Department of Computer Science

The University of Hong Kong

COMP4801 Final Year Project

Interim Report

[MakerLab Project] Pick and Place Game App for 3D Printed Robotic Arm

Team member

Wong Ka Ngai, Benny (3035568881)

Wan Tsun Wai, Alan (3035569017)

Supervisors

Dr. T. W. Chim

Mr. David Lee

Date of Submission: 23/1/2022
Abstract
AI versus human players in games has been an increasingly popular topic, especially after the victory of AlphaGo. This project aims at developing a Connect Four AI for STEM education that utilizes a mobile application and a robotic arm to play with human opponents. To accomplish the objective, computer vision with OpenCV library will be used to recognize the board and a minimax algorithm will be implemented to calculate the next best move. Currently, the application developed can control the movement of the robotic arm via Bluetooth and recognize the color of discs inside the board. The next phase of this project is to improve accuracy in object recognition and build the unbeatable Connect Four AI. The components mentioned above will be integrated and fine-tuned for further testing soon.

Acknowledgment
Our team would like to express our sincere gratitude to Dr. T.W. Chim and Mr. David Lee for spending their time guiding us and giving us comments throughout the development of the project. Without their support, our team would struggle a lot in making progress. Additionally, our team would like to thank the Department of Computer Science and HKU CS MakerLab for offering this project's necessary resources and equipment.
Table of Contents

Abstract ............................................................................................................................... I
Acknowledgment ............................................................................................................... I
List of Figures.................................................................................................................... III
List of Tables ..................................................................................................................... III
Abbreviations .................................................................................................................. III
1 Introduction .................................................................................................................... 1
  1.1 Background ................................................................................................................ 1
  1.2 Motivation ................................................................................................................ 1
  1.3 Objective and Deliverable ......................................................................................... 3
  1.4 Report Outline ........................................................................................................ 3
2 Methodology .................................................................................................................. 4
  2.1 Software – Mobile Application ............................................................................... 4
    2.1.1 Integrated Development Environment ............................................................. 4
    2.1.2 Computer Vision ............................................................................................... 4
    2.1.3 AI Algorithm .................................................................................................... 5
  2.2 Hardware – Robotic Arm ........................................................................................ 6
  2.3 Hardware – Disc Stacker ......................................................................................... 7
3 Results and Discussion ............................................................................................... 8
  3.1 Current Status .......................................................................................................... 8
    3.1.1 App development ............................................................................................... 8
    3.1.2 UX/UI design ................................................................................................... 8
    3.1.3 Computer Vision .............................................................................................. 12
    3.1.4 Hardware ......................................................................................................... 13
  3.2 Limitations and Difficulties ..................................................................................... 14
4 Future Work ................................................................................................................ 15
  4.1 Auto home positioning ........................................................................................... 15
  4.2 3D modeling ........................................................................................................... 15
  4.3 Player VS AI mode ................................................................................................ 16
  4.4 Building mode ........................................................................................................ 16
5 Conclusion .................................................................................................................. 16
References ....................................................................................................................... 18
List of Figures

Figure 1: An example of a winning condition p.2
Figure 2: An example of a tie game p.2
Figure 3: An example of OpenCV board detection p.5
Figure 4: An example of Minimax tree p.6
Figure 5: An example of Alpha-Beta pruning p.6
Figure 6: Comparison between the original and the vacuum gripper p.7
Figure 7: The draft of the proposed disc stacker p.7
Figure 8: The color palette p.9
Figure 9: UX and UI of home activity p.9
Figure 10: UX and UI of Bluetooth activity p.10
Figure 11: UX and UI of controller activity p.10
Figure 12: Workflow of Player VS AI mode p.11
Figure 13: UX and unfinished UI of building activity p.11
Figure 14: Preparing custom dataset for training p.12
Figure 15: Detection result by the application p.12
Figure 16: Confusion matrix of training result p.13
Figure 17: An example of print failure with no support p.14

List of Tables

Table 1: Remaining schedule p.17

Abbreviations

3D Three-Dimensional
AI Artificial Intelligence
GUI Graphical User Interface
IDE Integrated Development Environment
IOS iPhone OS
STEM Science, Technology, Engineering, and Mathematics
UI User Interface
UX User Experience
1 Introduction

The following chapter gives an introduction to this progress report. First, a brief background on AI and the robotic arm is given. After that, the motivation for doing this project is discussed, followed by the objective and deliverable. Then, an outline of the report is mentioned.

