**TRƯỜNG ĐẠI HỌC HÀNG HẢI VIỆT NAM  
KHOA CÔNG NGHỆ THÔNG TIN**

**-----\*\*\*-----**

****

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN “THỊ GIÁC MÁY TÍNH”**

***Đề tài:***

***Xây dựng hệ thống nhận dạng biển số xe ô tô***

***GVHD: Nguyễn Hữu Tuân***

***Sinh viên thực hiện: Phạm Văn Dương – Mã SV: 75056***

***Hải Phòng, tháng 11 năm 2020***

**TRƯỜNG ĐẠI HỌC HÀNG HẢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**-----\*\*\*-----**

**BÀI TẬP LỚN**

**HỌC PHẦN: THỊ GIÁC MÁY TÍNH**

**Mã đề tài: 02**

1. **Tên đề tài**

***Xây dựng hệ thống nhận dạng biển số xe ô tô***

1. **Mục đích**

Tìm hiểu và xây dựng chương trình nhận dạng biển số xe ô tô

1. **Công việc cần thực hiện**

* Tìm hiểu các vấn đề liên quan
* Đề xuất phương án và xây dựng chương trình
* Làm báo cáo bài tập lớn
* Bảo vệ bài tập lớn

1. **Yêu cầu**

* Kết quả làm bài tập lớn: Báo cáo bài tập lớn
* Báo cáo bài tập lớn phải được trình bày theo mẫu quy định (kèm theo), báo cáo có thể kết xuất thành tệp định dạng PDF và nộp qua email

1. **Tài liệu tham khảo**
   * Giáo trình Công nghệ Thị Giác Máy Tính – Nguyễn Hữu Tuân – Trường Đại học Hàng hải Việt Nam

**Hải Phòng, tháng 11 năm 2020**

**NGƯỜI HƯỚNG DẪN**

**Nguyễn Hữu Tuân**

Mục lục

[Lời nói đầu 6](#_Toc59049545)

[CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI 7](#_Toc59049546)

[1.1. Mục đích – Ngữ cảnh áp dụng: 7](#_Toc59049547)

[1.2. Phạm vi thực tế của bài toán: 7](#_Toc59049548)

[1.3. Các phương pháp có thể áp dụng: 7](#_Toc59049549)

[1.4. Cách thức đánh giá kết quả: 8](#_Toc59049550)

[CHƯƠNG 2: ÁP DỤNG CÁC KĨ THUẬT XỬ LÝ ẢNH CHO BÀI TOÁN 9](#_Toc59049551)

[2.1. Mô hình tổng quát của bài toán 9](#_Toc59049552)

[2.2. Mô hình chi tiết của bài toán 10](#_Toc59049553)

[2.3. Chi tiết thuật toán sử dụng 11](#_Toc59049554)

[2.3.1.Thuật toán K-NEAREST NEIGHBOR 11](#_Toc59049555)

[2.4. Các quá trình xử lý trong hệ thống 13](#_Toc59049556)

[2.4.1. Xác định vùng chứa biển số xe và tách biển số 13](#_Toc59049557)

[a. Xác định vùng chứa biển số xe ô tô 13](#_Toc59049558)

[b. Tách vùng biển số 13](#_Toc59049559)

[2.4.2 Phân đoạn ký tự 14](#_Toc59049560)

[CHƯƠNG 3: CÀI ĐẶT VÀ ĐÁNH GIÁ KẾT QUẢ HỆ THỐNG 23](#_Toc59049561)

[**3.1.** **Ngôn ngữ lập trình, nền tảng lập trình, thư viện sử dụng** 23](#_Toc59049562)

[**3.1.1.** **Thư viện sử dụng – OpenCV: Numpy, OpenCV2, các Module Math, OS, Random** 24](#_Toc59049563)

[3.2 . Đánh giá kết quả : 25](#_Toc59049564)

[3.3. Dữ liệu : 25](#_Toc59049565)

[3.4. Kết quả: 25](#_Toc59049566)

[CHƯƠNG 4: KẾT LUẬN 26](#_Toc59049567)

[4.1. Tóm lược quá trình thực hiện: 26](#_Toc59049568)

[4.2. Ưu / nhược điểm – Hướng phát triển 26](#_Toc59049569)

[4.2.1 Ưu điểm: 26](#_Toc59049570)

[4.2.2 Nhược điểm: 27](#_Toc59049571)

[4.2.3 Hướng phát triển: 27](#_Toc59049572)

