

Ain Shams University
Faculty of Engineering
CESS – CHEP

Mohammed Ehab Elsaeed
16P8160

Selected Topics in Software
Applications (Neural Networks)

CSE440

Project Documentation

Contents

1. Running Scripts	3
2. Implemented Bonuses	3
Input image of any size.	3
Tensorboard.....	3
Docker File for creating the environment and auto running the Training and Inference Scripts.	5
3. Model Summary.....	5
4. Hyperparameters	6
5. Environment.....	6
6. Screenshots	7
Docker Build & Run	7
Docker Container Model Training.....	7
Docker Container Inference Using Preselected Sample	8
Docker Container Inference Using User-Input Image	9
Windows Model Training and Showing Random Images Inference	9
Windows Tensorboard.....	11
Windows Inference Using Preselected Sample	12
Windows Inference Using User-Input Image	13

1. Running Scripts

I have implemented two scripts located in the root directory of the project that would make running the project a one-click operation.

a. 16P8160_Docker.cmd

This script will:

1. Build the Docker image from the Docker file I developed (**dockerfiles/cpu.Dockerfile**)
2. Run the built Docker image as root and pass to it a bash script I developed (**code/16p8160_runscript.sh**).
3. The bash script passed will:
 - a. Run Train.py.
 - b. Run Inference.py using a preselected sample image of the digit 7.
 - c. Run Inference.py again allowing you to run Inference.py using your own image. Please provide the full path (For an image in the running Docker container ex. `"/home/trainingSet/3/img_7.jpg"`).

Please note that it is not possible to show an image in Docker. The only dependency needed to run the docker script is to have Docker installed on your machine.

b. 16P8160_Windows.cmd

This script will:

1. Run Train.py.
2. Run Inference.py using a preselected sample image of the digit 7.
3. Run Inference.py again allowing you to run Inference.py using your own image Please provide the full path (For an image in the file system).
4. Open your browser to localhost:6060 and run Tensorboard automatically.

To run this script, you have to setup your environment just like mine, please check chapter 5 of this document.

2. Implemented Bonuses

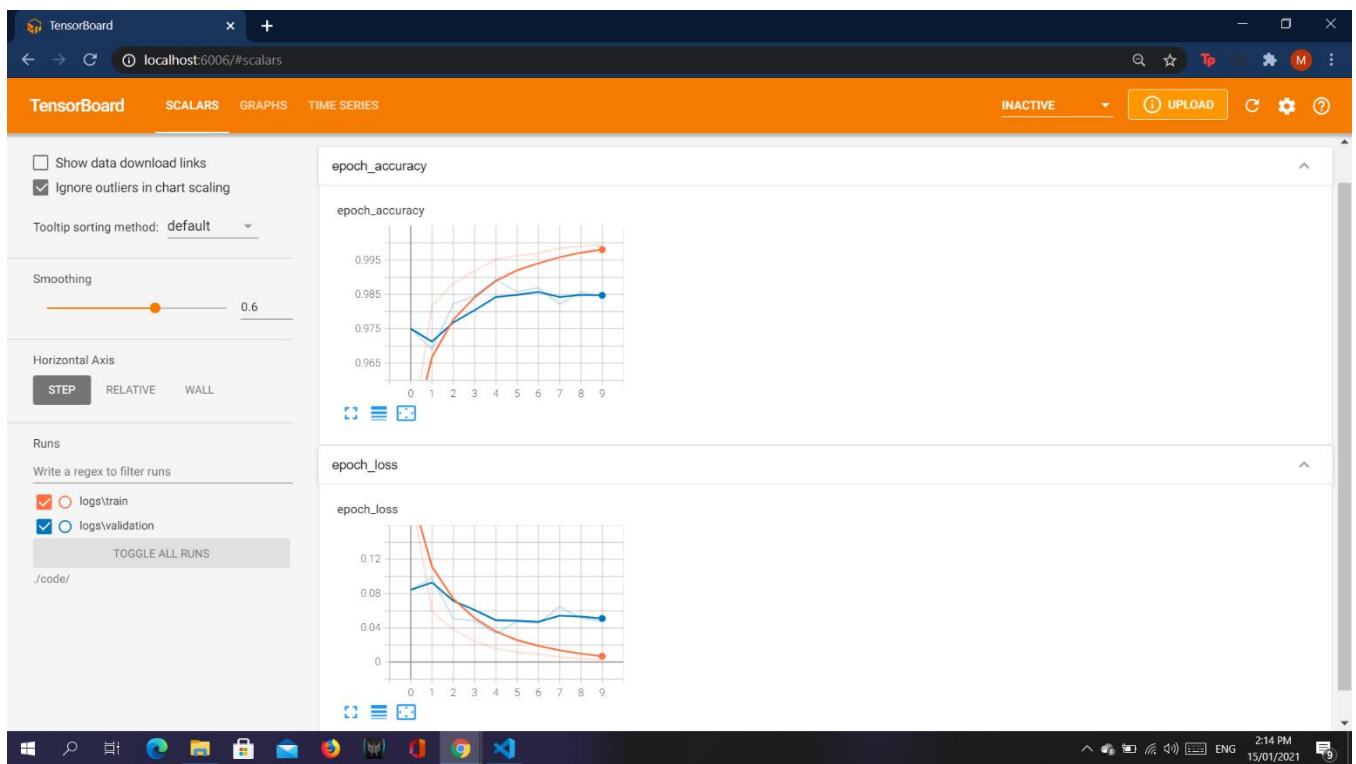
Input image of any size.

```
# Image resizing layer.Resize the batched image input to target height and width.
# The input should be a 4-D tensor
keras.layers.experimental.preprocessing.Resizing(
    28, 28, interpolation="bilinear"
),
# Output Shape: (n, 28, 28, 3), where n is the number of images fed to the model
```

