Project and Professionalism

(6CS020)

A1: Project Proposal

Student Id : 2049822

Student Name : Sashank Dulal

Supervisor : Sangay Lama

Submitted on : Tuesday, November 16,2021

Acknowledgements

My sincere gratitude goes towards University of Wolverhampton and Herald College Kathmandu for this programme and course availability. Secondly, I would like to thank module leaders Mr. Nirmal Thapa and Mr. Biraj Dulal for such experience regarding the project and for much needed guidance. Finally, special thanks to our supervisor Mr. Sangay Lama for his time and effort to make this project better with his knowledge and support.

Table of Contents

Table of Figures

1 Statement of Proposal

1.1 Project Title

The name of the proposed project is “Chessmate”.

1.2 Project Statement

Chessmate is a chess playing bot that engages in a chess match with the user on a physical chess board.

1.3 Academic Question

- How will the bot get the best possible movement for the move produced by the user?
- How will the chess pieces be moved using the arm?
- What will be used to track the movement of chess pieces?

1.4 Aims and Objectives

1.4.1 Aims

- Investigate the feasibility of different robotic methods for movement of small objects in a confined space.
- Perform research on different algorithms used for moving the robots within a fixed area.
- Construct a robot using research findings that will be able to play chess against a user opponent.

1.4.2 Objectives

- A robotic system will be designed and developed using the analysis of research findings.
- Implement different image processing algorithms to detect and track movements.
- The robot will be made in such a way that it will be communicating with custom made/ open-source API that can generate chess moves for any given move of the user.

1.5 Artefacts to be developed

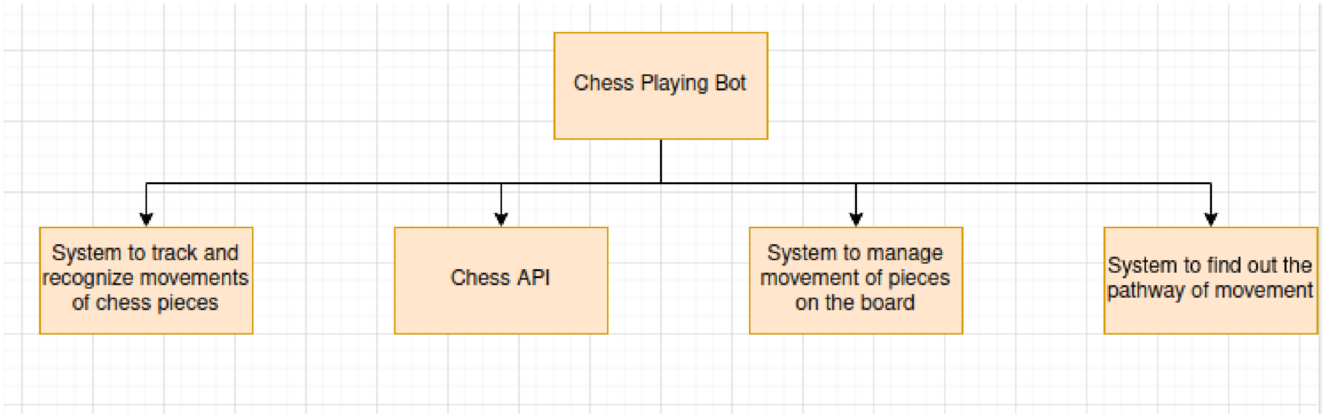

Figure 1: Functional Decomposition Diagram (FDD) of the proposed system

- System to detect movements of chess pieces placed by the user on the board.

  This system is responsible for tracking the position of chess pieces on the board and detecting the moves of the user.

- Chess API

  This system generates best possible moves based on the moves done by the opponent user.

- System for movement of pieces in the board.

  This system takes in the recommendations made by the chess engine to physically move the chess pieces on the chess board.

- System to find out the pathway of the movement.

  This system finds out the optimal pathway for the chess piece to move from one place to another.

