# Introduction

## Lý do chọn đề tài

* Ngày nay, với sự phát triển của khoa học công nghệ kỹ thuật, con người càng tìm ra được nhiều khía cạnh khác nhau của khoa học kỹ thuật, một trong những khía cạnh đó là Học Máy (Machine Learning). Đây là một ngành nghiên cứu khoa học nghiên cứu các thuật toán cho phép máy tính có thể học được các khái niệm của con người. Học máy có nhiều ứng dụng trong khắp các ngành khoa học như là:
* Xử lý ngôn ngữ tự nhiên: xử lý văn bản, giao tiếp,…
* Nhận dạng: nhận dạng tiếng nói, chữ viết tay,…
* Tìm kiếm
* Sinh học: phân loại chuỗi gene,…
* Vật lý: phân tích ảnh thiên văn, tác động giữa các hạt,…
* Nhận dạng là bài toán xuất hiện cách đây khá lâu và vẫn luôn thu hút được nhiều sự quan tâm, nghiên cứu. Đặc biệt là trong vài thập niên gần đây, do sự thúc đẩy của quá trình tin học hoá trong mọi lĩnh vực, bài toán nhận dạng không còn dừng lại ở mức độ nghiên cứu nữa mà nó trở thành một lĩnh vực đáp dụng vào thực tế
* Xuất phát từ yêu cầu thực tế, đang rất cần có nhưng nghiên cứu về vấn đề này. Chính vì vậy tôi đã chọn đề tài nhận dạng ký tự trong ảnh làm đề tài với mong muốn phần nào áp dụng vào bài toán thực tế.

## Lợi ích của đề tài

* Hiểu được Machine Learning là gì .
* Những ứng dụng của Machine Learning vào thực tế.
* Cụ thể hơn trong đề tài này giúp ta hiểu hơn về thuật toán k-Nearest Neighbor
* Không những thế còn giúp ta làm có khả năng làm việc nhóm, đọc tài liệu tốt hơn…

# Goal

Bài toán đã đặt ra phải giải quyết được những yêu cầu sau:

* Một sản phẩm phần mềm đề phục vụ con người
* Nhận dạng được các kí tự lạ trong ảnh
* Kí tự mờ mà mắt thường không nhìn thấy được
* Nhận dạng các kí tự khó đọc

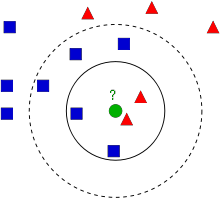
**3. Thuật toán k-Nearest Neighbor**

**3.1 k-Nearest Neighbor là gì ?**

* Là một trong những thuật toán supervised-learning đơn giản nhất trong Machine Learning.
* Ý tưởng của KNN là tìm ra output của dữ kiệu dựa trên thông tin của những dữ liệu training gần nó nhất

**3.2 Thuật toán k-NN làm việc như thế nào ?**

* Bước 1: Xác định tham số k = số láng giềng gần nhất.
* Bước 2: Tính toán khoảng cách giữa mẫu thử và những mẫu training.
* Bước 3: Sắp xếp khoảng cách và xác định k khoảng cách nhỏ nhât.
* Bước 4: Thu thập giá trị thuộc tính của k láng giềng gần nhất
* Bước 5: Sử dụng giá trị trung bình của k láng giềng gần nhất để phán đoán giá trị của Object đang được phán đoán



**Hình minh họa**

* Giả sử có một bài toán được đặt ra: mình đang thích một bạn gái xinh xắn, tuy nhiên mình là SKY vậy nên mình cần biết cô gái này có phải là fan GD hay không. Làm thế nào để xác định cô gái có phải fan GD hay ko, mình quyết định tìm hiểu xem bạn cô gái này có phải fan GD hay không, nếu như cô ấy chơi với nhiều người là fan GD thì gần như có thể kết luận cổ là fan GD. Sau một thời gian tìm hiểu thì mình đã vẽ tất cả thông tin trên hình vẽ ở trên.
* Có thể dễ dàng thấy trên hình có 2 loại : hình vuông màu xanh và tam giác màu đỏ được phân bố trên hình, trong đó thì hình màu xanh biểu diễn cho những người là fan GD, màu đỏ biểu diễn những người không phải là fan GD. Tiếp đến chấm tròn màu xanh là cô gái mình đang muốn biết là fan GD hay không, khoảng cách giữa chấm tròn và các điểm còn lại biểu thị mức độ thân thiết của cô bạn gái với những người bạn.
* Vậy với những dữ liệu trên làm thế nào để ta xác định cô gái thuộc nhóm nào? Phương pháp đơn giản nhất là kiểm tra xem cô gái chơi thân với người nào nhất tức là tìm xem điểm gần chấm xanh nhất thuộc class nào(hình vuông màu xanh hay tam giác màu đỏ). Từ hình trên ta có thể dễ dàng thấy điểm gần chấm xanh nhất là hình tam giác màu đỏ, do đó nó sẽ được phân loại thành tam giác đỏ. Phương pháp này gọi là simply Nearest Neighbour vì điểm cần phân loại chỉ phụ thuộc nào điểm gần nó nhất
* Có một vấn đề trong phương pháp trên, có thể là điểm màu đỏ gần điểm ta đang xét nhất tuy nhiên xung quanh đó có rất nhiều điểm xanh. Vì vậy, việc xét điểm gần nhất là chưa đủ, thay vào đó ta sẽ xét k điểm gần nhất. Giả sử, ta lấy k=3, dự theo hình ảnh trên, ta có thể dễ dàng nhận ra có 2 điểm đỏ và 1 điểm xanh gần điểm ta đang xét nhất. Do đó, chấm xanh vẫn được phân loại thành hình tam giác đỏ. Nếu ta lấy k=7, thì hiện tại xung quanh có 5 điểm xanh 2 điểm đỏ, lúc này chấm xanh lại được xếp vào hình vuống xanh. Vì vậy, việc chọn giá trị k thực sự quan trọng. Có 1 điều không biết các bạn có nhận ra không nhưng nếu ta lấy k=4 thì sẽ có 2 điểm xanh, 2 điểm đỏ, đây là 1 tie. Vì vậy, người ta thường chọn k là số lẻ. Đó là ý tưởng của KNN
* Với trường hợp k=4 hay là các trường hợp tie, lúc này LNN sẽ so sánh khoảng cách của các điểm neighbour với điểm ta đang xét

