In [112…

```
#Verify the CPU setup
import tensorflow as tf; print(tf.reduce_sum(tf.random.normal([1000, 1000])))
```

```
tf.Tensor(85.178375, shape=(), dtype=float32)
```

In [113…

```
#verify GPU setup
import tensorflow as tf; print(tf.config.list_physical_devices('GPU'))
```

```
[]
```

1. Install Dependencies and Setup

In [114…

```
#import tensorflow flow and opencv for the model
# TensorFlow  used for building, training, and deploying deep learning models, i
# OpenCV (Open Source Computer Vision Library) is an open-source computer vision
# Computer vision tasks like object detection, face recognition.
!pip install tensorflow opencv-python matplotlib
```

```
Requirement already satisfied: tensorflow in c:\users\farhan\appdata\local\progra
ms\python\python312\lib\site-packages (2.17.0)
Requirement already satisfied: opencv-python in c:\users\farhan\appdata\local\pro
grams\python\python312\lib\site-packages (4.10.0.84)
Requirement already satisfied: matplotlib in c:\users\farhan\appdata\local\progra
ms\python\python312\lib\site-packages (3.9.0)
Requirement already satisfied: tensorflow-intel==2.17.0 in c:\users\farhan\appdat
a\local\programs\python\python312\lib\site-packages (from tensorflow) (2.17.0)
Requirement already satisfied: absl-py>=1.0.0 in c:\users\farhan\appdata\local\pr
ograms\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tensorf
low) (2.1.0)
Requirement already satisfied: astunparse>=1.6.0 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tens
orflow) (1.6.3)
Requirement already satisfied: flatbuffers>=24.3.25 in c:\users\farhan\appdata\lo
cal\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->t
ensorflow) (24.3.25)
Requirement already satisfied: gast!=0.5.0,!=0.5.1,!=0.5.2,>=0.2.1 in c:\users\fa
rhan\appdata\local\programs\python\python312\lib\site-packages (from tensorflow-i
ntel==2.17.0->tensorflow) (0.6.0)
Requirement already satisfied: google-pasta>=0.1.1 in c:\users\farhan\appdata\loc
al\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->te
nsorflow) (0.2.0)
Requirement already satisfied: h5py>=3.10.0 in c:\users\farhan\appdata\local\prog
rams\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tensorflo
w) (3.11.0)
Requirement already satisfied: libclang>=13.0.0 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tens
orflow) (18.1.1)
Requirement already satisfied: ml-dtypes<0.5.0,>=0.3.1 in c:\users\farhan\appdata
\local\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0
->tensorflow) (0.4.0)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tens
orflow) (3.3.0)
Requirement already satisfied: packaging in c:\users\farhan\appdata\local\program
s\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tensorflow)
(24.1)
Requirement already satisfied: protobuf!=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.2
1.4,!=4.21.5,<5.0.0dev,>=3.20.3 in c:\users\farhan\appdata\local\programs\python
\python312\lib\site-packages (from tensorflow-intel==2.17.0->tensorflow) (4.25.3)
Requirement already satisfied: requests<3,>=2.21.0 in c:\users\farhan\appdata\loc
al\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->te
nsorflow) (2.32.3)
Requirement already satisfied: setuptools in c:\users\farhan\appdata\local\progra
ms\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tensorflow)
(71.1.0)
Requirement already satisfied: six>=1.12.0 in c:\users\farhan\appdata\local\progr
ams\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tensorflo
w) (1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tens
orflow) (2.4.0)
Requirement already satisfied: typing-extensions>=3.6.6 in c:\users\farhan\appdat
a\local\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.
0->tensorflow) (4.12.2)
Requirement already satisfied: wrapt>=1.11.0 in c:\users\farhan\appdata\local\pro
grams\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tensorfl
ow) (1.16.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in c:\users\farhan\appdata\loc
```

```
al\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->te
nsorflow) (1.65.1)
Requirement already satisfied: tensorboard<2.18,>=2.17 in c:\users\farhan\appdata
\local\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0
->tensorflow) (2.17.0)
Requirement already satisfied: keras>=3.2.0 in c:\users\farhan\appdata\local\prog
rams\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->tensorflo
w) (3.4.1)
Requirement already satisfied: numpy<2.0.0,>=1.26.0 in c:\users\farhan\appdata\lo
cal\programs\python\python312\lib\site-packages (from tensorflow-intel==2.17.0->t
ensorflow) (1.26.4)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from matplotlib) (1.2.1)
Requirement already satisfied: cycler>=0.10 in c:\users\farhan\appdata\local\prog
rams\python\python312\lib\site-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from matplotlib) (4.53.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from matplotlib) (1.4.5)
Requirement already satisfied: pillow>=8 in c:\users\farhan\appdata\local\program
s\python\python312\lib\site-packages (from matplotlib) (10.3.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from matplotlib) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\farhan\appdata\lo
cal\programs\python\python312\lib\site-packages (from matplotlib) (2.9.0.post0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in c:\users\farhan\appdata\loca
l\programs\python\python312\lib\site-packages (from astunparse>=1.6.0->tensorflow
-intel==2.17.0->tensorflow) (0.43.0)
Requirement already satisfied: rich in c:\users\farhan\appdata\local\programs\pyt
hon\python312\lib\site-packages (from keras>=3.2.0->tensorflow-intel==2.17.0->ten
sorflow) (13.7.1)
Requirement already satisfied: namex in c:\users\farhan\appdata\local\programs\py
thon\python312\lib\site-packages (from keras>=3.2.0->tensorflow-intel==2.17.0->te
nsorflow) (0.0.8)
Requirement already satisfied: optree in c:\users\farhan\appdata\local\programs\p
ython\python312\lib\site-packages (from keras>=3.2.0->tensorflow-intel==2.17.0->t
ensorflow) (0.12.1)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\farhan\appdat
a\local\programs\python\python312\lib\site-packages (from requests<3,>=2.21.0->te
nsorflow-intel==2.17.0->tensorflow) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\farhan\appdata\local\prog
rams\python\python312\lib\site-packages (from requests<3,>=2.21.0->tensorflow-int
el==2.17.0->tensorflow) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\farhan\appdata\loca
l\programs\python\python312\lib\site-packages (from requests<3,>=2.21.0->tensorfl
ow-intel==2.17.0->tensorflow) (2.2.2)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\farhan\appdata\loca
l\programs\python\python312\lib\site-packages (from requests<3,>=2.21.0->tensorfl
ow-intel==2.17.0->tensorflow) (2024.6.2)
Requirement already satisfied: markdown>=2.6.8 in c:\users\farhan\appdata\local\p
rograms\python\python312\lib\site-packages (from tensorboard<2.18,>=2.17->tensorf
low-intel==2.17.0->tensorflow) (3.6)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in c:\users
\farhan\appdata\local\programs\python\python312\lib\site-packages (from tensorboa
rd<2.18,>=2.17->tensorflow-intel==2.17.0->tensorflow) (0.7.2)
Requirement already satisfied: werkzeug>=1.0.1 in c:\users\farhan\appdata\local\p
rograms\python\python312\lib\site-packages (from tensorboard<2.18,>=2.17->tensorf
low-intel==2.17.0->tensorflow) (3.0.3)
Requirement already satisfied: MarkupSafe>=2.1.1 in c:\users\farhan\appdata\local
\programs\python\python312\lib\site-packages (from werkzeug>=1.0.1->tensorboard<
```

```
2.18,>=2.17->tensorflow-intel==2.17.0->tensorflow) (2.1.5)
Requirement already satisfied: markdown-it-py>=2.2.0 in c:\users\farhan\appdata\l
ocal\programs\python\python312\lib\site-packages (from rich->keras>=3.2.0->tensor
flow-intel==2.17.0->tensorflow) (3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in c:\users\farhan\appdata
\local\programs\python\python312\lib\site-packages (from rich->keras>=3.2.0->tens
orflow-intel==2.17.0->tensorflow) (2.18.0)
Requirement already satisfied: mdurl~=0.1 in c:\users\farhan\appdata\local\progra
ms\python\python312\lib\site-packages (from markdown-it-py>=2.2.0->rich->keras>=
3.2.0->tensorflow-intel==2.17.0->tensorflow) (0.1.2)
```