2 Project Proposal

2.1 Introduction

Chess is a 2-player strategy game played on an 8x8 board with 16 pieces for each player. With the aim of trapping the opponent's king piece (Also called "Checkmate"), the players devise their own strategies and moves to get to that aim. Since the invention of chess as a form of a mind game, people have been trying to assert their dominance in terms of mental abilities through chess (Chess.com, 2021). And with the advent of chess theories, the game have become more and more difficult for a normal person to understand. From a business standpoint, the global chess market has been generally rising in most of the countries. The North American industry is expected to hit 40 million US dollars by 2022. These all statistics point to the fact that the equipment and resources related to chess are on demand (Statista.com, 2021).

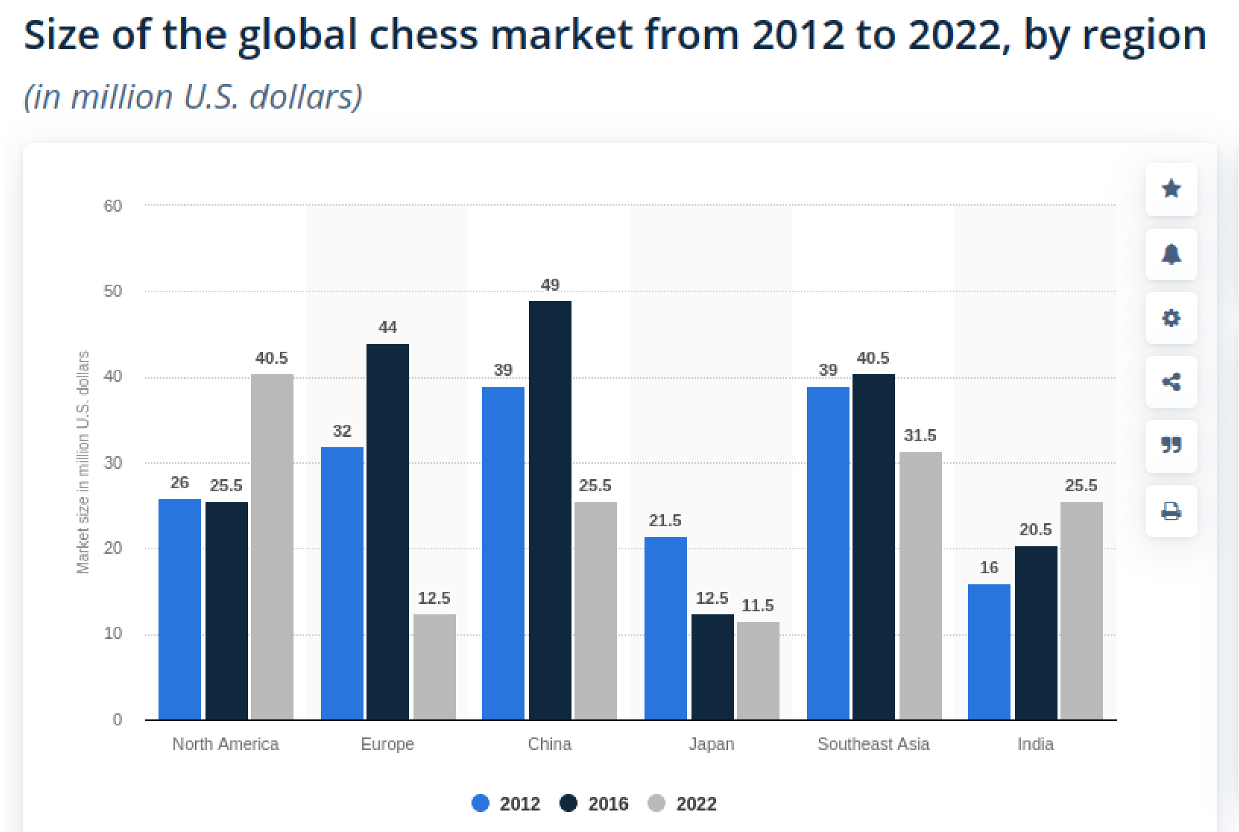

Figure 2: Global Chess Market Size Comparison from 2012 to 2022 (Statista.com, 2021)

With the increase in the whole industry, the availability of intelligent and specialized equipment for chess is very low. Similarly, the published resources are also vague to understand and expensive to manufacture. The main objective of this research is to perform an in-depth analysis on implementation of Artificial Intelligence (AI) to detect and recognize chess moves, analyze different methods to physically move the chess pieces with the help of a robotic system and determine the best way to move the pieces based on the analysis of different aspects of those methods and then construct the robot based on the research findings. At the end of this research, a fully functioning robot capable of playing chess against a user will be constructed with the help of the research materials and findings.

