ĐỀ TÀI: HỆ THỐNG CHẤM CÔNG NHẬN DẠNG KHUÔN MẶT BẰNG MÔ HÌNH FACENET

**MỤC LỤC**

[THÔNG TIN CHUNG i](#_Toc201033298)

[LỜI CẢM ƠN ii](#_Toc201033299)

[LỜI CAM ĐOAN iii](#_Toc201033300)

[DANH MỤC HÌNH ẢNH vi](#_Toc201033301)

[DANH MỤC BẢNG BIỂU vii](#_Toc201033302)

[PHẦN MỞ ĐẦU 1](#_Toc201033303)

[1. Lý do chọn đề tài 1](#_Toc201033304)

[2. Mục tiêu nghiên cứu 1](#_Toc201033305)

[3. Ý nghĩa khoa học và thực tiễn 2](#_Toc201033306)

[4. Cấu trúc báo cáo 2](#_Toc201033307)

[CHƯƠNG 1. GIỚI THIỆU VÀ TỔNG QUAN VỀ ĐỀ TÀI 4](#_Toc201033308)

[1.1. Hệ thống điểm danh và quản lý truyền thống 4](#_Toc201033309)

[1.1.1. Các hệ thống điểm danh và quản lý truyền thống hiện nay 4](#_Toc201033310)

[1.1.2. Các vấn đề hiện tải của hệ thống điểm danh và quản lý truyền thống 4](#_Toc201033311)

[1.2. Giải pháp và phạm vi thực hiện đề tài 5](#_Toc201033312)

[1.2.1. Giải pháp 5](#_Toc201033313)

[1.2.2. Phạm vi thực hiện 5](#_Toc201033314)

[1.3. Mô tả bài toán 6](#_Toc201033315)

[1.3.1. Mô tả chung 6](#_Toc201033316)

[1.3.2. Input của bài toán 7](#_Toc201033317)

[1.3.3. Output của bài toán 8](#_Toc201033318)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT VÀ XỬ LÝ DỮ LIỆU 10](#_Toc201033319)

[2.1. Cơ sở lý thuyết và thuật toán 10](#_Toc201033320)

[2.1.1. Tổng quan về học máy 10](#_Toc201033321)

[2.1.2. Tổng quan về thị giác máy tính 11](#_Toc201033322)

[2.1.3. Công nghệ sử dụng (FaceNet) 12](#_Toc201033323)

[2.2. Thu thập và xử lý dữ liệu 14](#_Toc201033324)

[2.2.1. Thu thập dữ liệu 14](#_Toc201033325)

[2.2.2. Phân tích bộ dữ liệu 14](#_Toc201033326)

[2.2.3. Tiền xử lý dữ liệu 15](#_Toc201033327)

[CHƯƠNG 3. PHÁT TRIỂN MÔ HÌNH 20](#_Toc201033328)

[3.1. Phát triển mô hình 20](#_Toc201033329)

[3.1.1. Kiến trúc InceptionResnetV1 20](#_Toc201033330)

[3.1.2. Sử dụng mô hình huấn luyện trước (Pre-trained Model) 21](#_Toc201033331)

[3.1.3. Triển khai với thư viện facenet\_pytorch 21](#_Toc201033332)

[3.1.4. Vai trò của mô hình trong FaceNet 22](#_Toc201033333)

[3.2. Huấn luyện mô hình 23](#_Toc201033334)

[3.2.1. Khởi tạo Bộ Tối Ưu Hóa và Điều Chỉnh Tốc Độ Học Tập 23](#_Toc201033335)

[3.2.2. Vòng Lặp Huấn Luyện Chính 23](#_Toc201033336)

[CHƯƠNG 4. ĐÁNH GIÁ VÀ TRIỂN KHAI HỆ THỐNG 26](#_Toc201033337)

[4.1. Đánh giá mô hình 26](#_Toc201033338)

[4.2. Triển khai hệ thống 27](#_Toc201033339)

[KẾT LUẬN 37](#_Toc201033340)

[1. Tổng kết kết quả đạt được 37](#_Toc201033341)

[2. Hạn chế của hệ thống 37](#_Toc201033342)

[3. Hướng phát triển 38](#_Toc201033343)

[TÀI LIỆU THAM KHẢO 40](#_Toc201033344)

# DANH MỤC HÌNH ẢNH

[Hình 3.1. Triển khai mô hình FaceNet. 22](#_Toc201033287)

[Hình 3.2. Huấn luyện mô hình. 25](#_Toc201033288)

[Hình 4.1. Giao diện đăng nhập. 28](#_Toc201033289)

[Hình 4.2. Giao diện đăng ký. 29](#_Toc201033290)

[Hình 4.3. Giao diện quên mật khẩu. 30](#_Toc201033291)

[Hình 4.4. Giao diện xác thực OTP. 31](#_Toc201033292)

[Hình 4.5. Giao diện đặt lại mật khẩu. 32](#_Toc201033293)

[Hình 4.6. Giao diện hiển thị thông tin của các nhân viên. 33](#_Toc201033294)

[Hình 4.7. Giao diện thêm một nhân viên mới. 34](#_Toc201033295)

[Hình 4.8. Giao diện hiển thị danh sách chấm công theo ngày 35](#_Toc201033296)

[Hình 4.9. Giao diện hiển thị thông tin điểm danh của một nhân viên trong một tháng. 36](#_Toc201033297)

# DANH MỤC BẢNG BIỂU

[Bảng 4.1. Bảng đánh giá mô hình 26](#_Toc201033279)

# PHẦN MỞ ĐẦU

## 1. Lý do chọn đề tài

Trong bối cảnh cuộc cách mạng công nghiệp 4.0, trí tuệ nhân tạo (AI) và thị giác máy tính (Computer Vision) đang định hình lại cách thức vận hành của các tổ chức hiện đại, từ giáo dục, doanh nghiệp đến các lĩnh vực công cộng. Hệ thống điểm danh truyền thống, dựa trên các phương pháp như gọi tên thủ công, ký tên trên giấy, hoặc sử dụng thẻ từ/vân tay, bộc lộ nhiều hạn chế nghiêm trọng: tốn thời gian, dễ xảy ra gian lận (điểm danh hộ), thiếu chính xác, và khó khăn trong việc quản lý dữ liệu. Những vấn đề này không chỉ làm giảm hiệu quả hoạt động mà còn cản trở quá trình chuyển đổi số trong các môi trường như trường học, nhà máy, hay tổ chức sự kiện.

Sự phát triển mạnh mẽ của các công nghệ nhận diện khuôn mặt, đặc biệt là các mô hình học sâu như FaceNet và MTCNN, đã mở ra cơ hội xây dựng các hệ thống điểm danh tự động thông minh, minh bạch và hiệu quả. Đề tài **Hệ thống chấm công nhận dạng khuôn mặt bằng mô hình FaceNet** được lựa chọn nhằm tận dụng những tiến bộ này để giải quyết bài toán thực tiễn, đáp ứng nhu cầu quản lý điểm danh trong các môi trường có quy mô lớn. Việc triển khai hệ thống trên thiết bị nhúng như Raspberry Pi không chỉ đảm bảo tính khả thi về chi phí mà còn phù hợp với xu hướng phát triển các giải pháp công nghệ gọn nhẹ, dễ tiếp cận. Đề tài không chỉ mang ý nghĩa học thuật mà còn có tiềm năng ứng dụng rộng rãi, góp phần thúc đẩy sự đổi mới trong quản lý và vận hành.

## 2. Mục tiêu nghiên cứu

Đề tài hướng đến việc thiết kế và phát triển một **hệ thống điểm danh** dựa trên nhận diện khuôn mặt, sử dụng các công nghệ học máy và thị giác máy tính để thay thế các phương pháp truyền thống. Các mục tiêu cụ thể bao gồm:

* **Tự động hóa quy trình điểm danh**: Xây dựng hệ thống nhận diện khuôn mặt thời gian thực, giảm thiểu thời gian và công sức so với các phương pháp thủ công.
* **Đảm bảo độ chính xác và minh bạch**: Ứng dụng các thuật toán học sâu như FaceNet và MTCNN để nhận diện khuôn mặt với độ chính xác cao, ngăn chặn gian lận (điểm danh hộ) thông qua xác thực sinh trắc học.
* **Quản lý dữ liệu hiệu quả**: Tạo ra cơ sở dữ liệu điểm danh dễ truy xuất, hỗ trợ phân tích thời gian thực và xuất báo cáo, đáp ứng nhu cầu quản lý của các tổ chức.
* **Triển khai trên thiết bị nhúng**: Phát triển hệ thống hoạt động ổn định trên các thiết bị như Raspberry Pi kết hợp với camera, đảm bảo tính gọn nhẹ, chi phí thấp và khả năng ứng dụng thực tế.
* **Xây dựng giao diện thân thiện**: Thiết kế giao diện người dùng (UI) đơn giản, dễ sử dụng, cùng khả năng tích hợp với các nền tảng như ứng dụng di động hoặc API trong tương lai.

## 3. Ý nghĩa khoa học và thực tiễn

**Về mặt khoa học**, đề tài đóng góp vào việc nghiên cứu và ứng dụng các kỹ thuật tiên tiến trong lĩnh vực trí tuệ nhân tạo và thị giác máy tính, cụ thể là các mô hình học sâu như FaceNet và MTCNN. Đề tài tập trung vào các khía cạnh quan trọng như thu thập dữ liệu, tiền xử lý, trích chọn đặc trưng, xây dựng và tối ưu hóa mô hình, cũng như triển khai trên thiết bị nhúng. Những nội dung này không chỉ củng cố kiến thức lý thuyết về học máy mà còn mở ra cơ hội khám phá các kỹ thuật nâng cao như tăng cường dữ liệu (data augmentation), căn chỉnh khuôn mặt (face alignment), và tối ưu hóa mô hình cho phần cứng hạn chế.

**Về mặt thực tiễn**, hệ thống điểm danh tự động mang lại giải pháp thiết thực cho các môi trường như trường học, doanh nghiệp vừa và nhỏ, bệnh viện, hoặc sự kiện lớn. Hệ thống giúp tiết kiệm thời gian, nâng cao tính minh bạch, và cung cấp dữ liệu đáng tin cậy cho việc quản lý nhân sự hoặc sinh viên. Việc triển khai trên thiết bị nhúng như Raspberry Pi đảm bảo tính khả thi về chi phí và khả năng ứng dụng rộng rãi. Hơn nữa, hệ thống có tiềm năng mở rộng để tích hợp với các nền tảng thông báo (SMS, ứng dụng di động) và phân tích dữ liệu nâng cao, góp phần thúc đẩy xu hướng chuyển đổi số trong quản lý.

## 4. Cấu trúc báo cáo

Báo cáo được tổ chức thành 5 chương, phản ánh quy trình phát triển hệ thống từ lý thuyết đến thực tiễn, đảm bảo tính logic và bao quát các yêu cầu của đề tài:

* **Chương 1**: Khai thác tổng quan đề tài, xác định bài toán trí tuệ nhân tạo, phạm vi nghiên cứu và nền tảng lý thuyết cốt lõi.
* **Chương 2**: Trình bày quy trình thu thập dữ liệu, các kỹ thuật tiền xử lý và phương pháp trích xuất đặc trưng cho hệ thống.
* **Chương 3**: Mô tả chi tiết quá trình thiết kế, xây dựng và huấn luyện các mô hình học máy phục vụ nhận diện khuôn mặt.
* **Chương 4**: Phân tích hiệu quả của mô hình thông qua các độ đo đánh giá và quá trình triển khai hệ thống trong môi trường thực tế.
* **Chương 5**: Tổng kết những thành tựu đạt được, nhận diện hạn chế, đề xuất hướng phát triển tương lai và cách thức quản lý dự án.

