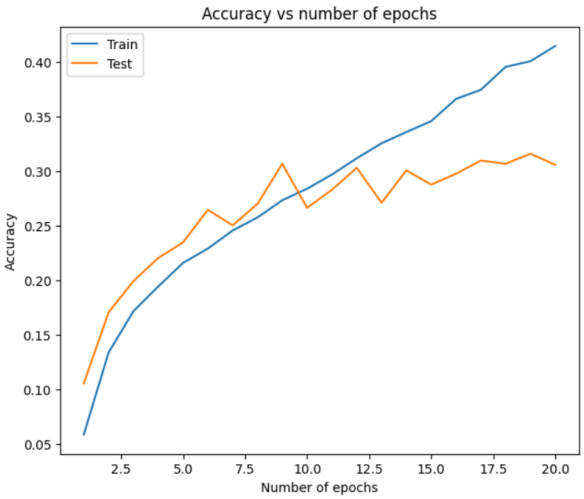
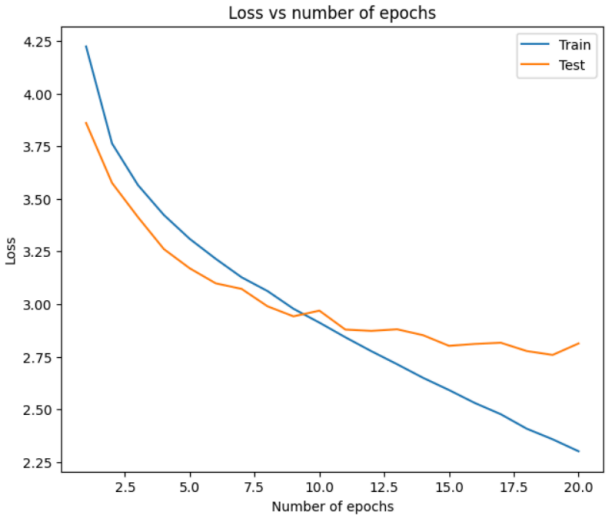
Sports Images Classification

Name: Siddharth Rajandekar

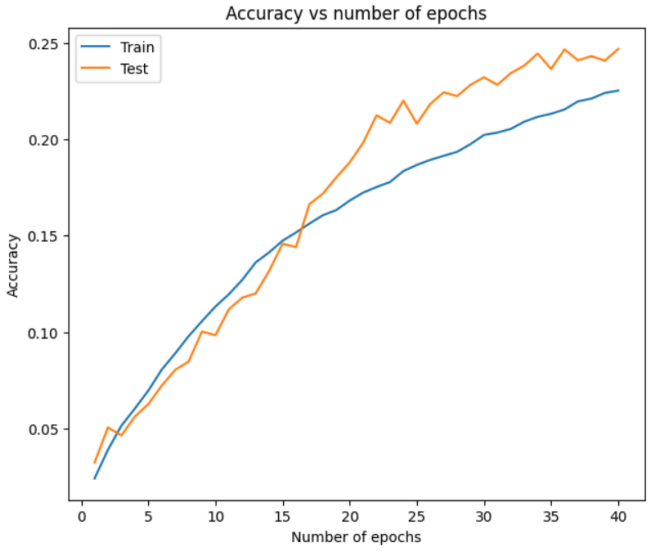
LeNet

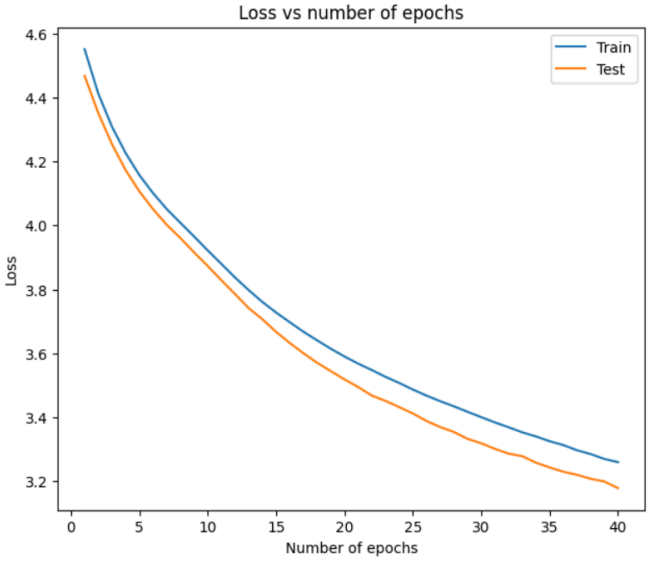
Number of Epochs: 20, Learning Rate = 0.001, Batch Size = 128
Train accuracy: 0.414742
Test accuracy: 0.315934