In [115…

```
!pip list
```

```
Package                   Version
------------------------- --------------
absl-py                   2.1.0
altair                    5.3.0
anyio                     4.4.0
argon2-cffi               23.1.0
argon2-cffi-bindings      21.2.0
arrow                     1.3.0
asttokens                 2.4.1
astunparse                1.6.3
async-lru                 2.0.4
attrs                     23.2.0
Babel                     2.15.0
beautifulsoup4            4.12.3
bleach                    6.1.0
blinker                   1.8.2
cachetools                5.3.3
certifi                   2024.6.2
cffi                      1.16.0
charset-normalizer        3.3.2
click                     8.1.7
colorama                  0.4.6
comm                      0.2.2
contourpy                 1.2.1
cycler                    0.12.1
debugpy                   1.8.2
decorator                 5.1.1
defusedxml                0.7.1
distlib                   0.3.8
executing                 2.0.1
fastjsonschema            2.20.0
filelock                  3.15.4
flatbuffers               24.3.25
fonttools                 4.53.0
fqdn                      1.5.1
gast                      0.6.0
gitdb                     4.0.11
GitPython                 3.1.43
google-pasta              0.2.0
grpcio                    1.65.1
h11                       0.14.0
h5py                      3.11.0
httpcore                  1.0.5
httpx                     0.27.0
idna                      3.7
ipykernel                 6.29.5
ipython                   8.26.0
ipywidgets                8.1.3
isoduration               20.11.0
jedi                      0.19.1
Jinja2                    3.1.4
json5                     0.9.25
jsonpointer               3.0.0
jsonschema                4.22.0
jsonschema-specifications 2023.12.1
jupyter                   1.0.0
jupyter_client            8.6.2
jupyter-console           6.6.3
jupyter_core              5.7.2
jupyter-events            0.10.0
```

```
jupyter-lsp               2.2.5
jupyter_server            2.14.2
jupyter_server_terminals  0.5.3
jupyterlab                4.2.4
jupyterlab_pygments       0.3.0
jupyterlab_server         2.27.3
jupyterlab_widgets        3.0.11
keras                     3.4.1
kiwisolver                1.4.5
libclang                  18.1.1
Markdown                  3.6
markdown-it-py            3.0.0
MarkupSafe                2.1.5
matplotlib                3.9.0
matplotlib-inline         0.1.7
mdurl                     0.1.2
mistune                   3.0.2
ml-dtypes                 0.4.0
namex                     0.0.8
nbclient                  0.10.0
nbconvert                 7.16.4
nbformat                  5.10.4
nest-asyncio              1.6.0
notebook                  7.2.1
notebook_shim             0.2.4
numpy                     1.26.4
opencv-python             4.10.0.84
opt-einsum                3.3.0
optree                    0.12.1
overrides                 7.7.0
packaging                 24.1
pandas                    2.2.2
pandocfilters             1.5.1
parso                     0.8.4
pillow                    10.3.0
pip                       24.1.2
platformdirs              4.2.2
prometheus_client         0.20.0
prompt_toolkit            3.0.47
protobuf                  4.25.3
psutil                    6.0.0
pure_eval                 0.2.3
pyarrow                   16.1.0
pycparser                 2.22
pydeck                    0.9.1
Pygments                  2.18.0
pyparsing                 3.1.2
python-dateutil           2.9.0.post0
python-json-logger        2.0.7
pytz                      2024.1
pywin32                   306
pywinpty                  2.0.13
PyYAML                    6.0.1
pyzmq                     26.0.3
qtconsole                 5.5.2
QtPy                      2.4.1
referencing               0.35.1
requests                  2.32.3
rfc3339-validator         0.1.4
rfc3986-validator         0.1.1
```

```
rich                      13.7.1
rpds-py                   0.18.1
Send2Trash                1.8.3
setuptools                71.1.0
six                       1.16.0
smmap                     5.0.1
sniffio                   1.3.1
soupsieve                 2.5
stack-data                0.6.3
streamlit                 1.36.0
tenacity                  8.4.2
tensorboard               2.17.0
tensorboard-data-server   0.7.2
tensorflow                2.17.0
tensorflow-intel          2.17.0
termcolor                 2.4.0
terminado                 0.18.1
tinycss2                  1.3.0
toml                      0.10.2
toolz                     0.12.1
tornado                   6.4.1
traitlets                 5.14.3
types-python-dateutil     2.9.0.20240316
typing_extensions         4.12.2
tzdata                    2024.1
uri-template              1.3.0
urllib3                   2.2.2
virtualenv                20.26.3
watchdog                  4.0.1
wcwidth                   0.2.13
webcolors                 24.6.0
webencodings              0.5.1
websocket-client          1.8.0
Werkzeug                  3.0.3
wheel                     0.43.0
widgetsnbextension        4.0.11
wrapt                     1.16.0
```

In [116…

```python
# for building, training, and deploying deep learning models,
import tensorflow as tf
# for intracting operating system  and manuplating files and directories
import os
# for plotting various graph
from matplotlib import pyplot as plt
```

In [117…

```python
# Avoid OOM errors by setting GPU Memory Consumption Growth
gpus = tf.config.experimental.list_physical_devices('GPU')
for gpu in gpus:
    tf.config.experimental.set_memory_growth(gpu, True)
```

2. Remove dodgy images

In [118…

```python
# Working with image and video processing functions
import cv2
# determines the type of image contained in a file or byte stream.
import imghdr
```

In [119…

```
# set directory where data is saved
data_dir = 'deeplearning'
```

In [120…

```
#list all the directory present in working folder
# join the directory to access the images in one line
os.listdir(os.path.join(data_dir,'puppies'))
```