1.1 Background

Early in 1997, a chess computer Deep Blue defeated the world champion Garry Kasparov, marking as a symbolic testament to the rise of AI [1]. Since then, AI versus human players in games has become widely researched. In 2017, an AI player AlphaGo defeated the best human player Ke Jie in the game GO [2], again demonstrating AI's potential in playing games. In order to translate the AI decision into action, very often a human agent in between is needed. A robotic arm can be utilized to automate this process. The use of robotic arms can be traced back to 1954 when George Devol invented the first industrial arm [3]. Unlike human beings, robotic arms were able to perform with high precision and work for long hours. However, they could only follow a predefined moving path and operate without real-time calculations.

With AI algorithms and computer vision, robotic arms in the current decade can learn to make decisions without following the predefined rules by humans. As such, robotic arms can now be put into more complicated working environments like medical use. It is possible to utilize robotic arms in decision-making games to play against human players with modern powerful chips.

1.2 Motivation

Connect Four is a two-player connection board game published in 1974 [4]. A standard Connect four game contains a vertical board with seven-column and six-row, 21 yellow discs and 21 red discs for two players. Players alternate turns to drop one disc from the top into any of the seven slots. The player wins if four of his discs connect to a vertical, horizontal or diagonal line (see Figure 1). The drawing condition only occurs when the board is complete and no player can fulfill the winning condition mentioned above (see Figure 2).
Connect Four is simple to play but hard to master. It is no easy task to make an AI Connect Four robot as the game requires the robot to be precise in picking and placing and have strategic thinking at the same time. The motivations for doing this project are illustrated below.

Connect Four is unique in itself that unlike most board games, its board is vertically placed. In order to successfully place a disc inside the board, precision control of the robotic arm is crucial, attributed to the minimal gap of the column slot. Also, it is a common practice for horizontal board games to keep the mobile phone at a fixed distance to capture the board. However, as Connect Four has a vertical board, the distance between the phone and the board may vary to calibrate correctly. It adds extra difficulties to our project.

In addition, Connect Four is a solved strategy game, where the first player has an unbeatable winning strategy. This game was independently solved by James Dow Allen and Victor Allis in 1988 [5]. In theory, it is possible to build an unbeatable AI for this project. Therefore, it is believed that the AI player will be a worthy opponent for human players. Unfortunately, in comparison with unsolved games like chess, solved games like Connect Four are less popular to study. As a result, there is no existing OpenCV or PyTorch library to use. Development in
OpenCV or PyTorch library specializing in Connect Four is needed for this project, which should fill up the gap of missing libraries. It is also hoped that this project can bring some extra insights and discussions to the solved game community.

Last but not least, after finishing this project, the robot can be displayed in public spaces to play with people for entertaining and educational purposes. Unlike other complicated games, connect four is easy to pick up and everyone can enjoy it even if they are just children. Furthermore, students’ interest in computer science and AI can be aroused by playing with the unbeatable AI. It is believed that this project can be helpful for STEM education.

1.3 Objective and Deliverable
This project proposes to develop a mobile application for a 3D printed pick-and-place robotic arm that can play Connect Four with human opponents. The project can be broken into two parts: a mobile application for the AI player and a fine-tuned robotic arm. The mobile application should be able to capture and analyze the board to calculate the next move. Then, it sends the command and position data to the robotic arm. The application should also have an easy-to-use graphical user interface that minimizes the user’s learning time. Furthermore, the robotic arm should be able to receive the command from the application via Bluetooth and then pick and place the discs quickly and accurately.

