AN-NAJAH NATIONAL UNIVERSITY
FACULTY OF ENGINEERING
AND INFORMATION TECHNOLOGY
COMPUTER ENGINEERING DEPARTMENT

Hardware Graduation Project:
Rubik Cube Solver

SUBMITTED BY

Burhan Azem

Bassam Tuffaha

SUPERVISOR

Dr. Samer Arandi

Presented in partial fulfillment of the requirements for the Bachelor's degree in Computer Engineering.

Semester 2, 2024

Dedication

We would like to thank everyone who supported us, including friends, family, Teachers.
This work wouldn't be done without you.

Acknowledgment

To everyone who contributed to making this project possible, we extend our heartfelt gratitude. We thank our esteemed An-Najah National University and the teachers from whom we learned so much, enabling us to grow and serve our country and community.

Special thanks to our supervisor, Dr. Samer Arandi, who dedicated his time to support us and consistently showed his trust in our abilities.

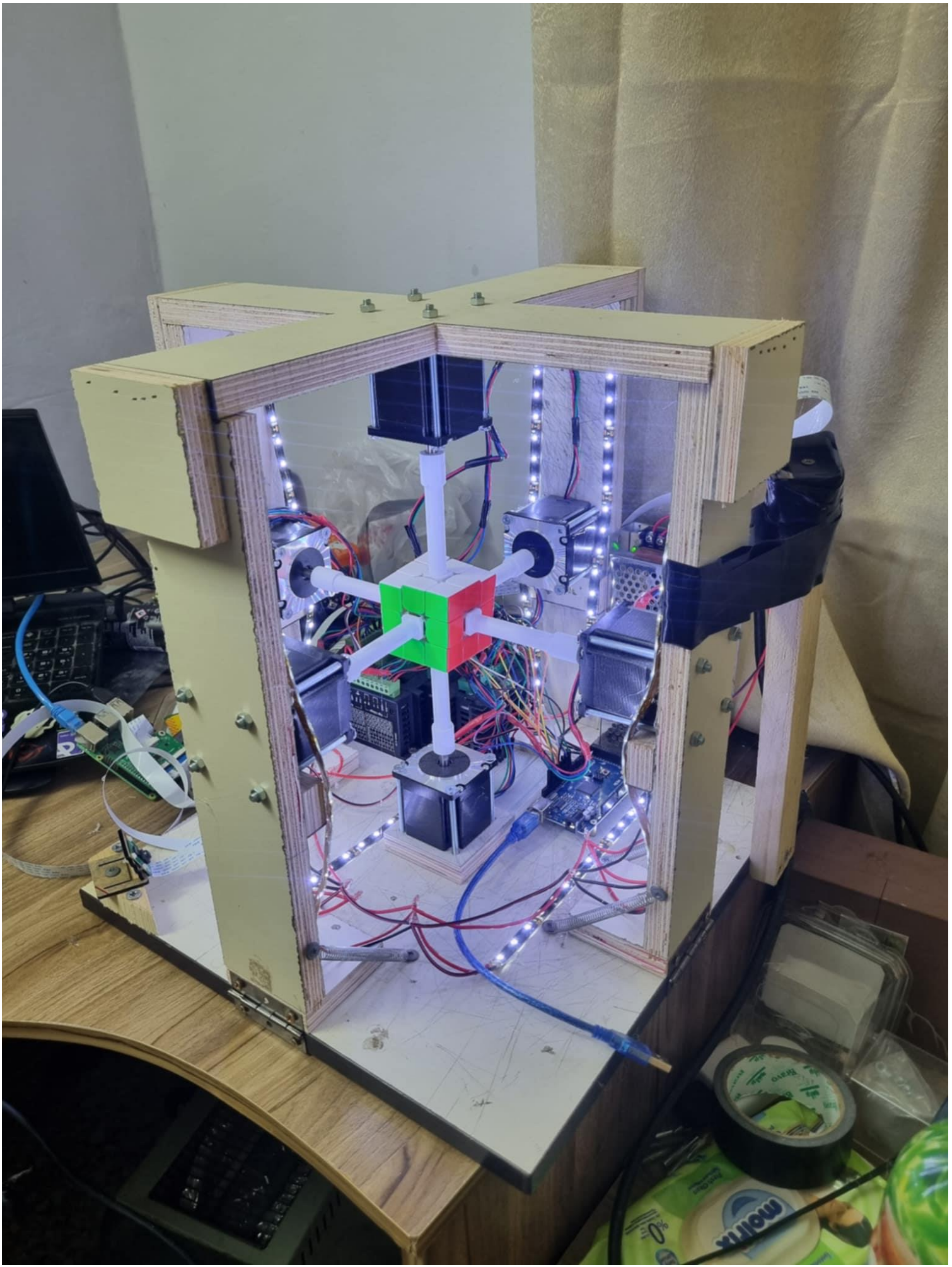
To our families and friends, who stood by us through thick and thin, offering unwavering support and assistance, we offer a sincere and heartfelt "Thank You."