# Lời nói đầu

Trong xu thế của cuộc cách mạng công nghiệp 4.0 đã kéo theo sự phát triển nhanh của các ngành khoa học và công nghệ. Trí tuệ nhân tạo (AI – Artificical Intelligence) là một lĩnh vực phát triển vượt bậc và tạo ra nhiều ứng dụng phục vụ cuộc sống của con người. Trên thế giới và ở Việt Nam đã có nhiều công trình nghiên cứu các hệ thống thông minh ứng dụng trí tuệ nhân tạo như bệnh viện thông minh, giao thông thông minh, nhà thông minh, xe tự lái,… Trong đó, ứng dụng nhận dạng biển số xe ô tô được sử dụng rộng rãi trong các bãi giữ xe cấp phát vé tự động, các trạm thu phí, hệ thống theo dõi phương tiện tham gia giao thông và phát hiện các phương tiện vi phạm

# CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

* 1. **Mục đích – Ngữ cảnh áp dụng:**

- Mục đích: Áp dụng các kĩ thuật xử lý ảnh và thị giác máy tính để nhận dạng biển số xe từ một hình được chụp từ thiết bị camera hoặc video của xe thành dạng văn bản để phục vụ cho nhu cầu kiểm tra, tìm kiếm, truy tìm nguồn gốc

- Ngữ cảnh áp dụng: Bài toán áp dụng đối với một cá nhân sử dụng máy tính, với hình ảnh các biển số xe ô tô được chụp lại từ camera hoặc từ một video

* 1. **Phạm vi thực tế của bài toán:**
* Trạm thu phí
* Bãi đậu xe
* Trạm kiểm soát
* Hệ thống xử phạt giao thông
  1. **Các phương pháp có thể áp dụng:**

Để giải quyết được bài toán nhận dạng biển số xe ô tô,yêu cầu đặt ra là phải giải quyết được 3 bài toán con.

-Bài toán con thứ nhất: xác định vùng chứa biển số xe và tách biển số

-Bài toán con thứ hai: phân đoạn từng ký tự

-Bài toán con thứ ba: nhận dạng ký tự.

## Cách thức đánh giá kết quả:

* Dữ liệu , tiêu chí đánh giá : dựa trên những thông số , quá trình đọc ảnh, kết quả vận hành của hệ thống dựa trên sự thành công trong việc nhận dạng, phát hiện , từ đó có thể đưa ra kết quả biển số xe một cách chinh xác
* Tốc độ, tài nguyên sử dụng : dựa trên sự kiểm nghiệm của hệ thống khi đang học tập để đưa ra kết quả
* Sự dễ dàng trong việc triển khai và vận hành : là cách thức thử nghiệm, phân tích để đưa ra thành quả của bài toán

# CHƯƠNG 2: ÁP DỤNG CÁC KĨ THUẬT XỬ LÝ ẢNH CHO BÀI TOÁN

* 1. **Mô hình tổng quát của bài toán**

* 1. **Mô hình chi tiết của bài toán**

Tách vùng biển số khả thi

Ảnh RGB của xe ô tô

Đầu vào (Video / File Ảnh)

Tìm và tách vùng ký tự

Phân đoạn từng kí tự

Giải thuật KNN để nhận diện kí tự

Nhận dạng biển số

* 1. **Chi tiết thuật toán sử dụng**

**2.3.1.Thuật toán K-NEAREST NEIGHBOR**

a. Định nghĩa:

K-nearest neighbor (KNN) là một trong những thuật toán học có giám sát đơn giản nhất trong Machine Learning.Ý tưởng của là KNN là tìm ra output của dữ liệu dựa trên thông tin của những dữ liệu traning gần nó nhất.

b. Quy trình làm việc của thuật toán KNN:

* Bước 1: xác định tham số K= số láng giềng gần nhất
* Bước 2: tính khoảng các đối tượng cần phân lớp với tất cả các đối tượng trong training data
* Bước 3: sắp xếp khoảng cách theo thứ tự tăng dần và xác định K láng giềng gần nhất với đối tượng cần phân lớp
* Bước 4: lấy tất cả các lớp của K láng giềng gần nhất
* Bước 5: dựa vào phần lớn lớp của K để xác định cho đối tượng cần phân lớp