Tensorboard.

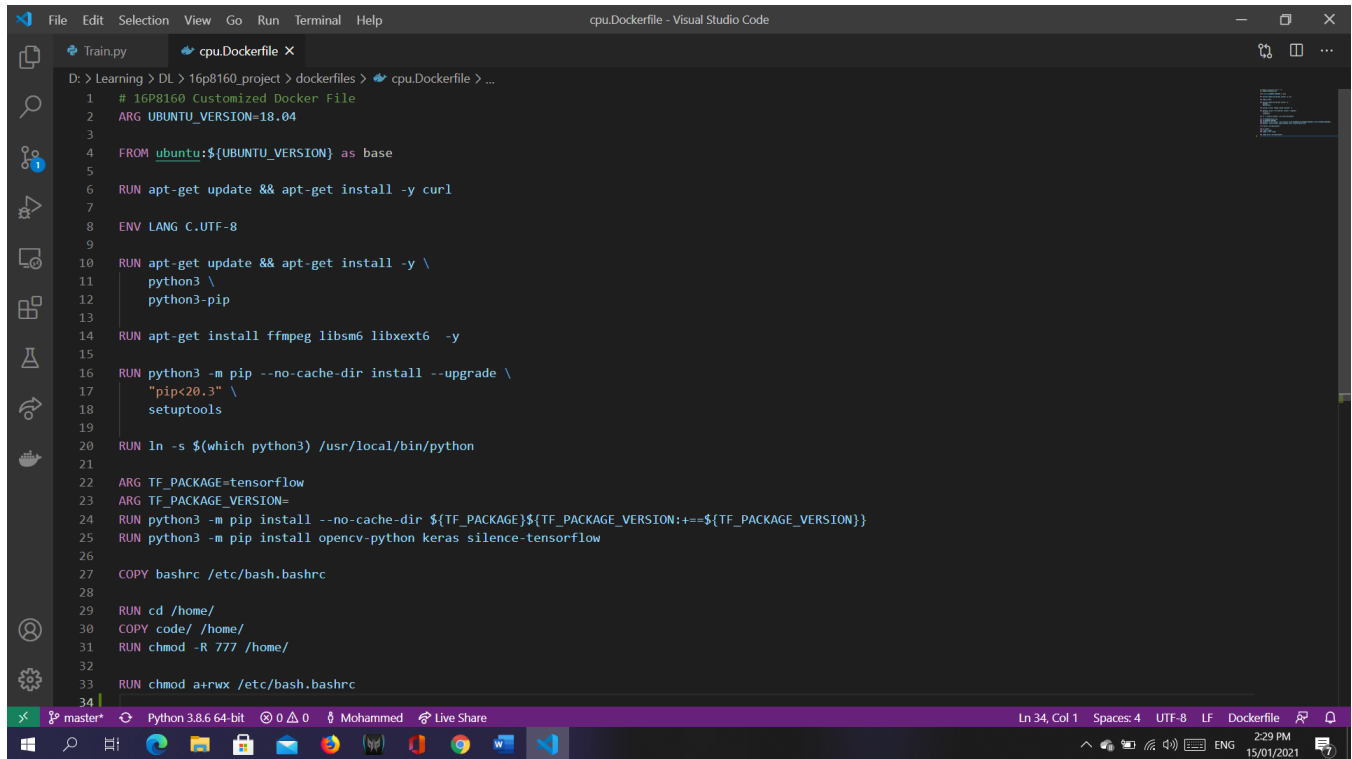
```
# BONUS1, tensorboard
tensorboard_callback = tf.keras.callbacks.TensorBoard(log_dir=f"{rootDir}/logs")
```

```
# training the model
model.fit(
    x_train,
    y_train_one_hot,
    # To fit in memory easily
    batch_size=128,
    # Through Trials, 10 epochs yielded acceptable accuracy and loss results
    epochs=10,
    # Using 98% of the data for training
    validation_split=0.02,
    # for BONUS1 Tensorboard
    callbacks=[tensorboard_callback]
)
```



Docker File for creating the environment and auto running the Training and Inference Scripts.

