A Checkers Playing Robot

Darren Lewis[1]**, Donald Bailey**[2]

[1]Institute of Technology and Engineering
[2]Institute of Information Sciences and Technology
Massey University
Palmerston North, New Zealand

Email: Darren_D_Lewis@hotmail.com, D.G.Bailey@massey.ac.nz

Abstract: This paper describes the design and construction of a board game playing robot that can be used to interactively play a human opponent in three-dimensional space. The system consists of an input system, a game engine, and an output system. The input system is in the form of a vision system to identify all the necessary information required to play the chosen game of checkers. The game engine determines the move that the robot should make. The output system then implements the move made by the game engine on the real-world game board.

Keywords robotics, checkers, board games, vision systems.

1. INTRODUCTION

The main aim of this project was to create a complete robotic system that incorporated a number of different technologies. A board game was chosen as the platform for the robot, because having the robot interactively playing against a human opponent in the real world requires several inter-related systems to work together.

The robotic system to play a board game in the real world requires an input system, game engine and an output system. The function of the input system is for the computer to determine the state of the game. The game engine is required to decide on a move for the robot to make for the output system to implement it on the game board in the real world. The relationship between these is shown in figure 1.

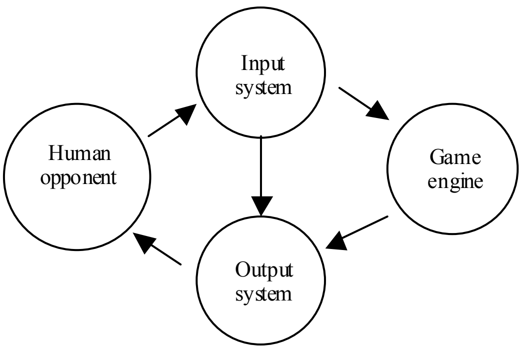

Figure 1. The System Diagram

1.1 The Choice of Board Game

The choice of board game determines the complexity of both the manipulator in the output system, and also the programming of the game engine.

The game of Anglo-American checkers was chosen due to its simplicity. The pieces used in game play are in the form of flat disks that all have the same dimensions. This simplifies the manipulator in the output system, as the manipulator does not have to adapt to large variations in the size of the pieces. The game is also well known due to its simplicity. This allows a simple game engine algorithm to be programmed. Although the system is greatly simplified by the choice of checkers, the main principles for a robotic board game playing system are still demonstrated.

1.2 The Game of Checkers

Anglo-American checkers is a two-player game that is played on an 8 by 8 checkered board. The colour of the squares alternate from black to red in each direction with a black square at each player's lower left corner. The pieces are also red and black, and can only move on the black squares.

The goal of the game is to either capture all of the opposing pieces by jumping them, or to stop them from being able to make a move, with jumping being mandatory. If neither player can do this, the game is a draw.

The players move their own pieces in an alternating order starting with the red team, with the pieces consisting of pawns and kings. A pawn can become a king when it reaches opposite end of the game board. A pawn can move one square diagonally forward onto an empty square unless jumping, while a king can move and jump in any direction.

Jumping can be performed if a square diagonally in front of a piece is occupied by an enemy piece with the square beyond empty. However, the direction rules still apply. A piece can continue to jump enemy pieces until there are no jumps available.

2. THE INPUT SYSTEM

The input system is required to:
- Identify the position of each piece on the game board
- Identify the team and rank of each piece
- Detect when the human opponent has made a move

One solution for identifying the position of each piece is to have an array of proximity sensors mounted under the game board. With this approach however, the identification of the team and rank difficult.

Therefore, a vision system is chosen as it eliminates these problems due to the fact that all the required information can be obtained from an image.

2.1 The Camera

While only five or six pixels across the width of each square on the game board are required to detect each piece, the resolution needs to be higher to identify the rank (whether the piece is a pawn or a king). Having 30 pixels across the width of each square should be sufficient for this. This means that an image of 240 by 240 pixels can represent all the required information of the game board. However, captured pieces must be placed off the board. This means that a greater number of pixels is required in one direction. Therefore the minimum image size is 320 by 240 pixels. The modest resolution requirements allow an inexpensive web-camera to be used to acquire the images.

The web-camera chosen has the ability to capture 640 by 480 pixel images, which meets the requirements. However, the 640 by 480 pixel setting provides a lower quality image than the 352 by 288 pixel setting. This suggests that the sensor resolution is 352 by 288 pixels images are interpolated up to the 640 by 480 pixel image size. As 352 by 288 exceeds the minimum resolution, this was used.

It is desired to have the line of sight of the camera perpendicular to the game board surface to avoid perspective distortion. This means that every square on the image of the game board has approximately the same number of pixels allocated to it as every other square. This is achieved by mounting the camera directly above the centre of the game board (as shown in figure 2) at a height that maximises the size of the game board in the images. The camera is also aligned with the game board to simplify the calibration.

