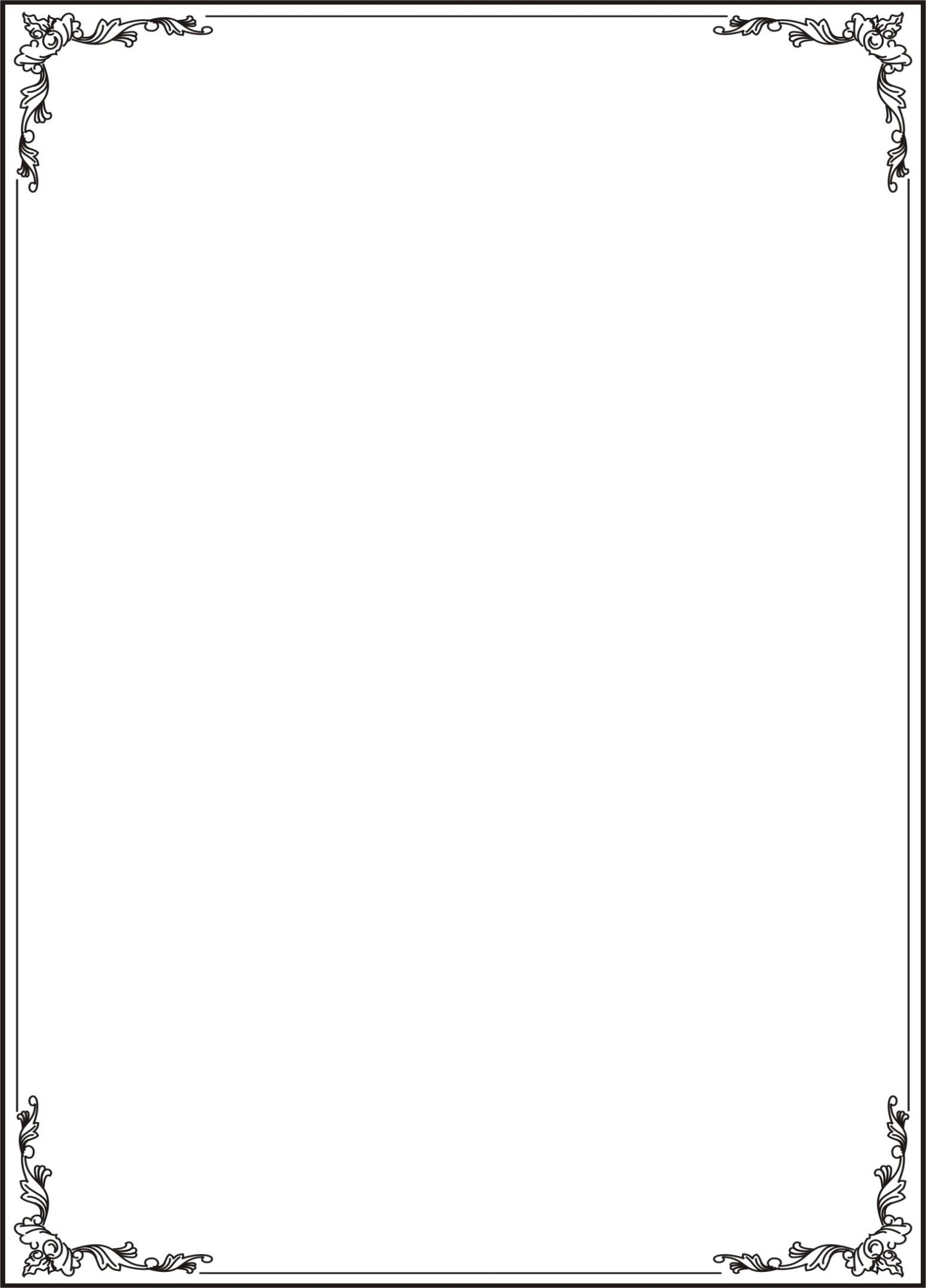
**TRƯỜNG ĐẠI HỌC LAO ĐỘNG – XÃ HỘI**



**KHÓA GIÁO DỤC ĐẠI CƯƠNG**

**--------\*\*\*--------**

**BÁO CÁO BÀI TẬP LỚN**

**TRÍ TUỆ NHÂN TẠO**

**ĐỀ TÀI**

ỨNG DỤNG MẠNG NERON TÍNH CHẬP NHẬN DẠNG ẢNH VÂN TAY

|  |  |
| --- | --- |
| Giảng viên hướng dẫn | Nguyễn Anh Thơ |
| Sinh viên thực hiện | Nguyễn Xuân Hiệp |
| Mã sinh viên | 1117093230 |
| Lớp tín chỉ | D16CN01 |

Hà Nội, năm 2023

# MỤC LỤC

[MỤC LỤC 1](#_Toc153725635)

[I. TÌM HIỂU TỔNG QUAN 3](#_Toc153725636)

[1. Mục Đích của Bài Toán Mục Đích của Bài Toán 3](#_Toc153725637)

[2 Phương Pháp Nhận Dạng và Ứng Dụng Hình Ảnh Vân Tay 4](#_Toc153725638)

[3 Các lĩnh vực liên quan 5](#_Toc153725639)

[II. MÔ HÌNH HỌC MẠNG NƠ-RON TÍCH CHẬP (CNN) 6](#_Toc153725640)

[1 Giới thiệu về cấu trúc và nguyên lý hoạt động của mạng nơ-ron tích chập 6](#_Toc153725641)

[2 Cấu trúc mạng nơ-ron tích chập 10](#_Toc153725642)

[III. ỨNG DỤNG MẠNG NƠ-RON TÍCH CHẬP CHO BÀI TOÁN NHẬN DẠNG HÌNH ẢNH VÂN TAY 13](#_Toc153725643)

[1. Nhận dạng và Xác minh Vân tay: 13](#_Toc153725644)

[2. Tiền xử lý và Biểu diễn Đặc trưng: 13](#_Toc153725645)

[3. Học Tự động và Phát hiện Gia mô Mẫu vân tay: 13](#_Toc153725646)

[4. Mở rộng sang Ứng dụng An ninh và Quản lý: 14](#_Toc153725647)

[IV. TRIỂN KHAI MÃ NGUỒN 15](#_Toc153725648)

[1. Tải và Cài đặt IDE Pycharm 15](#_Toc153725649)

[2. Tải và cài python 15](#_Toc153725650)

[3. Thư viện 15](#_Toc153725651)

[4. Dữ Liệu 15](#_Toc153725652)

[4. Tiến hành viết mã 16](#_Toc153725653)

