Gesture Recognition Project

NOTES

1. We have downloaded keras implementation for efficientnets from below location and have provided as part of zip file for the project:

   https://github.com/titu1994/keras-efficientnets

2. We have downloaded keras implementation for AdaBound optimizer from this location and have provided as part of zip file for the project:

   https://github.com/titu1994/keras-adabound

3. We have uploaded all models on google drive to save size of project zip file and provided links to respective models at appropriate places. Please download models from the links provided.

4. For testing/evaluation, please use model 30 (EfficientNet_B0 + GRU).

   Model saved - Size: 52.8 MB –
   (https://drive.google.com/file/d/1yr2SGfnf6ITgVcx4IE6x9CBOWdXAl4-S/view?usp=sharing)

   If needed, further testing/evaluation can be done using next best model also which is model 28 (MobileNet + GRU).

   Model saved – Size: 42.1 MB -
   (https://drive.google.com/file/d/1XuaWXSswKAvO-7U2h5IYTZxxYEUeRBf7/view?usp=sharing)

No of Images

We tried to experiment with different no of images from each folder for each video in input data set like 10/20/30 and found better results with 30, meaning all images in a folder for a video.

Batch Size

We tried with different batch sizes of - 10/20/30/32/64

For some models, batch size of 64 was working fine but with many other models, we were facing GPU memory issue.

For many models, batch size of 20/30/32 was working fine but with some other models, we were facing GPU memory issue (mainly in case of transfer learning models).

For all models, batch size of 10 was working absolutely fine. In fact, training was happening slightly faster with batch size as 10 in comparison to other batch sizes. And we were getting good accuracy with batch size as 10 in comparison to other batch sizes.

So, in order to be consistent across models + achieve slightly faster training + achieve good accuracy, we decided to keep batch size as 10.

Number of Epochs

We tried with different number of epochs - 10/20/30/40

Beyond 20 epochs, not much learning was happening and we were reaching plateau again and again. Not much gain was observed in accuracy also. So, we decided to keep number of epochs as 20.

- For some models, even 10 epochs were enough but we still have kept 20 epochs to keep it consistent and show variations in other models.
- For some models with l2 regularizer, we have trained with 30 epochs just to show that there was not much benefit with more number of epochs.
- For transfer learning models, we have kept number of epochs as 10 as model was already hitting categorical accuracy (> 0.98-0.99).

Number of Models

We have shown below total of 32 models.

- First 3 models, compare Adam, AdaBound and SGD optimizer. We researched about optimizers and found keras implementation of AdaBound and tried to compare its performance.
- Next 6 models on LSTM
- Next 7 models on GRU
- Next 13 models on Conv3D
- Next 3 models on Transfer Learning

Comparison of Models

We have compared models within their families. Like comparison of (Conv2D + LSTM) model is done with another (Conv2D + LSTM) model only, (Conv2D + GRU) model with another (Conv2D + GRU) model, Conv3D model with another Conv3D model and Transfer Learning model with another Transfer Learning model.

However, at the end, we have suggested our best model among all these. We have also noticed that:

- GRU models performed better than LSTM models.
- Conv3D models performed better than GRU models.
- Transfer Learning models outperformed all others.

LSTM Based Models

1 (A) - Model saved – Size: 8.71 MB –
(https://drive.google.com/file/d/1SGra9-Nvx0TInDkhV7Z-37GjuxmPleNO/view?usp=sharing)

1 (B) - Model saved – Size: 8.71 MB –
(https://drive.google.com/file/d/1eR4gbP65ytasPGyn9R8IiwItQn65gdGh/view?usp=sharing)

1 (C) - Model saved – Size: 5.82 MB –
(https://drive.google.com/file/d/1oELfA8J0SwtvcYWr-6jPoigjRM0KOoYy/view?usp=sharing)

2 - Model saved - Size: 8.71 MB –
(https://drive.google.com/file/d/1FVH0V3YUkI4xYv8s5TvL-tkdFllgzZxy/view?usp=sharing)

3 - Model saved - Size: 37.7 MB –
(https://drive.google.com/file/d/1zInJcf6fRrkLjNAT255fUObgxwvZE2oO/view?usp=sharing)

