|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D** | **Resource Exhausted Error** | **Decreased the batch size to fit into the memory.** |
| **2** | **Conv3D** | **Resource Exhausted Error** | **Resized the image from 100 to 90 and 80. Also reduced the layers.** |
| **3** | **Conv3D** | **Accuracy: 0.16** | **Decreased learning rate and increased momentum** |
| **4** | **Conv3D** | **Accuracy: 0.16** | **Increase the depth of the network and changed initial number of kernels** |
| **5** | **Conv3D** | **Accuracy: 0.51** | **Architecture looks good as it gives good accuracy on train and val. Let’s try tweaking few hyperparameters to further improve the accuracy** |
| **6** | **Conv3D** | **Accuracy: 0. 71** | **Let’s try ConvLSTM and see how well it performs than Conv3D** |
| **7** | **ConvLSTM** | **Accuracy: 0.28** | **Need to look into the architecture** |
| **8** | **ConvLSTM** | **Accuracy: 0.62** | **Changed architecture and learning rate. Model started to overfit. Let’s introduce dropouts to tackle this.** |
| **9** | **ConvLSTM** | **Accuracy: 0.70** | **Overfitting again. Let’s try changing VGG16 architecture to mobilenet. And instead of LSTM let’s use** |
| **10** | **ConvGRU** | **Accuracy: 0.66** | **Overfitting. Increasing the dropout rate** |
| **11** | **ConvGRU** | **Accuracy: 0.63** | **Optimized** |
|  |  |  |  |
| **Final Model** | **Conv3D** | **Accuracy: 0. 71** | **Very Simple yet powerful model than others.** |
|  |  |  |  |