DISCLAIMER

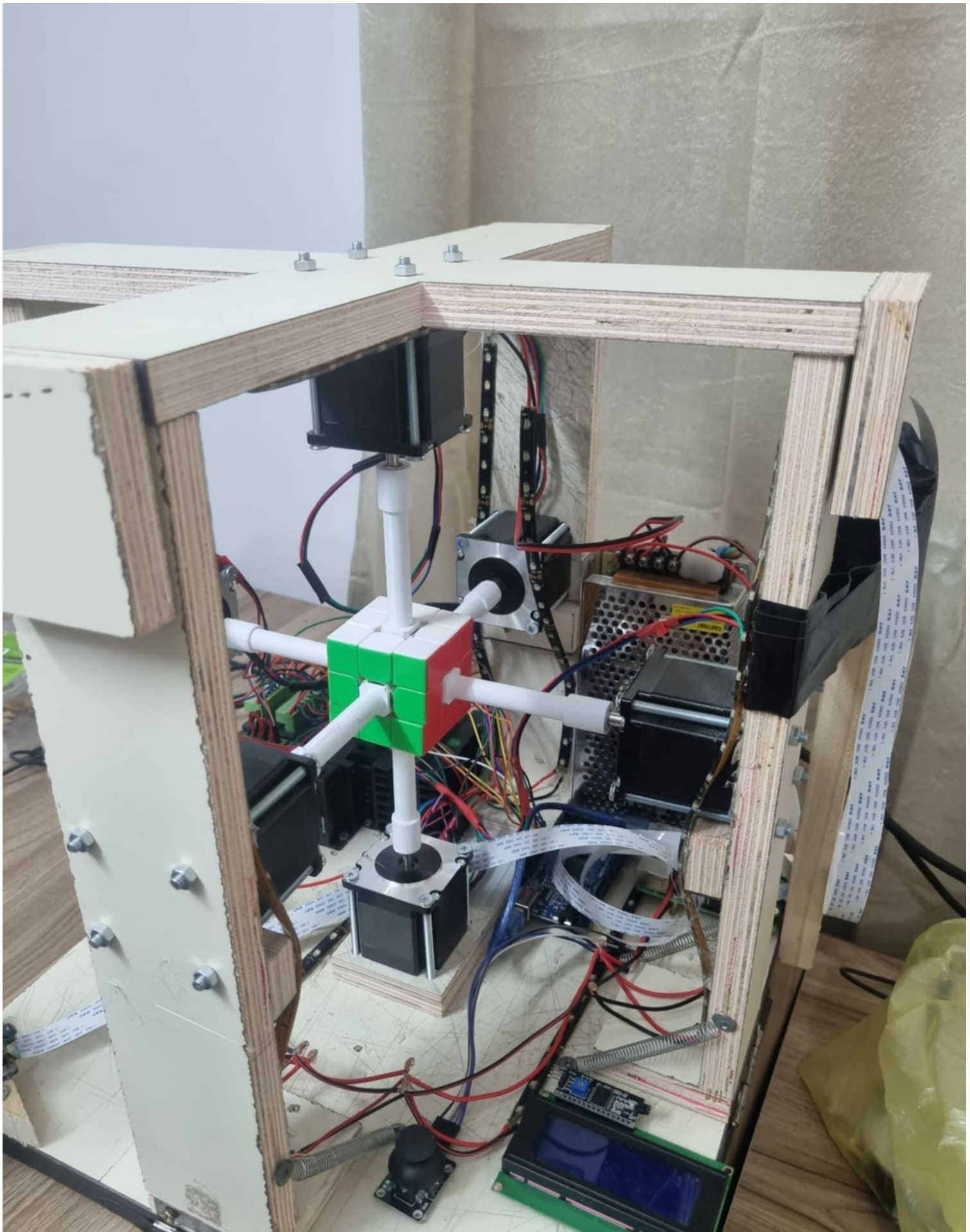
This report was written by student(s) at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of the assessment, and it may contain language as well as content errors. The views expressed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