Docker file is at **dockerfiles/cpu.Dockerfile**



```
D: > Learning > DL > 16p8160_project > dockerfiles > cpu.Dockerfile > ...
1 # 16P8160 Customized Docker File
2 ARG UBUNTU_VERSION=18.04
3
4 FROM ubuntu:${UBUNTU_VERSION} as base
5
6 RUN apt-get update && apt-get install -y curl
7
8 ENV LANG C.UTF-8
9
10 RUN apt-get update && apt-get install -y \
11     python3 \
12     python3-pip
13
14 RUN apt-get install ffmpeg libsm6 libxext6 -y
15
16 RUN python3 -m pip --no-cache-dir install --upgrade \
17     "pip<20.3" \
18     setuptools
19
20 RUN ln -s $(which python3) /usr/local/bin/python
21
22 ARG TF_PACKAGE=tensorflow
23 ARG TF_PACKAGE_VERSION=
24 RUN python3 -m pip install --no-cache-dir ${TF_PACKAGE}${TF_PACKAGE_VERSION:+==${TF_PACKAGE_VERSION}}
25 RUN python3 -m pip install opencv-python keras silence-tensorflow
26
27 COPY bashrc /etc/bash.bashrc
28
29 RUN cd /home/
30 COPY code/ /home/
31 RUN chmod -R 777 /home/
32
33 RUN chmod a+rxw /etc/bash.bashrc
34
```

3. Model Summary

Layer	Input Shape	Output Shape	Number of Parameters
Resizing Interpolation Method: Bilinear	(n, any, any, 3)	(n, 28, 28, 3)	0
2D Convolution Filters: 25 Kernel Size: 3x3 Padding: valid Activation Function: Relu	(n, 28, 28, 3)	(n, 26, 26, 25)	700
2D Max pooling Size: 2x2 Strides: 2, 2 Padding: valid	(n, 26, 26, 25)	(n, 13, 13, 25)	0

Flattening	(n, 13, 13, 25)	(n, 4225)	0
Dense (Fully Connected) Units (Neurons): 250 Activation Function: Relu	(n, 4225)	(n, 250)	1056500
Dense (Fully Connected) Units (Neurons): 10 Activation Function: SoftMax	(n, 250)	(n, 10)	2510

4. Hyperparameters

Optimizer: ADAM

Learning Rate: 0.001

Epochs: 10

Batch Size: 128

Validation Split=0.02

5. Environment

OS: Windows 10 Version 10.0.18363 Build 18363

silence-tensorflow version 1.1.1

Tensorflow version 2.4

Keras version 2.4.3

Numpy version 1.19.4

opencv-python version 4.4.0.46

NVIDIA CUDA version 11.2

NVIDIA CUDNN version 8.0.5.39-windows-x64 for CUDA 11.1

6. Screenshots

Docker Build & Run

```
C:\Windows\system32\cmd.exe

D:\Learning\DL\16p8160_project>docker build -f ./dockerfiles/cpu.Dockerfile -t 16p8160 .
[*] Building 203.2s (19/19) FINISHED
-> [internal] load build definition from cpu.Dockerfile
-> [internal] load .dockerignore
-> [internal] load metadata for docker.io/library/ubuntu:18.04
-> [auth] library/ubuntu:pull token for registry-1.docker.io
-> [internal] load build context
-> [internal] transferring context: 2.36MB
-> [1/13] FROM docker.io/library/ubuntu:18.04@sha256:fd25e706f3dea2a5ff705d8c3353cf37f80307798f3e360a13e9185040
-> CACHED [2/13] RUN apt-get update && apt-get install -y curl
-> CACHED [3/13] RUN apt-get update && apt-get install -y python3 python3-pip
-> CACHED [4/13] RUN apt-get install ffmpeg libsm6 libxext6 -y
-> CACHED [5/13] RUN python3 -m pip --no-cache-dir install --upgrade "pip<20.3" setuptools
-> CACHED [6/13] RUN ln -s $(which python3) /usr/local/bin/python
-> CACHED [7/13] RUN python3 -m pip install --no-cache-dir tensorflow
-> [8/13] RUN python3 -m pip install opencv-python keras silence-tensorflow
-> [9/13] COPY bashrc /etc/bash.bashrc
-> [10/13] RUN cd /home/
-> [11/13] COPY code/ /home/
-> [12/13] RUN chmod -R 777 /home/
-> [13/13] RUN chmod a+rx /etc/bash.bashrc
-> exporting to image
-> exporting layers
-> writing image sha256:3c214474bf6f748e5ab0a18acbf81952ef65c76aed06f0a7cfc120a58844519
-> naming to docker.io/library/16p8160

D:\Learning\DL\16p8160_project>docker run -t -i --privileged 16p8160 bash /home/16p8160_runscript.sh
|||||

||||| - WILL TEST TRAINING NOW - |||||

|||||
Images Array Shape: (42000, 28, 28, 3) Labels Array Shape: (42000, 1)
Images Array ndim: 4 Labels Array ndim: 2
Images Array dtype: uint8 Labels Array dtype: uint8
```

