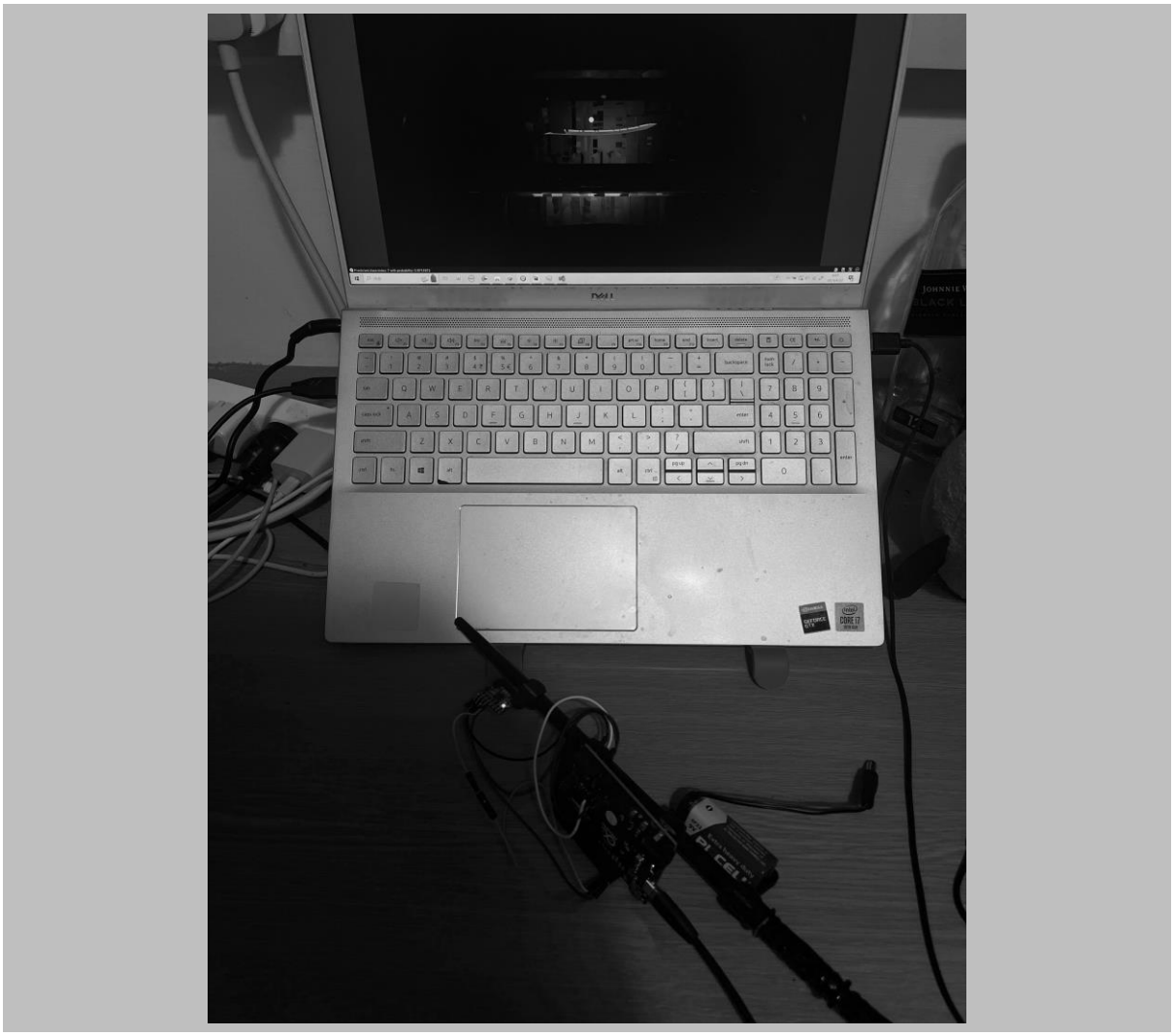


# Infinite Wand Project with ML

Yijun Ma

Through machine learning training of action models, integrated into Unity applications, and paired with my cyber wand, to create a magic interaction logic based entirely on motion recognition, say goodbye to buttons.

