ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO BÀI TẬP 2
NHẬN DẠNG VÂN TAY BẰNG DEEP LEARNING

BỘ MÔN: SINH TRẮC HỌC

Giảng viên hướng dẫn:

Lê Hoàng Thái

Lê Thanh Phong

PHẠM GIA KHIÊM

20120307 – PHẠM GIA KHIÊM

Thành phố Hồ Chí Minh - 2023

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO BÀI TẬP 2
NHẬN DẠNG VÂN TAY BẰNG DEEP LEARNING

BỘ MÔN: SINH TRẮC HỌC

Giảng viên hướng dẫn:

LÊ HOÀNG THÁI

LÊ THANH PHONG

Thành phố Hồ Chí Minh - 2023

LỜI CẢM ƠN

Trong quá suốt quá trình học tập môn học sinh trắc học và thực hiện bài tập này, em đã nhận được nhiều sự hướng dẫn góp ý, giúp đỡ tận tình từ thầy cô, bạn bè.

Em rất cảm ơn thầy Lê Hoàng Thái và thầy Lê Thanh Phong, khoa Công nghệ thông tin trường Đại học Khoa học Tự nhiên, Đại học Quốc gia TPHCM đã hướng dẫn và truyền đạt rất nhiều kiến thức bổ ích với chúng em trong suốt thời gian qua.

Cảm ơn chân thành những lời góp ý của bạn bè và sự góp sức của bản thân các thành viên trong nhóm hoàn thành đồ án.

Trân trọng cảm ơn!

MỤC LỤC

DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt	Từ nguyên mẫu	Ý nghĩa
1	FRB	Fingerprint Recognition Biometrics	Nhận dạng dấu vân tay
2	FAR	False Accept Rate	Tỉ lệ chấp nhận sai
3	FRR	False Reject Rate	Tỉ lệ từ chối sai
4	SOCOFing	Sokoto Coventry Fingerprint Dataset	Tên của bộ dữ liệu
5	CNN	Convolution Neural Network	Mạng học sâu tích chập
6	FCN	Fully Connected Network	Mạng học sâu liên kết đủ các node

DANH MỤC BẢNG BIỂU

DANH MỤC HÌNH ẢNH

TỔNG QUAN VỀ NHÓM

STT	Họ và tên	Mã số sinh viên	Email
1	Phạm Gia Khiêm	20120307	20120307@student.hcmus.edu.vn
2			
3			

TỔNG QUAN BÀI TOÁN

1. Giới thiệu bài toán

1.1. Sinh trắc học là gì ?

Sinh trắc học (Biometric) là phương pháp sử dụng các đặc điểm sinh trắc học độc nhất của con người để xác định, nhận dạng và xác thực danh tính. Cụ thể, Biometric là thuật ngữ từ Hy Lạp được tạo thành từ "bio" – nghĩa là "cuộc sống" và "metric" – nghĩa là "thước đo" [1]. Vì vậy biometric mang hàm ý là thước đo các đặc điểm sinh học của mỗi người. Từ dấu vân tay, các đường nét trên khuôn mặt cho đến dáng đi, không có hai người nào trên thế giới có thể có sinh trắc học giống nhau, thậm chí là cả những cặp sinh đôi.

Tính độc đáo và riêng biệt này đã làm cho Biometric trở thành phương tiện đáng tin cậy để nhận dạng một người, vì so với mật khẩu hay mã pin, các đặc điểm sinh học thường rất khó để giả mạo và sao chép.

Bên cạnh đó, bảo mật sinh trắc học cũng đem đến sự nhanh chóng và thuận tiện cao. Với công nghệ này, người dùng sẽ không còn phải ghi nhớ các chuỗi pin hay mật khẩu phức tạp. Và không giống như mật khẩu hay các thông báo mã xác thực, họ cũng không thể quên hay làm thất lạc sinh trắc học của mình.

Quan trọng nhất là, khác với các hệ thống được bảo mật bằng mật khẩu, sẽ thực sự khó để hack một hệ thống được bảo vệ bởi sinh trắc học.

Hình 1. Ví dụ về sinh trắc học

Các phương pháp bảo mật Biometric hiện nay bao gồm: mã hóa, xác thực và xác minh người dùng. Trong đó:

- Mã hóa dữ liệu là quá trình biến đổi thông tin gốc thành dạng khó đọc để ngăn chặn việc truy cập trái phép vào thông tin.
- Còn xác thực là quá trình xác định người dùng đúng là ai.

- Xác minh là quá trình xác định xem người dùng có phải là người được ủy quyền hay không.

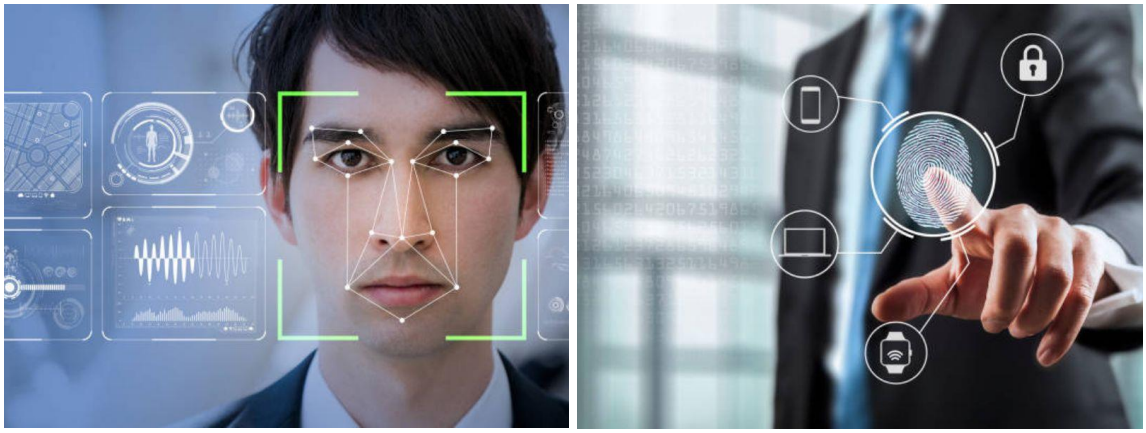
1.2. Bảo mật trong sinh trắc học

Biometrics được sử dụng khá rộng rãi và được chia thành 3 nhóm chính [1]:

- Sinh trắc học sinh học.
- Sinh trắc học hình thái.
- Sinh trắc học hành vi.