Docker Container Model Training

```
Select C:\Windows\system32\cmd.exe

-> exporting layers
-> writing image sha256:3c214474bf6f748e5ab0a18acbf81952ef65c76aed06f0a7cfc120a58844519
-> naming to docker.io/library/16p8160

D:\Learning\DL\16p8160_project>docker run -t -i --privileged 16p8160 bash /home/16p8160_runscript.sh
|||||

||||| - WILL TEST TRAINING NOW - |||||

|||||
Images Array Shape: (42000, 28, 28, 3) Labels Array Shape: (42000, 1)
Images Array ndim: 4 Labels Array ndim: 2
Images Array dtype: uint8 Labels Array dtype: uint8
Images Array dtype Name: uint8 Labels Array dtype Name: uint8
DATA LOADED FROM FILE SYSTEM
Epoch 1/10
322/322 [=====] - 4s 13ms/step - loss: 0.4190 - accuracy: 0.8749 - val_loss: 0.0907 - val_accuracy: 0.9762
Epoch 2/10
322/322 [=====] - 4s 12ms/step - loss: 0.0643 - accuracy: 0.9808 - val_loss: 0.0738 - val_accuracy: 0.9798
Epoch 3/10
322/322 [=====] - 4s 12ms/step - loss: 0.0375 - accuracy: 0.9884 - val_loss: 0.0730 - val_accuracy: 0.9821
Epoch 4/10
322/322 [=====] - 4s 12ms/step - loss: 0.0234 - accuracy: 0.9934 - val_loss: 0.0611 - val_accuracy: 0.9869
Epoch 5/10
322/322 [=====] - 4s 12ms/step - loss: 0.0170 - accuracy: 0.9947 - val_loss: 0.0626 - val_accuracy: 0.9869
Epoch 6/10
322/322 [=====] - 4s 12ms/step - loss: 0.0109 - accuracy: 0.9970 - val_loss: 0.0663 - val_accuracy: 0.9833
Epoch 7/10
322/322 [=====] - 4s 12ms/step - loss: 0.0085 - accuracy: 0.9978 - val_loss: 0.0766 - val_accuracy: 0.9821
Epoch 8/10
322/322 [=====] - 4s 13ms/step - loss: 0.0060 - accuracy: 0.9984 - val_loss: 0.0888 - val_accuracy: 0.9869
Epoch 9/10
322/322 [=====] - 4s 13ms/step - loss: 0.0033 - accuracy: 0.9993 - val_loss: 0.0655 - val_accuracy: 0.9857
Epoch 10/10
322/322 [=====] - 4s 13ms/step - loss: 0.0043 - accuracy: 0.9988 - val_loss: 0.0720 - val_accuracy: 0.9857
Model: "sequential"
```

```
Select C:\Windows\system32\cmd.exe

|||||
Images Array Shape: (42000, 28, 28, 3) Labels Array Shape: (42000, 1)
Images Array ndim: 4 Labels Array ndim: 2
Images Array dtype: uint8 Labels Array dtype: uint8
Images Array dtype Name: uint8 Labels Array dtype Name: uint8
DATA LOADED FROM FILE SYSTEM
Epoch 1/10
322/322 [=====] - 4s 13ms/step - loss: 0.4190 - accuracy: 0.8749 - val_loss: 0.0907 - val_accuracy: 0.9762
Epoch 2/10
322/322 [=====] - 4s 12ms/step - loss: 0.0643 - accuracy: 0.9808 - val_loss: 0.0738 - val_accuracy: 0.9798
Epoch 3/10
322/322 [=====] - 4s 12ms/step - loss: 0.0375 - accuracy: 0.9884 - val_loss: 0.0730 - val_accuracy: 0.9821
Epoch 4/10
322/322 [=====] - 4s 12ms/step - loss: 0.0234 - accuracy: 0.9934 - val_loss: 0.0611 - val_accuracy: 0.9869
Epoch 5/10
322/322 [=====] - 4s 12ms/step - loss: 0.0170 - accuracy: 0.9947 - val_loss: 0.0626 - val_accuracy: 0.9869
Epoch 6/10
322/322 [=====] - 4s 12ms/step - loss: 0.0109 - accuracy: 0.9970 - val_loss: 0.0663 - val_accuracy: 0.9833
Epoch 7/10
322/322 [=====] - 4s 12ms/step - loss: 0.0085 - accuracy: 0.9978 - val_loss: 0.0766 - val_accuracy: 0.9821
Epoch 8/10
322/322 [=====] - 4s 13ms/step - loss: 0.0060 - accuracy: 0.9984 - val_loss: 0.0888 - val_accuracy: 0.9869
Epoch 9/10
322/322 [=====] - 4s 13ms/step - loss: 0.0033 - accuracy: 0.9993 - val_loss: 0.0655 - val_accuracy: 0.9857
Epoch 10/10
322/322 [=====] - 4s 13ms/step - loss: 0.0043 - accuracy: 0.9988 - val_loss: 0.0720 - val_accuracy: 0.9857
Model: "sequential"

Layer (type)                Output Shape              Param #
-----
resizing (Resizing)          (None, 28, 28, 3)         0
conv2d (Conv2D)              (None, 26, 26, 25)        700
max_pooling2d (MaxPooling2D) (None, 13, 13, 25)        0
Flatten (Flatten)            (None, 4225)              0
dense (Dense)                 (None, 250)               1056500
dense_1 (Dense)              (None, 10)                2510
-----
Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0

Showing Images only works on OS with GUI
|||||
```