4 - Model saved - Size: 37.9 MB –
(https://drive.google.com/file/d/1AUmLyDvnuXge3Nyk1HNkw9lcqsWGoUNp/view?usp=sharing)

5 - Model saved - Size: 30.6 MB –
(https://drive.google.com/file/d/1iMQsFVTfUeTtoIXjtVYc0nsxV-iFHfc2/view?usp=sharing)

6 - Model saved - Size: 31.4 MB –
(https://drive.google.com/file/d/1IeLO3PnmPcozIYTM_hkUb2G6zr-Zg4JZ/view?usp=sharing)

7 - Model saved - Size: 7.41 MB –
(https://drive.google.com/file/d/1yyYLD8FQCRIiyWMytUX-qkvecO01vzTa/view?usp=sharing)

1 (A)	conv2d_lstm_4cl with Adam as optimizer Conv2D 4 layers (Neurons - 16/32/64/128) + LSTM + Adam (optimizer) + Filter (3,3) + Strides (2,2)	- 18th Epoch (Best Model) - Loss - 0.1582 - Categorical Accuracy - 0.9682 - Val Loss - 0.6497 - Val Categorical Accuracy - 0.7300	Bad model. Loss is bit high and validation accuracy is low. Better than 1 (B) and 1 (C) so Adam optimizer is used by default for rest of models.
1 (B)	conv2d_lstm_4cl with AdaBound as optimizer Conv2D 4 layers (Neurons - 16/32/64/128) + LSTM + AdaBound (optimizer) + Filter (3,3) + Strides (2,2)	- 18th Epoch - Loss - 0.3664 - Categorical Accuracy - 0.9070 - Val Loss - 0.7163 - Val Categorical Accuracy - 0.6700	Bad model. Loss is bit high and validation accuracy is low.

1 (C)	conv2d_lstm_4cl with SGD as optimizer Conv2D 4 layers (Neurons - 16/32/64/128) + LSTM + SGD (optimizer) + Filter (3,3) + Strides (2,2)	- 20th Epoch - Loss - 0.1711 - Categorical Accuracy - 0.9731 - Val Loss - 0.6679 - Val Categorical Accuracy - 0.7400	Bad model. Loss is bit high and validation accuracy is low.
2	conv2d_lstm_4cl_dropouts Conv2D 4 layers (Neurons - 16/32/64/128) + Dropouts + LSTM + Filter (3,3) + Strides (2,2)	- 20th Epoch - Loss - 0.3163 - Categorical Accuracy - 0.8921 - Val Loss - 0.6778 - Val Categorical Accuracy - 0.7400	Bad model. Loss is bit high and validation accuracy is low.
3	conv2d_lstm_4cl_filter Conv2D 4 layers (Neurons - 16/32/64/128) + LSTM + Filter (1,1)	- 17th Epoch - Loss - 0.1688 - Categorical Accuracy - 0.9637 - Val Loss - 0.5379 - Val Categorical Accuracy - 0.7900	Not a good model in comparison to model 5. Loss is bit high and validation accuracy is low.
4	conv2d_lstm_4cl_filter_l2_regularizer Conv2D 4 layers (Neurons - 16/32/64/128) + LSTM + Filter (1,1) + L2 Regularizer	- 20th Epoch - Loss - 1.1496 - Categorical Accuracy - 0.7428 - Val Loss - 1.3266 - Val Categorical Accuracy - 0.6700	Bad model. Loss is high and accuracy is low.
5	**conv2d_lstm_5cl_filter** **Conv2D** **5 layers** **(Neurons - 16/32/64/128/256)** **+ LSTM** **+ Filter (1,1)**	**- 13th Epoch** **- Loss - 0.1200** **- Categorical Accuracy - 0.9806** **- Val Loss - 0.4473** **- Val Categorical Accuracy - 0.8600**	**Quite a good model. Loss is low and accuracy is high. Validation accuracy is also comparable.**
6	conv2d_lstm_5cl_filter_l2_regularizer - 30 epochs Conv2D 5 layers (Neurons - 16/32/64/128/256) + LSTM + Filter (1,1) + L2 Regularizer	- 20th Epoch - Loss - 0.9676 - Categorical Accuracy - 0.9025 - Val Loss - 1.1560 - Val Categorical Accuracy - 0.8200	Not a good model in comparison to model 5. Loss is high. Please notice that after 20 epochs, there was no improvement in validation loss as explained earlier.
7	conv2d_lstm_6cl_filter Conv2D 6 layers (Neurons - 16/32/64/128/256/256) + LSTM + Filter (1,1)	- 11th Epoch - Loss - 0.5187 - Categorical Accuracy - 0.8149 - Val Loss - 0.6837 - Val Categorical Accuracy - 0.7800	Not a good model in comparison to model 5. Loss is bit high and accuracy is bit low.