With the finished mobile application and robotic arm, this project should be able to deliver a Connect Four AI that plays with human opponents using the robotic arm and mobile application and a ready-to-use Connect Four OpenCV library. In addition, the final product should demonstrate the strength of AI algorithms in decision making and object recognition so that the product can be utilized for STEM education.

1.4 Report Outline
The remaining paper is organized as follows. Chapter 2 describes the methodology involved in this project so that the robotic arm can be able to pick and place discs automatically. Chapter 3 then states the project's current status, limitations, and difficulties. Chapter 4 follows by introducing the planned future undertakings. A short conclusion and the remaining schedule are provided in Chapter 5.
2 Methodology

To achieve the aforementioned objectives, the proposed product of this project is divided into two major components, namely the software component and the hardware component. In particular, the objective of the software component is to construct a mobile application that captures the board and generates the next move. Various tools and methods are to be utilized, including Android Studio, OpenCV library and Alpha-Beta pruning. They are introduced in Chapter 2.1.

The objective of the hardware component, which is the robotic arm, is to execute the command received precisely. However, modifications to the robotic arm are necessary in order to adapt to this project. Suggested enhancements are introduced in Chapter 2.2. Chapter 2.3 proposes a disc stacker to facilitate picking and placing discs.

2.1 Software – Mobile Application

Before building a mobile application, software must be chosen for development. Chapter 2.1.1 explains the rationale of utilizing Android Studio. Chapter 2.1.2 details the methodology of image recognition using the OpenCV library. Chapter 2.1.3 describes how Alpha-Beta pruning determines the next move to play.

2.1.1 Integrated Development Environment

An integrated development environment (IDE) is a software application that provides comprehensive facilities, allowing computer programmers to develop software for a particular platform [6]. Android Studio is the IDE chosen to develop the mobile application in this project. Android Studio has provided numerous support to its developers since its first release in 2014 [7]. It offers a plethora of packages for developers, of which the Bluetooth package applies to this project. Compared to other mobile operating systems, more than 70% of mobile phone users use Android [8]. It is believed that development in Android matches most users.

2.1.2 Computer Vision

Computer vision, a field of AI, derives meaningful information from visual inputs like images. It enables computers to observe and understand. The mobile application will adapt computer vision to recognize the board. To accomplish this, the OpenCV library will be used. OpenCV is a powerful open-source computer vision library that provides optimized algorithms to accomplish various computer vision tasks, including but not limited to object identification, face recognition
and model extraction [9]. The mobile application will utilize OpenCV, Pytorch to detect the board, analyze the state and transform it into machine-readable data (see Figure 3). In particular, YOLOv5 will be used for the model to detect the discs. YOLOv5 is an open-source object detection model widely used on projects like multi-objects detection, mask detection and face detection. It used the Pytorch framework that can easily be converted to ONNX format for android or PTL format for iOS. The model also supports calculation speed up to 140 FPS and the size is very small. Therefore, YOLOv5 was selected for this project's primary computer vision model.

![Figure 3: An example of OpenCV checkerboard detection](image)

2.1.3 AI Algorithm

With the current game state obtained, the application calculates the next move and sends the corresponding command to the robotic arm. In this project, a minimax algorithm with Alpha-Beta pruning will be used, as minimax is widely used in two player turn-based games. In minimax, the two players are called maximizer and minimizer, where the maximizer tries to attain the highest possible score and the minimizer does the converse [10]. The algorithm is contingent upon the minimax tree, which is similar to the decision tree. Each layer of the tree consists of either maximized nodes or minimized nodes. A node represents a board state which has an associated value. The algorithm traverses the tree and chooses the path that obtains the optimal result (see Figure 4). As Connect Four is a zero-sum game, where a player’s loss equals another player’s gain, minimax can be further simplified to negamax. Only a single version of the recursive function is needed as the score of a position to a player is the negation of another player's score.
However, it is still time-consuming to traverse when the tree is substantial. Alpha-Beta pruning optimizes the algorithm by cutting off branches that need not be searched as a better move is available (see Figure 5) [11]. It significantly reduces computational time.

