# Project: RealGestureX

Solo project: Luigi Liu (ll3840)

## Introduction

I plan to develop an Integrated Static and Dynamic Hand Gesture Recognition System capable of real-time recognition of both static and dynamic hand gestures. By utilizing advanced hand tracking technologies and deep learning models, this project aims to create an intuitive interface for human-computer interaction, with applications in smart home controls, accessibility tools, and interactive gaming.

## Objectives

**Primary Objective**: To design and implement a gesture recognition system that accurately identifies a set of predefined static and dynamic hand gestures using PyTorch.

**Secondary Objectives**:
1. Develop a custom dataset that includes both static and dynamic gestures for model training and validation.
2. Integrate hand tracking using MediaPipe Hands to extract meaningful hand landmarks.
3. Optimize the model architecture to achieve real-time performance with high accuracy.
4. Deploy the system in a user-friendly interface that maps recognized gestures to specific commands.

## Project Plan for Completion

The system will recognize a range of gestures, including static gestures such as pointing, open palm, thumb and index finger touching, thumb and middle finger touching, and fist, as well as dynamic gestures like swipe up, swipe down, swipe left, and swipe right.

**Custom Dataset Creation**: Given the specificity of the required gestures, a custom dataset will be developed to ensure comprehensive coverage and high-quality samples. Each gesture sequence will be accurately labeled, with separate directories for training, validation, and testing sets to facilitate unbiased model evaluation. The dataset creation will involve using a hand tracking model, such as MediaPipe Hands, to extract and save hand landmarks for model training. Temporal augmentation may be applied to vary the speed of gesture execution, capturing different movement dynamics, and noise injection will introduce slight variations to landmark positions to enhance the model's robustness against real-world variances.

Tools and Frameworks:
- Camera Setup: Use high-resolution webcams to capture clear images and videos of hand gestures.
- Hand Tracking: Implement MediaPipe Hands to extract 21 3D hand landmarks per frame, providing detailed spatial information.

## Network Architecture Design

In this project, we use a dual-model architecture specifically designed to handle both static and dynamic gestures, with hand tracking providing the landmarks that serve as input features. A hand tracking model, such as MediaPipe Hands, extracts 21 hand landmarks per frame, which are then processed by our gesture recognition system.

1. **Static Gesture Model**: The first model handles static gestures, such as pointing, fist, and open palm. This model processes only a single frame of landmark data to recognize gestures that do not rely on movement over time. Its simplicity enables it to operate efficiently in real-

time, as it processes a single frame at a time. Initially, I plan to use a basic neural network with ReLU activation and linear layers to keep the model lightweight and responsive. Depending on the initial performance results, I may adjust the network structure or incorporate additional layers to optimize accuracy and reliability.

2. **Dynamic Gesture Model**: The second model is designed for dynamic gestures, such as swiping up, swiping down, and other gestures with directional movement. For this model, we utilize an LSTM (Long Short-Term Memory) network to handle sequences of landmark data across multiple frames, allowing it to capture temporal information and effectively differentiate between gestures based on movement patterns. LSTMs are well-suited for recognizing temporal sequences, making them ideal for analyzing motion-based gestures over time.

**Advantages of This Architecture**:
- Real-Time Efficiency: By using two models tailored for specific gesture types, the system focuses computational resources on relevant feature sets for each gesture type, enhancing responsiveness for real-time scenarios.
- Gesture Type Specialization: The static and dynamic models can be fine-tuned separately, improving recognition accuracy for each gesture type compared to a single model handling both types.
- Modular Design: The modular approach allows independent updates and optimizations to each model, providing flexibility for future expansions, such as adding new gestures or adjusting recognition parameters.

**Disadvantages of This Architecture**:
- Increased System Complexity: Managing two models requires additional considerations for integration, model switching, and calibration to ensure consistent performance across gesture types.
- Higher Resource Consumption: Running two models concurrently, especially with an LSTM for dynamic gestures, may demand more memory and processing power, potentially challenging for devices with limited resources.

