Hand Gesture Recognition for Interaction with Computers

Group 7: Sukanya Baichwal, Hailiang Dong, Hasmitha Jalla, Ananya Reddy Katpally

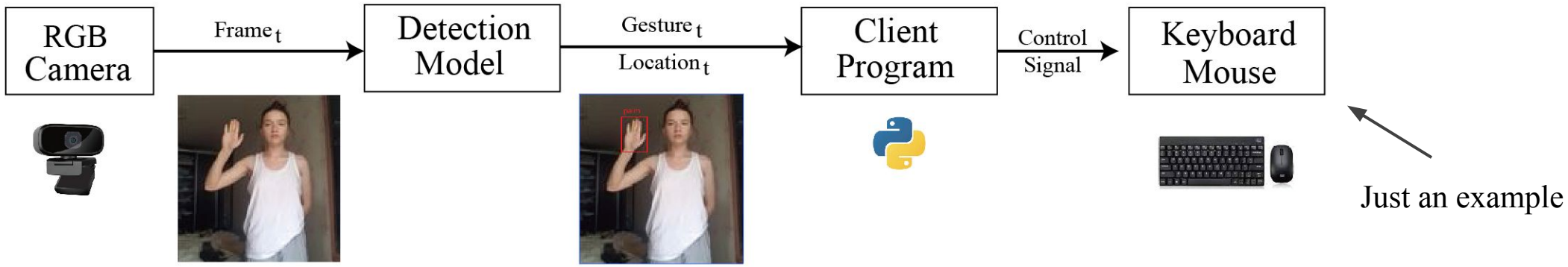
Overview

Goal

- Design a computer vision system that can be used to control (interact with) the computers through standard RGB camera in real-time (e.g. mute the PC using the stop gesture) !

Approach

- We train an object detection model for hand gesture recognition (why not a classification model ?).
- We design and implement a python client program that (1) reads frame from camera; (2) detect the gesture using above model; (3) conduct user defined actions based on the recognition results.



Object Detection - Model

- We use the **Detectron2** framework to train a customized object detection model.
- The model architecture we chosen is **Faster RCNN** with Feature Pyramid Networks (**FPN**), and use **ResNet-101** as the backbone to extract the features from the input image.
- The model is pre-trained on COCO dataset.

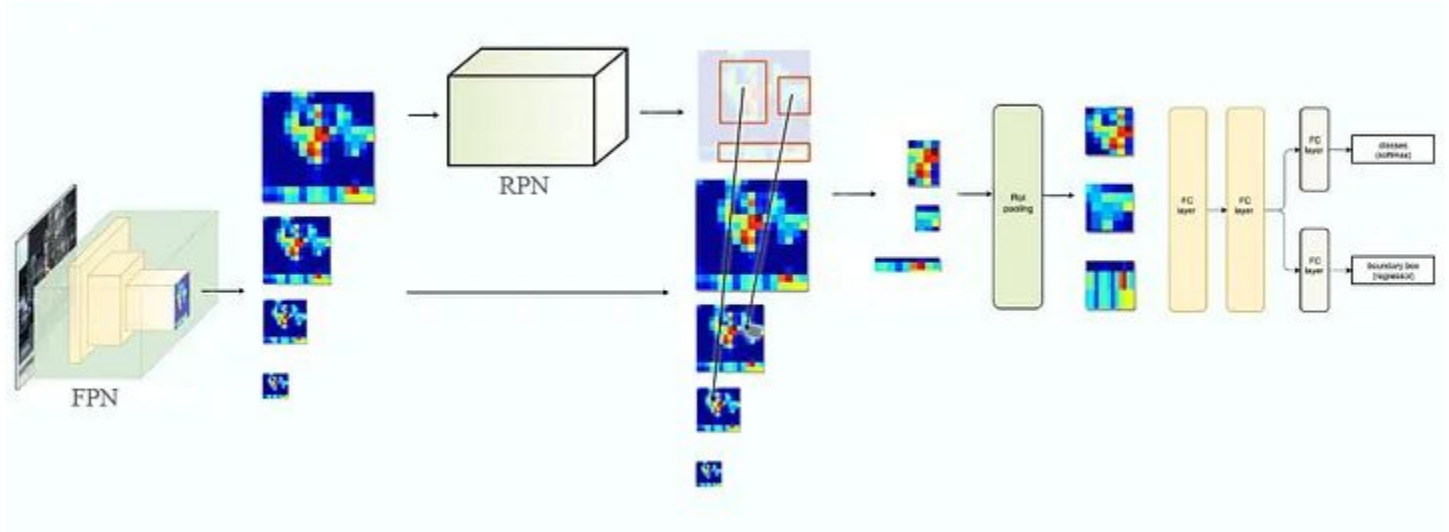
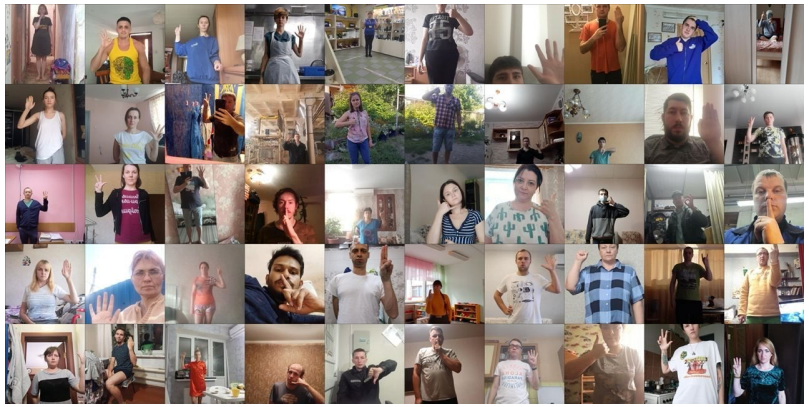


