

**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA: CÔNG NGHỆ THÔNG TIN**

**--------------------------**



**BÀI TẬP LỚN MÔN:**

**MỘT SỐ PHƯƠNG PHÁP TÍNH TOÁN MỀM**

**ĐỀ TÀI:**

**NHẬN DIỆN BIỂN SỐ XE BẰNG PHƯƠNG PHÁP KNN**

**GVHD: Ts.Vũ Việt Thắng**

**Danh sách thành viên (Nhóm 03):**

|  |  |
| --- | --- |
| **Vũ Văn Hùng** | **2017601398** |
| **Phạm Hoàng Tiến** | **2017601450** |
| **Phạm Văn Yên** | **2017600216** |

***Hà Nội, tháng 12 năm 2020***

MỤC LỤC

[LỜI NÓI ĐẦU 2](#_Toc60217919)

[CHƯƠNG 1: MỤC TIÊU ĐỀ TÀI. 4](#_Toc60217920)

[1. GIỚI THIỂU TỔNG QUAN VỀ ĐỀ TÀI. 4](#_Toc60217921)

[2. MỤC TIÊU. 4](#_Toc60217922)

[CHƯƠNG 2: TÌM HIỂU VỀ THUẬT TOÁN KNN. 5](#_Toc60217923)

[1. GIỚI THIỂU VỀ THUẬT TOÁN KNN. 5](#_Toc60217924)

[2. QUY TRÌNH LÀM VIỆC CỦA THUẬT TOÁN KNN. 6](#_Toc60217925)

[3. ƯU, NHƯỢC ĐIỂM CỦA KNN. 6](#_Toc60217926)

[3.1. ƯU ĐIỂM. 6](#_Toc60217927)

[3.2. NHƯỢC ĐIỂM. 6](#_Toc60217928)

[4. GIỚI THIỆU MODULE TRAINNING DATA. 7](#_Toc60217929)

[CHƯƠNG 3. QUY TRÌNH THỰC HIỆN SẢN PHẨM. 9](#_Toc60217930)

[1. CÁC BƯỚC THỰC HIỆN. 9](#_Toc60217931)

[1.1. TÁCH LẤY KHUNG CHỨA MÃ BIỂN SỐ XE. 10](#_Toc60217932)

[1.2. TÁCH LẤY TỪNG KÝ TỰ TRONG KHUNG. 11](#_Toc60217933)

[2. QUY TRÌNH. 12](#_Toc60217934)

[2.1. TIỀN XỬ LÝ ẢNH. 12](#_Toc60217935)

[2.2. TÌM NHỮNG KÝ TỰ CÓ THỂ LÀ BIỂN SỐ. 14](#_Toc60217936)

[2.3. TÌM CHUỖI KÝ TỰ PHÙ HỢP. 15](#_Toc60217937)

[2.4. TRÍCH XUẤT LẤY KHUNG CHỨA CHUỖI KÝ TỰ PHÙ HỢP NHẤT. 16](#_Toc60217938)

[2.5. TIỀN XỬ LÝ BIỂN SỐ. 18](#_Toc60217939)

[2.6. TÌM CHUỖI KÝ TỰ PHÙ HỢP TRONG KHUNG. 19](#_Toc60217940)

[2.7. NHẬN DIỆN KÝ TỰ TRONG KHUNG. 21](#_Toc60217941)

[CHƯƠNG 4. SO SÁNH VỚI MLP 23](#_Toc60217942)

[1. SO SÁNH VỚI MLP. 23](#_Toc60217943)

[1.1. MULTI-LAYER PERCEPTRON (MLP). 23](#_Toc60217944)

[KẾT LUẬN. 24](#_Toc60217945)

[1. VỀ KNN. 24](#_Toc60217946)

[2. HƯỚNG PHÁT TRIỂN. 24](#_Toc60217947)

[TÀI LIỆU THAM KHẢO. 24](#_Toc60217948)

# LỜI NÓI ĐẦU

*Trong quá trình học môn xử lý ảnh chúng ta không chỉ quan tâm đến lý thuyết và những bài toán khô khan trên giảng đường, mà còn phải mang những kiến thức mang tính hàn lâm đó vào thực tế để có thể nắm rõ ý nghĩa của môn học khi áp dụng vào thực tiễn.*

*Vì lý do đó nhóm chúng tôi quyết định tìm hiểu và nghiên cứu về phần mềm lập trình bằng ngôn ngữ PYTHON và thư viện OPenCV để ứng dụng vào lĩnh vực xử lý ảnh và xử lý video. Qua những bài tập từ dễ đến khó, chúng tôi đã tích luy được một chút kiến thức về lĩnh vực xử lý ảnh, vì vậy nhóm đã quyết đỉnh thực hiển Tiểu Luận của môn học là đề tài.*