https://youtu.be/UVLyB4cW7W8
https://vimeo.com/937585639?share=copy

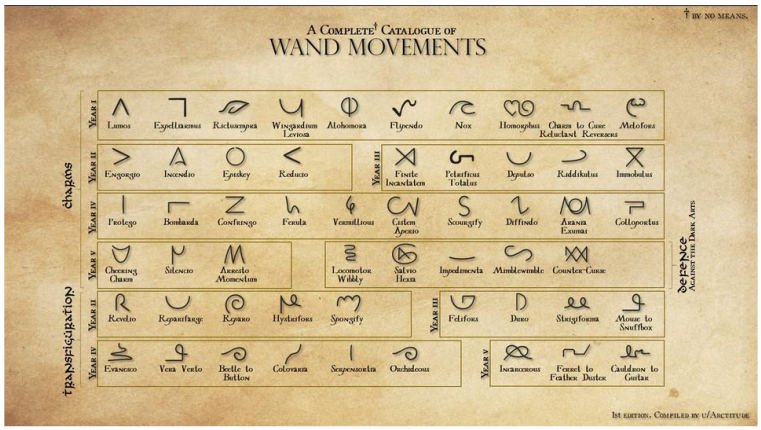
# Introduction

This project is an extension of my cyber wand, a modular wand interaction device based on physical computing, and serves as the final assignment for this course. In this project, I trained a deep learning model capable of recognizing specific motions and integrated it into a Unity project. Paired with my cyber wand, I designed interactions where different gestures trigger different spells (such as turning on lights, levitating feathers, and restoration), making the user's movements a decisive factor in the experience. Through this technology, real-world individuals can cast spells in a virtual world, creating a sense of "magic reenactment."

# Concept and Background Research

This project's concept is inspired by Giorgio Agamben's reflections on what magic is and its relationship with joy, as well as my own thoughts on human-computer interaction. Regarding the former, Agamben suggests that magic is a source of joy found in children's experiences. Magic lies in invocation; it is a kind of secret science, an unspeakable code that exists beyond our experiential world and the knowledge-based symbolic systems. As for the latter, upon realizing that most current human-computer interactions are controlled by "buttons," I believe that in order to prevent dialogue between humans and machines from becoming a form of labor, it is necessary to make human-computer interaction a kind of magic. This entails eliminating existing button mechanisms and instead, on the basis of reprogramming, allowing machines to respond to human invocation. This may involve playfulness, but it also needs to evoke a sense of novelty in users.

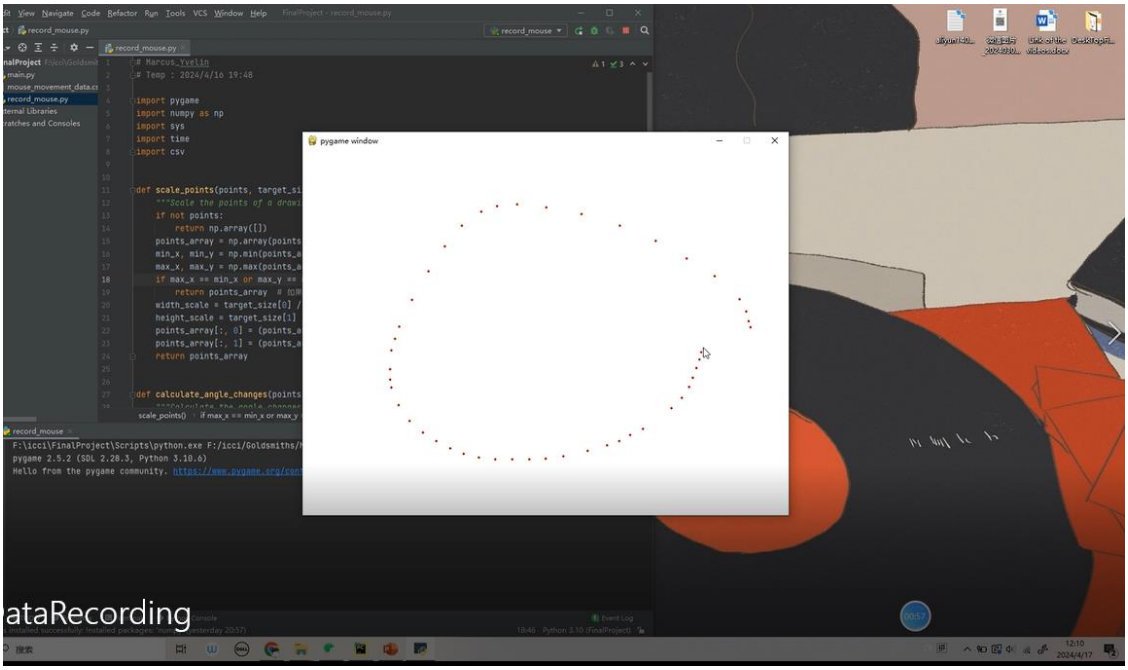
However, the concept of magic is abstract and imaginative. Fortunately, I happened to have a Warner Bros. Harry Potter wand toy at hand, which became the intermediary for my interaction. Inspired by Harry Potter, the spells also became the content of this project's initial experiments. As an experimental project, I encoded certain spells based on the Harry Potter spell system and then completed an interactive program that users can experience.