Out[120…

```
['.ipynb_checkpoints',
 '0000655001540319655.jpg',
 '03-june_puppies.jpg',
 '0992_01_4x3L.jpg',
 '1VPra30HqtU1iPTkFaM2fow.png',
 '2021-11-24_Singleton_Puppy_Syndrome_One_Puppy_Litter.jpg',
 '210601-main-pups-canine-companions-for-independence-ew-113p.jpg',
 '36b4ef433ac9c2cf9d34e54aa3a38340.jpg',
 '536413-gettyimages-1207430371-458be6339696aa7d3539e96b06bd08c9.jpg',
 '811hvGypcPL.png',
 'aabb34_461127eb3a14469b86f7ad942f931a79mv2.jpg',
 'AdobeStock274064877_360x316.jpeg',
 'AdobeStock_472713009.jpeg',
 'akc-pupill-hdr-new-new.jpg',
 'Alaskan-Malamute-puppy-laying-down-outdoors.jpg',
 'are-antler-chews-safe-for-puppies-747450.jpg',
 'Aussiechon20puppy.png',
 'benefits-of-getting-a-puppy-600x400.jpg',
 'Cavalier-King-Charles-Spaniel-puppies-sitting-together-on-the-couch.jpeg',
 'cHJpdmF0ZS9sci9pbWFnZXMvd2Vic2l0ZS8yMDIyLTA1L25zODIzMC1pbWFnZS5qcGc.jpg',
 'CompletePuppyGuide--TrainingResources2CShoppingList2CandMore7CLivingMinnaly.j
pg',
 'Corgi-puppy-1-scaled.jpg',
 'cute_black_puppy_pooping_outside_480x480.jpg',
 'dog-puppy-on-garden-royalty-free-image-1586966191.jpg',
 'Dog_food_for_puppies.jpg',
 'DSC7970-e1568922693978.jpg',
 'finding-homes-for-puppies.jpg',
 'five-cheapest-sources-of-puppies.jpg',
 'getting-a-puppy.jpg',
 'GettyImages-512366437-e1688677726208.jpg',
 'Golde33443.jpg',
 'Golden-retriever-puppy-chewing-on-toy_Photology1971_Shutterstock-600x401.jp
g',
 'happy-beagle-cropped.jpg',
 'how-many-puppies-samoyed-mother-large-litter-1.jpg',
 'how-to-identify-puppy-mill-open-graph_image.jpg',
 'husky-puppy-on-dog-walk.jpg',
 'image.jpeg',
 'image10.jpeg',
 'image11.jpeg',
 'image12.jpeg',
 'image13.jpeg',
 'image14.jpeg',
 'image15.jpeg',
 'image16.jpeg',
 'image17.jpeg',
 'image18.jpeg',
 'image19.jpeg',
 'image2.jpeg',
 'image20.jpeg',
 'image21.jpeg',
 'image22.jpeg',
 'image23.jpeg',
 'image24.jpeg',
 'image25.jpeg',
 'image26.jpeg',
 'image27.jpeg',
 'image28.jpeg',
 'image29.jpeg',
```

```
                    'image3.jpeg',
                    'image30.jpeg',
                    'image31.jpeg',
                    'image32.jpeg',
                    'image33.jpeg',
                    'image34.jpeg',
                    'image35.jpeg',
                    'image36.jpeg',
                    'image37.jpeg',
                    'image38.jpeg',
                    'image39.jpeg',
                    'image4.jpeg',
                    'image40.jpeg',
                    'image41.jpeg',
                    'image42.jpeg',
                    'image43.jpeg',
                    'image44.jpeg',
                    'image45.jpeg',
                    'image46.jpeg',
                    'image47.jpeg',
                    'image48.jpeg',
                    'image49.jpeg',
                    'image5.jpeg',
                    'image50.jpeg',
                    'image51.jpeg',
                    'image52.jpeg',
                    'image53.jpeg',
                    'image54.jpeg',
                    'image55.jpeg',
                    'image56.jpeg',
                    'image57.jpeg',
                    'image6.jpeg',
                    'image7.jpeg',
                    'image8.jpeg',
                    'image9.jpeg',
                    'images.jpg',
                    'images10.jpg',
                    'images100.jpg',
                    'images101.jpg',
                    'images102.jpg',
                    'images103.jpg',
                    'images104.jpg',
                    'images105.jpg',
                    'images106.jpg',
                    'images107.jpg',
                    'images108.jpg',
                    'images109.jpg',
                    'images11.jpg',
                    'images110.jpg',
                    'images111.jpg',
                    'images112.jpg',
                    'images113.jpg',
                    'images114.jpg',
                    'images115.jpg',
                    'images116.jpg',
                    'images117.jpg',
                    'images118.jpg',
                    'images119.jpg',
                    'images12.jpg',
                    'images120.jpg',
```

```
                    'images121.jpg',
                    'images122.jpg',
                    'images123.jpg',
                    'images124.jpg',
                    'images125.jpg',
                    'images126.jpg',
                    'images127.jpg',
                    'images128.jpg',
                    'images129.jpg',
                    'images13.jpg',
                    'images130.jpg',
                    'images131.jpg',
                    'images132.jpg',
                    'images133.jpg',
                    'images134.jpg',
                    'images135.jpg',
                    'images136.jpg',
                    'images137.jpg',
                    'images138.jpg',
                    'images139.jpg',
                    'images14.jpg',
                    'images140.jpg',
                    'images141.jpg',
                    'images142.jpg',
                    'images143.jpg',
                    'images144.jpg',
                    'images145.jpg',
                    'images146.jpg',
                    'images147.jpg',
                    'images148.jpg',
                    'images149.jpg',
                    'images15.jpg',
                    'images150.jpg',
                    'images151.jpg',
                    'images152.jpg',
                    'images153.jpg',
                    'images154.jpg',
                    'images155.jpg',
                    'images156.jpg',
                    'images157.jpg',
                    'images158.jpg',
                    'images159.jpg',
                    'images16.jpg',
                    'images160.jpg',
                    'images161.jpg',
                    'images162.jpg',
                    'images163.jpg',
                    'images164.jpg',
                    'images165.jpg',
                    'images166.jpg',
                    'images167.jpg',
                    'images168.jpg',
                    'images169.jpg',
                    'images17.jpg',
                    'images170.jpg',
                    'images171.jpg',
                    'images172.jpg',
                    'images173.jpg',
                    'images174.jpg',
                    'images175.jpg',
```

```
                 'images176.jpg',
                 'images177.jpg',
                 'images178.jpg',
                 'images179.jpg',
                 'images18.jpg',
                 'images180.jpg',
                 'images181.jpg',
                 'images182.jpg',
                 'images183.jpg',
                 'images184.jpg',
                 'images185.jpg',
                 'images186.jpg',
                 'images187.jpg',
                 'images188.jpg',
                 'images189.jpg',
                 'images19.jpg',
                 'images190.jpg',
                 'images191.jpg',
                 'images192.jpg',
                 'images193.jpg',
                 'images194.jpg',
                 'images195.jpg',
                 'images196.jpg',
                 'images197.jpg',
                 'images198.jpg',
                 'images199.jpg',
                 'images2.jpg',
                 'images20.jpg',
                 'images200.jpg',
                 'images201.jpg',
                 'images202.jpg',
                 'images203.jpg',
                 'images204.jpg',
                 'images205.jpg',
                 'images206.jpg',
                 'images207.jpg',
                 'images208.jpg',
                 'images209.jpg',
                 'images21.jpg',
                 'images210.jpg',
                 'images211.jpg',
                 'images212.jpg',
                 'images213.jpg',
                 'images214.jpg',
                 'images215.jpg',
                 'images216.jpg',
                 'images217.jpg',
                 'images218.jpg',
                 'images219.jpg',
                 'images22.jpg',
                 'images220.jpg',
                 'images221.jpg',
                 'images222.jpg',
                 'images223.jpg',
                 'images224.jpg',
                 'images225.jpg',
                 'images226.jpg',