GRU Based Models

8 - Model saved - Size: 6.83 MB –
(https://drive.google.com/file/d/1CtAze2s8SKZ4XZ21_2J_jEzv3cOLA0ir/view?usp=sharing)

9 - Model saved - Size: 6.84 MB –
(https://drive.google.com/file/d/1fUZ9BD0YiyFavvP8gNgnoDcyKaLos5Pb/view?usp=sharing)

10 - Model saved - Size: 28.3 MB –
(https://drive.google.com/file/d/1-BIUC0GYe0E6y4aXHFDO3QfkPtUaWx7R/view?usp=sharing)

11 - Model saved - Size: 28.5 MB –
(https://drive.google.com/file/d/1NILyALTP9U0xtC8C1I7uhsh8Hu4gXOt5/view?usp=sharing)

12 - Model saved - Size: 23.1 MB –
(https://drive.google.com/file/d/1aqo2mkTjvMxOWEJfto3DwfnAGbq8BRzS/view?usp=sharing)

13 - Model saved - Size: 23.9 MB –
(https://drive.google.com/file/d/10FDSn2iX0J_509hZxhpp9wzKXUi6MHSr/view?usp=sharing)

14 - Model saved - Size: 5.91 MB –
(https://drive.google.com/file/d/1QsBBfEXg2XSFvfAyiwHe_t7C32VBOFc0/view?usp=sharing)

8	conv2d_gru_4cl Conv2D 4 layers (Neurons - 16/32/64/128) + GRU + Filter (3,3) + Strides (2,2)	- 7th Epoch - Loss - 0.2087 - Categorical Accuracy - 0.9369 - Val Loss - 0.6504 - Val Categorical Accuracy - 0.7400	Bad model. Loss is bit high and validation accuracy is low.
9	conv2d_gru_4cl_dropouts Conv2D 4 layers (Neurons - 16/32/64/128) + Dropouts + GRU + Filter (3,3) + Strides (2,2)	- 4th Epoch - Loss - 0.9201 - Categorical Accuracy - 0.6438 - Val Loss - 0.8932 - Val Categorical Accuracy - 0.6900	Bad model. Loss is high and accuracy is low.
10	**conv2d_gru_4cl_filter** **Conv2D** **4 layers (Neurons - 16/32/64/128)** **+ GRU** **+ Filter (1,1)**	**- 18th Epoch** **- Loss - 0.1479** **- Categorical Accuracy - 0.9622** **- Val Loss - 0.4118** **- Val Categorical Accuracy - 0.8300**	**Model is good. Loss is low. Accuracy is good and validation accuracy is comparable.**

