

```
In [33]: import os
import cv2
import matplotlib.pyplot as plt
import numpy as np

import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.preprocessing import image
from tensorflow.keras.optimizers import RMSprop

from colorthief import ColorThief
import matplotlib.pyplot as plt
import colorsys
import webcolors
```

```
In [34]: mg = image.load_img("Downloads/Project/Db/training/Shirt/20.jpeg")
image = cv2.imread("Downloads/Project/Db/training/Shirt/20.jpeg")
image_rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
```

```
In [7]: train= ImageDataGenerator(rescale=1/255)
validation=ImageDataGenerator(rescale=1/255)
```

```
In [27]: train_data=train.flow_from_directory("Downloads/Project/Db/training/",
                                              target_size=(200,200),
                                              batch_size=3,
                                              class_mode='binary')
validation_dataset=train.flow_from_directory("Downloads/Project/Db/validation/",
                                              target_size=(200,200),
                                              batch_size=3,
                                              class_mode='binary')
```

Found 33 images belonging to 2 classes.
Found 0 images belonging to 0 classes.

```
In [9]: train_data.class_indices
```

```
Out[9]: {'Pant': 0, 'Shirt': 1}
```

```
In [10]: model = tf.keras.models.Sequential([ tf.keras.layers.Conv2D(16,(3,3),activation ="relu",
tf.keras.layers.MaxPool2D(2,2),
#
tf.keras.layers.Conv2D(32,(3,3),activation ="relu")
tf.keras.layers.MaxPool2D(2,2),
#
tf.keras.layers.Conv2D(64,(3,3),activation ="relu")
tf.keras.layers.MaxPool2D(2,2),
##
tf.keras.layers.Flatten(),
##
tf.keras.layers.Dense(512,activation= 'relu'),
##
tf.keras.layers.Dense(1,activation='sigmoid')
])
```

```
In [11]: model.compile(loss="binary_crossentropy",optimizer=RMSprop(learning_rate=0.001),metri
```

[illegible]