                 'images227.jpg',
                 'images228.jpg',
                 'images229.jpg',
```

```
                 'images23.jpg',
                 'images230.jpg',
                 'images231.jpg',
                 'images232.jpg',
                 'images233.jpg',
                 'images234.jpg',
                 'images235.jpg',
                 'images236.jpg',
                 'images237.jpg',
                 'images238.jpg',
                 'images239.jpg',
                 'images24.jpg',
                 'images240.jpg',
                 'images241.jpg',
                 'images242.jpg',
                 'images243.jpg',
                 'images244.jpg',
                 'images245.jpg',
                 'images246.jpg',
                 'images247.jpg',
                 'images248.jpg',
                 'images249.jpg',
                 'images25.jpg',
                 'images250.jpg',
                 'images251.jpg',
                 'images252.jpg',
                 'images253.jpg',
                 'images254.jpg',
                 'images255.jpg',
                 'images256.jpg',
                 'images257.jpg',
                 'images258.jpg',
                 'images259.jpg',
                 'images26.jpg',
                 'images260.jpg',
                 'images261.jpg',
                 'images262.jpg',
                 'images263.jpg',
                 'images264.jpg',
                 'images265.jpg',
                 'images266.jpg',
                 'images267.jpg',
                 'images268.jpg',
                 'images269.jpg',
                 'images27.jpg',
                 'images270.jpg',
                 'images271.jpg',
                 'images272.jpg',
                 'images273.jpg',
                 'images274.jpg',
                 'images275.jpg',
                 'images276.jpg',
                 'images277.jpg',
                 'images278.jpg',
                 'images279.jpg',
                 'images28.jpg',
                 'images280.jpg',
                 'images281.jpg',
                 'images282.jpg',
                 'images283.jpg',
```

```
                 'images284.jpg',
                 'images285.jpg',
                 'images286.jpg',
                 'images287.jpg',
                 'images288.jpg',
                 'images289.jpg',
                 'images29.jpg',
                 'images290.jpg',
                 'images291.jpg',
                 'images292.jpg',
                 'images293.jpg',
                 'images294.jpg',
                 'images295.jpg',
                 'images296.jpg',
                 'images297.jpg',
                 'images298.jpg',
                 'images299.jpg',
                 'images3.jpg',
                 'images30.jpg',
                 'images300.jpg',
                 'images301.jpg',
                 'images302.jpg',
                 'images303.jpg',
                 'images304.jpg',
                 'images305.jpg',
                 'images306.jpg',
                 'images307.jpg',
                 'images308.jpg',
                 'images309.jpg',
                 'images31.jpg',
                 'images310.jpg',
                 'images311.jpg',
                 'images312.jpg',
                 'images313.jpg',
                 'images314.jpg',
                 'images315.jpg',
                 'images316.jpg',
                 'images317.jpg',
                 'images318.jpg',
                 'images319.jpg',
                 'images32.jpg',
                 'images320.jpg',
                 'images321.jpg',
                 'images322.jpg',
                 'images323.jpg',
                 'images324.jpg',
                 'images325.jpg',
                 'images326.jpg',
                 'images327.jpg',
                 'images328.jpg',
                 'images329.jpg',
                 'images33.jpg',
                 'images330.jpg',
                 'images331.jpg',
                 'images332.jpg',
                 'images333.jpg',
                 'images334.jpg',
                 'images335.jpg',
                 'images336.jpg',
                 'images337.jpg',
```

```
                 'images338.jpg',
                 'images339.jpg',
                 'images34.jpg',
                 'images340.jpg',
                 'images341.jpg',
                 'images342.jpg',
                 'images343.jpg',
                 'images344.jpg',
                 'images345.jpg',
                 'images346.jpg',
                 'images347.jpg',
                 'images348.jpg',
                 'images349.jpg',
                 'images35.jpg',
                 'images350.jpg',
                 'images351.jpg',
                 'images352.jpg',
                 'images353.jpg',
                 'images354.jpg',
                 'images355.jpg',
                 'images356.jpg',
                 'images357.jpg',
                 'images358.jpg',
                 'images359.jpg',
                 'images36.jpg',
                 'images360.jpg',
                 'images361.jpg',
                 'images362.jpg',
                 'images363.jpg',
                 'images37.jpg',
                 'images38.jpg',
                 'images39.jpg',
                 'images4.jpg',
                 'images40.jpg',
                 'images41.jpg',
                 'images42.jpg',
                 'images43.jpg',
                 'images44.jpg',
                 'images45.jpg',
                 'images46.jpg',
                 'images47.jpg',
                 'images48.jpg',
                 'images49.jpg',
                 'images5.jpg',
                 'images50.jpg',
                 'images51.jpg',
                 'images52.jpg',
                 'images53.jpg',
                 'images54.jpg',
                 'images55.jpg',
                 'images56.jpg',
                 'images57.jpg',
                 'images58.jpg',
                 'images59.jpg',
                 'images6.jpg',
                 'images60.jpg',
                 'images61.jpg',
                 'images62.jpg',
                 'images63.jpg',
                 'images64.jpg',
```

```
'images65.jpg',
'images66.jpg',
'images67.jpg',
'images68.jpg',
'images69.jpg',
'images7.jpg',
'images70.jpg',
'images71.jpg',
'images72.jpg',
'images73.jpg',
'images74.jpg',
'images75.jpg',
'images76.jpg',
'images77.jpg',
'images78.jpg',
'images79.jpg',
'images8.jpg',
'images80.jpg',
'images81.jpg',
'images82.jpg',
'images83.jpg',
'images84.jpg',
'images85.jpg',
'images86.jpg',
'images87.jpg',
'images88.jpg',
'images89.jpg',
'images9.jpg',
'images90.jpg',
'images91.jpg',
'images92.jpg',
'images93.jpg',
'images94.jpg',
'images95.jpg',
'images96.jpg',
'images97.jpg',
'images98.jpg',
'images99.jpg',
'IMG_1368-d789352.jpeg',
'istock-511313058.jpg',
'Labpuppy.jpg',
'labrador-puppy.jpg',
'litter-of-puppies-lying-together-in-green-grass-031324.jpg',
'nAk0IIkia4IAggZg00k0p1caring-for-those-new-puppies.jpg.jpg',
'newborn-puppies-bathroom-1.jpg',
'newborn-puppy.jpg',
'pbdBG-white.jpg',
'Puppies-are-like-newborns.jpg',
'puppies-are-the-best-models-omica-photography-fb26.png',
'puppies.jpg',
'puppy-0-2-months-physical-milestones-1240x640.jpg',
'puppy-1679276175142.webp',
'Puppy-Breeder-Referral-v3.png',
'puppy-class-3.jpg',
'Puppy-Day-compress_4e31bb4d-74a5-4960-af97-9e6bb7756a81.jpg',
'puppy-facts.jpg',
'puppy-feeding-schedule.jpg',
'puppy-life-book-1_3x2.jpg',
'puppy-socialization.jpg',
'puppy-vaccinations.jpg',
```

```
 'PuppyZiggyAtHome-e1590163382501.jpeg',
 'puppy_quiz_header_update_2024_1140.png',
 'r-d-smith-mwzVasRlDpc-unsplash.jpg',
 'Relleno-6968_resource.jpg-546904.jpg',
 'samoyed-labrador-mix-puppy-walking-on-the-grass_Abramova-Kseniya_Shutterstock
-600x403.jpeg',
 'sddefault.jpg',
 'shelter-dog-cropped-1.jpg',
 'short-coated-tan-puppy-stockpack-unsplash-scaled-e1698756703855.jpg',
 'shutterstock_336435884.jpg',
 'ten-reasons-why-you-shouldnt-get-a-puppy.jpg',
 'terrier-puppy-running-on-grass-outside.jpg',
 'VIER20PFOTEN_2019-03-15_001-2886x1999-1920x1330.jpg',
 'When-Can-My-Puppy-Go-Outside-1200x900.jpg',
 'when-do-puppies-start-walking_800x800.jpg',
 'ying-nature-playful-beauty-generated-by-artificial-intelligence_188544-84973.
jpg']
```

