**A Project Report on**

**AN INTEGARTED OPTIMIZATION AND VISION BASED FRAMEWORK FOR INVERSE KINEMATICS OF SERIAL ROBOTIC MANIPULATOR**

***Submitted in partial fulfillment of the requirements for the award of the degree of***

**Bachelor of Technology**

**in**

**MECHANICAL ENGINEERING**

**BY**

**P. BAPIRAJU 20P31A0347**

**M. DURGA PRASAD 21P35A0335**

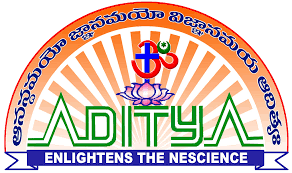
**V. PAVAN KUMAR 20P31A0366**

**M. DURGA PRASAD 21P35A0336**

**Under the esteemed guidance of**

**Dr. CH. V. V. M. J. Satish, M.Tech, Ph.D**

**Associate Professor**



**DEPARTMENT OF MECHANICAL ENGINEERING**

**ADITYA COLLEGE OF ENGINEERING & TECHNOLOGY**

(Permanently Affiliated to JNTUK, Approved by AICTE)

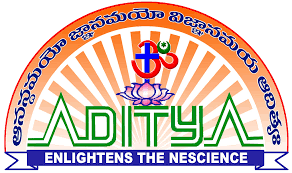
Recognized by UGC under section 2(f) and 12(B) of UGC Act 1955.

Surampalem, E.G. Dist., AP – 533 437.

**(2020-2024)**

**ADITYA COLLEGE OF ENGINEERING & TECHNOLOGY**

**DEPARTMENT OF MECHANICAL ENGINEERING**

**

**CERTIFICATE**

This is to certify that the project report entitled **“AN INTEGARTED OPTIMIZATION AND VISION BASED FRAMEWORK FOR INVERESE KINEMATICS OF SERIAL ROBOTIC MANIPULATOR”** Submitted by

**P. BAPIRAJU 20P31A0347**

**M. DURGA PRASAD 21P35A0335**

**V. VINAY KUMAR 20P31A0366**

**M. DURGA PRASAD 21P35A0336**

in partial fulfillment of requirement for the award of Bachelor of Technology degree in Mechanical Engineering during the academic year **2020-2024**.The results embodied in this project report have not been submitted to any other institute or University for the award of any degree or diploma.

**PROJECT GUIDE HEAD OF THE DEPARTMENT**

**Dr. CH. V. V. M. J. SATISH, M.Tech, Ph.D Dr. DANAIAH PULI, M.Tech, Ph.D**

**Associate Professor Associate Professor & Head**

**Dept. of Mechanical Engineering Dept. of Mechanical Engineering**

**EXTERNAL EXAMINER**

**DECLARATION**

We hereby declare that this project titled **“AN INTEGRATED OPTIMIZATION AND VISION BASED FRAMEWORK FOR INVERSE KINEMATICS OF SERIAL ROBOTIC MANIPULATOR**” has been under taken by us. This work has been submitted to Aditya College of Engineering & Technology Surampalem, in partial fulfillment for the award of degree of Bachelor of Technology in Mechanical Engineering.

We further declare that this project work has not been submitted in full or part for the award of any degree of this or any other educational institutions.

**P. BAPIRAJU 20P31A0347**

**M. DURGA PRASAD 21P35A0335**

**V. PAVAN KUMAR 20P31A0366**

**M. DURGA PRASAD 21P35A0336**

**ACKNOWLEDGEMENT**

We are thankful to our Guide **Dr. CH. V. V. M. J. SATISH, M.Tech, Ph.D,** Associate professor, Department of Mechanical Engineering for being a project guide and for his constant support and encouragement throughout the project.

We wish to convey our sincere thanks to **Dr.** **DANAIAH PULI M.Tech., Ph.D.,** Associate Professor &Head, Department of Mechanical Engineering, ADITYA COLLEGE OF ENGINEERING & TECHNOLOGY, Surampalem for his constant support and encouragement throughout the project.

We wish to record our deep sense of gratitude and profound thanks to our Principal **Dr. DOLA SANJAY**, **M.Tech., Ph.D.**, ADITYA COLLEGE OF ENGINEERING & TECHNOLOGY, for his cooperation and help in completion of our project and course.

Finally, we also thank all the faculty members of the department of Mechanical Engineering for rending co-operation during the period of project work.

We also wish to thank our management and friends for have been constant source of inspiration.

**With sincere regards,**

**P. BAPIRAJU 20P31A0347**

**M. DURGA PRASAD 21P35A0335**

**V. PAVAN KUMAR 20P31A0366**

**M. DURGA PRASAD 21P35A0336**

**Course Outcome mapping with PO’s and PSO’s**

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Name** | Project Work | **Class** | IV B.Tech |
| **Faculty Name** | Dr. CH. V. V. M. J. Satish | **Regulation** | R20 |
| **Academic Year** | 2023-24 | **Semester** | II |
| **Project Title** | Integrated optimization and vision based framework for inverse kinematics in serial robotic manipulator. | | |

**COURSE OUTCOMES (COs):**Upon completion of the course, students will be able to:

|  |  |  |
| --- | --- | --- |
| **CO Number** | **CO Statement** | **Taxonomy** |
| **CO1** | Identify the area of Project | Remember |
| **CO2** | Illustrate the literature of the Project | Understand |
| **CO3** | Determine the problem and plan of action | Apply |
| **CO4** | Experimentation/Design Analysis/Prototype | Create |
| **CO5** | Result & Analysis | Analyze |
| **CO6** | Conclusion, Scope for future work and documentation | Evaluate |

**CO-PO/PSO MATRIX:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** | **PSO3** |
| **CO1** | 3 | 2 |  | 1 |  | 2 |  | 1 | 3 |  | 1 | 2 | 1 | 1 | 1 |
| **CO2** | 2 | 2 |  | 1 |  | 1 | 1 | 2 | 3 |  |  | 2 | 2 |  |  |
| **CO3** | 3 | 2 | 3 | 2 | 1 | 2 | 2 | 2 | 3 | 2 | 3 | 2 | 3 | 1 | 2 |
| **CO4** | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 | 3 | 2 | 3 | 2 | 3 | 2 | 2 |
| **CO5** | 3 | 3 | 2 | 3 | 3 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 2 | 2 | 2 |
| **CO6** | 3 | 3 |  | 3 |  | 2 | 2 | 2 | 3 | 2 | 2 | 2 |  |  | 2 |
| **Course** | 2.83 | 2.5 | 2.67 | 2.00 | 2.33 | 1.83 | 1.8 | 1.83 | 3.0 | 2.25 | 2.40 | 2.17 | 2.20 | 1.50 | 1.80 |

|  |  |  |  |
| --- | --- | --- | --- |
| PO1 | Engineering Knowledge | PO7 | Environment & Sustainability |
| PO2 | Problem Analysis | PO8 | Ethics |
| PO3 | Design / Development of Solutions | PO9 | Individual & Team Work |
| PO4 | Conduct Investigations of complex problems | PO10 | Communication Skills |
| PO5 | Modern Tool usage | PO11 | Project Management & Finance |
| PO6 | Engineer & Society | PO12 | Life-long Learning |

**Table of Contents**

[**List of tables** ii](#_Toc165883480)

[**ABSTRACT** iii](#_Toc165883481)

[**CHAPTER 1** 1](#_Toc165883482)

[**INTRODUCTION** 1](#_Toc165883483)

[**1.1** **Robotic arm definition** 1](#_Toc165883484)

[**1.2** **Robotic description** 1](#_Toc165883485)

[**1.3** **How robots are used across industries** 2](#_Toc165883486)

[**1.3.1** **Industrial** 2](#_Toc165883487)

[**1.3.2** **Farming and Agriculture** 3](#_Toc165883488)

[**1.3.3** **Healthcare** 3](#_Toc165883489)

[**1.3.4** **Logistics** 3](#_Toc165883490)

[**1.3.5** **Retail and Hospitality** 3](#_Toc165883491)

[**1.3.6** **Smart Cities** 3](#_Toc165883492)

[**1.4** **Types of robots** 4](#_Toc165883495)

[**1.4.1** **Classification of robots** 5](#_Toc165883496)

[**1.4.2** **Common types of industrial robots:** 5](#_Toc165883497)

[**CHAPTER 2** 6](#_Toc165883498)

[**LITERATURE REVIEW** 6](#_Toc165883499)

[**CHAPTER 3** 8](#_Toc165883500)

[**INVERSE KINEMATICS IN ROBOTICS** 8](#_Toc165883501)

[**3.1** **Inverse kinematics of robotic arm** 9](#_Toc165883503)

[**CHAPTER 4** 11](#_Toc165883507)

[**COMPUTER VISION** 11](#_Toc165883508)

[**4.1** **Optimized path planning for end effector** 11](#_Toc165883509)

[**4.2** **Arduino programming with MATLAB and Simulink** 12](#_Toc165883513)

[**CHAPTER 5** 13](#_Toc165883514)

[**COMPONENTS OF THE ROBITIC ARM** 13](#_Toc165883515)

[**5.1** **Arduino uno r3 and their description** 13](#_Toc165883516)

[**5.2** **Main specification of arduino** 13](#_Toc165883518)

[**5.3** **Arduino software for coding** 14](#_Toc165883519)

[**5.3.1** **Arduino IDE** 14](#_Toc165883520)

[**5.3.2** **Toolbar button** 14](#_Toc165883522)

[**5.3.3** **Verify/Compile** 15](#_Toc165883524)

[**5.3.4** **Uploading code to Arduino** 15](#_Toc165883525)

[**5.4** **Servo motor** 15](#_Toc165883526)

[**5.4.1** **Controlling of servo motors** 16](#_Toc165883528)

[**5.5** **Gripper** 18](#_Toc165883532)

[**5.5.1** **Working mechanism of gripper** 18](#_Toc165883533)

[**5.5.2** **Features of gripper** 19](#_Toc165883536)

[**5.5.3** **Specifications** 19](#_Toc165883537)

[**5.6** **Interfacing servo motor with arduino uno** 19](#_Toc165883538)

[**5.6.1** **Circuit diagram for more clear view** 20](#_Toc165883540)

[**5.6.2** **Motor connections to Arduino board** 20](#_Toc165883542)

[**CHAPTER 6** 22](#_Toc165883544)

[**METHOLODOGY** 22](#_Toc165883545)

[**6.1** **Introduction to mechanical designing and modelling** 22](#_Toc165883546)

[**6.2** **Construction of Main structure** 23](#_Toc165883548)

[**6.3** **ASSEMBLY OF ROBOTIC ARM** 25](#_Toc165883550)

[**CHAPTER 7** 29](#_Toc165883559)

[**ADVANTAGES, DISADVANTAGES AND APPLICATIONS OF ROBOTIC ARM** 29](#_Toc165883560)

[**7.1** **ADVANTAGES** 29](#_Toc165883561)

[**7.2** **DISADVANTAGES OF ROBOTIC ARMS IN INDUSTRY** 30](#_Toc165883562)

[**7.3** **APPLICATIONS OF ROBOTIC ARM** 31](#_Toc165883563)

[**RESULT** 34](#_Toc165883569)

[**CONCLUSION** 35](#_Toc165883571)

[**REFERENCES** 36](#_Toc165883572)

[**APPENDIX** 38](#_Toc165883573)