**2.2 Hardware – Robotic Arm**

The robotic arm is provided by HKU CS MakerLab. However, enhancements to the robotic arm are necessary in order to adapt to play Connect Four. For instance, end stop switches and extra modules will be installed to accomplish auto home positioning. Furthermore, to efficiently pick and place discs, modifications will be made to the gripper, like replacing sponges located at the tips of the claw with other materials to increase the accuracy of placing discs. It is also posited to replace the claw gripper with a vacuum gripper, as sucking is believed to be a more stable way to pick up discs than gripping (see Figure 6). If a vacuum gripper is to be used, extra modules will be designed and printed to install the gripper. Further testing and experiments on comparing the two grippers will be conducted in the future.
Figure 6: Comparison between the original claw gripper (left) and the vacuum gripper (right)

2.3 Hardware – Disc Stacker

The claw gripper cannot pick up laying down discs and place them inside the vertical board. Thus, a disc stacker will be designed and printed so the robotic arm can pick up discs vertically at the same position. 3D modeling software like AutoDesk will be used to design the stacker, and it will be printed using 3D printers. The posited disc stacker stores discs vertically. When a disc is picked, a new disc rolls out to refill (see Figure 7). Therefore, the claw gripper can grip discs continuously at the exit of the disc stacker until there is no disc left.

Figure 7: The draft of the proposed disc stacker
3 Results and Discussion

This chapter demonstrates the results achieved in the interim of this project. Chapter 3.1 states the current status, including the functions done and decisions made. Chapter 3.2 discusses the limitations and difficulties identified.

3.1 Current Progress

In the interim of this project, the accomplished tasks can be subdivided into four sections. First, the current status in app development will be discussed, followed by the UX/UI design. Next, the computer vision model developed will be introduced. Finally, hardware-related progress will be mentioned.

3.1.1 App development

Meetings have been convened regarding the choice of the development platform. React Native was the posited platform at the start of the project as it offers support to both Android and iOS devices. However, Android Studio was chosen because of its significant number of ready-to-use packages, including the Bluetooth module that can be utilized by the application. Android Studio has a more straightforward structure than React Native, which likely indicates that it is easier to learn. This decision is justifiable because the cross-platforming advantage given by React Native is insignificant in this project.

Currently, the application is under development. The application can utilize Bluetooth to search for nearby devices. Once connected with the robotic arm, users can manually control the movement of the robotic arm by using the arm controller provided by the app. An overview of the current application will be illustrated below in Section 3.1.2.

3.1.2 UX/UI design

The application is developed and the UI is designed based on the UX mockup drawn. The app's color theme followed suggestions from Material Design [12]. Teal and Pink are chosen as the primary and secondary colors as they are the colors of the Connect Four discs (see Figure 8).
It is a prerequisite for the app to connect with the robotic arm via Bluetooth to use its functions. This prerequisite is accomplished by disabling all buttons except for the Bluetooth button at the home activity if no connection is detected (see Figure 9).

When the Bluetooth button is clicked, the Bluetooth activity is launched. Users press the scan button to scan nearby devices. When the robotic arm is discovered, users click the item to establish the connection (see Figure 10). The app returns to the home page, and toast messages “Connecting” are shown. The toast message “Connected” is shown once the connection is constructed and all buttons are enabled. The Bluetooth button icon changes to disable Bluetooth so that users can click it again to disconnect.
When the arm controller button is clicked, controller activity is launched. Users control the arm by using the buttons provided. Buttons at the bottom allow users to control the robotic arm manually, while buttons in the middle move the arm to predefined positions (see Figure 11).

Users play Connect Four with the AI in Player VS AI mode. The posited workflow is organized as follows. The player first decides the one going first, then uses the mobile phone to capture the board. Once the board is recognized, the game can be started. During the game, scores will be awarded or deducted for every move made by the player. The player can use the hint button to
get tips for the optimal and rewarding move. The AI constantly tells the number of moves remaining to defeat the player to make the game more intriguing. After the game is finished, the app shows the player's score (see Figure 12). It is believed that by utilizing the scoring system, players are encouraged to attain a higher score in the next game.

