Gesture Recognition – Deep Learning

Problem Statement

As a data scientist at a home electronics company which manufactures state of the art smart televisions. We want to develop a cool feature in the smart-TV that can recognize five different gestures performed by the user which will help users control the TV without using a remote.

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.

Observations

- The resolution of Image, number of frames in sequence and number of layers in the model have more impact on training time
- Batch Size around 15-40 would be ideal
- The resolution can be 160X160, 120X120 depending upon the model performance
- With Image Resolution 160X160 and Number of Frames = 30 we are getting Out of Memory Issue. So we need to crop image further while experimenting with model
- With Image resolution 160X160 and Batch size 20 & 40 we have observed model is overfitting and there is huge gap between Training and Validation Accuracy
- Reducing Image Resolution to 120X120 along with reduction in layers, lowering learning rate to 0.2 has removed the overfitting and we have managed to reduce gap between Training & Validation Accuracy. With this model we can see Validation Loss also reduced to < 1.
- CNN based model has given us good result and we have managed to achieve 92% training and 79% validation accuracy in the model
- For more detailed information on the Observations and Inference, please refer below Tables for more details.

Experiment#	Model	Hyper-parameters	Result	Decision + Explanation

1	Conv3D	image_height=160
		image_width=160
		frames_to_sample=30
		batch_size=15
			Refer to Sample Models. This is giving OOM error.	1. Image Resolution and number of frames in sequence have more impact on training time
				2. Batch Size around 15-40 would be ideal
				3. The resoulution can be 160X160, 128X128, 120X120 depending upon the model performance

2	Conv3D	image_height=160
		image_width=160
		frames_to_sample=20
		batch_size=40
		num_epochs=15
			Model1:
			Training Accuracy = 0.93
			Validation Accuracy = 0.21
				Validation Loss doesn't improved post 1.67941
				Huge Gap between Training & Validation Accuracies
				Model Overfitting
				Add Some Dropouts

3	Conv3D	image_height=160
		image_width=160
		frames_to_sample=20
		batch_size=20
		num_epochs=15
		Dropout = 0.5
			Model2:
			Training Accuracy = 0.82
			Validation Accuracy = 0.22
				Model is overfitting
				Validation Loss doesn't improved post 2.7660
				Lets reduce filter size, image resolution and apply learning rate 0.02

4	Conv3D	image_height=120
		image_width=120
		frames_to_sample=16
		batch_size=30
		num_epochs=25
		Dropout = 0.5
		LR = 0.02
			Model3:
			Training Accuracy = 0.79
			Validation Accuracy = 0.72
			At Epoch = 22
			Training Accuracy = 0.81
			Validation Accuracy = 0.73
				This has reduced overfitting however accuracy still needs to be increased.
				Lets reduce Dropout and number of layers
				
				model-00022-0.49264-0.81222-0.77594-0.73000.h5

5	Conv3D	image_height=120
		image_width=120
		frames_to_sample=16
		batch_size=20
		num_epochs=25
		Dropout = 0.25
		LR = 0.02
			Model4:
			Training Accuracy = 0.92
			Validation Accuracy = 0.75
			At Epoch = 21
			Training Accuracy = 0.92
			Validation Accuracy = 0.79
				This has helped us get more improvement in the model
				Validation Loss further reduced to 0.7048
				Still there is a gap between these training & validation accuracies.
				Lets try reducing dropout
				
				model-00021-0.24294-0.91629-0.61091-0.79000.h5

6	Conv3D	image_height=128
		image_width=128
		frames_to_sample=16
		batch_size=15
		num_epochs=25
		Dropout = 0.2
		LR = 0.01
			Model5:
			Training Accuracy = 0.93
			Validation Accuracy = 0.79
			At Epoch = 24
			Training Accuracy = 0.93
			Validation Accuracy = 0.81
				Validation Loss further reduced to 0.5998
				This is the best model with Conv3D Architecture.
				
				model-00024-0.19574-0.92609-0.57976-0.81000.h5

7	Conv3D + LSTM	image_height=120
		image_width=120
		frames_to_sample=18
		batch_size=15
		num_epochs=40
			Model6:
			Training Accuracy = 0.69
			Validation Accuracy = 0.62
				Though there is no improvement in Val Loss after 0.78924 at Epoch 26 we can see both training and validation accuracy is 66%

		Dropout = 0.2
		LSTM Cells = 128
		LR = 0.01
				Lets try reducing No. frames to 10 and LSTM Cells to 64

8	Conv3D + LSTM	image_height=120
		image_width=120
		frames_to_sample=16
		batch_size=15
		num_epochs=40
		Dropout = 0.2
		LSTM Cells = 64
		LR = 0.01
			Model7:
			Training Accuracy = 0.53
			Validation Accuracy = 0.59
				With Image Resolution 120X120, LSTM Cells 64 we are unable to significant improvement in the Accuracy.
				Lets try reducing no. of frames to 10

9	Conv3D + LSTM	image_height=120
		image_width=120
		frames_to_sample=10
		batch_size=15
		num_epochs=40
		Dropout = 0.2
		LSTM Cells = 64
		LR = 0.01
			Model8:
			Training Accuracy = 0.56
			Validation Accuracy = 0.58
				Though model is not overfitting the accuracy for the model is around ~50%
				Let's try selecting alternate frames from 5-26 since after analyzing first 5 and last 5 frames doesn't have much significant gesture information
				Also increasing learning rate to 0.002

10	Conv3D + LSTM	image_height=120
		image_width=120
		frames_to_sample=11
		batch_size=15
		num_epochs=50
		Dropout = 0.2
		LSTM Cells = 64
		LR = 0.01
			Model9:
			Training Accuracy = 0.46
			Validation Accuracy = 0.54
				Validation Loss is > 1
				Model is having 54% accuracy so it's not a good model.
				Let's try reducing layers and increasing learning rate to 0.02

11	Conv3D	image_height=120
		image_width=120
		frames_to_sample=16
		batch_size=15
		num_epochs=22
		Dropout = 0.2
		LR = 0.01
			Model10:
			Training Accuracy = 0.33
			Validation Accuracy = 0.33
				Model is having 33% accuracy so it's not a good model.

Summary

- After doing all the experiments, we finalized **Model4 – CNN**, which performed well.

  Reasons:

  - Training Accuracy: 92%
  - Validation Accuracy: 79%
  - Number of Parameters 504,709 less compared to other models
  - Weight of the model is 5.89MB which can easily fit in any webcam