This is a sample write-up. The write-up need not be in tabular form.

It doesn’t state that ConvLSTM will give you better results than Conv3D. The explanation should be as detailed as possible so that the logic behind the decision is conveyed. Also, there are a lot of things you can experiment with in the generator function and elsewhere. Please do not forget to specify the exact metric values, here Accuracy which drives your decision.

You can draw inspiration from the concepts taught in the Industry demo in CNNs to experiment with the data and different architectures.

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D** | **Throws Generator error** | **Crop the images correctly, try to overfit on less amount of data** |
| **2** | **Conv3D** | **Model not trainable as a lot of parameters** | **Reduce the size of the image/Reduce the number of layers** |
| **3** | **Conv3D** | **Accuracy: 0.21** | **Increase the amount of trainable data/ reduce the filter size** |
|  |  |  |  |
|  |  |  |  |
| **2** | **Conv3D** | **Accuracy: 0.32** | **Reduce Cropping** |
| **3** | **Conv3D** | **Accuracy : 0.38** | **………………** |
|  |  |  |  |
| **l-1th** | **Conv3D** | **Accuracy: 0.45** | **Try ConvLSTM as Conv3D not giving desired accuracy** |
| **lth** | **ConvLSTM** | **Accuracy: …….** | **…………..** |
|  |  |  |  |
| **Final Model** | **……………….** | **………….** | **…………………** |
|  |  |  |  |

**Model** #1:

Model : Con3D

Trainable params: 99,269

Non-trainable params: 576

Results :

**categorical\_accuracy: 0.9668**

**val\_categorical\_accuracy: 0.5700**

**Inferences:**

**On the training set the model perfoms well , but on the validation set shows HIGH Variance.**

**Model** #2:

Model : **Conv2D + TimeDistributed + GRU**

Trainable params: 99,269

Non-trainable params: 576

Results :

**categorical\_accuracy: 0.8160**

**val\_categorical\_accuracy: 0.3100**

**Inferences:**

**Performs poorer than the previous model both in the train and validation data.**

**Chaging the pooling to layer and removing the GRU to check another model.**

**Model** #3:

Model : **Conv2D + TimeDistributed + GlobalAveragePooling2D**

Trainable params: 13,589

Non-trainable params: 192

Results :

**categorical\_accuracy: 0.8160**

**val\_categorical\_accuracy: 0.3100**

**Inferences:**

**Performs poorer than the previous model both in the train and validation data.**

**Training with the LSTM model to see any change.**

**Model** #4:

Model : **Conv3D + LSTM**

Trainable params: 2,668,485

Non-trainable params: 448

Results :

**categorical\_accuracy: 0.9653**

**val\_categorical\_accuracy: 0.5400**

**Inferences:**

**With more traininig params, performance on train set is top again, but still the model suffers from high variance in val set**

**Model** #5:

Model : **Conv3D(4 layers)**

Trainable params: 910,965

Non-trainable params: 1,008

Results :

**categorical\_accuracy: 0.9759**

**val\_categorical\_accuracy: 0.5300**

**Inferences:**

**The training data performance has slightly increase but hte val data accuracy has decreased**

**Model** #Final:

Model : **Transfer Learning (mobilenet) + TimeDistributed + LSTM + GRU**

Trainable params: 3,669,317

Non-trainable params: 23,936

Results :

**categorical\_accuracy: 0.9991**

**val\_categorical\_accuracy: 0.9853**

**Inferences:**

**T**he architecture combines the efficiency of MobileNet, the temporal modeling capabilities of LSTM and GRU, and the sequential processing provided by TimeDistributed layers. This combination seems well-suited for the task of gesture recognition, leading to improved accuracy and generalization on the given problem statement.