Báo cáo được trình bày rõ ràng, đi kèm mã nguồn sạch sẽ, comment đầy đủ, quản lý phiên bản qua Git, và tài liệu tham khảo chi tiết, nhằm đảm bảo tính khoa học, dễ tra cứu và bảo trì. Đề tài không chỉ là một nghiên cứu học thuật mà còn mang lại giá trị thực tiễn, góp phần vào sự đổi mới trong quản lý điểm danh và ứng dụng AI.

# GIỚI THIỆU VÀ TỔNG QUAN VỀ ĐỀ TÀI

## 1.1. Hệ thống điểm danh và quản lý truyền thống

### 1.1.1. Các hệ thống điểm danh và quản lý truyền thống hiện nay

Hiện nay, nhiều nơi vẫn sử dụng các phương pháp điểm danh truyền thống, bao gồm: Gọi tên thủ công:

* Giảng viên hoặc quản lý đọc tên từng người trong danh sách, học sinh hoặc nhân viên trả lời để xác nhận sự có mặt.
* Ký tên trên giấy: Mọi người ký tên vào một bảng điểm danh hoặc sổ tay để ghi nhận sự hiện diện.
* Sử dụng bảng điểm danh Excel: Một số nơi nhập thông tin điểm danh vào file Excel, thường sau khi gọi tên hoặc kiểm tra thủ công.
* Hệ thống thẻ từ hoặc vân tay: Một số doanh nghiệp dùng máy quét thẻ hoặc vân tay để chấm công, nhưng vẫn cần nhân viên xếp hàng và thao tác thủ công.

Những hệ thống này phổ biến vì đơn giản và không đòi hỏi công nghệ phức tạp. Tuy nhiên, chúng thường chỉ phù hợp cho quy mô nhỏ và không tận dụng được các công nghệ hiện đại như AI hay thị giác máy tính.

### 1.1.2. Các vấn đề hiện tải của hệ thống điểm danh và quản lý truyền thống

Phương pháp điểm danh thủ công, chẳng hạn như gọi tên theo danh sách hoặc yêu cầu người học/nhân viên ký tên vào bảng điểm danh, vẫn đang được sử dụng rộng rãi tại nhiều cơ sở giáo dục và doanh nghiệp. Tuy nhiên, các phương pháp này tồn tại nhiều hạn chế nghiêm trọng, bao gồm:

* **Tốn thời gian:** Việc gọi tên từng cá nhân trong các lớp học đông (ví dụ, 50-100 sinh viên) hoặc tại các nhà máy có hàng trăm công nhân có thể mất từ 5-15 phút mỗi buổi, gây lãng phí thời gian quý báu dành cho giảng dạy hoặc sản xuất.
* **Dễ xảy ra gian lận:** Hiện tượng điểm danh hộ (một người điểm danh thay cho người khác) rất phổ biến, đặc biệt trong môi trường học đường, làm giảm tính minh bạch và công bằng.
* **Thiếu chính xác:** Trong quá trình gọi tên, giảng viên hoặc quản lý có thể gọi nhầm, bỏ sót, hoặc ghi nhận sai thông tin, dẫn đến dữ liệu điểm danh không đáng tin cậy.
* **Khó khăn trong quản lý dữ liệu:** Việc tổng hợp, lưu trữ và phân tích dữ liệu điểm danh từ các bảng giấy hoặc file Excel thủ công thường phức tạp, tốn công sức, dễ xảy ra sai sót và khó truy xuất khi cần báo cáo hoặc kiểm tra.

Những hạn chế này không chỉ làm giảm hiệu quả hoạt động mà còn ảnh hưởng đến chất lượng quản lý và ra quyết định của các tổ chức.

## 1.2. Giải pháp và phạm vi thực hiện đề tài

### 1.2.1. Giải pháp

Để khắc phục các vấn đề trên, hệ thống điểm danh tự động sử dụng công nghệ nhận diện khuôn mặt được đề xuất như một giải pháp toàn diện. Hệ thống này tích hợp các công nghệ tiên tiến như:

* **FaceNet:** Một mô hình học sâu (Deep Learning) được sử dụng để chuyển đổi hình ảnh khuôn mặt thành các vectơ đặc trưng (facial embeddings) với độ chính xác cao. FaceNet giúp so sánh và xác thực danh tính dựa trên các đặc điểm khuôn mặt, đảm bảo độ tin cậy trong việc nhận diện.
* **MTCNN (Multi-task Cascaded Convolutional Networks):** Một thuật toán phát hiện khuôn mặt mạnh mẽ, sử dụng mạng nơ-ron tích chập để xác định chính xác vị trí và vùng chứa khuôn mặt trong ảnh hoặc video, ngay cả trong điều kiện ánh sáng phức tạp hoặc góc quay không lý tưởng.

Hệ thống được triển khai trên các thiết bị nhúng (như Raspberry Pi hoặc các bo mạch tương tự) kết hợp với camera độ phân giải cao, cho phép ghi nhận và xử lý hình ảnh trong thời gian thực. Phần mềm đi kèm được phát triển với giao diện thân thiện, dễ sử dụng, và có khả năng tích hợp với các thiết bị ngoại vi như màn hình, máy quét hoặc cảm biến. Mục tiêu chính của hệ thống là:

* Tự động hóa quy trình điểm danh, giảm thiểu thời gian và công sức.
* Tăng tính minh bạch, hạn chế gian lận thông qua xác thực sinh trắc học (khuôn mặt).
* Cung cấp dữ liệu điểm danh chính xác, dễ dàng truy xuất và phân tích.

### 1.2.2. Phạm vi thực hiện

Hệ thống này được thiết kế với các tính năng chính:

*Nhận diện khuôn mặt:*

* Nhận diện chính xác khuôn mặt của sinh viên hoặc nhân viên, kể cả trong điều kiện ánh sáng không tốt hay khi người đó đeo kính, đội mũ.
* Hỗ trợ nhận diện nhiều nhân viên trong phòng cùng một lúc, tối ưu hoá thời gian điểm danh so với các phương pháp truyền thống.

*Quản lý dữ liệu tự động:*

* Mọi thông tin điểm danh được lưu tự động vào cơ sở dữ liệu, dễ dàng tra cứu hoặc xuất ra báo cáo.
* Hệ thống hỗ trợ gửi thông báo điểm danh qua ứng dụng, giúp quản lý và nhân viên có thể nắm rõ tình hình điểm danh của mình ngay lập tức, hoặc có thể nhận được sms (tích hợp trong tương lai).
* Có thể phân tích dữ liệu theo thời gian thực.

*Giao diện dễ sử dụng:*

* Giao diện được thiết kế đơn giản, ai cũng có thể sử dụng được.
* Hỗ trợ các tính năng xem lịch sử điểm danh, thông tin cá nhân hoặc xuất dữ liệu ra file (tích hợp trong tương lai).

***Đối tượng sử dụng:*** Hệ thống hướng đến các trường học, các công ty vừa và nhỏ. Nó cũng có thể được tuỳ chỉnh cho các địa điểm làm việc khác như bệnh viện, khách sạn hoặc lại tại các sự kiện lớn.

## 1.3. Mô tả bài toán

### 1.3.1. Mô tả chung

Checking Attendance System là một giải pháp tự động hóa toàn diện quá trình chấm công cho các doanh nghiệp và tổ chức, nhằm tối ưu hóa việc quản lý thời gian vào ra của nhân viên. Hệ thống tích hợp công nghệ nhận xử lý ảnh, thị giác máy tính để phát hiện và xác thực thông tin điểm danh, giúp ghi nhận dữ liệu một cách nhanh chóng và chính xác. Sản phẩm không chỉ dừng lại ở việc ghi nhận thời gian làm việc mà còn cho phép phân tích dữ liệu theo thời gian thực, đưa ra các báo cáo chi tiết giúp doanh nghiệp kiểm soát hiệu quả hoạt động của nguồn nhân lực.

Điểm nổi bật của Checking Attendance System so với các ứng dụng hiện có trên thị trường là sự kết hợp các công cụ xử lý ảnh, học máy và thị giác máy tính hiện đại, đồng thời đáp ứng được lượng yêu cầu xử lý lớn, cung cấp giao dụng người dùng đơn giản và có khả năng tích hợp các thiết bị ngoại vi. Giúp tự động hoá được quy trình chấm công, giảm thiểu sai sót do thao tác thủ công và tạo ra môi trường làm việc hiện đại, linh hoạt.

Với mong muốn cung cấp một công cụ có khả năng cạnh tranh hàng đầu trong ngành, Checking Attendance System đặt mục tiêu vượt qua các sản phẩm truyền thống. Sự thành công của hệ thống phụ thuộc vào việc xây dựng một nền tảng có khả năng mở rộng, tiên phong trong việc sử dụng các công nghệ hiện đại để có thể dự báo xu hướng làm việc, tối ưu hoá hiệu suất, nâng cao trải nghiệm người dùng. Trong tương lai, sản phẩm tiếp tục hướng đến sự đổi mới, cập nhật công nghệ liên tục, nhằm đáp ứng nhu cầu về các giải pháp quản lý, góp phần vào sự phát triển bền vững của doanh nghiệp trong thời kì phát triển của trí tuệ nhân tạo.

### 1.3.2. Input của bài toán

Input của bài toán nhận diện khuôn mặt cho hệ thống điểm danh tự động bao gồm các dữ liệu hình ảnh hoặc video được thu thập từ camera trong thời gian thực hoặc từ cơ sở dữ liệu đã lưu trữ trước đó. Cụ thể, dữ liệu đầu vào bao gồm:

* **Hình ảnh khuôn mặt**: Các hình ảnh chứa khuôn mặt của các cá nhân cần điểm danh, chẳng hạn như sinh viên trong lớp học, nhân viên trong doanh nghiệp, hoặc người tham gia sự kiện. Mỗi hình ảnh có thể bao gồm một hoặc nhiều khuôn mặt, với độ phân giải và chất lượng khác nhau tùy thuộc vào điều kiện chụp (ánh sáng tự nhiên, nhân tạo, hoặc ánh sáng yếu).
* **Đặc điểm đa dạng**: Dữ liệu đầu vào phản ánh sự đa dạng về góc quay (chính diện, nghiêng, xoay), biểu cảm khuôn mặt (cười, nghiêm túc, nhăn mặt), và các yếu tố che chắn (đeo kính, đội mũ, hoặc trang điểm). Điều này đòi hỏi hệ thống phải xử lý các biến thể thực tế để đảm bảo tính tổng quát hóa.
* **Nguồn dữ liệu**: Input được thu thập từ hai nguồn chính: (1) tập dữ liệu công khai như **Labelled Faces in the Wild (LFW)**, cung cấp 13.233 hình ảnh thuộc 5.749 danh tính để huấn luyện và đánh giá mô hình, và (2) dữ liệu nội bộ, thu thập từ các đối tượng mục tiêu (sinh viên, nhân viên) thông qua camera trong các môi trường thực tế như lớp học hoặc nơi làm việc, với số lượng hình ảnh từ 5-10 ảnh mỗi cá nhân ở các góc quay và điều kiện ánh sáng khác nhau.
* **Định dạng dữ liệu**: Hình ảnh được lưu dưới dạng tệp số (thường là JPEG hoặc PNG), được chuẩn hóa về kích thước (ví dụ: 224x224 pixel) và được gán nhãn danh tính (tên, mã định danh) để hỗ trợ quá trình huấn luyện mô hình.

Input được xử lý thời gian thực thông qua camera tích hợp với thiết bị nhúng như Raspberry Pi, nơi các thuật toán như MTCNN được sử dụng để phát hiện và trích xuất khuôn mặt trước khi đưa vào mô hình học sâu. Sự đa dạng và chất lượng của dữ liệu đầu vào là yếu tố quan trọng để đảm bảo hệ thống hoạt động hiệu quả trong các điều kiện môi trường không kiểm soát.

