

# **Đồ án tự động nhận diện điểm và mã số sinh viên được viết tay**

Phan Gia Huy-18520068-CS114.K21.KHTN

Link github:

[https://github.com/huyphangia/CS114.K21.KHTN/blob/master/%C4%90%E1%BB%93\\_%C3%A1n.ipynb](https://github.com/huyphangia/CS114.K21.KHTN/blob/master/%C4%90%E1%BB%93_%C3%A1n.ipynb)

## Ý tưởng:

- Lấy ý tưởng từ vấn đề của một giáo viên dạy chính trị, trong một lần đứng lớp đã kể rằng cô và đồng nghiệp rất hay bị nhầm khi nhập điểm thi, điểm kiểm tra từ bài mà mình chấm vào file trên excel vì số lượng bài rất nhiều, phải mất nhiều công sức để rà soát xem mình nhập đã đúng chưa và đôi khi dẫn đến tình trạng sinh viên khiếu nại, phúc khảo điểm.

# Tóm tắt đề án

- **Input:** Ảnh chụp có gồm hai khung có chứa MSSV và điểm của sinh viên
- **Output:** Lưu điểm của sinh viên và mã số sinh viên trong input vào file csv

# Các bước thực hiện

- Bước 1: Xây dựng một model nhận diện chữ số viết tay bằng dữ liệu tự thu thập được
- Bước 2: Nhận diện khung chứa điểm và khung chứa MSSV, sau đó trích xuất hai khung chứa số ra hai ảnh khác nhau
- Bước 3: Nhận diện các số trong dãy, tách chúng thành các số đơn lẻ và trích xuất thành các ảnh tương ứng
- Bước 4: Dùng Model đã train ở bước 1 để nhận diện các ảnh đã cắt ở bước 3 và lưu vào file csv

# Bước 1: Thu thập dữ liệu và xây dựng model

Dữ liệu:

- Ảnh chụp bằng điện thoại
- Dữ liệu gồm 200 tấm ảnh các số từ 0 -9
- Dữ liệu được 2 bạn tự thu thập: Nguyễn Anh Khoa -18520923 và Phan Gia Huy-18520068
- Bộ dữ liệu gồm 2000 tấm ảnh các số chia đều từ 0-9.

# Trích xuất đặc trưng

- Vector hóa
- Không dùng HOG do ảnh đầu vào nhỏ
- Dùng standard scale để scale dữ liệu

# Model

Chọn hai model chính là:

- KNN: Đối với các bài toán Multiclass Classification, thuật toán KNN tỏ ra rất hữu dụng, chạy nhanh, cho kết quả khá chính xác .
- SVM: do tập dữ liệu nhỏ và thuật toán SVM rất phù hợp với bộ dữ liệu không quá lớn và đối với những bài toán có số chiều (dimention) lớn, thì SVM là một trong những lựa chọn rất tốt, trong bài toán này có 784 dimention trong vector, khá lớn

# Đánh giá và Fine-tuning:

- Đánh giá:
  - Dùng `cross_val_score`
  - Số fold là 3 cho các loại model
  - Dự đoán accuracy của model trên từng fold dựa trên model được train ở các fold còn lại
  - Có dùng dữ liệu đã được scale để đánh giá thử, tuy nhiên kết quả đạt được gần giống như dữ liệu chưa được scale, nên trong bước tiếp theo dùng dữ liệu chưa scale



# Đánh giá và Fine-tuning:

- Fine-tuning:
  - Dùng gridsearch
  - Tìm các hyper-parameter tối ưu phù hợp với từng thuật toán
  - Ở đây, đối với KNN là tìm weights và n\_neighbors, sau khi chạy đã tìm được weights là uniform và n\_neighbors=1 là những hyper-parameter tối ưu nhất với thuật toán
  - Đối với SVM là tìm C, tìm được C=0.001

# Test trên tập testing data và kết luận

- Test trên data sau khi đã tinh chỉnh tham số tuy nhiên kết quả không nâng cao nhiều
- Kết luận:
  - Do dữ liệu tự thu thập quá nhỏ, dẫn đến hiện tượng overfit trên training data
  - Ảnh tự chụp chất lượng không đảm bảo dẫn đến accuracy không cao

# Test trên tập testing data và kết luận

- Hướng khắc phục:
  - Xây dựng bộ data to hơn
  - Sử dụng bộ data mnist
  - Dùng các kỹ thuật feature engineering mới
  - Sử dụng các thuật toán phức tạp hơn trong deeplearning

## **Bước 2: Nhận diện khung chứa điểm và khung chứa MSSV, sau đó trích xuất hai khung chứa số ra hai ảnh khác nhau**

2.1 Chụp ảnh và load ảnh lên colab (do người dùng thực hiện)

2.2 Tiến hành nhận diện khung và tách ảnh

- Xử lý ảnh
- Tìm khung ảnh chứa MSSV và điểm
- Trích xuất hai khung ra thành hai ảnh riêng biệt

**Bước 3+4: Từ hai ảnh ở bước 2, nhận diện, tách dãy số thành các số riêng lẻ và predict các số ấy, lưu vào file csv**

- Xử dụng hai ảnh đã được cắt từ bước 2
- Các bước xử lý ảnh tương tự như bước 2
- Ghi kết quả dự đoán được vào file Csv

# Kết luận:

**Tuy đề án lần này, model em làm chưa được như mong muốn dẫn đến hệ thống có kết quả chưa tốt, nhưng đã giúp em học hỏi thêm rất nhiều thứ từ việc lọc data, chọn model, đánh giá, tinh chỉnh, cách trích xuất hình ảnh,... Quan trọng nhất là kiến thức về Machine Learning được nâng cao hơn, bên cạnh đó là việc tự mình tìm hiểu, tạo ra ,so sánh, đánh giá, tối ưu những các phương án để xử lý vấn đề, tiếp cận với các bước làm dự án máy học và xây dựng hệ thống dựa trên máy học trong thực tế.**