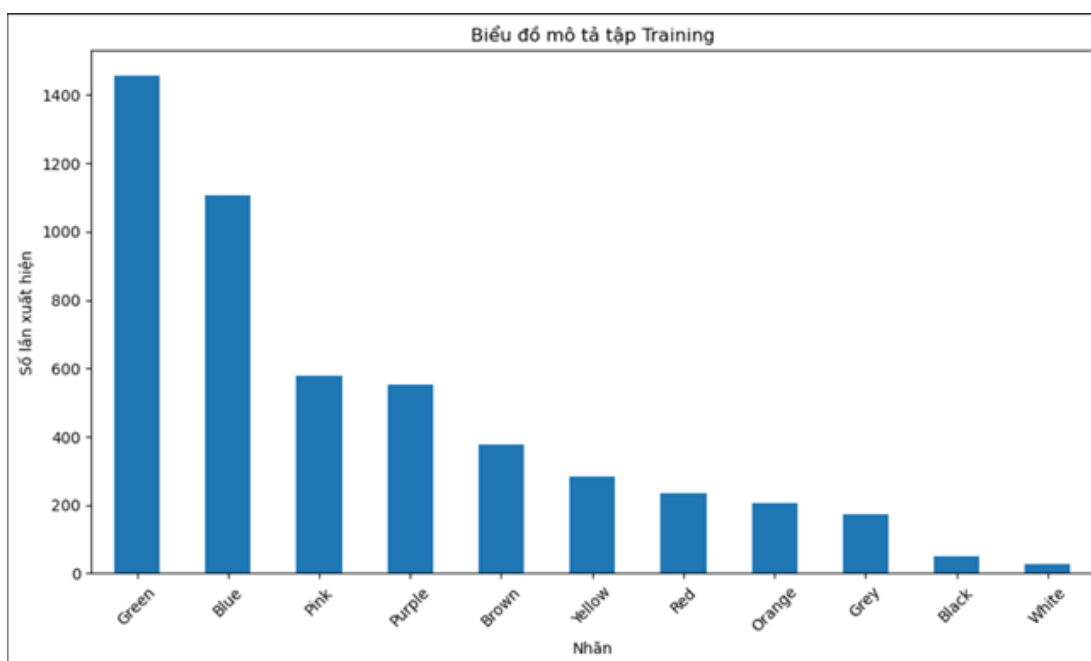


0.1 Mô tả tập dữ liệu

Tập dữ liệu của mô hình bao gồm 2 loại: tập Training và tập Test. Tập Training gồm 5053 mẫu dữ liệu được chia làm 11 màu: đen, xanh dương, xanh lá, nâu, xám, cam, hồng, tím, đỏ, trắng và vàng được mô tả cụ thể trong hình 0.1. Tập Test gồm 307 mẫu dữ liệu với đầy đủ 11 màu kể trên. Mỗi mẫu dữ liệu trong bài toán là một hình ảnh được phân tích theo biểu đồ màu RGB thành 3 đặc trưng và 1 nhãn, cụ thể trong hình 0.2. Tập Training được sử dụng để huấn luyện mô hình còn tập Test được sử dụng để đánh giá độ chính xác.



Hình 0.1: Biểu đồ mô tả tập dữ liệu training

	red	green	blue	label
0	20	139	240	Blue
1	174	83	72	Brown
2	144	249	131	Green
3	168	25	156	Pink
4	30	182	136	Green
5	199	150	175	Pink
6	199	93	154	Pink
7	231	243	25	Yellow
8	48	213	76	Green

Hình 0.2: Dữ liệu Training/Test

0.2 Trích xuất các đặc trưng từ ảnh

Như đã nêu ở phần trước, mô hình này sử dụng biểu đồ màu RGB để trích xuất các đặc trưng màu sắc của ảnh.

0.2.1 Khai báo các thư viện

Thư viện os: được sử dụng trong quá trình duyệt từng tệp trong thư mục.

Thư viện cv2: được sử dụng trong quá trình xử lý ảnh, cụ thể là chia ảnh thành các kênh màu RGB

Thư viện numpy: được sử dụng để tính toán số học và ma trận, cụ thể là đưa ra giá trị lớn nhất trong mảng

```
1 import os
2 import cv2
3 import numpy as np
4 # from color_recognition_api import KNN
```

Hình 0.3: Khai báo các thư viện cần thiết khi trích xuất đặc trưng màu ảnh

0.2.2 Xây dựng hàm

a, Đối với mẫu dữ liệu Training/Test

Để lấy các đặc trưng, ta sử dụng hàm `color_histogram_of_image` trong `color_recognition_api`.

b, Đối với mẫu dữ liệu Predict

0.3 Sử dụng phương pháp K-nearest Neighbors để phân loại màu ảnh

0.4 Đánh giá mô hình

0.5 Kết quả thu được

```

6  def color_histogram_of_test_image(test_src_image):
7      ...
8      Trích xuất đặc trưng màu RGB để đem đi dự đoán
9      ...
10
11     # load the image
12     image = test_src_image
13
14     chans = cv2.split(image) #chia ảnh thành các kênh màu, theo cv2 thì là 3 kênh màu theo thứ tự BGR
15     colors = ('b', 'g', 'r') #tên các kênh màu
16
17     feature_data = '' #chứa đặc trưng màu của 1 ảnh
18     counter = 0 #có vai trò để kiểm tra đang ở kênh màu nào
19     for (chan, color) in zip(chans, colors):
20         counter = counter + 1
21
22         hist = cv2.calcHist([chan], [0], None, [256], [0, 256])
23         #hist = cv2.calcHist(images, channels, mask, histSize, ranges)
24
25         elem = np.argmax(hist) #trả về chỉ số lớn nhất trong mảng hist
26
27         if counter == 1:
28             blue = str(elem)
29         elif counter == 2:
30             green = str(elem)
31         elif counter == 3:
32             red = str(elem)
33             feature_data = red + ',' + green + ',' + blue
34             # print(feature_data)
35
36     # Mo file va ghi de len file data
37     with open('./testing_dataset/testing_image.data', 'w') as myfile:
38         myfile.write(feature_data)
39

```

Hình 0.5: Trích xuất đặc trưng của mẫu dữ liệu Test