**OBSERVATIONS:**

* It was observed that as the Number of trainable parameters increase, the model takes much more time for training.
* A large batch size can throw *GPU Out of memory error,* and thus here we had to play around with the batch size till we were able to arrive at an optimal value of the batch size which our GPU could support.
* Increasing the batch size greatly reduces the training time but this also has a negative impact on the model accuracy. This made us realise that there is always a trade-off here on basis of priority -> If we want our model to be ready in a shorter time span, choose larger batch size else you should choose lower batch size if you want your model to be more accurate.

| Experiment Number | Model | Result | Parameters | Decision + Explanation |
| --- | --- | --- | --- | --- |
| 1 | Conv3D | Training accuracy: 0.42  Validation accuracy:0.0000e+00 | 7,708,349 | Creating a model with 160x160 image size,epochs=10 and batch\_size=32 |
| 2 | Conv3D | Training accuracy: 0.25  Validation accuracy:0.19 | 1,283,909 | Reduce the batch size to 20 and image dimensions to 60x60 with 30 frames |
| 3 | Conv3D | Training Accuracy: 0.34  Validation accuracy:0.21 | 1,429,621 | adding more layers and increase the no of epochs to 15 |
| 4 | Conv3D | Training Accuracy: 0.92  Validation accuracy:0.85 | 2,177,829 | change the image resolution to 80x80 ,with frames=30 and by keeping batchsize=10 and epochs=20 |
| 5 | Conv2D + GRU | Training Accuracy: 0.56  Validation accuracy:0.68 | 1,000,293 | custom conv2d + GRU |
| 6 | Conv2D + GRU | Training Accuracy: 0.68  Validation accuracy: 0.63 | 1,934,949 | Let us add more dense nuerons and gru cells and examine the results. |
| 7 | Conv2D + GRU | Training Accuracy: 0.80  Validation accuracy:0.69 | 2,034,021 | add layers in GRU |
| 8 | Conv2D + GRU | Training Accuracy: 0.82  Validation accuracy:0.76 | 2,034,021 | Add dropouts,and set the learning rate to 0.0001 |

**Thus final model is Exp-4 with Training Accuracy: 0.92, Validation accuracy:-0.85 with least number of parameters.**