Hand Gesture Recognition

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D**  **-Batch size = 64**  **-Epochs =15**  **-Number of frames =30**  **- Image size = (120,120)**  **-Kernel size = (3,3,3)**  **-Activation = relu** | **Train accuracy =75.94**  **Validation accuracy =37.50** | **The model was overfitting.**  **Added dropout layers with a dropout value of 0.25** |
| **2** | **Conv3D**  **-Batch size = 64**  **-Epochs =15**  **-Number of frames =30**  **- Image size = (120,120)**  **-Kernel size = (3,3,3)**  **-Activation = relu**  **-Dropout =0.25** | **Train accuracy =59.81**  **Validation accuracy =28.12** | **Adding dropout layer did not help with the problem of overfitting. Built another model considering only 20 frames, decreasing the batch size to 20, increasing the number of epochs to 25 and adding dropout layers with a dropout value of 0.5.** |
| **3** | **Conv3D**  **-Batch size = 20**  **-Epochs =25**  **-Number of frames =20**  **- Image size = (120,120)**  **-Kernel size = (3,3,3)**  **-Activation = relu**  **-Dropout =0.5** | **Train accuracy =39.22**  **Validation accuracy =23.0** | **The model is underfitting it didn’t learn much. In the next model we have used all the frames, kept the image size same, increased the batch size to 32 and reduced the filter size to (2,2,2).** |
| **4** | **Conv3D**  **-Batch size = 32**  **-Epochs =25**  **-Number of frames =30**  **- Image size = (120,120)**  **-Kernel size = (2,2,2)**  **-Activation = relu** | **Train accuracy =83.75**  **Validation accuracy =43.75** | **The train accuracy increased but the model was still overfitting. Built another model keeping everything same and adding dropout layers with dropout value of 0.25 and reduced the number of epochs to 15.** |
| **5** | **Conv3D**  **-Batch size = 32**  **-Epochs =15**  **-Number of frames =30**  **- Image size = (120,120)**  **-Kernel size = (2,2,2)**  **-Activation = relu**  **- Dropout = 0.25** | **Train accuracy =93.28**  **Validation accuracy =87.5** | **This is the best model so far.**  **Built another model. Built another model keeping everything same and adding dropout layers with dropout value of 0.5.** |
| **6** | **Conv3D**  **-Batch size = 32**  **-Epochs =15**  **-Number of frames =30**  **-Image size = (120,120)**  **-Kernel size = (2,2,2)**  **-Activation = relu**  **- Dropout = 0.5** | **Train accuracy =48.18**  **Validation accuracy =18.75** | **The model has clearly underfitted.** **In an attempt to reduce the number of parameters, we have reduced the number of frames to 16, decreased the batch size to 20, increased the number of epochs 20 and reduced the size of the filter in the subsequent layers.** |
| **7** | **Conv3D**  **-Batch size = 20**  **-Epochs =20**  **-Number of frames =16**  **- Image size = (120,120)**  **-Kernel size = (3,3,3), then reduced to (2,2,2)**  **-Activation = relu**  **-Dropout = 0.25** | **Train accuracy =40.2**  **Validation accuracy =33.0** | **The performance given by the model isn’t up to the mark. Built another model by increasing the dropout value to 0.5, increasing the batch size to 64, decreasing the epochs to 15 and increasing the kernel to (3,3,3).** |
| **8** | **Conv3D**  **-Batch size = 64**  **-Epochs =15**  **-Number of frames =16**  **- Image size = (120,120)**  **-Kernel size = (3,3,3)**  **-Activation = relu**  **-Dropout = 0.5** | **Train accuracy =42.11**  **Validation accuracy =12.5** | **We have been successful in reducing the number of parameters but it has cost us dearly in terms of accuracy. Built another model with an architecture similar to model 7, added dropout layer with a dropout value of 0.25 and set filter size to (3,3,3).** |
| **9** | **Conv3D**  **-Batch size = 64**  **-Epochs =15**  **-Number of frames =16**  **- Image size = (120,120)**  **-Kernel size = (3,3,3)**  **-Activation = relu**  **-Dropout = 0.25** | **Train accuracy =75.94**  **Validation accuracy =62.5** | **We have seen some improvement in terms of accuracy but the problem of overfitting has still not been solved. Next, we started creating LSTM models** |
| **10** | **Conv2D-LSTM**  **-Batch size = 20**  **-Epochs = 20**  **-Number of frames =18**  **- Image size = (120,120)**  **-Kernel size = (3,3)**  **-Activation = relu**  **-Dropout = 0.25** | **Train accuracy =56.86**  **Validation accuracy =53.0** | **There's hardly any overfitting with this one, but we still have a lot of room for improvement in terms of accuracy. In an attempt to improve the accuracy, we have added another hidden layer to the same architecture.** |
| **11** | **Conv2D-LSTM**  **-Batch size = 20**  **-Epochs = 20**  **-Number of frames =18**  **- Image size = (120,120)**  **-Kernel size = (3,3)**  **-Activation = relu**  **-Dropout = 0.25** | **Train accuracy =48.04**  **Validation accuracy =37.0** | **Addition of another hidden layer did not help us with any increase in the accuracy. Built another model using the same architecture as model 10 and increased the dropout value to 0.5** |
| **12** | **Conv2D-LSTM**  **-Batch size = 20**  **-Epochs = 20**  **-Number of frames =18**  **-Image size = (120,120)**  **-Kernel size = (3,3)**  **-Activation = relu**  **-Dropout = 0.5** | **Train accuracy =42.16**  **Validation accuracy = 55.0** | **The overall performance of this model is not up to the mark. Neither the accuracy achieved is good nor the problem of overfitting has been addressed. Built another model using all the 30 frames, dropout value of the 0.5, batch size of 64 and number of epochs 15.** |
| **13** | **Conv2D-LSTM**  **-Batch size = 64**  **-Epochs = 15**  **-Number of frames =30**  **-Image size = (120,120)**  **-Kernel size = (3,3)**  **-Activation = relu**  **-Dropout = 0.5** | **Train accuracy =50.64**  **Validation accuracy =28.12** | **The model is under performing. Built another model reducing the size of image (100,100). Rest of the architecture is same.** |
| **14** | **Conv2D-LSTM**  **-Batch size = 64**  **-Epochs = 15**  **-Number of frames =30**  **-Image size = (100,100)**  **-Kernel size = (3,3)**  **-Activation = relu**  **-Dropout = 0.5** | **Train accuracy =62.57**  **Validation accuracy =12.5** | **The accuracy has certainly increased but the model is still overfitting on the validation set. In the next model, we have reduced the number of frames to 16 keeping the size of image as (100, 100). Rest all parameters are same as Model 14.** |
| **15** | **Conv2D-LSTM**  **-Batch size = 64**  **-Epochs = 15**  **-Number of frames =16**  **-Image size = (100,100)**  **-Kernel size = (3,3)**  **-Activation = relu**  **-Dropout = 0.5** | **Train accuracy =58.0**  **Validation accuracy =12.5** | **The model seems to have achieved a better accuracy but the problem of overfitting is persistent.** |

**Best model**

After building a total of 15 models, the best model which we have selected keeping in mind the number of parameters, the accuracy and the amount of overfitting, is Model 5.

The specifications of the best model is:

1. Training accuracy: 93%

2. Validation accuracy: 87.5%

3. Number of Frames: 30

4. Image Size: 120, 120

5. Filter Size: 2,2,2

6. Batch Size: 32

7. Number of Epochs: 15

8. Total Parameters: 840k (approximately)

A plot showing the variation of loss and accuracy across epochs is shown below.

