Following is the summary of experiments and observations for the project.

* Note that dropouts (usually it was set to 0.25 in dense connections) and batch normalization play an important role for regularization but not explicitly mentioned them in the model column.
* I used batch size of 10 for all the models as it was not giving any memory issues and models were training well with it.
* I experimented with either 20 images OR 30 images per video.
* Experimented with either 120 by 120 image size OR 160 by 160 image size.
* Final models are highlighted in yellow along with their accuracy and loss numbers.
* Data transformations/augmentation didn’t seem to be helping much in improving accuracy or loss so avoided them in the final constructive modeling.

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv 3D Model using 30 frames per video**  **+ 16, 32, 64, 128 filters conv 3D layers each followed by max pool 3D**  **+ 256 dense nodes**  **+ 128 dense nodes**  **+ image size 120 by 120** | **Train Loss: 0.1924**  **Train Accuracy: 93.9%**  **Val Loss: 0.9976**  **Val Accuracy: 72%** | **Low validation accuracy as compared to training accuracy.** |
| **2** | **Conv 3D Model using 20 frames per video**  **+ 16, 32, 64, 128 filters conv 3D layers**  **+ 256 dense nodes**  **+ 128 dense nodes**  **+ image size 120 by 120** | **Train Loss: 0.2441**  **Train Accuracy: 92.16%**  **Val Loss: 0.5798**  **Val Accuracy: 77%** | **Low loss and high accuracy and comparable between training and validation data sets.**  **Parameters - ~1.9 million.** |
| **3** | **Conv 3D Model with 30 frames per video**  **+ 16, 32, 64, 128 filters conv 3D layers**  **+ 256 dense nodes**  **+ 128 dense nodes**  **+ image size 160 by 160** | **Train Loss: 0.2440**  **Train Accuracy: 91.4%**  **Val Loss: 0.5858**  **Val Accuracy: 80%** | **Parameters are on higher side and even the model accuracy is not good enough.**  **Parameters-~3 million** |
| **4** | **Conv 3D Model with 20 frames per video**  **+ 16, 32, 64, 128 filters conv 3D layers**  **+ 256 dense nodes**  **+ 128 dense nodes**  **+ image size 160 by 160** | **Train Loss: 0.2498**  **Train Accuracy: 90.95%**  **Val Loss: 0.5493**  **Val Accuracy: 82%** | **120 by 120 image size is better for training than 160 by 160 image size.** |
| **5** | **Conv 3D Model with 30 frames per video**  **+ 16, 32, 64, 128 filters conv 3D layers**  **+ 256 dense nodes**  **+ 128 dense nodes**  **+ Random data transformations on training data set** | **Train Loss: 0.6699**  **Train Accuracy: 74.81%**  **Val Loss: 1.3239**  **Val Accuracy: 55%** | **Accuracy dropped with addition of random data transformation significantly.** |
| **6** | **Conv 3D Model with 30 frames per video**  **+ 8, 16, 32, 64 filters conv 3D layers**  **+ 256 dense nodes**  **+ 128 dense nodes**  **+ image size 120 by 120** | **Train Loss: 0.1649**  **Train Accuracy: 94.72%**  **Val Loss: 0.5062**  **Val Accuracy: 80%** | **Parameters are on lower side ~900K**  **But slightly low validation accuracy.** |
| **7\*** | **Conv 3D Model with 20 frames per video**  **+ 16, 32, 64, 128 filters conv 3D layers**  **+ 128 dense nodes**  **+ 64 dense nodes**  **+ image size 120 by 120** | **Train Loss: 0.2618**  **Train Accuracy: 91.4%**  **Val Loss: 0.4813**  **Val Accuracy: 86%** | **Difference between train loss and validation loss is less and both follow similar trend** |
| **8** | **Conv 3D Model with 20 frames per video**  **+ 32, 64, 128, 256 filters conv 3D layers**  **+ 128 dense nodes**  **+ 128 dense nodes**  **+ image size 120 by 120** | **Train Loss: 0.3797**  **Train Accuracy: 86.27%**  **Val Loss: 0.7064**  **Val Accuracy: 68%** | **Increasing number of filters increased the number of parameters to ~2.7 million and validation accuracy is lower only.** |
