For this case study, tried building models with Conv3D and Resnet. The Resnet model did not work, it was throwing a lot of errors, will have to figure out how to make it work.

So the final model is built on Conv3D, with total 21 layers, run over 100 epochs. Following are the parameters:

Total params: 37,941,637

Trainable params: 37,941,253

Non-trainable params: 384

|  |  |  |  |
| --- | --- | --- | --- |
| **Experiment Number** | **Model** | **Result** | **Decision + Explanation** |
| **1** | **Conv3D** | **View random image** | **Viewed one random image and checked its shape** |
| **2** | **Conv3D** | **Resized image** | **Resized sample image to shape (128, 128)** |
| **3** | **Conv3D** | **Cropped image** | **Center cropped sample image, final shape (100, 100)** |
| **4** | **Conv3D** | **Data Generator** | **Applied the same resize and crop function to Generator function** |
| **5** | **Conv3D** | **Data Generator** | **Yield remaining batches from Generator function** |
| **6** | **Conv3D** | **Model was throwing error for negative output** | **Reduced pool\_size for MaxPoll3D to 1** |
| **7** | **Conv3D** | **Accuracy: around 0.21** | **Learning rate used 0.1** |
| **8** | **Conv3D** | **Accuracy: around 0.35** | **Learning rate used 0.01** |
| **9** | **Conv3D** | **Accuracy: training = 0.77**  **Val = 0.65** | **Learning rate used 0.001** |
| **Final Model** | **Conv3D** | **Learning rate = 0.0010** | **loss: 1060.9886 - categorical\_accuracy: 0.7783 - val\_loss: 689.8799 - val\_categorical\_accuracy: 0.6500 - lr: 0.0010** |