ConvLSTM gives better results than Conv3D. The explanation should be as detailed as possible so that the logic behind the decision is conveyed. Also, there are a lot of things you can experiment with in the generator function and elsewhere. Please do not forget to specify the exact metric values, here Accuracy which drives your decision.

You can draw inspiration from the concepts taught in the Industry demo in CNNs to experiment with the data and different architectures.

Note: all the experiments are carried out and present in Assignment_notebook_all_models.ipynb

The final model – model 13 with the generator + model.h5 file is present in Assignment_final_model.ipynb

Experiment Number	Model	Result	Decision + Explanation
1	Conv3D [simple convolution model layers, with batch size as 100]	Training accuracy: 87 Validation accuracy:47 overfitting	Next step is that we can add dropout and batch normalization layers.
2	Conv3D [simple convolution model layers, with batch size as 100+ dropout and batch normalization]	Training accuracy: 98 Validation accuracy:23 Overfitting	Reduce the batch size to see if the accuracies converge
3	Conv3D with batch size as 64	Training accuracy: 95 Validation accuracy:32 Overfitting	Let's try with a different optimizer- Optimizer : SGD
4	Conv3D+ batch size 64+dropout+SGD	Training accuracy: 91 Validation accuracy:20 Overfitting	Let's change the image resize and cropping to 160 x 160
5	CONV3D+ image size 160x160	Training accuracy: 86 Validation accuracy:29 Overfitting	Let's reduce the batch size to 32 and try by adding L2 regularization in the neural network layers to reduce overfitting
6	Conv3D+ 32 batch_size+ l2 regularization	Training accuracy: 80 Validation accuracy:38	Let's try convlstm as not able to get desired accuracy with conv3d

		Overfitting	
7	**Time distributed+GRU+ 64 batch size**	**OOM error**	**Retry with 32 batch size and 120x120 image size**
8	**Time distributed+GRU+ 32 batch size**	**Training accuracy: 76 Validation accuracy:25 Overfitting**	**Lets try data augmentation to see if it improves**
9	**Time distributed+GRU+ 32 batch size+ data augmentation**	**Training accuracy: 35 Validation accuracy:31 Underfitting**	**Lets add more GRU layers and reduce the dropout percentage, to make the model learn the complex patterns.**
10	**Time distributed+GRU+ 32 batch size+ data augmentation+ GRU layers**	**Training accuracy: 43 Validation accuracy:23 Underfitting**	**Let's add an LSTM layer of lstm cells and check if the accuracy improves**
11	**Time distributed+GRU+ 32 batch size+ data augmentation+ LSTM layers**	**Training accuracy: 39 Validation accuracy:30 Underfitting**	**Analysing the images post processing with the generator, found out that the crop and flip data augmentations were not useful, removing these and retrying**
12	**Time distributed+GRU+ 32 batch size+ data augmentation+ LSTM layers + only resizing the images**	**Training accuracy: 95 Validation accuracy:85 Overfitting**	**Adding additional L2 regularization**
13	**Time distributed+ 32 batch size+ data augmentation+ LSTM layers + only resizing the images + L2 regularization**	**Training accuracy: 95 Validation accuracy:71 Overfitting**	**Observed reduced validation loss, lastly trying with reduced network layer** **[THIS IS THE BEST MODEL SO FAR]**
14	**Model 13 and removing a neural network layer with 256 neurons**	**Training accuracy: 95 Validation accuracy:71 Overfitting**	