c. Ưu điểm, nhược điểm của thuật toán

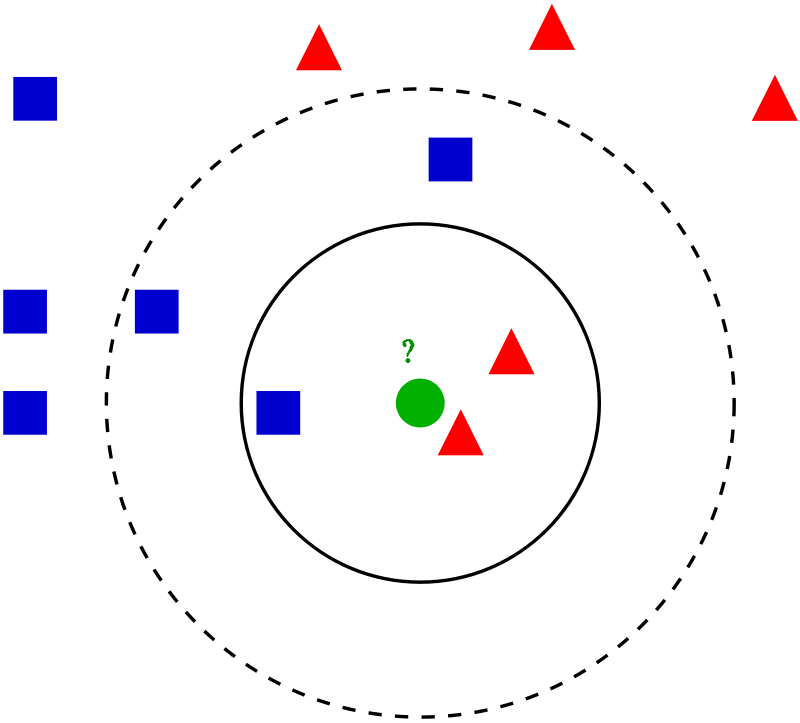
+Ưu điểm:

* + Dễ sử dụng và cài đặt
  + Việc dự đoán kết quả của dữ liệu mới dễ dàng
  + Độ phức tạp tính toán nhỏ

+Nhược điểm:

* KNN nhiễu dễ đưa ra kết quả không chính xác khi k nhỏ.
* Cần thời gian lưu training set, khi dữ liệu training và test tăng lên nhiều sẽ mất nhiều thời gian tính toán

d. Ví dụ minh họa:



Giả sử bài toán được đặt ra: mình thích màu hồng và mình cũng muốn biêt người bạn mình có thích màu hồng không. Qua thời gian tìm hiểu mình đã thu thập được một số dữ liệu và đã biểu hiện dưới dạng hình vẽ trên.

Ta dễ dàng nhìn thấy có hai loại: hình vuông biểu diễn cho những người thích màu hồng, tam giác biểu diễn những người không thích màu hồng, hình tròn là người bạn muốn biết có thích màu hồng không. Khoảng cách giữa chấm tròn và các điểm còn lại biểu diễn độ thân thiết của bạn đó với những người bạn

Phương pháp đơn giản nhất để kiểm tra xem bạn đó chơi thân với bạn nào nhất , tức là tìm xem điểm gần chấm tròn thuộc class nào (hình vuông hay tam giác). Từ hình trên ta thấy điểm gần chấm tròn là hình tam giác, do đó nó sẽ được phân vào lớp tam giác

Có một vấn đề trong phương pháp trên, xung quanh chấm tròn xuất hiện rất nhiều hình vuông nên việc lấy điểm gần nhất là chưa khả thi. Vì vậy, ta sẽ xét K điểm gần nhất. Giả sử lấy K=3, dựa theo hình trên ta thấy có hai hình tam giác và một hình vuông có khoảng cách gần chấm xanh nhất, do đó chấm tròn được phân vào lớp tam giác. Lấy K=7, ta có năm hình vuông và hai tam giác, lúc này chấm được xếp vào lớp hình vuông. Trường hợp lấy K=4, nhận thấy có hai hình vuông và hai tam giác, đây là trường hợp có điểm bằng nhau, với trường hợp này KNN sẽ xử lý bằng cách so sánh tổng khoảng cách của các hình gần nhất với điểm đang xét

Do sự xuất hiện của trường hợp có điểm bằng nhau, vì vậy ta thường chọn K là số lẻ. Đó cũng là ý tưởng của KNN

**2.4. Các quá trình xử lý trong hệ thống**

**2.4.1. Xác định vùng chứa biển số xe và tách biển số**

### a. Xác định vùng chứa biển số xe ô tô

Xác định vùng chứa biển số xe được chi thành 2 giai đoạn chính:

-Giai đoạn 1:Xác định vị trí của biển số xe trong ảnh đầu vào hoặc từ một video

-Giai đoạn 2: dùng các phương pháp để tách biển số xe ra

Xác định vị trí của biển số

Ảnh chỉ chứa biển số

Tách vùng biển số

Ảnh

**Sơ đồ tổng quát của khối xác định vùng chứa biển số xe**

### b. Tách vùng biển số

Tách vùng biển số được chia làm 2 bước:

-Bước 1: xác định vùng chứa ký tự,sau đó loại bỏ các vùng không có ký tự vì đó không phải là vùng chứa biển số