*“ NHẬN DIỆN BIỂN SỐ XE QUA ẢNH CHỤP”*

*Dưới đây là phương pháp và quá trình thực hiện đề tài.*

*Xin cảm ơn đã theo dõi !!!*

# CHƯƠNG 1: MỤC TIÊU ĐỀ TÀI.

## GIỚI THIỂU TỔNG QUAN VỀ ĐỀ TÀI.

* Trong thời đại ngày nay mọi việc con người có thể làm thì hầu như máy móc cũng có thể làm được. Đó là nhờ những người lập trình viên đã biến những chiếc máy tính và những ngoại vi của nó trở thành những hệ thống thông minh có thể thực hiện bất cứ công việc gì.
* Đơn giản như công việc gửi xe, chỉ vài năm trước đây khi đi đến bất kì bãi gửi xe nào chúng ta cũng thấy tất cả xe đều phải được nhân viên coi xe gi bằng giấy, việc này là vô cũng bất tiện và nhiều rủi ro như giấy rất dễ bị hư hao, rách hoặc dính nước… Không những vậy việc ghi giấy không thể tránh khỏi sai sót khi người nhân viên coi xe một thoáng mất tập trung và ghi sai, vì vậy rất mất thời gian.
* Nhưng giờ đây đó không còn là vấn đề khi chúng ta có công nghệ nhận diện biển số xe bằng thẻ từ. công việc này gồm 3 giai đoạn chính đó là Lấy hình ảnh có chứa biển số xe, xử lý ảnh đó để lấy đoạn mã số ghi trên biển số, cuối cùng là lưu chúng vào thẻ.
* Trong đó giai đoạn quan trọng nhất đó chính là thu thập đoạn mã biển số xe từ hình ảnh thu được.
* Trong đề tài này chúng tôi sẽ trình bày phương pháp làm sao để có thể nhận diện được đoạn mã đó.

## MỤC TIÊU.

* Trong quá trình đề tài nhóm đề ra 2 mục tiêu chính.
* Có kiến thức cơ bản về lập trình Python. Vì đây là ngôn ngữ vô cùng phổ biến và quan trọng, nhất là trong lĩnh vực AI và Machine Learning.
* Tìm hiểu thư viện OpenCV qua đó ứng dụng nó nhận diện biển số xe một cách tương đối chính xác.

# CHƯƠNG 2: TÌM HIỂU VỀ THUẬT TOÁN KNN.

## GIỚI THIỂU VỀ THUẬT TOÁN KNN.

* KNN (K-Nearest Neighbors) là một trong những thuật toán học có giám sát đơn giản nhất được sử dụng nhiều trong khai phá dữ liệu và học máy.
* Ý tưởng của thuật toán này là nó không học một điều gì từ tập dữ liệu học (nên KNN được xếp vào loại lazy learning), mọi tính toán được thực hiện khi nó cần dự đoán nhãn của dữ liệu mới.
* Lớp (nhãn) của một đối tượng dữ liệu mới có thể dự đoán từ các lớp (nhãn) của k hàng xóm gần nó nhất. Hay ngắn gọn ý tưởng là tìm ra output của dữ liệu dựa trên thông tin của những dữ liệu training gần nó nhất.
* Đầu vào (Input):
* Thuật toán có 2 đầu vào:
* Một là tập các dữ liệu đã biết trước kiểu (loại) của từng dữ liệu (hay còn gọi là tập huấn luyện - training set).
* Đầu vào thứ 2 là dữ liệu, chúng ta chưa biết kiểu(loại) dữ liệu đó.
* Đầu ra (Output):
* Đầu ra của thuật toán KNN là kiểu dữ liệu của đầu vào thứ 2.

## QUY TRÌNH LÀM VIỆC CỦA THUẬT TOÁN KNN.

* **Bước 1**: Xác định tham số K = số láng giềng gần nhất.
* **Bước 2**: Tính khoảng cách 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 lớp cho đối tượng cần phân lớp.