Epoch 1/30
3/3 [=====] - 4s 421ms/step - loss: 14.4196 - accuracy: 0.556
Epoch 2/30
3/3 [=====] - 1s 373ms/step - loss: 1.8009 - accuracy: 0.33
Epoch 3/30
3/3 [=====] - 1s 359ms/step - loss: 0.6864 - accuracy: 0.778
Epoch 4/30
3/3 [=====] - 1s 373ms/step - loss: 0.7306 - accuracy: 0.556
Epoch 5/30
3/3 [=====] - 1s 384ms/step - loss: 0.4954 - accuracy: 0.778
Epoch 6/30
3/3 [=====] - 1s 396ms/step - loss: 0.6108 - accuracy: 0.667
Epoch 7/30
3/3 [=====] - 1s 392ms/step - loss: 0.5432 - accuracy: 0.778
Epoch 8/30
3/3 [=====] - 1s 472ms/step - loss: 0.3050 - accuracy: 1.000
Epoch 9/30
3/3 [=====] - 1s 376ms/step - loss: 0.3999 - accuracy: 0.778
Epoch 10/30
3/3 [=====] - 1s 454ms/step - loss: 0.2265 - accuracy: 1.000
Epoch 11/30
3/3 [=====] - 1s 367ms/step - loss: 0.3198 - accuracy: 0.889
Epoch 12/30
3/3 [=====] - 1s 377ms/step - loss: 0.3505 - accuracy: 0.778
Epoch 13/30
3/3 [=====] - 1s 373ms/step - loss: 0.0524 - accuracy: 1.000
Epoch 14/30
3/3 [=====] - 1s 398ms/step - loss: 0.1249 - accuracy: 1.000
Epoch 15/30
3/3 [=====] - 2s 426ms/step - loss: 0.0596 - accuracy: 1.000
Epoch 16/30
3/3 [=====] - 2s 600ms/step - loss: 0.6082 - accuracy: 0.778
Epoch 17/30
3/3 [=====] - 1s 408ms/step - loss: 0.3338 - accuracy: 0.778
Epoch 18/30
3/3 [=====] - 1s 455ms/step - loss: 0.3247 - accuracy: 0.889
Epoch 19/30
3/3 [=====] - 2s 544ms/step - loss: 0.2192 - accuracy: 1.000
Epoch 20/30
3/3 [=====] - 1s 438ms/step - loss: 0.0157 - accuracy: 1.000
Epoch 21/30
3/3 [=====] - 2s 484ms/step - loss: 0.0164 - accuracy: 1.000
Epoch 22/30
3/3 [=====] - 1s 436ms/step - loss: 0.0105 - accuracy: 1.000
Epoch 23/30
3/3 [=====] - 1s 379ms/step - loss: 0.0046 - accuracy: 1.000

```
00
Epoch 24/30
3/3 [=====] - 1s 380ms/step - loss: 0.0065 - accuracy: 1.00
00
Epoch 25/30
3/3 [=====] - 1s 383ms/step - loss: 0.0033 - accuracy: 1.00
00
Epoch 26/30
3/3 [=====] - 1s 389ms/step - loss: 0.0022 - accuracy: 1.00
00
Epoch 27/30
3/3 [=====] - 1s 379ms/step - loss: 7.9672e-04 - accuracy:
1.0000
Epoch 28/30
3/3 [=====] - 1s 469ms/step - loss: 0.0014 - accuracy: 1.00
00
Epoch 29/30
3/3 [=====] - 1s 500ms/step - loss: 0.0074 - accuracy: 1.00
00
Epoch 30/30
3/3 [=====] - 1s 375ms/step - loss: 0.0012 - accuracy: 1.00
00
```

```
In [30]: dir_path='Downloads/Project/Db/test'
shirt=[]
pant=[]
for i in os.listdir(dir_path ):
    img=image.load_img(dir_path+'/'+i,target_size=(200,200))
    plt.imshow(img)
    print(i)
    plt.axis("off")
    plt.show()

    x=image.img_to_array(img)
    x=np.expand_dims(x,axis=0)
    images=np.vstack([x])

    val=model.predict(images)
    if val==0:
        print("It is pant")
    else:
        print("It is shirt")
```

2.jpg



1/1 [=====] - 0s 51ms/step
It is pant
red.jpg



1/1 [=====] - 0s 38ms/step
It is shirt

```
In [24]: a=[]
# Load the image
for i in os.listdir(dir_path):
    img=ColorThief(dir_path+'/'+i)
    print(i)
    dominant_color = img.get_color(quality=1)
    dominant_color_hex = "#{:02x}{:02x}{:02x}".format(*dominant_color)

    print("Dominant Color:", dominant_color_hex)
    a.append(dominant_color_hex)
print(a)
```

```
2.jpg
Dominant Color: #181513
red.jpg
Dominant Color: #440e19
['#181513', '#440e19']
```

```
In [25]: def hex_to_rgb(hex_color):
    hex_color = hex_color.lstrip('#')
    return tuple(int(hex_color[i:i+2], 16) for i in (0, 2, 4))

def color_similarity(color1, color2):
    r1, g1, b1 = color1
    r2, g2, b2 = color2
    distance = ((r1 - r2) ** 2 + (g1 - g2) ** 2 + (b1 - b2) ** 2) ** 0.5
    return 100 - (distance / (255 * 3)) * 100

shirt_color_hex = a[0]
pant_color_hex = a[1]

shirt_color_rgb = hex_to_rgb(shirt_color_hex)
pant_color_rgb = hex_to_rgb(pant_color_hex)

similarity_percentage = color_similarity(shirt_color_rgb, pant_color_rgb)
print("Percentage of the color match:", similarity_percentage, "%")
```

Percentage of the color match: 94.12345966196365 %

