

DEEP LEARNING



Deep learning Report

General Preprocessing

- Reading and resizing all images (size=224)
- Apply normalization (eg, `"/255.0"`)
- Apply label encoding (eg, `" to_categorical"`)

General Techniques Used

- Different optimizers such as (Adam,SGD)
- Early stopping (avoid overfitting)
- L2 regularization (reduce overfitting)
- BatchNormalization (do normalization between layers, helps to speed training)
- Dropout (to reduce overfitting by randomly turning off part of the network off and let the rest of the network train)
- Bottleneck (to reduce number of parameters between layers)
- Data Augmentation (ImageDataGenerator)

About Building Complex models:

For models such as mobile net v2, vgg19 and inception models,

We have tried to build their architectures from scratch and loading then loading their weights whether from tensor hub or GitHub

To feed them to the models than make them unchangeable weights for training for our new data.

Mobile net v2 model

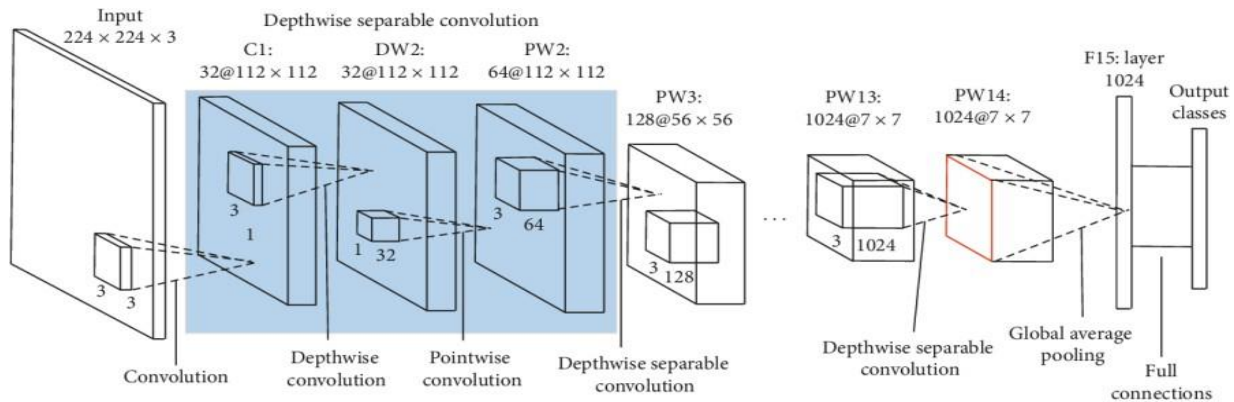


FIGURE 1: Architecture of MobileNet.

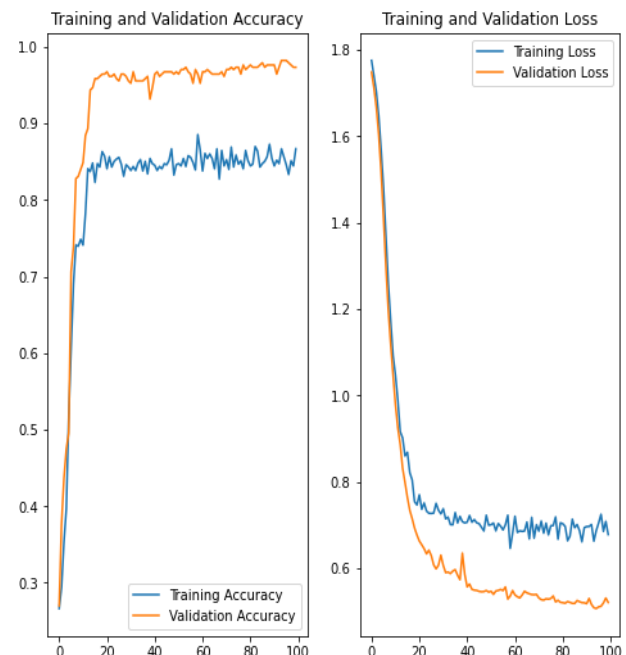
- Training:
As Batches:
loss: 0.6783 - accuracy: 0.8668
val_loss: 0.5210 - val_accuracy: 0.9733

As one batch on evaluation:
acc:0.994 , Val acc:0.973

- Training with Data Augmentation:
As Batches:
loss: 0.6916 - accuracy: 0.8490
val_loss: 0.5303 - val_accuracy: 0.9674