[5. Chạy demo 17](#_Toc153725654)

[V. KẾT LUẬN 18](#_Toc153725655)

[VI. TÀI LIỆU THAM KHẢO 19](#_Toc153725656)

**LỜI MỞ ĐẦU**

Deep Learning làmột thuật toán dựa trên một số ý tưởng từ não bộ tới việc tiếp thu nhiều tầng biểu đạt, cả cụ thể lẫn trừu tượng, qua đó làm rõ nghĩa của các loại dữ liệu. Deep Learningđược ứng dụng trong nhận diện hình ảnh, nhận diện giọng nói, xử lý ngôn ngữ tự nhiên

Hiện nay rất nhiều các bài toán nhận dạng sử dụng deep learning để giải quyết do deep learning có thể giải quyết các bài toán với số lượng lớn, kích thước đầu vào lớn với hiệu năng cũng như độ chính xác vƣợt trội so với các phƣơng pháp phân lớp truyền thống

Những năm gần đây, ta đã chứng kiến đƣợc nhiều thành tựu vượt bậc trong ngành Thị giác máy tính (Computer Vision). Các hệ thống xử lý ảnh lớn như Facebook, Google hay Amazon đã đưa vào sản phẩm của mình những chức năng thông minh như nhận diện khuôn mặt ngƣời dùng, phát triển xe hơi tự lái hay drone giao hàng tự động.

Convolutional Neural Network (CNNs – Mạng nơ-ron tích chập) là một trong những mô hình Deep Learning tiên tiến giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay. Trong luận văn cao học này, em đi vào nghiên cứu về mạng neural cũng như mạng Convolution (tích chập) cũng như tưởng của mô hình CNNs trong phân lớp ảnh (Image Classification), và áp dụng trong việc xây dựng hệ thống nhận dạng biển số xe tự động

# I. TÌM HIỂU TỔNG QUAN

## 1. Mục Đích của Bài Toán Mục Đích của Bài Toán

Nhận dạng hình ảnh vân tay là một trong những phương pháp xác định và xác minh danh tính của một người dùng dựa trên các đặc điểm độc nhất của vân tay của họ. Mỗi vân tay có những đặc điểm độc nhất, bao gồm các vạch nẻo, điểm cuối, và các đặc trưng khác, tạo ra một mẫu vân tay duy nhất không giống ai khác.

1.1.Giới thiệu về vấn đề nhận dạng hình ảnh vân tay

Nhận dạng hình ảnh vân tay đóng vai trò quan trọng trong việc xác định và xác minh danh tính của một người dùng dựa trên các đặc điểm độc nhất của vân tay. Mục tiêu chính của việc nghiên cứu nhận dạng vân tay là phát triển các hệ thống có khả năng xác định vân tay một cách chính xác và an toàn. Việc này đem lại nhiều lợi ích, bao gồm:

Sự tăng cường bảo mật là một trong những lý do quan trọng khiến nhận dạng vân tay trở nên phổ biến. Công nghệ này giúp ngăn chặn truy cập trái phép vào thông tin cá nhân, dữ liệu quan trọng và các hệ thống quan trọng khác.

Sự tiện lợi và tốc độ trong việc xác minh danh tính thông qua vân tay là một điểm mạnh lớn. Tính năng này không chỉ giúp tăng cường tiện ích cho việc xác thực người dùng trên các thiết bị di động mà còn hỗ trợ trong các hệ thống an ninh lớn.

Ứng dụng của nhận dạng vân tay rất đa dạng, được áp dụng rộng rãi trong nhiều lĩnh vực như an ninh, quản lý cửa khẩu, thẻ thông minh và trong các thiết bị di động. Sự linh hoạt này giúp nâng cao tính bảo mật và ngăn chặn truy cập không được phép vào các hệ thống và thông tin quan trọng.

1.2 Lý Do Quan Trọng của Nghiên Cứu và Phát Triển Trong Lĩnh Vực Nhận Dạng Vân Tay:

Việc nghiên cứu và phát triển trong lĩnh vực nhận dạng vân tay đóng vai trò quan trọng trong việc xây dựng và cải thiện các hệ thống bảo mật, nâng cao sự tiện ích và đảm bảo tính chính xác và an toàn trong việc xác minh danh tính người dùng.

Bảo mật cao của vân tay là điểm mạnh lớn nhất trong việc xác minh cá nhân. Mỗi người đều có một mẫu vân tay duy nhất, tạo nên tính duy nhất và khó sao chép, làm cho việc sử dụng vân tay trở thành một trong những phương pháp an toàn nhất trong xác minh danh tính cá nhân.

Ứng dụng rộng rãi của việc nhận dạng vân tay đem lại sự tiện lợi và linh hoạt trong nhiều lĩnh vực khác nhau. Từ công nghiệp đến cuộc sống hàng ngày, việc áp dụng vân tay trong xác minh danh tính đóng vai trò quan trọng, giúp tăng cường bảo mật và nâng cao trải nghiệm người dùng.

Khả năng ngăn chặn gian lận thông tin và xâm nhập vào các hệ thống không được phép là một ưu điểm quan trọng của hệ thống nhận dạng vân tay. Việc sử dụng vân tay giúp ngăn chặn gian lận và truy cập trái phép, đảm bảo tính toàn vẹn của thông tin và dữ liệu quan trọng.

## 2 Phương Pháp Nhận Dạng và Ứng Dụng Hình Ảnh Vân Tay

2.1 Các phương pháp truyền thống và hiện đại trong nhận dạng vân tay.  
Trong lĩnh vực nhận dạng hình ảnh vân tay, có sự đa dạng trong việc áp dụng các phương pháp truyền thống và hiện đại. Các phương pháp truyền thống như đặc điểm thô (Minutiae-based) và đặc điểm cấu trúc (Ridge-based) tập trung vào việc xác định và sử dụng các đặc trưng cơ bản của vân tay như các điểm cuối, điểm phân nhánh và cấu trúc các đường nẻo để nhận dạng.

Ngoài ra, công nghệ hiện đại như mạng nơ-ron tích chập (CNNs), machine learning và deep learning đang được áp dụng rộng rãi trong việc nhận dạng vân tay. CNNs cho phép học và nhận dạng các đặc trưng phức tạp từ hình ảnh vân tay, trong khi machine learning sử dụng các thuật toán như SVMs, Random Forests và k-NN để phân loại và nhận dạng vân tay dựa trên các đặc trưng được trích xuất. Đặc biệt, deep learning thông qua các mô hình như CNNs, RNNs, và các kiến trúc mạng nơ-ron khác đã mang lại tiềm năng lớn cho việc học và nhận dạng các đặc trưng phức tạp của vân tay từ dữ liệu hình ảnh.

