Gesture Recognition Case Study

Nikhil Sonawane

Shashikant Singh

**Problem Statement:**

A home electronics company wants to develop a cool feature to their smart TV that can recognize five different gestures performed by the user which helps user control the TV without remote. A model which can accurately detect the five gestures has to be developed.

The five gestures are as follows:

* Thumbs up: Increase the volume
* Thumbs down: Decrease the volume
* Left swipe: 'Jump' backwards 10 seconds
* Right swipe: 'Jump' forward 10 seconds
* Stop: Pause the movie

**Methodology:**

Generator function:

Creating a function that generates required batch of sequences (videos) with required number of frames per video. The frames are preprocessed as follows inside the generator function

1. Cropping the image to capture most part of the gesture. There are two types of video sizes 360X360 and 120X160. Cropping has been done differently for each of the video types after inspecting few frames from each of them respectively
2. The images are resized to a common size. Different image sizes like 81X81X3, 80X80X3, 112X112X3 & 120X120X3 were tried to get the best model. For final model the frame size is set to 80X80X3 (same can be seen in the jupyter notebook)
3. After resizing, the images are finally normalized

**Model Building and Training:**

Conv3D and CNN+RNN are the two types of architectures used for video classification tasks. 3D convolutions apply a 3-dimensional filter to the dataset and the filter moves in 3-directions (x, y, z) to calculate the low-level feature representations.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Experiment | Batch size | Image dimension | Model | Train Accuracy | Validation Accuracy | Remarks | Decision |
| 1 | 64 | 80 x 80 | CONV3D | 0.64 | 0.25 | Model is Overfitting | Image size increased |
| 2 | 64 | 120 x 120 | - | - | Memory error | Batch Size decreased |
| 3 | 32 | 120 x 120 | 0.52 | 0.48 | Model is has low accuracy | Image size decreased |
| 4 | 32 | 60 x 60 | 0.67 | 0.66 | Model has decent accuracy |  |
| 5 | 32 | 80X80 | CNN+RNN (LSTM) | 0.68 | 0.59 | Conv3D has better accuracy |  |

**Observations:**

* Image size increases the number of trainable parameters. These parameters can be reduced by changing the filter sizes
* For the same model, the batch size of 32 and 64 had similar run time per epoch
* Due to Out of Memory error batch size was later limited to 32
* Number of epochs can also be experimented to increase the accuracy of models. Due to computational time number of epochs were fixed to 30

**Conclusion :**

CONV3D Model has Accuracy of 67% and Validation Accuracy of 66%