Figure 12: Workflow of Player VS AI mode

To better utilize the mobile phone and enrich the interactions of the app, the user and the hardware, a building mode is proposed. In building mode, the app randomly gives a pattern to the player (see Figure 13). The player’s task is to build the corresponding pattern within the time limit by manually controlling the robotic arm to pick and place discs inside the board.

Figure 13: UX and unfinished UI of building activity
3.1.3 Computer Vision

Computer vision contributes a vital role in the application. Since there is no existing library for Connect Four, a dataset must be prepared. First, photos of the board and the discs were captured and labeled manually (see Figure 14). Then, the YOLOv5s model, one of the YOLOv5 models, was used for supervised learning to generate the custom model in the Pytorch framework. After translation, the model was applied in the app for board detection (see Figure 15).

![Figure 14: Preparing custom dataset for training](image)

![Figure 15: Detection result by the application](image)

The performance of the trained model was satisfactory. The true positive rate for both pink and green reached 0.93 and 0.73 (see Figure 16). However, there are some problems needed to be addressed. For example, the detection algorithm was too sensitive to recognize irrelevant objects.
Also, multiple objects were detected in the same area. These problems will be further discussed in the Future Work section.

![Confusion matrix of training result](image)

*Figure 16: Confusion matrix of training result*

### 3.1.4 Hardware

Tests have been conducted to identify and repair all malfunctioned modules of the robotic arm. The robotic arm is also enhanced by installing end stop switches and the related modules. As a result, the current robotic arm can function properly and is ready to implement auto-home positioning. Moreover, training on utilizing 3D printers is finished. Experiments on modifying printing parameters are carried out to familiarize with print settings. For example, Figure 17 demonstrates a print with no support structure generated, leading to the displacement of materials at certain levels. The knowledge acquired in 3D printing should address future printing needs.
3.2 Limitations and Difficulties

In the interim of this project, it is foreseeable that time constraints will be the major limitation. As the project needs to be finished on time and Android Studio is the chosen platform, one limitation is that the final mobile application delivered will not provide iOS support. This limitation is unavoidable. However, it is insignificant as well. As cross-platform support is not the primary objective of this project, this limitation is considered a minor drawback.

Another limitation is found in picking up discs. Ideally, the robotic arm can act like a human, picking a disc from a group of laying down discs under the vertical board and placing it inside the board. Object recognition on the laying down discs is compulsory to accomplish this. However, as the mobile phone camera is used to capture and analyze the vertical board, it is not feasible to simultaneously capture the horizontal laying down discs. Also, another AI model will be needed for recognizing and picking the laying down discs. Due to time constraints, it is believed that object recognition on picking discs cannot be finished on time. To mitigate this, one possible solution is to use disc stackers such that the arm can pick up discs at a fixed position without any recognition needed.

When developing the computer vision component for obtaining the board state, difficulties were encountered. First, the model was too sensitive towards objects with similar colors as pink and green. Undesirable objects were detected in the frame. A posited solution to this problem is first to capture and isolate the board, eliminating objects outside the board.

Another difficulty was that the model might have multiple detections on the same area, detection boxes might overlap and interfere. An algorithm called Non-max Suppression will be used to
tackle this problem. The algorithm will loop through all possible boxes and compare their probabilities and areas. Once multiple boxes in the same area are found, the probabilities will be compared and rejected those with lower probabilities. So multiple boxes will not appear in the same area.

4 Future Work
This chapter describes the works to be done in the remaining months. They can be briefly summarized into four main categories: auto home positioning, 3D modeling, player VS AI mode and building mode.

4.1 Auto home positioning
In order to implement the auto home function, Arduino codes will be modified. Currently, the Arduino board consists of the control code only. Users are allowed to control the robotic arm through wire-connection or Bluetooth connection. The home position needs to be manually configured whenever the robotic arm starts up. After applying the end-stop switches, it is possible to detect the end position of each axis, making the auto home function possible. Arduino codes will be modified to implement this function so the robotic arm will perform a sequence of movements after starting up. When it detects the end positions, the robotic arm will calculate and return to the home position by itself.