### 1.3.3. Output của bài toán

Output của bài toán là kết quả điểm danh tự động, cung cấp thông tin về sự hiện diện của các cá nhân trong một khoảng thời gian nhất định, được biểu diễn dưới các dạng sau:

* **Danh sách điểm danh**: Một danh sách chứa thông tin về các cá nhân được nhận diện, bao gồm tên, mã định danh, và trạng thái hiện diện (có mặt, vắng mặt). Dữ liệu này được tạo ra dựa trên quá trình so sánh vectơ đặc trưng của khuôn mặt (trích xuất từ FaceNet) với cơ sở dữ liệu khuôn mặt đã đăng ký.
* **Thời gian ghi nhận**: Thời gian chính xác khi mỗi cá nhân được nhận diện, thường được ghi lại dưới dạng dấu thời gian (timestamp) theo định dạng ngày, giờ, phút, giây, để hỗ trợ phân tích và báo cáo.
* **Báo cáo điểm danh**: Dữ liệu điểm danh được lưu trữ tự động vào cơ sở dữ liệu và có thể xuất ra dưới dạng báo cáo chi tiết (ví dụ: bảng Excel, PDF), bao gồm tổng hợp số lượng người có mặt/vắng mặt theo ngày, tuần, hoặc tháng, phục vụ cho việc quản lý và ra quyết định.
* **Thông báo thời gian thực (tương lai)**: Trong các phiên bản nâng cao, hệ thống có thể tích hợp với các nền tảng thông báo như SMS hoặc ứng dụng di động để gửi thông tin điểm danh ngay lập tức đến người quản lý hoặc cá nhân liên quan, dù tính năng này hiện đang được lên kế hoạch phát triển thêm.
* **Độ chính xác và thông tin bổ sung**: Output bao gồm cả mức độ tin cậy của việc nhận diện (confidence score), giúp đánh giá độ tin cậy của kết quả và phát hiện các trường hợp nhận diện không chắc chắn, cần kiểm tra thủ công.

Output của hệ thống không chỉ cung cấp thông tin điểm danh mà còn hỗ trợ các chức năng quản lý dữ liệu, như tra cứu lịch sử điểm danh, phân tích xu hướng vắng mặt, hoặc xuất báo cáo theo yêu cầu. Các kết quả này được hiển thị thông qua giao diện người dùng thân thiện, được thiết kế đơn giản để phù hợp với mọi đối tượng sử dụng, từ giảng viên, quản lý đến nhân viên không chuyên về công nghệ. Việc triển khai trên thiết bị nhúng như Raspberry Pi đảm bảo output được xử lý và hiển thị kịp thời, phù hợp với yêu cầu thực tế của các môi trường như trường học hoặc doanh nghiệp.

# CƠ SỞ LÝ THUYẾT VÀ XỬ LÝ DỮ LIỆU

## 2.1. Cơ sở lý thuyết và thuật toán

### 2.1.1. Tổng quan về học máy

Học máy (Machine Learning) là một phân nhánh quan trọng của trí tuệ nhân tạo (Artificial Intelligence - AI), tập trung vào việc phát triển các thuật toán và mô hình cho phép máy tính học từ dữ liệu và tự cải thiện hiệu suất mà không cần lập trình cụ thể cho từng nhiệm vụ. Trong những thập kỷ gần đây, học máy đã trở thành trụ cột cho nhiều ứng dụng công nghệ tiên tiến, từ nhận diện giọng nói và hình ảnh, xử lý ngôn ngữ tự nhiên, đến các hệ thống khuyến nghị và dự đoán.

Học máy có thể được chia thành ba loại chính: học có giám sát (supervised learning), học không giám sát (unsupervised learning), và học tăng cường (reinforcement learning). Học có giám sát là phương pháp học mà mô hình được huấn luyện trên một tập dữ liệu có gán nhãn. Nhiệm vụ của mô hình là học mối quan hệ giữa đầu vào và đầu ra để dự đoán nhãn cho dữ liệu mới. Ví dụ, trong nhận diện chữ viết tay, mô hình được huấn luyện trên các hình ảnh của chữ cái với nhãn tương ứng. Học không giám sát là phương pháp mà dữ liệu không có nhãn và mô hình phải tự khám phá các cấu trúc hoặc mẫu trong dữ liệu. Một ví dụ phổ biến là phân cụm (clustering), nơi mô hình nhóm các đối tượng tương tự nhau vào cùng một cụm. Học tăng cường là phương pháp học dựa trên việc tương tác với môi trường và nhận phản hồi dưới dạng phần thưởng hoặc hình phạt. Mô hình học cách tối ưu hóa hành động để tối đa hóa phần thưởng tích lũy. Học tăng cường thường được sử dụng trong các ứng dụng như robot và trò chơi.

Hiện nay, học máy đã và đang được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau. Trong y tế, học máy được sử dụng để chẩn đoán bệnh, dự đoán sự bùng phát của dịch bệnh, và cá nhân hóa liệu trình điều trị cho bệnh nhân. Các hệ thống học máy có khả năng phân tích hình ảnh y tế như MRI và X-quang, giúp phát hiện sớm các dấu hiệu bệnh lý mà bác sĩ có thể bỏ sót. Các ngân hàng và công ty tài chính sử dụng học máy để phát hiện gian lận, dự đoán rủi ro, và cung cấp các khuyến nghị đầu tư. Các mô hình học máy có thể phân tích hàng triệu giao dịch trong thời gian thực để phát hiện các mẫu gian lận. Các hệ thống thương mại điện tử ứng dụng học máy để đưa ra các khuyến nghị sản phẩm dựa trên sở thích và hành vi mua sắm của khách hàng, giúp tăng doanh số và cải thiện trải nghiệm người dùng. Amazon và Netflix là những ví dụ điển hình về việc sử dụng học máy để cá nhân hóa trải nghiệm khách hàng. Trong lĩnh vực truyền thông, giải trí, học máy giúp cá nhân hóa nội dung, từ đề xuất phim và bài hát, đến việc tạo ra các nội dung mới dựa trên sở thích của người dùng. Spotify và YouTube sử dụng học máy để gợi ý nhạc và video phù hợp với thị hiếu của từng người dùng.

Mặc dù học máy đã đạt được nhiều thành tựu đáng kể, nhưng vẫn còn nhiều thách thức cần phải đối mặt. Một trong số đó là vấn đề về đạo đức và bảo mật dữ liệu. Việc sử dụng dữ liệu cá nhân trong các mô hình học máy đặt ra câu hỏi về quyền riêng tư và sự minh bạch. Ngoài ra, các mô hình học máy phức tạp thường khó giải thích, gây khó khăn trong việc hiểu và tin tưởng các quyết định của chúng

### 2.1.2. Tổng quan về thị giác máy tính

Thị giác máy tính (Computer Vision) là một lĩnh vực then chốt trong trí tuệ nhân tạo (AI), tập trung vào việc phát triển các thuật toán và hệ thống cho phép máy tính phân tích, xử lý và hiểu thông tin từ hình ảnh hoặc video, mô phỏng khả năng thị giác của con người. Trong thời đại số hóa, khi dữ liệu hình ảnh và video tăng trưởng theo cấp số nhân, thị giác máy tính đã trở thành một công cụ không thể thiếu, thúc đẩy sự đổi mới trong nhiều lĩnh vực như y tế, giao thông, giáo dục, và quản lý doanh nghiệp. Với sự hỗ trợ của học máy (Machine Learning) và học sâu (Deep Learning), thị giác máy tính không chỉ tái hiện khả năng nhìn mà còn vượt qua giới hạn của con người trong việc xử lý thông tin thị giác phức tạp, chẳng hạn như nhận diện hàng nghìn đối tượng trong thời gian thực hoặc phát hiện các chi tiết siêu nhỏ.

Quá trình xử lý trong thị giác máy tính bao gồm nhiều giai đoạn quan trọng, từ thu nhận dữ liệu đến ra quyết định. Ban đầu, hình ảnh hoặc video được thu thập thông qua camera hoặc các thiết bị cảm biến. Sau đó, dữ liệu trải qua tiền xử lý để cải thiện chất lượng, bao gồm các kỹ thuật như giảm nhiễu, chuẩn hóa độ sáng, hoặc tăng cường dữ liệu (data augmentation). Tiếp theo, các đặc trưng quan trọng như đường nét, góc cạnh, hoặc vectơ đặc trưng được trích xuất để mô tả nội dung hình ảnh. Cuối cùng, các thuật toán học máy hoặc học sâu được áp dụng để nhận diện đối tượng, phân loại, hoặc đưa ra dự đoán, tạo ra kết quả phục vụ các ứng dụng thực tế. Trong bối cảnh đề tài **Hệ thống chấm công nhận dạng khuôn mặt bằng mô hình FaceNet**, thị giác máy tính đóng vai trò nền tảng, cho phép hệ thống tự động nhận diện khuôn mặt và ghi nhận điểm danh với độ chính xác cao.

Sự phát triển của thị giác máy tính gắn liền với những tiến bộ trong học sâu, đặc biệt là sự ra đời của các mạng nơ-ron tích chập (Convolutional Neural Networks - CNN). Các mô hình như FaceNet và MTCNN (Multi-task Cascaded Convolutional Networks) đã nâng cao đáng kể hiệu suất của các bài toán nhận diện khuôn mặt. FaceNet, với khả năng chuyển đổi hình ảnh khuôn mặt thành các vectơ đặc trưng (facial embeddings), cho phép so sánh và xác thực danh tính với độ tin cậy cao. Trong khi đó, MTCNN sử dụng mạng nơ-ron tích chập để phát hiện và căn chỉnh khuôn mặt, hoạt động hiệu quả ngay cả trong điều kiện ánh sáng phức tạp hoặc góc quay không lý tưởng. Những công nghệ này không chỉ mang lại hiệu quả vượt trội mà còn mở ra tiềm năng ứng dụng trong các hệ thống thông minh như điểm danh tự động, an ninh, hoặc quản lý nhân sự.

Thị giác máy tính có phạm vi ứng dụng rộng rãi, tác động sâu sắc đến nhiều lĩnh vực của đời sống và công nghiệp. Trong giáo dục, các hệ thống điểm danh tự động giúp tiết kiệm thời gian, tăng tính minh bạch và giảm gian lận. Trong doanh nghiệp, thị giác máy tính hỗ trợ quản lý chấm công, kiểm soát truy cập, hoặc tối ưu hóa quy trình sản xuất thông qua phát hiện lỗi sản phẩm. Trong y tế, công nghệ này được sử dụng để phân tích hình ảnh y khoa, hỗ trợ chẩn đoán bệnh lý như ung thư hoặc bệnh tim. Ngoài ra, thị giác máy tính còn đóng vai trò quan trọng trong giao thông (xe tự hành, nhận diện biển báo), bán lẻ (phân tích hành vi khách hàng), và giải trí (tạo hiệu ứng thực tế ảo). Đối với đề tài này, thị giác máy tính mang lại giải pháp thực tiễn cho bài toán điểm danh, góp phần thúc đẩy chuyển đổi số trong quản lý giáo dục và doanh nghiệp.

Mặc dù đạt được nhiều thành tựu, thị giác máy tính vẫn đối mặt với các thách thức đáng kể. Các vấn đề như nhận diện trong điều kiện ánh sáng yếu, xử lý dữ liệu lớn trong thời gian thực, hoặc đảm bảo tính bảo mật và quyền riêng tư của dữ liệu khuôn mặt đòi hỏi sự cải tiến liên tục. Ngoài ra, việc triển khai các mô hình học sâu trên thiết bị nhúng với tài nguyên hạn chế, như Raspberry Pi, yêu cầu tối ưu hóa hiệu suất và giảm độ phức tạp tính toán. Những thách thức này không chỉ là động lực cho nghiên cứu mà còn mở ra cơ hội để phát triển các giải pháp sáng tạo, như hệ thống điểm danh tự động được đề xuất trong đề tài. Với tiềm năng và những bước tiến không ngừng, thị giác máy tính hứa hẹn sẽ tiếp tục đóng vai trò quan trọng trong việc định hình tương lai của trí tuệ nhân tạo và các ứng dụng thông minh.