## Technical Implementation

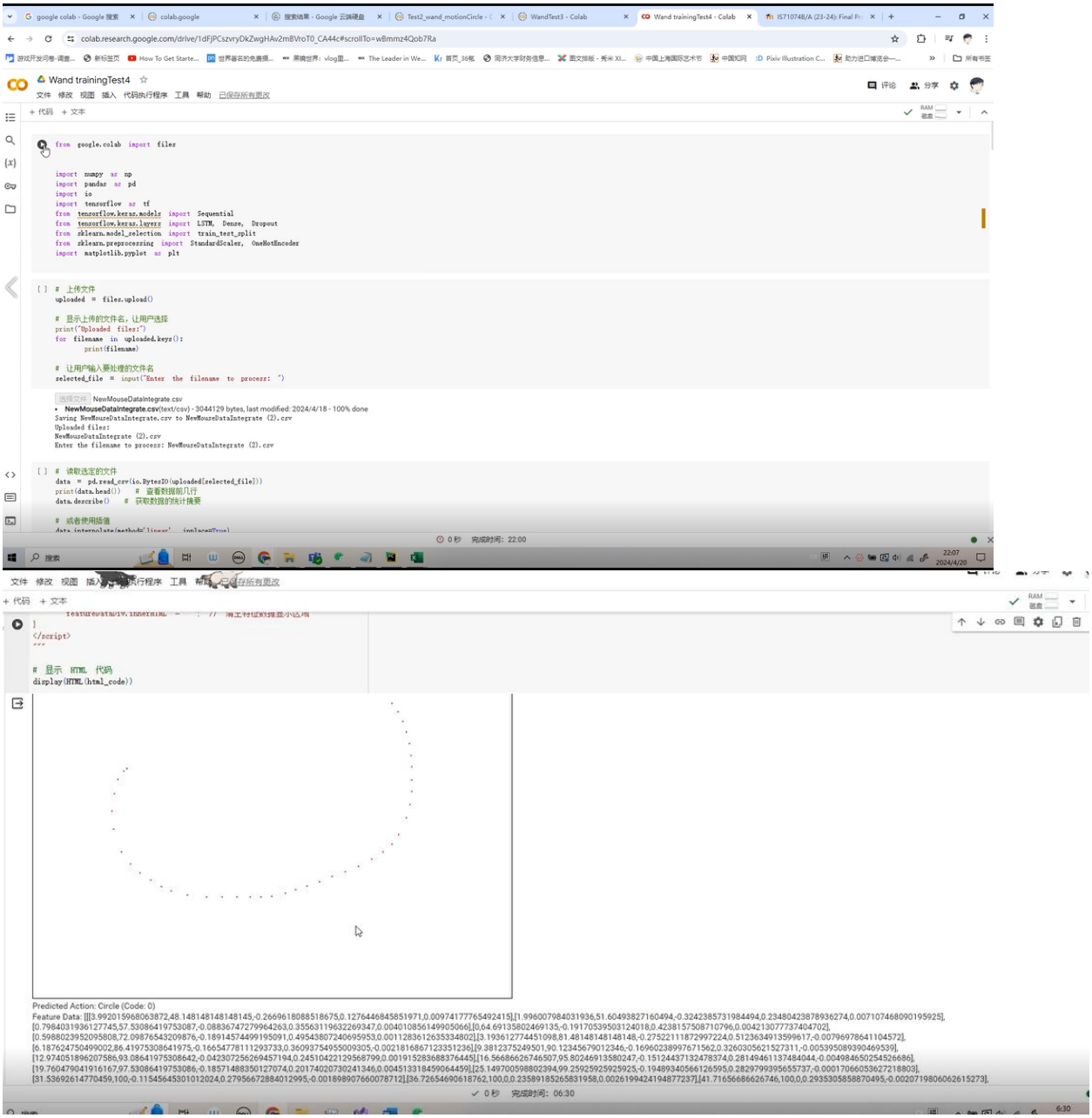
Due to the completion of physical computing for the wand, this project is divided into two technical aspects: model training and Unity development. Another aspect is enabling Unity to call the trained model.

In the model training phase, I used Python to write data collection code. Through a Pygame window, 50 path coordinates of mouse movements for different actions, along with corresponding changes in angles, speeds, and accelerations, were output. In the end, I obtained three specific actions and one interference action, totaling 800 sets of 40,000 data points. These were stored in a CSV file with proper labeling.