4.2 3D modeling
To allow the robotic arm to grip a disc, a disc stacker will be built for storing the discs. Since it is a tailor-made disc stacker, the 3D model will be designed and 3D printers will be used to print it out. After printing, the stacker will roll the disc out automatically and the robotic arm will grip from it.

Moreover, a funnel-like module may also be designed and printed if the accuracy of placing discs is not high enough. The module will be installed on top of the board, working as an assistant to help place discs inside the column slots. This measure will act as the last resort for compensating the accuracy issue.
4.3 Player VS AI mode
The computer vision, minimax, and GUI components must be done to complete the Player VS AI mode.

For the computer vision, framing will be used to capture the board first and trim off any irrelevant background to reduce the possibility of background object detection. Furthermore, a non-max Suppression algorithm will also be used to improve the overlapping box problem as only the box with the highest probability will be kept. Nonetheless, to improve the accuracy of the existing object detection algorithm, a hybrid method will be used. First, data will be gathered from the existing method and a new contour detection method. Then board state will be processed to give a more accurate result of the disc location. After that, the board state will be transferred to the minimax algorithm for optimal move analysis.

The minimax algorithm will be constructed using Pascal Pons’s Connect Four solver as a reference [13]. Under the guidance of the solver, it is believed that the final AI algorithm built will be unbeatable, able to predict the remaining number of moves to win and evaluate moves done by the player. During the game, the GUI will display the current board state, the number of moves remaining for the AI to win, and the player's current score. If the hint button is pressed, the AI algorithm will evaluate the game state and the GUI will display the scores of each possible move.

4.4 Building mode
As the controller section is completed, the remaining task is to implement a randomizer that arbitrarily picks a pattern from a list of board patterns and a countdown timer. Once they are finished, the player can receive a random pattern and build it using the controller within the time limit.

5 Conclusion
This project aims to construct a Connect Four AI player that automatically picks and places discs using a mobile application and a robotic arm. Android Studio was chosen to be the development platform and the application is under development. Currently, the application provides basic functionalities of controlling the robotic arm and recognizing the discs. The next phase of this project is to finish improving computer vision accuracy and build the minimax algorithm. After that, parts will be integrated and fine-tuned for testing.
In particular, a hybrid object detection method will be studied and implemented. Contour detection will be used and combined with the existing detection method. Minimax algorithm will also be studied to build an unbeatable AI. After combining all the parts into the application, a scoring system will be set up to rate the player's performance.

Despite the difficulties we have encountered, the project's progress is still on track. The remaining time of the project will be planned as follows.

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>January 2022</td>
<td>Finish Interim Report</td>
</tr>
<tr>
<td></td>
<td>Enhance Vision</td>
</tr>
<tr>
<td></td>
<td>Learn Arduino programming</td>
</tr>
<tr>
<td></td>
<td>Implement Auto-home positioning</td>
</tr>
<tr>
<td>February 2022</td>
<td>Fine-tuning</td>
</tr>
<tr>
<td></td>
<td>Implement Minimax Algorithm</td>
</tr>
<tr>
<td></td>
<td>Learn 3D modeling</td>
</tr>
<tr>
<td></td>
<td>Finish Building Mode</td>
</tr>
<tr>
<td>March 2022</td>
<td>Prepare final report</td>
</tr>
<tr>
<td></td>
<td>Integrate all functions</td>
</tr>
<tr>
<td></td>
<td>Improve computer vision AI</td>
</tr>
<tr>
<td></td>
<td>Finish the scoring and hints system</td>
</tr>
<tr>
<td>April 2022</td>
<td>Prepare Report</td>
</tr>
<tr>
<td></td>
<td>Finish Player VS AI mode</td>
</tr>
<tr>
<td></td>
<td>Deliver final report</td>
</tr>
<tr>
<td></td>
<td>Deliver final product</td>
</tr>
</tbody>
</table>

*Table 1: Remaining schedule*
References


