Gesture Recognition case study

We have started with data generator function. we have used two types of data generator. One without augmentation and other is with augmentation.

We have tried 6 types of models with the below changes:

- model based on Conv3d
- Conv2D+GRU
- Transfer Learning using VGGNET-19, Mobilenet, ImageNet.

We have use Hyper-tuning parameter like:

- batch size
- optimizer
- no of frame to sample
- learning rate
- Dropouts
- Batch normalization
- add a greater number of dense layers
- Activation function to achieve best model.

We have use affine type of augmentation inside in generator function.

Below is the table where we have captured data with multiple models.

Experiment Number	Model	Training Accuracy (in %)	Val Accuracy (in %)	Decision + Explanation
1	**Conv3D**	**21.57**	**15**	**Tried with simple Conv3D model, with** • **Batch size 20.** • **SGD optimizer** • **Learning rate is 0.01** • **Epoch = 35**
2	**Conv3D**	**46.06**	**67**	**Change Conv3D model, with below parameter** • **Batch size 20.** • **Adam optimizer** • **Learning rate is 0.001** • **Epoch = 25**
3	**Conv2DVGGNettransfer Learning**	**95.48**	**80**	**As with the above two results are not good, so we decided to change model. The parameter is** • **Epoch = 35** • **SGD optimizer**

				• **Learning rate is 0.01**
4	**Conv2DMobilenettransfer Learning +GRU**	**98.79**	**84**	**As with the VggNet the gap between Training and validation accuracy is approx. 16%, hence we change our model to mobile net with augmentation and found results are slightly better. The parameter is** • **Epoch = 20** • **Batch size = 5** • **Adam optimizer** • **Learning rate is 0.01** • **Affine Augmentation**
5	**Conv2DMobilenettransfer Learning + GRU but training all layer**	**99.77**	**99**	**The parameter is** • **Epoch = 20** • **Batch Size =5** • **Frame to sample = 16** • **Adam optimizer** • **Learning rate is 0.01** • **Affine Augmentation** • **Dropout = 0.25** **But looks like overfitting.**
Final Model	**Conv2DMobilenettransfer Learning + GRU but training all layer**	**93.97**	**80**	**The parameter is** • **Epoch = 25** • **Batch Size =20** • **Frame to sample = 15** • **Adam optimizer** • **Learning rate is 0.00001** • **Affine Augmentation** • **Dropout = 0.15** **This is our final model as, this model looks good, as per result of Training Accuracy and validation Accuracy.**