2.2 Problem Domain

I. No Proper method of training by playing physical match for newcomers

   If a newcomer wants to gain proper experience at chess, he/she have to physically play that game and since they are not well-trained, playing with another professional can be discouraging for them and it is difficult to find the person who agrees to play with an unexperienced player

II. Costly trainings and coaches

   In order to get good at chess, proper training and regular practice is required and the training programs and coaches are expensive to maintain.

2.3 Project as a Solution

I. Even the newest inexperienced players get to play with the standard opponent.

II. The cost of getting trained drastically decreases as it is a one-time investment. Once the bot has been bought, users can play whenever they like. Similarly, the bot would be constructed out of locally-sourced parts, repairing and even upgrading would not be difficult or expensive.

2.4 Initial Research into Sources of Information

2.4.1 Gambit: An Autonomous Chess-Playing Robotic System

This paper presents a 6-DoF chess playing robot system named Gambit which is capable of playing chess on a physical board against humans. It includes a low-cost sensor for perception, custom-made robot arm and learning algorithms for detection and recognition of objects on the board. The authors of the project mainly view this project as a way of exploring the perception and manipulation in a "noisy, less constrained real-world environment". Gambit does not require the chess pieces to be exactly modelled or instrumented as it monitors the state of the board and tracks the movement of the pieces that the opponent has made and communicates with the human opponent using spoken-language interface.

The perception system tracks the position of pieces on the board and the board is not fixed relative to the board and the board is calibrated continuously throughout the game. On the other hand, the system makes use of an open-source arm design as it was moderate in cost (about $18000 for parts) and their major aim was to enable smooth motion and interaction. It consisted of a 6-DoF arm with a gripper attached ad one end. As shown in figure. 1, the DoFs 1, 2 and 3 are used for positional control whereas 4, 5 and 6 are used for orientation control with the help of roll-pitch-roll spherical wrist.

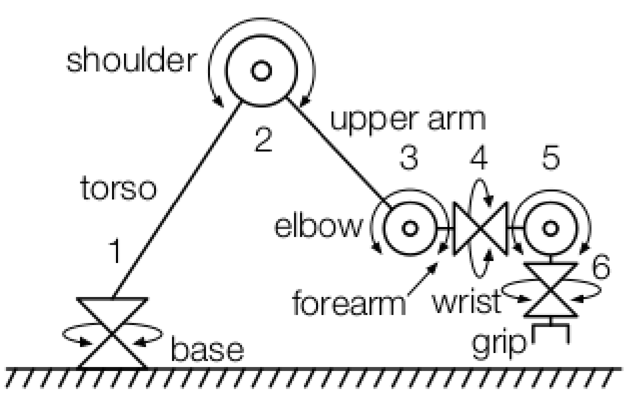

Figure 3: Schematic of the arm construction (Matuszek, et al., 2011)

For sensing purposes, the system consisted of a shoulder-mounted depth-sensing camera (similar to Xbox Kinect) along with a camera attached to the gripper. The depth-sensing camera provided color information along with depth of each pixel and worked within a range of 0.5m to 5m. Since the minimum working range was very high, it presented a problem that the camera had to be mounted high but the authors solved this issue by mounting the camera facing backwards of the torso of the arm so that the distance is increased.

The driver software for the system ran on a dedicated Intel Atom net-top PC with CAN and RS-485 PCI cards whereas controlling of the arm was done using a separate computer that handled ROS operations. And the system for detection and recognition of board consisted of four hierarchical classifiers mainly for detecting squares, pieces/backgrounds and two for recognition of types of chess pieces. Each piece was labelled out of pre-defined pieces {B, N, K, P, Q, and R}. The hierarchy of the vision algorithm is depicted in the figure below.