**3.3 Ưu và nhược điểm**

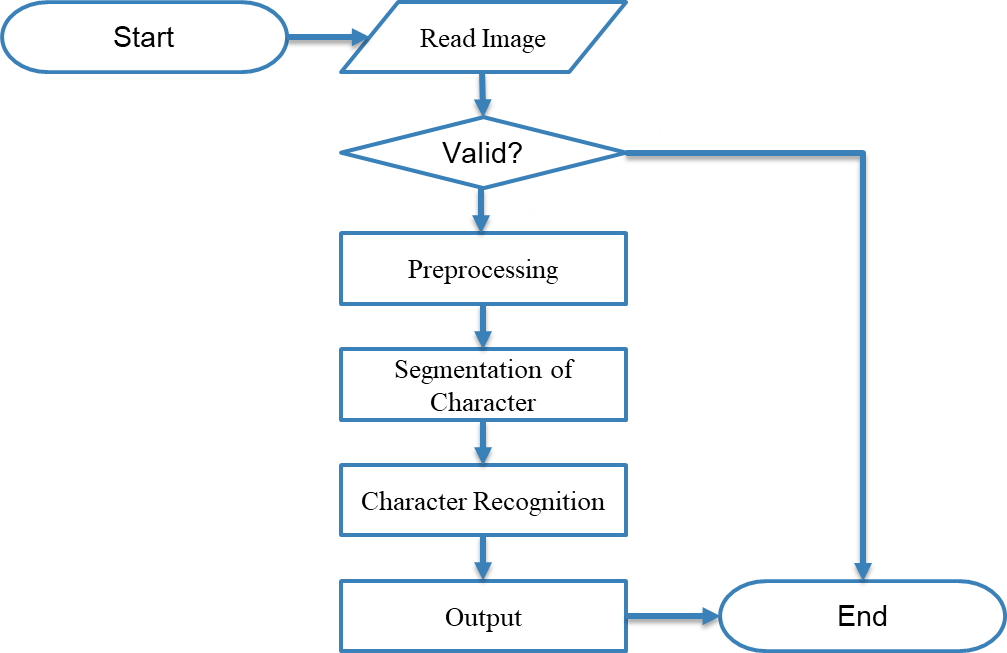
a) Ưu điểm

* Dễ sử dụng và cài đặt
* Độ phức tạp tính toán nhỏ
* Việc dự đoán kết quả của dữ liệu mới rất đơn giản.

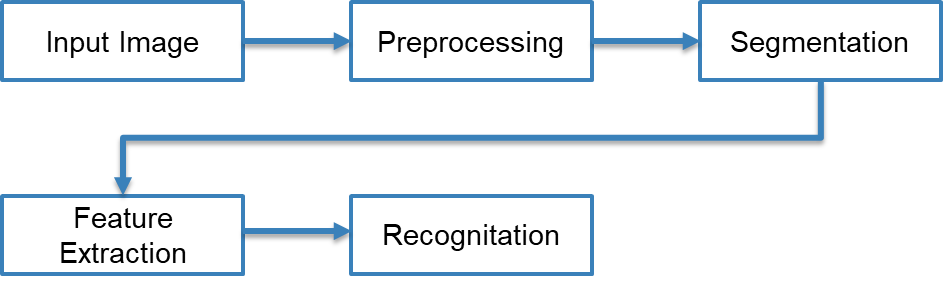
### b) Nhược điểm

* Với K nhỏ, kinh gặp nhiễu dễ đưa ra kết quả ko chính xác
* Cần thời gian lưu trainning set, khi dữ liệu trainning và test tăng lên thì sẽ tốn rất nhiều thời gian

**4. Flowchart**



**5. Block Diagram**



**INPUT** :

****

* **OUTUT : Based**

Thuật toán như sau: với mỗi ảnh 2D chuyển thành ảnh 1D. Sau đó ghép các ảnh 1D thành ảnh 2D dùng để huấn luyện.

(*Ảnh 1D là ảnh có height = 1, có thể xem nó là mảng 1 chiều. Còn ảnh 2D có height > 1 và cũng xem như là mảng 2 chiều.  
Thí dụ: ảnh 2D có kích thước 10 x 10 pixels chuyển thành ảnh 1D có kích thước là 100 x 1 pixels*)

Ảnh ký tự 2D và chuyển sang 1D (mình tăng height cho các bạn dễ nhìn)  
  


Sau khi ghép các ký tự từ 0 – 9 được hình như sau:  
  


Tiếp theo là tạo 1 danh sách các nhãn của ảnh train, danh sách nhãn cũng là ảnh 1D theo thứ tự. Do đó danh sách nhãn là **0123456789**