Number of Epochs: 40, Learning Rate = 0.0001, Batch Size = 128
Train accuracy: 0.225315
Test accuracy: 0.246901

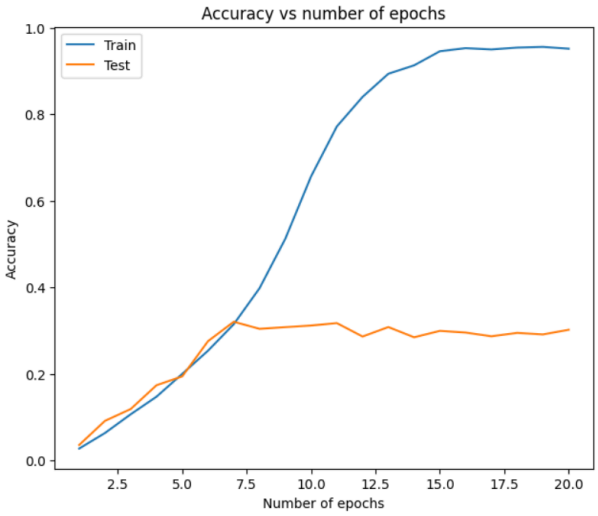


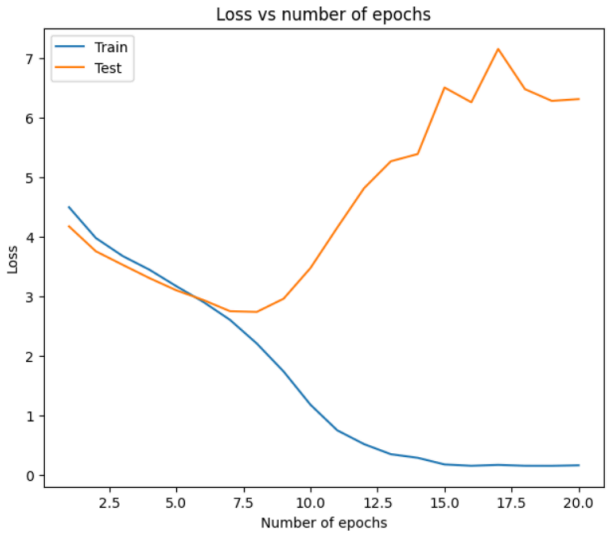


CONCLUSION:
As we can see from the graph, the model is a clear case of underfitting. Even after running for 40 epochs with 10e-4 learning rate, the model is giving very less accuracy. This is a case of low bias. So, we need a more complicated model.