[**Code for optimization** 38](#_Toc165883574)

[**Code for object detection through colour** 39](#_Toc165883575)

**List of figures**

[Figure 1. pick and place arms 14](#_Toc165717695)

[Figure 2. implementation plan of robotic arm 14](#_Toc165717696)

[Figure 3. D.O.F of of I.K. robotic arm 18](#_Toc165717697)

[Figure 4. Jointed robotic arm 19](#_Toc165717698)

[Figure 5. Inverse kinematics for a 2-joint robot arm 19](#_Toc165717699)

[Figure 6. finding the joint angles 20](#_Toc165717700)

[Figure 7. Object picking robot 21](#_Toc165717701)

[Figure 8. equation 22](#_Toc165717702)

[Figure 9. Path tracing robotic arm 22](#_Toc165717703)

[Figure 10. Arduino UNO 23](#_Toc165717704)

[Figure 11. Arduino UNO R3 Software 24](#_Toc165717705)

[Figure 12. Arduino UNO R3 Toolbar 25](#_Toc165717706)

[Figure 13. Servo motor Model MG995 26](#_Toc165717707)

[Figure 14. Servo motor MG995 detailed parts 26](#_Toc165717708)

[Figure 15. connection of servo motor wires 26](#_Toc165717709)

[Figure 16. Angular rotation of servo motors 27](#_Toc165717710)

[Figure 17. Working mechanism of gripper 28](#_Toc165717711)

[Figure 18. Gripper 28](#_Toc165717712)

[Figure 19. Interfacing servo motor with Arduino UNO 29](#_Toc165717713)

[Figure 20. Circuit diagram of Arduino to servo motor 30](#_Toc165717714)

[Figure 21. connections to Arduino board 31](#_Toc165717715)

[Figure 22. 2D design model of robotic arm 33](#_Toc165717716)

[Figure 23. Free body 3D diagram of robotic arm 34](#_Toc165717717)

[Figure 24. Bottom view of base portion 35](#_Toc165717718)

[Figure 25. Top view of base portion 35](#_Toc165717719)

[Figure 26. Aluminum robotic arms with cutting portions 36](#_Toc165717720)

[Figure 27. Bread board, jumper wires 36](#_Toc165717721)

[Figure 28. Hons, screws and rivets 37](#_Toc165717722)

[Figure 29. Parts to be Assembled 37](#_Toc165717723)

[Figure 30. Robotic Arm Assembly front view 38](#_Toc165717724)

[Figure 31. Robotic Arm Assembly top view 38](#_Toc165717725)

[Figure 32. Inspection robot 42](#_Toc165717726)

[Figure 33. Water jet cutting robot 42](#_Toc165717727)

[Figure 34. Robotic palletizing system 42](#_Toc165717728)

[Figure 35. Robotic palletizing system 43](#_Toc165717729)

[Figure 36. Painting robot 43](#_Toc165717730)

# **List of tables**

[Table 1. RESULT TABLE 34](#_Toc165883759)

# **ABSTRACT**

In this study, our aim is to mimic object picking application with robotic arm from shelves in a warehouse environment, and our goal is to develop a robust, low-cost, and practical robotic arm system for that purpose. The designed robotic arm correctly picks the desired items from shelf locations and place them into desired bins. Fully autonomous robotic arm operation is accomplished using developed algorithms and Arduino microcontroller. The developed system is tested and evaluated in terms of (i) number of items picked-and-placed correctly, and (ii) increase in the number of objects picked-placed under a certain desired time. For performance evaluations, object weight/size capability results, results of maximum load, and required force results are presented.

Robot manipulator is an essential motion subsystem component of robotic system for positioning, orientating object so that robot can perform useful task. The main objectives of this project are to design and implement a 4-DOF pick and place robotic arm. This project can be self-operational in controlling, stating with simple tasks such as gripping, lifting, placing and releasing. In this project, the focus is on 4-DOF articulated arm. Articulated arm consists of revolute joints that allowed angular movement between adjacent joint. Four servo motors were used in this project to perform four degree of freedom (4-DOF). There are numerous dimensions over which robotic arms can be evaluated, such as torque, payload, speed, range, repeatability and cost, to name a few. Robot manipulators are designed to execute required movements. Their controller design is equally important. The robot arm is controlled by a serial servo controller circuit board. The controller used for servo motor actuation is at mega 15 development board.

Robots are springing up everywhere like mushrooms having found a steady hold in the production industry. Not only do they increase the productivity and efficiency of the system, but they also improve the accuracy and the uniformity of the products. They are a sign of an ever developing technology. One of the most important indications would be the industrial pick and place robot. The main objectives of this project are to design and implement a 4-DOF pick and place robotic arm. This project can be operational in controlling, stating with simple tasks such as gripping, lifting, placing and releasing.

# **CHAPTER 1**

# **INTRODUCTION**

In today’s world, the robotic systems are becoming a part of our daily life with a new application field every day. They are used in the wide range of application areas including healthcare, manufacturing, agriculture, and education. One of the most popular areas of robotic system use is warehouse automation for pick and place application.

It is true that human arms can do many works at a time but it always involves risk of injury during work involving lifting of heavy items and picking of radioactive substances. Moreover, medical industry needs an arm for people who lost it in accidents. Also, it is seen that efficiency of a person decreases as his age advances due to which one involved in rough work becomes unproductive and inefficient.

In manufacturing industry and nuclear industry, a large fraction of the work is repetitive and judicious application of automation will most certainly result in optimum utilization of machine and manpower. A pneumatic `Pick and Place' Robot has been developed to achieve automation in applications where great sophistication is not needed and simple tasks like picking up of small parts at one location and placing them at another location can be done with great ease.

# **1.1 Robotic arm definition**

A robotic arm is a robot manipulator and a mechanical arm, usually programmable, with similar functions to a human arm. The links of such a manipulator are connected by joints allowing either rotational motion or translational displacement. The links of the manipulator can be considered to form a kinematic chain. The business end of the kinematic chain of the manipulator is called the end effectors and it is analogous to the human hand. The end effectors can be designed to perform any desired task such as welding, gripping, spinning etc., depending on the application. The robot arms can be autonomous or controlled manually and can be used to perform a variety of tasks with great accuracy. The robotic arm can be fixed or mobile (i.e. wheeled) and can be designed for industrial or home applications.

# **1.2 Robotic description**

Industrial automation in terms of robotics is now a part and parcel of both industrial and human advancement. Robot arm is one off the most buzzing word in industrial automation. The arm is linked with some separate part. The links of such a manipulator are connected by joints allowing either rotational motion such as in an articulated robot or translational (linear) displacement. The links of the manipulator can be considered to form a kinematic chain. The terminus of kinematic chain of the manipulator is called the end effector and it is analogous to the human hand.

A typical robotic arm has the following components:

• Links and joints

• Actuators

• Controller

• End-effector

A link is considered as a rigid body that defines the relationship between two corresponding joint axes of a manipulator. Manipulators consist of rigid links, which are connected by joints that allow relative motion of corresponding links. The links move to position with the end-effector. Actuators perform the same role the muscles perform in the human arm – they convert stored energy into movement energy. Actuators are used force to move a robot’s manipulator joints. The three common types of actuators currently using in contemporary robots are pneumatic, hydraulic, and electrical actuators. Electric motor-driven actuators perform smoother movements, can be controlled very accurately, and are very reliable. However, these actuators cannot deliver as much power as hydraulic actuators of comparable mass. Nevertheless, for modest power actuator functions, electrical actuators are often preferred. The various types of electric motors used as actuators for robotic applications are direct current (DC) motors, stepper motors and servo motors. The controller is the main part that processes information and carries out instructions in a robot. It is the robot’s ‘brain’ and controls the robot’s movements. It is usually a computer of some type which is used to keep information about the robot and the working process and execute programs which operate the robot. It contains programs, data algorithms, logic analysis and various other processing activities which enable the robot to perform its intended function. End-effector is a device at the end of a robotic arm, designed to interact with the open world. The exact nature of performance of this device depends on the application of the robot. Typical functions of the end-effector consist grasping, pushing and pulling, twisting, using tools, performing insertions, welding and various types of assembly activities. Thus, the major types of robot end-effectors are grippers.

# **1.3 How robots are used across industries**

Businesses and government agencies use robotics in a variety of ways. All five of the common robot types are deployed to enhance outcomes and reduce the burden on employees so they can focus on the most-valuable and most-critical tasks.

# **1.3.1 Industrial**

The manufacturing industry has long been at the forefront of using various types of robots to achieve business results. AMRs, AGVs, articulated robots, and cobots are all deployed on factory floors and in warehouses to help expedite processes, drive efficiency, and promote safety—often in conjunction with programmable logic controllers. They’re used across a variety of applications, including welding, assembly, materials transportation, and warehouse security.

# **1.3.2 Farming and Agriculture**

AMRs are helping farmers harvest their crops more quickly and efficiently—and they’re using impressive intelligence capabilities to do it. Agricultural robots can assess ripeness, move any branches or leaves out of the way, and pick the crop precisely and delicately to avoid causing any harm to the product.

# **1.3.3 Healthcare**

Various types of robots are used in the healthcare industry to enhance the patient experience. AMRs are used to deliver medication, disinfect surfaces, or provide mobile telepresence functionality. Cobots are also used to assist medical professionals during rehabilitation or to help nurses better serve their patients.

# **1.3.4 Logistics**

Robotics help logistics and shipping companies to deliver goods quickly and efficiently. They use AMRs and AGVs as warehouse robots that help them process items, expedite operations, and increase accuracy. They also employ AMRs to take shipments the last mile and ensure safe delivery to customers.

# **1.3.5 Retail and Hospitality**

Robotics can be used to enhance the customer or guest experience in a variety of ways. Retail and hospitality companies are using robotics to automate inventory processes, provide concierge or way-finding services, clean various environments, and assist customers with their luggage or valet parking.

# **1.3.6 Smart Cities**

Robotics help create smarter and safer cities. Humanoid robots offer way-finding and information services. AMRs are used to deliver goods and conduct routine security patrols. Robotics also help expedite building construction, conduct site surveys, and collect building modeling information.

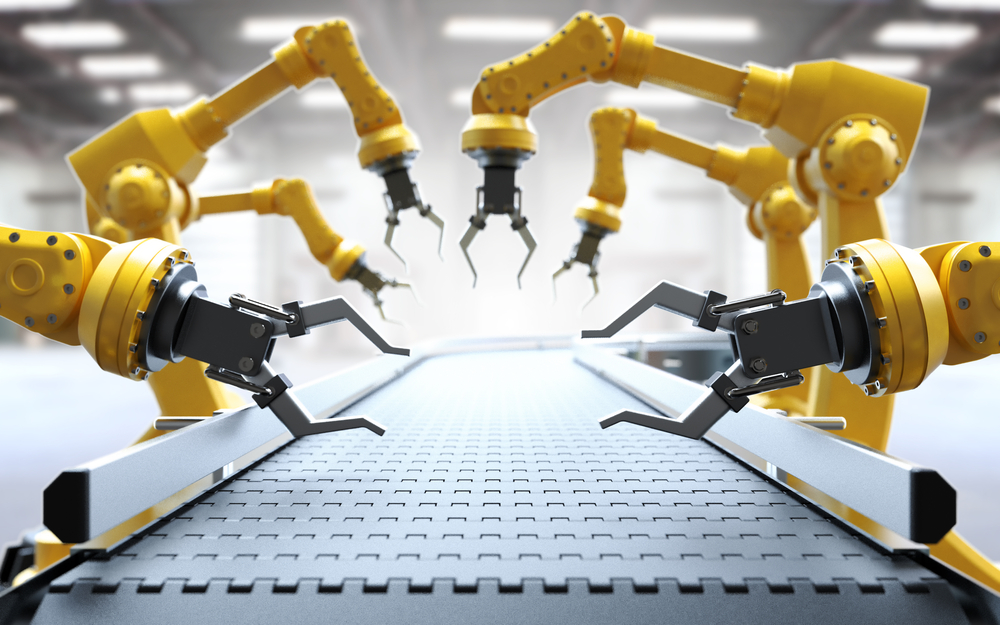


Figure 1. pick and place arms



Figure 2. implementation plan of robotic arm

# **1.4 Types of robots**

The six most common types of robots are :

Autonomous mobile robots (AMRs),

Automated guided vehicles (AGVs),

Articulated robots,

Humanoids,

Cobots, and hybrids.