Docker Container Inference Using Preselected Sample

```
Select C:\Windows\system32\cmd.exe

Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0

Showing Images only works on OS with GUI
|||||

||||| - WILL TEST INFERENCE NOW USING MY SAMPLE - |||||

|||||
Model: "sequential"

Layer (type)                Output Shape              Param #
-----
resizing (Resizing)          (None, 28, 28, 3)         0
conv2d (Conv2D)              (None, 26, 26, 25)        700
max_pooling2d (MaxPooling2D) (None, 13, 13, 25)        0
Flatten (Flatten)            (None, 4225)              0
dense (Dense)                 (None, 250)               1056500
dense_1 (Dense)              (None, 10)                2510
-----
Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0

['Inference.py', '/home/trainingSet/7/img_10015.jpg']
(1, 28, 28, 3)
[[0. 0. 0. 0. 0. 0. 0. 1. 0. 0.]]
Predicted Class: [7]
|||||
```


Docker Container Inference Using User-Input Image

```
Select C:\Windows\system32\cmd.exe
[[0. 0. 0. 0. 0. 0. 0. 1. 0. 0.]]
Predicted Class: [7]
|||||

||||| - TEST INFERENCE USING YOUR SAMPLE - |||||

|||||
Model: "sequential"
Layer (type)                Output Shape                Param #
=====
resizing (Resizing)         (None, 28, 28, 3)           0
conv2d (Conv2D)             (None, 26, 26, 25)          700
max_pooling2d (MaxPooling2D) (None, 13, 13, 25)          0
Flatten (Flatten)           (None, 4225)                0
dense (Dense)               (None, 250)                 1056500
dense_1 (Dense)             (None, 10)                  2510
=====
Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0

['Inference.py']
Enter The Path to the image you want to predict
/home/trainingSet/7/img_10015.jpg
(1, 28, 28, 3)
[[0. 0. 0. 0. 0. 0. 0. 1. 0. 0.]]
Predicted Class: [7]

D:\Learning\DL\16p8160_project>pause
Press any key to continue . . .
```

Windows Model Training and Showing Random Images Inference


```
C:\Windows\system32\cmd.exe

Waiting for 0 seconds, press a key to continue ...

D:\Learning\DL\16p8160_project>python ./code/Train.py
Images Array Shape: (42000, 28, 28, 3) Labels Array Shape: (42000, 1)
Images Array ndim: 4 Labels Array ndim: 2
Images Array dtype: uint8 Labels Array dtype: uint8
Images Array dtype Name: uint8 Labels Array dtype Name: uint8
DATA LOADED FROM FILE SYSTEM
Epoch 1/10
322/322 [=====] - 4s 7ms/step - loss: 0.4168 - accuracy: 0.8736 - val_loss: 0.0966 - val_accuracy: 0.9667
Epoch 2/10
322/322 [=====] - 1s 4ms/step - loss: 0.0684 - accuracy: 0.9797 - val_loss: 0.0712 - val_accuracy: 0.9798
Epoch 3/10
322/322 [=====] - loss: 0.0387 - accuracy: 0.9879 - val_loss: 0.0669 - val_accuracy: 0.9786
Epoch 4/10
322/322 [=====] - 1s 4ms/step - loss: 0.0263 - accuracy: 0.9926 - val_loss: 0.0730 - val_accuracy: 0.9774
Epoch 5/10
322/322 [=====] - 1s 4ms/step - loss: 0.0183 - accuracy: 0.9949 - val_loss: 0.0768 - val_accuracy: 0.9750
Epoch 6/10
322/322 [=====] - 1s 4ms/step - loss: 0.0107 - accuracy: 0.9971 - val_loss: 0.0756 - val_accuracy: 0.9857
Epoch 7/10
322/322 [=====] - 1s 4ms/step - loss: 0.0076 - accuracy: 0.9977 - val_loss: 0.0631 - val_accuracy: 0.9869
Epoch 8/10
322/322 [=====] - 1s 4ms/step - loss: 0.0066 - accuracy: 0.9981 - val_loss: 0.0651 - val_accuracy: 0.9833
Epoch 9/10
322/322 [=====] - 1s 4ms/step - loss: 0.0058 - accuracy: 0.9983 - val_loss: 0.0576 - val_accuracy: 0.9881
Epoch 10/10
322/322 [=====] - 1s 4ms/step - loss: 0.0031 - accuracy: 0.9991 - val_loss: 0.0657 - val_accuracy: 0.9845
Model: "sequential"
Layer (type)                Output Shape                Param #
=====
resizing (Resizing)         (None, 28, 28, 3)           0
conv2d (Conv2D)             (None, 26, 26, 25)          700
max_pooling2d (MaxPooling2D) (None, 13, 13, 25)          0
Flatten (Flatten)           (None, 4225)                0
dense (Dense)               (None, 250)                 1056500
dense_1 (Dense)             (None, 10)                  2510
=====
Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0
```