**Suitability of LSTM Over CNN or Complex Models**:
While Convolutional Neural Networks (CNNs) and other complex architectures could provide detailed spatial feature extraction, they are generally less suited for real-time applications due to their high computational demands. Real-time gesture recognition requires models that can process data swiftly to ensure a smooth user experience. Complex models like deep CNNs would likely introduce latency, slowing down response time and impacting interactivity. Instead, the LSTM model for dynamic gestures strikes a balance between accuracy and speed, allowing for efficient temporal pattern recognition without significant computational delays. This choice aligns with real-time processing requirements and ensures the system remains responsive for live interaction.

# Innovation and Originality

The proposed dual-model architecture for hand gesture recognition introduces a novel approach by integrating specialized models for static and dynamic gestures, utilizing hand tracking for landmark extraction. This design leverages MediaPipe Hands to extract 21 hand landmarks per frame, serving as input features for the recognition system.

**Key Innovations**:
- Dual-Model Architecture: By employing separate models for static and dynamic gestures, the system optimizes computational resources and enhances recognition accuracy for each gesture type.
- Integration of LSTM for Dynamic Gestures: Utilizing Long Short-Term Memory (LSTM) networks for dynamic gestures enables effective analysis of temporal sequences, capturing movement patterns over time.
- Real-Time Processing Capability: The architecture is designed to operate efficiently in real-time scenarios, ensuring prompt and accurate gesture recognition suitable for interactive applications.

Several studies have explored hand gesture recognition using similar methodologies:

- “Automatic Indian Sign Language Recognition Using MediaPipe Holistic and LSTM Network" (2023): This research employs MediaPipe Holistic for feature extraction and LSTM networks for recognizing isolated words in Indian Sign Language, achieving notable accuracy rates.
- "Next-Gen Dynamic Hand Gesture Recognition: MediaPipe, Inception-v3 and LSTM-Based Enhanced Deep Learning Model" (2024): This study presents a hybrid model combining MediaPipe for hand tracking, Inception-v3 for feature extraction, and LSTM for temporal analysis, focusing on dynamic gesture recognition.
- "MediaPipe with LSTM Architecture for Real-Time Hand Gesture Recognition" (2024): This paper proposes a real-time gesture recognition system integrating MediaPipe for hand detection and LSTM for gesture classification, demonstrating high accuracy on custom datasets.

While existing studies have utilized MediaPipe and LSTM for gesture recognition, the proposed system distinguishes itself through a specialized dual-model approach, implementing separate models for static and dynamic gestures, which allows for tailored optimization and enhances recognition performance for each gesture type. Additionally, it offers comprehensive gesture coverage by addressing both static and dynamic gestures within a unified framework, creating a more holistic solution compared to models that focus solely on one gesture type. Finally, the system emphasizes real-time application, with an architecture explicitly designed for real-time processing to ensure responsiveness and suitability for interactive applications—a focus that may not be prioritized in other studies.

## Expected Outcomes and Success Metrics

The expected outcomes for this hand gesture recognition system include achieving an overall accuracy of 85% or higher across both static and dynamic gestures, providing reliable recognition results for diverse gesture types. In terms of real-time performance, the system is designed to process gestures with a latency of less than 200 milliseconds, ensuring smooth, responsive interactions suitable for real-time applications. Additionally, the system aims to demonstrate robustness by accurately recognizing gestures under varying conditions, including different lighting environments, hand orientations, and across multiple users.
Currently, initial model testing has been conducted with around 100 training samples per gesture, yielding a training accuracy of 70% and a validation accuracy of 55%. Although no testing accuracy is available yet due to ongoing data collection, I anticipate that the model's accuracy will improve as more data is gathered. Moreover, fine-tuning the model's hyperparameters is expected to further enhance performance, bringing it closer to the target accuracy.

## Potential Challenges

This project faces several challenges, primarily in achieving high accuracy across both static and dynamic gestures, especially given the current lower validation accuracy. Ensuring real-time performance with under 200 milliseconds latency is crucial, requiring careful balancing of computational efficiency, particularly with the LSTM model for dynamic gestures. Robustness under varying lighting, hand orientations, and across different users will also be essential, demanding extensive testing. Finally, the dual-model architecture adds complexity in integration, requiring smooth transitions between static and dynamic gesture recognition to maintain a seamless user experience.