**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN MÔN:**

**NHẬP MÔN LÝ THUYẾT NHẬN DẠNG**

**ĐỀ TÀI:**

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

**GVHD: TS.Trần Hùng Cường**

**Nhóm: 09**

**Danh sách thành viên**

|  |  |
| --- | --- |
| **Vũ Văn Hùng** | **2017601398** |
| **Phạm Văn Yên** | **2017600216** |
| **Phạm Hoàng Tiến** | **2017601450** |
| **Nguyễn Tùng Lâm** | **2017600911** |

***Hà Nội, tháng 12 năm 2020***

Mục lục

[GHI CHÚ VIẾT TẮT 3](#_Toc58022175)

[CHƯƠNG 1: TỔNG QUAN VỀ NHẬN DẠNG 4](#_Toc58022176)

[1. Đặt vấn đề 4](#_Toc58022177)

[2. Giới thiệu về nhận dạng 4](#_Toc58022178)

[3. Quy trình nhận dạng 4](#_Toc58022179)

[4. Ứng dụng nhận dạng 4](#_Toc58022180)

[CHƯƠNG 2: MỤC TIÊU ĐỀ TÀI 5](#_Toc58022181)

[1. Giới thiệu tổng quan về đề tài 5](#_Toc58022182)

[2. Mục tiêu 5](#_Toc58022183)

[CHƯƠNG 3: TÌM HIỂU THUẬT TOÁN KNN 6](#_Toc58022184)

[1. Giới thiệu về thuật toán KNN 6](#_Toc58022185)

[2. Quy trình làm việc của thuật toán KNN 6](#_Toc58022186)

[3. Ưu, nhược điểm của KNN 6](#_Toc58022187)

[- Ưu điểm 6](#_Toc58022188)

[- Nhược điểm của KN 6](#_Toc58022189)

[CHƯƠNG 4: THỰC HIỆN ĐỀ TÀI 7](#_Toc58022190)

[1. Module trainning data 7](#_Toc58022191)

[2. Quy trình thực hiện 9](#_Toc58022192)

[2.1. Các bước thực hiện 9](#_Toc58022193)

[2.1.1. Tách lấy khung chứa mã biển số xe 9](#_Toc58022194)

[2.1.2. Tách lấy từng ký tự trong khung 10](#_Toc58022195)

[2.2. Quy trình 11](#_Toc58022196)

[2.2.1. Tiền xử lý 11](#_Toc58022197)

[2.2.2. Tìm những ký tự có thể là biển số 13](#_Toc58022198)

[2.2.3. Tìm chuỗi ký tự phù hợp 14](#_Toc58022199)

[2.2.4. Trích xuất lấy khung chứa chuỗi ký tự phù hợp nhất 15](#_Toc58022200)

[2.2.5. Tiền xử lý 17](#_Toc58022201)

[2.2.6. Tìm chuỗi ký tự phù hợp trong khung 17](#_Toc58022202)

[2.2.7. Nhận diện ký tự trong khung 19](#_Toc58022203)

[KẾT LUẬN 21](#_Toc58022204)

[TÀI LIỆU THAM KHẢO 22](#_Toc58022205)

# GHI CHÚ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **Tên viết tắt** | **Tên đầy đủ** | **Tên tiếng việt** |
| KNN | K-Nearest Neighbors | Người hàng xóm gần |
| NLP | Natural Language Processing | Xử lý ngôn ngữ tự nhiên |
| AI | Artificial intelligence | Trí tuệ nhân tạo |
| CNTT |  | Công nghệ thông tin |

