Dataset

100 Sports Classes, 13572 Training Samples, 500 Validation Samples, 500 Test Samples, 224x224 RGB images.

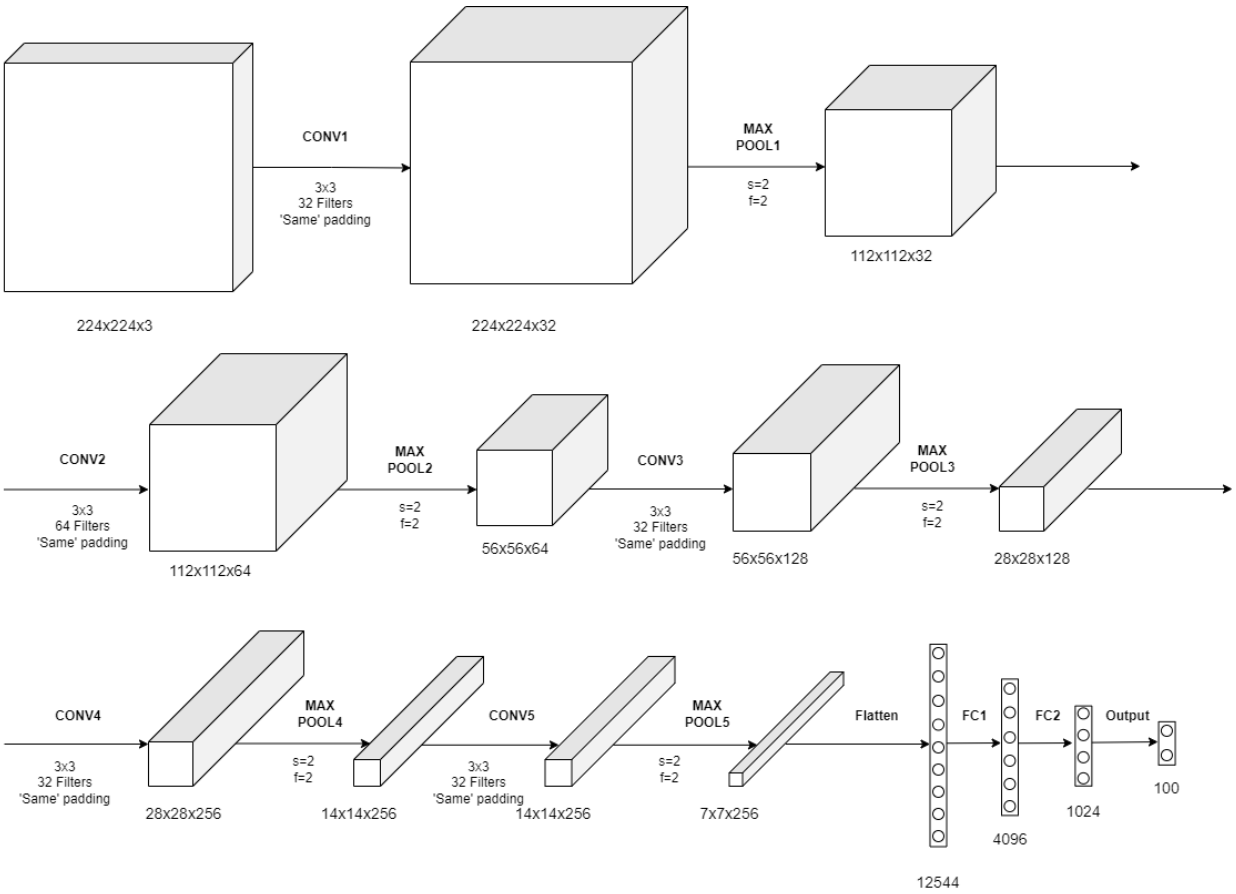
Initial Model

An initial model is run with 5 Conv layers, 5 Max Pooling layers, 2 Fully Connected layers, and 1 Output layer. The total number of trainable parameters: 56,660,644