### 2.1.3. Công nghệ sử dụng (FaceNet)

#### 2.1.3.1. Khái niệm

FaceNet là một kiến trúc mạng học sâu tiên tiến được phát triển bởi đội ngũ nghiên cứu của Google và giới thiệu vào năm 2015, đánh dấu một bước ngoặt quan trọng trong lĩnh vực nhận diện khuôn mặt dựa trên học sâu (Deep Learning) [Schroff et al., 2015, "FaceNet: A Unified Embedding for Face Recognition and Clustering"]. Khác với các phương pháp truyền thống sử dụng các bộ phân loại trực tiếp (như softmax) để nhận diện khuôn mặt, FaceNet tập trung vào việc học một không gian nhúng (embedding space) với thứ nguyên cố định. Trong không gian này, các vectơ đặc trưng (feature vectors) tương ứng với hình ảnh khuôn mặt của cùng một cá nhân được ánh xạ sao cho có khoảng cách gần nhau, trong khi các vectơ của những cá nhân khác nhau được đặt ở các vị trí xa nhau. Quá trình này được thực hiện thông qua một hàm ánh xạ phi tuyến từ không gian hình ảnh ban đầu sang một không gian vectơ đặc trưng, thường có kích thước 128 hoặc 512 chiều, tùy thuộc vào cấu hình mô hình.

Sự đổi mới của FaceNet nằm ở cách tiếp cận không dựa vào việc huấn luyện một bộ phân loại cố định cho từng danh tính, mà thay vào đó, nó tạo ra một biểu diễn đặc trưng chung (universal embedding) có thể được sử dụng để nhận diện, phân cụm (clustering), hoặc xác thực (verification) khuôn mặt trong các ứng dụng thực tế. Điều này mang lại tính linh hoạt cao, cho phép mô hình thích nghi với các tập dữ liệu mới mà không cần huấn luyện lại toàn bộ mạng, đặc biệt phù hợp với hệ thống điểm danh tự động, nơi cần xử lý một lượng lớn danh tính trong các điều kiện môi trường không kiểm soát.

#### 2.1.3.2. Điểm đặc trưng

FaceNet nổi bật với một số đặc trưng độc đáo, góp phần khẳng định hiệu quả vượt trội của nó trong nhận diện khuôn mặt, bao gồm:

* **Metric Learning**: Thay vì sử dụng các lớp phân loại cuối cùng như trong các mô hình truyền thống, FaceNet áp dụng phương pháp học dựa trên thước đo. Phương pháp này tập trung vào việc tối ưu hóa không gian đặc trưng sao cho khoảng cách giữa các vectơ nhúng phản ánh chính xác mối quan hệ giữa các khuôn mặt. Điều này giúp mô hình học được những đặc trưng phân biệt mà không bị giới hạn bởi số lượng danh tính cụ thể, tạo ra một không gian nhúng có khả năng áp dụng rộng rãi, đặc biệt trong các ứng dụng nhận diện khuôn mặt thời gian thực.
* **Triplet Loss**: Hàm mất mát chính của FaceNet, được gọi là Triplet Loss, đóng vai trò quan trọng trong việc định hình không gian nhúng. Triplet Loss hoạt động bằng cách so sánh ba mẫu hình ảnh – một ảnh gốc, một ảnh của cùng danh tính, và một ảnh của danh tính khác – để đảm bảo rằng khoảng cách giữa các ảnh cùng danh tính nhỏ hơn khoảng cách với ảnh của danh tính khác, với một khoảng cách lề nhất định. Phương pháp này giúp mô hình học được các đặc trưng phân biệt mạnh mẽ, đảm bảo rằng các khuôn mặt cùng một người được nhóm lại gần nhau, trong khi các khuôn mặt khác nhau được tách biệt rõ rệt, phù hợp để xử lý các tập dữ liệu đa dạng như LFW trong hệ thống điểm danh.

Những đặc trưng này không chỉ nâng cao độ chính xác của FaceNet mà còn làm cho nó trở thành một công cụ mạnh mẽ khi triển khai trên các thiết bị nhúng như Raspberry Pi, đặc biệt sau khi được tối ưu hóa để giảm tải tính toán. Trong đề tài này, FaceNet được lựa chọn để trích xuất vectơ đặc trưng từ hình ảnh khuôn mặt, cung cấp nền tảng vững chắc cho việc so sánh và xác thực danh tính trong hệ thống điểm danh tự động.

## 2.2. Thu thập và xử lý dữ liệu

### 2.2.1. Thu thập dữ liệu

Tập dữ liệu **Labelled Faces in the Wild (LFW)** là một trong những bộ dữ liệu công khai tiêu biểu, được thiết kế dành riêng cho các nghiên cứu về nhận diện và xác thực khuôn mặt trong môi trường không kiểm soát. LFW bao gồm **13.233 hình ảnh khuôn mặt** thuộc **5.749 danh tính** (identities), trong đó **1.680 danh tính có từ hai hình ảnh trở lên**, và phần còn lại chỉ xuất hiện trong một hình ảnh duy nhất. Các hình ảnh này được thu thập tự động từ các nguồn trực tuyến như bài báo, trang tin tức, và các trang web công cộng, đảm bảo sự đa dạng về bối cảnh, ánh sáng, góc quay, biểu cảm, và đặc điểm khuôn mặt như đeo kính, đội mũ, hoặc trang điểm.

Quá trình thu thập dữ liệu LFW được thực hiện bằng cách sử dụng các công cụ tìm kiếm hình ảnh trên internet kết hợp với thuật toán phát hiện khuôn mặt để trích xuất các hình ảnh phù hợp. Mỗi hình ảnh được gán nhãn với tên của cá nhân tương ứng, được lưu trữ trong tệp **people.csv**, cung cấp thông tin về danh tính và số lượng hình ảnh liên quan. Tính đa dạng của LFW, với các hình ảnh được chụp trong điều kiện thực tế (không kiểm soát ánh sáng, tư thế, hay môi trường), khiến nó trở thành một nguồn dữ liệu lý tưởng để đánh giá hiệu suất của các mô hình nhận diện khuôn mặt trong các ứng dụng thực tiễn như hệ thống điểm danh tự động.

Trong bối cảnh đề tài **Hệ thống chấm công nhận dạng khuôn mặt bằng mô hình FaceNet**, LFW được sử dụng như một nguồn dữ liệu công khai bổ trợ, kết hợp với dữ liệu nội bộ thu thập từ môi trường thực tế (ví dụ: hình ảnh khuôn mặt của sinh viên hoặc nhân viên trong lớp học hoặc công ty). Dữ liệu nội bộ được thu thập thông qua camera trong các buổi học hoặc ca làm việc, sử dụng các công cụ gán nhãn như CVAT hoặc LabelImg để gắn danh tính cho từng khuôn mặt. Sự kết hợp giữa LFW và dữ liệu nội bộ giúp tăng cường tính đa dạng và khả năng tổng quát hóa của mô hình, đảm bảo hệ thống hoạt động hiệu quả trong các điều kiện thực tế tương tự như môi trường triển khai của hệ thống điểm danh.

### 2.2.2. Phân tích bộ dữ liệu

Tập dữ liệu LFW được thiết kế để mô phỏng các điều kiện thực tế của bài toán nhận diện khuôn mặt, do đó nó mang nhiều đặc điểm nổi bật và thách thức. Với tổng cộng **13.233 hình ảnh**, LFW bao gồm các danh tính với số lượng hình ảnh không đồng đều, từ một hình ảnh duy nhất đến hàng chục hình ảnh cho một số danh tính nổi tiếng. Điều này tạo ra một bài toán nhận diện khuôn mặt không cân bằng (imbalanced dataset), đòi hỏi các kỹ thuật xử lý đặc biệt để đảm bảo mô hình không bị thiên lệch (bias) đối với các danh tính có nhiều hình ảnh hơn. Tệp **people.csv** cung cấp thông tin chi tiết về phân bố danh tính, với mỗi dòng chứa tên cá nhân và số lượng hình ảnh tương ứng, hỗ trợ việc phân tích và tổ chức dữ liệu.

Đặc điểm nổi bật của LFW là sự đa dạng về điều kiện chụp ảnh. Hình ảnh trong LFW được thu thập từ các nguồn công khai, do đó chúng bao gồm các biến thể về:

* **Ánh sáng**: Từ ánh sáng tự nhiên ngoài trời, ánh sáng trong nhà, đến điều kiện ánh sáng yếu.
* **Góc quay và tư thế**: Khuôn mặt có thể xuất hiện ở các góc nghiêng, xoay, hoặc không chính diện.
* **Biểu cảm**: Các biểu cảm đa dạng như cười, nghiêm túc, hoặc nhăn mặt.
* **Yếu tố che chắn**: Một số hình ảnh có người đeo kính, đội mũ, hoặc trang điểm, làm tăng độ phức tạp của bài toán nhận diện.
* **Chất lượng hình ảnh**: Các hình ảnh có độ phân giải và mức độ nén khác nhau, từ chất lượng cao đến hình ảnh bị mờ hoặc nhiễu.

Những đặc tính này khiến LFW trở thành một tập dữ liệu lý tưởng để kiểm tra khả năng tổng quát hóa của các mô hình học sâu trong môi trường thực tế. Trong đề tài, LFW được sử dụng để huấn luyện và đánh giá mô hình nhận diện khuôn mặt, đặc biệt với các thuật toán như FaceNet và MTCNN, nhằm đảm bảo hệ thống điểm danh tự động có thể xử lý các điều kiện không lý tưởng tương tự như trong lớp học hoặc môi trường làm việc. Tuy nhiên, do số lượng hình ảnh giới hạn cho một số danh tính, việc kết hợp với dữ liệu nội bộ là cần thiết để đảm bảo mô hình phù hợp với các đối tượng cụ thể trong ứng dụng điểm danh (ví dụ: sinh viên hoặc nhân viên của một tổ chức).

### 2.2.3. Tiền xử lý dữ liệu

Tiền xử lý dữ liệu là một bước quan trọng trong việc xây dựng hệ thống điểm danh tự động bằng khuôn mặt, đặc biệt khi sử dụng mô hình học sâu như FaceNet. Bộ dữ liệu **Labeled Faces in the Wild (LFW)** được chọn vì tính đa dạng và phù hợp cho các tác vụ nhận diện khuôn mặt trong điều kiện thực tế. Phần tiền xử lý dữ liệu bao gồm các bước như tải dữ liệu, căn chỉnh khuôn mặt, chuẩn hóa, và tổ chức dữ liệu để đảm bảo đầu vào chất lượng cao cho mô hình. Các bước này được thực hiện trên môi trường Google Colab với sự hỗ trợ của thư viện facenet\_pytorch và phương pháp MTCNN.

#### 2.2.3.1. Chuẩn bị môi trường

##### a. Thiết lập Google Colab

Môi trường Google Colab được sử dụng để tận dụng GPU, giúp tăng tốc độ xử lý dữ liệu và huấn luyện mô hình. Các bước thiết lập bao gồm:

* **Kích hoạt GPU**: Vào menu Runtime -> Change runtime type -> Chọn Hardware accelerator là GPU.
* **Kết nối Google Drive**: Mount Google Drive để lưu trữ dữ liệu LFW và kết quả xử lý.
* **Cài đặt thư viện**: Cài đặt các thư viện cần thiết như torch, facenet\_pytorch, và các công cụ hỗ trợ khác.

