# **PROLOGUE**

As a part of the Gesture Recognition Project, we have developed 5 deep learning RNN models and tested the results on the test and validation sets. I have used the Upgrad DL instance provided through Jarvis AI for the purpose. Out of the 5 models, the first 4 of them have been incrementally developed with Conv3Dlayers, whereas the 5th one has been designed using a combination of Conv2D and LSTM layers.

# HYPERPARAMETER(S)

Batch Size: 50 (Post experimenting with 10,20,30 sizes, 50 gives max CPU utilization)

Number of Epochs: 50 (We have tried 10, 30 and 50. Validation accuracy needed a substantial no of epochs to catch up with the Training accuracy and therefore we found out with 50 that result is achieved)

Input Dimensions: 120x120 (The idea was to get each image reshaped to a standard square dimension)

Optimizer: Adam

Metric: Accuracy

Loss Function: Categorical Cross-entropy

LEARNING RATE: 0.001

EXPERIMENTS, RESULTS AND MODEL ARCHITECTURES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Architecture** | **Training Result** | **Validation Result** | **Decision + Explanation** |
| **1** | **Conv3D** | Model: "sequential"  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Layer (type) Output Shape Param #  =================================================================  conv3d (Conv3D) (None, 12, 118, 118, 16) 1312    max\_pooling3d (MaxPooling3D (None, 6, 59, 59, 16) 0  )    conv3d\_1 (Conv3D) (None, 4, 57, 57, 32) 13856    max\_pooling3d\_1 (MaxPooling (None, 2, 28, 28, 32) 0  3D)    flatten (Flatten) (None, 50176) 0    dense (Dense) (None, 128) 6422656    dense\_1 (Dense) (None, 5) 645    =================================================================  Total params: 6,438,469  Trainable params: 6,438,469  Non-trainable params: 0  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | loss: 1.6084 - categorical\_accuracy: 0.1886 | val\_loss: 1.6048 - val\_categorical\_accuracy: 0.1500 | **We started off with a very basic architecture as a base model. As we could see the total number of params were around 6.5 million, and it gave poor accuracy of 15% on the validation set** |
| **2** | **Conv3D** | Model: "sequential\_1"  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Layer (type) Output Shape Param #  =================================================================  conv3d\_2 (Conv3D) (None, 12, 118, 118, 32) 2624    conv3d\_3 (Conv3D) (None, 10, 116, 116, 64) 55360    max\_pooling3d\_2 (MaxPooling (None, 5, 58, 58, 64) 0  3D)    batch\_normalization (BatchN (None, 5, 58, 58, 64) 256  ormalization)    conv3d\_4 (Conv3D) (None, 3, 56, 56, 128) 221312    max\_pooling3d\_3 (MaxPooling (None, 3, 28, 28, 128) 0  3D)    batch\_normalization\_1 (Batc (None, 3, 28, 28, 128) 512  hNormalization)    conv3d\_5 (Conv3D) (None, 3, 26, 26, 256) 295168    max\_pooling3d\_4 (MaxPooling (None, 3, 13, 13, 256) 0  3D)    batch\_normalization\_2 (Batc (None, 3, 13, 13, 256) 1024  hNormalization)    conv3d\_6 (Conv3D) (None, 3, 11, 11, 512) 1180160    conv3d\_7 (Conv3D) (None, 3, 9, 9, 512) 2359808    max\_pooling3d\_5 (MaxPooling (None, 3, 4, 4, 512) 0  3D)    batch\_normalization\_3 (Batc (None, 3, 4, 4, 512) 2048  hNormalization)    flatten\_1 (Flatten) (None, 24576) 0    dense\_2 (Dense) (None, 512) 12583424    batch\_normalization\_4 (Batc (None, 512) 2048  hNormalization)    dense\_3 (Dense) (None, 5) 2565    =================================================================  Total params: 16,706,309  Trainable params: 16,703,365  Non-trainable params: 2,944 | loss: 7.0040e-04 - categorical\_accuracy: 1.0000 | val\_loss: 1.2936 - val\_categorical\_accuracy: 0.6700 | **We enhanced the previous model with additional Conv3D layers and added Batch Normalization to standardize the feature maps. This improved the validation accuracy, reduced the loss but caused severe overfitting. Added to that the total number of trainable params shot up to around 16.7 million, with 2944 non-trainable weights** |
| **3** | **Conv3D** | Model: "sequential\_2"  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Layer (type) Output Shape Param #  =================================================================  conv3d\_8 (Conv3D) (None, 12, 118, 118, 32) 2624    conv3d\_9 (Conv3D) (None, 10, 116, 116, 64) 55360    max\_pooling3d\_6 (MaxPooling (None, 5, 58, 58, 64) 0  3D)    batch\_normalization\_5 (Batc (None, 5, 58, 58, 64) 256  hNormalization)    dropout (Dropout) (None, 5, 58, 58, 64) 0    conv3d\_10 (Conv3D) (None, 3, 56, 56, 128) 221312    max\_pooling3d\_7 (MaxPooling (None, 3, 28, 28, 128) 0  3D)    batch\_normalization\_6 (Batc (None, 3, 28, 28, 128) 512  hNormalization)    dropout\_1 (Dropout) (None, 3, 28, 28, 128) 0    conv3d\_11 (Conv3D) (None, 3, 26, 26, 256) 295168    max\_pooling3d\_8 (MaxPooling (None, 3, 13, 13, 256) 0  3D)    batch\_normalization\_7 (Batc (None, 3, 13, 