## ƯU, NHƯỢC ĐIỂM CỦA KNN.

### ƯU ĐIỂM.

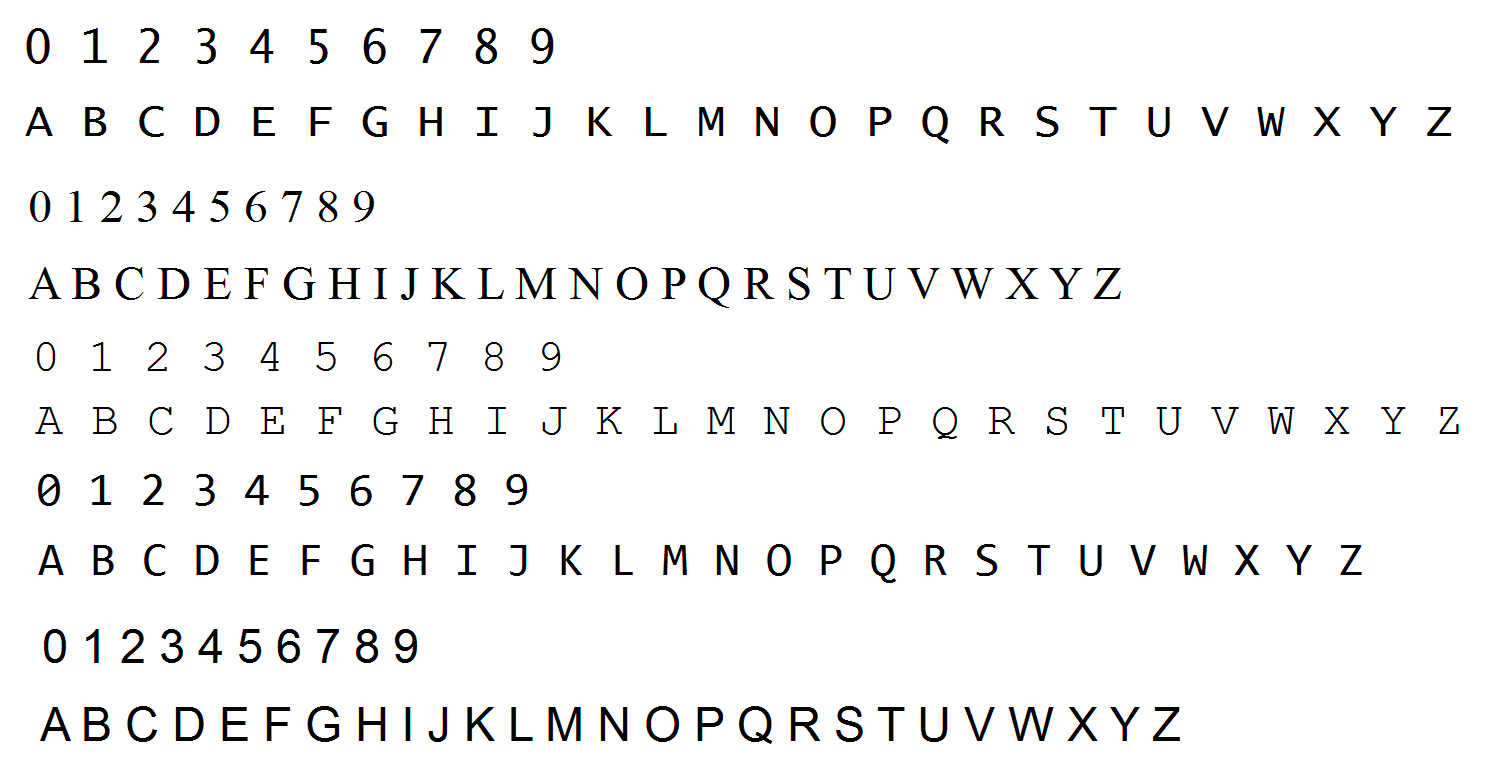
* Dễ sử dụng và cài đặt.
* Độ phức tạp tính toán nhỏ.
* Việc dự đoán kết quả của dữ liệu mới rất đơn giản.
* Không cần giả sử gì về phân phối của các lớp.

### NHƯỢC ĐIỂM.

* KNN rất nhạy cảm với nhiễu khi K nhỏ.
* Như đã nói, KNN là một thuật toán mà mọi tính toán đều nằm ở khâu “**Test**”. Trong đó việc tính khoảng cách tới từng điểm dữ liệu trong “**Training Set**” sẽ tốn rất nhiều thời gian, đặc biệt là với các cơ sở dữ liệu có số chiều lớn và có nhiều điểm dữ liệu. Với K càng lớn thì độ phức tạp cũng sẽ tăng lên. Ngoài ra, việc lưu toàn bộ dữ liệu trong bộ nhớ cũng ảnh hưởng tới hiệu năng của KNN.
* Với K nhỏ dễ gặp nhiễu dẫn tới kết quả đưa ra không chính xác.
* Cần nhiều thời gian để thực hiện do phải tính toán khoảng cách với tất cả các đối tượng trong tập dữ liệu.
* Cần chuyển đổi kiểu dữ liệu thành các yếu tố định tính.

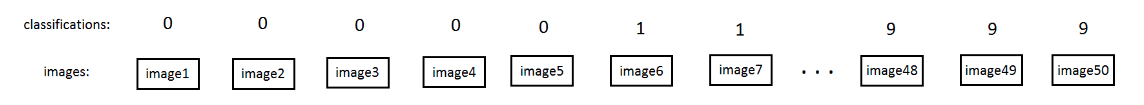
## GIỚI THIỆU MODULE TRAINNING DATA.

* Thực hiện code module dùng để trainning các tập dữ liệu với các chữ cái từ A-Z và các chữ số từ 0-9.