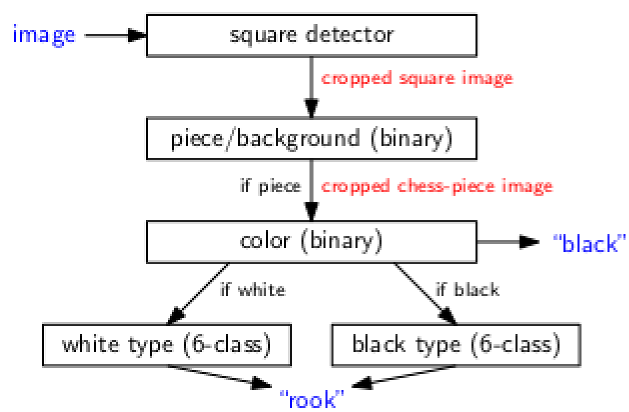

Figure 4: Hierarchy of the chess-piece recognition algorithms (Matuszek, et al., 2011)

(Matuszek, et al., 2011)

2.5 MarineBlue: A Low-cost Chess Robot.

This paper focuses on the details of a chess robot named MarineBlue. Emphasis has been given on the algorithms involving computer vision and robot manipulation in this paper. MarineBlue is an autonomous robot that can recognize moves placed by the user, generate moves for that scenario and move that particular piece with the help of an arm. (URTING & BERBERS, 2003).

2.6 Artefacts

2.6.1 System to detect and track movement of chess pieces on the board

This system will be responsible for recognition and detection of movements on the chess board. It will comprise of a sensor for vision purposes and a processing hardware for computational tasks. An image processing library such as opencv or tensorflow will be used to develop the tracking algorithm. Similarly, the sensor will be fixed to a point to avoid any errors due to calibration. Once the system detects movement, it will transfer the information to the system handling chess gameplay.

2.6.2 System to generate chess moves

This system generates valid and competitive chess moves based on the moves placed by the user. When the user have already placed a move and the system detects a move, it passes the move to this system and the validity of the move is checked at first. If the move is valid, then the move is passed through the chess API to generate best possible move against that particular move placed by the user. On the other hand, the chess API will either be an open-source API or will be developed from scratch for our own purposes.

2.6.3 System for movement of chess pieces on the board

Under this system, a hardware that can physically move the chess pieces in the board will be developed. The system will be able to move the pieces based on the moves provided by the chess API and the pathway will be calculated by a different algorithm. Different methods of movement such as mechanical arm or magnetic method have been considered for this project and will be decided by analyzing their strengths and weaknesses.

Mechanical Arm:- An arm with 4-DOF will be constructed using servos and LEGO-style blocks and the arm will be capable of picking the chess piece and move it around the chess board.

Magnetic Method: - A platform that lies below the chess board that resembles a typical CNC machine will be constructed and a magnet will be attached to the moving part of the machine and a magnet will be glued on the bottom part of each and every chess pieces.

2.6.4 System to find out pathway for moving chess pieces

This system will be responsible for generating pathway for the movement of chess pieces on the board for the bot. The algorithm to find out the optimal pathway for the chess piece to move will be based on the type of hardware to be developed based on the analysis. For instance, if a mechanical arm is to be constructed, an algorithm such as inverse kinematics would be developed in order to find out the angle required for each servos to get from one point to another whereas if the magnetic method is to be considered, an algorithm involving G-code or any other methods will be used. The algorithm for finding pathway will be dependent on the robotic method used.

2.7 Tools and technologies required

Hardware Requirements

- Mid-range PC (Laptop or Desktop) with at least Intel i5 or similar processor, 8GB RAM, 256GB SSD is required for SSH connection to Raspberry Pi.
- Hardwares such as Raspberry PI along with camera module, high torque servos.
- Chess Board

Technical Requirements

- Knowledge of Embedded systems such as Arduino and Raspberry PI.
- Clear understanding of image processing techniques and libraries.
- Experience of working with Linux environments (especially SSH/VNC and scripting).

Software Requirements

- Raspbian OS for Raspberry PI along with python as well as libraries such as tensorflow, opencv.
- Arduino IDE for programming arduino board.
- SSH software for accessing Raspberry PI over the network.