**Mã lệnh cài đặt**:

# Mount Google Drive

from google.colab import drive

drive.mount('/content/drive')

# Cài đặt thư viện

!pip install torch torchvision

!pip install facenet\_pytorch

!pip install opencv-python matplotlib scikit-image pandas

!pip install tqdm

##### b. Cấu hình và hằng số

Các đường dẫn và tham số được định nghĩa để tổ chức dữ liệu và điều chỉnh quá trình tiền xử lý:

* **Đường dẫn:**
  + LFW\_RAW\_DIR: Thư mục chứa dữ liệu LFW gốc (/content/drive/My Drive/model/ttnt/archive/lfw-deepfunneled).
  + ALIGNED\_LFW\_DIR: Thư mục lưu trữ ảnh LFW đã căn chỉnh (/content/drive/My Drive/Facenet\_Project/lfw\_aligned).
  + MODEL\_SAVE\_PATH: Đường dẫn lưu mô hình đã huấn luyện (/content/drive/My Drive/Facenet\_Project/facenet\_lfw.pth).
* **Tham số:**
  + IMAGE\_SIZE: 160 (kích thước ảnh đầu ra sau căn chỉnh).
  + BATCH\_SIZE: 64 (kích thước batch).
  + TRIPLET\_MARGIN: 0.2 (margin cho Triplet Loss).
  + EMBEDDING\_DIM: 512 (kích thước vector embedding).

#### 2.2.3.2. Tải và giải nén dữ liệu

* **Phương pháp** Dữ liệu LFW được tải từ Kaggle và giải nén vào LFW\_RAW\_DIR. Thư viện facenet\_pytorch hỗ trợ truy cập và xử lý dữ liệu.
* **Lưu ý**: Đảm bảo cấu trúc thư mục phù hợp (root/person\_name/image.jpg) và đủ dung lượng lưu trữ (khoảng 200MB)

#### 2.2.3.3. Các bước tiền xử lý dữ liệu

##### a. Căn chỉnh khuôn mặt (Face Alignment) với MTCNN

***Mục tiêu của việc căn chỉnh khuôn mặt:***

Trước khi sử dụng các ảnh trong tập dữ liệu LFW để huấn luyện mô hình, cần đảm bảo rằng khuôn mặt trong mỗi ảnh được định vị chính xác, với các đặc điểm quan trọng (mắt, mũi, miệng) được căn chỉnh để nằm ở vị trí nhất quán. Điều này rất quan trọng vì các ảnh trong LFW được thu thập từ nhiều nguồn khác nhau trên internet, dẫn đến sự biến thiên lớn về góc quay, ánh sáng, và tư thế. Nếu không căn chỉnh, mô hình có thể gặp khó khăn trong việc học các đặc trưng khuôn mặt một cách hiệu quả, đặc biệt khi sử dụng các mô hình như FaceNet, vốn yêu cầu đầu vào có kích thước và định hướng chuẩn.

***Sử dụng MTCNN (Multi-task Cascaded Convolutional Networks):***

MTCNN là một phương pháp tiên tiến dựa trên mạng nơ-ron tích chập để phát hiện và căn chỉnh khuôn mặt. Phương pháp này hoạt động theo ba giai đoạn liên tiếp:

* **Giai đoạn 1 - Phát hiện sơ bộ (P-Net)**: Sử dụng một mạng nhỏ để xác định các vùng có khả năng chứa khuôn mặt trong ảnh, giảm kích thước ảnh để tăng tốc độ xử lý.
* **Giai đoạn 2 - Tinh chỉnh (R-Net)**: Một mạng sâu hơn được áp dụng để loại bỏ các vùng không phải khuôn mặt và cải thiện độ chính xác của các vùng được phát hiện.
* **Giai đoạn 3 - Định vị chi tiết (O-Net)**: Mạng này thực hiện việc phát hiện các điểm mốc khuôn mặt (landmarks) như góc mắt, mũi, và miệng, từ đó căn chỉnh khuôn mặt sao cho các điểm mốc này nằm ở vị trí cố định.

Quá trình này được thực hiện trên toàn bộ tập ảnh thô trong thư mục LFW\_RAW\_DIR, thường là thư mục lfw-deepfunneled chứa các ảnh đã được căn chỉnh sơ bộ trước đó. Tuy nhiên, để đáp ứng yêu cầu của mô hình nhận diện hiện đại, MTCNN sẽ thực hiện lại quá trình căn chỉnh để đảm bảo chất lượng cao hơn, với kích thước đầu ra được đặt là 160x160 pixel.

***Quy trình thực hiện:***

* **Đếm số lượng ảnh**: Trước tiên, hệ thống sẽ quét toàn bộ thư mục chứa ảnh thô để đếm tổng số lượng ảnh cần xử lý. Điều này giúp người dùng có cái nhìn tổng quan về khối lượng công việc và theo dõi tiến độ.
* **Xử lý từng người**: Mỗi thư mục con trong LFW\_RAW\_DIR đại diện cho một danh tính (ví dụ: "George\_W\_Bush" hoặc "Angelina\_Jolie"), và bên trong đó chứa nhiều ảnh của cùng một người. Hệ thống sẽ duyệt qua từng thư mục này.
* **Xử lý từng ảnh**: Mỗi ảnh được mở dưới dạng định dạng RGB (Red, Green, Blue) để đảm bảo tính nhất quán về màu sắc. Sau đó, MTCNN được áp dụng để phát hiện khuôn mặt. Nếu phát hiện thành công, khuôn mặt được trích xuất dưới dạng tensor, chuyển đổi thành ảnh PIL (Python Imaging Library), và lưu vào thư mục tương ứng trong ALIGNED\_LFW\_DIR với cùng tên file gốc.
* **Xử lý lỗi**: Trong trường hợp không phát hiện được khuôn mặt (do ánh sáng kém, góc quay bất lợi, hoặc chất lượng ảnh thấp), hệ thống sẽ ghi lại thông tin lỗi vào một file log riêng biệt, giúp người dùng dễ dàng theo dõi và khắc phục sau này. Tương tự, nếu xảy ra lỗi kỹ thuật (ví dụ: file bị hỏng), lỗi cũng được ghi lại chi tiết để phân tích.
* **Theo dõi tiến trình**: Một thanh tiến trình được hiển thị để người dùng có thể theo dõi thời gian thực hiện, đặc biệt hữu ích khi xử lý hàng nghìn ảnh trong tập dữ liệu LFW.

***Kết quả****:*

Sau khi hoàn tất, số lượng khuôn mặt đã căn chỉnh thành công sẽ được báo cáo. Điều này cho phép người dùng đánh giá hiệu quả của quá trình căn chỉnh và quyết định có cần điều chỉnh tham số MTCNN (như ngưỡng phát hiện hoặc kích thước tối thiểu khuôn mặt) hay không.

##### b. Chuẩn bị Dataset và DataLoader

***Mục đích:***

Sau khi có tập ảnh đã căn chỉnh, bước tiếp theo là chuẩn bị dữ liệu theo cách phù hợp để huấn luyện mô hình nhận diện khuôn mặt. Phương pháp triplet loss yêu cầu mỗi mẫu dữ liệu bao gồm ba ảnh: một ảnh neo (anchor), một ảnh dương tính (positive - cùng người với anchor), và một ảnh âm tính (negative - người khác). Đồng thời, dữ liệu cần được chia thành các tập train, validation, và test để đánh giá hiệu suất mô hình.

***Biến đổi dữ liệu (Augmentation):***

Để tăng cường khả năng tổng quát hóa của mô hình, các biến đổi ngẫu nhiên được áp dụng lên ảnh đã căn chỉnh. Cụ thể:

* **Lật ngang ngẫu nhiên**: Với xác suất 50%, ảnh sẽ được lật ngang để mô phỏng các góc nhìn khác nhau.
* **Thay đổi độ sáng và độ tương phản**: Mức độ thay đổi được giới hạn trong 20% để duy trì đặc trưng khuôn mặt nhưng vẫn tạo sự đa dạng.
* **Xoay ngẫu nhiên**: Ảnh có thể xoay tối đa 10 độ để mô phỏng các góc quay nhẹ.
* **Chuẩn hóa**: Giá trị pixel được chuẩn hóa về khoảng [-1, 1] với giá trị trung bình 0.5 và độ lệch chuẩn 0.5, phù hợp với đầu vào của nhiều mô hình học sâu.

Những biến đổi này giúp mô hình học được các đặc trưng khuôn mặt trong nhiều điều kiện khác nhau, từ đó cải thiện độ chính xác khi áp dụng trên dữ liệu thực tế.

##### c. Tạo Dataset triplet:

Một lớp tùy chỉnh tên là TripletFaceDataset được thiết kế để tạo các triplet từ tập ảnh đã căn chỉnh. Quá trình này bao gồm:

* **Phân loại danh tính**: Xác định các danh tính (người) có ít nhất hai ảnh, vì triplet yêu cầu một anchor và một positive từ cùng danh tính.
* **Chọn mẫu**: Ngẫu nhiên chọn một anchor và một positive từ cùng danh tính, sau đó chọn một negative từ một danh tính khác. Nếu không đủ danh tính khác, negative có thể được chọn từ cùng danh tính nhưng ảnh khác.
* **Kiểm tra hợp lệ**: Nếu không có danh tính nào có ít nhất hai ảnh, hệ thống sẽ báo lỗi để người dùng kiểm tra lại dữ liệu.

##### **d**. Chia tập dữ liệu:

Tập dữ liệu triplet được chia thành ba phần:

* **Train set (70%)**: Dùng để huấn luyện mô hình.
* **Validation set (15%)**: Dùng để điều chỉnh siêu tham số và kiểm tra hiệu suất trong quá trình huấn luyện.
* **Test set (15%)**: Dùng để đánh giá cuối cùng sau khi huấn luyện.

Phân chia này được thực hiện ngẫu nhiên để đảm bảo tính đại diện của các danh tính trong từng tập.

##### e. Cấu hình DataLoader:

DataLoader được sử dụng để tải dữ liệu theo batch, với các thông số:

* **Batch size**: 64 ảnh cho mỗi batch, giúp tối ưu hóa việc sử dụng bộ nhớ GPU.
* **Shuffle**: Áp dụng cho tập train để tăng tính ngẫu nhiên, trong khi validation và test không shuffle để giữ thứ tự kiểm tra nhất quán.
* **Num workers**: Sử dụng 2 worker để tải dữ liệu song song, tăng tốc độ trên các hệ thống đa lõi.
* **Pin memory**: Bật chế độ này để tăng tốc độ truyền dữ liệu từ CPU sang GPU.

***Kết quả:***

Sau khi hoàn tất, hệ thống sẽ báo cáo số lượng batch trong mỗi epoch cho từng tập (train, validation, test), giúp người dùng hiểu rõ khối lượng dữ liệu và chuẩn bị cho quá trình huấn luyện.

##### Ý nghĩa và lợi ích

* **Chất lượng dữ liệu**: Việc căn chỉnh khuôn mặt đảm bảo rằng mô hình tập trung vào các đặc trưng quan trọng, giảm nhiễu từ góc quay hoặc ánh sáng.
* **Tăng cường dữ liệu**: Các biến đổi ngẫu nhiên giúp mô hình học được các biến thể tự nhiên, cải thiện khả năng tổng quát hóa.
* **Hiệu quả huấn luyện**: Việc sử dụng triplet loss với DataLoader tối ưu hóa cho phép mô hình học cách phân biệt khuôn mặt một cách hiệu quả, đặc biệt trong các ứng dụng thực tế như nhận diện khuôn mặt trong điều kiện không kiểm soát.