Contents

Images of Module:



Images of the Module:



Abstract

The importance of this project lies in showcasing the capabilities of modern robotics and artificial intelligence in solving complex puzzles and serving educational purposes. We aim to benefit three main groups with this project:

1. People who want to learn how to solve a Rubik's Cube step by step.
2. Students in AI and algorithms courses: To explore the differences in behavior among various algorithms.
3. Individuals seeking entertainment and a unique way to impress their friends.

Our Rubik Cube Solver “CubeMaster” with it’s main parts Raspberry Pi and Arduino Mega setup, two cameras. The Raspberry Pi works on image processing and figuring out the solving steps, while the Arduino Mega runs six stepper motors to move the cube and cameras to scan the Rubik Cube.

Users can change how fast and in what way the cube is solved through using four different algorithms “ Begginers method”, “CFOP”, “Kociemba” and “Depth-First Search (DFS)” which also has a mode to teach you how to solve the cube step by step. CubeMaster is great because it solves the cube quickly and makes learning fun. Adding to this, we have an LCD screen that shows a timer, the number of moves taken to solve the cube, and other information and joystick to move between modes.

Although there is a similar project, our CubeMaster project sets itself apart by emphasizing advanced image processing to achieve fully automatic detection of the Rubik's Cube's start point. We also will significantly reduce the time it takes to solve the cube to just a few seconds, far quicker than the typical 40-60 seconds seen in the other project. Moreover, we're dedicated to aiding learners and students interested in Rubik's Cube solving and AI algorithms through our control tool (LCD screen and joystick), making their educational journey simpler and more fun

1. Introduction

1.1 Problem Statement

Our "MasterCube" solver project aims to help many groups of society those search about learn and fun, the most prominent of which are:

1. People that need help to learn solving rubik cube step by step, which makes it easier for them to master how solving rubik cube physically and by using different modes and techniques.
2. Students and teachers of Artificial Intelligence and Algorithms courses, by clarifying the differences between types of algorithms and how each algorithm arrives at a solution.

   To achieve this, the project will implement basic AI algorithms, including:

   - **Blind Search Algorithms**:
     - Depth-First Search (DFS)
   - **Greedy Search Algorithms**:
     - Begginers method (Greedy Best-First Search)
   - **Optimal Search Algorithms**:
     - CFOP & Kociemba (A-star)

   These algorithms will help demonstrate the various approaches to problem-solving and their effectiveness in different scenarios.

1.2 Significance

The main goal of the project is to build Rubik cube solver that is able to solve the Rubik cube in just few seconds and simplify learning process of how to solve Rubik cube following the most optimal algorithms step by step. In addition to that “Master Cube” will help academic teachers to simplify the difference between algorithms in physical way.

1.3 Objectives and Scope

The purpose of the project “Master Cube” is to provide custom Rubik cube solver following our standards and suitable to achieve our objectives, and therefore requires the availability of the following services:

1. The ability to solve the Rubik cube starting from any state.
2. The ability to capture the initial state automatically without any help by using two cameras.
3. The ability to choose between three modes using joystick and LCD screen:
   - **Fast mode**: which will solve the rubic cube in fast possible way.
   - **Learning & Fun mode**: which will make the learner to choose between three algorithms to solve the rubik and move through the solution step by step using joystick .
   - **Teaching mode:** Enables educators to demonstrate the differences between AI algorithms by solving the Rubik's Cube using three different methods.
4. Ability to know the current solution state while motors solving it by representing it on the LCD screen.

1.4 Report Organization

- **Second Chapter**: This chapter covers key subjects we've previously studied, as well as insights from external courses. It also addresses the primary constraints and obstacles encountered during the project.
- **Third Chapter** (Literature Review): Here, we discuss the inspiration and rationale behind choosing our project idea, including our goals for undertaking this project.

  Fourth Chapter (Methodology): This chapter delves into the mindset and approach we adopted while building the application. It also outlines the features we offer and the technologies we employed.
- **Fifth Chapter**: We present our results, share the lessons learned from the project, and explore potential future developments that could further evolve our work.