Figure from https://jonathan-hui.medium.com/understanding-feature-pyramid-networks-for-object-detection-fpn-45b227b9106c

Object Detection - Dataset

Hand Gesture Recognition Image Dataset (HaGRID)

- Gestures are at a distance of 0.5 to 4 meters from the camera
- 34730 unique persons and scenes
- 18 classes (gestures), about 30k FullHD (1920*1080) images for each class (About 40GB disk space).



Preprocessing

- Randomly picked 1100 images from six out of 18 classes, guaranteeing each image is from a different person to maximize the diversity given limited amount of data.
- Down sample the original image to resolution 960*540, in order to reduce computation overhead.
- Use 1000 images for training and other 100 images for testing from each of the six classes.

Performance of Object Detection Model

Training details

- Trained for totally 128 epochs, with batch size 8.
- Base learning rate is 2e-5, multiply by 0.1 for every ~50 epochs.
- Takes about 1-day on one Nvidia A100 40GB GPU.

Evaluation of Bounding Box

mAP	AP-50	AP-75	AP-small	AP-medium	AP-large
83.4	97.5	96.7	70.1	79.2	85.7

Evaluation of Predicted Gesture

Detection Rate	Avg. Precision	Avg. Recall	Avg. F1	F1 Range
99.33 %	98%	98%	98%	97-99%

System Architecture - Client Program

Challenge

- The labels of each frame generated from the object detection model are quite NOISY.
- How to identify whether a certain gestures is presented ?
- How to elegantly identify the movement direction (up, down, left, right) of gestures ?

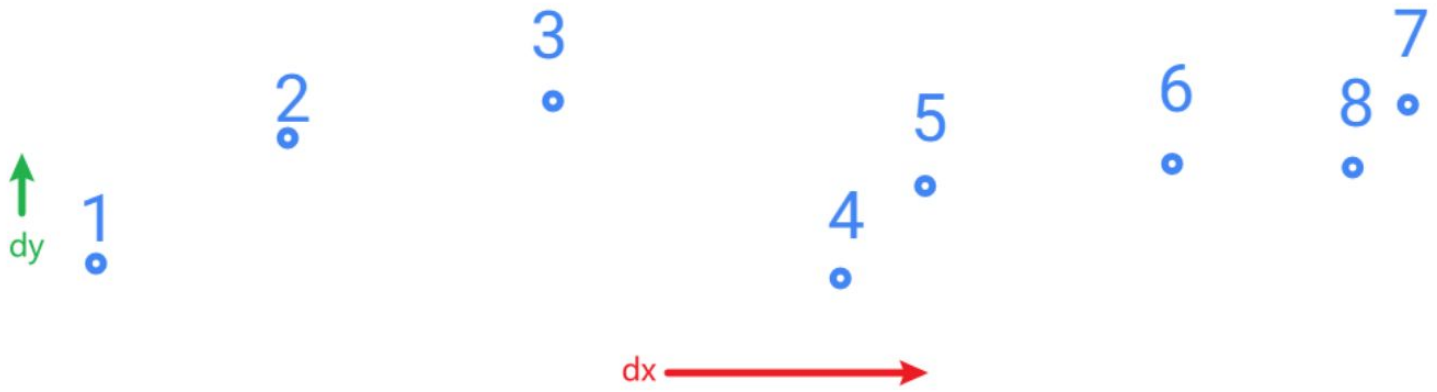
Solution

- If a gesture is consecutively detected for ~0.9 second, we think this gesture is presented and execute the corresponding action associated with it.
- To detection the movement of gesture, we use a queue to record the center coordinates of detected bounding box (may not consecutive). Once enough history is collected (~1.5s), we compute the direction based on history and execute the action.