# PHÁT TRIỂN MÔ HÌNH

## 3.1. Phát triển mô hình

Trong khuôn khổ dự án này, việc phát triển một hệ thống nhận diện khuôn mặt dựa trên phương pháp FaceNet đòi hỏi sự lựa chọn một kiến trúc mạng nơ-ron tiên tiến, và chúng tôi đã ưu tiên sử dụng **InceptionResnetV1** – một mô hình được đề xuất trong bài báo gốc của FaceNet (Schroff et al., 2015). Kiến trúc này không chỉ nổi bật về khả năng trích xuất các đặc trưng phân biệt từ hình ảnh khuôn mặt mà còn là nền tảng vững chắc để tối ưu hóa hiệu suất nhận diện thông qua chiến lược huấn luyện trước (pre-training) và tinh chỉnh (fine-tuning) trên tập dữ liệu **Labeled Faces in the Wild (LFW)**.

### 3.1.1. Kiến trúc InceptionResnetV1

InceptionResnetV1 là sự kết hợp tinh tế giữa hai khái niệm kiến trúc mạng nơ-ron sâu tiên tiến, mang lại hiệu quả vượt trội trong xử lý hình ảnh phức tạp như nhận diện khuôn mặt:

* **Inception Modules**: Các mô-đun Inception cho phép mạng học các đặc trưng ở nhiều cấp độ trừu tượng và quy mô khác nhau thông qua việc triển khai đồng thời các lớp tích chập (convolutional layers) với các kích thước kernel đa dạng (1x1, 3x3, 5x5) cùng với phép toán pooling. Cơ chế này không chỉ tăng cường khả năng thu thập thông tin phong phú từ dữ liệu hình ảnh mà còn tối ưu hóa việc phát hiện các đặc trưng cục bộ và toàn cục của khuôn mặt, chẳng hạn như hình dạng mắt, mũi, hoặc cấu trúc tổng thể.
* **Residual Connections (từ ResNet)**: Việc tích hợp các kết nối dư (skip connections) hoặc kết nối tắt (shortcut connections) giúp gradient truyền trực tiếp qua nhiều lớp mạng, khắc phục vấn đề biến mất gradient (gradient vanishing) thường gặp trong các mạng sâu. Điều này không chỉ đảm bảo khả năng hội tụ của mô hình trong quá trình huấn luyện mà còn nâng cao hiệu quả khi xử lý các kiến trúc có độ sâu lớn.

Cấu trúc tổng thể của InceptionResnetV1 bao gồm một chuỗi các khối Inception-ResNet được sắp xếp chồng lên nhau, dần dần giảm kích thước không gian của bản đồ đặc trưng (feature maps) đồng thời tăng chiều sâu (số lượng kênh). Sau các lớp tích chập và pooling, một phép pooling trung bình toàn cục (Global Average Pooling) được áp dụng để tổng hợp các đặc trưng không gian thành một vector đặc trưng duy nhất. Vector này sau đó được truyền qua một lớp dense (fully connected layer) để tạo ra vector embedding khuôn mặt có kích thước cố định, chẳng hạn 512 chiều, phục vụ cho nhiệm vụ nhận diện.

### 3.1.2. Sử dụng mô hình huấn luyện trước (Pre-trained Model)

Thay vì bắt đầu quá trình huấn luyện từ các trọng số ngẫu nhiên, chúng tôi đã khai thác lợi thế của một mô hình **pre-trained**, được huấn luyện trước trên tập dữ liệu **VGGFace2** – một tập hợp dữ liệu khổng lồ chứa hàng triệu hình ảnh của hàng nghìn danh tính khác nhau. Phương pháp này mang lại những lợi ích quan trọng sau:

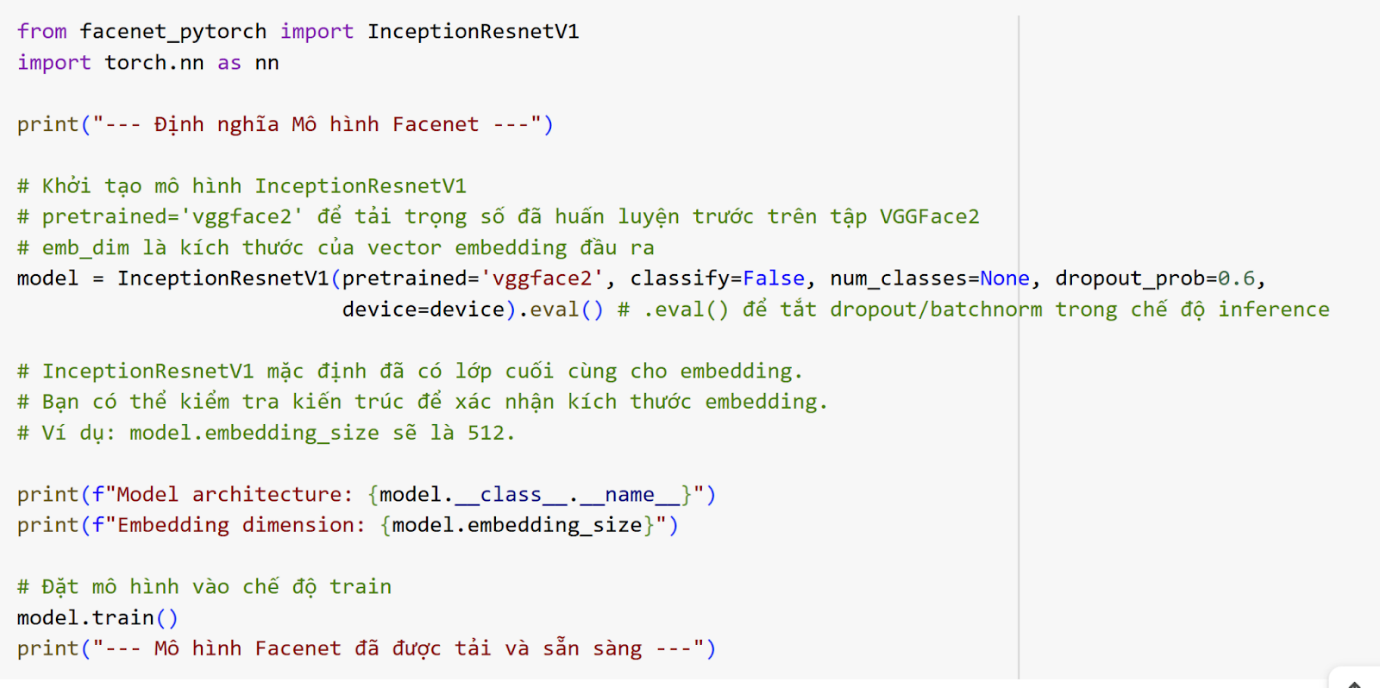
* **Học các đặc trưng tổng quát**: Mô hình pre-trained trên VGGFace2 đã tiếp thu các đặc trưng cấp thấp và trung bình phổ quát liên quan đến khuôn mặt, như hình dạng các bộ phận (mắt, mũi, miệng) và kết cấu da. Những đặc trưng này tạo nền tảng vững chắc, có thể được áp dụng linh hoạt cho nhiều tác vụ nhận diện khuôn mặt khác nhau.
* **Tăng tốc độ hội tụ**: Khởi đầu từ các trọng số đã được tối ưu hóa giúp mô hình hội tụ nhanh chóng hơn trong quá trình tinh chỉnh trên tập LFW, giảm đáng kể thời gian huấn luyện so với việc khởi tạo ngẫu nhiên.
* **Nâng cao hiệu suất**: Ngay cả với tập dữ liệu nhỏ hơn như LFW, việc tinh chỉnh một mô hình pre-trained thường mang lại độ chính xác cao hơn so với huấn luyện từ đầu. Điều này đặc biệt quan trọng trong bối cảnh LFW chủ yếu được sử dụng để đánh giá thay vì huấn luyện hoàn toàn.
* **Ngăn ngừa overfitting**: Với các tập dữ liệu hạn chế, huấn luyện từ đầu dễ dẫn đến hiện tượng quá khớp (overfitting). Sử dụng mô hình pre-trained giúp giảm thiểu rủi ro này bằng cách cung cấp các đặc trưng đã được tổng quát hóa, từ đó tăng cường khả năng khái quát hóa của mô hình.

### 3.1.3. Triển khai với thư viện facenet\_pytorch

Để triển khai hiệu quả, chúng tôi sử dụng thư viện **facenet\_pytorch**, cung cấp một triển khai sẵn có của InceptionResnetV1 với khả năng tích hợp trọng số pre-trained. Cụ thể:

Dòng code ***model = InceptionResnetV1(pretrained='vggface2', classify=False, num\_classes=None, dropout\_prob=0.6, device=device).eval()*** thực hiện:

* **InceptionResnetV1**: Khởi tạo kiến trúc mạng.
* **pretrained='vggface2'**: Tải trọng số đã được huấn luyện trên VGGFace2.
* **classify=False**: Loại bỏ lớp phân loại cuối cùng, đảm bảo mô hình chỉ trả về vector embedding thay vì dự đoán lớp.
* **num\_classes=None**: Không cần định nghĩa số lớp đầu ra khi classify=False.
* **dropout\_prob=0.6**: Áp dụng tỷ lệ dropout trong các lớp dense để giảm overfitting.
* **device=device**: Di chuyển mô hình lên GPU (nếu có) để tăng tốc độ tính toán.
* **.eval()**: Ban đầu đặt mô hình ở chế độ đánh giá, tắt các lớp như Dropout và BatchNorm để phục vụ inference. Sau đó, .train() được gọi để chuyển sang chế độ huấn luyện cho quá trình tinh chỉnh.



Hình 3.1. Triển khai mô hình FaceNet.

### 3.1.4. Vai trò của mô hình trong FaceNet

Trong hệ thống FaceNet, mô hình InceptionResnetV1 đóng vai trò quan trọng như một hàm ánh xạ (f(\cdot)) trong công thức Triplet Loss, được định nghĩa như sau:

Trong đó:

* : Hình ảnh Anchor.
* Hình ảnh Positive (cùng người với Anchor).
* : Hình ảnh Negative (khác người với Anchor).
* α: Margin (khoảng cách ngưỡng, ví dụ: 0.2).

Mô hình InceptionResnetV1 (đã được tải và tinh chỉnh) đóng vai trò là hàm f(⋅) để chuyển đổi các hình ảnh đầu vào (​,​,​) thành các vector embedding tương ứng. Mục tiêu của quá trình huấn luyện là tinh chỉnh các trọng số của mô hình f(⋅) để giảm thiểu giá trị L, nghĩa là khoảng cách giữa embedding của Anchor và Positive phải nhỏ hơn đáng kể so với khoảng cách giữa Anchor và Negative , với một khoảng cách lề α nhất định. Điều này đảm bảo khả năng phân biệt cao giữa các danh tính khác nhau, là nền tảng cho việc nhận diện và xác minh khuôn mặt..

## 3.2. Huấn luyện mô hình

Quá trình huấn luyện là giai đoạn then chốt nhằm tối ưu hóa các trọng số của mô hình Facenet, cho phép nó học cách trích xuất các đặc trưng khuôn mặt có khả năng phân biệt cao. Mục tiêu chính là để mô hình có thể tạo ra các vector embedding (nhúng) mà ở đó, khuôn mặt của cùng một người có khoảng cách gần nhau, trong khi khuôn mặt của những người khác nhau lại có khoảng cách xa hơn đáng kể.