Sự đa dạng này trong các phương pháp và công nghệ nhận dạng vân tay đang tạo nên những tiến bộ đáng kể trong lĩnh vực này, cải thiện hiệu suất và độ chính xác của các hệ thống nhận dạng, đồng thời mở ra nhiều cơ hội mới cho việc áp dụng rộng rãi trong thực tế, từ an ninh, quản lý thông tin cá nhân đến các ứng dụng hàng ngày trong cuộc sống.

Công nghệ và ứng dụng của việc nhận dạng hình ảnh vân tay trong thực tế.

Công nghệ nhận dạng hình ảnh vân tay ngày càng trở nên phổ biến và được tích hợp rộng rãi trong nhiều lĩnh vực khác nhau, mang lại lợi ích đáng kể về bảo mật và tiện ích cho người dùng trong các tình huống khác nhau.

Trên các thiết bị di động, việc sử dụng nhận dạng vân tay để mở khóa hoặc xác thực thanh toán đã trở thành một tiêu chuẩn, cung cấp một cách tiện lợi và an toàn cho người dùng.

Trong lĩnh vực an ninh công nghiệp và doanh nghiệp, công nghệ nhận dạng vân tay được áp dụng rộng rãi trong các hệ thống kiểm soát ra vào, quản lý thời gian làm việc và xác định danh tính người dùng, giúp nâng cao tính an toàn và tiện ích trong quản lý nội bộ.

Trong lĩnh vực y tế, việc sử dụng nhận dạng vân tay giúp xác định chính xác danh tính bệnh nhân và truy xuất hồ sơ y tế cá nhân một cách an toàn và thuận tiện.

Công nghiệp tài chính và ngân hàng cũng không nằm ngoài lĩnh vực áp dụng của công nghệ nhận dạng vân tay. Việc sử dụng vân tay trong các giao dịch tài chính và xác minh danh tính người dùng giúp tăng cường bảo mật và ngăn chặn các hành vi gian lận.

Top of Form

## 3 Các lĩnh vực liên quan

Lĩnh vực nhận dạng vân tay đã thu hút sự quan tâm lớn từ cộng đồng nghiên cứu, và đã có nhiều công trình nghiên cứu đáng chú ý được thực hiện trước đó.

Các nghiên cứu trước đó thường tập trung vào việc phân tích, nhận biết và xác định các đặc điểm độc nhất của vân tay như các điểm minutiae (điểm cuối, điểm phân nhánh), cấu trúc của các đường nẻo, và các đặc điểm hình học khác để phát triển các phương pháp nhận dạng vân tay.

Nhiều nghiên cứu trước đây đã tập trung vào các phương pháp truyền thống như phương pháp minutiae-based và ridge-based, mà điều này đặt trọng tâm vào việc xác định các điểm đặc trưng cơ bản của vân tay để tạo ra các thuật toán nhận dạng.

Ngoài ra, sự phát triển của công nghệ đã mở ra các hướng nghiên cứu mới, đặc biệt là trong việc áp dụng machine learning và deep learning trong nhận dạng vân tay. Các nghiên cứu gần đây tập trung vào việc sử dụng các mô hình học máy và mạng nơ-ron sâu để học và nhận dạng các đặc trưng phức tạp hơn từ dữ liệu vân tay.

# II. MÔ HÌNH HỌC MẠNG NƠ-RON TÍCH CHẬP (CNN)

## 1 Giới thiệu về cấu trúc và nguyên lý hoạt động của mạng nơ-ron tích chập

1.1 Định nghĩa mạng nơ-ron tích chập

Những năm gần đây, ta đã chứng kiến được nhiều thành tựu vượt bậc trong ngành Thị giác máy tính (Computer Vision). Các hệ thống xử lý ảnh lớn như Facebook, Google hay Amazon đã đƣa vào sản phẩm của mình những chức năng thông minh như nhận diện khuôn mặt người dùng, phát triển xe hơi tự lái hay drone giao hàng tự động.

Convolutional Neural Network (CNNs – Mạng nơ-ron tích chập) là một trong những mô hình Deep Learning tiên tiến giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay. Trong luận văn này, chúng ta sẽ trình bày về Convolution (tích chập) cũng như ý tưởng của mô hình CNNs trong phân lớp chữ viết áp dụng trong bài toán nhận dạng ảnh vân tay (Image Classification).

1.2 Convolution (tích chập)

Tích chập được sử dụng đầu tiên trong xử lý tín hiệu số (Signal processing). Nhờ vào nguyên lý biến đổi thông tin, các nhà khoa học đã áp dụng kĩ thuật này vào xử lý ảnh và video số.

Để dễ hình dung, ta có thể xem tích chập như một cửa sổ trượt (sliding window) áp đặt lên một ma trận. Bạn có thể theo dõi cơ chế của tích chập qua hình minh họa bên dưới.



*Minh họa tích chập*

Ma trận bên trái là một bức ảnh đen trắng. Mỗi giá trị của ma trận tương đương với một điểm ảnh (pixel), 0 là màu đen, 1 là màu trắng (nếu là ảnh grayscale thì giá trị biến thiên từ 0 đến 255).

Sliding window còn có tên gọi là kernel, filter hay feature detector. Ở đây, ta dùng một ma trận filter 3×3 nhân từng thành phần tương ứng (element-wise) với ma trận ảnh bên trái. Giá trị đầu ra do tích của các thành phần này cộng lại. Kết quả của tích chập là một ma trận (convoled feature) sinh ra từ việc trượt ma trận filter và thực hiện tích chập cùng lúc lên toàn bộ ma trận ảnh bên trái. Dưới đây là một vài ví dụ của phép toán tích chập.

Ta có thể làm mờ bức ảnh ban đầu bằng cách lấy giá trị trung bình của các điểm ảnh xung quanh cho vị trí điểm ảnh trung tâm.



*Ảnh mờ sau khi chập*

Ngoài ra, ta có thể phát hiện biên cạnh bằng cách tính vi phân (độ dị biệt) giữa các điểm ảnh lân cận.



*Ảnh được phát hiện biên sau khi chập*

1.3 Mô hình mạng nơ-ron tích chập

Bây giờ, Chúng ta đã biết thế nào là convolution. Vậy CNNs là gì? CNNs chỉ đơn giản gồm một vài layer của convolution kết hợp với các hàm kích hoạt phi tuyến (nonlinear activation function) như ReLU hay tanh để tạo ra thông tin trừu tượng hơn (abstract/higher-level) cho các layer tiếp theo.