System Architecture - Client Program

Algorithm

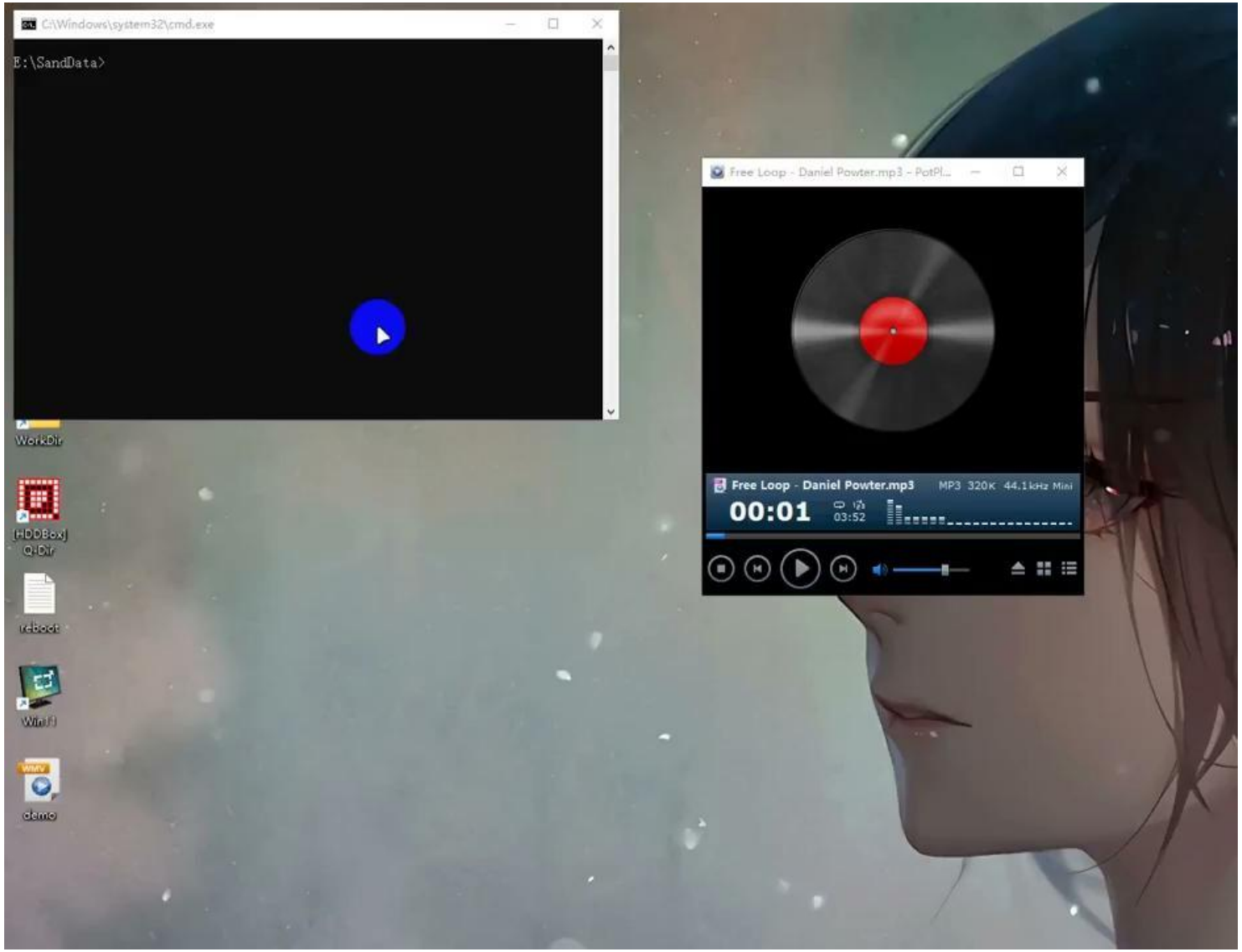
- Compute the significance of movement over both x and y axis using the extreme point
- The magnitude determines which axis we are trying to move



Important Implementation Details

- Our implementation is in the client-server style, this means the object detection model can be deployed in remote machine as a service for multiple users. (You don't need to have CUDA device locally, and you only need opencv and pyautogui libraries to run client program.)
- We use socket for send image (from client to server) and predictions (from server to client), compression is used to reduce the network overhead.
- A json file is used to define the mapping between gesture or movement to keyboard shortcuts.

Demo



Video URL

- An simple example of using our system to control the music playing along with the volume (REMOTE deployed).
- The action here is the keyboard control signal.
- Our system is NOT limited to the above type of actions. **Any action that can be implemented using Python** is compatible with our system.
- Codes will be publicly available on Github.

Questions ?