# **1.4.1 Classification of robots**

There are six main types of industrial robots:

cartesian, SCARA, cylindrical, delta, polar and vertically articulated. However, there are several additional types of robot configurations. Each of these types offers a different joint configuration. The joints in the arm are referred to as axes.

# **1.4.2 Common types of industrial robots:**

* **Articulated** - This robot design features rotary joints and can range from simple two joint structures to 10 or more joints. The arm is connected to the base with a twisting joint. The links in the arm are connected by rotary joints. Each joint is called an axis and provides an additional degree of freedom, or range of motion. Industrial robots commonly have four or six axes.
* **Cartesian** - These are also called rectilinear or gantry robots. Cartesian robots have three linear joints that use the Cartesian coordinate system (X, Y, and Z). They also may have an attached wrist to allow for rotational movement. The three prismatic joints deliver a linear motion along the axis.
* **Cylindrical** - The robot has at least one rotary joint at the base and at least one prismatic joint to connect the links. The rotary joint uses a rotational motion along the joint axis, while the prismatic joint moves in a linear motion. Cylindrical robots operate within a cylindrical-shaped work envelope.
* **Polar** - Also called spherical robots, in this configuration the arm is connected to the base with a twisting joint and a combination of two rotary joints and one linear joint. The axes form a polar coordinate system and create a spherical-shaped work envelope.
* **SCARA** - Commonly used in assembly applications, this selectively compliant arm for robotic assembly is primarily cylindrical in design. It features two parallel joints that provide compliance in one selected plane.

Typical industrial robots are articulated and feature six axes of motion (5 degrees of freedom). This design allows maximum flexibility. Six-axis robots are ideal for:

* Arc welding
* Spot Welding
* Material Handling
* Machine Tending Other Applications.

# **CHAPTER 2**

# **LITERATURE REVIEW**

Summary of the Literature This section summarizes the selected articles that were selected within the time period of 2000-2019. The 30 articles selected were closely studied and have been further analyzed to elaborate the value of AI based robots. After filtering the at-tributes such as authors, year of publication, research context, theoretical base and the findings of articles application contexts of robots from each article are extracted. Since most of the selected articles are based on social robots the practical applications are based on service sector. The value is then viewed through the applications.

most of the articles are based on tourism, hospitality management, healthcare and elderly care sectors. A considerable number of re-search articles are based on the general service sector as well. Novel ways of incorporating robots to the fields such as, financial, cinema and home use are also on the rise. Deviating from the traditional manufacturing environment, emerging concepts such as Robotic Process Automation (RPA) is frequently experimented in different fields. Service sector is one example where the use of robots is experimented to enhance the value creation (Madakam et al., 2019).

The selected articles define and explain the value of robots based on two perspectives. They are the perspective of the individual user and the perspective of the organisation or company. Purchase decisions of robots when viewed from an individual point of view and organizational view differ mainly according to the consumption values (Hur et al., 2012).

Moreover, a positive intention of robot acceptance and adoption is significant before incorporating the robots in activities. Further, foreseeing the value gained for users and competitive advantage ac-quired by organizations enable to enhance the robot usage (Čaić et al., 2018; Calitz et al., 2017; Sabanović, 2010).

Robots are embodied machines with sensors and actuators and some ( potentially limited) degree of artificial intelligence. Also, they can perform tasks associated with particular types of work (Royakkers and van Est 2015).

Robot Institute of America (RIA) defines a robot as a reprogrammable multifunctional manipulator designed to move materials, parts, tools or specialized devices through variable programmed motions for the performance of a variety of tasks (Hegel et al. 2009).

Robots are physical agents that perform tasks by manipulating the physical world. 'To do so, they are equipped with effectors such as legs, wheels, joints, and grippers (Russel and Norvig 2003).

An electromechanically designed machine, programmable by a computer and capable of carrying out a complex series of actions automatically. Also, robot accomplishes tasks by moving into the real world (Madakam et al. 2019).

# **CHAPTER 3**

# **INVERSE KINEMATICS IN ROBOTICS**

In robotics, inverse kinematics makes use of the kinematics equations to determine the joint parameters that provide a desired configuration (position and rotation) for each of the robot's end-effectors. Determining the movement of a robot so that its end-effectors move from an initial configuration to a desired configuration is known as motion planning.

Inverse kinematics transforms the motion plan into joint actuator trajectories for the robot. The movement of a kinematic chain, whether it is a robot or an animated character, is modeled by the kinematics equations of the chain. These equations define the configuration of the chain in terms of its joint parameters.

Forward kinematics uses the joint parameters to compute the configuration of the chain, and inverse kinematics reverses this calculation to determine the joint parameters that achieve a desired configuration. kinematic analysis Kinematic analysis is one of the first steps in the design of most industrial robots. Kinematic analysis allows the designer to obtain information on the position of each component within the mechanical system. This information is necessary for subsequent dynamic analysis along with control paths.

Inverse kinematics is an example of the kinematic analysis of a constrained system of rigid bodies, or kinematic chain. The kinematic equations of a robot can be used to define the loop equations of a complex articulated system. These loop equations are non-linear constraints on the configuration parameters of the system. The independent parameters in these equations are known as the degrees of freedom of the system. While analytical solutions to the inverse kinematics problem exist for a wide range of kinematic chains, computer modeling and animation tools often use Newton's method to solve the non-linear kinematics equations.

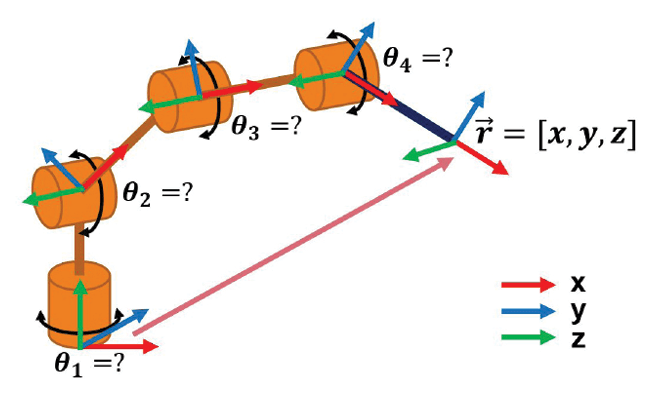


Figure 3. DOF of of I.K. robotic arm

# **3.1 Inverse kinematics of robotic arm**

Kinematics is the science of motion. In a two-joint robotic arm, given the angles of the joints, the kinematics equations give the location of the tip of the arm. Inverse kinematics refers to the reverse process. Given a desired location for the tip of the robotic arm, what should the angles of the joints be so as to locate the tip of the arm at the desired location. There is usually more than one solution and can at times be a difficult problem to solve. This is a typical problem in robotics that needs to be solved to control a robotic arm to perform tasks it is designated to do. In a 2-dimensional input space, with a two-joint robotic arm and given the desired coordinates.