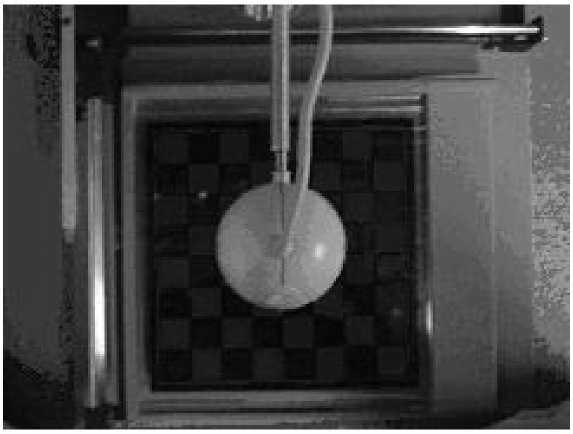

Figure 2. The Camera Location

2.2 The Lighting

The identification of the pieces requires images with constant lighting conditions. The vision system then does not have to work under a variety of different lighting conditions. The ambient light is insufficient to illuminate the game board as it can vary significantly in intensity with time. The solution is to provide constant lighting that is significantly brighter than the ambient light. This means that any change in the ambient light intensity is insignificant compared to the intensity of the constant light.

As the game board being used has a glossy surface, the light source is important. The light source cannot be mounted directly above the board beside the camera as the specular reflection from the game board would effectively blind the vision system. Therefore, the light source is mounted a small distance above the back edge of the game board so the light reflects off the game board at a low angle minimising the specular reflection, as shown in figure 3.

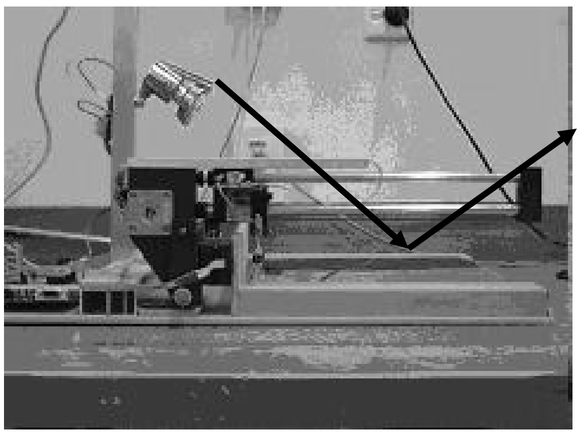

Figure 3. Placement of the Light Source

To fulfill the requirements of illuminating the game board, a 50-Watt AC halogen light source with an angle of illumination of approximately 45 degrees was chosen. This provides sufficient illumination of the game board and the 50 Hertz AC flicker is not detected by the camera.

2.3 *The Playing Pieces*

The playing pieces are made to be 22mm in diameter so that they fit within the 25mm squares on the game board. They are painted in the traditional black and red colours.

A king can be represented by placing two pieces on top of each other or by turning a piece over to reveal a ‘king’ symbol. The camera is mounted directly above the game board, so the captured images do not include information on the height of the pieces. Therefore, two pieces on top of each other cannot easily be detected, but pieces with symbols can. For this reason the method of representing a king by turning over a pawn to reveal a ‘king’ symbol is chosen. Another advantage of this method is that the manipulator does not have to move two pieces stacked on top of each other.

The identification symbol to represent a king has to be meaningful to both the vision system and the human opponent. The obvious identification symbol for the ‘king’ is a crown. This can be detected by the vision system, and a human would naturally assume that it is a king in the game of checkers.

As the game is played on the black squares of the game board, black pawns are difficult to detect due to the very low contrast between the squares and pawns. Therefore, the identification of a black pawn is simplified by the use of a small dot in the centre of the piece. The shape of the dot is significantly different from that of the crown to simplify the classification of each piece.

The black pieces are painted with a matt black paint to minimise specular reflections. This makes them appear black to the vision system. On the other hand the red pieces are painted with gloss red paint to maximise specular reflections, and therefore appear lighter in intensity to the vision system.

The identification symbols need contrast strongly with the black and red pieces. For these reasons white symbols are used on the black pieces, and black symbols are used on the red pieces as shown in figure 4.

2.4 *Rank Identification*

The pieces need to be classified as pawns, kings or enemy pieces. This information is used to create a virtual board of the current state of the game, and is determined by the use of basic image-processing techniques.

