Midterm Project Report
Smart Home Gesture Control Application

Technology requirements:

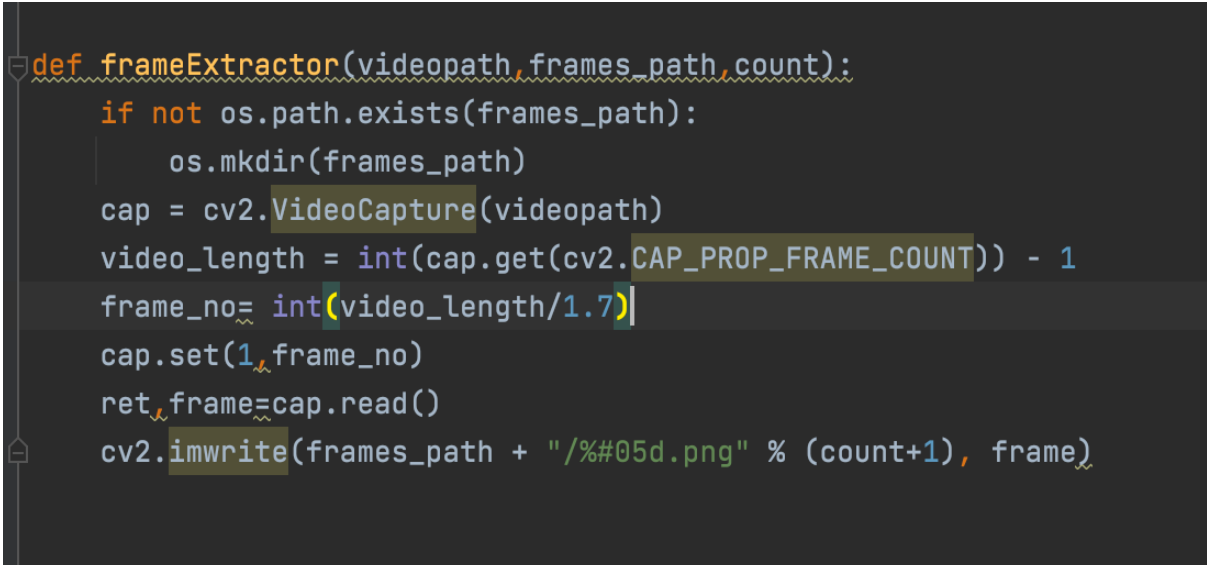
- TensorFlow
- Python 3.6.9
- OpenCV for Python
- Keras

Approach:

- I recorded the 3 videos for each gesture inorder to train the cnn model.
- I separated the data for train and testing purpose in 2:1 ratio.
- I extracted middle frames for each videos and trained the model on the basis of the images.
- After training the model I recorded the train and test vectors in a csv files.
- And I ran the model on test data to get the accuracy.

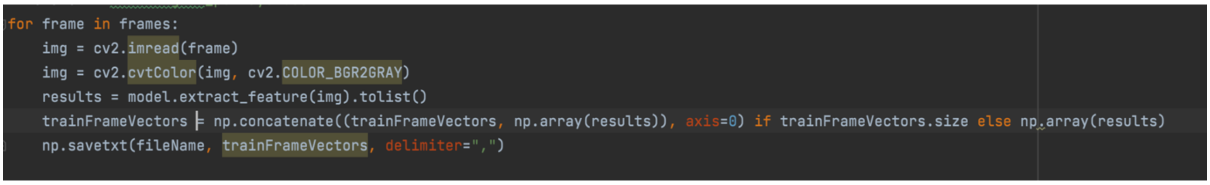
Implementation Details:

1. Get all the names of videos from the traindata and testdata folder.
2. Then we extract middle frames using function name “frameExtractor” provided in the frameExtractor.py.



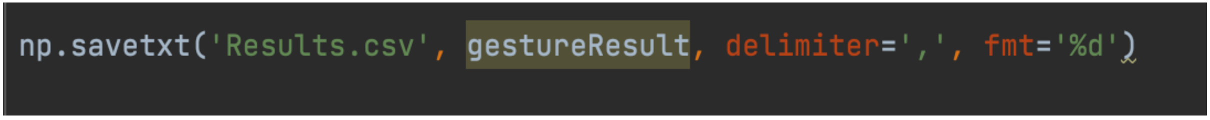
```python
def frameExtractor(videopath,frames_path,count):
    if not os.path.exists(frames_path):
        os.mkdir(frames_path)
    cap = cv2.VideoCapture(videopath)
    video_length = int(cap.get(cv2.CAP_PROP_FRAME_COUNT)) - 1
    frame_no= int(video_length/1.7)
    cap.set(1,frame_no)
    ret,frame=cap.read()
    cv2.imwrite(frames_path + "/%#05d.png" % (count+1), frame)
```

3. After extracting frames in the form of PNG files, each frame one by one is feed to the CNN model for training the model by calling method named model.extract_feature() which is present in handshape_feature_extractor.py



```python
for frame in frames:
    img = cv2.imread(frame)
    img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    results = model.extract_feature(img).tolist()
    trainFrameVectors = np.concatenate((trainFrameVectors, np.array(results)), axis=0) if trainFrameVectors.size else np.array(results)
    np.savetxt(fileName, trainFrameVectors, delimiter=",")
```

4. Repeat the above three steps for the testdata.
5. We extract the penultimate layers of Test and Train Dataset and store it in csv named ‘testset_penLayer.csv’ and ‘trainset_penLayer.csv’
6. After getting the penultimate layers of train and test dataset, we apply cosine similarity between the vectors of the gesture video. If the cosine similarity is max which is obtained after comparing the vectors . If we are using cosine distance we will take the minimum values.
7. Store the gesture number in Results.csv file



```
np.savetxt('Results.csv', gestureResult, delimiter=',', fmt='%d')
```