Sample

Y: [5], Y_pred: [5]



CA\Windows\system32\cmd.exe

Waiting for 0 seconds, press a key to continue ...

D:\Learning\DL\l6p8160_project>python ./code/Train.py
Images Array Shape: (42000, 28, 28, 3) Labels Array Shape: (42000, 1)
Images Array ndim: 4 Labels Array ndim: 2
Images Array dtype: uint8 Labels Array dtype: uint8
Images Array dtype Name: uint8 Labels Array dtype Name: uint8
DATA LOADED FROM FILE SYSTEM
Epoch 1/10
322/322 [=====] - 4s 7ms/step - loss: 0.4168 - accuracy: 0.736 - val_loss: 0.0966 - val_accuracy: 0.9667
Epoch 2/10
322/322 [=====] - 1s 4ms/step - loss: 0.0684 - accuracy: 0.9797 - val_loss: 0.0712 - val_accuracy: 0.9798
Epoch 3/10
322/322 [=====] - 1s 4ms/step - loss: 0.0387 - accuracy: 0.9879 - val_loss: 0.0669 - val_accuracy: 0.9786
Epoch 4/10
322/322 [=====] - 1s 4ms/step - loss: 0.0263 - accuracy: 0.9926 - val_loss: 0.0730 - val_accuracy: 0.9774
Epoch 5/10
322/322 [=====] - 1s 4ms/step - loss: 0.0183 - accuracy: 0.9949 - val_loss: 0.0768 - val_accuracy: 0.9750
Epoch 6/10
322/322 [=====] - 1s 4ms/step - loss: 0.0107 - accuracy: 0.9971 - val_loss: 0.0756 - val_accuracy: 0.9857
Epoch 7/10
322/322 [=====] - 1s 4ms/step - loss: 0.0076 - accuracy: 0.9977 - val_loss: 0.0631 - val_accuracy: 0.9869
Epoch 8/10
322/322 [=====] - 1s 4ms/step - loss: 0.0066 - accuracy: 0.9981 - val_loss: 0.0651 - val_accuracy: 0.9833
Epoch 9/10
322/322 [=====] - 1s 4ms/step - loss: 0.0058 - accuracy: 0.9983 - val_loss: 0.0576 - val_accuracy: 0.9881
Epoch 10/10
322/322 [=====] - 1s 4ms/step - loss: 0.0031 - accuracy: 0.9991 - val_loss: 0.0657 - val_accuracy: 0.9845
Model: "sequential"

Layer (type)	Output Shape	Param #
resizing (Resizing)	(None, 28, 28, 3)	0
conv2d (Conv2D)	(None, 26, 26, 25)	700
max_pooling2d (MaxPooling2D)	(None, 13, 13, 25)	0
Flatten (Flatten)	(None, 4225)	0
dense (Dense)	(None, 250)	1056500
dense_1 (Dense)	(None, 10)	2510

Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0

Sample

Y: [2], Y_pred: [2]

2

Windows taskbar: 2:02 PM 15/01/2021

CA\Windows\system32\cmd.exe

Waiting for 0 seconds, press a key to continue ...