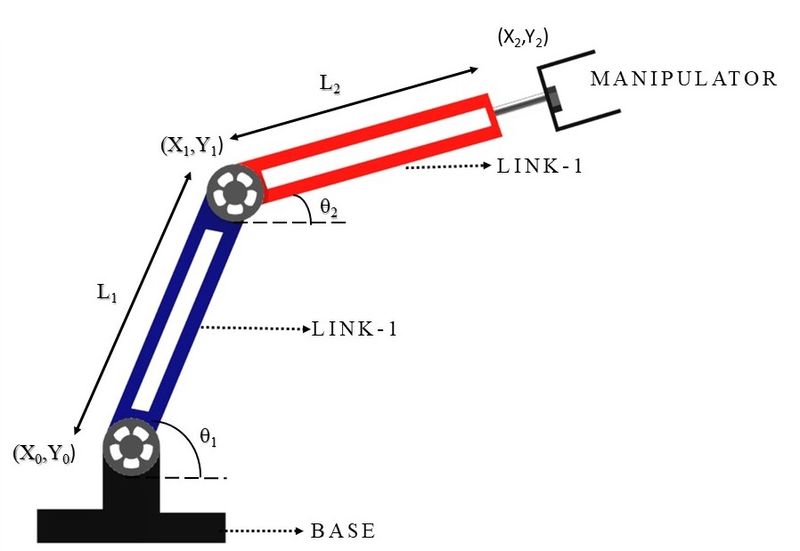


Figure 4. Jointed robotic arm

Chart, line chart

Description automatically generated

Figure 5. Inverse kinematics for a 2-joint robot arm

**a \* cos + b \* sin theta = c**

**theta = (tan (c/(+/- (a ^ 2 + b ^ 2 - c ^ 2))) – (tan(a/b)**

**a = a\_{2}\*S\_{2} ,b = a1+a2C2, c = y**

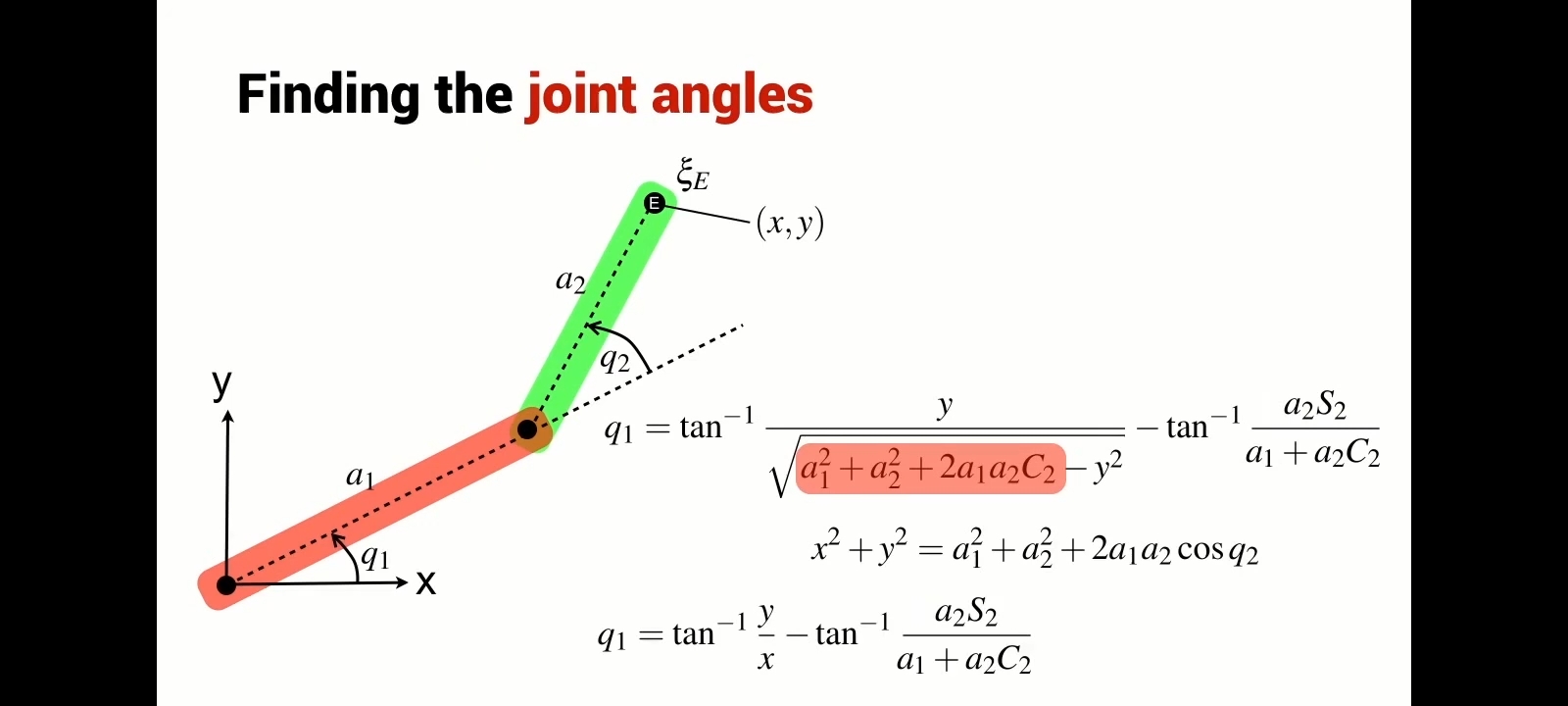


Figure 6. finding the joint angles

Apply sum of angles identities

Substitute cos q2 - c2,sinq2 - s2

X=a2 cos( q1+q2) + a1 cos q1

Y=a2sin(q1+q2)+ a1 sin q1

X= (a1+a2c2)cos q1 -a2s2 sinq1

Y= (a1+a2c2)sin q1 + a2s2 cosq1

# **CHAPTER 4**

# **COMPUTER VISION**

Computer vision in pick and place robotic arms involves using visual data to enhance the efficiency and accuracy of the arm's operations. This typically includes algorithms that analyze images or video streams to identify objects, determine their positions, orientations, and other relevant information, and then use this data to optimize the movement and actions of the robotic arm. By leveraging computer vision, the system can adapt to variations in the environment, such as changes in lighting or object placement, improving overall performance and productivity.

# **4.1 Optimized path planning for end effector**

Path planning can be considered as the way to obtain a trajectory between an initial and final point. The path position of a serial robot is specified in terms of the end-effector position in the Cartesian space. Different trajectories can be performed by means of actuators in order to determine a path of the end-effector from a start point to a goal point .



Figure 7. Object picking robot

The workspace characteristics depend on the link lengths, which can be considered as optimization parameters. A reasonable choice to assume as Xs start point is a robot position in which all actuators are in the zero position Then, goal point Xgoal can be chosen as a point, which is located at a fixed distance d, from the starting point. However, distance d has to satisfy the constraint equation.



Figure 8. equation

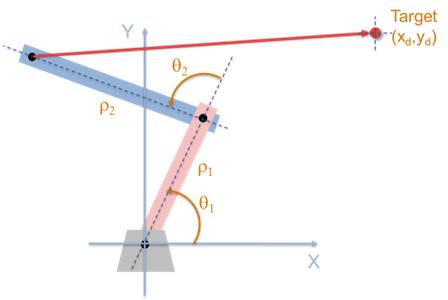


Figure 9. Path tracing robotic arm

The Jacobian matrix pseudoinverse method is proposed and it is the most commonly used method of robot inverse kinematics. The algorithm is unstable when the robot approaches the singular configuration. Four Jacobian-based methods of solving the inverse kinematics are evaluated. A novel formula for calculating the pseudoinverse is introduced for a class of redundant robots.The optimal approximation of the Jacobian pseudoinverse algorithm by the extended Jacobian algorithm

fmincon passes x to your objective function and any nonlinear constraint functions in the shape of the x0 argument. For example, if x0 is a 5-by-3 array, then fmincon passes x to fun as a 5-by-3 array. However, fmincon multiplies linear constraint matrices A or Aeq with x after converting x to the column vector x(:)

# **4.2 Arduino programming with MATLAB and Simulink**

[MATLAB support package for Arduino](https://www.mathworks.com/products/hardware/arduino.html) lets you write MATLAB programs that read and write data to your Arduino and access connected devices such as motors, LEDs, and I2C devices. Because MATLAB is a high-level interpreted language, prototyping and refining algorithms for your Arduino projects is easy, and you can see results from I/O instructions immediately.

# **CHAPTER 5**

# **COMPONENTS OF THE ROBITIC ARM**

# **5.1 Arduino uno r3 and their description**

The Arduino UNO is an open source micro controlled board based on the Microchip ATmega328 microcontroller and developed by Arduino.cc The board is equipped sets of digital and along Input/Output (I/O) pins that may be interfaced to various (shields) and other circuits. The board has 14 digital I/O pins (six capable of PWM output), 5 along I/O pins, and is programmable with the IDE (Integrated Development Environment), via a type of USB cable. It can be powered by the USB cable or by an external 9-Volts battery though it accepts voltages between 7 and 20 volts. It is like the arduino nano and Leonardo. The hardware reference design is distributed under Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available.

A picture containing text, electronics, circuit

Description automatically generated

Figure 10. Arduino UNO

Arduino UNO is a low-cost, flexible, and easy-to-use programmable open-source microcontroller board that can be integrated into a variety of electronic projects. This board can be interfaced with other Arduino boards, Arduino shields, Raspberry Pi boards and can control relays, LEDs, servos, and motors as an output.

# **5.2 Main specification of arduino**

Arduino UNO is a microcontroller board based on the ATmega328p. It has 14 digital input/output pins (of which 5 can be used as PWM outputs), 5 along inputs, a 15 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst-case scenario you can replace the chip for a few dollars and start over again.

* Microcontroller: ATmega328.
* Operating Voltage: 5V.
* Input Voltage (recommended): 7-12V.
* Input Voltage (limits): 5-20V.
* Digital I/O Pins: 14 (of which 5 provide PWM output)
* Analog Input Pins: 5.
* DC Current per I/O Pin: 40 mA.
* DC Current for 3.3V Pin: 50 mA.

# **5.3 Arduino software for coding**

The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low-cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics.

# **5.3.1 Arduino IDE**

The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment. The program or code written in the Arduino IDE is often called as sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension 'ino.'

Graphical user interface, text, application

Description automatically generated

Figure 11. Arduino UNO R3 Software

# **5.3.2 Toolbar button**

The icons displayed on the toolbar are New, Open, Save, Upload, and Verify. It is shown below

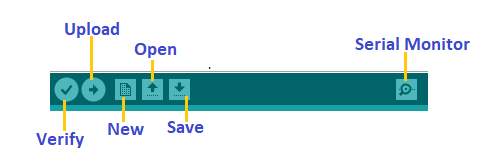


Figure 12. Arduino UNO R3 Toolbar

# **5.3.3 Verify/Compile**

It will check for the errors in the code while compiling. The memory in the console area is also reported by the IDE.

# **5.3.4 Uploading code to Arduino**

The Upload button compiles and runs our code written on the screen. It further uploads the code to the connected board. Before uploading the sketch, we need to make sure that the correct board and ports are selected.

We also need a USB connection to connect the board and the computer. Once all the above measures are done, click on the Upload button present on the toolbar. The latest Arduino boards can be reset automatically before beginning with Upload. In the older boards, we need to press the Reset button present on it. As soon as the uploading is done successfully, we can notice the blink of the Tx and Rx LED. If the uploading is failed, it will display the message in the error window.

We do not require any additional hardware to upload our sketch using the Arduino Bootloader. A **Bootloader** is defined as a small program, which is loaded in the microcontroller present on the board. The LED will blink on PIN 13.

# **5.4 Servo motor**

A **servo motor** is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision. If you want to rotate an object at some specific angles or distance, then you use a servo motor. It is just made up of a simple motor which runs through a **servo mechanism**. If motor is powered by a DC power supply then it is called DC servo motor, and if it is AC-powered motor then it is called AC servo motor. For this tutorial, we will be discussing only about the **DC servo motor working**. Apart from these major classifications, there are many other types of servo motors based on the type of gear arrangement and operating characteristics. A servo motor usually comes with a gear arrangement that allows us to get a very high torque servo motor in small and lightweight packages. Due to these features, they are being used in many applications like toy car, RC helicopters and planes, Robotics, etc.



Figure 13. Servo motor Model MG995

# **5.4.1 Controlling of servo motors**

Interfacing hobby Servo motors like s90 servo motor with MCU is very easy. Servos have three wires coming out of them. Out of which two will be used for Supply (positive and negative) and one will be used for the signal that is to be sent from the MCU. An MG995 Metal Gear Servo Motor which is most commonly used for RC cars humanoid bots etc. The picture of MG995 is shown below in figure.

All servo motors work directly with your +5V supply rails but we have to be careful on the amount of current the motor would consume if you planning to use more than two servo motors a proper servo shield should be designed.

A picture containing gear

Description automatically generated

Figure 14. Servo motor MG995 detailed parts

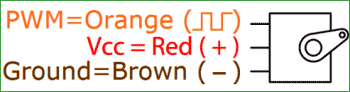


Figure 15. connection of servo motor wires

All motors have three wires coming out of them. Out of which two will be used for Supply (positive and negative) and one will be used for the signal that is to be sent from the MCU. Servo motor is controlled by PWM (Pulse with Modulation) which is provided by the control wires. There is a minimum pulse, a maximum pulse and a repetition rate. Servo motor can turn 90 degrees from either direction form its neutral position. The servo motor expects to see a pulse every 20 milli seconds and the length of the pulse will determine how far the motor turns. For example, a 1.5ms pulse will make the motor turn to the 90° position, such as if pulse is shorter than 1.5ms shaft moves to 0° and if it is longer than 1.5ms than it will turn the servo to 180°.

Servo motor works on PWM (Pulse width modulation) principle, means its angle of rotation is controlled by the duration of applied pulse to its Control PIN. Basically, servo motor is made up of DC motor which is controlled by a variable resistor (potentiometer) and some gears. High speed force of DC motor is converted into torque by Gears. We know that WORK= FORCE X DISTANCE, in DC motor Force is less and distance (speed) is high and in Servo, force is high, and distance is less. The potentiometer is connected to the output shaft of the Servo, to calculate the angle and stop the DC motor on the required angle.