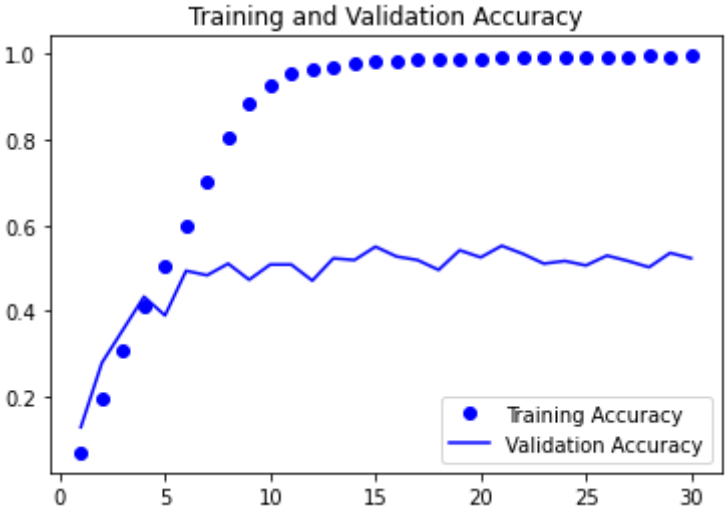
```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 224, 224, 32)      896

 max_pooling2d (MaxPooling2D  (None, 112, 112, 32)     0
 )

 conv2d_1 (Conv2D)           (None, 112, 112, 64)      18496

 max_pooling2d_1 (MaxPooling  (None, 56, 56, 64)       0
 2D)

 conv2d_2 (Conv2D)           (None, 56, 56, 128)       73856

 max_pooling2d_2 (MaxPooling  (None, 28, 28, 128)      0
 2D)

 conv2d_3 (Conv2D)           (None, 28, 28, 256)       295168

 max_pooling2d_3 (MaxPooling  (None, 14, 14, 256)      0
 2D)

 conv2d_4 (Conv2D)           (None, 14, 14, 256)       590080

 max_pooling2d_4 (MaxPooling  (None, 7, 7, 256)        0
 2D)

 flatten (Flatten)           (None, 12544)             0

 dense (Dense)               (None, 4096)              51384320

 dense_1 (Dense)             (None, 1024)              4195328

 dense_2 (Dense)             (None, 100)               102500

=================================================================
Total params: 56,660,644
Trainable params: 56,660,644
Non-trainable params: 0
_________________________________________________________________
```

Loss: Categorical-Crossentropy
Optimizer: RMSprop with a learning rate of 0.0001
Metric: Categorical Accuracy
Train Batch Size: 64, **Validation Batch Size:** 32

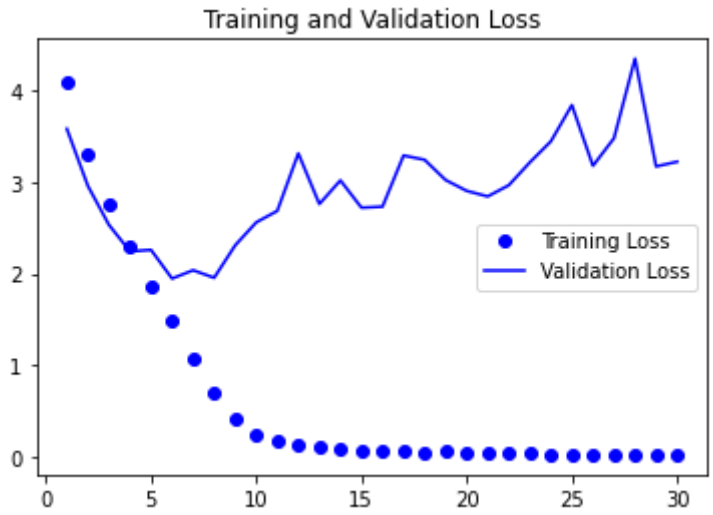
Network Architecture



Train and Validation Accuracy



Train and Validation Loss



Observation & Conclusion

After ~5-6 epochs, the model starts overfitting. The model did very well on the training dataset (~99.7% accuracy) but couldn't generalize well on the validation set (On average ~50% accuracy). The reason for such severe overfitting is that the model is too complex for the given dataset.

Solution

Solutions for overfitting:

- Add more data to the training dataset
- Data Augmentation
- Reduce the complexity of the model
- Add Dropout regularization
- Add L2 Regularization
- Use other architectures.

For convolutional neural networks, data augmentation, dropout, and using other architectures are the most practical ways of reducing overfitting. Augmenting and adding dropout regularization are the first things to try.