

```
In [23]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import tensorflow as tf

from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.optimizers import SGD
```

```
In [24]: data_train_path='C:/Python/archive (4)/marvel/train'
data_val_path='C:/Python/archive (4)/marvel/validation'
```

```
In [25]: img_width =180
img_height =180
```

```
In [26]: data_train= tf.keras.utils.image_dataset_from_directory (
    'C:/Python/archive (4)/marvel/train',
    shuffle=True,
    image_size=(img_width,img_height),
    batch_size=32,
    validation_split=False )
```

Found 2584 files belonging to 8 classes.

```
In [27]: data_cat= data_train.class_names
```

```
In [28]: data_val= tf.keras.utils.image_dataset_from_directory (
    'C:/Python/archive (4)/marvel/valid',
    shuffle=True,
    image_size=(img_width,img_height),
    batch_size=32,
    validation_split=False )
```

Found 451 files belonging to 8 classes.

```
In [29]: plt.figure(figsize=(10,10))
for image,labels in data_train.take(1):
    for i in range(9):
        plt.subplot(3,3,i+1)
        plt.imshow(image[i].numpy().astype('uint8'))
        plt.title(data_cat[labels[i]])
        plt.axis('off')
```

spider-man



loki



captain america



spider-man



thanos



spider-man



captain america



loki



black widow



```
In [30]: from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
graph = tf.compat.v1.get_default_graph()
```

```
In [31]: model = Sequential([
    layers.Rescaling(1./255),
    layers.Conv2D(16,3,padding='same',activation='relu'),
    layers.MaxPooling2D(),
    layers.Conv2D(32,3,padding='same',activation='relu'),
    layers.MaxPool2D(),
    layers.Conv2D( 64,3,padding='same',activation='relu'),
    layers.Flatten(),
    layers.Dropout(0.2),
    layers.Dense(128),
    layers.Dense(len(data_cat))
])
```

```
In [32]: model.compile(optimizer='adam',loss=tf.keras.losses.SparseCategoricalCrossentropy(from_1
```

```
In [33]: epochs_size =25  
history =model.fit(data_train, validation_data=data_val, epochs=epochs_size)
```

Epoch 1/25
WARNING:tensorflow:From C:\Users\nidhi\anaconda3\Lib\site-packages\keras\src\utils\tf_utils.py:492: The name tf.nn.RaggedTensorValue is deprecated. Please use tf.nn.compact_v1.nn.RaggedTensorValue instead.