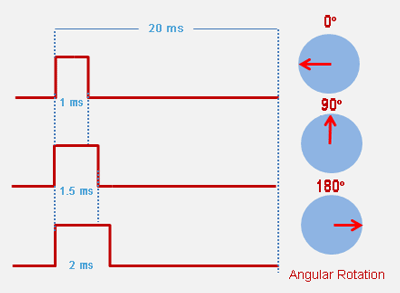


Figure 16. Angular rotation of servo motors

Servo motor can be rotated from 0 to 180 degrees, but it can go up to 210 degrees, depending on the manufacturing. This degree of rotation can be controlled by applying the Electrical Pulse of proper width, to its Control pin. Servo checks the pulse in every 20 milliseconds. The pulse of 1 MS (1 millisecond) width can rotate the servo to 0 degrees, 1.5ms can rotate to 90 degrees (neutral position) and 2 MS pulse can rotate it to 180 degree. All servo motors work directly with your +5V supply rails but we have to be careful about the amount of current the motor would consume if you are planning to use more than two servo motors a proper servo shield should be designed. Gears are used to control position in degrees

* Output shaft is used to connect with physical objects
* Potentiometer is for position feedback
* Pulse to voltage converter is for feedback voltage to potentiometer
* High bridge is to control the position of motor (clockwise or anti clockwise)
* Comparators make with amplifier

# **5.5 Gripper**

In the simplest terms, grippers are devices that enable robots to pick up and hold objects. When combined with a collaborative industrial robot arm, grippers enable manufacturers to automate key processes, such as inspection, assembly, pick and place and machine tending.

They are generally used when you are grasping parts by two parallel, flat surfaces. They can grip a part by either the gripper’s closing or opening motion. There are also angular two-jaw grippers with 90-degree pivoting fingers that can completely retract and provide extra clearance that is beneficial for certain application.

# **5.5.1 Working mechanism of gripper**

These grippers feature an electric motor and a controller. The controller provides a signal relating to the force, position, or the speed required of the robot. The gripper receives the signal and its motor carries out the desired motion. A mechanical gripper is used as an end effector in a robot for grasping the objects with its mechanically operated fingers. Mechanical Gripper. A mechanical gripper is used as an end effector in a robot for grasping the objects with its mechanically operated fingers. Grippers are the most used type of end-effectors. They can use different gripping methods.

Diagram

Description automatically generated with medium confidence

Figure 17. Working mechanism of gripper



Figure 18. Gripper

They are generally used when you are grasping parts by two parallel, flat surfaces. They can grip a part by either the gripper’s closing or opening motion. There are also angular two-jaw grippers with 90-degree pivoting fingers that can completely retract and provide extra clearance that is beneficial for certain application.

# **5.5.2 Features of gripper**

* Paw Maximum Opening Angle: 55mm (without expansion board when the maximum opening angle spacing of 55mm, the largest expansion board after the installation is 95mm)
* The plank kit is very light.
* Compatible with our [MG995](https://robu.in/product-tag/mg995/) Servo

# **5.5.3 Specifications**

1. The degree of Freedom (DOF): 2
2. Claw material: aluminium alloy
3. Claw weight: about 58g (without motor)
4. Claws overall length: 108mm (overall maximum length when closed claw)
5. Claws overall width: 98mm (maximum overall width of the claw when open)
6. Claw maximum opening angle: 55mm

# **5.6 Interfacing servo motor with arduino uno**

Servo motor control using Arduino Uno R3 can be easily done by interfacing servo motor with Arduino. Unlike other motors, servo motor is very easy to interface with Arduino or any other microcontroller due to its built-in controllers. We only need three pins to interface the servo motor with Arduino. Power pin, Ground pin and pulse signal pin.

A picture containing text, electronics, screenshot

Description automatically generated

Figure 19. Interfacing servo motor with Arduino UNO

Servo motor cannot move with continuous motion unless feedback potentiometer is connected. H bridge is used to rotate motor either in clockwise or anti clockwise direction. So the main advantage of servo motor is that it doesn’t require any interfacing circuit. However, the main disadvantage of servo motor is its lower speed and low power limitation as compared to other motors.

# **5.6.1 Circuit diagram for more clear view**

Diagram, schematic

Description automatically generated

Figure 20. Circuit diagram of Arduino to servo motor

For the servo motor, we have connected the control signal pin with Arduino digital pin 10. The VCC and GND pins are in common with the Arduino 5V and GND pins. You can, however, use an external 5V power supply to power the servo motor as well.

# **5.6.2 Motor connections to Arduino board**

The hardware system implementation is given in the block diagram of the physical system consists of the computer, and robotic arm. These devices are connected to one another over USB. The microcontroller is connected to the servo motors using jumper wires, and the motors are powered using a 5volt supply. The system’s components are shown and described in and the circuit design of the system is shown in the physical system constructed for the system consists of the circuit diagram, as shown in the circuit consists of an Arduino Uno microcontroller and four servo motors. The controller and the servos are powered using a 5 V power supply. The microcontroller is connected using a Universal Serial Bus (USB) to the computer that sends commands to the device to actuate the motors accordingly.

D C power

supply

s Servo 1

Computer

Servo 1

Arduino UNO

Micro controller

Servo 2

Servo 3

Servo 4

Diagram

Description automatically generated

Figure 21. connections to Arduino board

The process through which the arm actuates is described in further detail later on. The circuit was modeled in Proteus. shows the descriptions of the components. As shown in the circuit is used to implement the serial manipulator, The system includes three servos that move and position the gripper, which is actuated by a fourth servo motor. The circuit components are connected via acrylic parts using nuts and bolts. The robotic arm has four degrees of freedom: the rotation about x-axis, y-axis, and z-axis plus opening and closing of the gripper/ The measurements for the mechanical parts of the manipulator.

# **CHAPTER 6**

# **METHOLODOGY**

It is developed by using a simple microcontroller. In this case computerized scanning and figuring the required shape or simply functioning it with programming can make the work easier. The computer language uploaded to a microcontroller system will convert binary information to voltage variation which will function the joint of the arm by using mechanical component with calculated mechanics. For controlling the servo motor there will be three points of wire coming from the motor, two of them are ground and supply voltage and last one is for signal. In case of servo motor, it can rotate according to the programme and its range is 180 degrees to 0 degree.

This kind of servo loaded with the pin number 3,5,5,9 from the Arduino board. And both ground and supply voltage is given by the provided pin from Arduino called 5V and GND.4 servo work for joint movement and the movement is specified by objects position. This work covers material selection, design, programming and fabrication of a basic robotic arm system. It also covers the implementation of the kinematics of the arm but does not consider the details of the derivation of the kinematic equations.

• select a suitable material for the fabrication of a 2-DoF robotic arm;

• Obtain suitable design parameters for the robotic arm;

• Create a 3-d model of a robotic arm based on the design parameters;

• Fabricate the robotic arm

Inverse Kinematics of robotic arm

Distance minimization using FMINCON FUNCTION Function

The availability of a robotic arm that can be used for demonstrating and educational purposes in the Department of Mechanical Engineering will go a broad way in stimulating the interest of students in robotics. It will provide a tool to use for learning and experimenting with robotics. Students with a flair for programming can reprogram the robot to adapt it to different tasks. In this chapter covers the detailed design and method of construction of the robotic arm and its controller.

# **6.1 Introduction to mechanical designing and modelling**

Inspired by a human hand, the mechanical aspect of a robotic arm design constitutes of several linkages which can be thought of as to form a kinematic chain. The links are connected by joints, which provide the necessary rotational and translational capabilities to the mechanism. The computer controls the robot by rotating individual servo motors connected to each joint

The mechanical design of the robot arm is functioned on a robotic movement with similar functions to a human arm. The links of such a movement are connected by joints allowing rotational motion and the links of the manipulators considered to form a kinematic chain. For designing the arm, its force distribution was very important to analysis in case of force distribution. The torque exerting at each of the joints is going to calculate in this project and a servo with the required torque rating is being selected for each joint. Selecting a suitable servo controller and control software for the Robotic arm is developing

The robotic arm at first the loop starts by scanning its surroundings by rotating 350 degrees. As a sonar sensor is used for determining the co-ordinate of any object. While the sonar finds any object it will send a signal to the processing unit (Micro controller) then the output signal will stop the rotating stepper motor and start functioning the grabbing part. In case of grabbing a object the processor provided with Arduino uno r3 gives voltage signal from binary coded information. In this case the moving arm will move by its servo motor according to object’s co-ordinate. As this is a three-dimensional co-ordinate system, this can be displayed in matlab with graphical representation.

Diagram

Description automatically generated

Figure 22. 2D design model of robotic arm

# **6.2 Construction of Main structure**

In construction of robotic arm, the kinematics are copied by usual arm movement capability of a general human being. All the force are distributed such a way that the base can have the load with 350-degree moveability.5 joints are attached through servo motor with mechanical support of hard board. For shaping the hard board of the prototype of normal hexose and a cutter is used. Where fixed joint is required glue gum and metal screw. The business end of the kinematic chain of the manipulator is called the end effector or end-of-arm-tooling and it is analogous to the human hand. Figure shows the Free Body Diagram for mechanical design of the robotic arm.

As shown, the end effector is not included in the design because a commercially available gripper is used. In the base, a rateable platform made of plastic is used where the platform can have a 350-degree access and it gives the freedom need for the arm to reach a certain co-ordinate. The sonar sensor is attached with the rotating platform for 350-degree scanning. For locating the servo motor with the arm, 15kg/cm servo was attached with the elbow and 14kg/cm servo was attached with the wrist. In case of end effector an 8kg/cm servo is used for mechanical claw. A serial communication system was introduced to the robot arm where the distance of any object can be displayed through serial monitor as unit in cm.

**Diagram

Description automatically generated with medium confidence**

Figure 23. Free body 3D diagram of robotic arm

The robotic arm itself: Represented as a simple geometric shape such as a rectangle or a series of connected links.

External forces: These include forces such as gravity, applied loads, friction, and any other external forces acting on the robotic arm. For example, if the arm is lifting an object, the weight of the object and any applied forces or torques are depicted.

Joints and connections: The FBD shows where the robotic arm is connected to other components or fixed points, such as motors, hinges, or the base. The reactions at these connection points, such as support forces or torques, are also included in the diagram.

By analyzing the free body diagram, engineers and designers can determine the net forces and torques acting on the robotic arm, helping them to optimize its design, predict its motion, and ensure structural integrity. It's an essential step in the design, analysis, and optimization of robotic systems.

