Smart Home Gesture Control Application

Name: Aditi Shashank Joshi
ASU ID number: 1222838916

Problem Statement:

To develop a python application classifying Smart Home gestures using a CNN model

Objectives:

- Develop a python application that classifies specific gestures
- Train and test a CNN model

Technology Requirements:

- TensorFlow
- Python 3.6.9
- OpenCV for Python
- Keras

Goal Description:

This project is an extension to the Smart Home Gesture Control Application that I had built as a part of Assignment 2. The objective of this project is:

1. To learn to build an end-to-end application based on mobile computing and fog server for smart home control.
2. To learn the Implementation of Basic Machine Learning Algorithms

Implementation:

1. Training of CNN Model:
   A model was made, compiled, and trained on several training video frames of hand gestures using the Keras API in TensorFlow.

2. Main Program:
   1. The training expert videos are kept in the "train" folder.
   2. The "frameextractor.py" program is used to extract the middle frames from all the training videos, thus saving 1 frame for each video in "Frames_Train" folder.
   3. The HandShapeFeatureExtractor class is used to extract feature vector for each hand gesture image.
   4. The obtained feature vector is the penultimate layer for the training dataset.
   5. A numpy array for each image converted to grayscale was generated and passed through the model.
   6. Similarly, for the test videos also, middle frame is found out and feature vector is created and stored in a numpy ndarray
   7. Cosine Similarity is determined between the train_vector array and the test_vector array and minimum loss is calculated, and the corresponding label is predicted for test videos and stored in a numpy array.
   8. The numpy array of predictions generated is saved to the "Results.csv" file.

3. Input, Output & Results Path
   a. Input Videos Path:
      Training Videos = "traindata"
      Test Videos = "test"
   b. Output Frames Path:
      Training Frames = "frames_train"
      Test Frames = "frames_test"
   c. Prediction Results Path:
      "Results.csv"

4. Gesture Table:

Gesture name	Label
0	0
1	1
2	2

3	3
4	4
5	5
6	6
7	7
8	8
9	9
Decrease Fan Speed	10
FanOn	11
FanOff	12
Increase Fan Speed	13
LightOff	14
LightOn	15
SetThermo	16

Results:

The program compiled successfully in GradeScope with no errors and passed 18/51 test cases which shows a prediction accuracy of 5.2956/15. Keeping in mind the very low amount of available training data, the low prediction accuracy is justified.

Conclusion:

The accuracy may be further improved by increasing the amount of training data significantly and taking into consideration different external factors like hand sizes, ambient brightness levels etc while training