Monika write ups for different experiments

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D Model 1**  -Batch size 32  -Epoch 10  -Image size and no of Frames 64\*64, 15 frames  -Adam optimizer with 0.0002 learning rate | .Keras file was not saved due to “filePath” not formatted correctly | **Repair the filepath with an extension to .keras, and succeeded to save model file.** |
| **2** | **Conv3D Model 1**  -Batch size 32  -Epoch 10  -Image size and no of Frames 64\*64, 15 frames  -Adam optimizer with 0.0002 learning rate | The significant difference between training accuracy (57%) and validation accuracy (16%) across Trainable params: 143925 in neural network model suggests the presence of overfitting. | **Modify parameters to address the overfitting problem** |
| **3** | **Conv3D Model 2**  -Increased Filter Counts  -Dropout Layers  -BatchNormalization  -Reduced the learning rate to 0.0001  -Batch size 30  -Epoch 20 | -training accuracy: 0.2309  - loss: 2.4367  -val\_categorical\_accuracy: 0.2100  - val\_loss: 2.0044 | **Result shows we covered the overfitting problem now we try to raise accuracy by changing model**  **Now I’ll keep epochs lower because it exhaust my GPU RAM more.** |
| **4** | **Conv3D Model 3**  -10 epochs and 39 batch size  -image size increased to 84X84  -took 13 image frames  -reduced cropping to 3 % from 10 %  -Reduced the learning rate to 0.0001 | -categorical\_accuracy: 0.2414  -loss: 2.4675  -val\_categorical\_accuracy: 0.2100  -val\_loss: 1.6708 | **Change in model gives slight high**  **Accuracy in training data, where I reduced the epochs. Although model will perform well if epochs get increase.** |
| **5** | **Conv3D Model 4**  -15 epochs and 50 batch size  -image size increased to 100X100  -Reduced the learning rate to 0.001  -reduce the params by ---reducing layers in model  -Global Average Pooling instead of Flatten  -check with early stop callback | -categorical\_accuracy: 0.2482  -loss: 2.2167  -val\_categorical\_accuracy: 0.2200  -val\_loss: 1.6217 | **Due to early stop call back training got stop at 6th epochs**  **-The EarlyStopping callback in TensorFlow/Keras is used to monitor the training process and stop it early when the model's performance on the validation set stops improving.** |
| **6** | **Conv2D + LSTM model 1**  Epochs 8  Batch Size 8  Image frames as I/P 8  64\*64 Image size  Optimizer Adam learning\_rate=0.0005 | -categorical\_accuracy: 0.2726  -loss: 1.5508  -val\_categorical\_accuracy: 0.2700  -val\_loss: 1.6089 | **Not much difference between training and validation accuracy. Now we try to improve more accuracy.** |
| **7** | **Conv2D – LSTM model 2**  Added one more layers of Conv2D from previous iteration  Epochs 10  Batch Size 16  Image frames as I/P 8  64\*64 Image size  Optimizer Adam learning\_rate=0.0005 | -categorical\_accuracy: 0.4488  -loss: 1.3741  -val\_categorical\_accuracy: 0.4200  -val\_loss: 1.4453 | **Accuracy is improved while changing params in LSTM model, gives good accuracy** |
| **8** | **Pre-trained model(ResNet 50) for Transfer learning**  Epochs 5  Batch Size 4  Image frames as I/P 8  64\*64 Image size  Optimizer Adam learning\_rate=0.0001 | -categorical\_accuracy: 0.2762  -loss: 1.5854  -val\_categorical\_accuracy: 0.2800  -val\_loss: 1.5734 | **Seems validation data have good score, here we can perform more experiments** |
| **9** | **Pre-trained model 2(ResNet 50) for Transfer learning**  Epochs 8  Batch Size 8  Image frames as I/P 8  64\*64 Image size  Optimizer Adam learning\_rate=0.0001 | -categorical\_accuracy: 0.3147  -loss: 1.5565  -val\_categorical\_accuracy: 0.3800  -val\_loss: 1.4818 | **Did not get good accuracy needs to perform more epochs and different batch size to optimize Accuracy** |