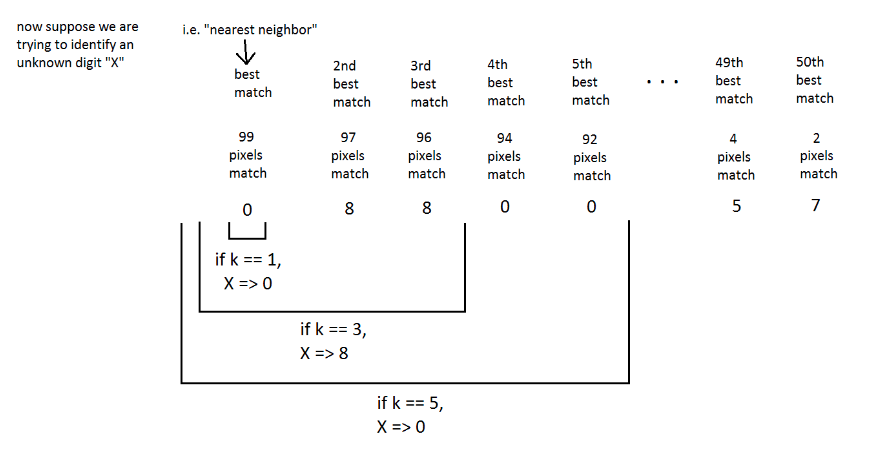
***Hình 1: Các tập dữ liệu đặc trưng dùng để trainning***

* Với các kiểu chữ đặc trưng cho biển số xe. Mỗi chữ cái hay chữ số đều có 5 trặc trưng khác nhau trong bộ dữ liệu trainning này.
* Giả sử, chúng ta nhận ra các chữ số từ 0-9 và mỗi chữ số đều có 5 ảnh đặc trưng vậy ta có tất cả: 5 (đặc trưng) \* 10 (chữ số) = 50 hình ảnh để đào tạo
* Giả sử kích thước mỗi hình ảnh là 10x10 pixel, tức là mỗi ảnh sẽ có tổng kích thước tối đa là 100px. Sau khi đào tạo xong sẽ tạo ra 2 cấu trúc dữ liệu song song:
* Group/Classification: là một bộ số cho biết hình ảnh tương ứng thuộc phân loại nào. Một đối tượng phân loại bằng một phiếu bầu nhiều người láng giềng của nó với đối tượng đặc trưng phổ biến nhất trong số k cụm láng giềng gần nhất với nó.
* A set of images/Flattened\_images: là một bộ hình ảnh đã được đào tạo, là những giá trị thuộc tính của đối tượng, giá trị này được coi là giá trị trung bình của k lân cận gần nhất.
* Mô tả thuật toán để tìm ra các tập dữ liệu đặc trưng



***Hình 2: Ví dụ mô tả thuật toán tìm tập dữ liệu đặc trưng***

* Ví dụ ta có 1 bộ dữ liệu như trên:
* Khi quá trình trainning hoàn tất



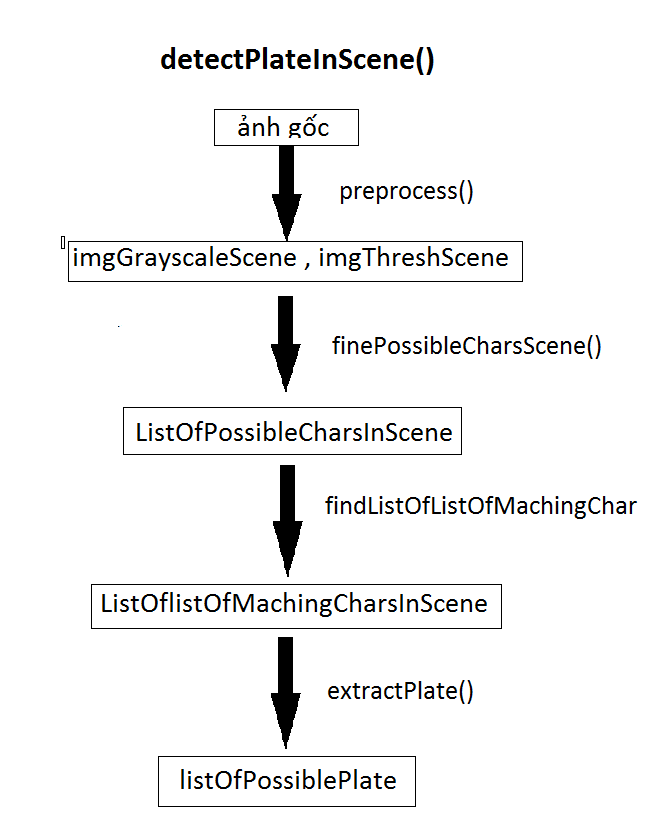
***Hình 3: Quá trình trainning***

* Giả dụ chúng ta đang cố gắng xác định 1 chữ số không xác định X thì:
* Khi số cụm k = 1 thì giá trị hàng xóm gần nhất của X = 0
* Khi số cum k = 3 thì giá trị hàng xóm gần nhất của X = 8
* Khi số cụm k = 5 Thì giá trị hàng xóm gần nhất của X = 0
* Tương tự với các trường hợp cụm còn lại (trong ví dụ này k\_min = 1 và k\_max = 50)
* Với dữ liệu đầu vào như hình 1 ta tạo ra những tập đặc trưng cho lớp và bộ thuộc tính của từng đối tượng với 5 đặc trưng khác nhau.
* (Classification/Set of image – Xem thêm ở file classifications.txt và flattened\_images.txt)