11	conv2d_gru_4cl_filter_l2_regularizer Conv2D 4 layers (Neurons - 16/32/64/128) + GRU + Filter (1,1) + L2 Regularizer	- 20th Epoch - Loss - 1.1798 - Categorical Accuracy - 0.7797 - Val Loss - 1.1083 - Val Categorical Accuracy - 0.8200	Not a good model in comparison to model 12. Loss is high and accuracy is low.
12	**conv2d_gru_5cl_filter** **Conv2D** **5 layers** **(Neurons - 16/32/64/128/256)** **+ GRU** **+ Filter (1,1)**	**- 19th Epoch** **- Loss - 0.0627** **- Categorical Accuracy - 0.9881** **- Val Loss - 0.3441** **- Val Categorical Accuracy - 0.81**	**Quite a good model. Loss is low and accuracy is high. Validation accuracy is relatively ok (above 80%).**
13	conv2d_gru_5cl_filter_l2_regularizer - 30 epochs Conv2D 5 layers (Neurons - 16/32/64/128/256) + GRU + Filter (1,1) + L2 Regularizer	- 30th Epoch - Loss - 1.0370 - Categorical Accuracy - 0.9006 - Val Loss - 1.2440 - Val Categorical Accuracy - 0.8200	Not a good model in comparison to model 12. Loss is high. Please notice that there is not much improvement between 20th to 30th Epoch as explained earlier.
14	conv2d_gru_6cl_filter Conv2D 6 layers (Neurons - 16/32/64/128/256/256) + GRU + Filter (1,1)	- 15th Epoch - Loss - 0.2263 - Categorical Accuracy - 0.9507 - Val Loss - 0.6850 - Val Categorical Accuracy - 0.7300	Not a good model in comparison to model 12. Loss is bit high and validation accuracy is low.

Conv3D Based Models

15 - Model saved - Size: 12.8 MB – (https://drive.google.com/file/d/1xhyHjHmueMXFhpLeucrFH2n2n96Q3m-y/view?usp=sharing)

16 - Model saved - Size: 12.8 MB – (https://drive.google.com/file/d/1MA-antE0PomzNg9_iurKaNPUP7yuecr3/view?usp=sharing)

17 - Model saved - Size: 12.8 MB – (https://drive.google.com/file/d/1ilSs7rsh6BhlI4t3n_M6MJPYpE_xX2la/view?usp=sharing)

18 - Model saved - Size: 9.62 MB – (https://drive.google.com/file/d/19nAdNrOH2oPBKWBZyQaZKacui8R32XlY/view?usp=sharing)

19 - Model saved - Size: 9.62 MB – (https://drive.google.com/file/d/1h5Egs0I9SKb8eLnBVZVws4ODEK5zp2-u/view?usp=sharing)

20 - Model saved - Size: 51.1 MB – (https://drive.google.com/file/d/1heTY6iBuI4caMb8T1-ugW7OqAvMhFsHm/view?usp=sharing)

21 - Model saved - Size: 51.1 MB –
(https://drive.google.com/file/d/14cw0dHSToF0pt-aWhafWX6xR5Qv48Z_3/view?usp=sharing)

22 - Model saved - Size: 51.1 MB –
(https://drive.google.com/file/d/1xcR7ACK4fUi0KSZya1XxCSzsMLdxvZzy/view?usp=sharing)

23 - Model saved - Size: 38.1 MB –
(https://drive.google.com/file/d/1CQqEAJHs4IpxANIoOhSUJCXh-A1lbyUP/view?usp=sharing)

24 - Model saved - Size: 38.1 MB –
(https://drive.google.com/file/d/1dgUs1vdpLGlMbXOuIT0mt2OnPEdGU_AY/view?usp=sharing)

25 - Model saved - Size: 38.9 MB –
(https://drive.google.com/file/d/1u6OTae6WLXRQRvLwDIhBSzE3MVlTarNa/view?usp=sharing)

26 - Model saved - Size: 38.9 MB –
(https://drive.google.com/file/d/1vs_iJvl1Qbk2Vq0LrZeiAhksV5fnYroz/view?usp=sharing)

27 - Model saved - Size: 39.7 MB –
(https://drive.google.com/file/d/1TJY6NX--Scb6d64vxNrQtQbhp3zmlpYq/view?usp=sharing)