D:\Learning\DL\l6p8160_project>python ./code/Train.py
Images Array Shape: (42000, 28, 28, 3) Labels Array Shape: (42000, 1)
Images Array ndim: 4 Labels Array ndim: 2
Images Array dtype: uint8 Labels Array dtype: uint8
Images Array dtype Name: uint8 Labels Array dtype Name: uint8
DATA LOADED FROM FILE SYSTEM
Epoch 1/10
322/322 [=====] - 4s 7ms/step - loss: 0.4168 - accuracy: 0.736 - val_loss: 0.0966 - val_accuracy: 0.9667
Epoch 2/10
322/322 [=====] - 1s 4ms/step - loss: 0.0684 - accuracy: 0.9797 - val_loss: 0.0712 - val_accuracy: 0.9798
Epoch 3/10
322/322 [=====] - 1s 4ms/step - loss: 0.0387 - accuracy: 0.9879 - val_loss: 0.0669 - val_accuracy: 0.9786
Epoch 4/10
322/322 [=====] - 1s 4ms/step - loss: 0.0263 - accuracy: 0.9926 - val_loss: 0.0730 - val_accuracy: 0.9774
Epoch 5/10
322/322 [=====] - 1s 4ms/step - loss: 0.0183 - accuracy: 0.9949 - val_loss: 0.0768 - val_accuracy: 0.9750
Epoch 6/10
322/322 [=====] - 1s 4ms/step - loss: 0.0107 - accuracy: 0.9971 - val_loss: 0.0756 - val_accuracy: 0.9857
Epoch 7/10
322/322 [=====] - 1s 4ms/step - loss: 0.0076 - accuracy: 0.9977 - val_loss: 0.0631 - val_accuracy: 0.9869
Epoch 8/10
322/322 [=====] - 1s 4ms/step - loss: 0.0066 - accuracy: 0.9981 - val_loss: 0.0651 - val_accuracy: 0.9833
Epoch 9/10
322/322 [=====] - 1s 4ms/step - loss: 0.0058 - accuracy: 0.9983 - val_loss: 0.0576 - val_accuracy: 0.9881
Epoch 10/10
322/322 [=====] - 1s 4ms/step - loss: 0.0031 - accuracy: 0.9991 - val_loss: 0.0657 - val_accuracy: 0.9845
Model: "sequential"

Layer (type)	Output Shape	Param #
resizing (Resizing)	(None, 28, 28, 3)	0
conv2d (Conv2D)	(None, 26, 26, 25)	700
max_pooling2d (MaxPooling2D)	(None, 13, 13, 25)	0
Flatten (Flatten)	(None, 4225)	0
dense (Dense)	(None, 250)	1056500
dense_1 (Dense)	(None, 10)	2510

Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0

Sample

Y: [7], Y_pred: [7]