Trong bài tập này, chúng ta sẽ nhấn mạnh vào nhóm thứ 2, sinh trắc học hình thái. Bởi vì Sinh trắc học hình thái (hay Morphological Biometrics) là phương pháp sử dụng cấu trúc của cơ thể người dùng để xác định danh tính.

Những đặc trưng vật lý của cơ thể như dấu vân tay, hình dạng khuôn mặt, mống mắt, mô hình mạch máu võng mạc, vân tĩnh mạch và các đặc điểm khác có thể được sử dụng để tạo ra bản đồ dấu vân tay, ảnh khuôn mặt hoặc các thông tin khác và được lưu trữ trong cơ sở dữ liệu nhận dạng.



Hình 2. Ví dụ về sinh trắc học hình thái

Trong Biometric Authentication thì nhận dạng mặt người (Face Recognition) và nhận dạng dấu vân tay (Fingerprint Recognition) là hai phương pháp thường được sử dụng nhất.

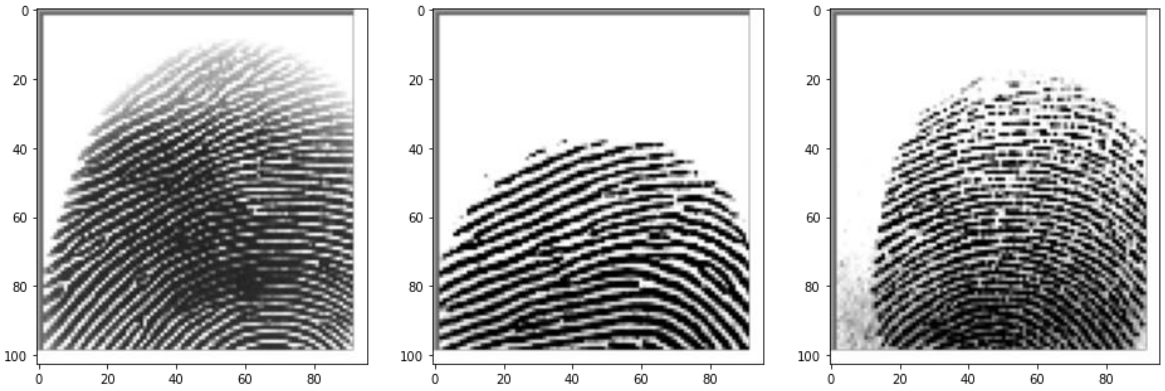
Trong bài tập này, ta sẽ thực hiện tìm hiểu và xây dựng mô hình Fingerprint Recognition Biometrics (Fingerprint Scanning) bằng Deep Learning.

2. Nhận dạng dấu vân tay

2.1. Giới thiệu về nhận dạng dấu vân tay

Nhận dạng dấu vân tay (Fingerprint Recognition Biometrics) hoặc mở khoá bằng vân tay (Fingerprint Scanning) là phương pháp nhận dạng sinh trắc học dựa trên các đặc

trưng vân tay của con người. Phương pháp này sẽ sử dụng các đường vân tay trên ngón tay để nhận diện hoặc xác thực, xác minh danh tính của người đó [1..3].



Hình 3. Một số hình dạng dấu vân tay

Dấu vân tay là các đặc điểm vật lý độc đáo và nhất quán có thể được sử dụng làm các nhận dạng sinh trắc học. Fingerprint Recognition Biometrics có thể được sử dụng cho mục đích xác thực (so khớp mẫu sinh trắc học của một người) và nhận dạng (xác định danh tính của một người).

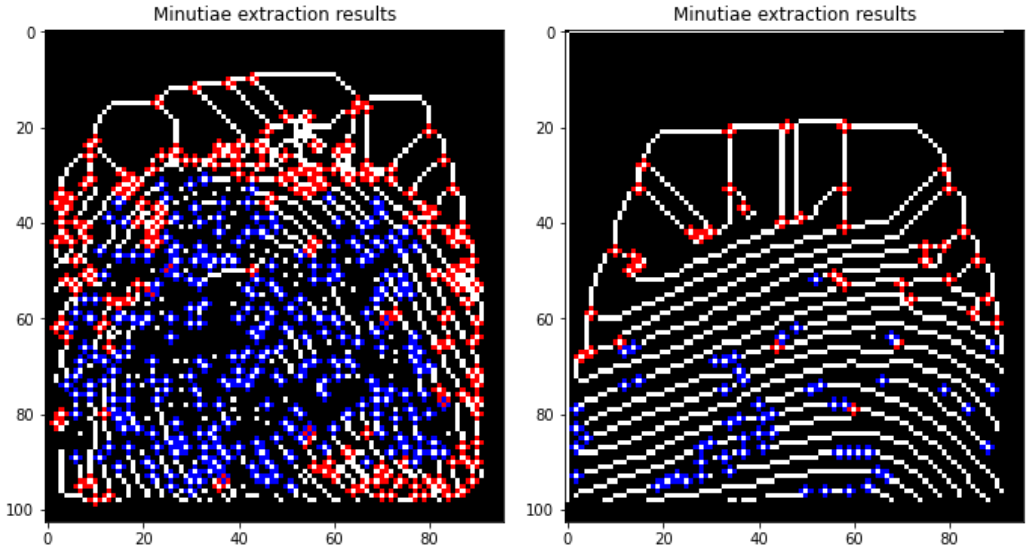
2.2. Ứng dụng nhận dạng dấu vân tay

Fingerprint Recognition Biometrics (FRB) [1...3] là một trong những loại sinh trắc học phổ biến và được sử dụng rộng rãi nhất, vì nó dễ dàng, tiết kiệm chi phí, đáng tin cậy và thuận tiện.