# **6.3 ASSEMBLY OF ROBOTIC ARM**

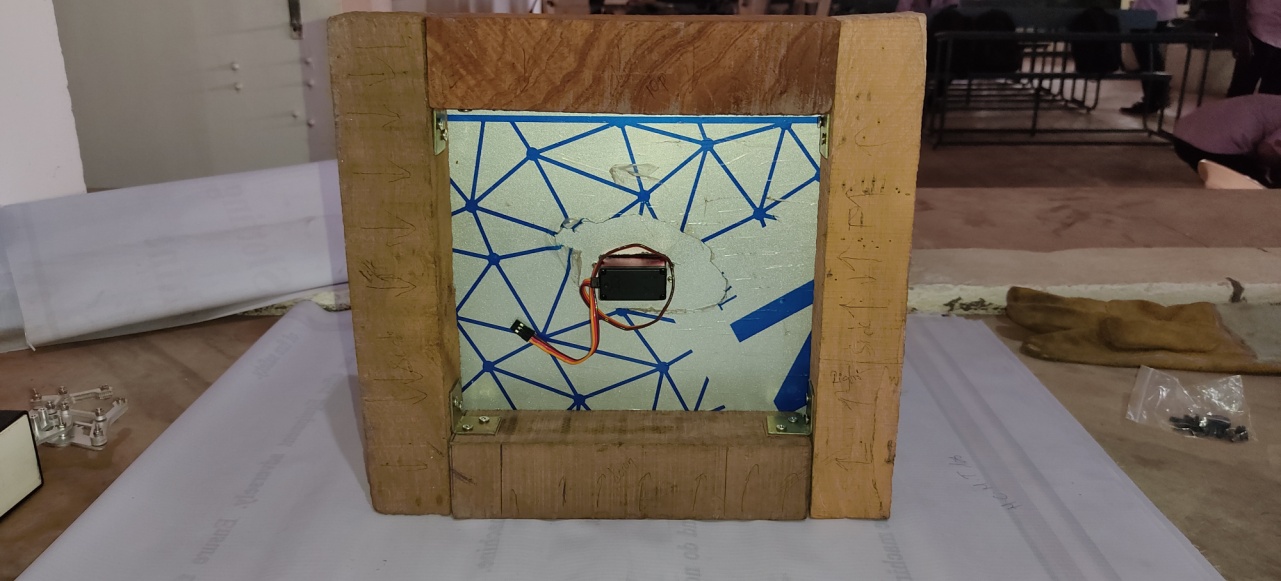


Figure 24. Bottom view of base portion

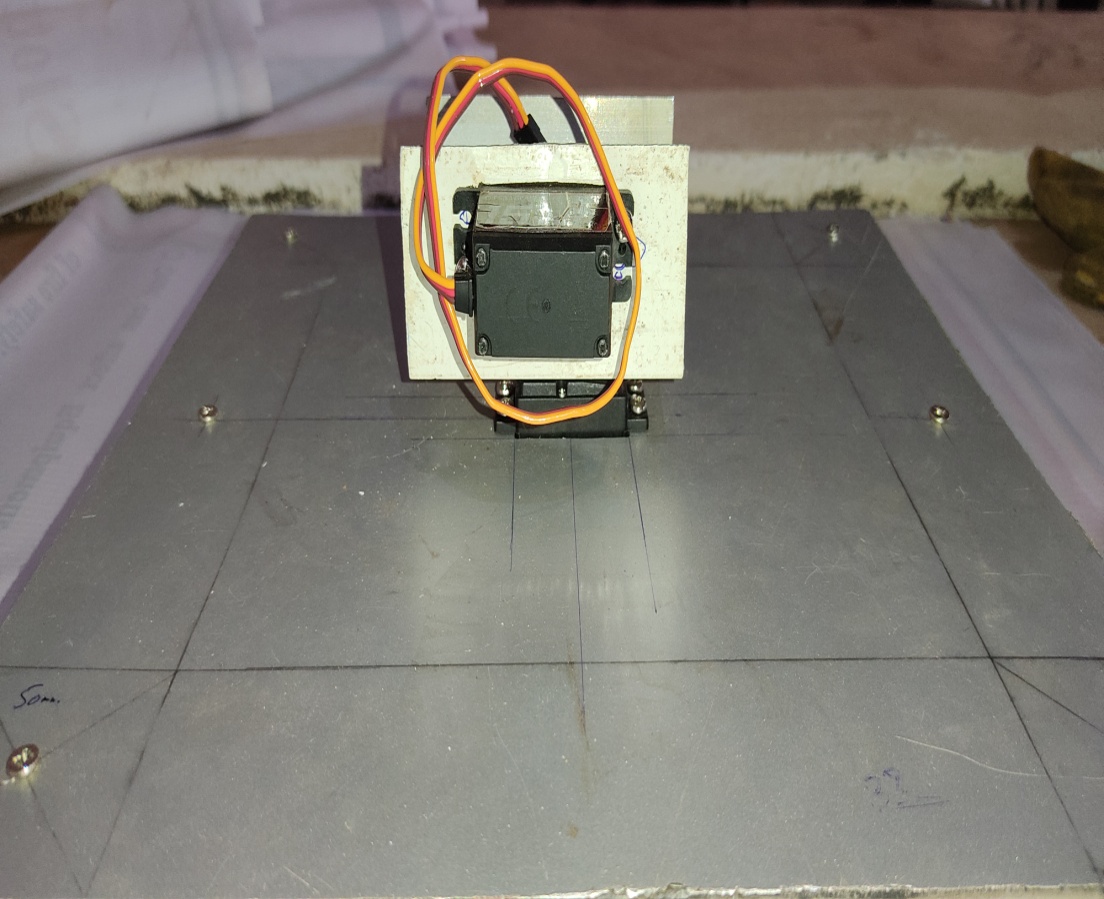
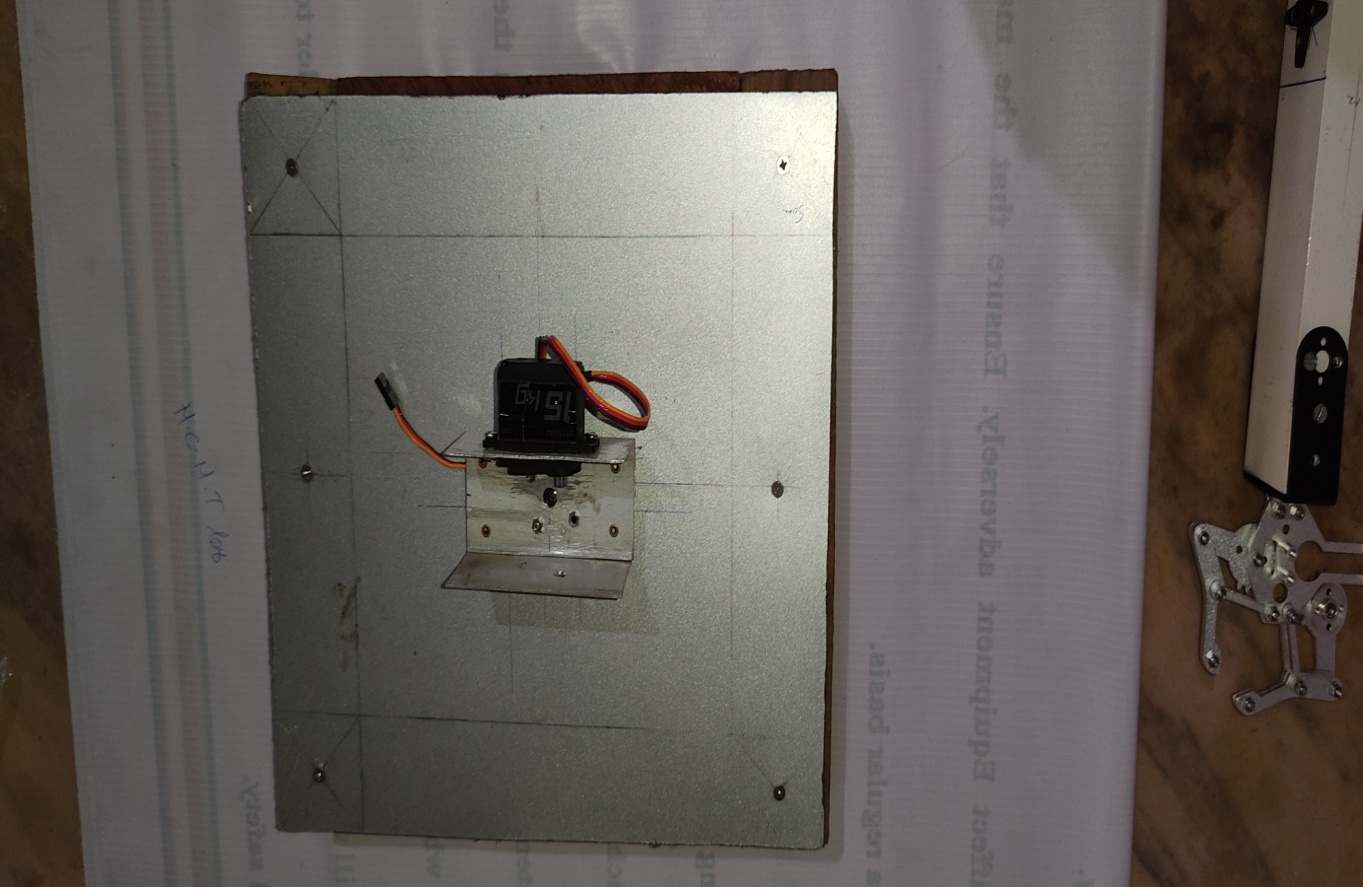


