Write Up

Model experiments were carried out on kaggle's GPU.

Gesture Recognition Project :

Problem Statement :

- Build a deep learning model to detect five gestures from videos captured through a smart tv's web camera.
- These gestures are used to control TV functionality
- The five gestures and their corresponding TV controls are :
  1. Thumbs Up — Increase the volume
  2. Thumbs Down — Decrease the volume
  3. Left Swipe — 'Jump' backwards 10 seconds
  4. Stop — Pause the movie.

Dataset :

- The data set contains train and val folders contain 663 , 100 video frames respectively.
- Two CSV files contain the frames and the corresponding class labels.
- Each video is made of 30 frames. Frames come in two sizes — 120×160 and 320×320 as they are recorded from two different sources.

Objectives :

- Develop a deep learning model that is able to classify the gesture based on the video frames.
- The deep learning model should have
  - High accuracy
  - Low memory footprint (to fit in a webcam memory (typically < 50MB))

Solution Approach :

- Two model architectures have been experimented with
  - 3D CNNs
  - 2D CNN + RNN

- Transfer learning using VGGNet and MobileNet have been experimented with
- Finally a retrained MobileNet 2d CNN + GRU network has been submitted with a training accuracy of 99% and validation accuracy of 81%. The parameter count for the same is 4036549. Memory foot print is 48MB on disk.

Experiments Conducted :

Conv 3D :

Objective : What is the maximum batch_size, image dimensions, frames that the GPU could take without a memory overflow

Name	Model Type	Image Size	Frames	batch size	Augmentation	Parameters	Training Accuracy	Validation Accuracy
1	conv3d	160×160	16	30	□	1736389	0.67	0.69
2	conv3d	100×100	30	30	□	687813	0.49	0.56
3	conv3d	100×100	30	60	□	687813	0.57	0.54
4	conv3d	100X100	16	60	□	687813	0.51	0.52
5	conv3d	100X100	16	80	□	687813	0.57	0.50
Untitled					□			

From the above experiments run for 2-3 epochs, we found that image resolution and number of frames in sequence have more impact on training time than batch_size

It was noticed that Conv3D models take a long time to train and filter size and depth of layers needed to be varied to arrive at the optimum model that does not overfit.

Batch normalisation and dropout variation were tried to improve the conv3d models.

CNN + RNN experiments :

Objective

CNN + RNN

Name	Model Type	Augmentation	Pre-trained	Parameters	Train Accuracy	Validation Accuracy	Remarks
MobileNet + GRU	CNN-RNN	□	☑	4036549	0.8959	0.73	
MobileNet + GRU	CNN-RNN	☑	☑	4036549	0.9017	0.72	

Aa Name	☰ Model Type	☑ Augmentation	☑ Pre-trained	# Parameters	# Train Accuracy	# Validation Accuracy	≡ Remarks
VGGNet + GRU(32)+GRU(16)	CNN-RNN	□	☑	15021653	0.81	0	
MobileNet + GRU	CNN-RNN	□	□	4036549	0.99	0.81	

Retrained MobileNet 2d CNN + GRU network has been submitted with a training accuracy of 99% and validation accuracy of 81%. The parameter count for the same is 4036549. Memory foot print is 48MB on disk.

Augmentation & Preprocessing Experiments :

- It was noticed that there is no useful content in 0-20 and 140-160 band of 120×160 images. Hence, they were cropped
- A skin detection filter was also tried but it gave poorer results than the model learning the features themselves.