-Bước 2: Tách biển số ra khỏi vùng trên

**Sơ đồ tách biển số**

**Ảnh RGB**

Biến ảnh RGB thành ảnh xám và tiến hành tách ngưỡng để tìm tất cả các biên(Contour)

Tìm vùng màu trắng

Tìm kiếm tất cả vùng chứa ký tự

Tìm tất cả đường biên là các ký tự khả thi trong ảnh và cho vào danh sách

Tìm vùng kí tự phù hợp

-Tiến hành so sánh kí tự hiện tại với kí tự trong danh sách

- Từ danh sách lớn các vùng kí tự trong ảnh biến nó thành danh sách các vùng kí tự phù hợp

Khoanh vùng các vùng kí tự phù hợp và tiến hành cắt vùng ảnh đó

Trích xuất biển số

**2.4.2 Phân đoạn ký tự**

Sau khi xác định được vị trí các vùng biển số khả thi, chúng ta tiến hành việc phân đoạn ký tự của biển số xe:

-Giai đoạn 1: Biến ảnh biển số thành ảnh xám và tiếp tục tách ngưỡng

Sau khi đã có được hình ảnh các vùng biển số xe khả thi được tách ra,chúng ta sẽ gọi hàm Preprocess() để tìm vùng trắng của ký tự để xác định được các đường viền bao quanh ký tự

-Giai đoạn 2: Tìm vùng đối tượng

+Bước 1: Gọi hàm findCharInPlate() để tìm các tất cả Contour hay là đường viền bao quanh ký tự ,sau đó ta dùng một hình chữ nhật có kích thước tương đương với ký tự để tiếp tục nhận dạng các ký tự riêng biệt

+Bước 2: Gọi hàm findListOfListMatchingChar để tập hợp lại danh các ký tự phù hợp

Bước 3:Gọi hàm removeInnerOverlapping để xóa bỏ các ký tự chồng lên nhau

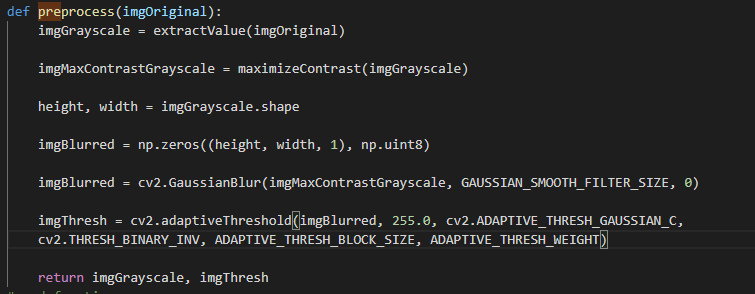
-Giai đoạn 3: Nhận dạng

+Với mỗi biển số khả thi, tìm danh sách dài nhất của các kí tự phù hợp tiềm năng kết hợp thuật toán K-Nearest Neighbor (KNN) để nhận diện ký tự trong chuỗi.

+Trích xuất ảnh của biển có chuỗi ký tự được công nhận nhiều nhất (biển số xe thực tế)