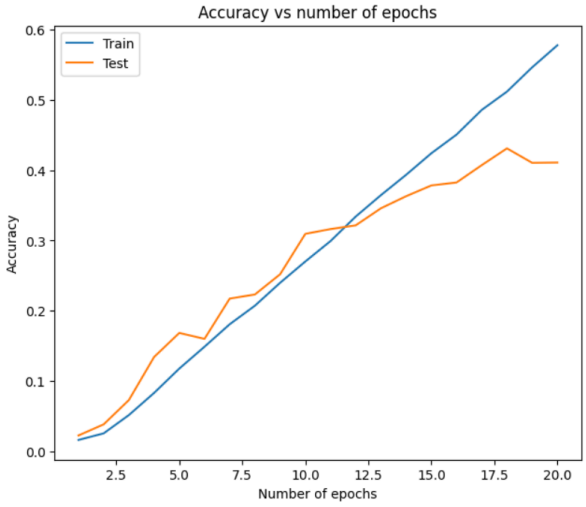
AlexNet

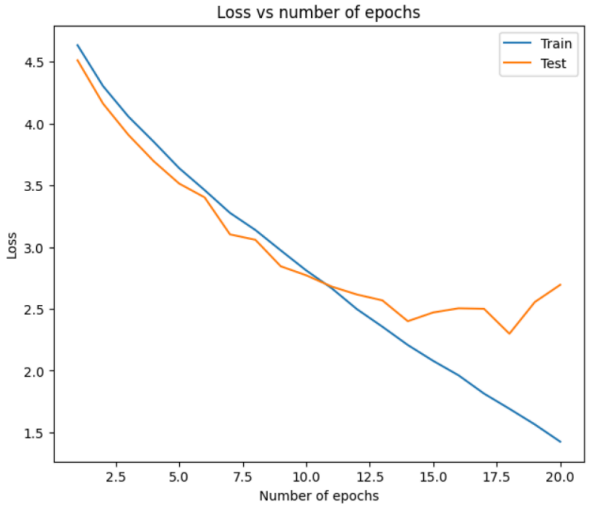
Number of Epochs: 20, Learning Rate = 0.001, Batch Size = 128
Train accuracy: 0.955716
Test accuracy: 0.320515





Number of Epochs: 20, Learning Rate = 0.001, Batch Size = 128 (With additional data augmentation)
Train accuracy: 0.577564
Test accuracy: 0.430967





CONCLUSION:

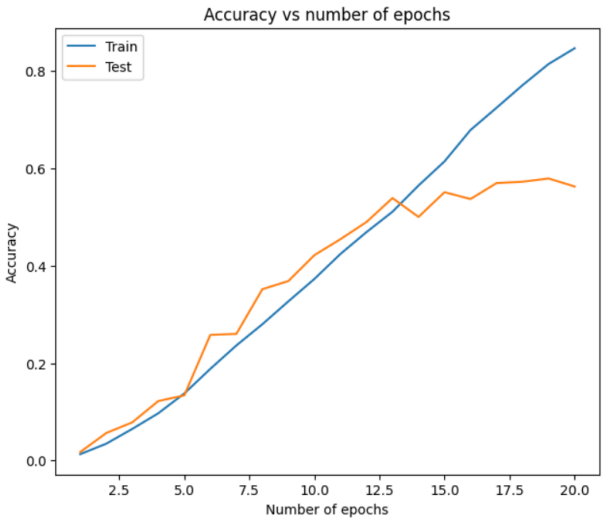
Next I tried AlexNet which is more complicated that LeNet. Initially after running the model it was overfitting the dataset and hence the accuracy becomes stagnant at around 32 %. So, I added additional data augmentation to reduce the overfitting problem and it seems to have worked. The accuracy is improved, but it seems that we could have further improved it by running it for more epochs, but it was taking a lot of time to train, so I decided to try a better model.

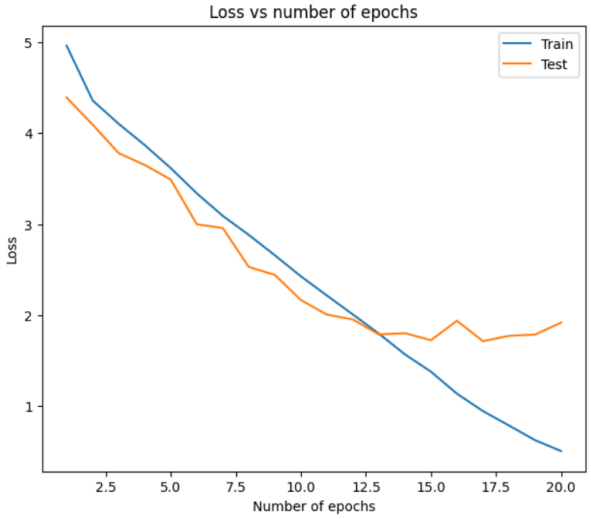
VGGNet16

Number of Epochs: 20, Learning Rate = 0.001, Batch Size = 32 (with SGD optimizer)

