The Robotic Foosball Table
(Technical Paper)

Game-Playing AI and Society
(STS Paper)

A Thesis Prospectus
In STS 4500
Presented to
The Faculty of the
School of Engineering and Applied Science
University of Virginia
In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science in Computer Engineering

By
Zachary Yahn

October 27, 2022

Technical Team Members:

Coleman Jenkins
Aidan Himley
Jake Long
Hudson Burke

On my honor as a University student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments.

ADVISORS

Harry Powell, PhD, Department of Electrical and Computer Engineering

Bryn E. Seabrook, PhD, Department of Engineering and Society

Introduction

What do we do when artificial intelligence (AI) begins to ask us: “shall we play a game?” How do we respond to systems that can challenge even the best human players at games that used to be bulwarks of human intelligence? Not only are these AI growing ever stronger, but their victories over human opponents also increasingly appear on televisions, streams, and social media. This proposed research paper aims to understand how the popularization of world champion AI affects the public’s perception of and policy’s reaction to these systems. Does it make us more aware of the dangers and often subtle influences they purport, or does it trivialize and gamify the very concept of AI itself?

This project also corresponds to an electrical and computer engineering capstone project: the robotic foosball table. Our group is designing a robot that plays half of a game of foosball (a form of table soccer), competing against a human who controls the other half. The goal is to provide enjoyable yet challenging competition for humans and potentially even surpass them. The robotic foosball table also offers another datapoint for how humans respond when we are no longer the best at games. As a result, these two projects will produce two outcomes. First, a technical project will involve a robot integrated with a physical foosball table. Second, an STS project will analyze the interplay of game-playing AI and society. These projects will combine in an Undergraduate Thesis Portfolio, consisting of an Executive Summary, a Technical Report, an STS Research Paper, and this Prospectus.

Technical Topic

The robotic foosball table will be a standard foosball table augmented with motors, cameras, and other sensors to enable automated playing on one side of the table. When brainstorming ideas for a capstone project, the group’s emphasis was on a project that was interactive and fun to play against. The computer will play as one team, in this case, the red shirt team, and a human will play as the other, the blue shirt team. The goal is for the computer system to be at least as good as a typical human player, although the design will allow for significantly above-average performance as well.

This project is divided into four subsystems, which define the flow of information through the system. First is the camera, which will stand on a tripod above the table. This camera will send RGB (red-green-blue) video frames to a Raspberry Pi, a small computer that fits in one's palm (Raspberry Pi Foundation, 2022). The Raspberry Pi will run a fully modified Linux operating system, so that the code running the custom image-processing software has highest priority in the CPU (central processing unit) (Linux Foundation, 2022). The image processing software will then segment the pixels of the image by using how close they are to the known color value of the ball. This ball will be a shade of blue to stand out from the rest of the table. The image software will thus look at each pixel and only keep the ones that are close to this shade of blue, which are the pixels of the ball. This will all function using the popular programming language Python and its OpenCV library (Python Software Foundation, 2022), (Open Source Computer Vision, 2020). Once the image processing software has identified the center of the ball, it will also use the center from the previous frame to calculate the ball's current velocity. It will calculate these values for every frame at thirty frames per second. It will then relay these position and velocity coordinates to the next subsystem: the microcontroller.

A microcontroller is a circuit board capable of programmable logic but does not include some of the key components of a full computer such as an operating system. The robotic foosball table will specifically make use of an MSP432, a popular microcontroller with sufficient memory and processing power for this application (Texas Instruments, 2017). This microcontroller will be responsible for planning the movements of the computer's players. Given the position and velocity of an incoming ball, the path planner will calculate where to move the players and how quickly. Strategically, the goal of the system is to play near-perfect defense and mediocre offense, noting that human players are more likely to eventually mess up and let a goal in on themselves. Simply put, the system wants to win a game of attrition. The microcontroller will also update its planning with every new frame that it receives, with each triggering a hardware interrupt that tells its software to recompute player trajectories. This microcontroller will then feed these trajectories to the next system: the hardware and printed circuit board (PCB).

The PCB will interpret instructions from the MSP432 and turn them into motor commands. Running a motor is more complicated than simply turning it on and off, and so this middleman will be necessary to gain finer control of the system. All hardware design will utilize KiCAD, a popular circuit design suite (KiCAD, 2022). The hardware will also have to consider several factors including thermal dissipation and handling high current. Its design will follow the regulations and guidelines provided in the datasheets of the motors and sensors. The hardware will also use a two-way communication with the MSP432, meaning it will also be able send data back to the microcontroller. The MSP432 will enable a form of closed-loop control with adaptive control. In other words, the hardware will send instructions to the motors, determine if those instructions created the desired output, and then send this feedback back to the MSP432 so that it can determine better instructions on the next loop. The hardware then connects to the final subsystem of the robotic foosball table: the mechanical assembly.

The mechanical assembly will consist of the motors, the belt-and-pulley systems that they move, and a few smaller components that are beyond the scope of this overview. There will be four motors total: two for rotational control of both rods, and two for linear control. These motors will all have rotational encoders that report how far the motor turns at any given time. The mechanical assembly will also include several custom 3D printed parts. The 3D modeling software of choice for this project is Fusion 360 (Autodesk, 2022). Put together, this system will be capable of propelling the ball forward at 1.3 meters per second and reacting to a similar speed. This will exceed the maximum hit speed and reaction speed of human players as determined by a study conducted for this project. By segmenting the project into these four subsystems, team members can complete the different components of the project in parallel. This project will conclude by the end of the Fall 2022 semester. By that point it will compete with, and defeat, a human player.