FRB đã được sử dụng trong hơn một thế kỷ và đã trở nên tự động hóa với sự tiến bộ của công nghệ máy tính. FRB sử dụng các mẫu cụ thể, như các gờ, thung lũng, cung, vòng và xoắn, để so sánh và khớp dấu vân tay.

Hình 4. Các đặc điểm ta có thể học trên dấu vân tay

FRB cũng có thể sử dụng các điểm nhỏ, là các điểm nơi các gờ kết thúc hoặc phân nhánh. FRB có thể cung cấp một mức độ bảo mật cao, vì dấu vân tay rất khó để giả mạo, mất hoặc xâm phạm.



Hình 5. Dùng machine learning để rút đặc trưng dấu vân tay [4]

Tuy nhiên, FRB cũng có một số hạn chế, như khả năng xảy ra lỗi, giả mạo hoặc tấn công. FRB có thể được sử dụng cho nhiều ứng dụng, như thực thi pháp luật, nhập cư, kiểm soát truy cập, mở khóa thiết bị và xác minh thanh toán.

2.3. Các thách thức hiện tại

Hiện tại, việc nhận dạng bằng dấu vân tay (Fingerprint Recognition Biometric) có khá nhiều thách thức [2]:

- Giảm tỉ lệ chấp nhận sai (False Accept Rate - FAR): Với bài toán nhận dạng, thì tỉ lệ này sẽ thường khá cao, do nếu các dấu vân tay không có trong bộ dữ liệu thì mô hình sẽ nhận dạng sai người, cho ra sai kết quả, điều này sẽ không tốt khi ứng dụng vào máy chấm công, điểm danh bằng dấu vân tay.
- Giảm tỉ lệ từ chối sai (False Reject Rate - FRR): Với bài toán xác thực dấu vân tay, tỉ lệ này thường sẽ khá cao, do dấu vân tay khi xác thực có thể bị thay đổi do các trường hợp ngoại cảnh như bị thương ở vân tay, dấu vân tay bị nhoè do tay dính nước,… Phương pháp xác thực dấu vân tay thường dùng trong mở khoá bằng vân tay, nếu FRR cao thì việc xác thực sẽ mất khá nhiều thời gian.

GIỚI THIỆU VỀ BỘ DỮ LIỆU

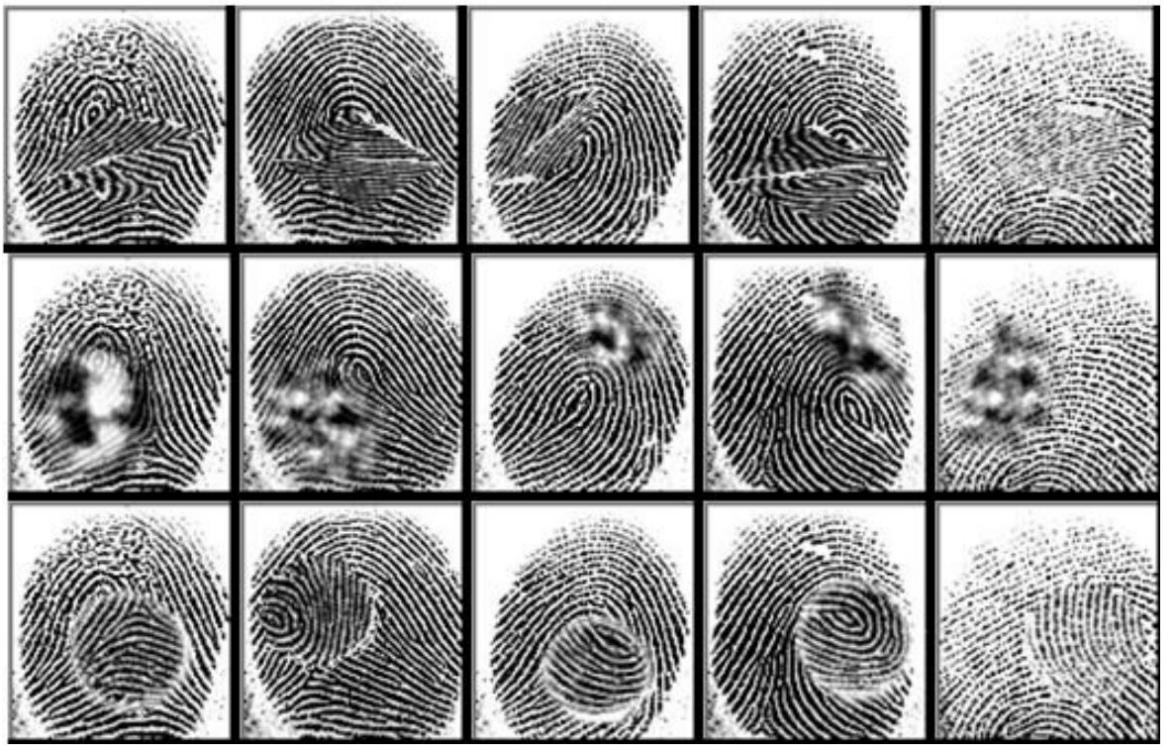
1. Sokoto Conventry Fingerprint Dataset

Sokoto Coventry Fingerprint Dataset (SOCOFing) [5] là một bộ dữ liệu vân tay được thu thập bởi các nhà nghiên cứu từ Đại học Coventry (Anh) và Đại học Sokoto (Nigeria). Bộ dữ liệu này bao gồm 6,000 ảnh vân tay từ 600 người châu Phi khác nhau.

SOCOFing có những thuộc tính độc đáo như nhãn cho giới tính, tay và tên ngón tay cũng như các phiên bản bị thay đổi tổng hợp với ba mức độ thay đổi khác nhau cho việc xóa, xoay trung tâm và cắt z (z-cut).