Trong mô hình Feedforward Neural Network (mạng nơ-ron truyền thẳng), các layer kết nối trực tiếp với nhau thông qua một trọng số w (weighted vector). Các layer này còn được gọi là có kết nối đầy đủ (fully connected layer) hay affine layer.

Trong mô hình CNNs thì ngƣợc lại. Các layer liên kết được với nhau thông qua cơ chế convolution. Layer tiếp theo là kết quả convolution từ layer trước đó, nhờ vậy mà ta có được các kết nối cục bộ. Nghĩa là mỗi nơ-ron ở layer tiếp theo sinh ra từ filter áp đặt lên một vùng ảnh cục bộ của nơ-ron layer trước đó.

Mỗi layer như vậy đƣợc áp đặt các filter khác nhau, thông thƣờng có vài trăm đến vài nghìn filter như vậy. Một số layer khác như pooling/subsampling layer dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu)

Trong suốt quá trình huấn luyện, CNNs sẽ tự động học đƣợc các thông số cho các filter. Ví dụ trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra thông số tối ưu cho các filter tương ứng theo thứ tự ***raw pixel > edges > shapes > facial > high-level features***. Layer cuối cùng được dùng để phân lớp ảnh.



*Mô hình mạng neural tích chập*

Trong đó:

INPUT: Tầng đầu vào

CONV: Tầng tích chập

RELU: Tầng kích hoạt. Thông qua hàm kích hoạt (activation function), thường là ReLU hoặc LeakyReLU để kích hoạt phi tuyến

POOL: Tầng tổng hợp, thông thường là Max pooling hoặc có thể là

Average pooling dùng để giảm chiều của ma trận đầu vào.

FC: Tầng kết nối hoàn toàn. Thông thường tầng này nằm ở sau cùng và kết nối với các đơn vị đại diện cho nhóm phân loại.

Có thể thấy một mạng neuron tích chập về cơ bản có 3 quá trình khác nhau:

Quá trình tích chập (convolution): Thông qua các tích chập giữa ma trận đầu vào với bộ lọc để tạo thành các đơn vị trong một tầng mới. Quá trình này có thể diễn ra liên tục ở phần đầu của mạng và thường sử dụng kèm với hàm kích hoạt ReLU. Mục tiêu của tầng này là trích suất đặc trưng hai chiều.

Quá trình tổng hợp (max pooling): Các tầng càng về sau khi trích xuất đặc trưng sẽ cần số lượng tham số lớn do chiều sâu được quy định bởi số lượng các kênh ở các tầng sau thường tăng tiến theo cấp số nhân. Điều đó làm tăng số lượng tham số và khối lượng tính toán trong mạng nơ ron. Do đó để giảm tải tính toán chúng ta sẽ cần giảm kích thước các chiều của khối ma trận đầu vào hoặc giảm số đơn vị của tầng. Vì mỗi một đơn vị sẽ là kết quả đại diện của việc áp dụng 1 bộ lọc để tìm ra một đặc trưng cụ thể nên việc giảm số đơn vị sẽ không khả thi. Giảm kích thước khối ma trận đầu vào thông qua việc tìm ra 1 giá trị đại diện cho mỗi một vùng không gian mà bộ lọc đi qua sẽ không làm thay đổi các đường nét chính của bức ảnh nhưng lại giảm được kích thước của ảnh. Do đó quá trình giảm chiều ma trận được áp dụng. Quá trình này gọi là tổng hợp nhằm mục đích giảm kích thước dài, rộng.

Quá trình kết nối hoàn toàn (fully connected): Sau khi đã giảm kích thước đến một mức độ hợp lý, ma trận cần được trải phẳng (flatten) thành một vector và sử dụng các kết nối hoàn toàn giữa các tầng. Quá trình này sẽ diễn ra cuối mạng CNN và sử dụng hàm kích hoạt là ReLU. Tầng kết nối hoàn toàn cuối cùng (fully connected layer) sẽ có số lượng đơn vị bằng với số classes và áp dụng hàm kích hoạt là softmax nhằm mục đích tính phân phối xác xuất.

CNNs có tính bất biến và tính kết hợp cục bộ (Location Invariance and Compositionality). Với cùng một đối tượng, nếu đối tượng này được chiếu theo các gốc độ khác nhau (translation, rotation, scaling) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể. Pooling layer sẽ cho bạn tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling).

Tính kết hợp cục bộ cho ta các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn thông qua convolution từ các filter. Đó là lý do tại sao CNNs cho ra mô hình với độ chính xác rất cao. Cũng giống như cách con người nhận biết các vật thể trong tự nhiên. Ta phân biệt dược một con chó với một con mèo nhờ vào các đặc trưng từ mức độ thấp (có 4 chân, có đuôi) đến mức độ cao (dáng đi, hình thể, màu lông).

layer dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu). Tuy nhiên, ta sẽ không đi sâu vào khái niệm của các layer này.

Trong suốt quá trình huấn luyện, CNNs sẽ tự động học được các thông số cho các filter. Ví dụ trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra thông số

## 2 Cấu trúc mạng nơ-ron tích chập

2.1 Convolutional Layer

Lớp convolutional (Convolutional Layer) là một phần quan trọng trong mạng neural convolutional (CNN) được sử dụng rộng rãi trong lĩnh vực xử lý ảnh và nhận dạng đối tượng. Lớp này đóng vai trò trích xuất các đặc trưng từ ảnh đầu vào bằng cách áp dụng bộ lọc convolutional (hay còn gọi là kernel hoặc filter) lên từng phần tử của ảnh.

Mỗi filter trong lớp convolutional có kích thước nhỏ hơn hoặc bằng kích thước của ảnh đầu vào. Qua quá trình tính toán, filter di chuyển trên toàn bộ ảnh và tính toán sự tương quan giữa filter và phần ảnh tương ứng. Quá trình này tạo ra một feature map, trong đó giá trị tại mỗi vị trí đại diện cho mức độ tương quan giữa filter và phần ảnh tương ứng.

Mỗi filter có thể được huấn luyện để nhận diện một loại đặc trưng cụ thể trong ảnh, chẳng hạn như cạnh, đường thẳng, hoặc hình tròn. Khi mạng CNN được huấn luyện, các filter sẽ được điều chỉnh sao cho tối ưu hóa việc trích xuất các đặc trưng quan trọng từ dữ liệu.

Kết quả của lớp convolutional là một tập hợp các feature maps, mỗi feature map đại diện cho một đặc trưng cụ thể được trích xuất từ ảnh đầu vào. Các feature maps này sẽ được đưa vào các lớp sau trong mạng CNN để tiếp tục quá trình học và phân loại.