# CHƯƠNG 1: TỔNG QUAN VỀ NHẬN DẠNG

## Đặt vấn đề

* Ngày nay, với sự phát triển mạnh mẽ của công nghệ đã và đang thúc đẩy nhanh tróng sự phát triển của xã hội. Công nghệ trở thành một phần không thể thiếu trong cuộc sống hằng ngày, nó xuất hiện ở khắp mọi nơi từ trường học, công sở, đến các xí nghiệp hay công ty. Không những thế nó còn thực hiện được nhiều nhiệp vụ đơn giản cũng như phức tạp để phục vụ lợi ích của con người.
* Gần đây nhất là sự trỗi dậy thuật ngữ **AI (Artificial intelligence)**. Nó **có thể được định nghĩa như một ngành của khoa học máy tính liên quan đến việc tự động hóa các hành vi thông minh giống con người, được xây dựng trên nền tảng lý thuyết chắc chắn, có ứng dụng thực tiễn cao. Và ngày nay nó đã được ứng dụng khá nhiều vào lĩnh vực như IoT(Internet of Things), Đồ họa, Nhận dạng, Machine Learning, Y học...**
* **Có thể thấy rằng việc áp dụng công nghệ vào đời sống đã và đang mang lại lợi ích vô cùng lớn. Và trong khuôn khổ nội dung của báo cáo, nhóm em xin nó đến một công nghệ đang được áp dụng rất nhiều trong thực tế đó là “Nhận dạng”.**

## Giới thiệu về nhận dạng

* Nhận dạng mẫu là khả [1] năng phát hiện sự sắp xếp các của đặc điểm hoặc dữ liệu mang lại thông tin về một hệ thống hoặc tập dữ liệu nhất định. Trong lĩnh vực công nghệ, một mẫu có thể là chuỗi dữ liệu lặp lại theo thời gian có thể sử dụng để dự đoán xu hướng, xác định các đối tượng cụ thể trong ảnh, sự xuất hiện của các từ hoặc cụm từ để xử lý ngôn ngữ tự nhiên (NLP).

## Quy trình nhận dạng

## Ứng dụng nhận dạng

* Nhận dạng là điều cần thiết đối với lĩnh vực CNTT, bao gồm phân tích dữ liệu, nhận dạng sinh trắc học, bảo mật và trí tuệ nhân tạo.
* Một số ứng dụng cụ thể:
  + Phần mềm nhận diện khuôn mặt để quản lý nhân lực.
  + Phầm mềm nhận diện biển số xe để quản lý xe.
* Và còn rất nhiều ứng dụng nữa mà chúng ta có thể áp dụng: kĩ thuật làm phim 3D, nhận dạng trong y học để phát hiện bệnh, ...

# CHƯƠNG 2: MỤC TIÊU ĐỀ TÀI

## Giới thiệu 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 3: TÌM HIỂU THUẬT TOÁN KNN

## Giới thiệu 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 và đầu ra**:** 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 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

* Độ phức tạp tính toán của quá trình training là bằng 0.
* Việc dự đoán kết quả của dữ liệu mới rất đơn giản.
* Không cần giả sử gì về phân phối của các class.