2.8 Development Cycle

Agile scrum is selected as a development methodology to be followed for this project. Since the requirements may change over time and agile approach allows the project to cope up with the changing requirements. Similarly, the major advantage of using scrum methodology is that it focuses on creation of deliverables at the end of each sprint and prioritizes incremental development method. So at the end of every sprint, we are left with a working version of the project. As the project will go through various iterations throughout the development cycle, the chance of it failing or having problems significantly goes down. The use of sprints and pre-set goals for each sprint means that the project is completed within the timeframe and prioritize the important aspects of the project for each sprint.

For our purpose, the sprint will last about 8-10 weeks and the whole project will be completed in 3 sprints. At the beginning of the project, the product backlog will be prepared and sprint backlog for each sprint will be prepared during the start of each sprint. Instead of daily standup, the meeting will be done on a weekly basis with the product owner/Supervisor and at the end of every sprint, sprint review and retrospective will be conducted.

2.8.1 Project Plan

Work Breakdown Structure

The whole project is divided into four major phases and at the end of all of those phases, the project will be declared complete.

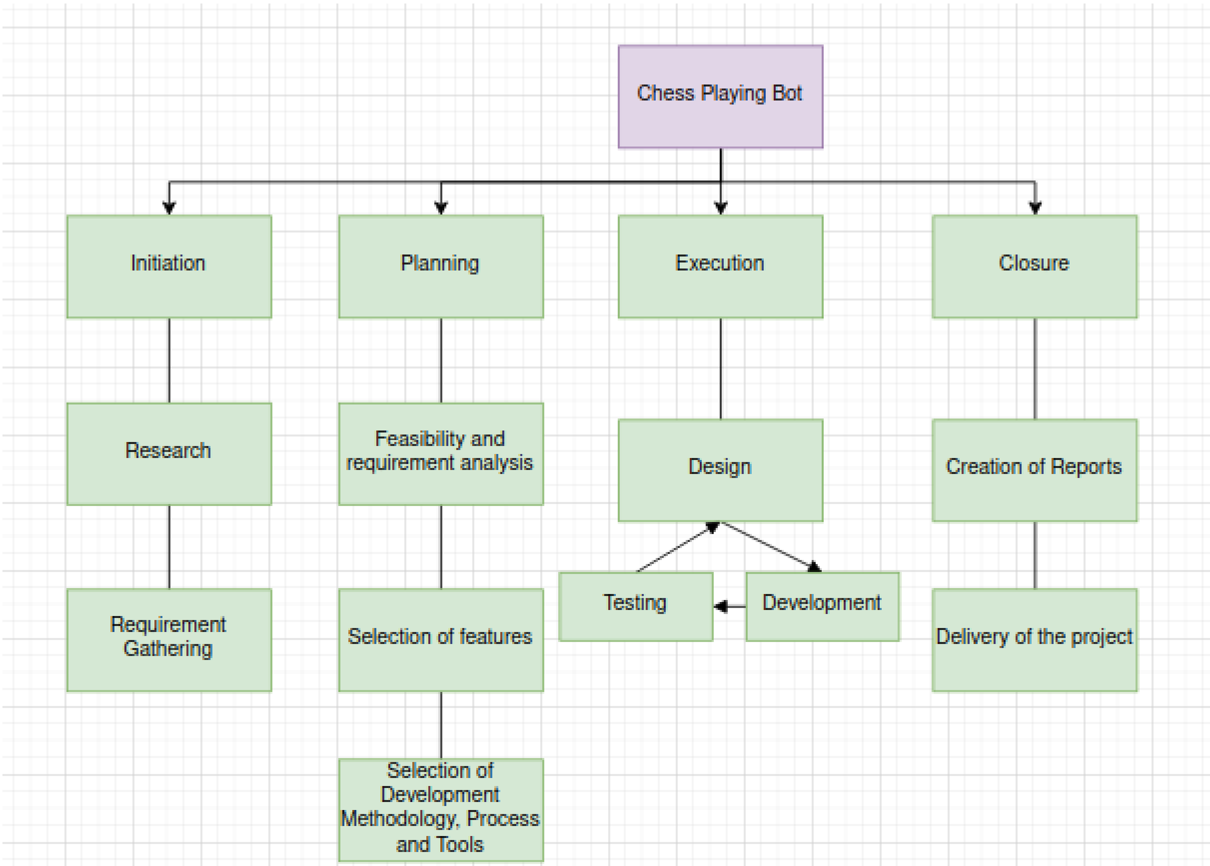

Figure 5: Work Breakdown Structure (WBS) of the proposed system

Similarly for testing, various tests such as integration testing, unit testing and functional testing will be conducted on a frequent basis.