Với khả năng trích xuất đặc trưng từ ảnh, lớp convolutional đóng vai trò quan trọng trong việc nhận dạng và phân loại các đối tượng trong ảnh. Qua đó, nó đã mang lại những tiến bộ đáng kể trong lĩnh vực xử lý ảnh và ứng dụng trong nhiều lĩnh vực khác nhau như nhận dạng khuôn mặt, phân loại sản phẩm, hay tự động lái xe.

2.2 Activation Functions

Lớp Activation Functions (hàm kích hoạt) là một phần quan trọng trong mạng neural, nó được sử dụng để giới hạn giá trị đầu ra của mỗi neuron trong mạng. Các hàm kích hoạt phổ biến bao gồm ReLU, LeakyReLU, ELU, Tanh và Sigmoid.

Hàm kích hoạt ReLU (Rectified Linear Unit) được sử dụng rộng rãi trong các mạng neural hiện đại. Nó loại bỏ các giá trị âm và chỉ giữ lại các giá trị dương. Điều này giúp giải quyết vấn đề vanishing gradient, một vấn đề phổ biến trong quá trình huấn luyện mạng neural.

Hàm kích hoạt LeakyReLU là một biến thể của ReLU. Nó cải thiện ReLU bằng cách cho phép một số giá trị âm nhỏ thông qua, để tránh tình trạng "dying" của neuron khi các giá trị âm quá nhỏ gây mất mát thông tin.

Hàm kích hoạt ELU (Exponential Linear Unit) cũng là một biến thể của ReLU. Nó cung cấp một độ cong mượt hơn và cho phép giá trị âm nhỏ hơn so với LeakyReLU. Điều này giúp khắc phục các vấn đề của ReLU và LeakyReLU.

Hàm kích hoạt Tanh và Sigmoid được sử dụng để áp dụng phép biến đổi đưa giá trị về khoảng [-1, 1] hoặc [0, 1]. Tanh thường được sử dụng trong các mạng neural với đầu ra dạng số thực, trong khi Sigmoid thường được sử dụng trong các mạng neural dùng cho bài toán phân loại nhị phân.

Các hàm kích hoạt này đóng vai trò quan trọng trong việc tạo ra độ linh hoạt và khả năng học của mạng neural. Chúng giúp mạng neural học các mô hình phức tạp và tăng cường khả năng biểu diễn thông tin trong dữ liệu. Sự lựa chọn hàm kích hoạt phù hợp là một yếu tố quan trọng để đạt được hiệu suất tốt của mạng neural trong các bài toán nhận dạng và phân loại.

2.3 Pooling Layer

Lớp Pooling Layer (lớp gộp) là một phần quan trọng trong mạng neural convolutional (CNN), nó được sử dụng để giảm kích thước không gian của các feature maps trong quá trình trích xuất đặc trưng. Có hai loại pooling phổ biến là Max Pooling và Average Pooling. Khi áp dụng Max Pooling, chúng ta chọn giá trị lớn nhất trong một vùng cục bộ (vùng pooling) của feature maps để đại diện cho vùng đó. Trong khi đó, khi áp dụng Average Pooling, chúng ta tính trung bình các giá trị trong vùng pooling để đại diện cho vùng đó. Lớp Pooling Layer có nhiều lợi ích. Đầu tiên, nó giúp loại bỏ nhiễu và thông tin không quan trọng từ feature maps. Bằng cách chọn giá trị lớn nhất hoặc trung bình, lớp Pooling Layer giúp tạo ra một phiên bản rút gọn và tóm tắt của feature maps, giảm bớt sự phức tạp và nhiễu trong dữ liệu. Thứ hai, lớp Pooling Layer cũng giúp giảm overfitting trong mạng neural. Overfitting xảy ra khi mạng neural học quá mức từ dữ liệu huấn luyện, dẫn đến việc không thể tổng quát hóa tốt cho dữ liệu mới. Bằng cách giảm kích thước không gian của feature maps, lớp Pooling Layer giúp giảm số lượng tham số và độ phức tạp của mạng, từ đó giảm nguy cơ overfitting. Thứ ba, lớp Pooling Layer giúp giảm chiều dữ liệu. Khi kích thước không gian của feature maps được giảm, số lượng thông tin cần xử lý cũng giảm đi. Điều này giúp tăng tốc độ tính toán và giảm bộ nhớ yêu cầu cho quá trình huấn luyện và dự đoán.

2.4 Fully-Connected Layer

Lớp Fully-Connected Layer (lớp kết nối đầy đủ) là một phần quan trọng trong mạng neural, nằm ở cuối mạng và được kết nối với output của các lớp convolutional và pooling. Lớp này thực hiện việc phân loại dựa trên các feature maps đã được trích xuất từ các lớp trước đó. Trong Fully-Connected Layer, mỗi neuron được kết nối với tất cả các neuron trong lớp trước đó. Điều này có nghĩa là mỗi neuron trong lớp này sẽ nhận đầu vào từ tất cả các feature maps của lớp trước đó. Các đầu vào này sẽ được trọng số hóa và kết hợp để tạo ra đầu ra của lớp Fully-Connected Layer. Lớp Fully-Connected Layer thực hiện việc phân loại dựa trên các feature maps đã được trích xuất từ các lớp trước đó. Qua quá trình huấn luyện, các trọng số trong lớp này sẽ được điều chỉnh để tối ưu hóa việc phân loại. Điều này cho phép mạng neural học các mối quan hệ phức tạp giữa các đặc trưng và các lớp đầu ra, và từ đó thực hiện việc phân loại dựa trên các đặc trưng này. Lớp Fully-Connected Layer thường được sử dụng trong các mạng neural cho bài toán phân loại, như mạng neural convolutional (CNN) cho việc nhận dạng hình ảnh. Các lớp trước đó (convolutional và pooling) giúp trích xuất các đặc trưng cấp cao từ hình ảnh, và lớp Fully-Connected Layer sẽ sử dụng các đặc trưng này để thực hiện phân loại.

