-r3-atmega328p-microcontroller-board

[41] Ram-e-shop, (2019). [Online] Available: https://ram-e-shop.com/product/kit-bluetooth-hc05/

[42] Roboter, (2019). [Online] Available: https://www.roboter-bausatz.de/en/diy-electronics/motors-servos/servos/246/sg90-9g-micro-servo-engine

[43] Jameco.com, (2019). [Online] Available: https://www.jameco.com/z/PIXY2-Charmed-Labs-Pixy2-Camera-Vision-Sensor_2278287.html

[44] Generation Robots, (2019) [Online] Available: https://www.generationrobots.com/en/403011-pixy2-camera-sensor.html

[45] Cmucam, (2019). [Online] Available: http://www.cmucam.org/projects/cmucam5/wiki/PixyMon_Overview