2. Constraints and Earlier Coursework

2.1 Constraints

2.1.1 Inexperience

We developed numerous modest demos and tests to verify the feasibility of our concept. However, these tests consumed time that could have been allocated to other tasks. Despite the extensive testing, research, and time invested, we still had to make certain decisions that are deemed safe risks because we couldn't test every component and couldn't afford for the overall project to fail.

2.1.2 Lack of funds

Some options required additional money that was unaffordable, especially given our budget, making the project more expensive.

2.1.3 Lack of Mechanical knowledge

The project involves many mechanical aspects that we only partially understood, but even after all of our study, anexperienced mechanical engineer was required to answer some problems and arrange the execution effectively. Considerable 3D printed pieces required some balance and mechanical understanding to repair; thus, the conceptsand original drawings were brought to the engineer students for modification.

2.1.4 Lack of Tooling

Some specific tools were necessary to cut, prepare, and assemble all of the pieces. Since these tools were either unavailable or too expensive for home use, an expert was called in to assist. Additionally, some pieces had to be printed using a 3D printer. A carpenter was also employed to help prepare the exterior model, which was made of wood.

2.2 Earlier coursework

2.2.1 Microcontrollers and PIC

These classes covered the fundamentals of the Arduino, including as basic serial communication and controlling steppers and servo motors.

2.2.2 Networks and Communication and Wireless

We used serial communication between an Arduino and a Raspberry Pi..

2.2.3 Critical Thinking and Research Skills

This course has provided me how to conduct research and write a report.

3. Literature Review

When building this project, our focus was on creating a custom Rubik's Cube solver that could be employed in several areas such as learning, academia, and entertainment. Our goal was to help people in these areas in the best possible way. To achieve this, we thoroughly examined the workings of the Rubik's Cube to make it useful in real-life scenarios. This can be explained as follows:

3.1 The Regular Learning: Solving Rubik Cube

In the normal situation, the new learners will buy a Rubik cube and will try to learn from the internet how to solve the Rubik cube in a fast way, covering all scenarios can Rubik cube be at initial state, but this way it’s boring, and in most scenarios, the learner will fail to continue.

The statistics reveal that only 1 in 20 people who own a Rubik's cube can actually solve it, Solving the Rubik's cube is a challenge, but one that will have a great impact on the child, so there is a need to do something to help in this manner.

Also, in the AI and Algorithms classes, there is a problem related to how students can see the difference between the algorithms in a way that grabs their attention.

So was our solution here to create custom Rubik cube solver that will be the key to all of the previous scenarios and to be as luxury technology to impress all who is will see it and to be evidence to our technical capabilities as computers engineers.

3.2 Similar Projects

There is was a similars projects, our CubeMaster project sets itself apart by emphasizing advanced image processing to achieve fully automatic detection of the Rubik's Cube's start point. We also will significantly reduce the time it takes to solve the cube to just a few seconds, far quicker than the typical 40-60 seconds seen in the other project. Moreover, we're dedicated to aiding learners and students interested in Rubik's Cube solving and AI algorithms through our project’s joystick, making their educational journey simpler and more fun.

4. Methodology

This section contains detailed information about the techniques and methods we used to develop the project, from designing and assembling the mechanical structure to controlling the motors, holding all the other parts as well as how they are linked together to produce the final product.

4.1 Choosing the idea

In the beginning, we did not know what we wanted to do as a graduation project, and there were many ideas on the Internet and many ideas that had been created in the past years by colleagues in the Department of Computer Engineering, and therefore we were rejecting the idea of repeating a project, so we wanted to do a project to be distinguished to come with new ideas or to add to new features to existing project, that's why we took a lot of time to come up with the idea of the project.

When we were sitting in the university and discussing, the main idea we were looking for is making a project that we would be proud of, but also helps the community and leaves and mark, so we started thinking how we can make a project that will impress the people and in the same time help them real cases

After we agreed on the idea of the project, we consulted Dr.Samer Arandi about the initial idea and he is encouraged and gave us some ideas to improve to general idea by add joystick hand and give us some ideas how we can start designing the project using AutoCAD step by step until we arrive the desired outcome.