# III. ỨNG DỤNG MẠNG NƠ-RON TÍCH CHẬP CHO BÀI TOÁN NHẬN DẠNG HÌNH ẢNH VÂN TAY

## 1. Nhận dạng và Xác minh Vân tay:

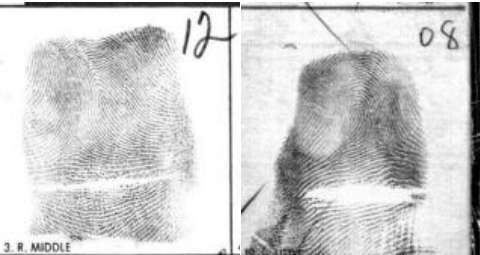
CNN được áp dụng để xây dựng các mô hình nhận dạng vân tay từ hình ảnh. Qua quá trình huấn luyện, mạng CNN có thể tự động học và nhận diện các đặc trưng độc nhất của từng ngón tay, bao gồm các dấu vân tay, hình dạng, và cấu trúc các đường vân. Các mô hình này có khả năng phân loại và nhận dạng vân tay một cách chính xác dựa trên các đặc trưng đã học.



CNN cũng có thể được sử dụng để xác minh tính chính xác của vân tay so với dữ liệu vân tay đã biết. Bằng cách so sánh các đặc trưng và biểu diễn số học của vân tay được chụp từ người dùng với dữ liệu vân tay được lưu trữ, mạng CNN có thể xác minh xem vân tay cung cấp có trùng khớp và chính xác không. Điều này thường được áp dụng trong các hệ thống bảo mật hoặc trong quá trình xác thực người dùng.

## 2. Tiền xử lý và Biểu diễn Đặc trưng:

CNN không chỉ được sử dụng để loại bỏ nhiễu và cải thiện chất lượng ảnh vân tay, mà còn có khả năng trích xuất các đặc trưng quan trọng. Các lớp tích chập của CNN có khả năng tạo ra biểu diễn đặc trưng của ảnh vân tay, giúp mô hình nhận dạng và phân loại vân tay dựa trên các đặc trưng này. Qua quá trình huấn luyện, CNN có thể học và nhận diện các đặc điểm đặc trưng độc nhất của vân tay, bao gồm các dấu vân tay, hình dạng và cấu trúc các đường vân.



## 3. Học Tự động và Phát hiện Gia mô Mẫu vân tay:

Mạng CNN có khả năng tự động học và nhận biết các mẫu và đặc điểm của vân tay thông qua quá trình huấn luyện. Điều này mang lại khả năng phát hiện các mẫu vân tay mới mà mô hình chưa từng thấy trước đó, tăng cường khả năng đa dạng và linh hoạt của mô hình nhận dạng. CNN cũng có thể sử dụng để phát hiện các thay đổi trong vân tay, như sự biến đổi về hình dạng hoặc phát hiện các mô hình vân tay giả mạo, từ đó nâng cao tính bảo mật và chính xác của hệ thống.

Bằng việc sử dụng CNN trong tiền xử lý và phân tích ảnh vân tay, không chỉ giúp cải thiện chất lượng ảnh mà còn mở ra những tiềm năng mới trong việc nâng cao hiệu suất, độ chính xác và đáng tin cậy của hệ thống nhận dạng và an ninh sử dụng vân tay. Điều này đóng vai trò quan trọng trong việc áp dụng công nghệ để đáp ứng nhu cầu an ninh và xác minh trong các lĩnh vực quan trọng.



## 4. Mở rộng sang Ứng dụng An ninh và Quản lý:

Việc áp dụng mạng nơ-ron tích chập (CNN) trong nhận dạng và xác minh vân tay đã mở ra một loạt các ứng dụng quan trọng trong lĩnh vực an ninh và quản lý. Việc này không chỉ cải thiện độ chính xác và hiệu suất mà còn mang lại nhiều lợi ích thực tiễn trong các hoạt động kiểm soát và quản lý.

CNN đã được tích hợp vào các hệ thống kiểm soát ra vào, tạo điều kiện cho việc nhận dạng và xác minh người dùng thông qua vân tay với độ chính xác cao. Bằng cách sử dụng CNN, việc theo dõi thời gian thực và quản lý ra vào trở nên chính xác và hiệu quả hơn, từ đó tăng cường bảo mật và ngăn chặn truy cập trái phép trong các khu vực quan trọng.

Các tổ chức an ninh và quản lý có thể tận dụng ưu điểm của CNN để cải thiện quy trình xác minh và nhận dạng người dùng. Việc tích hợp CNN vào hệ thống quản lý an ninh không chỉ tăng cường hiệu suất mà còn giúp giảm thiểu rủi ro và tăng cường tính bảo mật trong các hoạt động quản ly.



# IV. TRIỂN KHAI MÃ NGUỒN

## 1. Tải và Cài đặt IDE [Pycharm](https://www.jetbrains.com/pycharm/)

Pycharm là một nền tảng kết kết hợp được JetBrains phát triển như một IDE (Môi trường phát triển tích hợp) để phát triển các ứng dụng cho[lập trình](http://t3h.edu.vn/)trong Python. Một số ứng dụng lớn như Tweeter, Facebook, Amazon và Pinterest sử dụng Pycharm để làm IDE Python của họ.

Bài viết dưới đây sẽ giới thiệu chi tiết cho bạn về Pycharm cũng như hướng dẫn cách cài đặt và sử dụng Pycharm

## 2. Tải và cài python

**Bước 1**: Tải Python tại đây: [https://www.python.org/downloads/](https://quantrimang.com/url?q=aHR0cHM6Ly93d3cucHl0aG9uLm9yZy9kb3dubG9hZHMv), chọn phiên bản bạn cần, phiên bản mới nhất khi tôi viết bài viết này là Python 3.10.6.

**Bước 2**: Chạy tệp trình cài đặt và làm theo các bước để cài đặt Python

Trong quá trình cài đặt, hãy chọn **Add Python to environment variables**. Thao tác này sẽ thêm Python vào các biến môi trường và bạn có thể chạy Python từ bất kỳ phần nào của máy tính.

Ngoài ra, bạn có thể chọn đường dẫn nơi Python sẽ được cài đặt

## 3. Thư viện

Trước khi tiến hành viết mã chúng ta cần cài các thư viện sau để ứng dụng CNN

os Được sử dụng để tương tác với hệ thống tập tin và thư mục trong Python, chẳng hạn như đọc các tệp trong một thư mục cụ thể.

cv2 (OpenCV) Thư viện OpenCV (Open Source Computer Vision) là một thư viện mã nguồn mở chuyên về thị giác máy tính, được sử dụng để xử lý và xử lý ảnh.

matplotlib.pyplot Thư viện matplotlib cung cấp các công cụ vẽ đồ thị và hình ảnh trong Python. Trong đoạn mã của bạn, matplotlib.pyplot được sử dụng để hiển thị kết quả thông qua các hình ảnh.