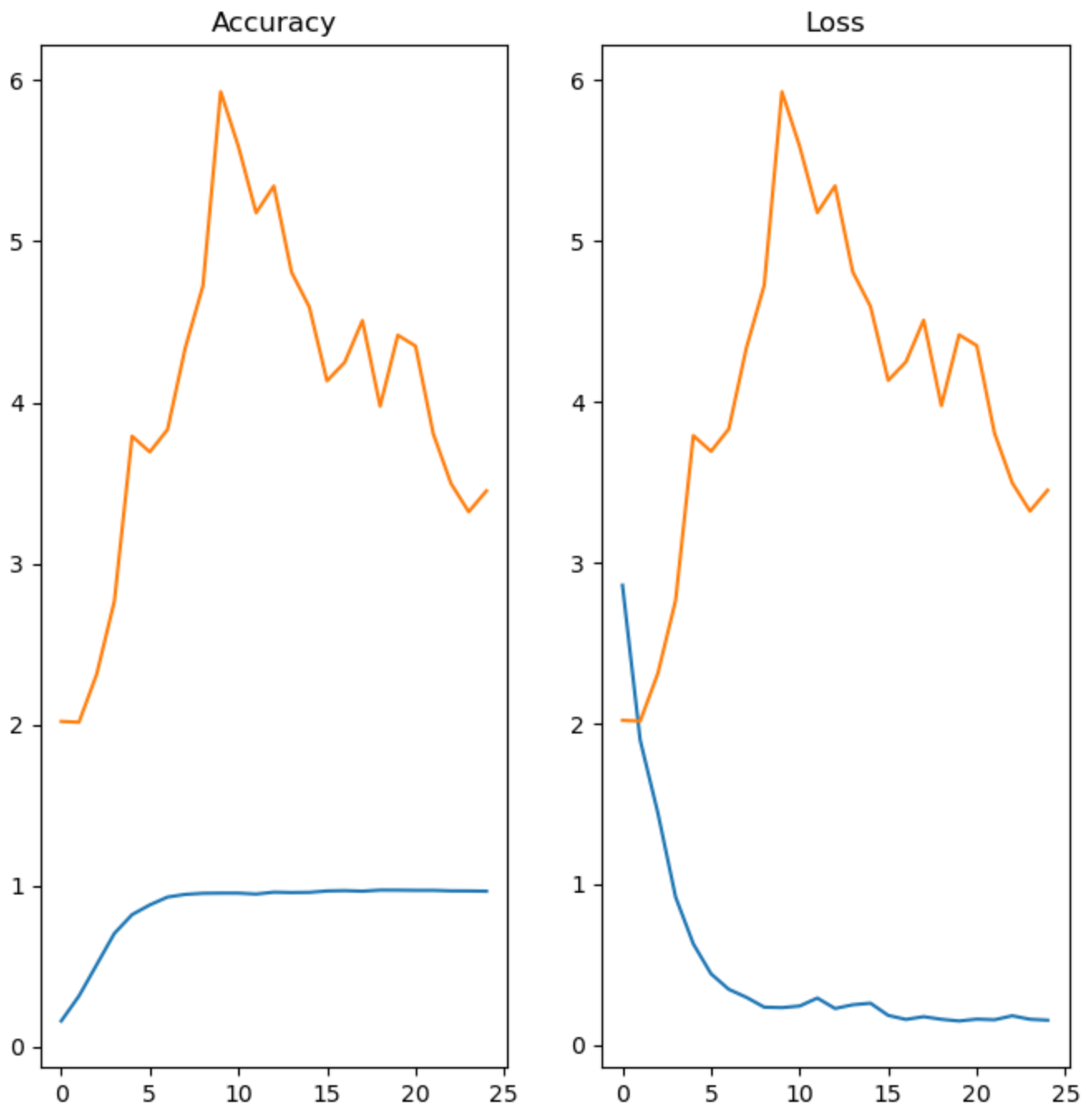
81/81 [=====] - 31s 353ms/step - loss: 2.8598 - accuracy: 0.1602 - val_loss: 2.0199 - val_accuracy: 0.2217
Epoch 2/25
81/81 [=====] - 28s 340ms/step - loss: 1.8963 - accuracy: 0.3150 - val_loss: 2.0147 - val_accuracy: 0.2373
Epoch 3/25
81/81 [=====] - 26s 311ms/step - loss: 1.4430 - accuracy: 0.5097 - val_loss: 2.3124 - val_accuracy: 0.1907
Epoch 4/25
81/81 [=====] - 30s 361ms/step - loss: 0.9205 - accuracy: 0.7039 - val_loss: 2.7662 - val_accuracy: 0.2018
Epoch 5/25
81/81 [=====] - 31s 369ms/step - loss: 0.6295 - accuracy: 0.8204 - val_loss: 3.7906 - val_accuracy: 0.1996
Epoch 6/25
81/81 [=====] - 25s 307ms/step - loss: 0.4433 - accuracy: 0.8808 - val_loss: 3.6925 - val_accuracy: 0.1951
Epoch 7/25
81/81 [=====] - 30s 365ms/step - loss: 0.3471 - accuracy: 0.9296 - val_loss: 3.8315 - val_accuracy: 0.2173
Epoch 8/25
81/81 [=====] - 31s 378ms/step - loss: 0.2970 - accuracy: 0.9462 - val_loss: 4.3381 - val_accuracy: 0.2129
Epoch 9/25
81/81 [=====] - 26s 312ms/step - loss: 0.2361 - accuracy: 0.9528 - val_loss: 4.7243 - val_accuracy: 0.2062
Epoch 10/25
81/81 [=====] - 28s 344ms/step - loss: 0.2333 - accuracy: 0.9539 - val_loss: 5.9287 - val_accuracy: 0.1774
Epoch 11/25
81/81 [=====] - 33s 394ms/step - loss: 0.2430 - accuracy: 0.9539 - val_loss: 5.5893 - val_accuracy: 0.1951
Epoch 12/25
81/81 [=====] - 25s 303ms/step - loss: 0.2926 - accuracy: 0.9481 - val_loss: 5.1769 - val_accuracy: 0.2062
Epoch 13/25
81/81 [=====] - 25s 308ms/step - loss: 0.2277 - accuracy: 0.9598 - val_loss: 5.3432 - val_accuracy: 0.1951
Epoch 14/25
81/81 [=====] - 31s 382ms/step - loss: 0.2510 - accuracy: 0.9574 - val_loss: 4.8082 - val_accuracy: 0.2151
Epoch 15/25
81/81 [=====] - 29s 345ms/step - loss: 0.2606 - accuracy: 0.9586 - val_loss: 4.5931 - val_accuracy: 0.1929
Epoch 16/25
81/81 [=====] - 25s 306ms/step - loss: 0.1849 - accuracy: 0.9675 - val_loss: 4.1337 - val_accuracy: 0.2084
Epoch 17/25
81/81 [=====] - 31s 374ms/step - loss: 0.1597 - accuracy: 0.9694 - val_loss: 4.2486 - val_accuracy: 0.1863
