Our strategy for this project was to create several models using two different types of architectures (Conv2d-GRU & Conv3d) and arrive at the one which gives the best results in terms of training and validation accuracy. The aim was to produce a model which is able learn enough information on the training set and also use the learnt information to classify the data on the validation set. This is reflected in the training and validation accuracy & the difference between the training and validation accuracy.

Various experiments are conducted, and results are recorded in the below tabular format.

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| 1 | ConvLSTM | Training accuracy is 81% and validation accuracy is 72% in epoch 12 | Activation function is relu in Dense layer, numbers are 10, pixel is Normalized by dividing 255, Batch size is 64 with 4 Conv2d and 1 GRU with 2 dense layers |
| 2 | ConvLSTM | Training accuracy is jumped to 77% and validation accuracy is 69% in epoch 16 | Activation function is relu in Dense layer, numbers are 10, pixel is Normalized by dividing 255, 4 Conv2d and 1 GRU with 2 dense layers. Batch size is increased from 64 to 128 but accuracy has gone down, so it shows that this batch size is not optimum |
| 3 | ConvLSTM | There is significant drop in validation accuracy to around 30%. However, training accuracy is very high | Batch Normalization is added in Conv2d having 64 neurons. Accuracy has gone down as the weights will be very small during back propagation due to further Normalization (added in Conv2d), so model is unable to learn weights properly |
| 4 | ConvLSTM | Training accuracy is jumped to 84% and validation accuracy to 72% in epoch 10 | Activation function is changed to tanh in Dense layer, Gaussian Normalization and batch size is 64. Model is more prone to overfitting if tanh function in Dense layer as compared to RELU |
| 5 | ConvLSTM | Training accuracy is 86% and validation accuracy is 71% in epoch 14 | Activation function is changed to relu in Dense layer and pixel is Normalized by dividing 255, Batch size is 64 and Increase the size of filter from 3\*3 to 5\*5. |
| 6 | ConvLSTM | Training accuracy is 80% and validation accuracy is 68% in epoch 19 | Adding one more Conv2d layer with 128 neurons leads to more overfitting as addition of further layers leads to extraction of more minute features |
| 7 | ConvLSTM | Training accuracy is 20% and validation accuracy is 26% in epoch 15 | Addition of two additional dense layer with 128 neurons. There is substantial decrease in accuracy, so it means model is unable to learn much |
| 8 | ConvLSTM | Training accuracy is 78% and validation accuracy is 70% in epoch 11 | Number of frames are increased from 10 to 15 in each video. It is extracting a greater number of features, so it is taking more time for the execution |
| **Conv3D Model** |  |  |  |
| 9. | Batch Data Generator Class | Customizable image dimensions, no of frames & batch size. | It was created for rapid experimentation to be performed on various parameters |
| 10. | Conv3D – 3 CNN units with batch normalization & dropouts. 2 Flatten layers and a classification layer. | Training accuracy – 24%  Validation accuracy – 21& | Model is not learning properly. Add additional CNN unit and remove batch normalization. |
| 11. | Conv3D – 3 CNN units with dropouts. 2 Flatten layers and a classification layer. | Training Accuracy – 96%  Validation Accuracy – 60% | Model is able to learn but it is still overfitting. An increase in the number of CNN units to accommodate 4 units. |
| 12. | Conv3D – 4 CNN units with dropouts, 2 flatten layers and an output layer. | Training accuracy – 83%  Validation Accuracy – 64% | Huge improvement from the previous model. Still, it is overfitting. Batch Normalization and dropouts are used between the CNN units. |
| 13. | Conv3D - 4 CNN units with dropouts, 2 flatten layers and an output layer | Training Accuracy – 94%  Validation Accuracy - 65% | Model is still overfitting but with a marginal improvement from the previous model. Filter size was increased in the first 2 convolutional layers from 3\*3\*3 to 4\*4\*4 as bigger filter size extracts smaller feature maps (bigger features) which could lead to reduction in overfitting. However, it does not yield expected result |
| 14 | Conv3D - 4 CNN units with dropouts, 2 flatten layers and an output layer | Training Accuracy – 89%  Validation Accuracy - 63% | Model is still overfitting but with a marginal improvement from the previous model. Filter size was increased in all 4 convolutional layers from 3\*3\*3 to 4\*4\*4 as bigger filter size extracts smaller feature maps (bigger features) which could lead to reduction in overfitting. However, it does not yield expected result |

**Recommendation:**

Conv2D+GRU is giving better accuracy with batch size 64, 10 frames per sequence, relu activation function is used in the network and SoftMax is used for the output layer, pixel is Normalized by dividing 255.