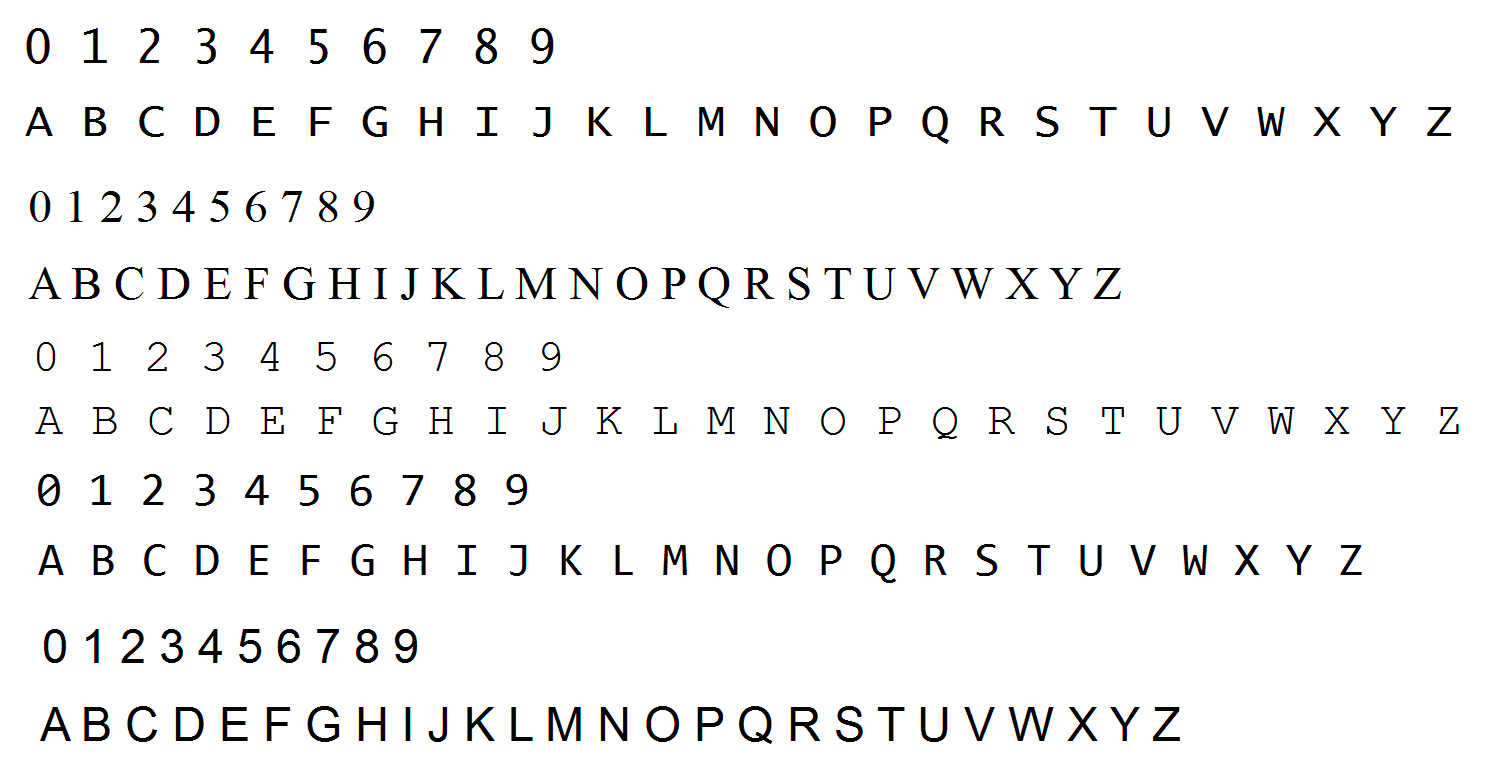
### Nhược điểm của KN

* 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.