15	conv3d_4cl Conv3D 4 layers (Neurons - 16/32/64/128) + Filter (3,3,3)	- 17th Epoch - Loss - 0.4414 - Categorical Accuracy - 0.8313 - Val Loss - 0.5673 - Val Categorical Accuracy - 0.8000	Not a good model in comparison to model 23 and 25. Loss is bit high and accuracy is bit low.
16	conv3d_4cl_dropouts Conv3D 4 layers (Neurons - 16/32/64/128) + Dropouts + Filter (3,3,3)	- 3rd Epoch - Loss - 1.2867 - Categorical Accuracy - 0.5099 - Val Loss - 1.2558 - Val Categorical Accuracy - 0.5000	Bad model. Loss is high and accuracy is low.
17	conv3d_4cl_l2_regularizer Conv3D 4 layers (Neurons - 16/32/64/128) + Filter (3,3,3) + L2 Regularizer	- 19th Epoch - Loss - 2.4142 - Categorical Accuracy - 0.8548 - Val Loss - 2.4841 - Val Categorical Accuracy - 0.8700	Not a good model in comparison to model 23 and 25. Loss is high.
18	conv3d_4cl_filter Conv3D 4 layers (Neurons - 16/32/64/128) + Filter (1,1,1)	- 6th Epoch - Loss - 0.5503 - Categorical Accuracy - 0.7916 - Val Loss - 0.4442 - Val Categorical Accuracy - 0.8400	Bad model. Loss is bit high and accuracy is low.

19	conv3d_4cl_filter_l2_regularizer Conv3D 4 layers (Neurons - 16/32/64/128) + Filter (1,1,1) + L2 Regularizer	- 19th Epoch - Loss - 1.5189 - Categorical Accuracy - 0.8746 - Val Loss - 1.6352 - Val Categorical Accuracy - 0.7600	Bad model. Loss is high and validation accuracy is low.
20	conv3d_5cl Conv3D 5 layers (Neurons - 16/32/64/128/256) + Filter (3,3,3)	- 14th Epoch - Loss - 0.6580 - Categorical Accuracy - 0.7438 - Val Loss - 0.7985 - Val Categorical Accuracy - 0.6600	Bad model. Loss is high and accuracy is low.
21	conv3d_5cl_dropouts Conv3D 5 layers (Neurons - 16/32/64/128/256) + Dropouts + Filter (3,3,3)	- 11th Epoch - Loss - 0.6189 - Categorical Accuracy - 0.7716 - Val Loss - 1.3617 - Val Categorical Accuracy - 0.4200	Bad model. Loss is high and accuracy is low.
22	conv3d_5cl_l2_regularizer Conv3D 5 layers (Neurons - 16/32/64/128/256) + Filter (3,3,3) + L2 Regularizer	- 19th Epoch - Loss - 2.2079 - Categorical Accuracy - 0.8742 - Val Loss - 2.3546 - Val Categorical Accuracy - 0.8100	Not a good model in comparison to model 23 and 25. Loss is high.
23	**conv3d_5cl_filter** **Conv3D** **5 layers** **(Neurons - 16/32/64/128/256)** **+ Filter (1,1,1)**	**- 8th Epoch** **- Loss - 0.2612** **- Categorical Accuracy - 0.9181** **- Val Loss - 0.3557** **- Val Categorical Accuracy - 0.9100**	**A very good model. Low loss and very high accuracy. Validation accuracy is almost same.** **Though best model was at 8th epoch, loss and accuracies kept oscillating in rest of epochs but did not go worse.**
24	conv3d_5cl_filter_l2_regularizer Conv3D 5 layers (Neurons - 16/32/64/128/256) + Filter (1,1,1) + L2 Regularizer	- 20th Epoch - Loss - 2.3258 - Categorical Accuracy - 0.9627 - Val Loss - 2.5665 - Val Categorical Accuracy - 0.8500	Not a good model in comparison to model 23 and 25. Loss is high.
25	**conv3d_6cl_filter** **Conv3D** **6 layers** **(Neurons - 16/32/64/128/256/256)** **+ Filter (1,1,1)**	**- 15th Epoch** **- Loss - 0.1391** **- Categorical Accuracy - 0.9612** **- Val Loss - 0.4143** **- Val Categorical Accuracy - 0.8600**	**Quite a good model. Low loss and very high accuracy. Validation accuracy is also quite good, within 10% variation.**