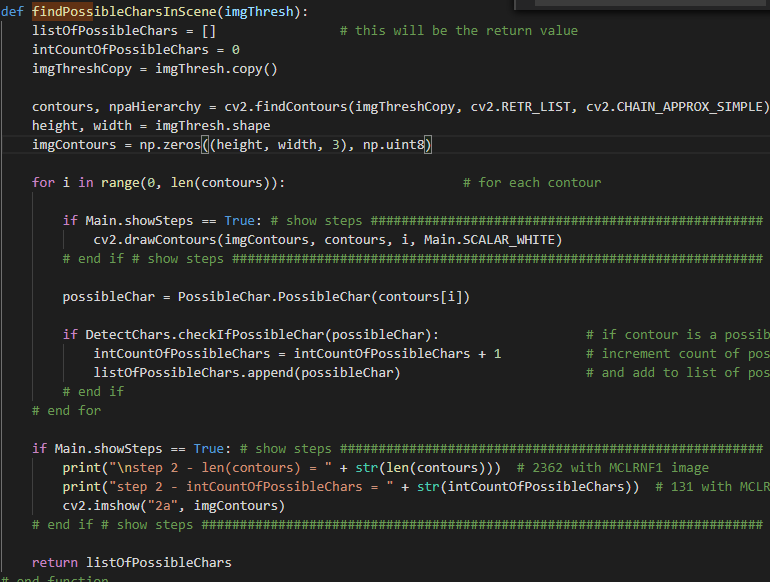
**2.5. Các hàm được sử dụng**

**-preprocess()**:biến ảnh thành ảnh xàm và tách ngưỡng ảnh

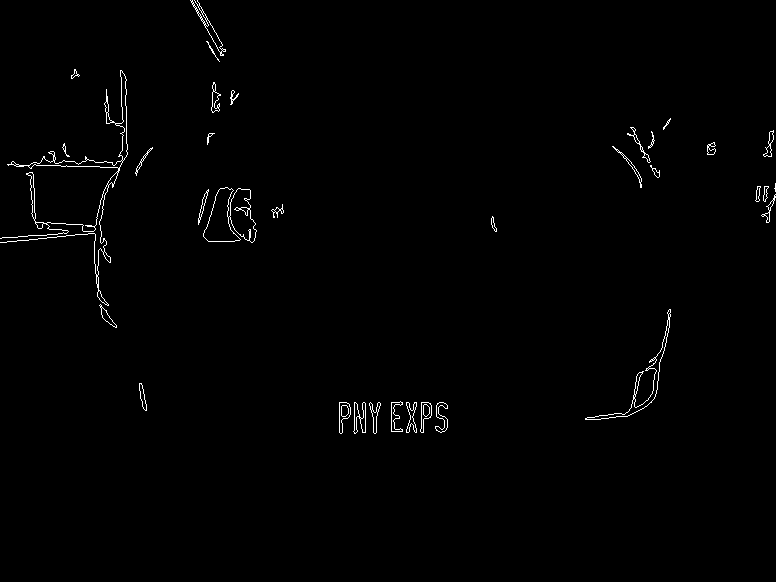
****



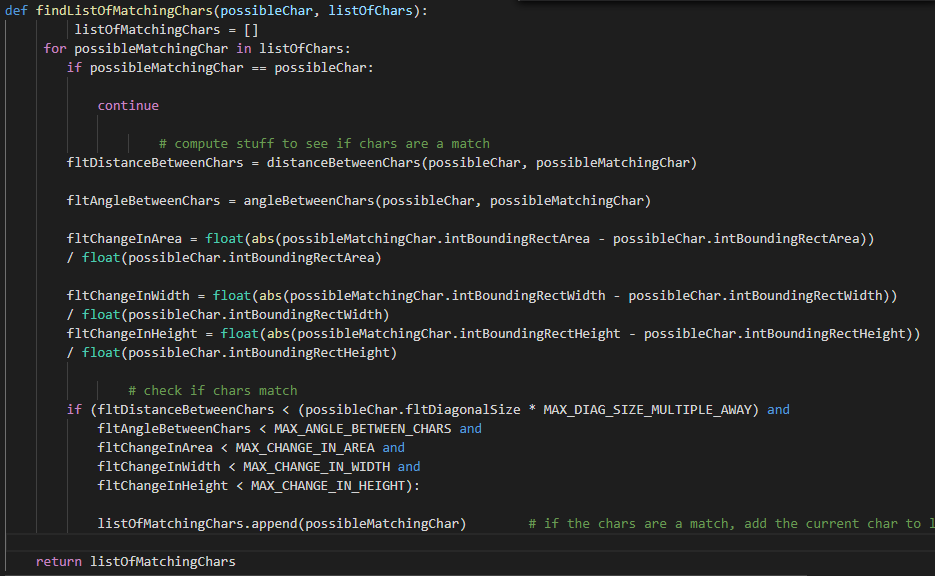
**-findPossibleCharsInScene()**:tìm tất cả các ký tự(đường viền) của ảnh và thêm vào danh sách,sau đó dùng phương pháp tính kích thước giới hạn của một hình chữ nhật có kích thước tương đương với các kí tự để tiếp tục nhận dạng kí tự