2.8.2 Gantt chart

In the Gantt-chart below, the whole timeline starting from the initiation to the completion of the project is depicted and each sprint is set to approximately 8-10 weeks depending on the type of work being done.

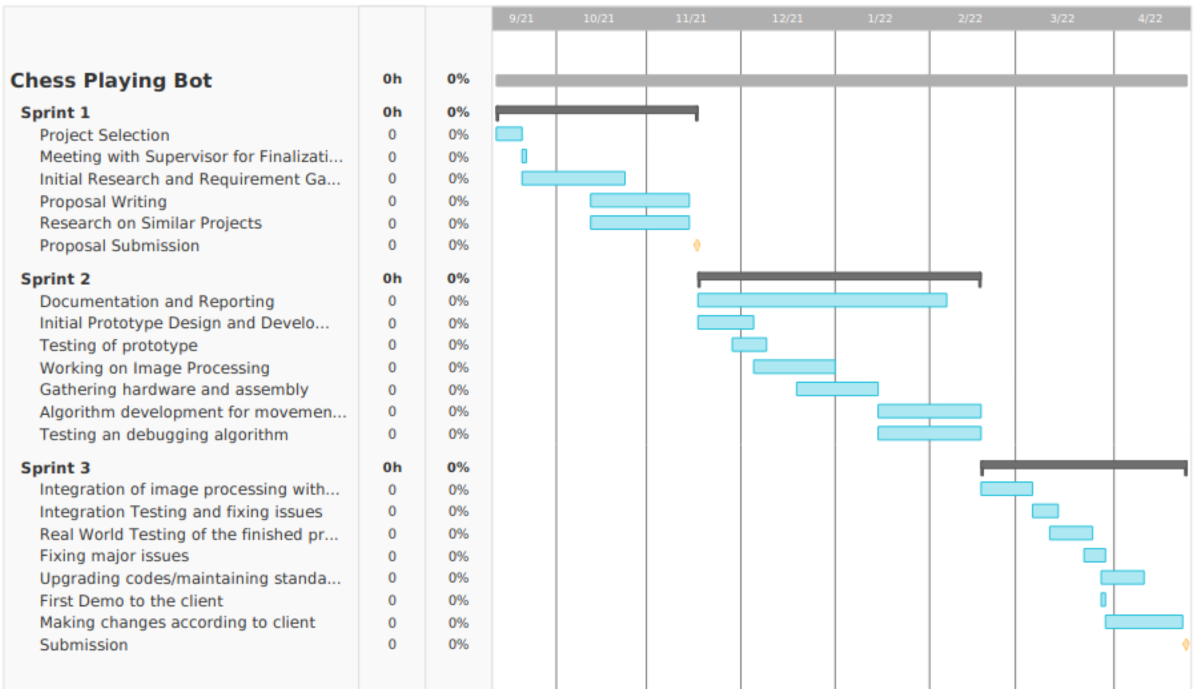

Figure 6: Gantt chart

3 References

Chess.com. (2021, 11 01). What Russia Taught the World About Chess. Retrieved from Chess.com: https://www.chess.com/article/view/russia-world-chess

Matuszek, C., Mayton, B., Aimi, R., Deisenroth, M. P., Bo, L., Chu, R., . . . Fox, D. (2011). Gambit: An Autonomous Chess-Playing Robotic System. 2011 IEEE International Conference on Robotics and Automation, 1-7.

Statista.com. (2021, 11 01). Global Chess Market. Retrieved from statista.com: https://www.statista.com/statistics/809953/global-chess-market-size/

URTING, D., & BERBERS, Y. (2003). MarineBlue: A Low-Cost Chess Robot. 1-7.