**Gesture Recognition – Deep learning**

**Problem Statement**

We are aiming to create an impressive functionality for our smart TV that can accurately identify five distinct gestures performed by users, enabling them to control the TV without the need for a remote. The table below presents the experiments conducted to construct a model capable of predicting the gestures based on the provided dataset.

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D** | Train Accuracy: 0.15, Validation Accuracy: 0.15 | The Model is not learning anything throughout the epochs, the loss is not decreasing. Reducing the batch size further. |
| **2** | **Conv3D** | Train Accuracy: 0.15, Validation Accuracy: 0.20 | No improvement in the model, lets add more layers to the model so that it can learn from data. |
| **3** | **Conv3D** | Negative Dimension Error. | The new CNN kernel sizes are not compatible with the output of previous layers. Let’s reduce the kernel size of new layers. |
| **4** | Conv3D | Train Accuracy: 0.20, Validation Accuracy: 0.20 | Still there is no improvement in the model. Let’s add Batch normalization layers after every CNN and dense layers. |
| **5** | Conv3D | Train Accuracy: 0.9062, Validation Accuracy: 0.2708 | Model is able to over-fit on less data (Ablation data set), Let’s Training on full data and increasing epochs to 50. |
| **6 & 7** | **Conv3D** | Train Accuracy: 0.9062, Validation Accuracy: 0.70 | Mode is having over-fitting as there is huge gap between training and validation accuracies. Let’s add some dropouts that the model can be generalized. |
| **Final Model** |  | Train Accuracy: 0.9673, Validation Accuracy: 0.9375 | This is the best model so far we can get. The validation accuracy is good and the numbers of parameters are 13,589. The model size is also so small 226KB. |

**In conclusion, Experiment #8, which utilized Time distributed Conv2D and ConvLSTM2D, outperformed all other models and boasted the lowest parameter count among them.**