In [121…

```
# set desired extension in which we want to keep our images
image_exts = ['jpeg','jpg', 'bmp', 'png']
```

In [122…

```
#create a loop for checking all the directories present in the working folder
for image_class in os.listdir(data_dir):
    print(image_class)
```

```
kitten
puppies
```

In [123…

```
#acessing any image just for checking its acessible or not
img = cv2.imread(os.path.join('deeplearning','Puppies','Alaskan-Malamute-puppy-l
img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

#Display the image
plt.imshow(img_rgb)
plt.axis('on')  # Optional: Hide axes
plt.show()
```

In [124…

```
import cv2
import matplotlib.pyplot as plt

# Load the image
img = cv2.imread('Alaskan-Malamute-puppy-laying-down-outdoors.jpg')

# Check if the image was loaded correctly
if img is None:
    print("Error: Image not loaded. Check the file path.")
else:
    # Convert the image from BGR to RGB format
    img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

    # Display the image
    plt.imshow(img_rgb)
    plt.axis('off')  # Optional: Hide axes
    plt.show()
```

```
Error: Image not loaded. Check the file path.
```

In [125…

```
# Loop through each class directory in the data directory
for image_class in os.listdir(data_dir):
    # Loop through each image file in the current class directory
    for image in os.listdir(os.path.join(data_dir, image_class)):
        # Construct the full path to the current image file
        image_path = os.path.join(data_dir, image_class, image)
        try:
            # Read the image file using OpenCV
            img = cv2.imread(image_path)
            # Determine the type of the image (e.g., jpeg, png) using imghdr
            tip = imghdr.what(image_path)
            # Check if the image type is not in the list of allowed extensions
            if tip not in image_exts:
                # Print a message indicating the image type is not allowed
                print('Image not in ext list {}'.format(image_path))
                # Remove the image file from the filesystem
                os.remove(image_path)
        except Exception as e:
            # Print a message indicating there was an issue with the image file
            print('Issue with image {}'.format(image_path))
            # Optionally remove the image file (currently commented out)
            # os.remove(image_path)
```

```
Issue with image deeplearning\kitten\.ipynb_checkpoints
Issue with image deeplearning\puppies\.ipynb_checkpoints
```

3. Load Data

In [126…

```
# Import the numpy library, which provides support for large, multi-dimensional
# along with a collection of mathematical functions to operate on these arrays
import numpy as np

# Import the pyplot module from the matplotlib library, which provides a MATLAB-
# for creating static, animated, and interactive visualizations in Python
from matplotlib import pyplot as plt
```

In [127…

```
# Load images from the specified directory ('deeplearning') and create a dataset
# This utility function from TensorFlow's Keras API scans the directory, identif
```

```
# and organizes the images into a dataset suitable for training machine learning
data = tf.keras.utils.image_dataset_from_directory('deeplearning')
```

```
Found 676 files belonging to 2 classes.
```

In [128…

```
data
```

Out[128…

```
<_PrefetchDataset element_spec=(TensorSpec(shape=(None, 256, 256, 3), dtype=tf.
float32, name=None), TensorSpec(shape=(None,), dtype=tf.int32, name=None))>
```

In [133…

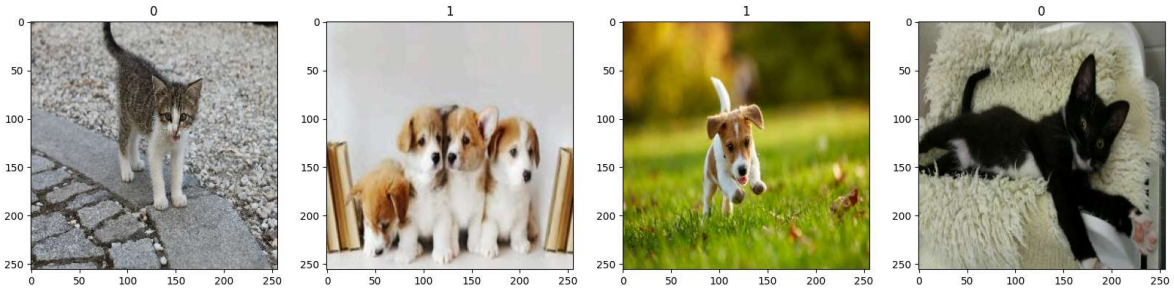
```
# Convert the TensorFlow dataset into a Numpy iterator.
# This allows you to iterate over the dataset and access the data in the form of
# which can be useful for further processing or inspection.
data_iterator = data.as_numpy_iterator()
```

In [134…

```
# Retrieve the next batch of data from the Numpy iterator.
# allowing you to access and process one batch of data at a time.
batch = data_iterator.next()
```

In [135…

```
# Create a figure with a grid of subplots (1 row and 4 columns) and set the over
fig, ax = plt.subplots(ncols=4, figsize=(20,20))

# Loop over the first four images in the batch.
for idx, img in enumerate(batch[0][:4]):
    # Display the current image in the corresponding subplot.
    ax[idx].imshow(img.astype(int))

    # Set the title of the subplot to the label of the current image.
    # The label is accessed from the batch's labels array (batch[1]) using the c
    ax[idx].title.set_text(batch[1][idx])
```



In [136…

```
#class 0=kitten
#class 1=puppies
batch[1]
```

Out[136…

```
array([0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 0, 1, 1])
```

4. Scale Data

In [137…

```
# Normalize the images in the dataset by dividing each pixel value by 255.
# The lambda function applies this transformation to each image (x), while keepi
data = data.map(lambda x, y: (x / 255, y))
```

In [138…

```
# Convert the normalized TensorFlow dataset into a Numpy iterator.
# This allows you to iterate over the normalized dataset and access the data in
scaled_iterator = data.as_numpy_iterator()
```

In [139…
```
# Retrieve the next batch of data from the scaled Numpy iterator.
batch = scaled_iterator.next()
```

In [140…
```
# Find and return the maximum pixel value in the batch of normalized images.
# Since the images have been normalized to the range [0, 1], this should return
batch[0].max()
```

Out[140…
```
1.0
```

In [141…
```
# Find and return the minimum pixel value in the batch of normalized images.
batch[0].min()
```

Out[141…
```
0.0
```

5. Split Data

In [142…
```
# Return the number of batches in the TensorFlow dataset.
len(data)
```

Out[142…
```
22
```

In [143…
```
# Calculate the size of the training dataset.
# This is 70% of the total number of batches in the dataset.
train_size = int(len(data) * 0.7)

# Calculate the size of the validation dataset.
# This is 20% of the total number of batches in the dataset, plus one additional
val_size = int(len(data) * 0.2) + 1

# Calculate the size of the test dataset.
# This is 10% of the total number of batches in the dataset, plus one additional
test_size = int(len(data) * 0.1) + 1
```

In [144…
```
train_size+val_size+test_size
```

Out[144…
```
23
```

In [145…
```
# Create the training dataset by taking the first `train_size` batches from the
train = data.take(train_size)

# Create the validation dataset by skipping the first `train_size` batches and t
val = data.skip(train_size).take(val_size)

# Create the test dataset by skipping the first `train_size` + `val_size` batche
test = data.skip(train_size + val_size).take(test_size)
```

6. Build Deep Learning Model

In [146…
```
# Import the Sequential class from TensorFlow Keras, which allows for the creati
from tensorflow.keras.models import Sequential

# Import Conv2D layer for applying convolutional operations on 2D input (e.g., i
# This layer extracts features by sliding convolutional filters over the input i
from tensorflow.keras.layers import Conv2D
```

```
# Import MaxPooling2D layer for down-sampling the spatial dimensions (width and
# This helps in reducing the computational load and controlling overfitting.
from tensorflow.keras.layers import MaxPooling2D

# Import Dense layer for creating fully connected layers where each neuron is co
# This layer is typically used in the final stages of a network for classificati
from tensorflow.keras.layers import Dense

# Import Flatten layer to convert the 2D matrix of feature maps into a 1D vector
# This is necessary before passing the data to fully connected layers.
from tensorflow.keras.layers import Flatten

# Import Dropout layer to randomly set a fraction of input units to zero during
# This helps to prevent overfitting by making the network more robust.
from tensorflow.keras.layers import Dropout
```