2. Tổng quan về bộ dữ liệu SOCOFing

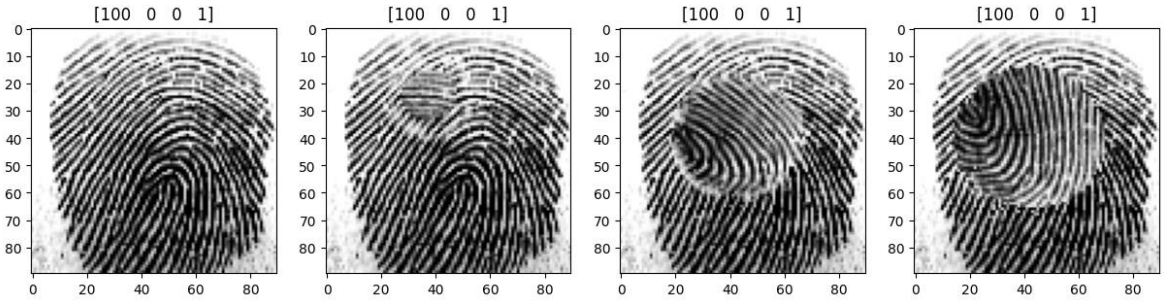
Như đã biết ở trên, SOCOFing gồm 6,000 ảnh vân tay gốc của 600 người châu Phi khác nhau. Ngoài ra, bộ dữ liệu cũng đã được thay đổi tổng hợp dựa trên ba mức độ và ba phương pháp thay đổi là xoá, xoay trung tâm và cắt z. Các thay đổi được thực hiện bằng công cụ STRANGE [5], một khung làm việc mới cho việc tạo ra các thay đổi tổng hợp trên ảnh vân tay.



Hình 6. Hình ảnh các dấu vân tay được thay đổi theo 3 phương pháp là xoá, xoay trung tâm và cắt z

Dữ liệu bao gồm tổng cộng 55,273 ảnh vân tay. Trong đó có

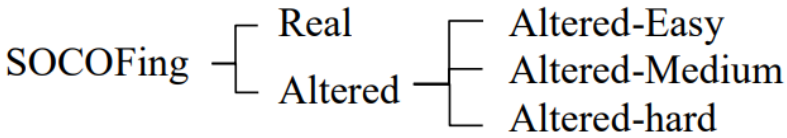
- Ảnh bị thay đổi ở mức độ dễ: 17,934 ảnh.
- Ảnh bị thay đổi ở mức độ trung bình: 17,067 ảnh.
- Ảnh bị thay đổi ở mức độ khó: 14,272 ảnh.



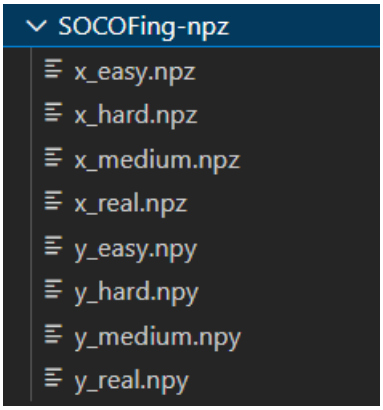
Hình 7. Hình ảnh dấu vân tay gốc và ba mức độ huỷ hoại từ dễ (easy) đến trung bình (medium) và đến khó (hard)

3. Khám phá dữ liệu

Theo bài báo [5], bộ dữ liệu được chia thành 2 tập dữ liệu nhỏ là Altered và Real, và trong tập dữ liệu Altered thì ta sẽ có thêm 3 tập dữ liệu nhỏ được chia ra theo 3 mức độ là easy, medium và hard.



Tuy nhiên, thực tế, dữ liệu mà ta thu thập được sẽ là SOCOFing-npz [], dữ liệu này sẽ gồm 8 tập dữ liệu nhỏ được lưu dưới dạng npz, ta sẽ chỉ cần đọc trực tiếp dữ liệu vào mà không cần xử lý quá nhiều.



Hình 8. Cách thức lưu trữ của dataset

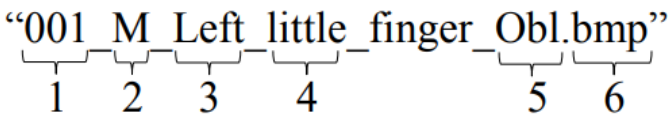
Sau khi đọc dữ liệu vào, ta sẽ xem kích cỡ của dữ liệu đọc vào, lúc này ta sẽ thấy dữ liệu có kích thước như sau:

```
Shapes:                                Feature shape          Label shape
-------------------------------------------------------------------------
Finger Real data:                    (6000, 90, 90, 1)        (6000, 4)
Finger Altered Easy data:            (17931, 90, 90, 1)       (17931, 4)
Finger Altered Medium data:          (17067, 90, 90, 1)       (17067, 4)
Finger Altered Hard data:            (14272, 90, 90, 1)       (14272, 4)
```

Hình 9. Kích thước của dataset

Sau khi kiểm tra thì hầu như Feature shape hoàn toàn khớp với số lượng hình ảnh mà bài báo đề cập, tuy nhiên dữ liệu hình ảnh lại có kích thước là (90, 90, 1).

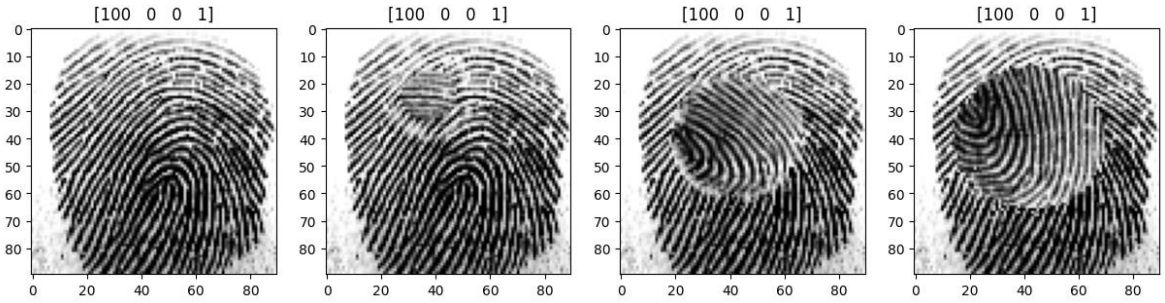
Đối với nhãn (label) của dữ liệu, trong bài báo [5], tác giả có định nghĩa dựa trên tên file hình ảnh.