26	conv3d_6cl_filter_l2_regularizer Conv3D 6 layers (Neurons - 16/32/64/128/256/256) + Filter (1,1,1) + L2 Regularizer	- 20th Epoch - Loss - 2.7496 - Categorical Accuracy - 0.9627 - Val Loss - 2.9634 - Val Categorical Accuracy - 0.8500	Not a good model in comparison to model 23 and 25. Loss is high.
27	**conv3d_7cl_filter** **Conv3D** **7 layers** **(Neurons - 16/32/64/128/256/256/256)** **+ Filter (1,1,1)**	**- 7th Epoch** **- Loss - 0.2789** **- Categorical Accuracy - 0.9055** **- Val Loss - 0.5064** **- Val Categorical Accuracy - 0.8200**	**Model is good. Loss is low. Accuracy is good and validation accuracy is also good, within 10% variation.** **Though best model was at 7th epoch, loss and accuracies kept oscillating in rest of epochs but did not go worse.**

Transfer Learning Based Models

28 - Model saved – Size: 42.1 MB – (https://drive.google.com/file/d/1XuaWXSswKAvO-7U2h5IYTZxxYEUeRBf7/view?usp=sharing)

29 - Model saved – Size: 49.1 MB – (https://drive.google.com/file/d/1zrAvP8Ee-Qy-HKDwA6-3xIGgRgVz9LTE/view?usp=sharing)

30 - Model saved - Size: 52.8 MB – (https://drive.google.com/file/d/1yr2SGfnf6ITgVcx4IE6x9CBOWdXAl4-S/view?usp=sharing)

28	**mobilenet_transfer_learning_gru** **- 10 epochs** **MobileNet** **+ GRU**	**- 10th Epoch** **- Loss - 0.0295** **- Categorical Accuracy - 0.9861** **- Val Loss - 0.1590** **- Val Categorical Accuracy - 0.9600**	**A very good model. Low loss and high accuracy. Validation accuracy is almost same.**
29	mobilenet_v2_transfer_learning_gru - 10 epochs MobileNet V2 + GRU	- 10th Epoch - Loss - 0.1222 - Categorical Accuracy - 0.9627 - Val Loss - 0.5233 - Val Categorical Accuracy - 0.7800	Not a good model in comparison to model 28. Validation accuracy is low.
30	**efficientnet_b0_transfer_learning_gru** **- 10 epochs** **EfficientNet B0** **+ GRU**	**- 7th Epoch** **- Loss - 0.0269** **- Categorical Accuracy - 0.9936** **- Val Loss - 0.0594** **- Val Categorical Accuracy - 0.9800**	**Best model so far. Lowest loss and highest accuracy. Validation accuracy is almost same.**

Conclusion:

In this project, we prepared 3 types of models involving Conv2D + RNN (LSTM / GRU), Conv3D and Transfer Learning.

- In case of Conv2D + LSTM, best model is model 5 (size 30.6 MB)
- In case of Conv2D + GRU, best model is model 12 (size 23.1 MB)
- In case of Conv3D, best model is model 23 (size 38.1 MB)
- In case of Transfer learning, best models are model 28 (size 42.1 MB) as well as model 30 (size 52.8 MB).

Best Models: Transfer Learning Architecture

- If 52.8 MB size is manageable in memory of webcam then final chosen model is model 30. Its training time is bit higher (but manageable) than model 28 but once trained, it has lowest loss and highest accuracy.
- If model 30 is not acceptable due to memory size of webcam then model 28 is next best. Its training time is low. It has second lowest loss and second highest accuracy.
- Both model 28 and 30 from transfer learning architecture, are very good.

Best Model: Conv3D Architecture

If transfer learning models are not acceptable and memory is still an issue then model 23 is next best. It has both accuracies almost equal (~0.91). Though best model was at 8th epoch, loss and accuracies kept oscillating in rest of epochs but did not go worse. We have explained earlier as well, not all models needed to be trained for 20 epochs. We have just kept it 20 to keep it consistent and show variations in other models.

Best Model: Conv2D + GRU Architecture

If size of the model, is still an issue due to memory size of webcam then model 12 is next best. It has low loss and high accuracy. Validation accuracy is relatively ok (above 80%).

Best Model: Conv2D + LSTM Architecture

If someone wants to use LSTM then model 5 is best among LSTM models. It has low loss and high accuracy. Validation accuracy is also comparable.

Our Suggestion:

Model 30 (EfficientNet_B0 + GRU) followed by Model 28 (MobileNet + GRU)