Omar Saeed
FURSCA
7/22/2024

Building and Implementing a Chess Robot

This summer, I endeavored on building and implementing a chess robot. The goal of this project was to build a system of devices communicating with each other in order to see and recognize chess moves, come up with the best move, and then make the next move without knocking over any chess pieces. The robot consisted of a three jointed arm with a rotating wrist and gripper, a webcam, and two computer units (so far). The goals of this project led me to develop specific codes to achieve these goals.

The first of these programs deals with the computer vision aspect of the project. Firstly, a definition of computer vision is necessary. Computer vision is a branch of computer science that deals with image processing using machine learning models, AI, and other forms of analysis. My computer vision system needed to recognize any move made on the board. At first, I was waiting on materials to arrive. I set up a basic computer vision system that could analyze moves from screenshots of a digital chess game. I spent the first week of the project working on this aspect. By the end of the first week, the webcam and robotic arm had arrived. This meant I can build up my final vision system. Initially, analyzing the image was difficult. I had to balance glare and dim lighting as well as getting the system to differentiate between the board and the pieces (since they were the same colors; black and white). I solved both of these problems with one solution. To be able to apply my solution, I first needed to have the pieces painted. The white pieces got painted red and the black pieces got painted blue. Now that I have pieces with a distinct color, I can apply a mask to the image. This solved the problem of glare since it showed up as white on a final image. The mask only filtered for red colors or blue colors. At that point, there was only one step left. Adding the functionality of finding a move involved comparing an image of the board before the move was made and an image of the board after the move was made. By analyzing the differences, the program was successful in finding the moves being made by the player.

The chess engine was incorporated into the computer vision system as well. Once the image analysis aspect was done, the computer vision system produced a move in the format of the initial position of the piece and the final position (ex: e2e4). Once the vision system found out the move of the player, it would send said move to the chess engine, the chess engine would then play the player back by producing a response move. This was relatively easy to implement as there are many chess engines out there. The engine I used was stockfish.

The robotic arm was the next phase of the research. When it arrived, it came unassembled. Assembly took almost a week to complete. The screws were small and the arm had some pretty tight corners in need of being screwed. Once assembled it

comprised two main components; the arm and a chip that facilitates communications with other devices. The next phase was to connect the arm and get it moving. I used a USB connection with my computer that made it possible to connect to the arm. The next step was to find a way to create systematic communications with the arm. There were many programs that allowed you to move the arm directly, however, this wouldn't work for my project. To solve this problem, I found a way for a program to send information to the USB port the robot was connected to. The only thing left was to get the program to generate the correct information to send to the port. This was a multi step process. What is firstly necessary, is to figure out how the robot needs to be oriented to reach for a piece correctly. To accomplish this, a basic understanding of kinematic and inverse kinematics is necessary. Kinematics is a series of equations that takes in the angles of the joints and the length of the arms and produces a final position. Inverse kinematics is a series of equations that takes in a desired position and the length of the arms and produces the angles necessary for the tip of the arm to reach that position. Once the angles are given, they can be converted to inputs that the robot can understand. This aspect of the project was also achieved. However, at this point in the project, the robot does not reach the exact desired position. It is slightly off center. There are methods to deal with this that I plan on implementing for the rest of the summer.

This project will be immensely beneficial to my career as a physicist and a computer scientist. Prior to this project, I knew a lot about theoretical applications of computer algorithms and mechanics. However, I did not have the confidence or the problem solving skills to work on a project that is similar to this. After this project, I can safely say that I am confident to tackle projects like this. I also can see myself greatly improving at this craft. All of the aspects of this project can be improved upon. More sophisticated techniques exist to solve the move finding problem and the robotic movement. All that is left is to actually work on it. Future plans for this project are to present my work at Elkin Isaac and other conferences throughout the country.