### 3.2.1. Khởi tạo Bộ Tối Ưu Hóa và Điều Chỉnh Tốc Độ Học Tập

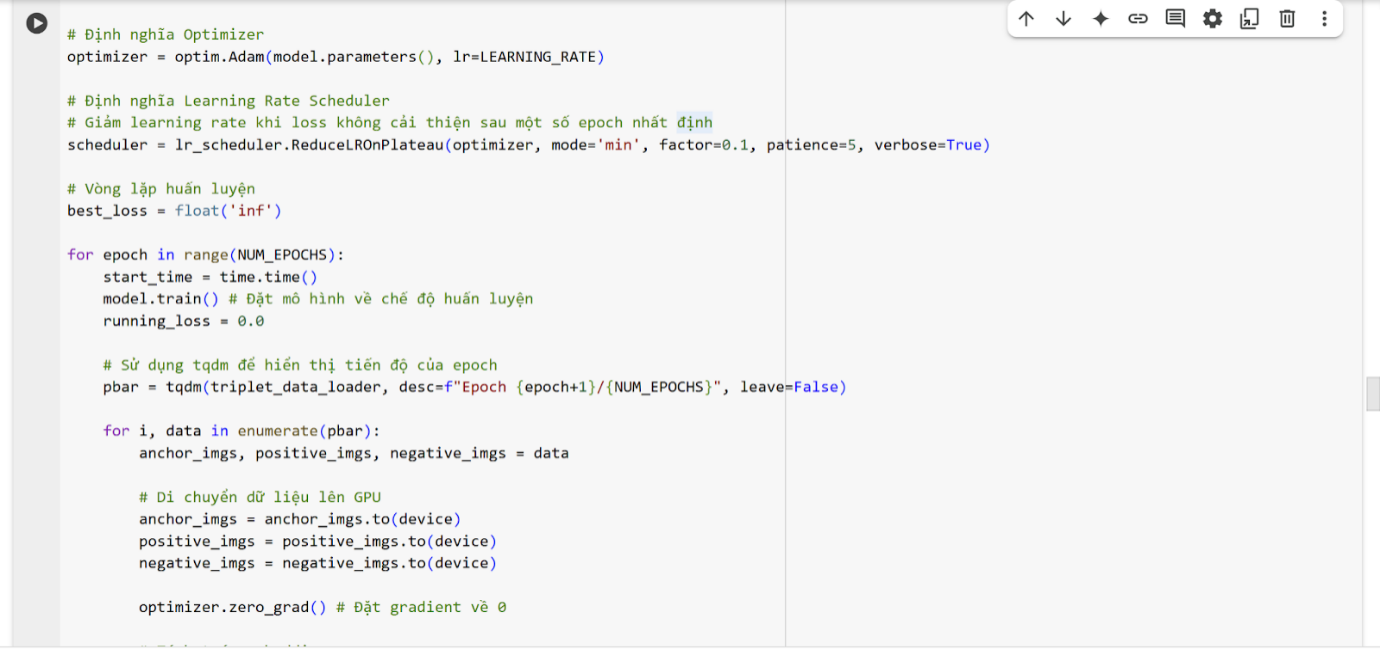
Để dẫn dắt mô hình hội tụ hiệu quả, chúng tôi đã cấu hình một bộ tối ưu hóa (optimizer) và một bộ điều chỉnh tốc độ học tập (learning rate scheduler).

* **Bộ Tối Ưu Hóa Adam:** Chúng tôi lựa chọn thuật toán **Adam (Adaptive Moment Estimation)** làm bộ tối ưu hóa chính. Adam nổi bật nhờ khả năng điều chỉnh tốc độ học tập một cách linh hoạt cho từng tham số của mô hình. Điều này được thực hiện bằng cách ước lượng các moment bậc nhất và bậc hai của gradient, giúp quá trình tối ưu diễn ra nhanh chóng và ổn định, đặc biệt hiệu quả với các mô hình mạng nơ-ron phức tạp. Tốc độ học tập ban đầu được đặt là 0.001, một giá trị phổ biến và hiệu quả cho thuật toán Adam.
* **Bộ Điều Chỉnh Tốc Độ Học Tập ReduceLROnPlateau:** Nhằm cải thiện khả năng hội tụ và tránh mắc kẹt ở các cực tiểu cục bộ, chúng tôi sử dụng torch.optim.lr\_scheduler.LambdaLR với ***lr\_lambda=lambda epoch: min(1.0, (epoch + 1) / 3).*** Bộ điều chỉnh này tăng tốc độ học tập tuyến tính trong 3 epoch đầu tiên, sau đó giữ nguyên. Cơ chế này cho phép mô hình tìm kiếm các giải pháp tối ưu hơn trong không gian tham số khi nó tiến gần đến điểm hội tụ

### 3.2.2. Vòng Lặp Huấn Luyện Chính

Quá trình huấn luyện được thực hiện qua **20 epoch** , với mỗi epoch bao gồm việc xử lý toàn bộ tập dữ liệu huấn luyện theo từng batch.

* **Chuẩn bị Epoch:** Vào đầu mỗi epoch, mô hình được đặt vào chế độ huấn luyện (model.train()) để kích hoạt các cơ chế như Dropout và cập nhật thống kê của Batch Normalization. Một biến running\_loss được khởi tạo để theo dõi tổng hàm mất mát trong suốt epoch, và tqdm được sử dụng để cung thị thanh tiến độ trực quan, giúp giám sát quá trình.
* **Xử lý Batch Dữ liệu:** Trong mỗi lần lặp của epoch, một batch dữ liệu triplet (Anchor, Positive, Negative) được lấy từ triplet\_data\_loader. Các tensor ảnh này ngay lập tức được chuyển lên GPU (.to(device)) để tận dụng khả năng tính toán song song, tăng tốc đáng kể quá trình xử lý.
* **Tính Toán Gradient và Cập Nhật Trọng Số:**
* **Xóa Gradient Cũ:** Trước khi tính toán gradient cho batch hiện tại, optimizer.zero\_grad() được gọi để đảm bảo rằng các gradient tích lũy từ các batch trước đó được đặt về 0. Điều này ngăn chặn việc gradient bị cộng dồn không mong muốn.
* **Trích Xuất Embeddings:** Các ảnh Anchor, Positive và Negative của batch được đưa qua mô hình Facenet (model()) để tạo ra các vector embedding tương ứng.
* **Tính Triplet Loss**: Giá trị của hàm Triplet Loss được tính toán dựa trên ba vector embedding này, với tham số TRIPLET\_MARGIN (0.2) đảm bảo có một khoảng cách tối thiểu giữa cặp âm và cặp dương.
* **Lan Truyền Ngược (Backpropagation):** loss.backward() thực hiện quá trình lan truyền ngược, tự động tính toán gradient của hàm mất mát đối với tất cả các trọng số của mô hình.
* **Cập Nhật Trọng Số:** Cuối cùng, optimizer.step() sử dụng các gradient đã tính toán để cập nhật trọng số của mô hình, di chuyển nó theo hướng giảm thiểu hàm mất mát.
* **Giám sát và Lưu Mô Hình:**
* Sau mỗi batch, giá trị loss hiện tại được cập nhật vào running\_loss và hiển thị trên thanh tiến độ.
* Khi một epoch kết thúc, loss trung bình (epoch\_loss) được tính toán.
* **Điều chỉnh tốc độ học:** scheduler.step(epoch\_loss) được gọi để điều chỉnh tốc độ học dựa trên loss trung bình của epoch.
* **Lưu mô hình tốt nhất:** Hệ thống theo dõi best\_loss đạt được. Nếu epoch\_loss hiện tại tốt hơn best\_loss đã ghi nhận, các trọng số của mô hình sẽ được lưu lại vào file (facenet\_lfw.pth). Điều này đảm bảo rằng chúng tôi luôn giữ lại phiên bản mô hình có hiệu suất tốt nhất trong suốt quá trình huấn luyện, phòng trường hợp loss bắt đầu tăng trở lại ở các epoch sau.
* Quá trình lặp đi lặp lại qua nhiều epoch cho phép mô hình dần dần học và tinh chỉnh khả năng tạo ra các embedding phân biệt hiệu quả cho nhiệm vụ nhận diện và xác minh khuôn mặt.



Hình 3.2. Huấn luyện mô hình.

# ĐÁNH GIÁ VÀ TRIỂN KHAI HỆ THỐNG

## 4.1. Đánh giá mô hình

Để đánh giá chất lượng của mô hình nhận diện khuôn mặt được huấn luyện, nhóm nghiên cứu sử dụng các chỉ số phổ biến trong lĩnh vực học máy và sinh trắc học, bao gồm: AUC (Area Under Curve), F1 Score, EER (Equal Error Rate), và các chỉ số phân loại chi tiết như precision, recall, và accuracy. Những chỉ số này giúp phân tích hiệu suất mô hình một cách toàn diện cả về độ chính xác tổng thể và khả năng phân biệt giữa các lớp.

***Diện tích dưới đường cong ROC (AUC):***

Chỉ số AUC (Area Under the Receiver Operating Characteristic Curve) là một trong những tiêu chuẩn phổ biến để đo lường khả năng phân biệt giữa hai lớp nhãn: “Same” (cùng người) và “Different” (khác người).

 **True Positive Rate (TPR)**: hay còn gọi là Recall.

 **False Positive Rate (FPR)**: tỷ lệ báo nhầm ảnh khác người là cùng người.

AUC biểu thị xác suất mà mô hình có thể phân biệt chính xác một cặp ảnh cùng người và một cặp ảnh khác người. Đường ROC được xây dựng dựa trên mối quan hệ giữa:

***Đánh giá F1 Score tối đa***

**F1 Score** là chỉ số tổng hợp giữa **Precision** (độ chính xác) và **Recall** (khả năng bao phủ), được tính theo công thức:

 Precision cao giúp hạn chế việc chấm công sai cho người không hợp lệ.

 Recall cao giúp đảm bảo người đúng không bị bỏ sót.

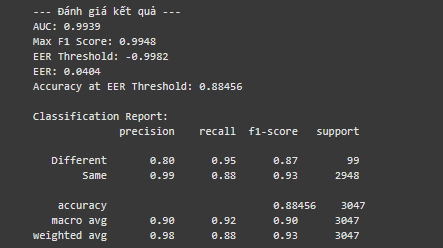
***Ngưỡng và chỉ số Equal Error Rate (EER)***

Equal Error Rate (EER) là một chỉ số phổ biến trong hệ thống xác thực sinh trắc học. Đây là ngưỡng mà tại đó tỷ lệ False Acceptance Rate (FAR) và False Rejection Rate (FRR) bằng nhau.

 **FAR**: xác suất chấp nhận nhầm người lạ.

 **FRR**: xác suất từ chối nhầm người thật.

EER càng thấp → hệ thống càng cân bằng giữa bảo mật và trải nghiệm người dùng.



|  |  |  |
| --- | --- | --- |
| **Chỉ số** | **Giá trị** | **Nhận xét** |
| AUC | 0.9939 | Mô hình phân biệt rất tốt giữa ảnh “Same” và “Different”. |
| Max F1 Score | 0.9948 | Mô hình cân bằng rất tốt giữa precision và recall |
| EER Threshold | -0.9982 | Ngưỡng điểm cân bằng sai số FAR = FRR (có thể dùng để thiết lập ngưỡng thực tế). |
| EER (Equal Error Rate) | 0.0040 | Tỷ lệ chấp nhận sai và từ chối sai gần như không đáng kể. |
| Accuracy tại ngưỡng EER | 0.88456 | Tương đối cao, đặc biệt vì đây là tại ngưỡng cân bằng rủi ro bảo mật. |

Bảng chỉ số tổng quát

Bên cạnh các chỉ số tổng quát như AUC, F1 Score, EER hay Accuracy tại ngưỡng cân bằng, bảng phân loại (classification report) cung cấp cái nhìn chi tiết hơn về khả năng nhận diện của mô hình trên từng lớp, từ đó giúp đánh giá hiệu quả vận hành trong môi trường thực tế.