Trong đó:

- 1. Mã định danh ID của từng người: 001 đến 600.
- 2. Giới tính: M – male, F – female.
- 3. Bàn tay: Left – bàn tay trái, Right – bàn tay phải.
- 4. Tên của vị trí ngón tay:
  - little: Ngón út.
  - ring: Ngón áp út.
  - middle: Ngón giữa.
  - index: Ngón trỏ.
  - thumb: Ngón danh.
- 5. Phương pháp thay đổi ảnh: Obl – obliteration, CR – central rotation, or Zcut.
- 6. Tên file: bmp

Tuy nhiên, trong bộ dữ liệu mà em thu thập thì không có tên dữ liệu, mà nhãn đã được xử lý và chuyển thành dạng danh sách gồm 4 phần tử.



Hình 10. Ví dụ minh hoạ về cách đánh nhãn của dataset

Ý nghĩa của từng phần tử trong một nhãn:

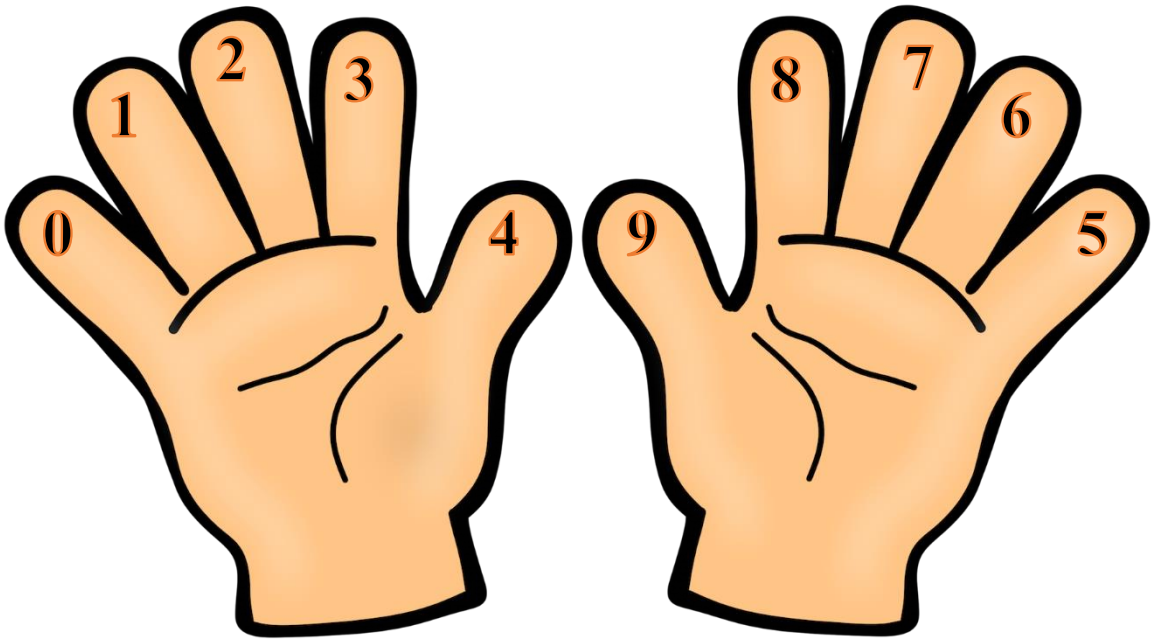
- Phần tử 1: Mã định danh ID của một người (001 đến 600).
- Phần tử 2: Giới tính (0 – Nam, 1 – Nữ).
- Phần tử 3: Bàn tay (0 – Tay trái, 1 – Tay phải).
- Phần tử 4: Tên của vị trí ngón tay
    - 0: Ngón út.
    - 1: Ngón áp út.
    - 2: Ngón giữa.
    - 3: Ngón trỏ.
    - 4: Ngón danh.

4. Xử lý dữ liệu

Với dữ liệu đã được xử lý từ trước, ta không cần xử lý dữ liệu quá nhiều, việc cần làm là ta sẽ xác định những nhãn mà ta cần dùng cho bài toán.

Ta sẽ chỉ dùng 2 nhãn: Mã định danh ID và vị trí của ngón tay trên bàn tay. Tuy nhiên ta chỉ cần 1 dữ liệu vị trí để định danh, mà nhãn ban đầu thì lại có 2 dữ liệu là bàn tay và vị trí ngón tay. Do đó ta sẽ phải tính toán và định nghĩa lại vị trí của ngón tay trên bàn tay.

Ta sẽ định nghĩa theo hình ảnh như sau:



Hình 11. Hình ảnh ví dụ cho cách xử lý dữ liệu

Ví dụ:

- [100, 0, 0, 1] là tay trái, ngón áp út của người nam có ID là 100 thì ta sẽ biến đổi thành [100, 1].
- [100, 0, 1, 1] là tay phải, ngón áp út của người nam có ID là 100 thì ta sẽ biến đổi thành [100, 6].

Sau khi đổi nhãn thành 2 phần tử cần thiết, ta sẽ biến đổi nhãn từ dạng numerical sang dạng categorical để model dự đoán kết quả phân lớp. Vì ta có 2 nhãn là ID và vị trí ngón tay, nên ta sẽ có 2 model.

- Model 1: SubjectID-Model, model này sẽ dự đoán phân lớp cho 600 người trong bộ dữ liệu.
- Model 2: fingerNum-Model, model này sẽ dự đoán phân lớp cho 10 ngón tay.

