**Building a Conv3D Neural Network Model for Gesture Recognition:**

**Generator Function:**

We have resized the images to (60,60) and (100,100) in the trial runs in the generator function. We have used less than 30 frames in the network learning.

The experiments/trials made to build the model are as follows:

1. We started with a simple neural network architecture with two 32 neuron layers and a dense layer connected to the softmax output layer. The image size and hyper parameters are initialized as below:

|  |  |  |
| --- | --- | --- |
| Model Architecture | 32 -2, 256(dense) | Start with a simple layer |
| Image Size | (60,60,3) | Image resized to a smaller size to see if that would suffice to learn the different gestures |
| Number of Frames | 10 | Frames starting from 14th till 24 considered as those are the prime frames which shows start to end of a gesture |
| Batch Size | 20 | Considered a smaller number. Tried with more than 40. The execution failed because of ‘Runtime exceeded error’. Could be OOM error. |
| Optimiser | SGD | SGD with defaults parameter values |
| Learning Rate | Default | Default provided by SGD |
| Dropout | None | Not included to check the learning first |
| Batch Normalisation | None | Not included to check the learning first |
| Number of Epochs | 10 | Start with a smaller number |
| Training Accuracy | 45 % |  |
| Validation Accuracy | 26 % |  |

1. Increased the number of layers and frames. Details given below:

|  |  |  |
| --- | --- | --- |
| Model Architecture | 32 -2, 64-2, 128-1, 256(dense) | Added more layers for more feature extraction |
| Number of Frames | 12 | Number of frames increased to read in more information for the video |
| Dropout | Added to the last layer-(0.25) and after dense – (0.5) | To keep a check on the Overfitting |
| Training Accuracy | 84 % |  |
| Validation Accuracy | 44 % |  |

1. Increased the image size and epochs: Details given below:

|  |  |  |
| --- | --- | --- |
| Image Size | (100,100) | Increased the image size to have better resolution to read more information about the images |
| Epochs | 15 | Increased number of epochs to reduce loss |
| Training Accuracy | 92 % |  |
| Validation Accuracy | 55 % |  |

1. Changed the learning rate: Details given below:

|  |  |  |
| --- | --- | --- |
| Learning Rate | 0.01 | Enforced a learning rate as specified |
| Training Accuracy | 99 % | Overfitting |
| Validation Accuracy | 62 % |  |

1. Dropout after all the Pooling layers, Complete data, Increased batch size: Details given below:

|  |  |  |
| --- | --- | --- |
| Dropouts | 0.25, 0.5(dense) | Added after all the Max Pooling layers to reduce the Overfitting |
| Input data | 663, 100 | Included full Data for training and validation. Increasing training data has to be reduce overfitting. |
| Batch Size | 30 |  |
| Training Accuracy | 82 % |  |
| Validation Accuracy | 41 % |  |

1. Removed few dropouts: Details given below:

|  |  |  |
| --- | --- | --- |
| Dropouts | 0.25, 0.5(dense) | Removed dropouts in the initial layers to retain the information. |
| Training Accuracy | 95 % |  |
| Validation Accuracy | 67 % |  |

1. Modified LR: Details given below:

|  |  |  |
| --- | --- | --- |
| Learning Rate | 0.001 | Decreased Learning rate to see if it improves the validation accuracy |
| Training Accuracy | 86 % |  |
| Validation Accuracy | 65 % |  |

1. Modified architecture and higher epochs: Details given below:

|  |  |  |
| --- | --- | --- |
| Model Architecture | 32 -2, 64-2, 128-2, 256(dense) | Added another layer of 128 neurons to extract more finer features |
| Number of Epochs | 20 | Increased to 20 from 15 to reduce loss |
| Training Accuracy | 96 % |  |
| Validation Accuracy | 69 % |  |

1. Used Adam optimiser: Details given below:

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Optimiser | Adam | Changed the optimizer to Adam to see the effect |
| Number of epochs | 10 |  |
| Training Accuracy | 23 % | Drop in the Accuracies |
| Validation Accuracy | 24 % |  |

1. Used Adagrad optimiser: Details given below:

|  |  |  |
| --- | --- | --- |
| Optimiser | Adagrad | Changed the optimizer to Adagrad to see the effect |
| Number of epochs | 10 |  |
| Training Accuracy | 32 % | Drop in the Accuracies |
| Validation Accuracy | 33 % |  |

1. Reverted to SGD and further reduced LR: Details given below:

|  |  |  |
| --- | --- | --- |
| Learning rate | 0.0001 | Further reduced |
| Optimiser | sgd | Changed the optimizer back to sgd |
| Number of epochs | 10 | Still with the lower epoch |
| Training Accuracy | 59 % | Low accuracy |
| Validation Accuracy | 53 % |  |

1. Reverted the LR and higher epochs: Details given below:

|  |  |  |
| --- | --- | --- |
| Learning rate | 0.001 | Reverted |
| Optimiser | sgd | Changed the optimizer back to sgd |
| Number of epochs | 25 | Increased epochs to reduce further loss |
| Training Accuracy | 97 % | Accuracy Increased. For one of the epochs |
| Validation Accuracy | 79 % |  |

1. Making the model deeper and additional dropouts : Details given below:

|  |  |  |
| --- | --- | --- |
| Model Architecture | 32 -2, 64-3, 128-3, 256(dense) | Additional 64, 128 layers. |
| Dropouts | 0.25 | Added dropouts after all the maxpooling to control overfitting |
| Training Accuracy | 92 % |  |
| Validation Accuracy | 74 % |  |

1. Making the model complex : Details given below:

|  |  |  |
| --- | --- | --- |
| Model Architecture | 64-2, 128-2, 256 – 2, 512(dense) | Started with the 64 layer of neurons |
| Training Accuracy | - | Resource exhausted error – 41 mil parameters |
| Validation Accuracy | - |  |

1. Reverted to earlier architecture: Details given below:

|  |  |  |
| --- | --- | --- |
| Model Architecture | 32-2,64-2,128-2,256-2,512-2, 1024(dense) | Additional 2 layers of 512 neurons – Making deeper NN |
| Dropouts | Included only last 2 dropouts |  |
| Training Accuracy | 93 % |  |
| Validation Accuracy | 69 % |  |

1. Reduce the complexity of the model: Details given below:

|  |  |  |
| --- | --- | --- |
| Model Architecture | 32-2,64-2,128-2,256-2,512-1, 1024(dense) | Have only one layer of 512 neurons |
| Dropouts | Included last 3 dropouts |  |
| Training Accuracy | 86 % |  |
| Validation Accuracy | 76 % |  |

1. One of the final set up - Removed one more dropout layer and introduced ReduceLR code: Details given below:

|  |  |  |
| --- | --- | --- |
| Model Architecture | 32-2,64-2,128-2,256-2,512-1, 1024(dense) | Have only one layer of 512 neurons |
| Dropouts | Included last 2 dropouts | Removed one more dropout to minimize loss of information |
| Learning rate | ReduceLRonPlateau | Included Reduce Lr on Plateau to reduce the learning rate if the observed ‘val\_loss’ doesn’t change for a certain number of epochs |
| Training Accuracy | 91 %, 94% | Of two epochs with higher accuracies in the same run |
| Validation Accuracy | 75 %, 74 % |  |