## 4. Dữ Liệu

<https://www.kaggle.com/datasets/ruizgara/socofing/>

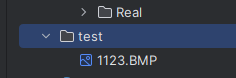
Bộ dữ liệu vân tay Sokoto Coventry (SOCOFing) là cơ sở dữ liệu vân tay sinh trắc học được thiết kế cho mục đích nghiên cứu học thuật. SOCOFing được tạo thành từ 6.000 hình ảnh dấu vân tay từ 600 đối tượng châu Phi và chứa các thuộc tính độc đáo như nhãn cho giới tính, tên bàn tay và ngón tay cũng như các phiên bản được thay đổi tổng hợp với ba mức độ thay đổi khác nhau để xóa, xoay trung tâm và cắt

## 4. Tiến hành viết mã

# Đọc thư viện và tệp cần thiết  
import os  
import cv2  
import matplotlib.pyplot as plt  
  
# Đọc ảnh mẫu (sample)  
sample = cv2.imread("test/1123.BMP")  
  
# Khởi tạo biến lưu điểm số tốt nhất và thông tin về ảnh có điểm số tốt nhất  
best\_score = 0  
filename = None  
image = None  
kp1, kp2, mp = None, None, None  
  
# Lặp qua từng tệp ảnh trong thư mục "SOCOFing/Real"  
counter = 0  
for file in os.listdir("SOCOFing\\Real"):  
 if counter % 1 == 0:  
 print(file)  
 counter += 1  
  
 # Đọc ảnh vân tay từ thư mục  
 fingerprint\_image = cv2.imread(os.path.join("SOCOFing\\Real", file))  
  
 # Kiểm tra nếu ảnh mẫu và ảnh vân tay có tồn tại  
 if sample is not None and fingerprint\_image is not None:  
 # Sử dụng thuật toán SIFT để tìm điểm đặc trưng và mô tả  
 sift = cv2.SIFT\_create()  
 keypoints\_1, descriptors\_1 = sift.detectAndCompute(sample, None)  
 keypoints\_2, descriptors\_2 = sift.detectAndCompute(fingerprint\_image, None)  
  
 # Kiểm tra xem các descriptors có giá trị không rỗng  
 if descriptors\_1 is not None and descriptors\_2 is not None:  
 # So khớp các điểm đặc trưng giữa ảnh mẫu và ảnh vân tay  
 matches = cv2.FlannBasedMatcher({'algorithm': 1, 'trees': 10}, {}).knnMatch(descriptors\_1, descriptors\_2, k=2)  
 match\_points = []  
  
 # Lọc các điểm khớp dựa trên ngưỡng  
 for p, q in matches:  
 if p.distance < 0.1 \* q.distance:  
 match\_points.append(p)  
  
 keypoints = min(len(keypoints\_1), len(keypoints\_2))  
  
 # Tính toán điểm số dựa trên số điểm khớp và số điểm đặc trưng  
 if keypoints > 0:  
 current\_score = len(match\_points) / keypoints \* 1  
  
 # Cập nhật điểm số tốt nhất và lưu thông tin về ảnh có điểm số tốt nhất  
 if current\_score > best\_score:  
 best\_score = current\_score  
 filename = file  
 image = fingerprint\_image  
 kp1, kp2, mp = keypoints\_1, keypoints\_2, match\_points  
  
# In ra tên file và điểm số khớp tốt nhất  
print("BEST MATCH:", filename)  
print("Score:", best\_score)  
  
# Hiển thị kết quả của điểm khớp tốt nhất (nếu có)  
if image is not None:  
 result = cv2.drawMatches(sample, kp1, image, kp2, mp, None)  
 result = cv2.resize(result, None, fx=4, fy=4)  
  
 # Sử dụng Matplotlib để hiển thị ảnh  
 plt.imshow(cv2.cvtColor(result, cv2.COLOR\_BGR2RGB))  
 plt.axis('off') # Tắt trục  
 plt.show()  
else:  
 print("Không tìm thấy ảnh với điểm khớp tốt nhất.")