MÔ HÌNH NHẬN DẠNG DẤU VÂN TAY

1. Sơ lượt mô hình

Ở mô hình cơ bản, ta sẽ cài đặt dựa trên nền tảng là CNN. Như đã phân tích phía trên, ta sẽ có 2 mô hình, tuy nhiên hai mô hình này chỉ khác nhau ở Output Layer, bởi vì tuỳ vào từng nhu cầu thì mô hình sẽ có output khác nhau:

- **SubjectID_Model**: Mô hình phân lớp ID của 600 người trong bộ dữ liệu.
- **FingerNum_Model**: Mô hình phân lớp 10 ngón tay trong bộ dữ liệu.

2. Ý tưởng cài đặt mô hình

Dựa trên mô hình đã có sẵn và mô hình VGG16 [6, 7, 8], ta sẽ chỉ lấy 4 lớp Convolution đầu của mô hình VGG16 và phần còn lại sẽ giống như model đã có sẵn trên kaggle.

Trong mã nguồn, em có cài đặt 2 loại mô hình cho bài, một là mô hình như đã đề cập phía trên, mô hình còn lại áp dụng Vision Transformer [9, 10] vào bài toán, tuy nhiên, kết quả của mô hình này không tốt bằng mô hình trên nên em sẽ chỉ đề cập chủ yếu đến mô hình cơ bản phía trên.

3. Xây dựng mô hình

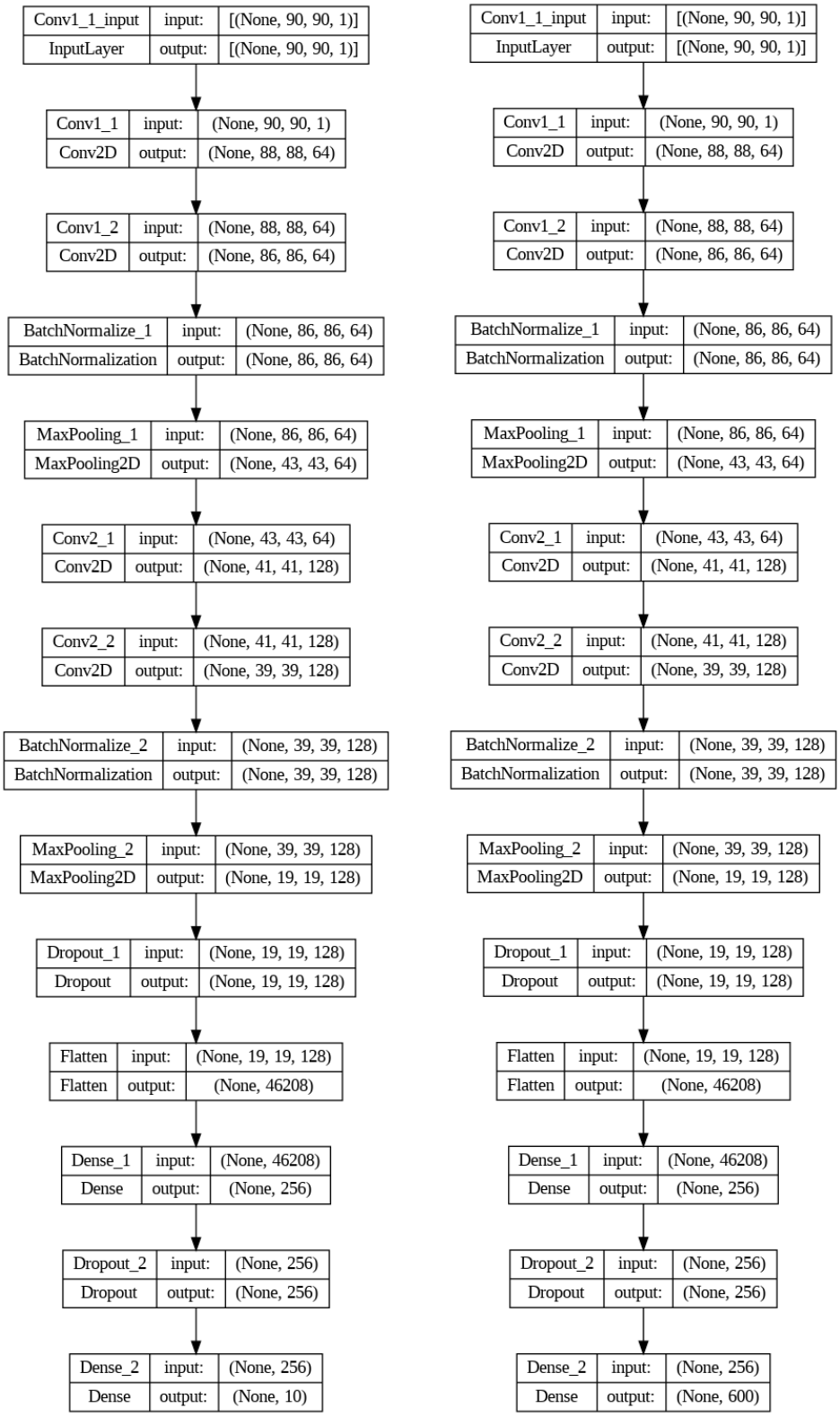
Mô hình có kiến trúc như sau:

- **Input layer:** Input của mô hình hình là ảnh dấu vân tay với kích thước input_shape = (90, 90, 1), là ảnh xám có kích thước chiều rộng và chiều dài lần lượt là 90 và 90.
- **Conv1_1:** Lớp convolution đầu tiên với 64 filter và kernel là (3, 3), lúc này, output mà ta sẽ có sau lớp Con2D_1 là (None , 88, 88, 64).
- **Conv1_2:** Lớp convolution thứ 2 với 64 filter và kernel là (3, 3), lúc này, output mà ta sẽ có sau lớp Con2D_2 là (None , 86, 86, 64).
- **BatchNormalize_1**: Chuẩn hóa dữ liệu sau 2 lớp Conv đầu tiên, output mà ta sẽ có sau lớp BatchNormalization_1 là (None , 86, 86, 64).
- **Maxpooling_1:** Ta sẽ rút chọn đặc trưng lại theo Maxpooling với kernel là (2,2), lúc này, output mà ta sẽ có sau lớp Maxpool2D_1 là (None, 43, 43, 64).
- **Conv2_1:** Lớp convolution thứ 3 với 128 filter và kernel là (3, 3), lúc này, output mà ta sẽ có sau lớp Conv2D_3 là (None, 41, 41, 128).
- **Conv2_2:** Lớp convolution thứ 4 với 128 filter và kernel là (3, 3), lúc này, output mà ta sẽ có sau lớp Conv2D_4 là (None, 39, 39, 128).