# CHƯƠNG 4: THỰC HIỆN ĐỀ TÀI

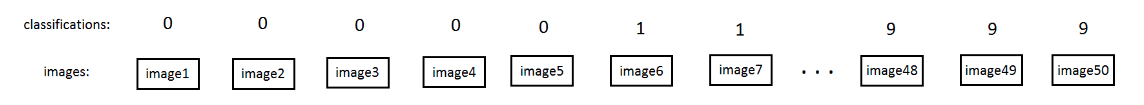
## Module trainning data

* 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

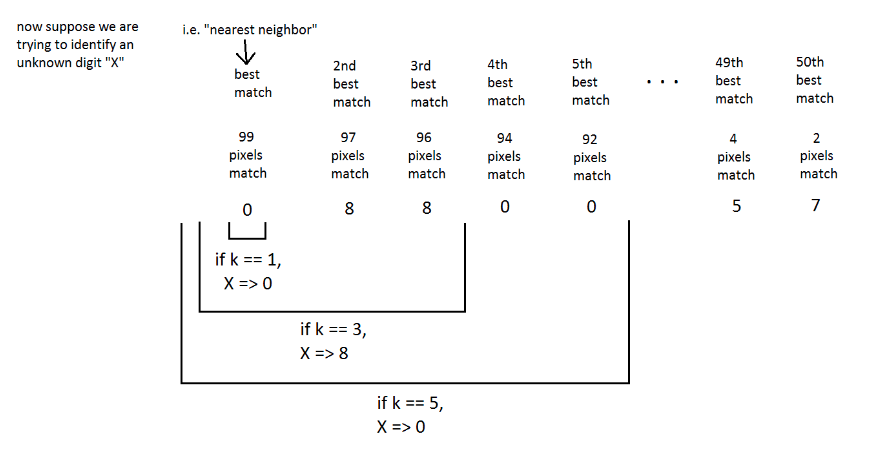


***Hình 4.1: Các tập dữ liệu đặc trưng dùng để trainning***

* 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



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



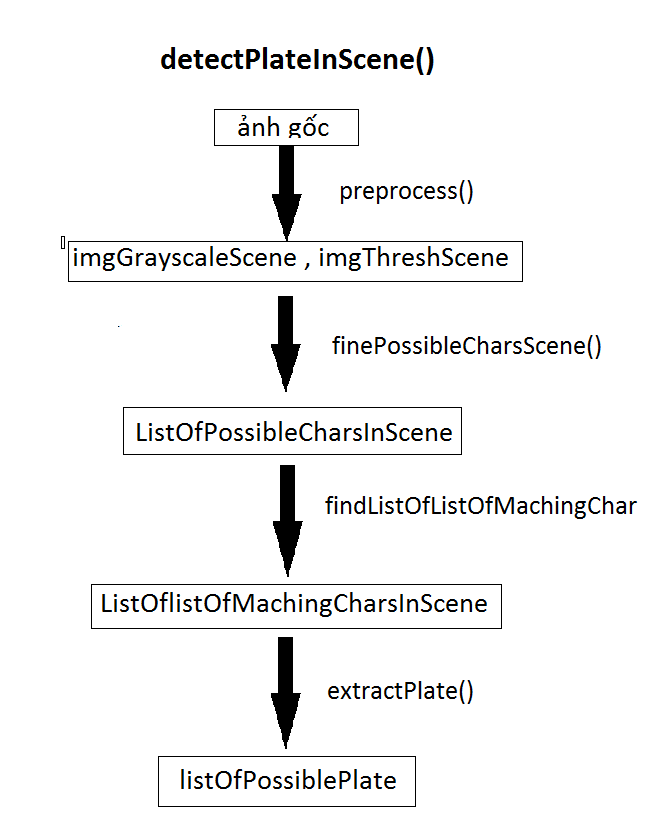
* Giả dụ chúng ta đang cố gắng xác định 1 chữ số không xác định X thì:
  + Khi số cụm k = 1 thì giá trị hàng xóm gần nhất của X = 0
  + Khi số cum k = 3 thì giá trị hàng xóm gần nhất của X = 8
  + Khi số cụm k = 5 Thì giá trị hàng xóm gần nhất của X = 0
  + Tương tự với các trường hợp cụm còn lại (trong ví dụ này k\_min = 1 và k\_max = 50)
* Với dữ liệu đầu vào như *hình 4.1* 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)