Train accuracy: 0.846608

Test accuracy: 0.579297





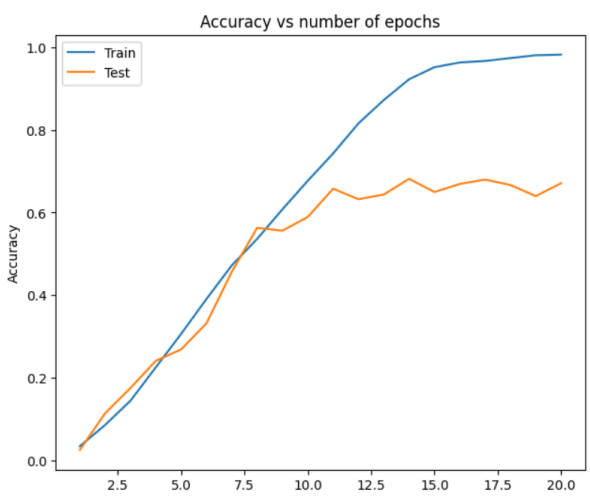
CONCLUSION:

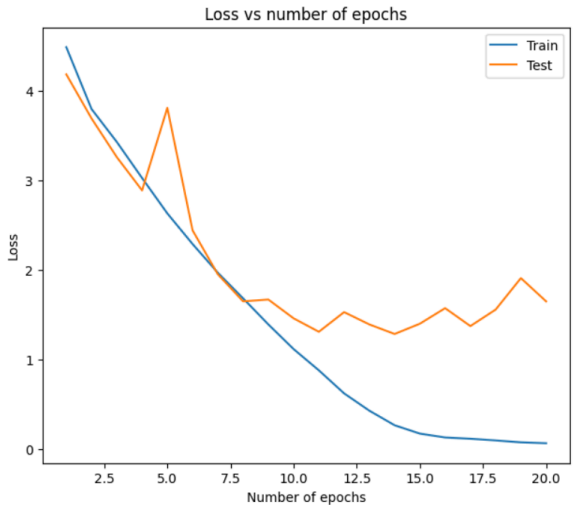
I first tried running VGGNet16 to see if it performs any better. But there were some issues with the model. Initially when I implemented it manually, it did not seem to be training and I could not figure out the issue due to time constraints. After using the in-built function also it did not give a very good accuracy and it was taking huge amount of time to train. So, I abandoned the effort of getting better results with this model.

ResNet18

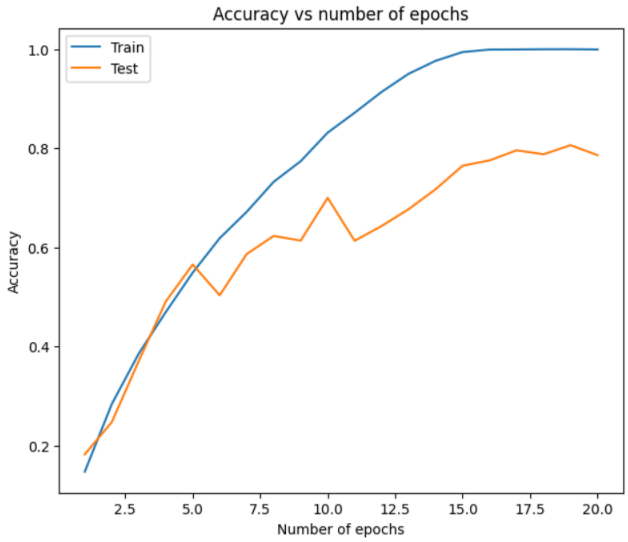
Number of Epochs: 20, Learning Rate = 0.01, Batch Size = 128