Figure 25. Top view of base portion

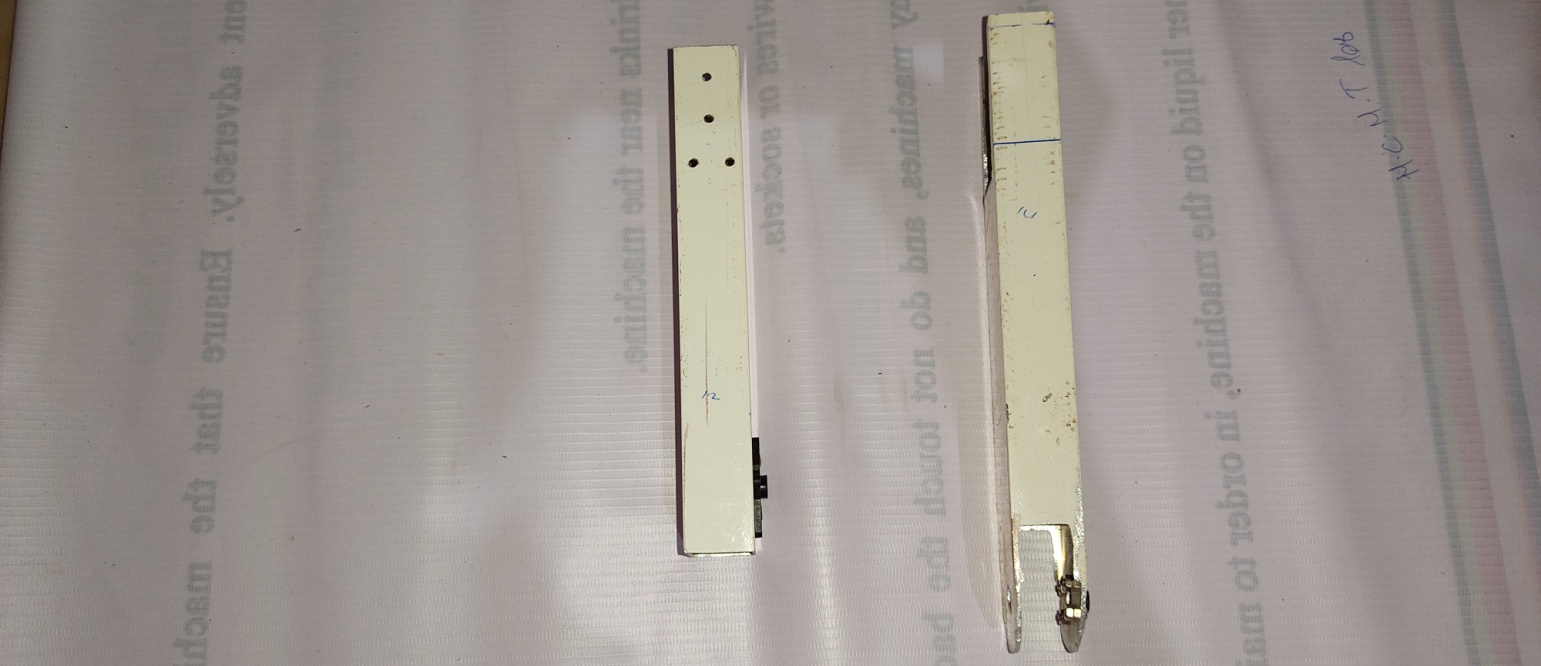
****

Figure 26. Aluminum robotic arms with cutting portions

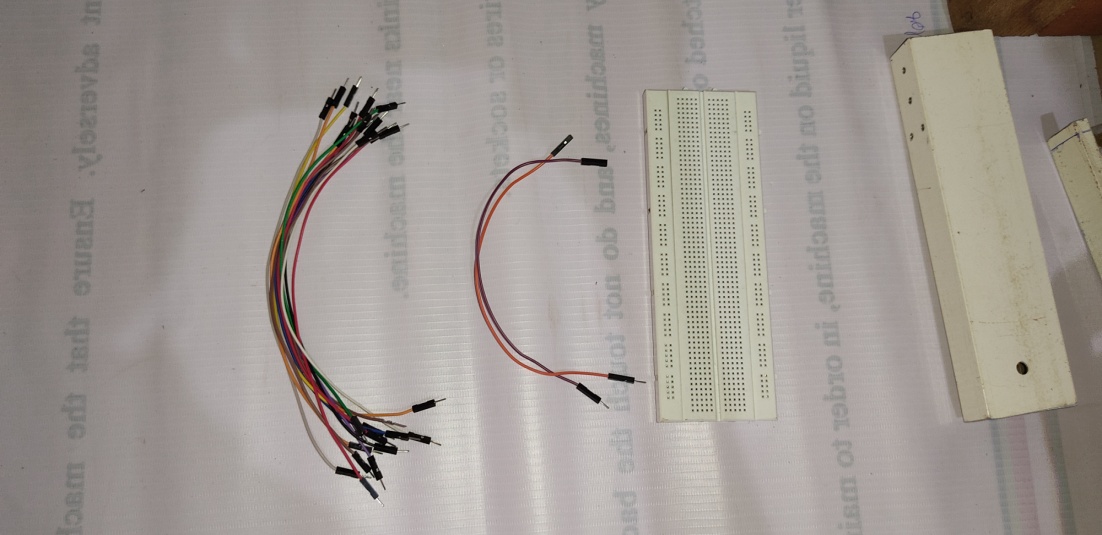


Figure 27. Bread board, jumper wires



Figure 28. Hons, screws and rivets

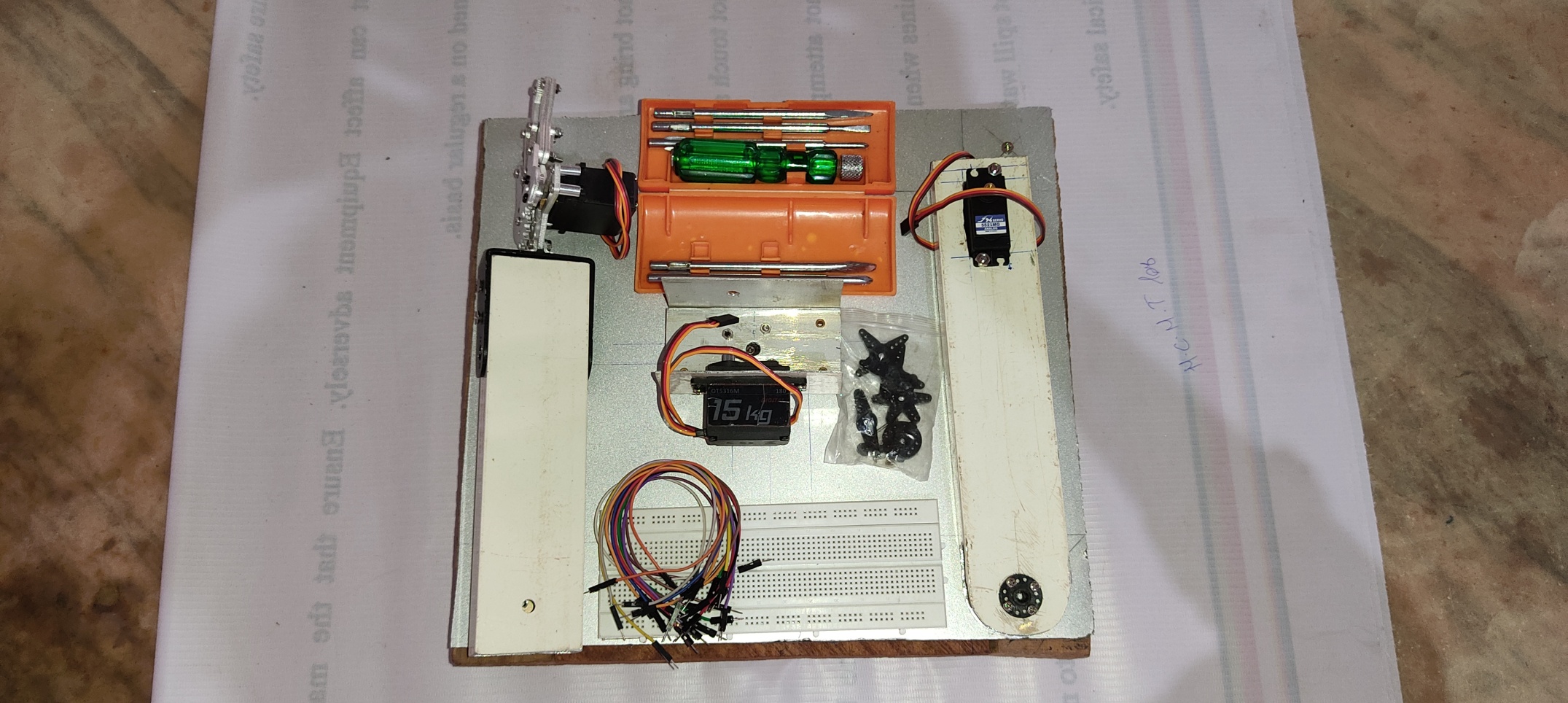


Figure 29. Parts to be Assembled

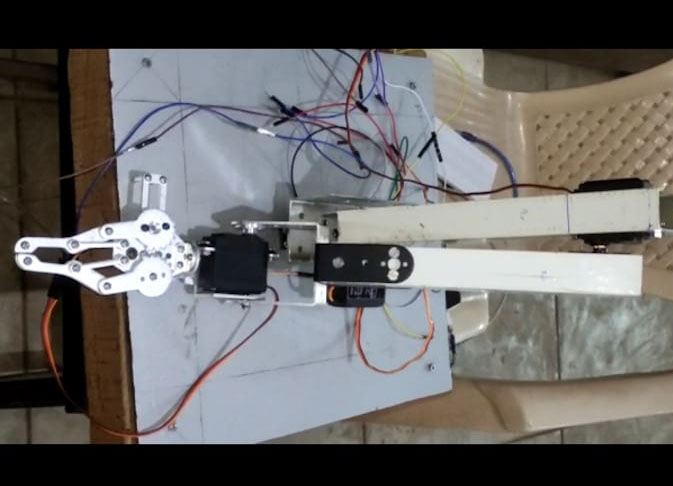


Figure 30. Robotic Arm Assembly front view

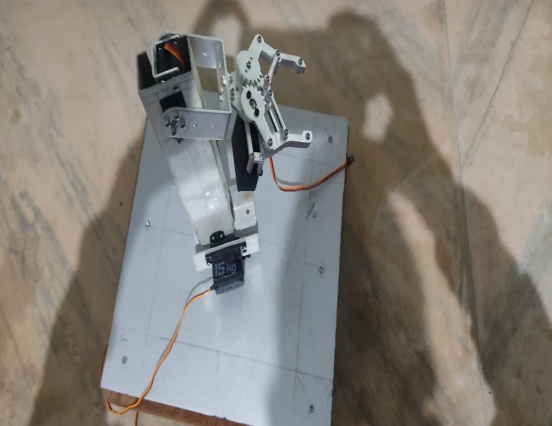
****

Figure 31. Robotic Arm Assembly top view

# **CHAPTER 7**

# **ADVANTAGES, DISADVANTAGES AND APPLICATIONS OF ROBOTIC ARM**

# **7.1 ADVANTAGES**

As a result of their ease of acquisition, collaborative robots have become the obvious answer for all companies in manufacturing and in other sectors who want to automate their processes.

They are the answer to traditional robots that were a preserve of the large companies with large reserves of cash and could afford to invest in an asset with a much longer rate of return on investment. Collaborative robotic arms do not need much of the space that was needed by traditional robots and have many other advantages that may lead a manufacturing entity to acquire them

* Precision and accuracy

One of the main advantages of a collaborative robotic arm in the factory is that they are highly precise and accurate. Robots work exactly how they are programmed to without any deviation hence their accuracy and precision. This then leads to improved quality of the products produced from the company and a reduction in defective products which also leads to a sings in cost of materials.

* Improved production capacity

Robotic arms help companies improve their production capacity. Robots are mechanical and do not require breaks and can work the whole day. They do not get sick or call in absent. In addition, they work faster and more accurately than any human worker. As a result, they will produce more per hour than any human worker.

* Fast and efficient

Speed and efficiency are the main competitive advantage that robots bring to an industrial operation. Robots are fast and are highly efficient in carrying out their tasks. They are accurate and precise; they can perform multiple tasks and can help organizations improve their financial status and market position.

* Improve factory working conditions

Work that is dull, dangerous, dirty or difficult is assigned and is suitable for robots. As a result, the workers who carried out this work previously, are now able to engage in other tasks in improved working conditions. Other tasks that are more engaging mentally and that are a much better to their health.

* Higher quality

Due to their high accuracy levels, robots can also be used to produce higher quality products which adhere to certain standards of quality, whilst also reducing the time needed for quality control

* Improved working environment

Industrial robots are often used for performing tasks which are deemed as dangerous for humans, as well as being able to perform highly laborious and repetitive tasks. Overall, by using industrial robots you can improve the working conditions and safety in your factory or production process. Robots don’t get tired and make dangerous mistakes, neither do they suffer from Reputative strain energy.

* Increased profitability

By increasing the efficiency of your production process, reducing the resource and time needed to complete it, and also achieving higher quality products, industrial robots can thus be used to achieve higher profitability levels overall, with lower cost per product.

* Longer working hours

Typically, people have to have breaks, get distracted and after time attention drops and pace slows. With a robot it can work 24/7, and keeps running at 100%. Typically, if you replace one person on a key process in a production line with a robot the output increases by 40% in the same working hours just because a robot has more stamina and never stops. Robots also don’t take holidays or have unexpected days off sick.

* Prestige

You set yourself at the cutting edge of your industry and wow your customers when they come to see you. As a marketing tool robot are fantastic, boost your brand image, and have often been used simply for the PR even if they don’t offer many benefits over a bespoke non-robotic system.

# **7.2 DISADVANTAGES OF ROBOTIC ARMS IN INDUSTRY**

* They need constant monitoring

Robots have to be monitored at all times to ensure that they do not get any mechanical faults which would cause them to stall. This could lead to losses to the company. With the advancements in technology, monitoring robots has become very easy. It can even be done remotely. The robots are able to send feedback of their processes via text to the operator. In case of a technical hitch, it can also be corrected remotely without having to stop the robot.

* Robots are not creative or innovative

Robots do not have the capacity to be creative or innovative. Robots can only complete tasks as they are instructed to through programming. Robots do not have the capacity to tell when consumer preferences are changing and what they want. Manufacturing companies therefore must always keep their eyes on the market and adapt to the market changes. Acquiring a robot enables companies the flexibility to change and adapt fast but that task cannot be delegated to the robots.