****



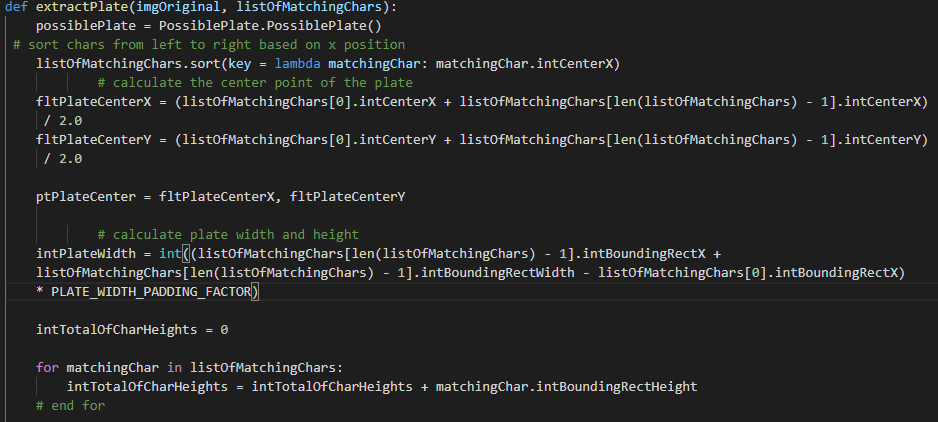


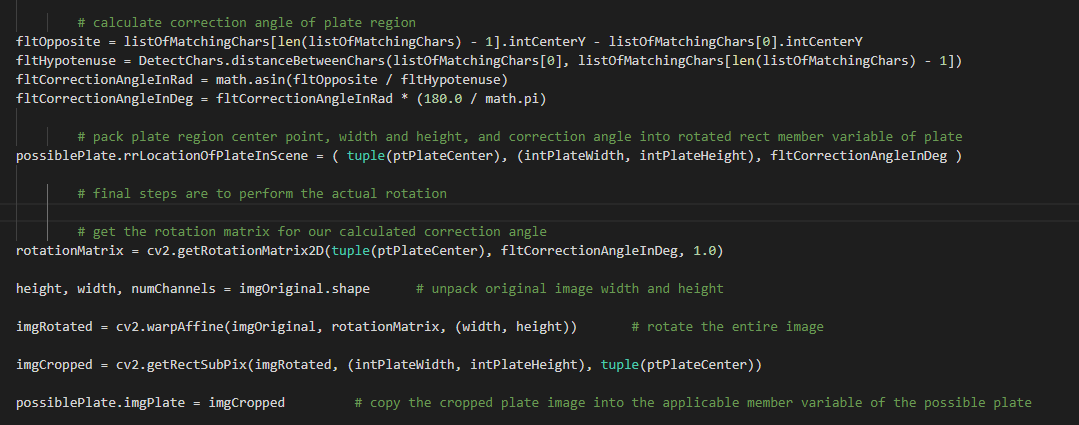
**-findListOfListMatchingChars():**cho một kí tự khả thi vào danh sách lớn các kí tự có thể,so sánh tất cả các kí tự trong danh sách lớn phù hợp với các kí tự hiện tại và trả về các kí tự phù hợp dưới dạng danh sách

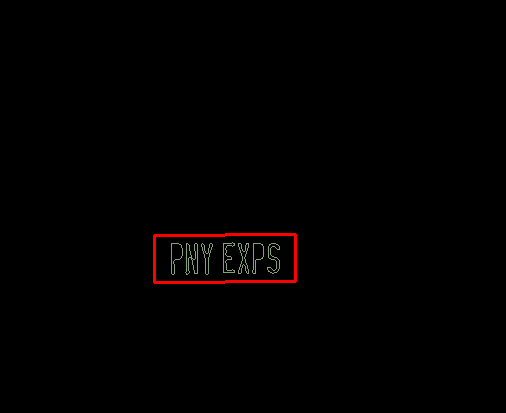
****

****

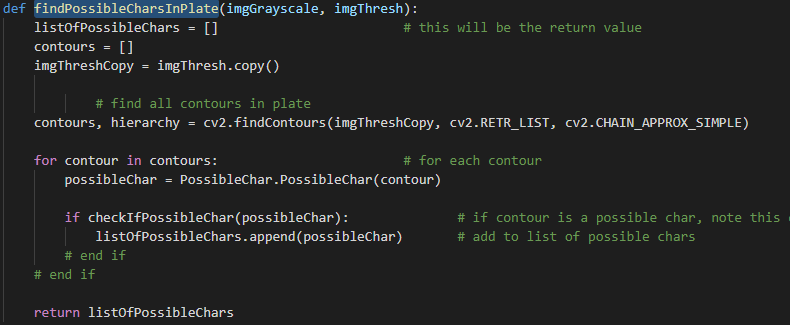
**-extractPlate():**Tìm được danh sách các vùng có thể là biển số