# CHƯƠNG 3. QUY TRÌNH THỰC HIỆN SẢN PHẨM.

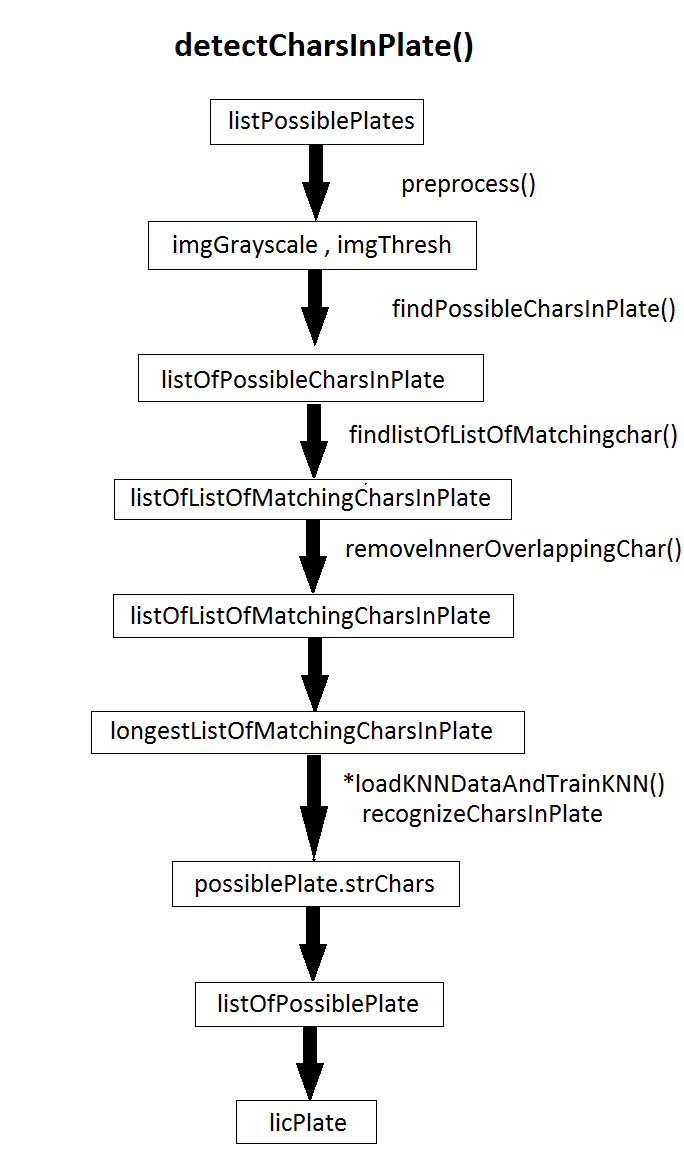
### CÁC BƯỚC THỰC HIỆN.

#### TÁCH LẤY KHUNG CHỨA MÃ BIỂN SỐ XE.

****

***Hình 4: Thứ tự các hàm thực hiện tách khung chứa mã biển số xe***

#### TÁCH LẤY TỪNG KÝ TỰ TRONG KHUNG.

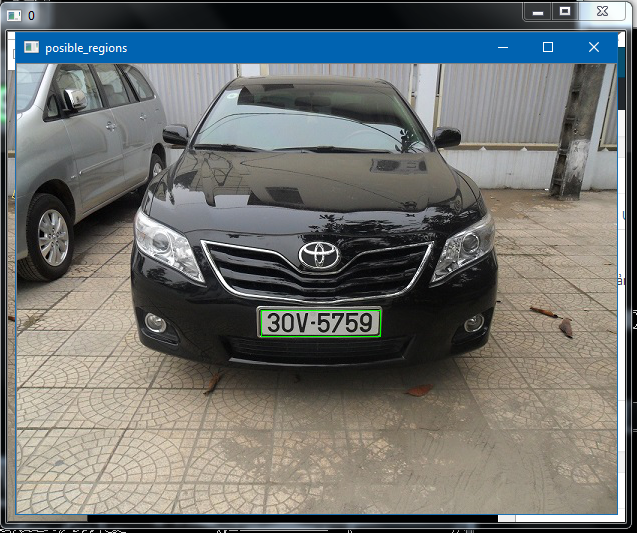
****

***Hình 5: Thứ tự các hàm thực hiện tách từng ký tự trong khung***

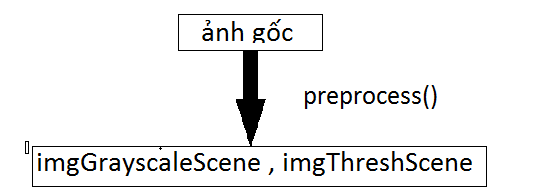
### QUY TRÌNH.