* Technical Know-how limitations

When acquiring a robot, often the acquiring organization does not have the necessary expertise. Collaborative robots have the capacity to perform numerous tasks within the factory and factory workers need the knowledge to make it work effectively. In addition, collaborative robots also require numerous add-ons and external fixtures that most buyers do not know about the aggregator or robot manufacturer ensures that they train the buyers of their robots. This training is an additional cost above the cost of buying the robot.

* Fear of job losses

Robots are seen to be taking over jobs that are already being carried out by human workers. With their cost effectiveness and mechanical ability, they can replace human workers in the factory once they have stronger capabilities. Collaborative robots are helping eliminate this fear. Rather than replacing the human worker, cobots are tools that assist the human worker in the factory.

* Capital cost

Whilst industrial robots can prove highly effective and bring you a positive ROI, implementing them might require a fairly high capital cost. That’s why, before making a decision we recommend considering both the investment needed and also the ROI you expect to achieve. Often the easiest way to get round this issue is to take out asset finance and the ROI of the robot more than pays for the interest on the asset finance.

* Expertise

Whilst industrial robots are excellent for performing many tasks, as with any other type of technology, they require more training and expertise to initially set up. The expertise of a good automation company with a support package will be very important. To minimize your reliance on automation companies you can train some of your engineers on how to program robots, but you will still need the assistance of experienced automation companies for the original integration of the robot.

# **7.3 APPLICATIONS OF ROBOTIC ARM**

* Arc Welding

Arc welding, or robot welding, became commonplace in the 1980s. One of the driving forces for switching to robot welding is improving the safety of workers from arc burn and inhaling hazardous fumes.

* Spot Welding

Spot welding joins two contacting metal surfaces by directing a large current through the spot, which melts the metal and forms the weld delivered to the spot in a very short time (approximately ten milliseconds).

* Materials Handling

Material handling units are utilized to move, pack and select products. They also can automate functions involved in the transferring of parts from one piece of equipment to another. Direct labour costs are reduced and much of the tedious and hazardous activities traditionally performed by human labour are eliminated.

* Machine Tending

Robotic automation for machine tending is the process of loading and unloading raw materials into machinery for processing and overseeing the machine while it does a job.

* Painting

Robotic painting is used in automotive production and many other industries as it increases the quality and consistency of the product. Cost savings are also realized through less rework.

* Picking, Packing and Palletizing

Most products are handled multiple times prior to final shipping. Robot picking and packaging increases speed and accuracy along with lowering production costs.

* Assembly

Robots routinely assemble products. Eliminating tedious and tiresome tasks. Robots increase output and reduce operational costs.

* Mechanical Cutting, Grinding, Deburring and Polishing

Building dexterity into robots provides a manufacturing option that is otherwise very difficult to automate. An example of this is the production of orthopaedic implants, such as knee and hip joints. Buffing and polishing a hip joint by hand can normally take minutes while a robot can perform the same function in just a few minutes.

* Gluing, Adhesive Sealing and Spraying Materials

Sealer robots are built with numerous robotic arm configurations that enable the robot to apply adhesives to any type of product. The primary benefit in this application is increased quality, speed and consistency of the final product.

* Other Processes

These include inspection, water jet cutting and soldering robots

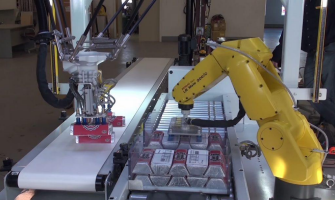


Figure 32. Inspection robot



Figure 33. Water jet cutting robot

Figure 34. Robotic palletizing system



Figure 35. Robotic palletizing system



Figure 36. Painting robot

# **RESULT**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S.NO** | **Cartesian coordinates**  **X, Y, Z** | **Angle 1** | **Angle 2** | **Angle**  **3** | **Angle**  **4** | **Error** |
| **1** |  |  |  |  |  |  |
| **2** |  |  |  |  |  |  |
| **3** |  |  |  |  |  |  |
| **4** |  |  |  |  |  |  |
| **5** |  |  |  |  |  |  |

Table 1. RESULT TABLE

# **CONCLUSION**

In this project, we demonstrated the successful utilization of inverse kinematic solutions for a 4-degree-of-freedom serial robot, employing optimization algorithms in working together with computer vision techniques. By extracting coordinate points from the vision system, we were able to upload them to Arduino software to accurately locate objects of various colours, such as red, blue, and green. This information facilitated precise movement of the robotic arm to desired locations, showcasing the practical application of advanced technologies in robotic manipulation tasks. Through this project, we have not only achieved a seamless integration of optimization and vision methodologies but also illustrated their efficacy in enhancing the functionality and adaptability of robotic systems in real-world scenarios.

Inverse kinematic solution of robot have obtained for different task space locations and improved the accuracy of the movement of the arm to the desired location with the precision of +/- 0.2 mm.

The use of vision sensors and camera’s to track the position and orientation of the end effector has enabled robot’s to perform complex tasks in unstructured environments, enabling safer and faster production, increased precision and repeatability of task’s, and more effective use of resources.

# **REFERENCES**

**[1]** Fu, K.S (King Sun) Robotics: Controlling, sensing, Vision and Intelligence international edition 1987

**[2]** Ondrej Hock and Jozef Sedo: “Inverse kinematics using transportation method for robotic arm”, ELEKTRO, 2018

**[3]** Jing Huang , Xianlun Wang , Dongsheng Liu and Yuxia Cui , A New Method for Solving Inverse Kinematics of an

Industrial Robot, 2012 International Conference on Computer Science and Electronics Engineering Year: 2012 , Volume: 3

**[4]** Sandeep Kumar, Aman Balyan and Manvi Chawla, Object Detection and Recognition in Images, Volume 5, Issue 4, ISSN:

2321-9939 IJEDR1704155 International, 2017

**[5]** Shashank Prasad; Shubhra Sinha; “Real-time Object Detection and Tracking in an Unknown Environment”, World Congress on Information and Communication Technologies, 2011

**[7]** Perez and B. Granger, IPython: a system for interactive scientific computing, Computing in Science & Engineering, vol.

9, no. 3, pp. 21–29, 2007.

**[8]** Bradski, The OpenCV library, DOCTOR DOBBS JOURNAL, vol. 25, no. 11, pp. 120–125, 2000.

**[9]** P.Ramachandran, Performance of various Python implementations for solving the 2D Laplace equation., 1991

**[10]** Brian Thorne and Raphaël Grasset, Richard Green, Using Python in Computer Vision: Performance and Usability, HIT Lab NZ, 11 November 2014.

**[11]** Far´es Jalled and Ilia Voronkov, Object Detection Using Image Processing, Moscow Institute of Physics & Technology, Department of Radio Engineering & Cybernetics, 23 Nov 2015.

**[12]** Joseph Schlecht and Björn Ommer, Contour-based Object Detection”, Interdisciplinary Center for Scientific Computing, University of Heidelberg, Germany, 2011.

**[14]**D. Whitney, “Resolved motion rate control of manipulators and human prostheses,” IEEE Transactions on Man Machine Systems, vol. 10, no. 2, pp. 47–53, 1959.View at: Publisher Site | Google Scholar

**[15]**I. Dulęba and M. Opałka, “A comparison of Jacobian-based methods of inverse kinematics for serial robot manipulators,” International Journal of Applied Mathematics and Computer Science, vol. 23, no. 2, pp. 373–382, 2013.

**[16]**S. Miyata, S. Miyahara, and D. Nenchev, “Analytical formula for the pseudoinverse and its application for singular path tracking with a class of redundant robotic limbs,” Advanced Robotics, vol. 31, no. 10, pp. 509–518, 2017.

**[17]**J. Karpińska and K. Tchoń, “Performance-oriented design of inverse kinematics algorithms: extended jacobian approximation of the jacobian pseudo-inverse,” Journal of Mechanisms and Robotics, vol. 4, no. 2, Article ID 021008, 2012.

**[18]**C. Wampler, “Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 15, no. 1, pp. 93–101, 1985

**[19]**Y. Nakamura, H. Hanafusa, and T. Yoshikawa, “Task-priority based redundancy control of robot manipulators,” The International Journal of Robotics Research, vol. 5, no. 2, pp. 3–15, 1987.

X. Wang, D. Zhang, and C. Zhao, “Inverse kinematics of a 7R 5-DOF robot with nonspherical wrist based on transformation into the 5r robot,” Mathematical Problems in Engineering, vol. 2017, Article ID 2074137, 12 pages, 2017.

**[20]**S. R. Buss and J.-S. Kim, “Selectively damped least squares for inverse kinematics,” Journal of Graphics Tools, vol. 10, no. 3, pp. 37–49, 2005.

**[21]**L. M. Phuoc, P. Martinet, S. Lee, and H. Kim, “Damped least square based genetic algorithm with Ggaussian distribution of damping factor for singularity-robust inverse kinematics,” Journal of Mechanical Science and Technology, vol. 22, no. 7, pp. 1330–1338, 2008.

# **APPENDIX**

# **Code for optimization**

% pos = [L2\*cosdd(th(1))\*cosdd(th(2)) + L3\*cosdd(th(1))\*cosdd(th(2))\*cosdd(th(3)) - L3\*cosdd(th(1))\*sindd(th(2))\*sindd(th(3))

%  L2\*cosdd(th(2))\*sindd(th(1)) + L3\*cosdd(th(2))\*cosdd(th(3))\*sindd(th(1)) - L3\*sindd(th(1))\*sindd(th(2))\*sindd(th(3))

%                             L2\*sindd(th(2)) + L3\*cosdd(th(2))\*sindd(th(3)) + L3\*cosdd(th(3))\*sindd(th(2))];

pos=[L2\*cosd(th(1))\*cosd(th(2)) + L3\*cosd(th(1))\*cosd(th(2))\*cosd(th(3)) - L3\*cosd(th(1))\*sind(th(2))\*sind(th(3))

 L2\*cosd(th(2))\*sind(th(1)) + L3\*cosd(th(2))\*cosd(th(3))\*sind(th(1)) - L3\*sind(th(1))\*sind(th(2))\*sind(th(3))

                       L1 - L2\*sind(th(2)) - L3\*cosd(th(2))\*sind(th(3)) - L3\*cosd(th(3))\*sind(th(2))]

fmin=sqrt((X-pos(1))^2 + (Y-pos(2))^2 + (Z-pos(3))^2 )

end

# **Code for object detection through colour**

tic

clear all;

clc;

l1 = 8.5;

l2= 18.5;

l3 = 17.5;

nl =3;

X = input('X-coordinate-: ');

Y = input('Y-coordinate-: ');

Z = input('Z-coordinate-: ');

ang =zeros(1,nl);

A = [];

b = [];

Aeq = [];

beq = [];

lb = [0, 0, 0];

ub = [180,180,180];

  [ang1,fval,exitflag,output]=fmincon(@(th) calfun(th,X,Y,Z,nl,l1,l2,l3),ang,A,b,Aeq,beq,lb,ub)

%  [ang1,fval,exitflag,output] = fminsearch(@(th)calfun(th,X,Y,Z,nl,l1,l2,l3),ang,optimset('MaxIter',20000000,'MaxFunEvals',20000000));

th=ang1