7

Windows taskbar: 2:02 PM 15/01/2021

```
C:\Windows\system32\cmd.exe

Waiting for 0 seconds, press a key to continue ...

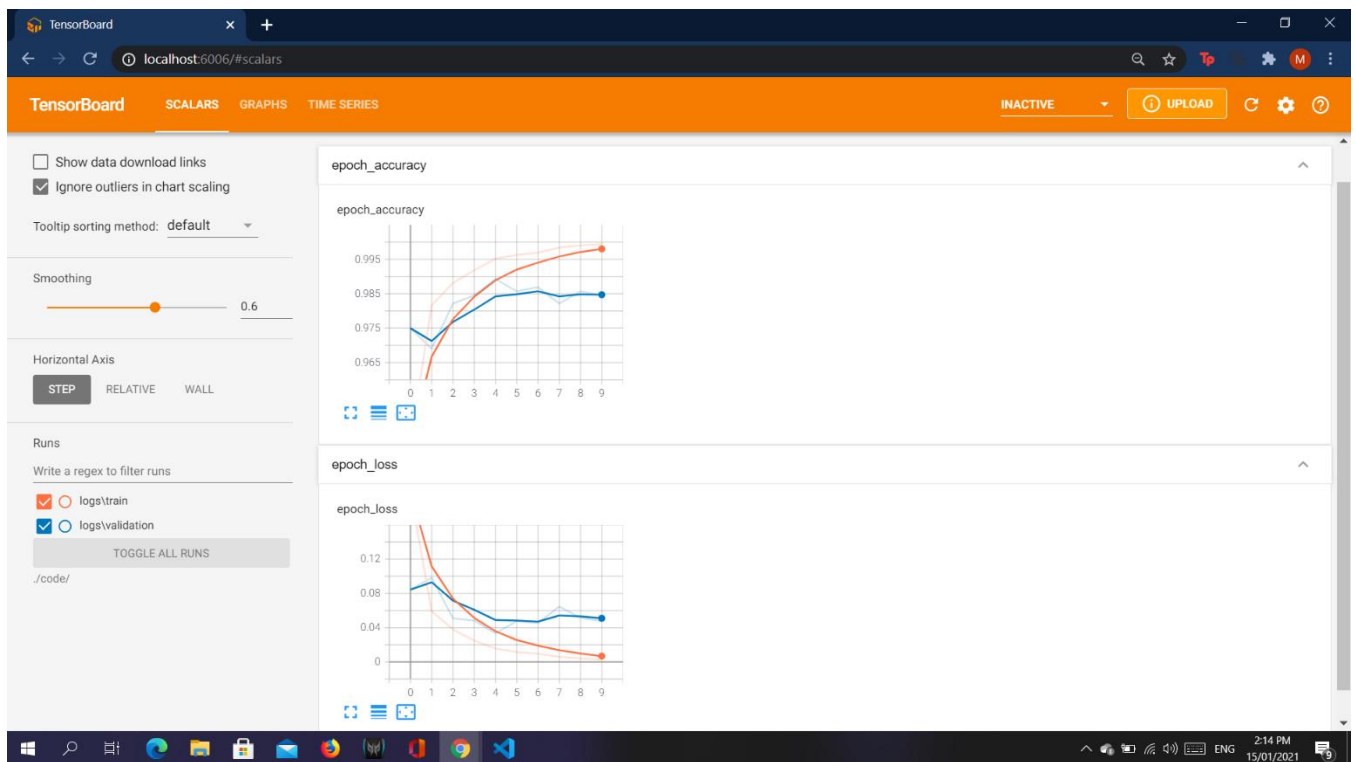
D:\Learning\DL\16p8160_project>python ./code/Train.py
Images Array Shape: (42000, 28, 28, 3) Labels Array Shape: (42000, 1)
Images Array ndim: 4 Labels Array ndim: 2
Images Array dtype: uint8 Labels Array dtype: uint8
DATA LOADED FROM FILE SYSTEM
Epoch 1/10
322/322 [=====] - 4s 7ms/step - loss: 0.4168 - accuracy: 0.736 - val_loss: 0.0966 - val_accuracy: 0.9667
Epoch 2/10
322/322 [=====] - 1s 4ms/step - loss: 0.0684 - accuracy: 0.9797 - val_loss: 0.0712 - val_accuracy: 0.9798
Epoch 3/10
322/322 [=====] - 1s 4ms/step - loss: 0.0387 - accuracy: 0.9879 - val_loss: 0.0669 - val_accuracy: 0.9786
Epoch 4/10
322/322 [=====] - 1s 4ms/step - loss: 0.0263 - accuracy: 0.9926 - val_loss: 0.0730 - val_accuracy: 0.9774
Epoch 5/10
322/322 [=====] - 1s 4ms/step - loss: 0.0183 - accuracy: 0.9949 - val_loss: 0.0768 - val_accuracy: 0.9750
Epoch 6/10
322/322 [=====] - 1s 4ms/step - loss: 0.0107 - accuracy: 0.9971 - val_loss: 0.0756 - val_accuracy: 0.9857
Epoch 7/10
322/322 [=====] - 1s 4ms/step - loss: 0.0076 - accuracy: 0.9977 - val_loss: 0.0631 - val_accuracy: 0.9869
Epoch 8/10
322/322 [=====] - 1s 4ms/step - loss: 0.0066 - accuracy: 0.9981 - val_loss: 0.0651 - val_accuracy: 0.9833
Epoch 9/10
322/322 [=====] - 1s 4ms/step - loss: 0.0058 - accuracy: 0.9983 - val_loss: 0.0576 - val_accuracy: 0.9881
Epoch 10/10
322/322 [=====] - 1s 4ms/step - loss: 0.0031 - accuracy: 0.9991 - val_loss: 0.0657 - val_accuracy: 0.9845
Model: "sequential"

Layer (type)                Output Shape              Param #
-----
resizing (Resizing)          (None, 28, 28, 3)         0
conv2d (Conv2D)              (None, 26, 26, 25)        700
max_pooling2d (MaxPooling2D) (None, 13, 13, 25)        0
Flatten (Flatten)            (None, 4225)              0
dense (Dense)                (None, 250)               1056500
dense_1 (Dense)              (None, 10)                2510
-----
Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0
```

Sample

Y: [0], Y_pred: [0]

Windows Tensorboard



Windows Inference Using User-Input Image

```
C:\Windows\system32\cmd.exe
""
|||----- TEST INFERENCE USING YOUR SAMPLE -----|||
""
Waiting for 1 seconds, press a key to continue ...
D:\Learning\DL\v16p8160_project>python ./code/inference.py
Model: "sequential"

Layer (type)                 Output Shape              Param #
=====
resizing (Resizing)          (None, 28, 28, 3)         0
conv2d (Conv2D)               (None, 26, 26, 25)        700
max_pooling2d (MaxPooling2D) (None, 13, 13, 25)        0
flatten (Flatten)             (None, 4225)              0
dense (Dense)                 (None, 250)               1056500
dense_1 (Dense)               (None, 10)                2510
Total params: 1,059,710
Trainable params: 1,059,710
Non-trainable params: 0

['./code/inference.py']
Enter The Path to the image you want to predict
D:\Learning\DL\v16p8160_project\code\trainingSet\3\img_7.jpg
(1, 28, 28, 3)
[[0. 0. 0. 1. 0. 0. 0. 0. 0.]]
Predicted Class: [3]
|||----- TEST INFERENCE USING YOUR SAMPLE -----|||
```