13, 256) 1024  hNormalization)    dropout\_2 (Dropout) (None, 3, 13, 13, 256) 0    flatten\_2 (Flatten) (None, 129792) 0    dense\_4 (Dense) (None, 512) 66454016    batch\_normalization\_8 (Batc (None, 512) 2048  hNormalization)    dense\_5 (Dense) (None, 5) 2565    =================================================================  Total params: 67,034,885  Trainable params: 67,032,965  Non-trainable params: 1,920  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | loss: 0.0018 - categorical\_accuracy: 1.0000 | val\_loss: 1.2397 - val\_categorical\_accuracy: 0.7000 | **In this model, we have removed some consecutive Conv3D layers and added a few dropout layers after each batch normalization from the previous model architecture. This has marginally improved the validation accuracy, but the loss remains practically same. Although the number of non-trainable parameters have reduced to 1920, the trainable parameters have more than quadrupled to 67 million.**  **Also, it is important to mention this model ran into memory overload issues post the 41st epoch and could not be trained further. Therefore we needed to reduce the number of weights as well as the pooling mechanism, to ensure that we get the appropriate accuracy.** |
| **4** | **Conv3D** | Model: "sequential"  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Layer (type) Output Shape Param #  =================================================================  conv3d (Conv3D) (None, 12, 118, 118, 32) 2624    conv3d\_1 (Conv3D) (None, 10, 116, 116, 64) 55360    max\_pooling3d (MaxPooling3D (None, 5, 58, 58, 64) 0  )    batch\_normalization (BatchN (None, 5, 58, 58, 64) 256  ormalization)    dropout (Dropout) (None, 5, 58, 58, 64) 0    conv3d\_2 (Conv3D) (None, 3, 56, 56, 128) 221312    max\_pooling3d\_1 (MaxPooling (None, 3, 28, 28, 128) 0  3D)    batch\_normalization\_1 (Batc (None, 3, 28, 28, 128) 512  hNormalization)    dropout\_1 (Dropout) (None, 3, 28, 28, 128) 0    conv3d\_3 (Conv3D) (None, 3, 26, 26, 256) 295168    max\_pooling3d\_2 (MaxPooling (None, 3, 13, 13, 256) 0  3D)    batch\_normalization\_2 (Batc (None, 3, 13, 13, 256) 1024  hNormalization)    dropout\_2 (Dropout) (None, 3, 13, 13, 256) 0    global\_average\_pooling3d (G (None, 256) 0  lobalAveragePooling3D)    dense (Dense) (None, 512) 131584    batch\_normalization\_3 (Batc (None, 512) 2048  hNormalization)    dense\_1 (Dense) (None, 5) 2565    =================================================================  Total params: 712,453  Trainable params: 710,533  Non-trainable params: 1,920  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  None | loss: 0.0448 categorical\_accuracy: 0.9900 | val\_loss: 0.4734 - val\_categorical\_accuracy: 0.9200 | **Only one major change worked wonders for improving the model. At the entry point of the Dense layer, the Flattening layer was replaced with Global Average Pooling layer. With this, the validation accuracy improved to 92%, and we have drastically reduced the trainable parameters by a factor of 9x to 710k from 67 million. Therefore, using Conv3D we have a good model.** |
| **5** | **Conv2D + LSTM** | Model: "sequential\_9"  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Layer (type) Output Shape Param #  =================================================================  time\_distributed\_5 (TimeDis (None, 14, 118, 118, 8) 224  tributed)    batch\_normalization\_20 (Bat (None, 14, 118, 118, 8) 32  chNormalization)    time\_distributed\_6 (TimeDis (None, 14, 116, 116, 16) 1168  tributed)    batch\_normalization\_21 (Bat (None, 14, 116, 116, 16) 64  chNormalization)    conv\_lstm2d\_1 (ConvLSTM2D) (None, 114, 114, 8) 6944    batch\_normalization\_22 (Bat (None, 114, 114, 8) 32  chNormalization)    time\_distributed\_7 (TimeDis (None, 114, 114, 64) 576  tributed)    batch\_normalization\_23 (Bat (None, 114, 114, 64) 256  chNormalization)    global\_average\_pooling2d (G (None, 64) 0  lobalAveragePooling2D)    dense\_12 (Dense) (None, 64) 4160    dense\_13 (Dense) (None, 5) 325    =================================================================  Total params: 13,781  Trainable params: 13,589  Non-trainable params: 192  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  None | loss: 0.0704 - categorical\_accuracy: 0.9800 | val\_loss: 0.3149 - val\_categorical\_accuracy: 0.9200 | **We wanted to try to reduce the number of trainable parameters, we tried to build a model using Conv2D + LSTM architecture. We could see that we are getting reduced Validation loss and similar validation accuracy of 92%. Additionally, we could see that the number of trainable parameters are only 13589 and only 192 non-trainable params** |
| **Final Model** | **Conv2D + LSTM** | **Model 5 from experiment no 5** |  |  | **As this model has less trainable parameters, reduced loss and same accuracy as Model 4 this will be our final model** |