

PHÂN LOẠI TÍN HIỆU ĐÈN GIAO THÔNG

Nguyễn Hữu Khang - 18520892 - CS114.K21

Link Github: <https://github.com/khangnguyenhuu/CS114.K21>

Phan Hoàng Nguyên - 18521163 - CS114.K21

Link Github: <https://github.com/18521163/CS114.K21>

Hồ Đăng Tuệ - 18521611 - CS114.K21

Link Github: <https://github.com/Dangtue-uit/CS114.K21>

Tóm tắt

- Tên đề tài: PHÂN LOẠI TÍNH HIỆU ĐÈN GIAO THÔNG
- Tóm tắt đồ án: Tìm hiểu về cách tìm thu thập, gán nhãn dữ liệu, tiền xử lí, rút trích đặc trưng ảnh, thực nghiệm bằng nhiều cách khác nhau để so sánh.
- Kết quả: Độ chính xác cao nhất đạt được khi khảo một số phương pháp trên tập test là 96%.
- Ảnh thành viên nhóm



Phan Hoàng Nguyễn



Nguyễn Hữu Khang



Hồ Đăng Tuệ

Bài toán

Giao thông thông minh là một vấn đề lớn, bao gồm rất nhiều bài toán con. Trong đó bài toán nhận diện tín hiệu đèn giao thông là bài toán rất quan trọng. Nó phục vụ cho rất nhiều mục đích như xử phạt nguội các hành vi vi phạm giao thông, xe tự hành.... Bài toán nhận diện tín hiệu đèn giao thông bao gồm 2 bài toán con cần giải quyết: Phát hiện đèn giao thông (Detect), Phân loại tín hiệu đèn giao thông (Classify). Đầu ra của bài toán phát hiện đèn giao thông là hình ảnh đèn giao thông (không bao gồm các vật thể nhiễu từ bên ngoài) đây cũng chính là đầu vào của bài toán phân loại tín hiệu đèn giao thông.

Bài toán

Bài toán phân loại tín hiệu đèn giao thông nhận đầu vào là một hình ảnh về đèn giao thông (chỉ chứa đèn giao thông) và cho đầu ra là kết quả dự đoán màu của đèn giao thông này.

Den xanh



Input: ảnh đèn giao thông

Output: dự đoán trạng thái màu đèn hiện tại.

Mục tiêu đề tài

- Tìm hiểu về phương pháp rút trích đặc trưng ảnh.
- Khảo sát một số thuật toán máy học.
- Thử giải quyết bài toán phân lớp đèn giao thông bằng cách lập trình truyền thống.
- Xây dựng một bộ dữ liệu dùng để phân lớp đèn giao thông.
- Xây dựng một hàm nhận đầu vào là hình đường dẫn đến hình ảnh và cho đầu ra là kết quả phân lớp.

Mô tả dữ liệu

- Cách thức thu thập: gồm 2 nguồn, Internet: từ các clip camera hành trình, google image, bộ data có sẵn. Thực tế: chụp các đèn giao thông trong Sài Gòn. Hầu hết ảnh thu thập chứa nhiều object gây nhiễu. Ta chỉ quan tâm đến đèn giao thông nên ta sẽ cắt các đèn giao thông ra. Có 2 cách, cách 1 là giải quyết bài toán nhận diện (detection), cách 2 là cắt bằng tay. Cách 2 có độ chính xác cao hơn, cũng như đơn giản hơn cách 1