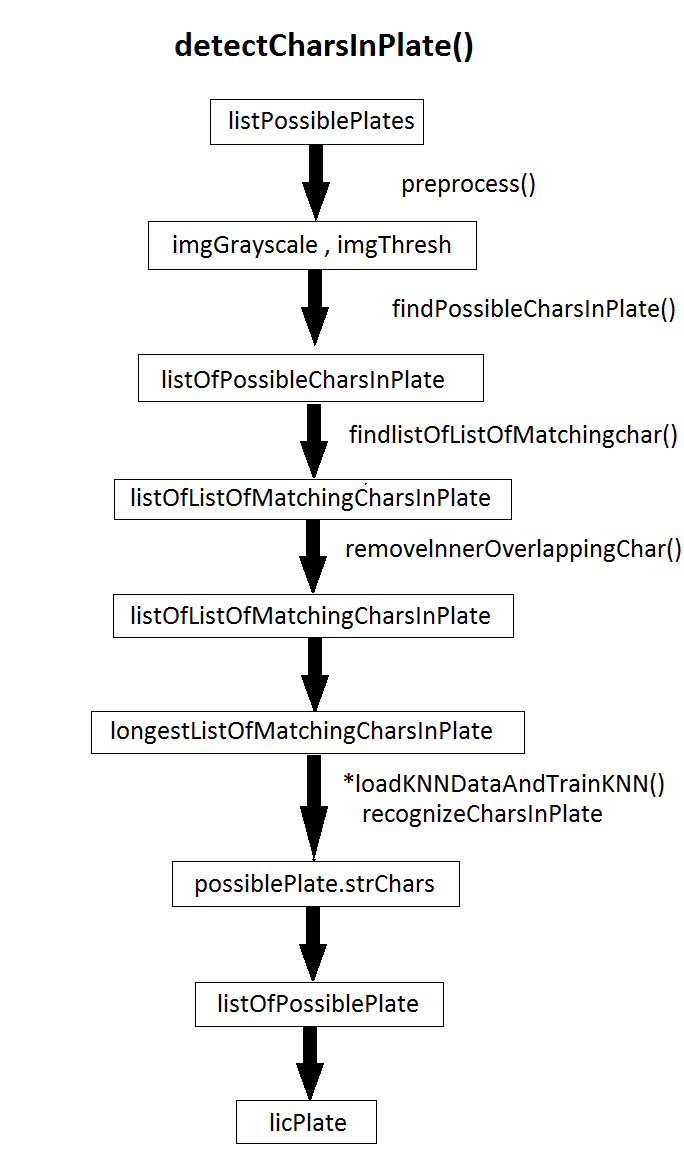
## Quy trình thực hiện

### Các bước thực hiện

#### Tách lấy khung chứa mã biển số xe

****

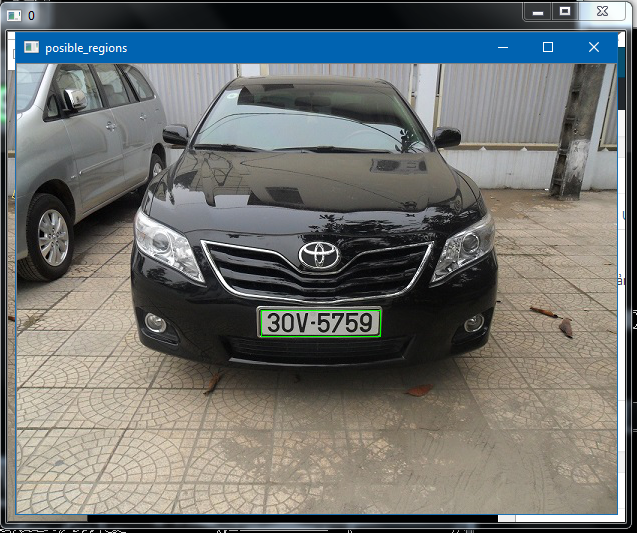
#### Tách lấy từng ký tự trong khung

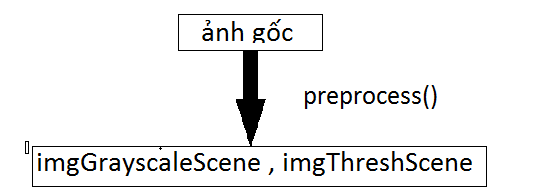
****

### Quy trình

#### Tiền xử lý

* 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 và imgThreshScene***



****

****

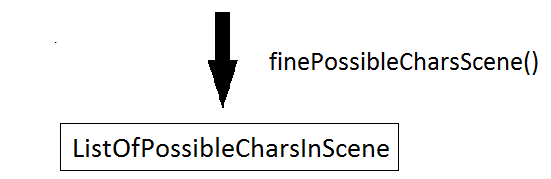
***Hình 4.2: imgGrayscaleScene***

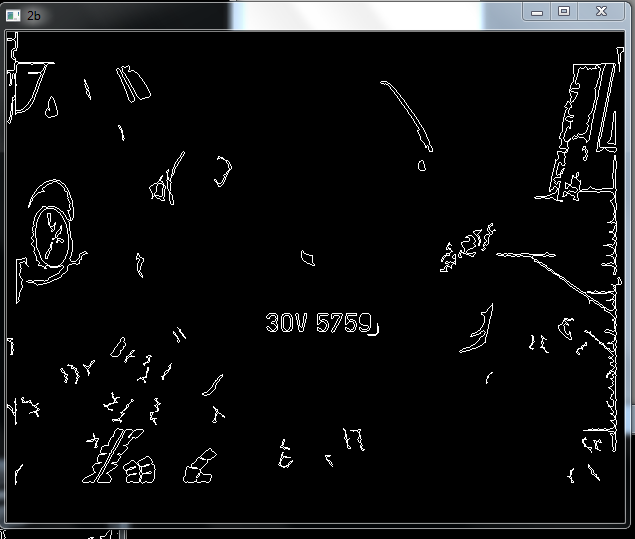
****

***Hình 4.3: imgThreshScene***

#### 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.

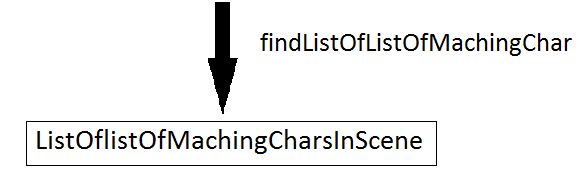


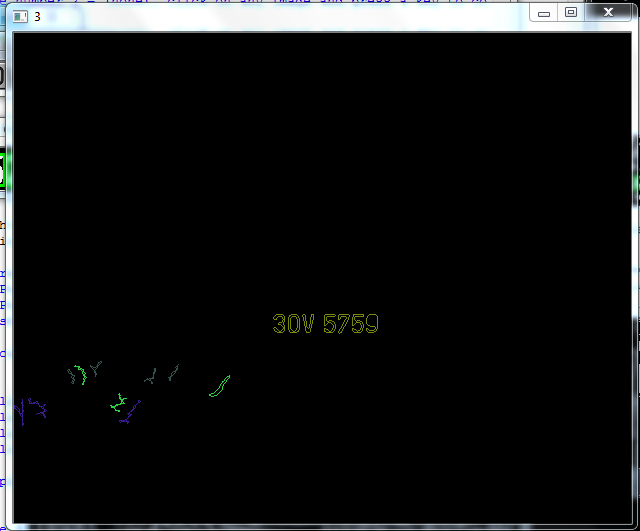
****

***Hình 4.4: 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.

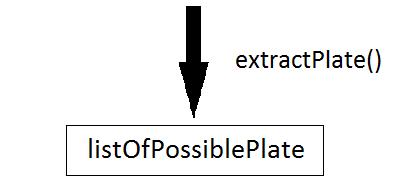


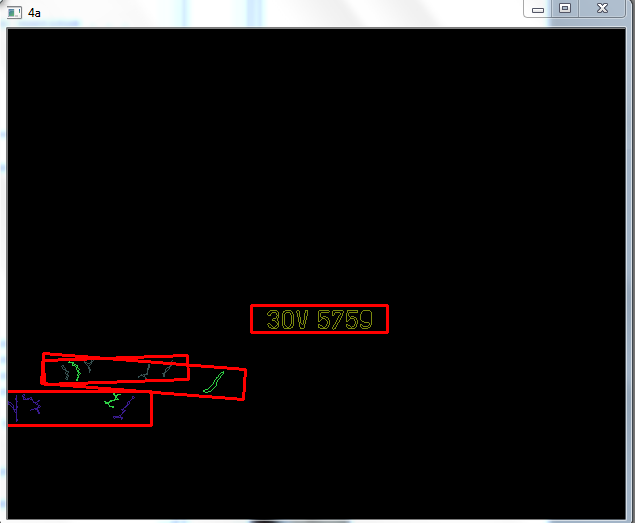
****

***Hình 4.5: 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:

****

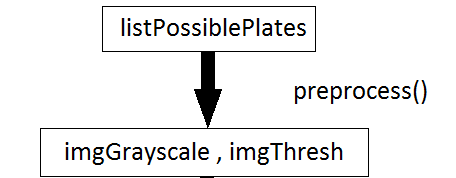