#### TIỀN XỬ LÝ ẢNH.

* Từ ảnh gốc ban đầu ta thực hiện chuyển đổi ảnh sang ảnh xám đồng thời đổi từ hệ màu RGB => HSV
* Sau đó lấy ngưỡng ảnh và tối ưu hóa độ tương phản.
* **Kết quả sẽ được ảnh *imgGrayscaleScene (Ảnh xám) và imgThreshScene (Ảnh lấy ngưỡng)***



***Hình 6: Ảnh minh hoạ (Ảnh gốc)***

****

****

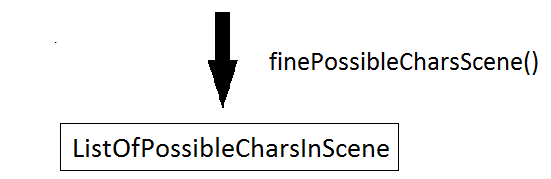
***Hình 7: imgGrayscaleScene (Ảnh xám)***

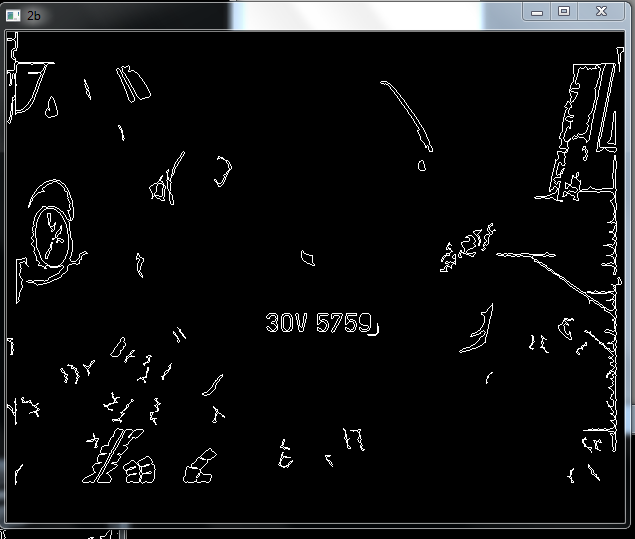
****

***Hình 8: imgThreshScene (Ảnh lấy ngưỡng)***

#### TÌM NHỮNG KÝ TỰ CÓ THỂ LÀ BIỂN SỐ.

* Ở bước này chúng ta tiếp tục loại bỏ những nét không phải là ký tự, chỉ giữ lại những nét có thể là kí tự để tiếp tục xử lý ở các bước tiếp theo.

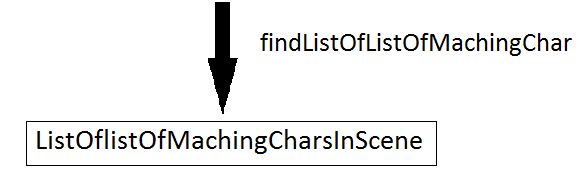


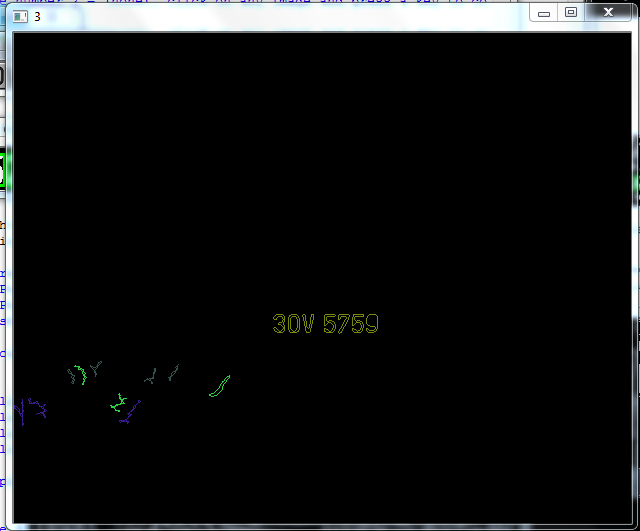
****

***Hình 9: Ảnh chứa những phần tử có thể là ký tự***

#### TÌM CHUỖI KÝ TỰ PHÙ HỢP.

* Sau khi có được những kí tự có thể ta cần đi tìm chuỗi những kí tự Phù hợp nhất để tạo thành một dãy biển số.