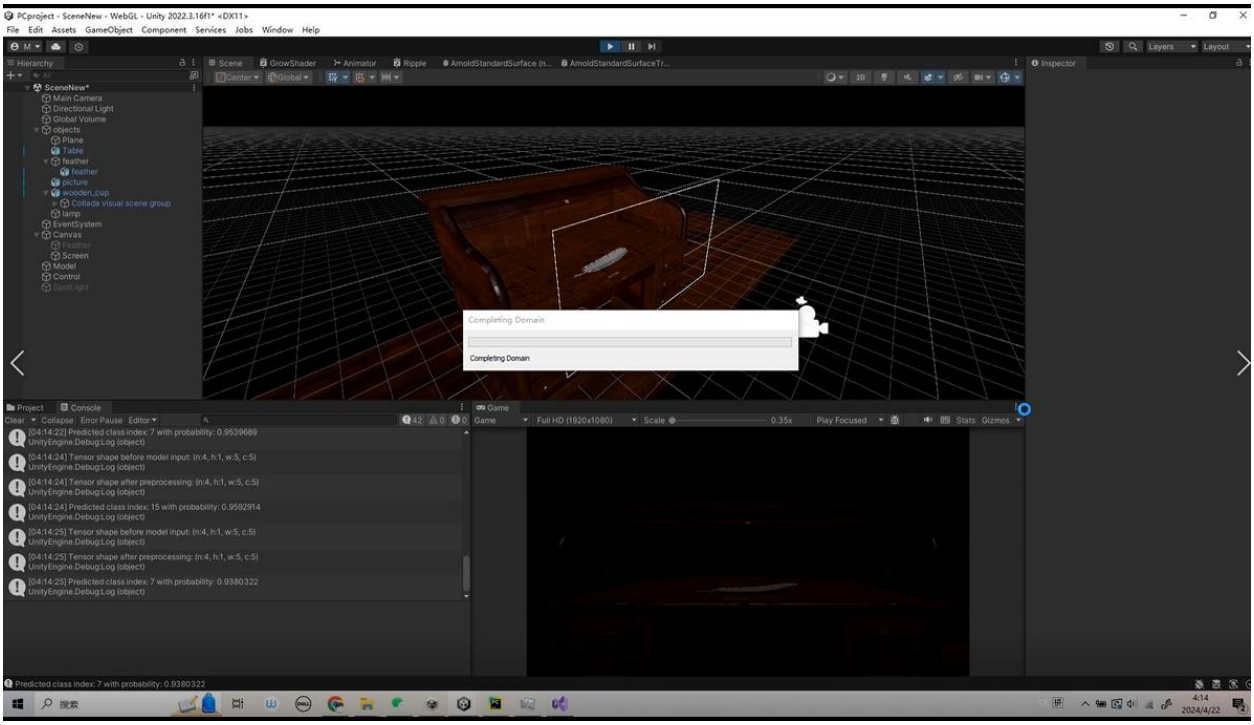



Afterward, I used Python on Google Colab to train an LSTM model, importing the dataset I collected. I exported an h5 model and tested it using a canvas program created with front-

end HTML code on Colab.



Subsequently, I imported the model, which was converted to the ONNX format using Barracuda, into Unity. I utilized canvas to obtain mouse data, processed the data, and passed it to the model. Based on the model's output, I triggered different feedback effects, such as turning on/off lights, making feathers float, and stopping actions. After completion, I demonstrated the project within the Unity environment.




最后接通魔杖程序实现通过魔杖的动作交互。

Finally, I connected the wand program to enable interaction through wand movements.

## Reflection and Future Development

The project currently has two significant shortcomings. Firstly, there may be issues with model training and loading, leading to insufficient recognition accuracy. Particularly, there is a noticeable difference in performance between Unity and Colab testing. Therefore, it is necessary to further optimize the model accuracy, model referencing in Unity, and data processing. Secondly, there is a lack of diversity in trained actions and designed interaction effects. With only three types currently available, a more comprehensive interaction system needs to be developed, both in terms of quantity and variety. To address these issues and enhance user experience, continuous optimization is required. Additionally, the visual impact of the screen itself can be improved to envelop users in a more immersive "magical" atmosphere.

## References

Theory from: Giorgio Agamben – Magic and Happiness, Profanation(2007)

No tutorial used.
With the help from GPT 4, as a tool. All these codes, in over 400 dialogues.
The whole idea is mine.