In [147…

```
model = Sequential()
```

In [148…

```
# Add a 2D convolutional layer with 16 filters, each of size 3x3, stride of 1, a
# The input shape is specified as (256, 256, 3), which represents 256x256 RGB im
model.add(Conv2D(16, (3, 3), 1, activation='relu', input_shape=(256, 256, 3)))

# Add a 2D max pooling layer to down-sample the spatial dimensions of the featur
model.add(MaxPooling2D())

# Add another 2D convolutional layer with 32 filters, each of size 3x3, stride o
model.add(Conv2D(32, (3, 3), 1, activation='relu'))

# Add another 2D max pooling layer.
model.add(MaxPooling2D())

# Add a third 2D convolutional layer with 16 filters, each of size 3x3, stride o
model.add(Conv2D(16, (3, 3), 1, activation='relu'))

# Add another 2D max pooling layer.
model.add(MaxPooling2D())

# Flatten the 3D feature maps into a 1D vector to prepare them for the fully con
model.add(Flatten())

# Add a fully connected (dense) layer with 256 units and ReLU activation functio
model.add(Dense(256, activation='relu'))

# Add the output layer with 1 unit and sigmoid activation function for binary cl
# This layer produces a value between 0 and 1 representing the probability of th
model.add(Dense(1, activation='sigmoid'))
```

In [149…

```
# Compile the model with the specified optimizer, loss function, and evaluation
# 'adam' is the optimizer used for training, which adjusts the learning rate dur
model.compile(optimizer='adam',
              # 'BinaryCrossentropy' is the loss function used for binary classi
              loss=tf.losses.BinaryCrossentropy(),
              # 'accuracy' is the metric used to evaluate the model's performanc
              metrics=['accuracy'])
```

In [150…

```
model.summary()
```

Model: "sequential_2"

Layer (type)	Output Shape
conv2d_7 (Conv2D)	(None, 254, 254, 16)
max_pooling2d_6 (MaxPooling2D)	(None, 127, 127, 16)
conv2d_8 (Conv2D)	(None, 125, 125, 32)
max_pooling2d_7 (MaxPooling2D)	(None, 62, 62, 32)
conv2d_9 (Conv2D)	(None, 60, 60, 16)
max_pooling2d_8 (MaxPooling2D)	(None, 30, 30, 16)
flatten_2 (Flatten)	(None, 14400)
dense_4 (Dense)	(None, 256)
dense_5 (Dense)	(None, 1)

Total params: 3,696,625 (14.10 MB)

Trainable params: 3,696,625 (14.10 MB)

Non-trainable params: 0 (0.00 B)

7. Train

In [151…

```
# Define the directory where TensorBoard logs will be saved.
logdir = 'logs'
```

In [152…

```
# Create a TensorBoard callback to log training metrics and other information to
# TensorBoard uses these logs to visualize training progress, metrics, and other
tensorboard_callback = tf.keras.callbacks.TensorBoard(log_dir=logdir)
```

In [153…

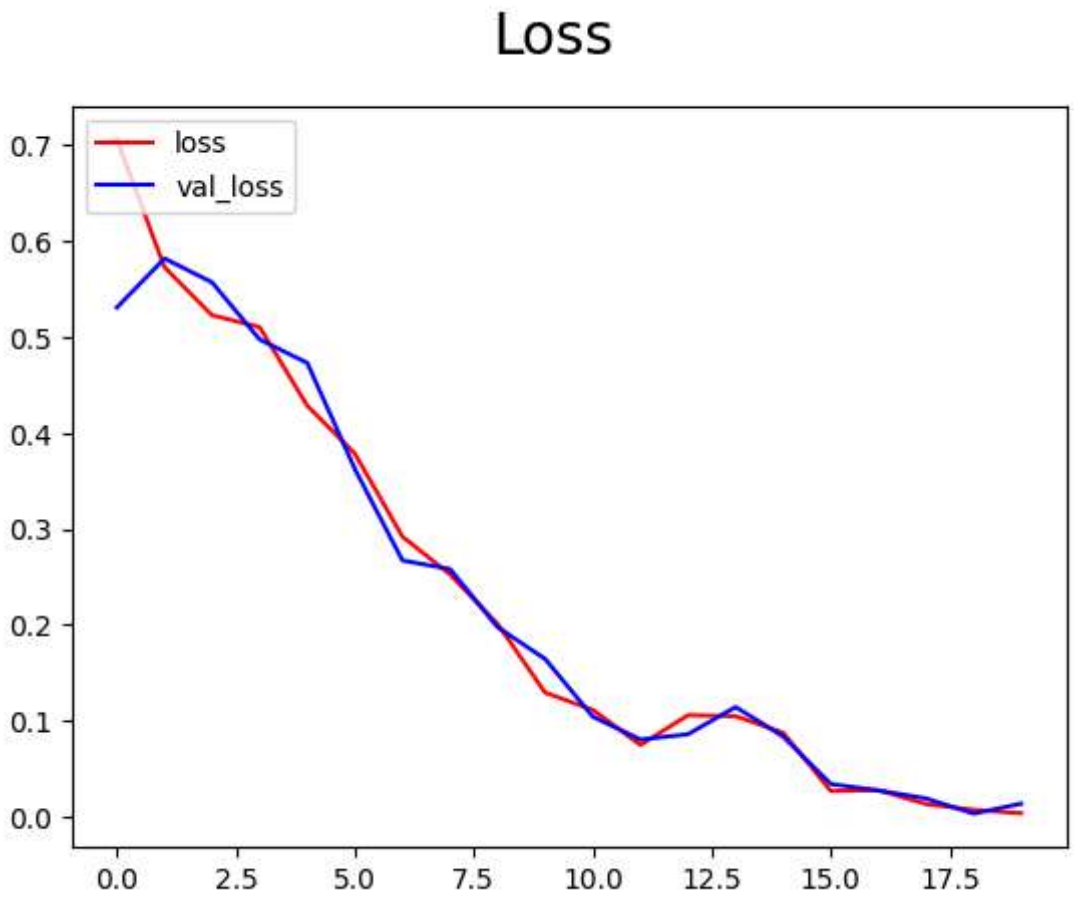
```
# Train the model on the training data for 20 epochs.
# During training, the model will also validate its performance on the validatio
# The TensorBoard callback is included to log training progress and metrics for
hist = model.fit(train,
                 epochs=20,
                 validation_data=val,
                 callbacks=[tensorboard_callback])
```

```
Epoch 1/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 7s 361ms/step - accuracy: 0.6843 - loss: 0.7680 - val_
accuracy: 0.7625 - val_loss: 0.5308
Epoch 2/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 346ms/step - accuracy: 0.7272 - loss: 0.5767 - val_
accuracy: 0.6875 - val_loss: 0.5819
Epoch 3/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 352ms/step - accuracy: 0.7848 - loss: 0.5027 - val_
accuracy: 0.7000 - val_loss: 0.5570
Epoch 4/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 330ms/step - accuracy: 0.7269 - loss: 0.5126 - val_
accuracy: 0.7312 - val_loss: 0.4975
Epoch 5/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 348ms/step - accuracy: 0.8052 - loss: 0.4094 - val_
accuracy: 0.8062 - val_loss: 0.4731
Epoch 6/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 324ms/step - accuracy: 0.8852 - loss: 0.3894 - val_
accuracy: 0.8625 - val_loss: 0.3626
Epoch 7/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 354ms/step - accuracy: 0.8959 - loss: 0.2697 - val_
accuracy: 0.8938 - val_loss: 0.2672
Epoch 8/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 358ms/step - accuracy: 0.9126 - loss: 0.2493 - val_
accuracy: 0.9062 - val_loss: 0.2583
Epoch 9/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 345ms/step - accuracy: 0.9412 - loss: 0.1773 - val_
accuracy: 0.9250 - val_loss: 0.1977
Epoch 10/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 333ms/step - accuracy: 0.9637 - loss: 0.1133 - val_
accuracy: 0.9438 - val_loss: 0.1646
Epoch 11/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 6s 375ms/step - accuracy: 0.9797 - loss: 0.1038 - val_
accuracy: 0.9563 - val_loss: 0.1045
Epoch 12/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 344ms/step - accuracy: 0.9800 - loss: 0.0767 - val_
accuracy: 0.9750 - val_loss: 0.0806
Epoch 13/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 6s 378ms/step - accuracy: 0.9791 - loss: 0.0754 - val_
accuracy: 0.9438 - val_loss: 0.0862
Epoch 14/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 335ms/step - accuracy: 0.9746 - loss: 0.0754 - val_
accuracy: 0.9812 - val_loss: 0.1145
Epoch 15/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 321ms/step - accuracy: 0.9749 - loss: 0.0924 - val_
accuracy: 0.9625 - val_loss: 0.0838
Epoch 16/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 321ms/step - accuracy: 0.9967 - loss: 0.0218 - val_
accuracy: 0.9937 - val_loss: 0.0343
Epoch 17/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 344ms/step - accuracy: 0.9994 - loss: 0.0276 - val_
accuracy: 1.0000 - val_loss: 0.0275
Epoch 18/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 338ms/step - accuracy: 1.0000 - loss: 0.0142 - val_
accuracy: 0.9937 - val_loss: 0.0193
Epoch 19/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 6s 399ms/step - accuracy: 1.0000 - loss: 0.0086 - val_
accuracy: 1.0000 - val_loss: 0.0034
Epoch 20/20
15/15 ━━━━━━━━━━━━━━━━━━━━ 5s 355ms/step - accuracy: 1.0000 - loss: 0.0031 - val_
accuracy: 0.9937 - val_loss: 0.0138
```