- **BatchNormalize_2**: Chuẩn hóa dữ liệu sau 2 lớp Conv tiếp theo, output mà ta sẽ có sau lớp BatchNormalization_1 là (None , 39, 39, 128).
- **Maxpooling_2:** Ta sẽ rút chọn đặc trưng lại theo Maxpooling với kernel là (2,2), lúc này, output mà ta sẽ có sau lớp Maxpool2D_2 là (None, 19, 19, 128).
- **Dropout_1:** Ta sẽ dropout một vài filter để mô hình không bị overfitting, output mà ta có sau lớp Dropout_1 là (19, 19, 128).
- **Flatten:** Duỗi các đặt trưng ra thành mảng có kích thước (None, 46208).
- **Dense_1:** Áp dụng một lớp Fully Connected Network (FCN) vào để mô hình học các đặc trưng đã được rút trích thông qua mạng CNN, output của lớp này sẽ là (None, 256).
- **Dropout_2:** Ta tiếp tục droupout một số node trong lớp Dense_1 để mô hình không bị overfitting, output của lớp này là (None, 256).
- **Dense_2:** Tiếp tục dùng một lớp FCN rút về kết quả output của model với kích thước là (None, 10) nếu là FingerNum_Model, (None, 600) nếu là SubjectID_Model, và sau đó sẽ được kích hoạt thông qua hàm softmax.
- **Output Layer:** Output của mô hình sẽ là (None, 10) nếu là FingerNum_Model, (None, 600) nếu là SubjectID_Model.

Bởi vì output đã được kích hoạt bởi hàm softmax, do đó ta phải dùng argmax để tìm vị trí lớp nhất trong output, điều này tương ứng với ID của người đó trong bộ dữ liệu.

Thông thường, để kiểm soát lại trường hợp dấu vân tay ngoài bộ dữ liệu, ta sẽ có một ngưỡng (threshold) nhất định để tránh trường hợp nhận người bên ngoài là người nội bộ. Threshold càng cao thì độ nhận dạng sẽ càng cao.



Hình 12. Kiến trúc của mô hình Finger Regconition với hình bên trái là kiến trúc của mô hình FingNum và hình bên phải là kiến trúc của mô hình SubjectID

4. Huấn luyện mô hình

Dữ liệu được chia thành 3 tập train, validation và test:

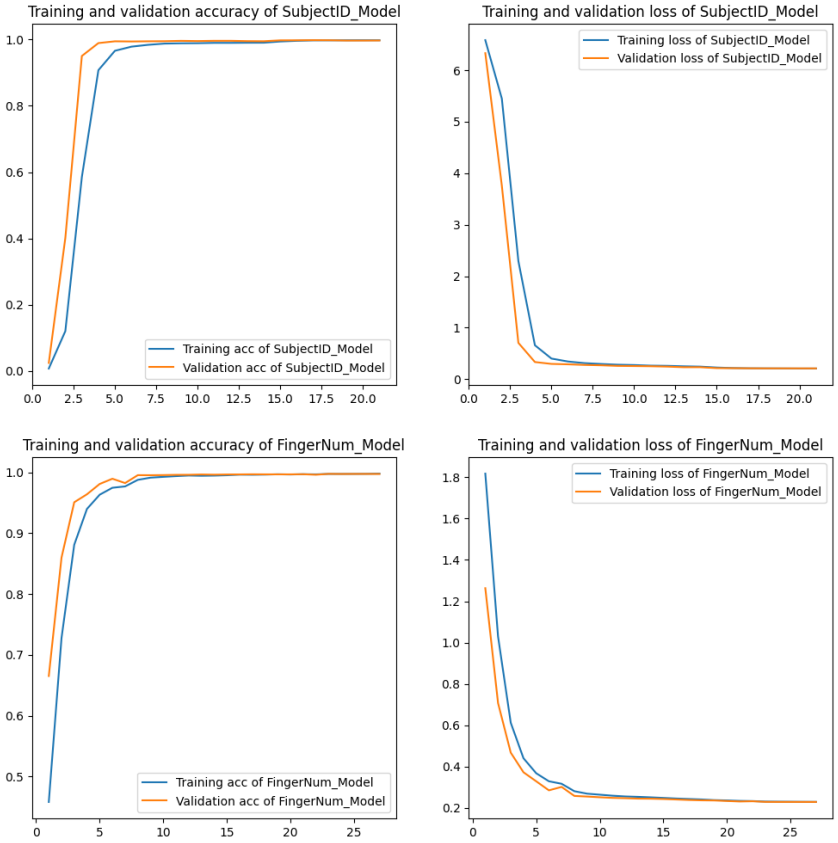
- Tập train và validation thuộc dữ liêu tập Altered, và được chia theo tỉ lệ 0.8 và 0.2.
- Tập test thuộc dữ liệu tập Real.