STS Topic

Game-playing AI first took the popular stage with the widely televised Deep Blue vs. Garry Kasparov match, when Deep Blue, a computer system designed by IBM, defeated the then world champion in chess in 1997 (Campbell et al., pp. 57-83). Years later, the AlphaGo vs. Lee Sedol match once again shocked the world, especially because many experts predicted that computers would not be that capable at Go for another decade (Silver et al., pp. 484-489). Both events garnered hundreds of millions of viewers. While these are impressive examples of AI in action, the controlled environments of games are starkly different from the real implementations of equally competent AI, such as in education systems (Perrota et al., 251-269). There is a complex interaction between the companies that deploy these systems, the governments that regulate them, and the people that watch them. The effects of how these systems are popularized, as well as what their intended goals are, innocently playing a game contrasted with determining a child's educational pathway, may have implications for the ways they are developed in the future and affect lives in the present.

To better understand these complex systems, the analysis will make use of two STS frameworks. The first is technological fix, which analyzes the way in which technology is applied as a solution to problems, often unnecessarily (Newberry, pp. 1901-1903). A key part of this theory is framing the problem correctly and objectively, to better understand whether or not technology is necessary at all. One criticism of technological fix is that it might dismiss technologies without fully considering the potential benefits and drawbacks (Scott, 2011). Second, technological momentum describes how technology and society interact over time, and how the progressive integration of a technology with society affects how the members of that society interact with it (Hughes). Importantly, the framework considers how technology and society interact in a cycle where both induce changes in the other. These two frameworks provide a crucial analysis of how game-playing AI interacts with society. Understanding this interaction might inform the development of future AI, its publicization, and its use cases.

Research Question and Methods

The research question for the STS research paper is: How does the publicity of game-playing AI systems like AlphaGo and DeepBlue affect peoples' perception of progress towards personal and societal applications of AI? Key aspects of this problem, which inform the keywords used to search for resources, include artificial intelligence, ethics, public opinion, game-playing, machine learning, and big data. These are the keywords that commonly relate to AI and its interactions with society. The sources used to answer this question will fall into three categories: case studies, surveys, and policy analysis. Case studies include examples of popular AI in the past, surveys are limited to those already conducted by other researchers, and policy analysis will include current AI policy and its focus on popular AI. In order to support the components of this research question, crucial sources will analyze the ethics of modern AI, such as the work of R. Alberto. These will support or refute the research thesis by understanding this problem with STS frameworks. Others, like Bunz et al., will discuss how AI is portrayed in the media and how that portrayal affects public opinion (pp. 9-32). These sources will instead provide the necessary background information and understanding of the current state of the problem.

Conclusion

The goal of this research endeavor is twofold: to engineer a technical project, and to approach and answer an STS research question. The technical project aims to produce a robotic foosball table capable of competing with a human opponent. The table makes use of computer vision, path planning, embedded electronics, and other contemporary technologies. The STS research question, on the other hand, strives to understand the interplay of popular game-playing AI systems with society, including how the public views other systems applied to everyday life. The technical project provides a unique vantage point for this problem, by offering a physical example of a game-playing system that people can actually touch and feel. Meanwhile, the STS research question aims to better understand how AI is really involved in society, potentially leading to safer, more ethical implementations of the technology. Together, these projects will shed light on an increasingly important technology, by creating and analyzing scenarios where humans and artificial intelligence collide.

References

Alberto, R. (2022). Images of artificial intelligence: A blind spot in AI ethics. *Philosophy & Technology* 35(1), doi:10.1007/s13347-022-00498-3

Autodesk. (2022). *Getting started with Fusion 360*. https://help.autodesk.com/view/fusion360/ENU/?guid=GUID-1C665B4D-7BF7-4FDF-98B0-AA7EE12B5AC2

Bunz, M., Braghieri, M. (2022). The AI doctor will see you now: assessing the framing of AI in news coverage. *AI & Society* 37(1), 9-32. doi:10.1007/s00146-021-01145-9

Campbell, M., Hoane, J., Hsu, F. (2002). Deep blue. *Artificial Intelligence* 134(1), 57-83. https://doi.org/10.1016/S0004-3702(01)00129-1

Hughes, T. P. *Technological Momentum*. (1994). Cambridge, Massachusetts. London, England. The MIT Press.

KiCAD. (2022). *KiCAD Docs*. https://docs.kicad.org/

Linux Foundation. (2022). *The Linux kernel*. https://docs.kernel.org/

Newberry, B. (2005). Technological Fix. In C. Mitcham (Ed.) *Encyclopedia of Science, Technology and Ethics*. (Volume 4., pp. 1901-1903). New York, New York. Macmillan Reference USA.

Open Source Computer Vision (2020). *OpenCV: OpenCV modules*. https://docs.opencv.org/4.x/

Perrota, C., Selwyn, N. (2019). Deep learning goes to school: toward a relational understanding of AI in education. *Learning, Media and Technology* 45(3), 251-269. https://doi.org/10.1080/17439884.2020.1686017

Python Software Foundation. (2022). *Python*. https://www.python.org/

Raspberry Pi Foundation. (2022). *Raspberry pi documentation*. https://www.raspberrypi.com/documentation/

Scott, D. (2011). The technological fix criticisms and the agricultural biotechnology debate. *Journal of Agricultural and Environmental Ethics* 24, 206-226. https://doi.org/10.1007/s10806-010-9253-7

Silver, D., Huang, A., Maddison, C. et al. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature* 529, 484–489. https://doi.org/10.1038/nature16961

Texas Instruments. (2017). *Documentation overview for SimpleLink MSP432 SDK*. https://softwaredl.ti.com/simplelink/esd/simplelink_msp432_sdk/1.40.01.00/docs/Documentation_Overview.html