| **9** | **Conv 3D Model with 30 frames per video**  **+ 5 by 5 filter size (16 conv 3d filters layer)**  **+ 3 by 3 filter size (32, 64, 128) conv 3D layers**  **+ 256 dense nodes**  **+ 128 dense nodes**  **+ image size 120 by 120** | **Train Loss: 0.5305**  **Train Accuracy: 80.09%**  **Val Loss: 0.9248**  **Val Accuracy: 62%** | **Model seems to be fitting pretty well but slightly low accuracy.**  **We trained for 15 epochs due to time constraint.**  **We will train it for 25 epochs and check again.** |
| **10** | **Mobilenet (retrain all weights)**  **+ GRU (128 cells)**  **+ Dense (128 nodes)**  **+ image size 120 by 120**  **+ 20 images per video** | **Train Loss: 0.0131**  **Train Accuracy: 99.7%**  **Val Loss: 0.1347**  **Val Accuracy: 97%** | **Retrained all the weights of Mobilenet.**  **Batch size = 10**  **Epochs = 20** |
| **11\*** | **Mobilenet (retrain all weights)**  **+ GRU (128 cells)**  **+ Dense (128 nodes)**  **+ image size 120 by 120**  **+ 30 images per video**  **+ random data transformations on the images** | **Train Loss: 0.0264**  **Train Accuracy: 99.40%**  **Val Loss: 0.0620**  **Val Accuracy: 98%** | **Retrained all the weights of Mobilenet.**  **Batch size = 10**  **Epochs = 20** |
| **12** | **Mobilenet (fine tune after 50th layer)**  **+ GRU (128 cells)**  **+ Dense (128 nodes)**  **+ image size 120 by 120**  **+ 20 images per video** | **Train Loss: 0.0102**  **Train Accuracy: 99.85%**  **Val Loss: 0.2673**  **Val Accuracy: 93%** | **Batch size = 10**  **Epochs = 20**  **Fine tuning not giving good validation accuracy.**  **Hence always opting to retrain the full network.** |
| **13** | **Mobilenet (retrain all weights)**  **+ LSTM (128 cells)**  **+ Dense (128 nodes)**  **+ image size 120 by 120**  **+ 20 images per video** | **Train Loss: 0.0389**  **Train Accuracy: 99.4%**  **Val Loss: 0.1949**  **Val Accuracy: 93%** | **Retrained all weights.**  **Batch size = 10**  **Epochs = 20** |
| **14** | **Mobilenet (retrain all weights)**  **+ GRU (128 cells)**  **+ Dense (64 nodes)**  **+ image size 120 by 120**  **+ 20 images per video** | **Train Loss: 0.0213**  **Train Accuracy: 99.4%**  **Val Loss: 0.0991**  **Val Accuracy: 98%** | **Retrained all weights.**  **Batch size = 10**  **Epochs = 20** |
| **15** | **Mobilenet (retrain all weights)**  **+ LSTM (128 cells)**  **+ Dense (64 nodes)**  **+ image size 120 by 120**  **+ 20 images per video**  **+ random data transformations on the images** | **Train Loss: 0.0422**  **Train Accuracy: 98.64%**  **Val Loss: 0.1888**  **Val Accuracy: 94%** | **Retrained all weights.**  **Batch size = 10**  **Epochs = 20** |
|  |  |  |  |
|  |  |  |  |
| **Final Models** | **Conv 3D Model with 20 frames per video**  **+ 16, 32, 64, 128 filters conv 3D layers**  **+ 128 dense nodes**  **+ 64 dense nodes**  **+ image size 120 by 120** | **Train Loss: 0.2618**  **Train Accuracy: 91.4%**  **Val Loss: 0.4813**  **Val Accuracy: 86%** | **Difference between train loss and validation loss is less and both follow similar trend** |
|  | **Mobilenet (retrain all weights)**  **+ GRU (128 cells)**  **+ Dense (128 nodes)**  **+ image size 120 by 120**  **+ 30 images per video**  **+ random data transformations on the images** | **Train Loss: 0.0264**  **Train Accuracy: 99.40%**  **Val Loss: 0.0620**  **Val Accuracy: 98%** | **Retrained all the weights of Mobilenet.**  **Batch size = 10**  **Epochs = 20** |