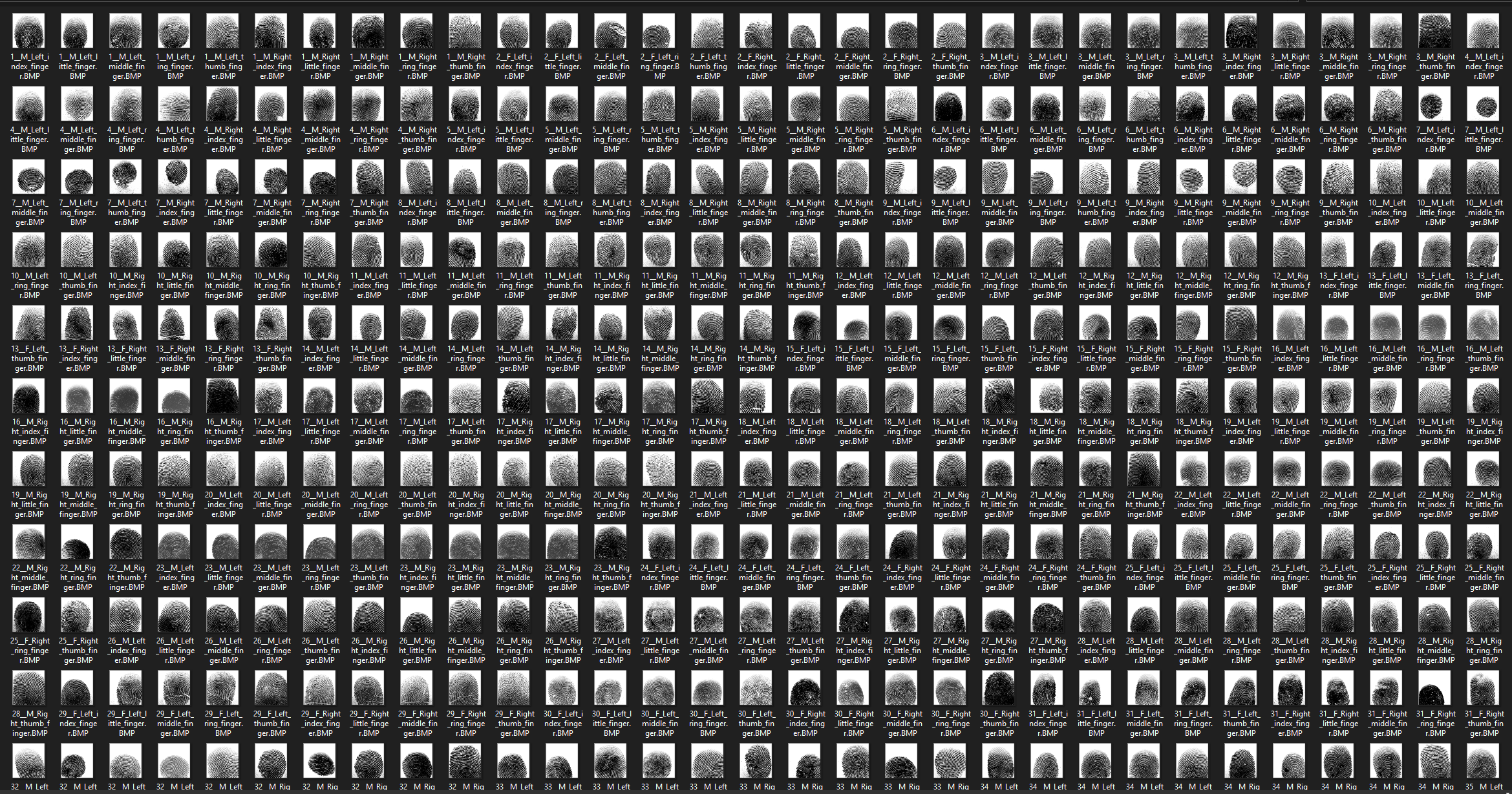
## 5. Chạy demo

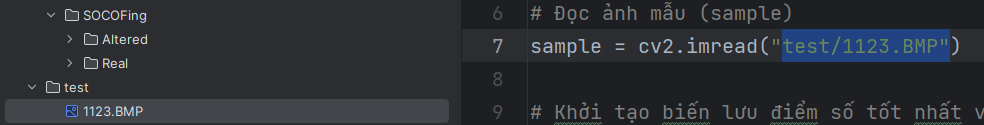
Đầu tiên ở mục Project có 1 derectory là “test” như hình bên dưới



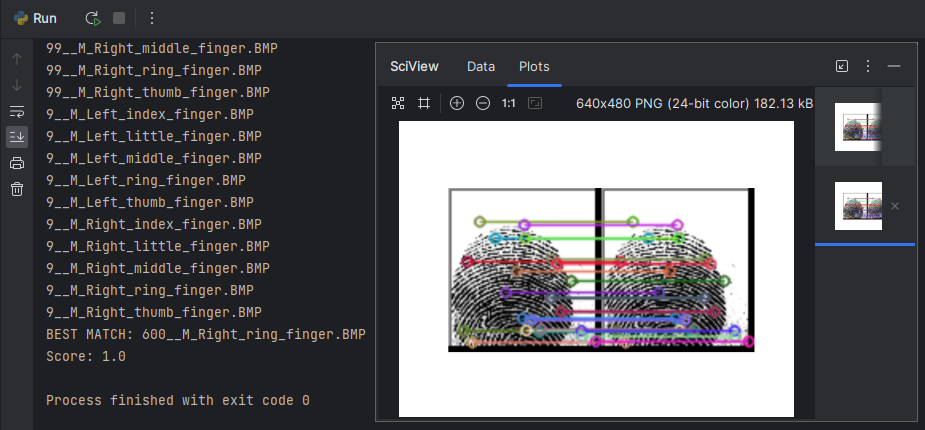
Tại đây có sẵn 1 file BMP để test

Hoặc có thế lấy dữ liệu ở trong thư mục “SOCOFing\\Real”





Sau đó đổi đường dẫn tới file test 1123.BMP như trên



Kết quả sau khi chạy sẽ hiển thị kết quả trùng khớp

# V. KẾT LUẬN

Hiệu suất đáng kinh ngạc CNNs đã chứng minh khả năng vượt trội trong việc nhận dạng và phân loại hình ảnh so với các phương pháp truyền thống. Khả năng học và tự điều chỉnh của chúng giúp cải thiện độ chính xác của các ứng dụng nhận dạng hình ảnh.

Ứng dụng đa dạng CNNs không chỉ dừng lại ở việc nhận dạng vật thể và đối tượng, mà còn được áp dụng rộng rãi trong nhiều lĩnh vực khác như y tế, nông nghiệp, tự động hóa, thực tế ảo, và nhiều ứng dụng khác.

Tiềm năng phát triển CNNs vẫn còn tiềm năng phát triển lớn. Các nhà nghiên cứu và các nhà phát triển có thể tập trung vào việc cải tiến hiệu suất, tốc độ, và tích hợp với các công nghệ mới để tạo ra ứng dụng mạnh mẽ hơn.

Thách thức Mặc dù CNNs mang lại nhiều lợi ích, nhưng cũng đối mặt với những thách thức như đòi hỏi lượng dữ liệu lớn, tính toán cao, và khả năng diễn giải dự đoán. Việc giải quyết những thách thức này là một trong những hướng phát triển quan trọng của CNNs trong tương lai.

Sự hỗ trợ từ cộng đồng Sự hợp tác giữa cộng đồng nghiên cứu, công nghiệp, và các tổ chức đào tạo đóng vai trò quan trọng trong việc nghiên cứu, phát triển và ứng dụng CNNs vào thực tế.

Trong quá trình thực hiện bài tập lớn môn học-nhập môn học máy và khai phá giữ liệu, nhóm em đã tìm hiểu và biết thêm về các ứng dụng của môn học trong thực tế. Qua đó giúp nhóm hoàn thiện sản phầm và báo cáo cho môn học.Với đề tài đưa ra là so sánh hai mẫu vân tay bất kỳ, nhóm em đưa ra cách giải quyết là sử dụng mạng noron nhân tạo(CNN) để tiến hành nhận dạng hình ảnh vân tay đầu vào và đưa ra dự đoán sau cùng.Vì trong thời gian ngắn nên bài tập lớn của nhóm em vẫn còn nhiều thiếu sót và chưa thể xử lý tốt nhất vấn đề đặt ra, nhóm mong được sự góp ý từ cô cũng như các bạn để có thể hoàn thiện hơn nữa.Chúng em xin chân thành cảm ơn!

# VI. TÀI LIỆU THAM KHẢO

<https://phamdinhkhanh.github.io/2019/08/22/convolutional-neural-network.html>

<https://medium.com/swlh/fingerprint-gender-classification-cnn-8b052a162a93>

<https://www.researchgate.net/publication/4357761_A_Fully_CNN_Based_Fingerprint_Recognition_System#:~:text=Abstract%20In%20this%20pape%20r%2C%20a%20fully%20cellular%20neural,is%20matched%20with%20th%20e%20fingerprints%20in%20the%20database>

<https://www.kaggle.com/datasets/ruizgara/socofing/>