8. Plot Performance

In [154…

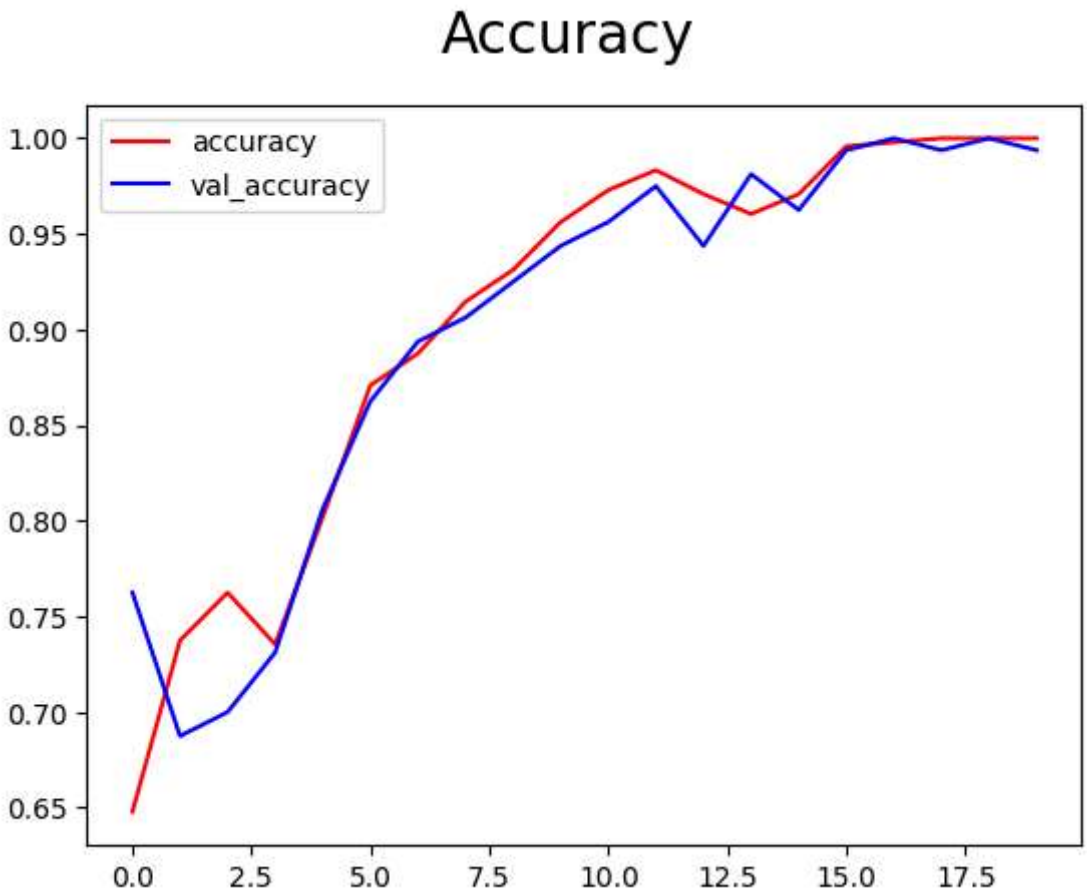
```
# Create a new figure for plotting the loss curves.
fig = plt.figure()

# Plot the training loss over epochs in red.
# 'hist.history['loss']' contains the loss values recorded during training.
plt.plot(hist.history['loss'], color='red', label='loss')

# Plot the validation loss over epochs in blue.
# 'hist.history['val_loss']' contains the loss values recorded during validation
plt.plot(hist.history['val_loss'], color='blue', label='val_loss')

# Set the title of the figure to 'Loss' with a font size of 20.
fig.suptitle('Loss', fontsize=20)

# Display a legend in the upper-left corner of the plot to distinguish between t
plt.legend(loc="upper left")

# Show the plot.
plt.show()
```



In [155…

```
# Create a new figure for plotting the accuracy curves.
fig = plt.figure()

# Plot the training accuracy over epochs in red.
# 'hist.history['accuracy']' contains the accuracy values recorded during traini
plt.plot(hist.history['accuracy'], color='red', label='accuracy')

# Plot the validation accuracy over epochs in blue.
```

```
# 'hist.history['val_accuracy']' contains the accuracy values recorded during va
plt.plot(hist.history['val_accuracy'], color='blue', label='val_accuracy')

# Set the title of the figure to 'Accuracy' with a font size of 20.
fig.suptitle('Accuracy', fontsize=20)

# Display a legend in the upper-left corner of the plot to distinguish between t
plt.legend(loc="upper left")

# Show the plot.
plt.show()
```



9. Evaluate

In [156…

```
# Import the Precision metric from TensorFlow Keras to evaluate the model's prec
# Precision measures the proportion of true positive predictions out of all posi
from tensorflow.keras.metrics import Precision

# Import the Recall metric from TensorFlow Keras to evaluate the model's recall.
# Recall measures the proportion of true positive predictions out of all actual
from tensorflow.keras.metrics import Recall

# Import the BinaryAccuracy metric from TensorFlow Keras to evaluate the model's
# BinaryAccuracy measures the proportion of correct predictions in binary classi
from tensorflow.keras.metrics import BinaryAccuracy
```

In [157…

```
pre = Precision()
re = Recall()
acc = BinaryAccuracy()
```

In [158…

```
# Iterate over batches in the test dataset converted to a Numpy iterator.
for batch in test.as_numpy_iterator():
    # Unpack the batch into features (X) and labels (y).
    X, y = batch

    # Use the model to make predictions on the batch of features (X).
    yhat = model.predict(X)

    # Update the state of the Precision metric with the true labels (y) and pred
    pre.update_state(y, yhat)

    # Update the state of the Recall metric with the true labels (y) and predict
    re.update_state(y, yhat)

    # Update the state of the BinaryAccuracy metric with the true labels (y) and
    acc.update_state(y, yhat)
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 106ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 66ms/step
```

In [159…

```
print(pre.result(), re.result(), acc.result())
```

```
tf.Tensor(1.0, shape=(), dtype=float32) tf.Tensor(0.962963, shape=(), dtype=float
32) tf.Tensor(0.9722222, shape=(), dtype=float32)
```