Mô hình sẽ được huấn luyện với tối đa 50 epochs và batchsize = 64 để mô hình học tốt hơn, tuy nhiên vì epochs khá cao nên ta sẽ có các lệnh Callbacks:

- EarlyStopping: Với patience = 4, xét theo monitor là 'val_accuracy'.
- ModelCheckpoint: Lưu vết mô hình trong quá trình huấn luyện, với monitor là 'val_accuracy' và ta chỉ lưu trọng số của mô hình tốt nhất.
- ReduceLROnPlateau: Giảm Learning Rate, giúp tăng tốc độ học cho mô hình.

5. Đánh giá mô hình

5.1. Đánh giá trên tập train và tập validation



Hình 13. Train - validation với loss và accuracy

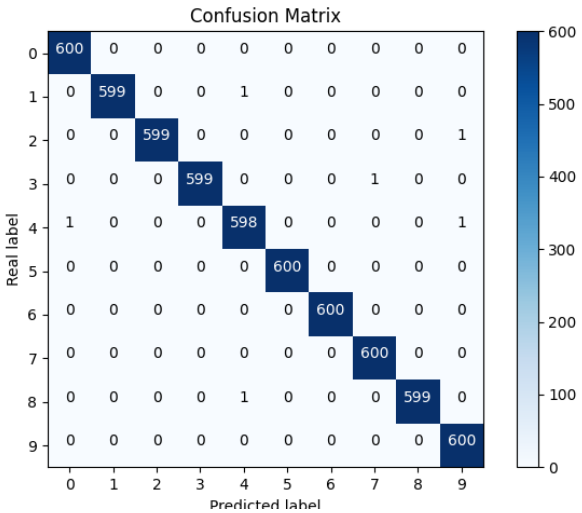
Ta có thể thấy mô hình được huấn luyện khá tốt với độ chính xác cao và độ lỗi thấp:

- Với mô hình **SubjectID_Model**:
  - Mô hình được huấn luyện với khoảng 21 epochs do mô hình không thể học được thêm nữa.
  - Accuracy của tập train và tập validation: Bắt đầu hội tụ tại epochs thứ 8 và đạt độ chính xác lần lượt ở tập train và validation là 0.9975 và 0.9982.
  - Loss của tập train và tập validation: Bắt đầu hội tụ tại epochs thứ 8 và đạt độ lỗi lần lượt ở tập train và validation là 0.2101 và 0.2000.
- Với mô hình **FingerNum_Model**:
  - Mô hình được huấn luyện với khoảng 27 epochs do mô hình không thể học được thêm nữa.
  - Accuracy của tập train và tập validation: Bắt đầu hội tụ tại epochs thứ 10 và đạt độ chính xác là 0.9976 và 0.9977
  - Loss của tập train và tập validation: Bắt đầu hội tụ tại epochs thứ 10 và đạt độ lỗi rất thấp khoảng 0.2308 và 0.2293.

5.2. Đánh giá trên tập test

Kiểm tra với tập test thì ta có thể thấy được kết quả đạt được của mô hình là rất tốt, và ta cũng có thể có cái nhìn trực quan hơn với ma trận confusion ở việc nhận diện 10 ngón tay (ở phần này em sẽ không làm confusion matrix cho ID vì số lượng nhiều, nhìn sẽ bị rối thay vì nhìn trực quan như trong Finger):

- Với nhận diện SubjectID, độ chính xác khoảng 99.75%.
- Với nhận diện FingerNum, độ chính xác khoảng 99.90%.



Hình 14. Ma trận hỗn loạn (Confusion matrix)

Với kết quả của tập test như trên, thì mô hình này có một chút hiệu quả hơn so với mô hình của [8], tuy không chênh lệch quá nhiều so với mô hình cũ nhưng kết quả vẫn rất cao và chính xác.

TÀI LIỆU THAM KHẢO

[1] Linh, T. (2023, April 14). *Công Nghệ Bảo Mật biometric scan – Sinh Trắc học là gì?*.Locker.
https://locker.io/blog/biometric-la-gi

[2] Devi, R. M., Keerthika, P., Suresh, P., Sarangi, P. P., Sangeetha, M., Sagana, C., & Devendran, K. (2022). Retina biometrics for personal authentication. *Machine Learning for Biometrics*, 87–104.
https://doi.org/10.1016/b978-0-323-85209-8.00005-5

[3] Alonso-Fernandez, F., Bigun, J., Fierrez, J., Fronthaler, H., Kollreider, K., & Ortega-Garcia, J. (2008). *Chapter 4: Fingerprint Recognition.*
https://www.diva-portal.org/smash/get/diva2:281340/FULLTEXT01.pdf

[4] Dijorajsenroy. (2020, November 5). *Fingerprint feature extraction for Biometrics*. Kaggle.
https://www.kaggle.com/code/dijorajsenroy/fingerprint-feature-extraction-for-biometrics

[5] Shehu, Y. I., Ruiz-Garcia, A., Palade, V., & James, A. (2018, July 24). *Sokoto Coventry Fingerprint Dataset*. arXiv.org.
https://arxiv.org/abs/1807.10609
Dataset:
V1: https://www.kaggle.com/datasets/dnyaneshwalwadkar/fingerprintdata-npz
V2: https://www.kaggle.com/datasets/ruizgara/socofing/data

[6] Simonyan, K., & Zisserman, A. (2015, April 10). Very deep convolutional networks for large-scale image recognition. arXiv.org.
https://arxiv.org/abs/1409.1556v6

[7] Understanding VGG16: Concepts, architecture, and performance. Datagen. (2023, May 22).
https://datagen.tech/guides/computer-vision/vgg16/

[8] Brianzz. (2020, July 9). *Subjectid&finger_cnnrecognizer*. Kaggle.
https://www.kaggle.com/code/brianzz/subjectid-finger-cnnrecognizer/notebook

[9] Xdevlabs. (2022, July 20). *Tổng quan về Vision Transformer (VIT)*. VinBigData.
https://vinbigdata.com/camera-ai/tong-quan-ve-vision-transformer-vit.html

[10] mayankgupta1609. (2023, September 20). *VIT_FT-2*. Kaggle.
https://www.kaggle.com/code/mayankgupta1609/vit-ft-2/notebook