4.2 Mechanical Part

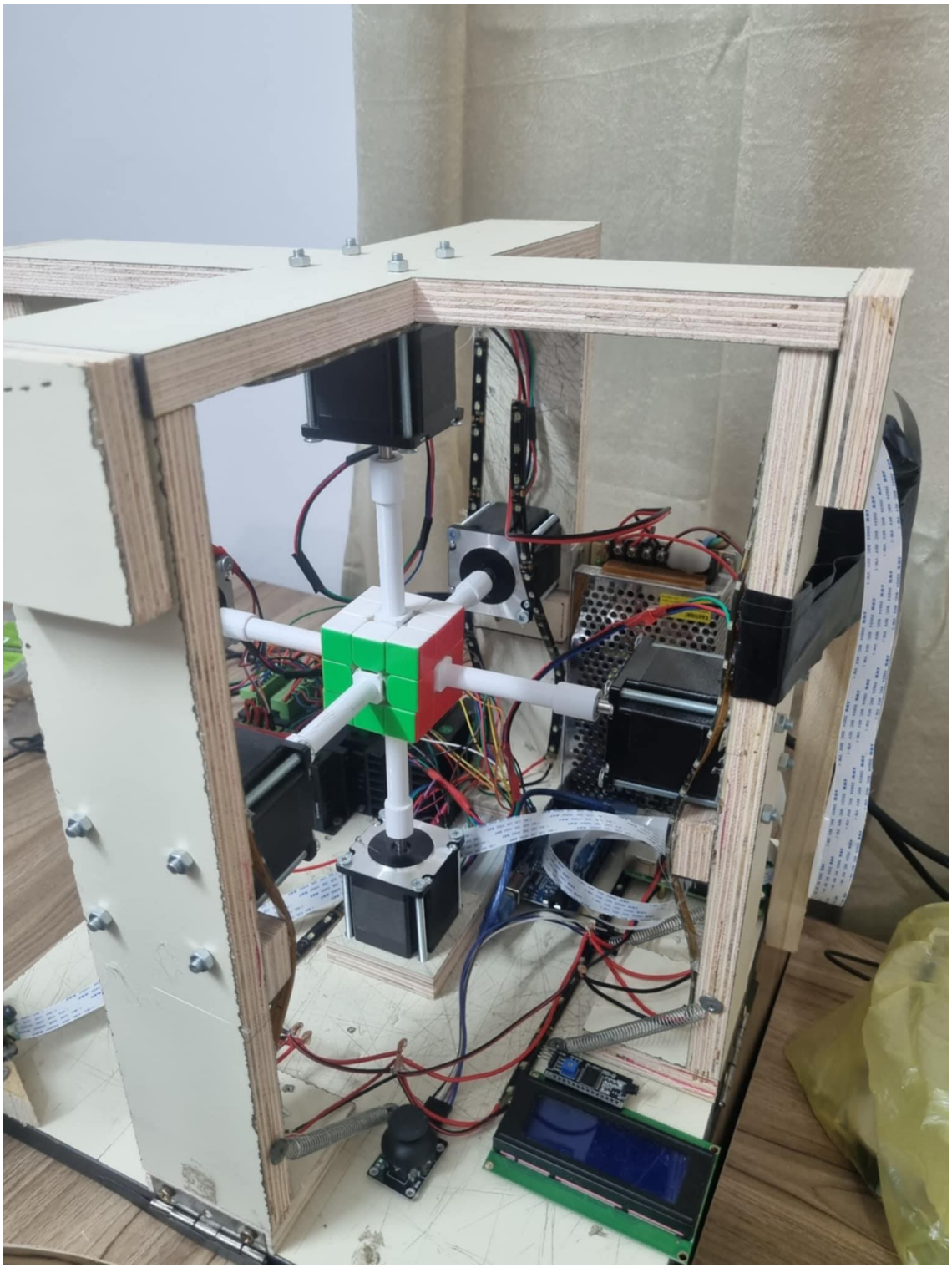
The mechanical component of the project consists of six motors controlled by drivers and an Arduino, which it will solve the Rubik cube by implementing the solutions produced by a Raspberry Pi, moving from one state to another as quickly as possible. This requires the following steps to be taken:

4.2.1 Mechanical Design

In the first step, we worked on designing the whole project structure using AutoCAD, including dimensions positions part in best possible way that will make the design more reliable and open to extensions, and also we needed some 3D-printed bodies to help support the model.