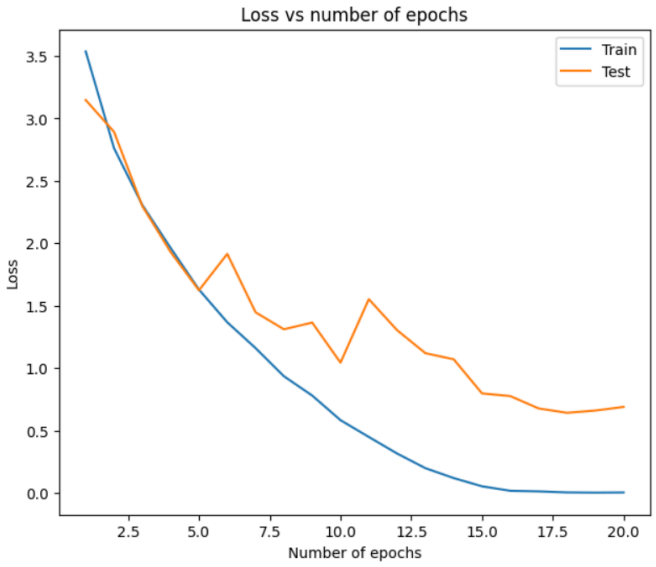
Train accuracy: 0.981540
Test accuracy: 0.681438



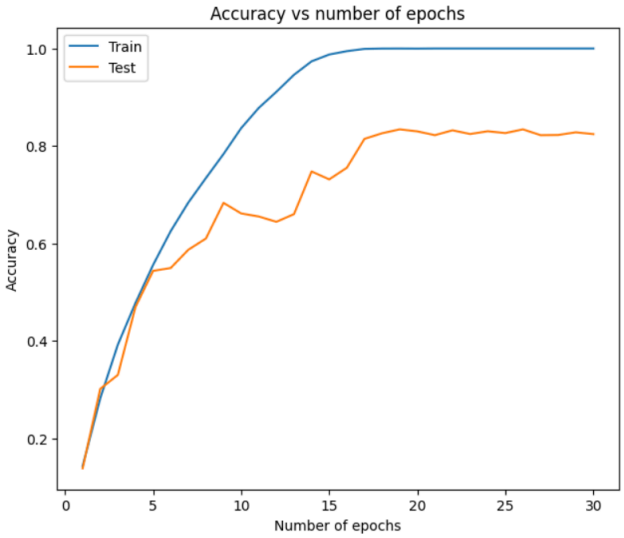


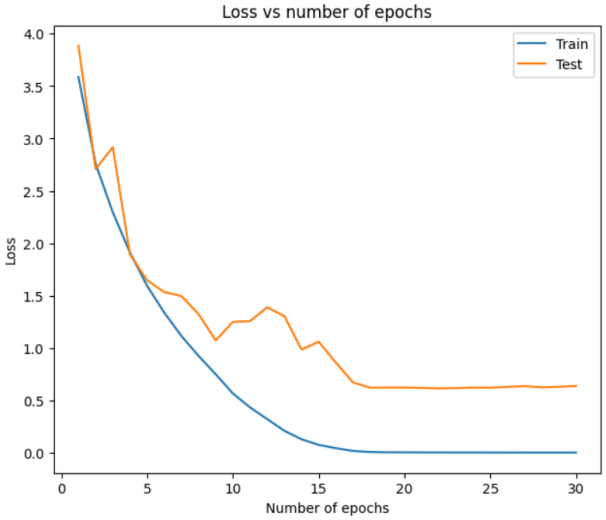
Number of Epochs: 20, Learning Rate = 0.001, Batch Size = 128
Train accuracy: 0.999926
Test accuracy: 0.806303





Number of Epochs: 30, Learning Rate = 0.001, Batch Size = 128
Train accuracy: 1.0
Test accuracy: 0.834051





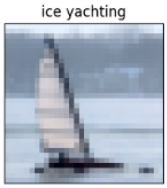
CONCLUSION:

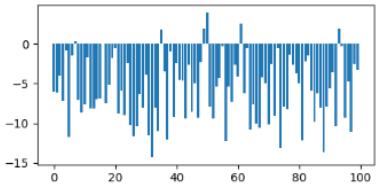
Next I ran ResNet18 which is much better than AlexNet and we already see that with a learning rate of just 10e-2, it performs much better than AlexNet. So, I decided to increase the learning rate because the model showed potential of performing even better and I was able to reach an accuracy of around 83 % which is the best among all the models I used for this project.