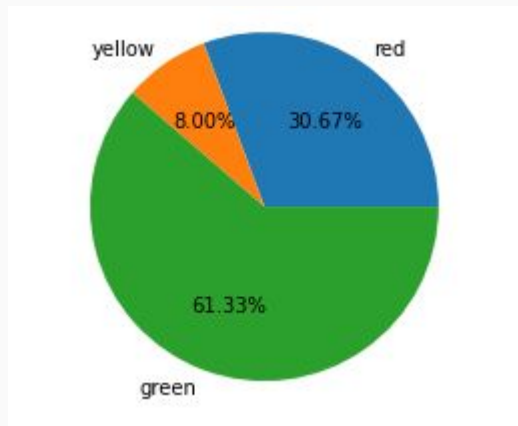


----Sau khi cắt ảnh---->



Mô tả dữ liệu

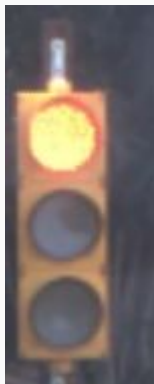
- Số lượng dữ liệu được thu thập là 1112 ảnh trong đó 1009 ảnh thu thập trên internet (dùng làm train set) và 103 ảnh thu thập từ thực tế (dùng làm test set)
- Có 341 ảnh đèn đỏ, 89 ảnh đèn vàng, 682 ảnh đèn xanh. Đèn vàng ít là do chỉ tồn tại rất ngắn trong thực tế (2-3s)



Ảnh phân bố dữ liệu

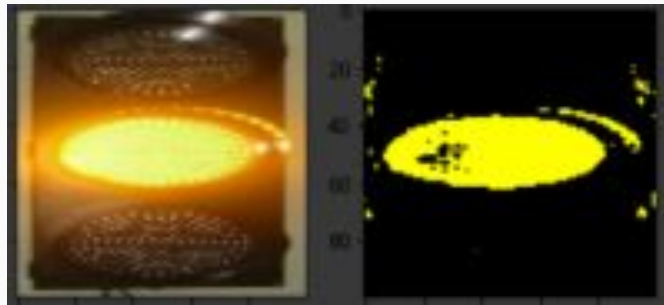
Mô tả dữ liệu

- Dữ liệu bị ảnh hưởng rất nhiều bởi ánh sáng. Ánh sáng lóa, làm cho màu đèn bị ảnh hưởng và thay đổi, môi trường buổi tối cũng là một thử thách, ở dưới là ảnh đèn đỏ, có thể nhầm thành đèn vàng. Và ảnh đèn xanh trong điều kiện buổi tối.



Tiền xử lý và rút trích đặc trưng

- **Tiền xử lý:** Tất cả ảnh đều được resize về 100x100px và sử dụng bộ lọc mờ trung bình để khử nhiễu
- **Rút trích đặc trưng:** Sử dụng phương pháp Color image segmentation tạo ra các ngưỡng màu (Color threshold) nếu các pixel của ảnh nằm trong các ngưỡng màu này thì ta sẽ giữ lại pixel đó, các pixel còn lại sẽ đưa về giá trị màu đen (có 3 ngưỡng màu cho đỏ, vàng, xanh lá).



Kết quả sau khi rút trích đặc trưng

Đưa vào thực nghiệm

Thực nghiệm bằng 3 phương pháp:

- Máy học có rút trích đặc trưng
- Máy học không rút trích đặc trưng
- Lập trình truyền thống.

* Sử dụng 3 thuật toán đó là **Naive Bayes, KNN, SVM**.

Kết quả

Máy học có rút trích đặc trưng:

```
report svm on test set
      precision    recall  f1-score   support

         0         0.27    0.43    0.33         46
         1         1.00    0.10    0.18         10
         2         0.21    0.13    0.16         53

 accuracy          0.26         109
 macro avg         0.49    0.22    0.23         109
 weighted avg      0.31    0.26    0.24         109
```

```
report Naive bayes on test set
      precision    recall  f1-score   support

         0         0.70    0.50    0.58         46
         1         0.67    0.60    0.63         10
         2         0.70    0.89    0.78         53

 accuracy          0.70         109
 macro avg         0.69    0.66    0.67         109
 weighted avg      0.70    0.70    0.68         109
```

```
report KNN on test set
      precision    recall  f1-score   support

         0         0.79    0.48    0.59         46
         1         0.67    0.80    0.73         10
         2         0.71    0.92    0.80         53

 accuracy          0.72         109
 macro avg         0.72    0.73    0.71         109
 weighted avg      0.74    0.72    0.71         109
```