As one batch on evaluation:
acc:0.9947 , Val acc:0.9673

- we got about 92% score on Kaggle



VGG19 Model

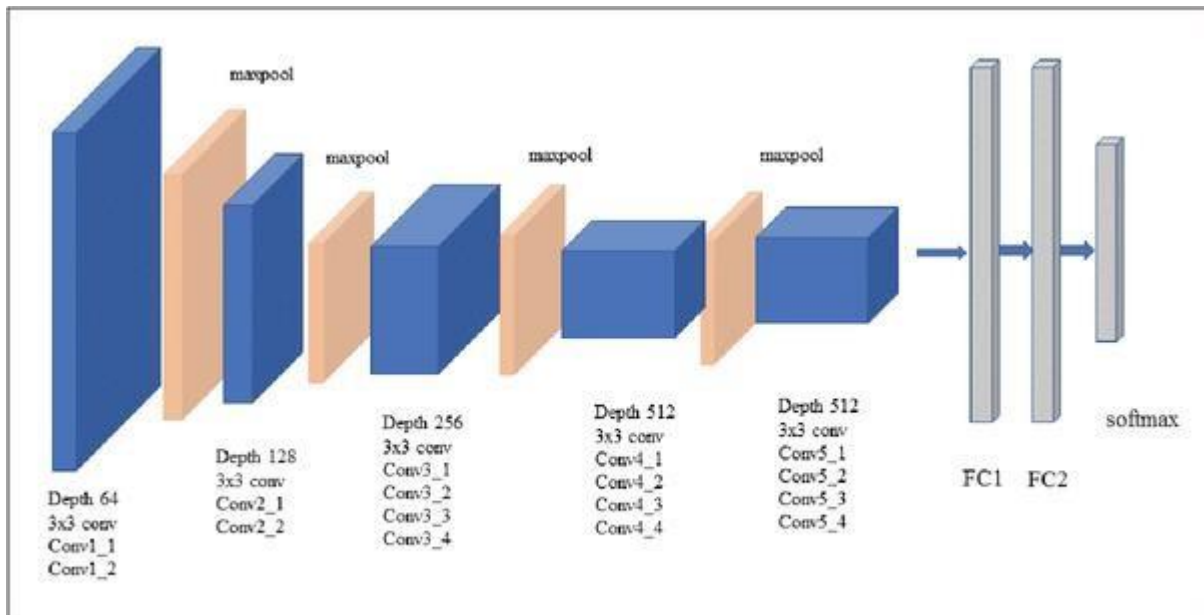
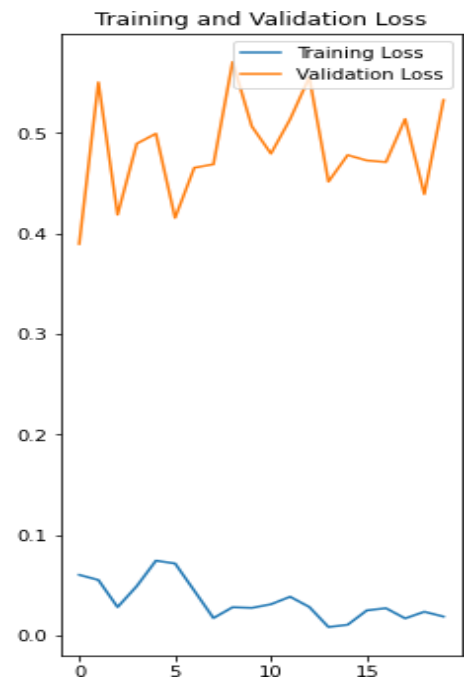
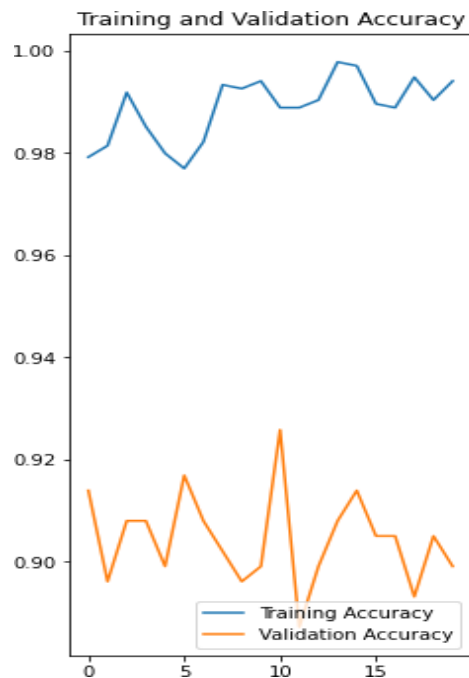


Fig. 3. VGG19 network architecture

- Training:
loss: 1.5752e-05 - accuracy: 1.0000
val_loss: 0.4037 - val_accuracy: 0.9199
- Training with Data Augmentation:
loss: 0.0190 - accuracy: 0.9940
val_loss: 0.5326 - val_accuracy: 0.8991

with these results,
we got about 85% score
on Kaggle



CNN Model

- Training:
loss: 1.0231 - accuracy: 0.9621
val_loss: 1.5686 - val_accuracy: 0.8487
- Training with Data Augmentation:
loss: 1.1228 - accuracy: 0.8966
val_loss: 1.2891 - val_accuracy: 0.8576

with these results, we get about 75% score on Kaggle

