**Hand Gesture Interface**

**Team Members:**

* Matthew Kasper (NetID: mtkasper)
* Matej Popovski (NetID: popovski)
* Aniket Patel (NetID: apatel267)

**Problem we are trying to solve:**

* Want to create a purely gesture-based system for interacting with computers as other high-fidelity interfaces, such as keyboard and mouse, are not suitable for all types of computer devices and people.

**Why we want to solve it:**

* It is inconvenient to interact with some computer devices via keyboard and mouse (e.g. VR goggles, smart TVs, etc).
* Provide a more comfortable way of interacting with computers to those with arthritis and/or carpal tunnel.
* Overall, this appears to be a highly engaging and challenging project.

**Current State of the Art:**

* There exist many hand gesture interfaces in devices such as Meta’s Oculus VR goggles and Microsoft Kinect, but most are proprietary and inflexible for use on

**Novel or Re-Implementation**

* We are working on a problem that existing solutions already address, but our goal is to build it from scratch ourselves to better understand the underlying techniques and implementation details. This approach is challenging, but it will give us deeper insight and allow us to customize the solution as needed.
* We plan to start by developing our own code (from scratch), but if necessary, we will integrate selected external libraries to enhance functionality and improve performance.
* For example, we can use MediaPipe Hands for landmark detection (existing solution) but design your own classification logic or ML model on top of those landmarks.

**Possible steps to solve the problem:**

* Create a simple project webpage to host information about our project
* Create an object detection system to create bounding boxes around the hand in an image. Or use existing methods to create the bounding boxes, such as YOLO or a cascade classifier.
* Create a custom neural network using pytorch or tensorflow to classify images of hands to specific gestures.
* Based on the classified gesture, perform certain actions (mouse movement, mouse click)
* Define the target gestures and the use case (static vs. dynamic).
* Capture video input and detect the hand region (MediaPipe or OpenCV).
* Extract key features (landmarks, distances, angles) from the detected hand.

**Performance Evaluation**

* The performance of the solution will be measured using standard techniques: we will have a training and testing set of hand gesture images.
* Furthermore, we will also be performing real-time tests to make sure the model predicts gestures correctly given a video feed.

**Rough Time:**

* **Weeks 1–4:** Define project scope and target gestures; set up environment and libraries (OpenCV, MediaPipe, etc.).
* **Weeks 4–8 (30 Oct):** Implement object detection (bounding boxes) of hand and static gesture classification.
* **Weeks 8–13 (11 Dec):** Map hand position and gestures to actions on the computer. Create a website to present the project.