4.2.2 Assembling

It is the part that demands the use of bolts and drill bits to put the components together, and it must be done correctly for the project to function properly.

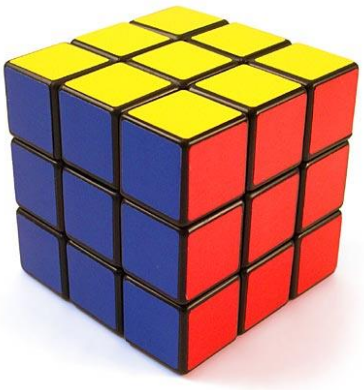
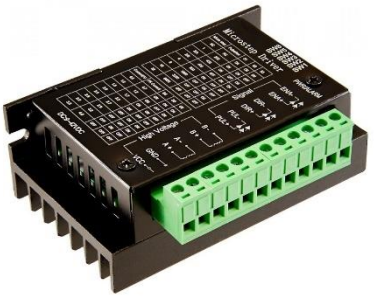


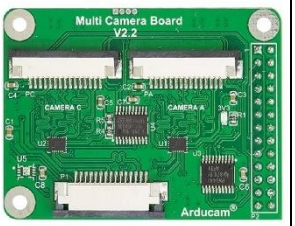
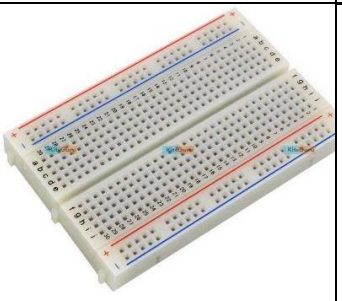
After we put everything in its place, as shown in the above figure, we began to work on the motors to ensure proper movement and so that there would be no mechanical problems that could disrupt us in the future, such as balance and weight distribution evenly so as not to cause vibrations that would sabotage the project. We also made sure the rubik cube can be moved so we can change its state as we want.

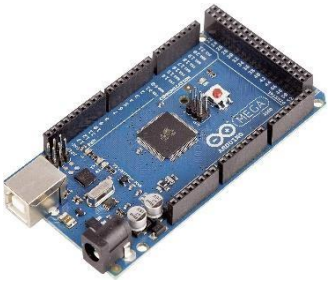
4.2.3 Parts

Item Name	Item Image	Quantity
Flexible Flat Cables (FFC)		3m length
Raspberry pi camera v2 × 2		2
Stepper Motor nema 23 * 6		6
Hinge		3

Camera Module Mount Holder		3

Rubik cube		**1**
Stepper moter driver b600 * 6		6

LCD		4
Arducam Multi Camera Adapter	ArduCam  	1
Wires		6
Bread board		1

power supply 12.5A		1
Arduino Mega		1
Raspberry pi 4 model b		3
Joystick		1

		1

Table 4.4: Parts

4.2.4 Rules and Functionality

* **Raspberry Pi 4 Model B**

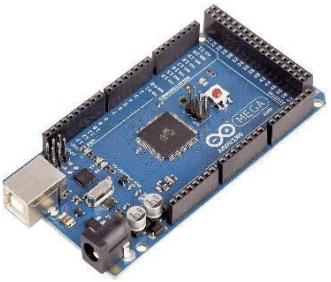
- Role: Central processing unit.

- Function: Runs the main software for image processing, cube state recognition, and solving algorithm computation. Sends control commands to the Arduino.

* **Arduino Mega**

- Role: Motor controller and modes management.

- Function: Receives movement commands from the Raspberry Pi and controls the stepper motors to manipulate the Rubik's Cube, and manage mode state.



* **Stepper Motors NEMA 23 × 6**

- Role: Physical cube manipulation.

- Function: Each motor is responsible for rotating a face of the Rubik's Cube. The motors perform precise movements to execute the solution steps.

* **Stepper Motor Driver B6600 × 6**

- Role: Motor control.

- Function: Drives the stepper motors based on signals received from the Arduino. Ensures accurate and controlled movements.

* **Raspberry Pi Camera V2 × 2**

- Role: Image capture.

- Function: Captures images of all six faces of the Rubik's Cube. Provides the necessary visual data for the Raspberry Pi to determine the cube's state.

*** Power Supply**

- Role: Provide power.

- Function: Supplies electrical power to the Raspberry Pi, Arduino, stepper motors, and other components.