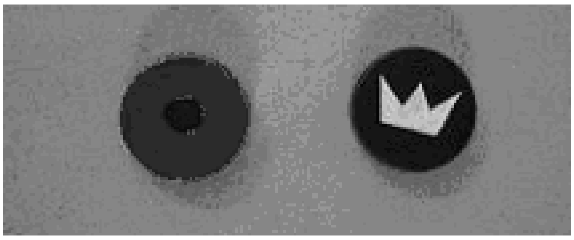

Figure 4. A Red Pawn and Black King

As the background is constant and does not change throughout game play, an image of the background may be subtracted from the image of the playing area to remove any extraneous detail. The background image is captured as part of the calibration process before the game is played. Background subtraction allows the changes in the image to be detected enabling the pieces to be segmented. The image can then be turned into a binary image by a simple threshold. From here the identification symbols in the image can be segmented.

Each of the regions can then be classified as a pawn, king or enemy piece. The classification is based on the shape of the identification symbols. This is made possible, as there is a significant difference in shape between the identifying symbols. The ranks of the black pieces are determined by classifying round white areas between 20 and 100 pixels as being a pawn, and non-round areas between 70 and 180 pixels as being a king.

The red pieces are detected as large round objects with a hole in the middle. Currently, they are only classified as ‘enemy’ due to the nature of the game engine described in section 3. However, they could be classified as enemy pawns or enemy kings by the shape of the hole inside the segmented region.

2.5 *Position Identification*

The position of each piece is required to create a virtual board of the current game state. As the squares on the game board are all the same size, the centres of each square can be calculated from the size and location of the board. When the system is started, the user is presented with an image of the game board, and must select the top left and bottom right corners. The centres of each square can then be found by dividing the length and width dimensions by eight.

The centre of mass of each segmented region in the binary image is used to determine the location of each piece. Each segmented region is assigned to the square whose centre is closest. The location, rank, and team of each piece are combined to create a virtual board on the computer.

2.6 *Move Detection*

The robotic system is required to determine when the human opponent has finished their move. In making a move, the human opponent has to place their hand under the camera to move a piece. This usually obscures some of the pieces detected by the vision system. Therefore, valid images must be identified. Checking the number of enemy pieces on the game board does this. Any move (either a simple move or a capture) only moves one of the player's pieces. A move can then be detected as being made when one enemy piece has been moved, and the number of enemy pieces on the game board has not changed.

3. THE GAME ENGINE

The purpose of the game engine is to select a move that the robot can make that is within the rules of the game. Therefore, the game engine in this case has to:
- Perform a legal 'jump' if available, otherwise
- Perform a legal 'standard' move if available

In the prototype, the game engine is kept simple by randomly choosing a legal move to make. This demonstrates the principle of a board game playing robot and still allows the implementation of a more complex game engine at a later date.

Therefore, the structure of the game engine algorithm checks to see what pieces can jump, and randomly select one of these moves. If no jumps are available, all possible standard moves are determined, and one of these is selected at random.

4. THE OUTPUT SYSTEM

The output system is required to implement the move determined by the game engine on the game board. This requires a 22mm circular piece to be positioned inside a 25mm square. Therefore, the output system is required to position a gripper to an accuracy of at least 3mm.

4.1 The Mechanical Arm

The game board is 240mm by 240mm. In order to move a piece to any position on the game board as well as being able to place captured pieces off the board, the robotic arm requires 370mm linear movement in the 'X' direction. However, only 240mm is required in the 'Y' direction. It is also necessary to be able to lift the pieces over each other when making a move. The pieces are 6mm in height, but the game board is slightly convex, being 3mm higher in the centre of the board than at the edges. This means the 'Z' direction requires a minimum of 9mm travel.

Possible designs of the mechanical arm include an anthropomorphic arm, and a 'XYZ' arm. The anthropomorphic design requires complex trajectory planning to position the gripper in the desired location. Therefore, the 'XYZ' design is used, as it only needs linear trajectory control. Due to the fact that the arm only requires moving a relatively small distance in the 'Y' direction, a cantilever design is possible as shown in figure 5.

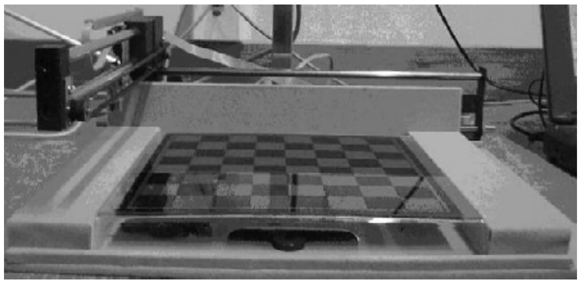

Figure 5. The Mechanical Design of the Robot

Stepper motors are used, as they allow for precise and repeatable open-loop control. The translation of the rotational motion of the stepper motors to linear motion of the arm, requires a system that is both simple, and does not allow the pinion of the motors to slip. Therefore, a toothed belt is used to drive the moving component in both the 'X' and 'Y' directions. The stepper motors used allow approximately 1mm linear motion per step when coupled through the pinions to the belts.