****

****

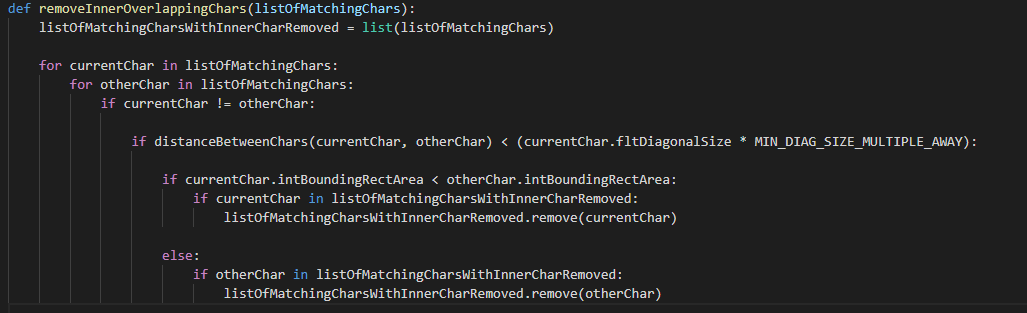
****

**-findPossibleCharsInPlate():**trả về các kí tự khả thi trong các vùng biển số được tìm thấy

****

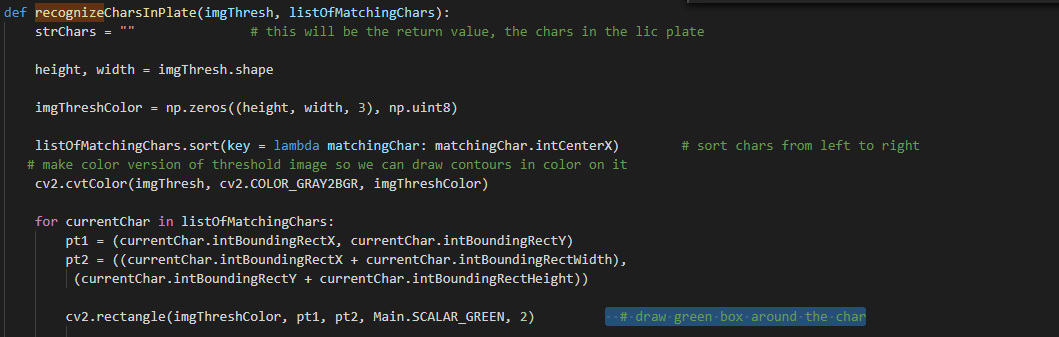
****

**-removeInnerOverlappingChars():**loại bỏ các kí tự chồng lên nhau,cho danh sách các kí tự phù hợp

****

****

**-recognizeCharsInPlate():**trả về các kí tự nhận dạng bằng cách vẽ hộp xanh lá cây tách biệt bao quanh từng kí tự

****

****

# CHƯƠNG 3: CÀI ĐẶT VÀ ĐÁNH GIÁ KẾT QUẢ HỆ THỐNG

* 1. **Ngôn ngữ lập trình, nền tảng lập trình, thư viện sử dụng**

**+)** *Ngôn ngữ lập trình* : python

+) *Visual Studio Code*: là một trình biên tập mã được phát triển bởi Microsoft dành cho Windows, Linux và macOS. Nó hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn.

+) *Thư viện sử dụng* : Numpy, OpenCV2, …

+ *Các Module* : OS, Math, Random

**- Đặc điểm của ngôn ngữ lập trình Python:**

***+ Ngôn ngữ lập trình đơn giản, dễ học:*** Python có cú pháp rất đơn giản, rõ ràng. Nó dễ đọc và viết hơn rất nhiều khi so sánh với những ngôn ngữ lập trình khác như C++, Java, C#.

***+ Miễn phí, mã nguồn mở:*** Có thể tự do sử dụng và phân phối Python, thậm chí là dùng nó cho mục đích thương mại. Vì là mã nguồn mở, chúng ta không những có thể sử dụng các phần mềm, chương trình được viết trong Python mà còn có thể thay đổi mã nguồn của nó. Python có một cộng đồng rộng lớn, không ngừng cải thiện nó mỗi lần cập nhật.

***+Khả năng tương thích:*** Các chương trình Python có thể di chuyển từ nền tảng này sang nền tảng khác và chạy nó mà không có bất kỳ thay đổi nào. Nó chạy liền mạch trên hầu hết tất cả các nền tảng như Windows, macOS, Linux.

***+Khả năng mở rộng và có thể nhúng:*** Chúng ta có thể dễ dàng kết hợp các phần code bằng C, C++ và những ngôn ngữ khác (có thể gọi được từ C) vào code Python.

***+Thư viện tiêu chuẩn lớn để giải quyết những tác vụ phổ biến***: Python có một số lượng lớn thư viện tiêu chuẩn giúp cho công việc lập trình trở nên dễ dàng hơn

***+Hướng đối tượng:*** Mọi thứ trong Python đều là hướng đối tượng

**- Lý do lựa chọn ngôn ngữ Python:**

+ Đây là ngôn ngữ có cú pháp vô cùng đơn giản

+ Lượng code phải viết ít hơn so với các ngôn ngữ khác

+ Cộng đồng hỗ trợ đồng đảo

+ Được thiết kế với ưu điểm dễ đọc,dễ nhớ,cấu trúc rõ ràng,thuận tiện cho người lập trình

* + 1. **Thư viện sử dụng – OpenCV: Numpy, OpenCV2, các Module Math, OS, Random**

**- OpenCV (Open Computer Vision)** là một thư viện mã nguồn mở hàng đầu cho xử lý về thị giác máy tính, machine learning, xử lý ảnh. OpenCV đươc viết bằng C/C++, vì vậy có tốc độ tính toán rất nhanh, có thể sử dụng với các ứng dụng liên quan đến thời gian thực.

