|  |
| --- |
| **Experiment** |
|  |
| 1 |
| 2 |
| 3 |
| 4 |

|  |
| --- |
| **Model** |
|  |
| **Simple Conv3D+MaxPool3D** |
| **Multiple Conv3D+MaxPool3D** |
| **Conv2D+RNN(GRU)** |
| **Conv2D+RNN(LSTM)** |

|  |  |  |
| --- | --- | --- |
| **Result** |  |  |
| Training Accuracy | Validation Accuracy | Validation Loss |
| 20.66% | 23% | 1.6076 |
| 83.56% | 65% | 1.19 |
| 96.23% | 62% | 1.33 |
| 92.46% | 67% | 1.08 |

|  |
| --- |
| **Decision+Explanation** |
|  |
| Training accuracy is 20.66% and validation accuracy is 23% which resembles simple conv3d is an underfit model which means modern is learning out of data we can add some more layers to avoid underfitting. The validation loss 1.6076 which can further decreased to avoid underfitting. |
| Training accuracy came as 83.56% with validation accuracy as 65% which has removed the underfitting validation accuracy can be further improved by doing few techniques like data augmentation and training for large no of epochs. As the epochs increase validation loss reached towards training loss which is around 1.19(val\_loss) |
| Training accuracy came as 96.23% with validation accuracy as 62% which has removed the underfitting validation accuracy can be further improved by doing few techniques like data augmentation and training for large no of epochs. As the epochs increase validation loss reached towards training loss which is around 1.33(val\_loss) |
| Training accuracy came as 92.46% with validation accuracy as 67% which has removed the underfitting validation accuracy can be further improved by doing few techniques like data augmentation and training for large no of epochs. As the epochs increase validation loss reached towards training loss which is around 1.0806(val\_loss) |

|  |
| --- |
| **Experiment** |
|  |
| 1 |
| 2 |
| 3 |
| 4 |

These experiments have been done on cropped image. The accuracies have got decreased and validation loss also increased for these experiments. Hence, for these experiments mentioned here cropped images have not been included.

Hence, if we compare Multiple Conv3D + MaxPool3D , Conv2D + RNN(GRU) & Conv2D + RNN(LSTM) , Conv2D + RNN(LSTM) outperforms other models in terms of accuracies, validation loss, overfitting and underfitting. Attached GPU usage below.

Conclusion: Conv2D + RNN(LSTM) we will use for Gesture Recognition TV Application.

Note: Code is not working to save model checkpoints at different epochs so that part is not included in the code.