Kết quả

Máy học không rút trích đặc trưng:

report Naive bayes on test set					
	precision	recall	f1-score	support	
0	0.75	0.85	0.80	46	
1	0.46	0.60	0.52	10	
2	0.93	0.77	0.85	53	
accuracy			0.79	109	
macro avg	0.71	0.74	0.72	109	
weighted avg	0.81	0.79	0.79	109	

report KNN on test set					
	precision	recall	f1-score	support	
0	0.90	0.78	0.84	46	
1	0.86	0.60	0.71	10	
2	0.84	0.98	0.90	53	
accuracy			0.86	109	
macro avg	0.87	0.79	0.82	109	
weighted avg	0.87	0.86	0.86	109	

report svm on test set					
	precision	recall	f1-score	support	
0	0.96	0.98	0.97	46	
1	1.00	0.70	0.82	10	
2	0.96	1.00	0.98	53	
accuracy			0.96	109	
macro avg	0.97	0.89	0.92	109	
weighted avg	0.96	0.96	0.96	109	

Kết quả

Lập trình truyền thống:

report test set	precision	recall	f1-score	support
0	0.92	0.98	0.95	46
1	1.00	0.80	0.89	10
2	0.98	0.96	0.97	53
accuracy			0.95	109
macro avg	0.97	0.91	0.94	109
weighted avg	0.96	0.95	0.95	109

Nhận xét:

Kết quả dự đoán của model máy học khi không rút trích đặc trưng cao hơn khi đã được rút trích cho thấy việc rút trích đặc trưng chưa tốt (có thể do các ngưỡng màu sắc).

Khó khăn & Hướng phát triển

Khó khăn:

- Dữ liệu bởi nhiều bởi ánh sáng khá nhiều.
- Hạn chế về các hướng tiền xử lí dữ liệu (Dữ liệu chỉ được làm mờ để khử bớt nhiễu), do nhóm khảo sát các phương pháp này chưa tốt.
- Vấn đề lệch dữ liệu ảnh hưởng rất lớn trong quá trình máy học (dữ liệu chủ yếu là đèn xanh).
- Chưa có kinh nghiệm trong khâu thu thập và gán nhãn dữ liệu dẫn đến có nhiều dữ liệu nhiễu.

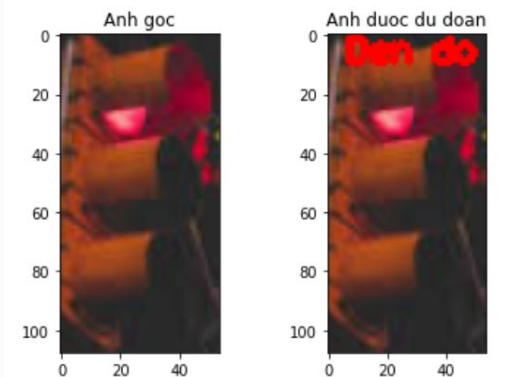
Hướng phát triển:

- Tìm hiểu thêm phương pháp xét vị trí của các màu trong đèn (vì màu đèn giao thông thường cố định theo thứ tự đỏ - vàng - xanh từ trên xuống dưới, không áp dụng được với các đèn nằm ngang).
- Sử dụng Deep learning để cải thiện độ chính xác (Các mạng phân loại như CNN, VGG-16).
- Phát triển bài toán rộng ra, không chỉ phân loại màu đèn nữa, mà sẽ vừa phát hiện vị trí đèn giao thông, vừa nhận diện màu của đèn giao thông đó, sử dụng các mô hình detect hiện đại như (YOLOv3, Detectron, ...).
- Áp dụng phương pháp tăng cường dữ liệu

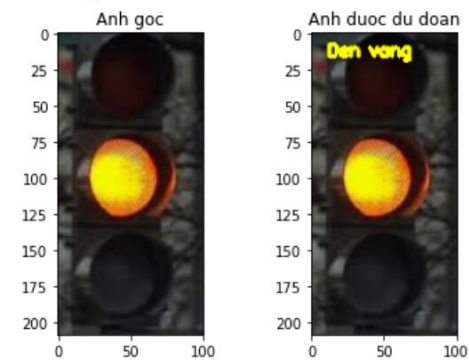
Demo

Link video demo: <https://bitly.com.vn/dhv9v>

Den do



Den vang



Den xanh

