**Deep Learning Project - Gesture Recognition**

**Problem Statement:**

Imagine you are working as a data scientist at a home electronics company which manufactures state of the art smart televisions. You want to develop a cool feature in the smart-TV that can recognize five different gestures performed by the user which will help users control the TV without using a remote.

Each gesture corresponds to a specific command:

* Thumbs up: Increase the volume
* Thumbs down: Decrease the volume
* Left swipe: 'Jump' backwards 10 seconds
* Right swipe: 'Jump' forward 10 seconds
* Stop: Pause the movie

**Model Performance Comparison:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model Architecture** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D (16 -> 32->64)**  **Optimizer: Adam**  **N\_frames:20**  **Batch size:30**  **Epochs:20** | Training accuracy: 30.43%  Validation accuracy: 52.50% | This Model displays a very low training accuracy. |
| **2** | **Conv3D (16 -> 32->64)**  **Optimizer: Adam**  **N\_frames:16**  **Batch size:64**  **Epochs:20** | Training accuracy: 94.65%  Validation accuracy: 37.50% | This model displays a very good training accuracy, but at the same time, validation dataset accuracy is very low.  **This factor makes the model Overfit.** |
| **3** | **Conv3D (16 -> 32->64)**  **Optimizer: Adam**  **N\_frames:20**  **Batch size:50**  **Epochs:30** | Training accuracy: 78.02%  Validation accuracy: 59.00% | This model displays an average training accuracy, also it is slightly **overfit.** |
| **4** | **Conv3D (16 -> 32->64->128)**  **Optimizer: Adam**  **N\_frames:20**  **Batch size:64**  **Epochs:20** | Training accuracy: 84.49%  Validation accuracy: 12.50% | This model displays a good training accuracy, but the validation accuracy seems to be very low. |
| **5** | **Conv3D (16 -> 32->64->128)**  **Optimizer: Adam**  **N\_frames:20**  **Batch size:50**  **Epochs:40** | Training accuracy: 84.62%  Validation accuracy: 74.00% | This model displays pretty good accuracies for both Training and Validation datasets. Let us try and optimize the network further to get better accuracy. |
| **6** | **Conv3D (16 -> 32->64->128)**  **Optimizer: Adam**  **N\_frames:20**  **Batch size:50**  **Epochs:40** | Training accuracy: 100%  Validation accuracy: 71% | This model has displayed a very good Training accuracy, and an average validation accuracy, causing a slight **overfit**. |
| **7** | **Conv2D+LSTM**  **Optimizer: Adam**  **N\_frames:20**  **Batch size:32**  **Epochs:30** | Training accuracy: 99.89%  Validation accuracy: 93.75% | This model has displayed a very good Training and validation accuracies |

**5. Final Model Selection:**

Model 7 is our best model. It has shown an accuracy of **99.89%** on the Training data and an accuracy of **93.75%** on the Validation dataset consistently over multiple Epochs. It follows a Conv2D+LSTM model architecture.