CS671 - Deep Learning and its Applications

Programming Assignment 2

Group - 20

Problem Statement

Design a non-sequential convolutional neural network for classifying the line dataset. This network will have 4 outputs based on the 4 kind of variations(length, width, color, angle). You are required to divide your network architecture into two parts :

a) Feature network

b) Classification heads

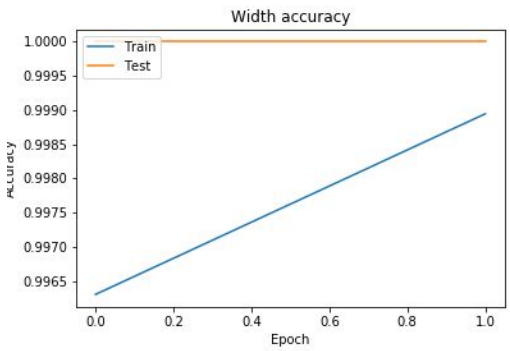
The feature network will be responsible for extracting the required features from the input and attached to it would be the four classification heads one for each variation.

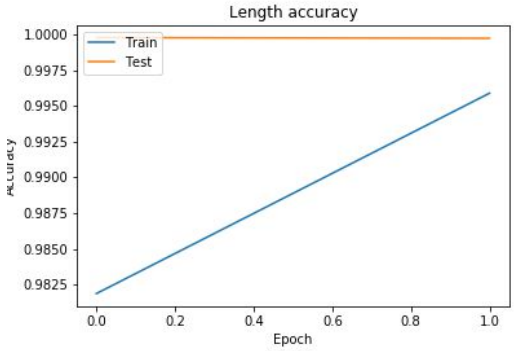
Model

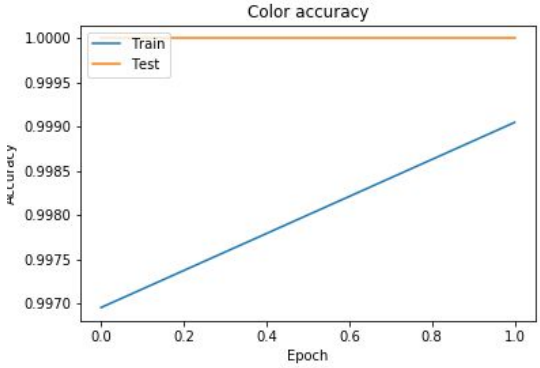

input_19: InputLayer
conv2d_64: Conv2D
batch_normalization_136: BatchNormalization
activation_135: Activation
max_pooling2d_44: MaxPooling2D
conv2d_65: Conv2D
batch_normalization_137: BatchNormalization
max_pooling2d_45: MaxPooling2D
conv2d_66: Conv2D
batch_normalization_138: BatchNormalization
activation_136: Activation
flatten_19: Flatten
dense_73: Dense
dense_74: Dense
dense_75: Dense
dense_76: Dense
batch_normalization_139: BatchNormalization
batch_normalization_140: BatchNormalization
batch_normalization_141: BatchNormalization
batch_normalization_142: BatchNormalization
activation_137: Activation
activation_138: Activation
activation_139: Activation
activation_140: Activation
color: Dense
length: Dense
width: Dense
angle: Dense