* Ứng dụng của OpenCV**:**

+ Nhận dạng ảnh

+ Xử lý hình ảnh

+ Phục hồi hình ảnh/video

+ Thực tế ảo

+ Các ứng dụng khác

-**Module Math**: Mô-đun này cung cấp quyền truy cập vào các hàm toán học được xác định bởi tiêu chuẩn C

-**Module Random**: Mô-đun này thực hiện các trình tạo số giả ngẫu nhiên cho các phân phối khác nhau.

-**Module OS**: Mô-đun này cung cấp một cách di động để sử dụng chức năng phụ thuộc vào hệ điều hành

## 3.2 . Đánh giá kết quả :

Hệ thống có thể phát hiện và nhận diện được chính xác biển số xe ô tô hay không.

## 3.3. Dữ liệu :

-Tập dữ liệu dataset bao gồm khoảng *1.500* hình ảnh được lấy từ nhiều nguồn khác nhau như Dataset,Google…

## 3.4. Kết quả:

Tiến hành thử nghiệm 3 lần,mỗi lần sử dụng 100 ảnh đầu vào được lấy từ nhiều nguồn khác nhau,1 số hình ảnh từ biển số tự chụp.Kết quả có được như sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Số lần thử** | **Số hình ảnh Demo** | **Số hình ảnh xác định được vùng chứ biển số** | **Số hình ảnh tách được kí tự** | **Số hình ảnh nhận dạng chính xác** | **Số hình ảnh nhận dang không chính xác** |
| 1 | 100 | 80 | 72 | 70 | 30 |
| 2 | 100 | 85 | 80 | 77 | 23 |
| 3 | 100 | 90 | 90 | 89 | 11 |

Kết luận : Tốc độ xử lý của hệ thống nhanh,gọn nhưng vẫn còn thiếu xót độ chính xác

Một số hình ảnh kết quả minh họa:

# CHƯƠNG 4: KẾT LUẬN

## 4.1. Tóm lược quá trình thực hiện:

- Mục đích: Áp dụng các thuật toán xử lý ảnh và thị giác để có thể nhận diện một cách chính xác biển số xe của ô tô nhằm phục vụ cho nhu cầu kiểm tra, tìm kiếm, truy tìm nguồn gốc. Hệ thống phân tích, phát hiện và nhận dạng được biển số xe ô tô từ hình ảnh đầu vào.

- Phương pháp sử dụng:

* Để xây dựng mô hình này ta chia nó thành các bước như sau:
* Bước 1: Định vị được vùng biển số của xe ô tô từ một ảnh đầu vào
* Bước 2: Sau khi đã định vị được vùng biển số xe,tiến hành tìm kiếm các kí tự có trong biển số đó
* Bước 3: Nhận dạng các kí tự vừa tìm kiếm được ở bước 2 và tiến hành đọc ra biển số

## 4.2. Ưu / nhược điểm – Hướng phát triển

### 4.2.1 Ưu điểm:

+) Chương trình dễ sử dụng và cài đặt

+) Chương trình có thể xử lý tốt với tập dữ liệu nhiễu (do chuyển ảnh đầu vào thành ảnh xám )

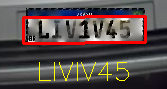
+) Độ phân tích, nhận dạng ở mức tương đối đáp ứng được sự chính xác (khoảng 70%)

### 4.2.2 Nhược điểm:

+) Tốc độ xử lý của chương trình còn chậm,do thuật toán KNN cần thời gian lưu training set, khi dữ liệu training và test tăng lên nhiều sẽ mất nhiều thời gian tính toán

+) Việc nhận dạng ký tự còn nhiều thiếu xót, chưa chính xác 100%

Cụ thể: Ký tự I ngắn viết hoa bị nhầm lẫn thành số 1 hoặc ngược lại,ký tự O nhầm lẫn với Q, Y thành V, …

### 4.2.3 Hướng phát triển:

-Tối ưu lại chương trình

-Sử dụng thư viện tensorflow để có thể nhận diện chính xác hơn

-Tinh chỉnh lại chương trình sao cho tương thích với mọi độ phân giải

.