As a cantilever design is used, the lifting of the pieces in the 'Z' direction requires the 'Z' direction stepper motor to be light in weight to minimize the torque applied on the cantilever arm. Therefore, a worm drive is used. This can provide a high torque allowing a smaller motor to be used.

The system must allow linear motion in the axis of movement while being rigidly supported in the other directions. The use of bushes sliding on rods accomplishes this. The weight and strength of the rods has to be considered. The weight of the rods in the 'X' direction does not matter as they form the base. However, they need to be strong, so they are made from steel. The 'Y' direction rods have to be light to minimise the torque on the cantilever arm, and they do not have to support much weight. Therefore, aluminium is used for these.

Possible solutions for the gripper include the use of a suction cup, a set of pincers and an electro-magnet. Of these solutions the electro-magnet was chosen, as it is very easy to control. However, this implies that the pieces have to be made out of a ferrous material. So the playing pieces were fabricated out of mild-steel.

4.2 The Step Waveform Generator

As open loop control is used for the stepper motors, it is necessary to ensure that the motors do not stall and miss steps otherwise the position of the manipulator can differ from that expected. Accelerating the motors up to the required speed can reduce the chance of missing steps. Therefore, the step waveform generator is required to provide the necessary waveform to drive the stepper motors as well as accelerating the motors.

The checkers playing system was designed to run on the Windows operating system, which is not a real time operating system. This means the step waveform cannot be implemented on the host computer, as the operating system may not provide a continuous output causing steps to be missed. The solution is to use a microcontroller that accepts a motor position sequence consisting of the number of steps to take in each direction as inputs, and generates the corresponding waveforms as outputs.

The number of steps to take in each direction is sent to the microcontroller via an RS-232 connection. A timer is then used to generate an interrupt every 24ms upon overflow. This series of timer interrupts can be used to toggle output pins at the necessary times to generate the required output to drive a stepper motor. The acceleration of the motors is achieved by decreasing the time interval between the points that the output can toggle at.

4.3 The Motor Position Sequence

The sequence of motor positions sent to the microcontroller requires calibration of the number of steps per pixel. As the position of the game board is fixed relative to the manipulator, the calibration of the number of steps to each edge of the board can be calculated off-line, by counting the number of steps required to move the manipulator to the desired positions. As the number of pixels from board edge to board edge in the images as well as the number of steps for the manipulator to move to the same locations on the game board is known, the number of steps per pixel can be calculated. The number of pixels to the centre of the piece that is to be moved can then be converted to the number of steps to the centre of that piece.

The current position of the manipulator is maintained in terms of 'steps'. The 'step' co-ordinates that follow the initial move can be determined by subtracting the current manipulator position from the next desired position. The move sequence can then be passed to a micro-controller.

4.4 The Motor Driver Circuit

The output from the micro-controller is only 3.3 Volts with relatively low current driving capability. A motor driver circuit is therefore required to switch the windings between 0 and 12 Volts. This allows the stepper motors to draw the necessary current to move the manipulator at the desired rate due to the fixed resistance of the motor coils.

The motor driver circuit is designed to be an expansion board for the micro-controller development board. This consists of three L298N dual 'H-bridge' motor driver chips and supporting circuitry that allows the direction of the current through the motor coils to be controlled, and therefore allows direction control of the motors.

5. DISCUSSION

All three subsystems (the input vision system, the game playing engine, and the output manipulator) must work together to create a game playing robot.

The vision system is very dependent on the light illuminating the game board. This means the vision system does not identify the pieces correctly if the robot is placed directly under a light source, or near a light source whose intensity is greater than that of the fixed light source used by the robot. This could be improved by including more advanced image-processing techniques to allow identification under varying ambient light. The glossy finish of the game board could also be removed.

While the system plays checkers against a human opponent by randomly choosing a legal move to make, the system requires an intelligent game engine in order to challenge the human opponent making the system more realistic.

The robotic arm can sometimes skip steps when moving the pieces, causing the pieces not to be lifted from the centre. This means that the piece is not placed in the centre of the square that the piece is being moved to. As the design of the robotic arm does not have any space to insert linear bearings, the simplest solution is to provide more power to the stepper motor driver circuit and therefore the motors.

Currently when the robots 'pawn' pieces need to be turned over to become a king, the human player is required to do this. Therefore, a mechanical system needs to be developed to perform this task to make the system more realistic.

This system, although simplistic, demonstrates the principles of a robotic board game playing robot and allows the human opponent to play interactively against the robot.

6. ACKNOWLEDGMENTS

The following was found useful in the preparation of this paper:

D Bailey, K Mercer, C Plaw, K Subramaniam, "Trax Playing Robot", *Proceedings of the IEEE International Conference on Mechatronics and Machine Vision in Practise*, pp 151-155, 2001

Game Colony, www.chesslab.com, 2000