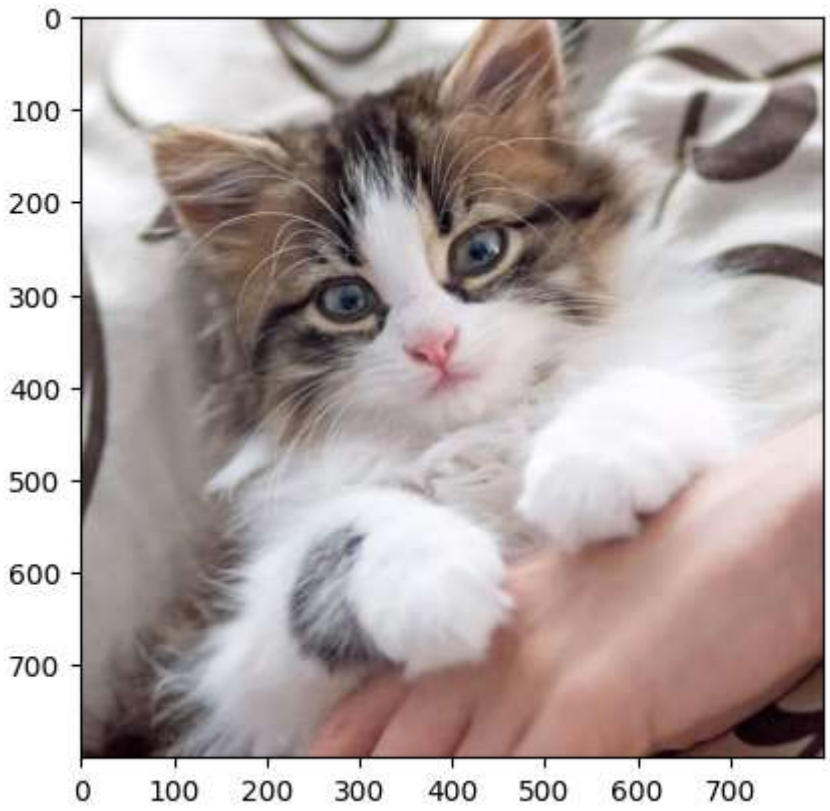
10. Test

In [183…

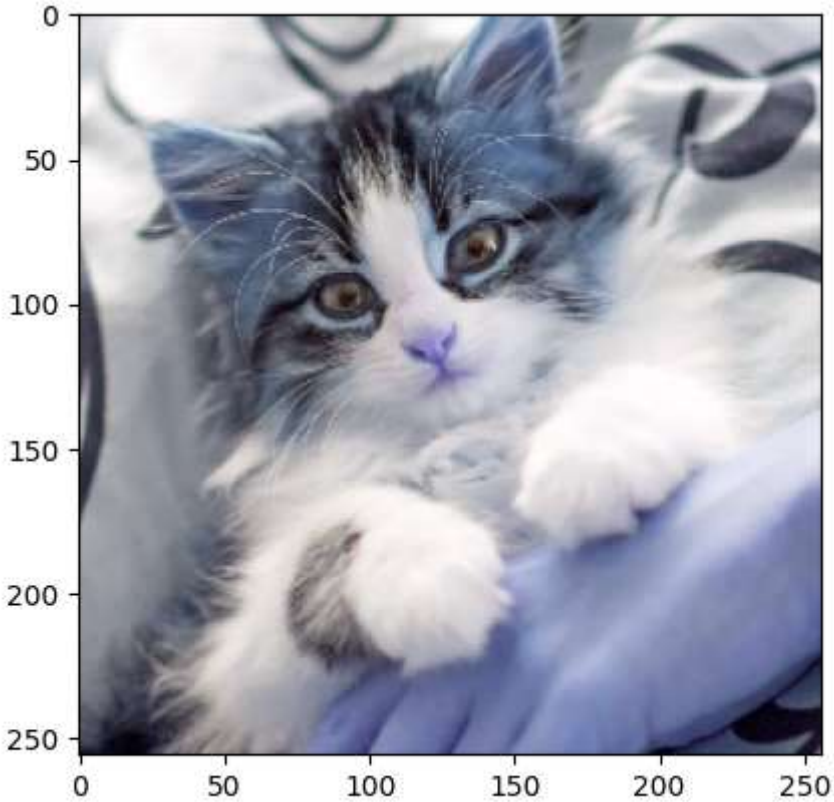
```
import cv2
```

In [197…

```
img = cv2.imread('cattt.jpg')
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
plt.show()
```



In [198…
```
resize = tf.image.resize(img, (256,256))
plt.imshow(resize.numpy().astype(int))
plt.show()
```



In [199…
```
# Make a prediction using the trained model for a single image.
# The image is first resized and normalized by dividing pixel values by 255.
# np.expand_dims adds an extra dimension to the image to match the expected inpu
yhat = model.predict(np.expand_dims(resize / 255, 0))
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 36ms/step
```

In [200…
```
yhat
```

Out[200…
```
array([[0.00319709]], dtype=float32)
```

In [201…
```
if yhat > 0.5:
    print(f'Predicted class is puppies')
else:
    print(f'Predicted class is kitten')
```

```
Predicted class is kitten
```

11. Save the Model

In [202…
```
# Import the `load_model` function from TensorFlow Keras to load a saved model.
# This function allows you to load a previously saved model from a file, includi
from tensorflow.keras.models import load_model
```

In [203…
```
model.save(os.path.join('models','imageclassifier.h5'))
```

```
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `ker
as.saving.save_model(model)`. This file format is considered legacy. We recommend
using instead the native Keras format, e.g. `model.save('my_model.keras')` or `ke
ras.saving.save_model(model, 'my_model.keras')`.
```

In [204…

```
os.path.join('models','imageclassifier.h5')
```

Out[204…

```
'models\\imageclassifier.h5'
```

In [205…

```
new_model = load_model('models\\imageclassifier.h5')
```

```
WARNING:absl:Compiled the loaded model, but the compiled metrics have yet to be b
uilt. `model.compile_metrics` will be empty until you train or evaluate the mode
l.
```

In [206…

```
yhatnew=new_model.predict(np.expand_dims(resize/255, 0))
```

```
WARNING:tensorflow:5 out of the last 10 calls to <function TensorFlowTrainer.make
_predict_function.<locals>.one_step_on_data_distributed at 0x000002562BAB6AC0> tr
iggered tf.function retracing. Tracing is expensive and the excessive number of t
racings could be due to (1) creating @tf.function repeatedly in a loop, (2) passi
ng tensors with different shapes, (3) passing Python objects instead of tensors.
For (1), please define your @tf.function outside of the loop. For (2), @tf.functi
on has reduce_retracing=True option that can avoid unnecessary retracing. For
(3), please refer to https://www.tensorflow.org/guide/function#controlling_retrac
ing and https://www.tensorflow.org/api_docs/python/tf/function for  more details.
```

```
WARNING:tensorflow:5 out of the last 10 calls to <function TensorFlowTrainer.make
_predict_function.<locals>.one_step_on_data_distributed at 0x000002562BAB6AC0> tr
iggered tf.function retracing. Tracing is expensive and the excessive number of t
racings could be due to (1) creating @tf.function repeatedly in a loop, (2) passi
ng tensors with different shapes, (3) passing Python objects instead of tensors.
For (1), please define your @tf.function outside of the loop. For (2), @tf.functi
on has reduce_retracing=True option that can avoid unnecessary retracing. For
(3), please refer to https://www.tensorflow.org/guide/function#controlling_retrac
ing and https://www.tensorflow.org/api_docs/python/tf/function for  more details.
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 93ms/step
```

In [207…

```
if yhatnew > 0.5:
    print(f'predicted class is puppies')
else:
    print(f'pridicted class is kitten')
```

```
pridicted class is kitten
```