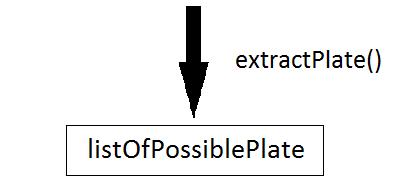


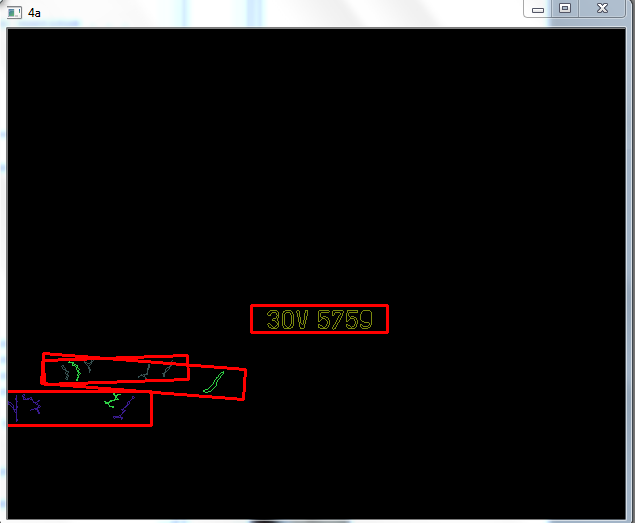
****

***Hình 10: Chuỗi ký tự phù hợp nhất***

#### TRÍCH XUẤT LẤY KHUNG CHỨA CHUỖI KÝ TỰ PHÙ HỢP NHẤT.

* Sau khi có được chuỗi kí tự phù hợp nhất như ở trên ta tiến hành vẽ một khung đỏ xung quanh nó.
* Sau đó tách lấy phần khung đỏ đó tương ứng bên ảnh gốc để được một ảnh độc lập, sau đó đem ảnh độc lập chỉ chứa kí tự phù hợp để tái tạo thành biển số đó đi xử lý tiếp.
* Trong quá trình tách nó sẽ nhận nhiều khung chứa kí tự có thể phù hợp vì thế nó sẽ đem tất cả những khung này đi xử lý để tìm ra biển số và cuối cùng chỉ có một khung duy nhất.
* Tạo ra biển số:

****

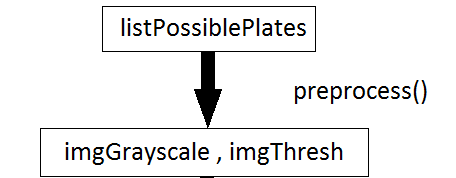
****

** **

***Hình 11: Kết quả trích xuất chuỗi ký tự***

#### TIỀN XỬ LÝ BIỂN SỐ.

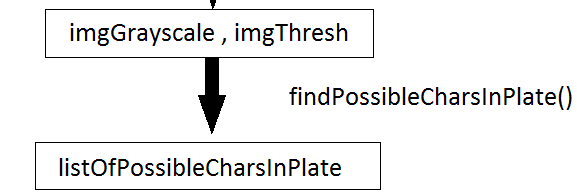
* Như vậy sau bước **1.2.4** chúng ta đã có được plate(khung) có chứa mã biển số phù hợp. Chúng ta sẽ tiếp tục thực hiện những công đoạn tiếp theo để có thể tách từng ký tự trong khung đó . đầu tiên là tiền xử lý.
* Sau bước tiền xử lý này cũng giống như ở bước **1.2.1** ta cũng sẽ nhận được ảnh xám và ảnh phân ngưỡng do sử dụng chung hàm preprocess().

****

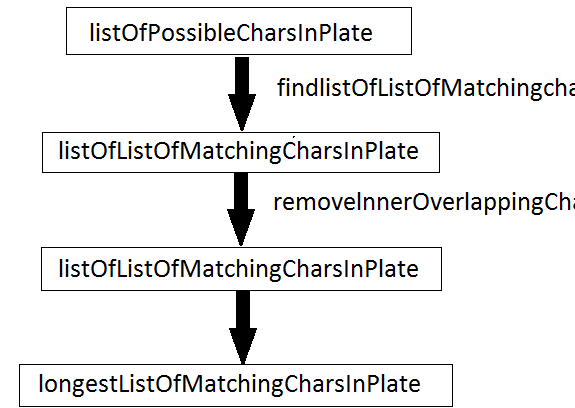
|  |  |
| --- | --- |
|  |  |
| ***Hình 12: Ảnh xám*** | ***Hình 13: Ảnh lấy ngưỡng*** |

#### TÌM CHUỖI KÝ TỰ PHÙ HỢP TRONG KHUNG.

* Chúng ta tiếp tục loại bỏ những phần không cần thiết và chỉ tách lấy phần chứa chuỗi kí tự phù hợp.
* Đồng thời tách lấy biên.