****

** **

***Hình 4.6: Kết quả trích xuất chuỗi ký tự***

#### Tiền xử lý

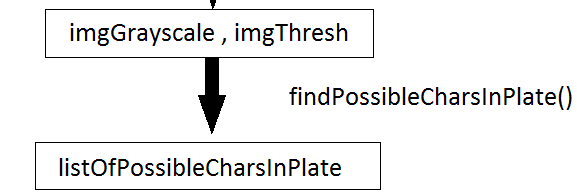
* Như vậy sau bước **2.2.4** chúng ta đã có được plate(khung) có chứa mã biển số phù hợp. chúng ta sẽ tiếp tục thực hiện những công đoạn tiếp theo để có thể tách từng ký tự trong khung đó . đầu tiên là tiền xử lý.
* Sau bước tiền xử lý này cũng giống như ở bước **2.2.1** ta cũng sẽ nhận được ảnh xám và ảnh phân ngưỡng do xử dụng chung hàm preprocess()

****

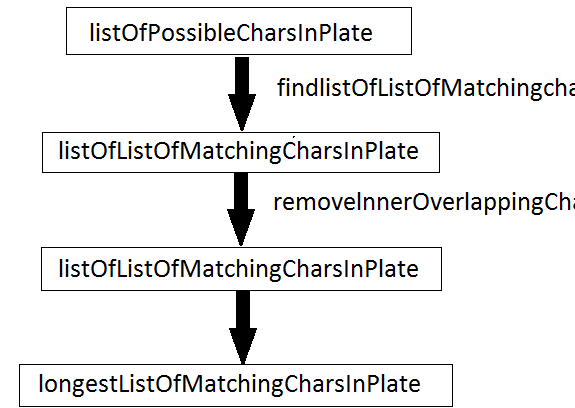
|  |  |
| --- | --- |
|  |  |
| ***Hình 4.7: Ảnh xám*** | ***Hình 4.8: Ảnh ngưỡng*** |

#### 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

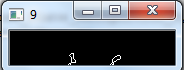
****

** **

****

** **

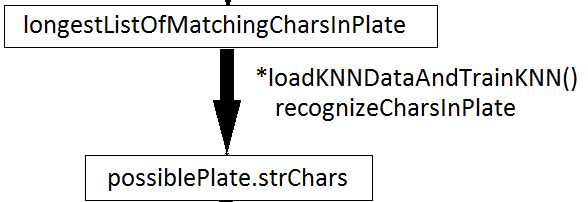
** **

** **

***Hình 4.9: Các quy trình thực hiện tìm chuỗi ký tự từ khung***

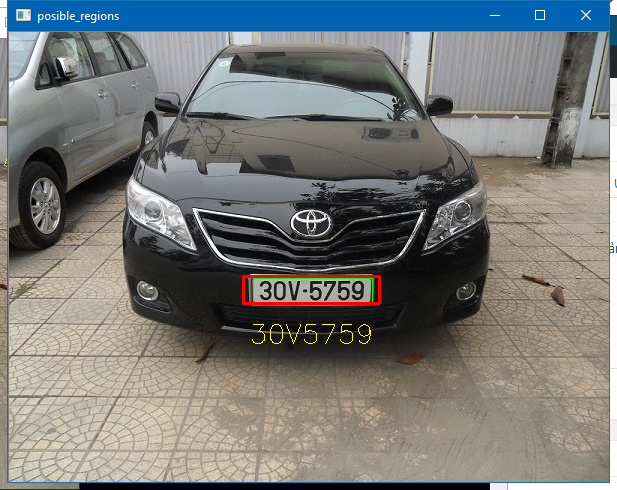
#### Nhận diện ký tự trong khung

* 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.

****

****

* 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

****

# 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
* **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

[1] [Khái niệm KNN là gì ? (Wiki)](https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm)

[2] [Nghiên cứu thuật toán KNN](https://www.slideshare.net/trongthuy3/luan-van-nghien-cuu-thuat-toan-knearest-neighbor-haym-9d)

[3] [Build file trainning Classifications.txt và Flattened\_images.txt](https://www.youtube.com/watch?v=c96w1JS28AY)

[4] [Document how to using KNN for ANPR](https://www.ijitee.org/wp-content/uploads/papers/v8i10/J87820881019.pdf)

[5] [Ưu, nhược điểm của KNN](https://machinelearningcoban.com/2017/01/08/knn/)

[6] [Kaggle.com](https://www.kaggle.com/)