Epoch 18/25
81/81 [=====] - 30s 362ms/step - loss: 0.1770 - accuracy: 0.9659 - val_loss: 4.5080 - val_accuracy: 0.1885
Epoch 19/25
81/81 [=====] - 26s 312ms/step - loss: 0.1606 - accuracy: 0.9729 - val_loss: 3.9765 - val_accuracy: 0.2062
Epoch 20/25
81/81 [=====] - 30s 369ms/step - loss: 0.1501 - accuracy: 0.9725 - val_loss: 4.4175 - val_accuracy: 0.2106

```
Epoch 21/25
81/81 [=====] - 30s 365ms/step - loss: 0.1615 - accuracy: 0.971
4 - val_loss: 4.3498 - val_accuracy: 0.2151
Epoch 22/25
81/81 [=====] - 25s 307ms/step - loss: 0.1577 - accuracy: 0.971
4 - val_loss: 3.8055 - val_accuracy: 0.2106
Epoch 23/25
81/81 [=====] - 26s 313ms/step - loss: 0.1826 - accuracy: 0.967
9 - val_loss: 3.4968 - val_accuracy: 0.2129
Epoch 24/25
81/81 [=====] - 34s 411ms/step - loss: 0.1608 - accuracy: 0.967
5 - val_loss: 3.3212 - val_accuracy: 0.2262
Epoch 25/25
81/81 [=====] - 28s 326ms/step - loss: 0.1551 - accuracy: 0.965
9 - val_loss: 3.4510 - val_accuracy: 0.2106
```

```
In [34]: epochs_range = range(epochs_size)
plt.figure(figsize=(8,8))
plt.subplot(1,2,1)
plt.plot(epochs_range,history.history['accuracy'],label='Training Accuracy')
plt.plot(epochs_range,history.history['val_loss'],label='Validation Accuracy')
plt.title('Accuracy')

plt.subplot(1,2,2)
plt.plot(epochs_range,history.history['loss'],label='Training Loss')
plt.plot(epochs_range,history.history['val_loss'],label='Validation Loss')
plt.title('Loss')
```

```
Out[34]: Text(0.5, 1.0, 'Loss')
```



```
In [36]: image = 'C:/Python/archive (4)/pic_033.jpg'
image = tf.keras.utils.load_img(image, target_size=(img_height,img_width))
img_arr =tf.keras.utils.array_to_img(image)
img_bat=tf.expand_dims(img_arr,0)

In [37]: predict =model.predict(img_bat)
1/1 [=====] - 0s 248ms/step

In [38]: score = tf.nn.softmax(predict)

In [39]: print('marvel hero in image is {} with accjuracy of{:0.2f}'.format(data_cat[np.argmax(sc
marvel hero in image is doctor strange with accjuracy of92.57

In [41]: model.save('Image_classification_marvel.keras')

In [ ]:
```