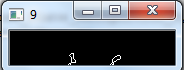
****

** **

****

** **

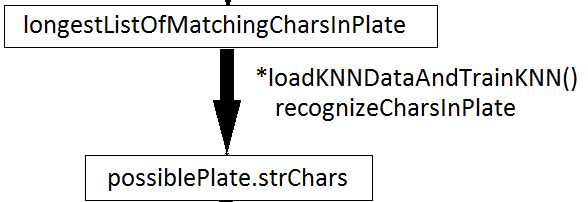
** **

** **

***Hình 14: Các quy trình thực hiện tìm chuỗi ký tự từ khung***

#### NHẬN DIỆN KÝ TỰ TRONG KHUNG.

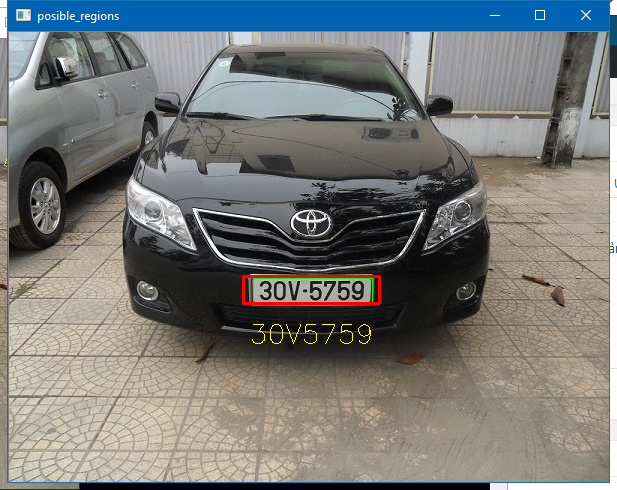
* Bước này các kí tự đã được tách riêng thành từng khung, công việc là nhận diện kí tự đó là kí tự gì.
* Sử dụng phương pháp KNN training với những text có sẵn.
* Đương nhiên khung chứa những nét không phù hợp bên trên sẽ không thể được nhận diện thành kí tự ở bước này.

****

****

***Hình 15: Nhận diện ký tự đã được tách riêng trong từng khung***

* Cuối cùng ta thu được chuỗi kí tự là mã số của biển số xe và in chuỗi kí tự này ra cửa sổ “ python shell “. Ngoài ra chúng ta cũng có thể viết chuỗi kí tự này lên ảnh gốc để quan sát và so sánh.

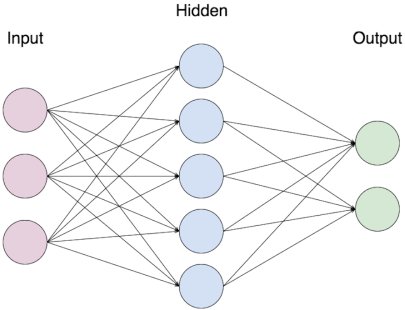
****

***Hình 16: Kết quả cuối cùng khi nhận diện thành công các ký tự của biển số xe***

# CHƯƠNG 4. SO SÁNH VỚI MLP

## SO SÁNH VỚI MLP.

### MULTI-LAYER PERCEPTRON (MLP).



* Multi-layer Perceptron (perceptron nhiều lớp): là tập hợp của các perceptron chia làm nhiều nhóm, mỗi nhóm tương ứng với một layer. Trong hình trên ta có một ANN với 3 lớp: Input layer (lớp đầu vào), Output layer (lớp đầu ra) và Hidden layer (lớp ẩn). Thông thường khi giải quyết một bài toán ta chỉ quan tâm đến input và output của một model, do vậy trong MLP nói riêng và ANN nói chung ngoài lớp Input và Output ra thì các lớp neuron ở giữa được gọi chung là Hidden (ẩn không phải là không nhìn thấy mà đơn giản là không quan tâm đến).

# KẾT LUẬN.

## VỀ KNN.

* Nhìn chung nhóm đã đạt được những mục tiêu đề ra ban đầu là hoàn thành đề tài nhận diện biển số xe và tìm hiểu về ngon ngữ lập trình Python, OpenCV, dù còn tương đối nhiều hạn chế và thiếu sót nhưng hi vọng nhóm sẽ cải thiện trong những lần thực hiện tiếp theo.
* Hạn chế:
* Hạn chế thứ nhất : code vẫn còn quá dài chưa thể tối ưu để ngắn hơn, gấy khó khăn trong quá trình đọc hiểu.
* Hạn chế thứ 2: nhận diện chưa chính xác tuyệt đối vẫn còn sai sót và chạy chỉ tương đối ổn định.
* Khi nhận diện bằng ảnh .png thì tương đối chính xác nhưng chuyển sang ảnh .jpg thì sai số rất nhiều và nhóm vẫn chưa tìm ra nguyên nhân
* Tốc độ xử lý và chính xác thấp hơn MLP

## HƯỚNG PHÁT TRIỂN.

* Từ đề tài ta có thể ứng dụng trực tiếp nó vào nhận diện biển số ở các bãi gửi xe.
* Nếu có thể phát triển lên có thể ứng dụng trong lĩnh vực trí tuệ nhân tạo như robot đọc chữ. Hoặc phát triển thành máy phát hiện kí tự hỗ trợ những người khiếm thính,mù lòa, …

# TÀI LIỆU THAM KHẢO.

* <https://www.slideshare.net/trongthuy3/luan-van-nghien-cuu-thuat-toan-knearest-neighbor-haym-9d> (Thuật toán KNN).
* <https://machinelearningcoban.com/2017/01/08/knn/> (Ưu, nhược về KNN).
* <https://www.kaggle.com/> (Tranning data).