* **Joystick**

- Role: Manual control input.

- Function: Allows manual control or input for initializing or troubleshooting the Rubik's Cube solver.

* **LCD**

- Role: Display output.

- Function: Displays the status, progress, and possibly the current state of the Rubik's Cube and the steps being performed.

2.1.1 Workflow Overview

1. Image Capture: The Raspberry Pi Camera V2 modules capture images of the Rubik's Cube.

2. Image Processing: The Raspberry Pi processes these images to determine the cube's state.

3. Solution Computation: The Raspberry Pi calculates the solution using a solving algorithm.

4. Command Transmission: The Raspberry Pi sends movement commands to the Arduino Mega based on the selected mode.

5. Physical Manipulation: The Arduino controls the stepper motors via the B6600 drivers to rotate the cube faces as needed.

6. Power Management: The power supply ensures all components have the necessary power.

7. User Interface: The joystick allows for manual control to modes and move between them, and the LCD provides visual feedback on the solver’s status, like current state, timer, current mode

3. Results And Problems Discussion

3.1 Drivers Synchronization

A technical problem we encountered was motor synchronization. Ensuring that all six motors operated in perfect harmony was crucial for the accurate and efficient solving of the Rubik's Cube. Initially, we faced issues where the motors would become desynchronized due mile centimeters in positions, causing misalignment and jamming during the cube's rotation. This required extensive debugging and the implementation of a more robust control algorithm to synchronize the motors precisely, ensuring smooth and coordinated movements throughout the solving process.

3.2 Designing the structure wood

There was also a problem in how to design and imagine the structure of the rubik cube solver in a way that allowed the six motors to rotate smoothly and absorb vibration. It took a while, but we eventually designed it appropriately using AutoCAD.

3.3 Broken parts

We encountered some situations where some parts broke and had to replace them, like the camera holder and the axis of rotation (3D printed part used to rotate the cube faces in angel 90, -90, 180)

3.4 Image processing accuracy

We had a big problem capturing the initial state of the Rubik's Cube with cameras due to the hidden squares of the cube. So, we resorted to using mirrors to capture the hidden squares and improve the lighting.

3.5 Final Results

The end result is a fantastic project that works well in the majority of the scenarios we tested and meets all of our objectives.

4. Conclusion

4.1 Summery

We developed a mechanical device to solve the Rubik's cube in a custom way that makes it useful and applicable in educational, academic, and fun areas. We took care to build it in a way that allows for additional features. For future work, we have taken a step to integrate technology into the world of puzzles to make it more fun and useful. Despite the constraints in time, resources, and hardware expertise, we successfully delivered a functional project with substantial potential for improvement. Our long-term vision is for this project to be the first in a series of initiatives aimed at enhancing and expanding the capabilities of such devices.

4.2 Improvements

There is a need for some improvements by adding metal rotation axis (used to rotate the cube faces) instead of plastic 3D printed, we should take care more about motors positions accuracy because the sequence rotations requires smooth, synchronous moves, need to reduce the hidden squares area by using smaller parts in the structure, and add LED s to enhance the lighting for better image processing.

4.3 Future Work

At overall the project is perfect for its current job, but one aspect of the future work is to make the rubik cube solver compatible with all Rubik's cube sizes, such as like 4*4, 5*5, 6*6 by making the rotation axis with variable length, and make better image processing experiment by using cameras with higher resolution.

4.4 Outcome

We have developed an efficient Rubik's Cube solver that is user-friendly and versatile. This solver can be utilized in various settings, including educational institutions, puzzle-solving competitions, and personal use. It is designed to be accessible for anyone, providing a solution that saves time, effort, and ensures accuracy.

Bibliography

Kacemba: Herbert Kociemba's optimal Rubik's Cube solver - Cube Explorer (ruwix.com)
raspberry pi 4 model b datasheet: raspberry-pi-4-datasheet.pdf (raspberrypi.com)
Arduino Mega datasheet: A000067-datasheet.pdf (arduino.cc)
Stepper Motors NEMA 23 datasheet: NEMA 23 Stepper Motor Datasheet, Specs & Applications (components101.com)
Stepper Motor Driver B6600 datasheet: TB6600-Stepper-Motor-Driver-BM3D-v1.1.pdf (bulkman3d.com)
Raspberry Pi Camera V2 datasheet: 2056179.pdf (farnell.com)