

DOCUMENTATION

Overview

This project implements Hand Gesture Recognition using python libraries that include OpenCV for image/video processing and TensorFlow for the machine learning model. The device captures the video from user's camera, processes the image/video and uses machine language model to recognize the hand gestures.

Components

The project has 2 main code files:

1. `saveimage.py`: It collects the images for the gesture recognition model.
2. `HGR_3.py`: It captures the user's video, processes the image/video for the model to recognize the gesture.

`saveimage.py`

Purpose: It was created to collect processed images in a folder so that a gesture recognition model can be trained. (The image processing applied for collecting the images is same as the one used for `HGR_3.py` script so that the model can easily recognize the processed images.) It collects the images for the following gestures: hello, rock, peace, thumbs up, ok and no detection.

How it works:

1. When the user runs the python script, the terminal starts collecting the images for each gesture or no detection from
(Line 6 `labels = ['No detection', 'hello', 'rock', 'peace', 'thumbs up', 'ok']`)
2. No of images it collects is 40 (Line 7 `number_images = 40`)
3. The `base_dir` defines where the script stores the collected images.
(Line 8 `base_dir = "C:/Users/rajpa/OneDrive/Desktop/Project 3301/collected images"`)

4. The gesture type and number of images collected for each gesture is displayed in the terminal to keep a track of what is going on.
5. Click the 'c' on keyboard to capture the images or 'q' to quit.

Customization: The code can be modified upon the user's requirements which includes the no of different gestures the user wants the model to be trained for, number of images to be collected for each gesture.

NOTE: Path given in line 8 `base_dir = "C:/Users/rajpa/OneDrive/Desktop/Project 3301/collected images"` should be modified according the user's environment path.

HGR_3.py

Purpose: Captures the video frame, processes the image and model predict the image.

How it works: It works is divided into 2 parts which is image processing for hand detection and gesture detection model for machine learning

Image Processing

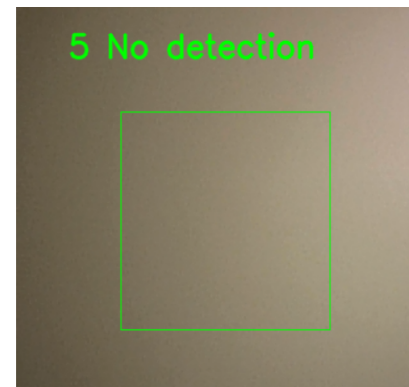
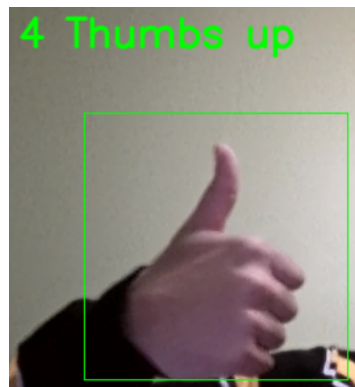
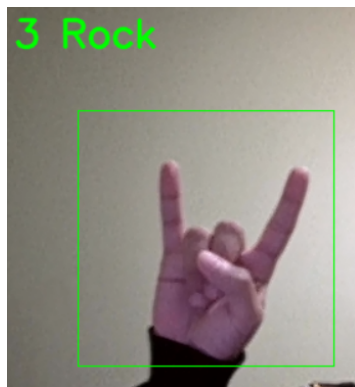
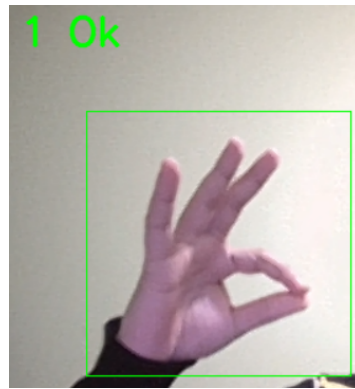
A select portion of the frame called gesture window is used where the user places their hands within its boundary. The image is taken from the frame, converted to grayscale from color image and a gaussian blur is applied on it.

The image is thresholded, such that the background is black and the hand is white. If that is not the case, then the image is inverted to make the background black and hand white after which the the inverted image is eroded and then dilated for the model to recognize the gesture.

Gesture Detection

The gesture detection is done by a model trained from web-based tool called as **Teachable Machine** which trains the model from sample images from which the model can be exported and be used.

Recognizable Gestures



Acknowledgements

Code for collecting Images - <https://www.youtube.com/watch?v=pDXdIXIaCco>
Training Machine Learning Model - <https://teachablemachine.withgoogle.com/>