

Problem Statement:

Imagine you are working as a data scientist at a home electronics company which manufactures state of the art smart televisions. You 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. The gestures are continuously monitored by the webcam mounted on the TV. Each gesture corresponds to a specific command. Each video is a sequence of 30 frames (or images). 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 We intend to use Conv3D to solve this deep learning problem.

Model Findings:

S.NO	Hyper Parameters	Params Count	Accuracy	Remarks	Action
1	Batch Size: 1 Frames: 10 Image Height: 50 Image Width: 50 Drop out: No Batch Normalization: Yes Layers: 16 > 32 > 64 >> 128 >> 5	Total: 366661 (1.40 MB) Trainable: 366437 (1.40 MB) Non-trainable: 224 (896.00 Byte)	Training: 0.994 Validation: 0.6300	1. Started with batch size 1 to check the model is working 2. Validation accuracy is not bad 3. High difference between Training vs Validation, shows model to be overfitting	1. Increase Batch Size 2. Increase Image Height and Width
2	Batch Size: 32 Frames: 10 Image Height: 100 Image Width: 100 Drop out: No Batch Normalization: Yes Layers: 16 > 32 > 64 >> 128 >> 5	Total: 1251397 (4.77 MB) Trainable: 1251173 (4.77 MB) Non-trainable: 224 (896.00 Byte)	Training: 1.0 Validation: 0.8750	1. Increased batch size + Image size, might have degraded performance 2. Validation Accuracy has increased 3. Training accuracy at 100% and High difference between Training vs Validation, shows model to be overfitting	1. Introduced Dropout with 0.5

3	Batch Size: 32 Frames: 10 Image Height: 100 Image Width: 100 Drop out: Yes, 0.5 Batch Normalization: Yes Layers: 16 > 32 > 64 >> 128 >> 5	Total: 1251397 (4.77 MB) Trainable: 1251173 (4.77 MB) Non-trainable: 224 (896.00 Byte)	Training: 0.6756 Validation: 0.4766	Suffered in both Training and Validation Accuracy	Change architecture starting with 32 layer
4	Batch Size: 32 Frames: 10 Image Height: 100 Image Width: 100 Drop out: Yes, 0.5 Batch Normalization: Yes Layers: 32 > 64 > 128 >> 256 >> 5	Total: 5000325 (19.07 MB) Trainable: 4999877 (19.07 MB) Non-trainable: 448 (1.75 KB)	Training: 0.8793 Validation: 0.7188	1. Training and Validation Accuracy increased 2. High number of trainable params increasing the model size	1. Increase batch size to 64 2. Crop image to 50 x 50
5	Batch Size: 64 Frames: 10 Image Height: 50 Image Width: 50 Drop out: Yes, 0.5 Batch Normalization: Yes Layers: 32 > 64 > 128 >> 256 >> 5	Total: 1461381 (5.57 MB) Trainable: 1460933 (5.57 MB) Non-trainable: 448 (1.75 KB)	Training: 0.7017 Validation: 0.7344	1. Validation accuracy is higher than training accuracy 2. Validation accuracy is not in acceptable range 3. High number of trainable params increasing the model size	NA

Conclusion:

Selected the best model from Model – 4 with,

Training Accuracy: 0.87

Validation Accuracy: 0.72

Batch Size: 32

Frames: 10

Image Height: 100

Image Width: 100

Drop out: Yes, 0.5

Batch Normalization: Yes

Layers: 32 > 64 > 128 >> 256 >> 5

Model File Name: model-00013-0.32161-0.87926-1.11509-0.71875.h5