Final Results

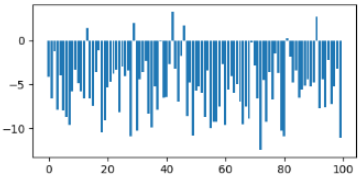
Network Architecture	Train Accuracy	Test Accuracy	Epochs
Resnet18	1.0	0.834051	30
AlexNet	0.955716	0.320515	20
AlexNet with Data Augmentation	0.577564	0.430967	20
VGGNet	0.846608	0.579297	20
LeNet	0.414742	0.315934	20

ice yachting

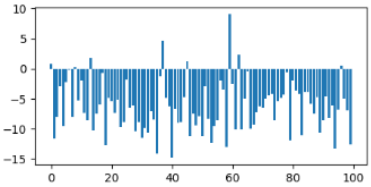




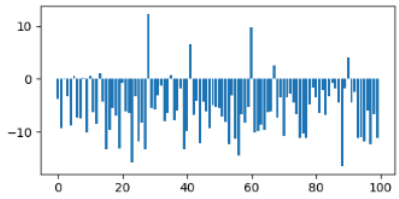
high jump



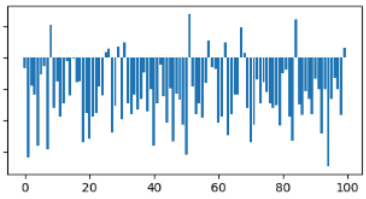
gaga



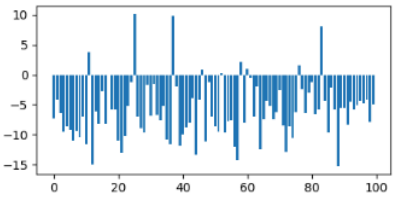
shuffleboard



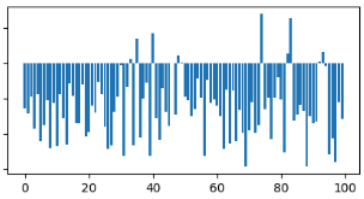
barell racing



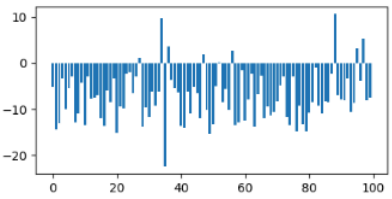
nascar racing



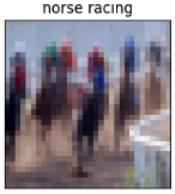
fly fishing



cricket



horse racing



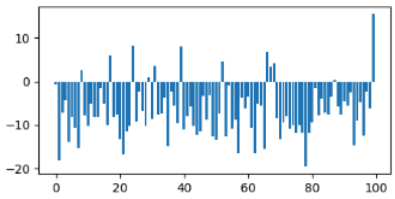


figure skating women

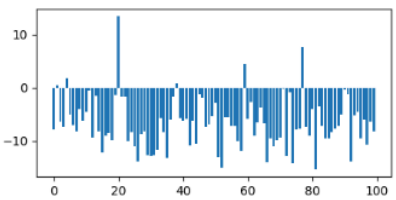
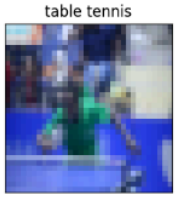
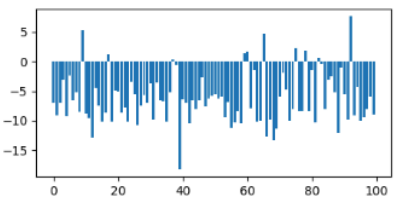


table tennis





swimming