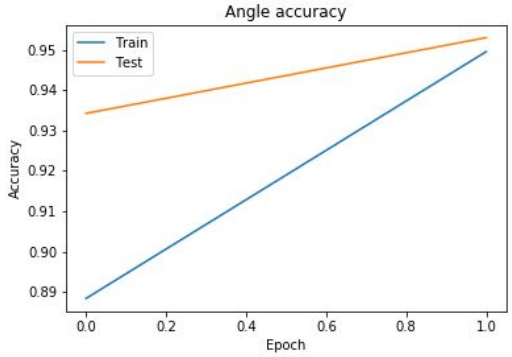
Learning Curves

Accuracy curves for output heads

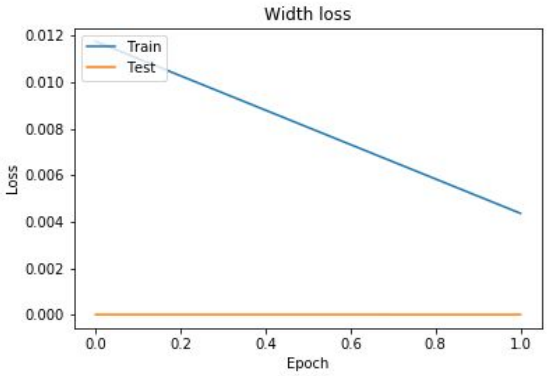


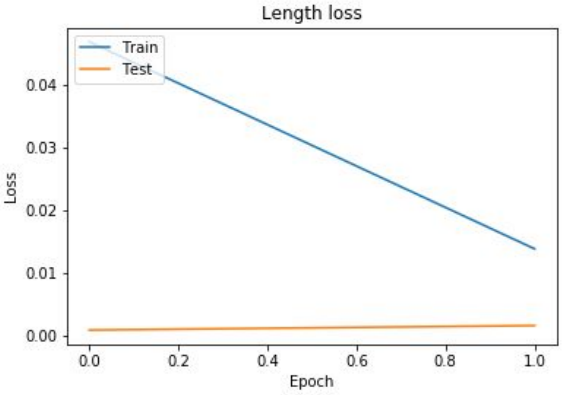


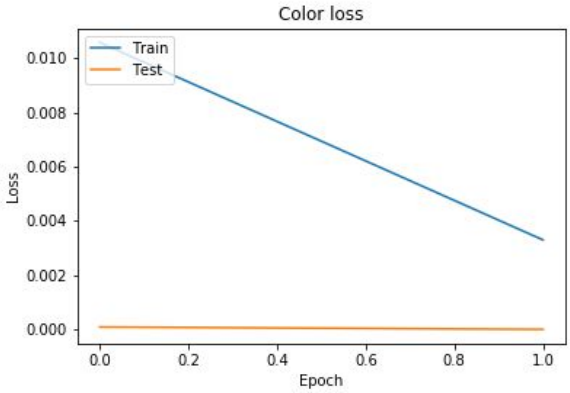


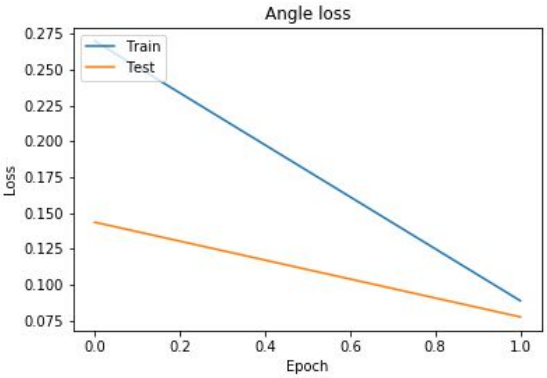


Loss curves for output heads

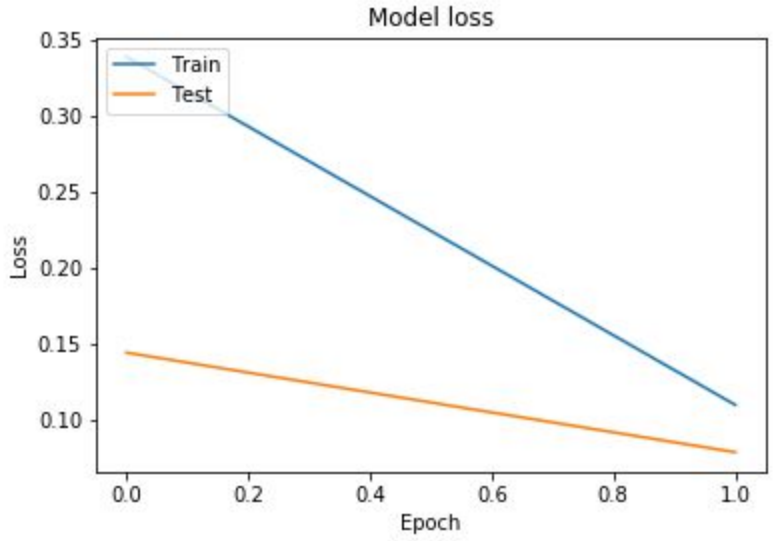


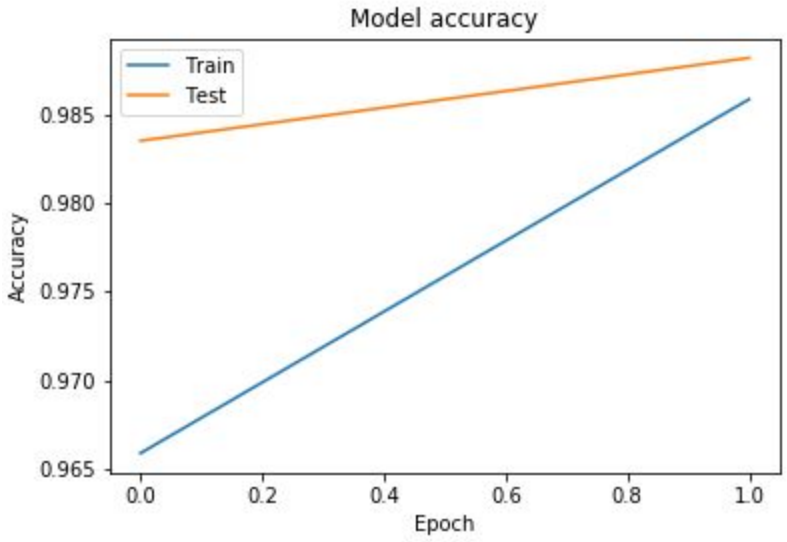






Learning curves for complete model





F-Scores

F1-scores for each output head :

- Color : 0.9631759867873927
- Length : 0.7723438586735037
- Width : 0.9951043689210557
- Angle : 0.9515091015077034

Final F1-score for model : **0.9205275**

Confusion Matrices

Confusion matrices for each output head :

- **Color :**

[[96240]

[706 8870]]

- **Length :**

[[96030]

[4165 5432]]

- **Width :**

[[94720]

[94 9634]]

- **Angle :**

```
[[1552 0    0    0    0    0    0    0    0    0   41]
 [   0 1629    0    0    0    0    0    0    0    0    0    0]
 [   0    0 1624    0    2    0    0    0    0    0    0    0]
 [   0    0    0 1608   17    0    2    0    0    0    0    0]
 [   0    0    0    0 1610    0    0    0    0    0    0    0]
 [   0    0    0    0    0  874  691    0    0    0    0    0]
 [   0    0    0    0    0  126 1484    0    0    0    0    0]
 [   0    0    0    0    0    0    4 1611   18    0    0    0]
 [   0    0    0    0    0    0    0    0 1497    0    0    0]
 [   0    0    0    0    0    0    0    0    0 1625    2    0]
 [   0    0    0    0    0    0    0    0    0    0 1617    0]
 [   0    0    0    0    0    0    0    0    0    0    0 1566]]
```

Variations Tried

- Increased the number of convolutional layers as well as the number of filters within the convolutional layer to improve the feature extraction.
- Reduced the number of max-pooling layers from 3 to 2 to maintain the granularity.
- Added the batch-normalization layers which also improved the accuracy of the model by reducing the amount by which the hidden unit values shift around (covariance shift).
- Reduced the network's architecture complexity to save the model from overfitting.
- Increased the number of hidden units in the classification layer of angle which improved the accuracy.

Inferences

- Increasing the model complexity may sometimes lead to the risk of overfitting.
- Training the model on increased number of epochs reduces the network validation accuracy by overfitting .
- Increasing the number of max-pooling layers can lead to the loss of features of the image.
- Increasing the number of units within a hidden layer helps to prevent underfitting.

- Accuracy on validation set :
    - Color : 100 %
    - Length : 100 %
    - Width : 100 %
    - Angle : 95.3 %
    - **Model : 98.8 %**