Đối với lớp **“Different”**, tức là những cặp ảnh không cùng một người, mô hình đạt precision là 0.80. Điều này có nghĩa là trong số các trường hợp được dự đoán là khác người, có 80% là chính xác, trong khi 20% còn lại là các trường hợp nhầm lẫn, có thể dẫn đến việc từ chối điểm danh cho người đúng. Tuy nhiên, với recall lên tới 0.95, mô hình cho thấy khả năng phát hiện rất tốt các trường hợp giả mạo, tức là phần lớn các ảnh không cùng người đều được mô hình nhận diện đúng. F1-score của lớp này là 0.87, cho thấy sự cân bằng khá tốt giữa độ chính xác và khả năng bao phủ. Mặc dù lớp “Different” chỉ chiếm một tỷ lệ rất nhỏ trong tập dữ liệu (99 mẫu so với 2948 mẫu “Same”), mô hình vẫn duy trì được hiệu suất ổn định, cho thấy tính chất tổng quát tốt.

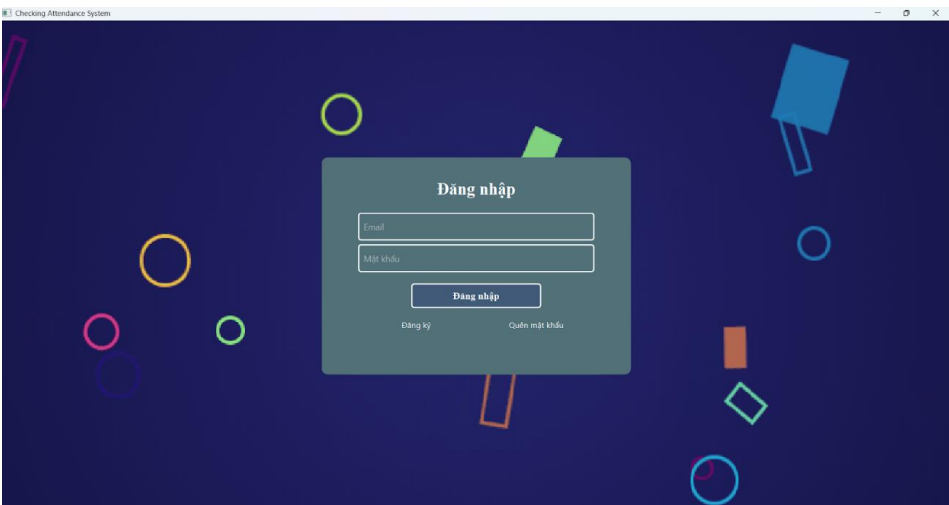
Trong khi đó, với lớp **“Same”**, là những cặp ảnh của cùng một người, mô hình đạt precision rất cao, lên tới 0.99. Điều này phản ánh rằng gần như tất cả các trường hợp được mô hình xác định là cùng người đều là chính xác, giúp hạn chế tối đa khả năng chấm công sai cho người không hợp lệ. Tuy vậy, recall của lớp này chỉ đạt 0.88, tức là khoảng 12% số ảnh cùng người bị mô hình từ chối nhận diện. Điều này có thể gây ảnh hưởng đến trải nghiệm người dùng, khi một số nhân sự thực sự hợp lệ không được xác nhận trong quá trình điểm danh do yếu tố ngoại cảnh như thay đổi ánh sáng, góc chụp hoặc chất lượng ảnh. F1-score của lớp này đạt 0.93, cho thấy mô hình vẫn duy trì được hiệu năng cao trong nhận diện người hợp lệ.

Về tổng thể, độ chính xác (accuracy) của mô hình tại ngưỡng cân bằng đạt 88.456%, một con số đáng kể trong bối cảnh dữ liệu không cân bằng. Chỉ số trung bình theo macro (macro average) cho thấy mô hình đạt precision là 0.90, recall là 0.92 và F1-score là 0.90. Điều này cho thấy hiệu suất giữa hai lớp được duy trì khá cân bằng. Trong khi đó, chỉ số trung bình có trọng số (weighted average), vốn tính đến tỷ lệ xuất hiện của mỗi lớp, đạt precision là 0.98, recall là 0.88 và F1-score là 0.93. Sự chênh lệch này phản ánh việc lớp “Same” chiếm đa số, ảnh hưởng đáng kể đến chỉ số trung bình tổng thể.

Kết quả trên cho thấy mô hình nhận diện khuôn mặt đang được triển khai không chỉ đạt hiệu suất cao về khả năng phân loại chính xác, mà còn thể hiện tính ổn định và đáng tin cậy khi áp dụng trong hệ thống chấm công thông minh. Mặc dù recall của lớp “Same” còn có thể được cải thiện để giảm thiểu hiện tượng bỏ sót người hợp lệ, song với AUC đạt 0.9939 và F1-score lên tới 0.9948, hệ thống đã chứng minh được năng lực vượt trội trong việc phân biệt giữa người dùng hợp lệ và các trường hợp giả mạo. EER chỉ ở mức 0.0040 tiếp tục khẳng định tính bảo mật cao, đồng thời duy trì được sự cân bằng hợp lý giữa tỷ lệ chấp nhận sai và từ chối sai. Đây là một yếu tố then chốt giúp đảm bảo hệ thống có thể hoạt động hiệu quả và ổn định trong môi trường thực tế, nơi yêu cầu vừa đảm bảo chính xác, vừa đảm bảo trải nghiệm người dùng.

## 4.2. Triển khai hệ thống

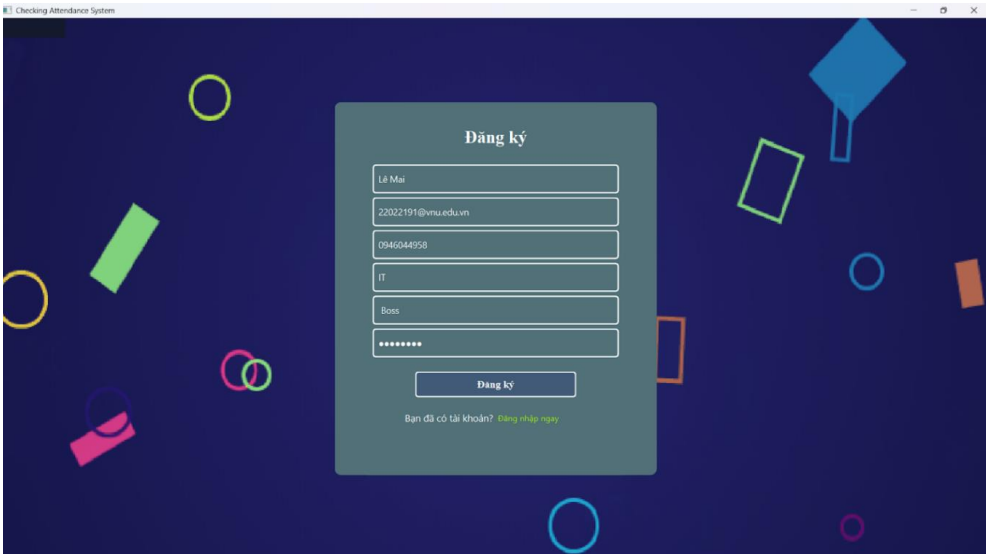
Trình bày các giao diện thực tế mà nhóm đã phát triển qua các hình ảnh giao diện của hệ thống và kèm theo mô tả chi tiết.



Hình 4.1. Giao diện đăng nhập.

Đây là giao diện của người quản lý dùng để đăng nhập tài khoản quản trị. Giao diện gồm có:

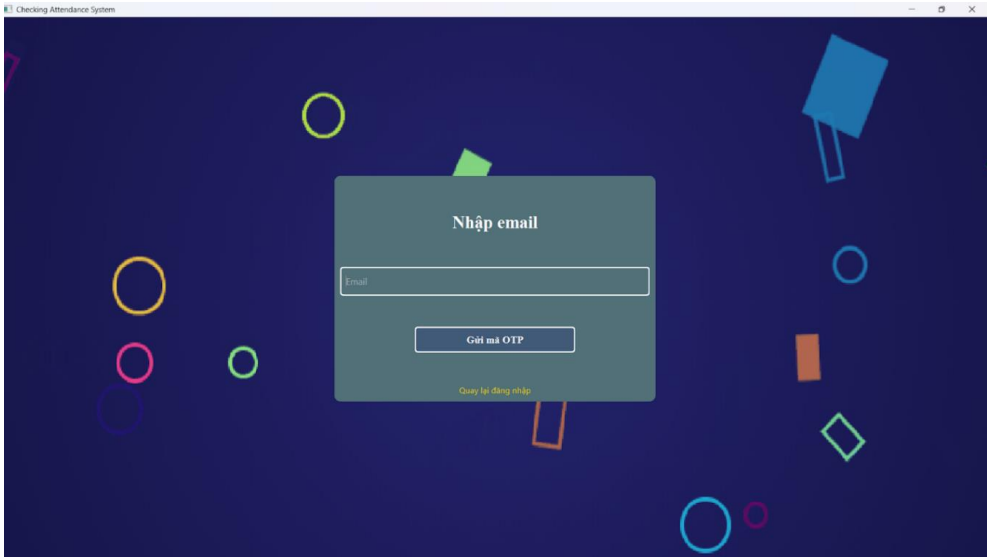
* Các ô nhập liệu: Các ô được dùng để nhập tài khoản và mật khẩu.
* Các đường liên kết đến giao diện đăng ký tài khoản và giao diện quên mật khẩu. Khi nhập đúng thông tin đăng nhập, giao diện sẽ được chuyển sang giao diện quản lý



Hình 4.2. Giao diện đăng ký.

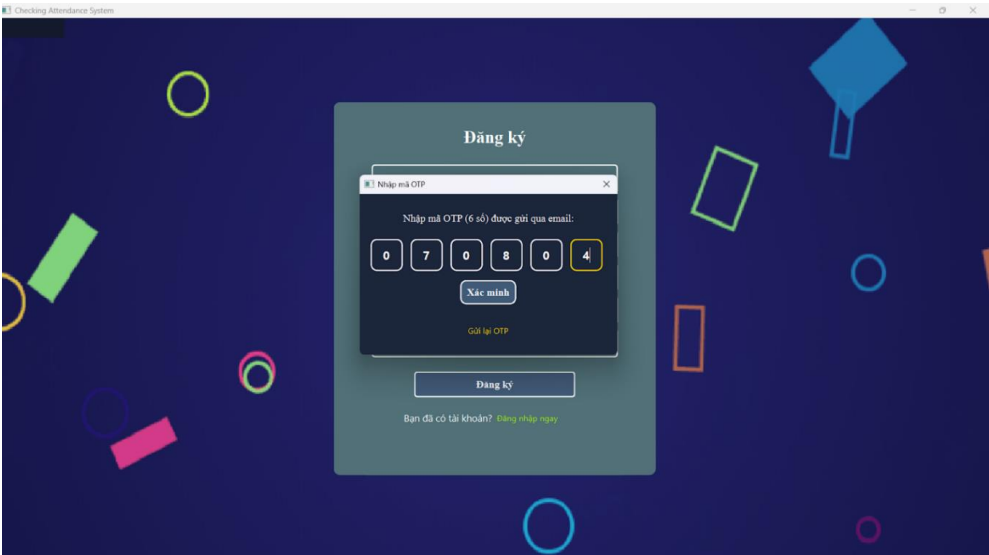
Ở giao diện này, người dùng thực hiện đăng ký một tài khoản mới để quản lý nhân viên (đã được cấp quyền), người dùng nhập các thông tin cần thiết bao gồm Họ tên, địa chỉ Email (Đúng định dạng email), Số điện thoại (Đúng định dạng số điện thoại), phòng ban, chức vụ và mật khẩu, trong đó mật khẩu phải điền đúng định dạng yêu cầu (có ít nhất 8 ký tự, bao gồm ít nhất một ký tự chữ số, một ký tự in hoa và một ký tự đặc biệt như “@#$%&”). Khi bấm nút đăng ký, một mã OTP sẽ được gửi đến email đã được nhập để yêu cầu xác thực tài khoản vừa đăng ký.

Nếu như đã có tài khoản quản lý trước đó, chuyển sang giao diện Đăng nhập bằng nút “Đăng nhập ngay”.



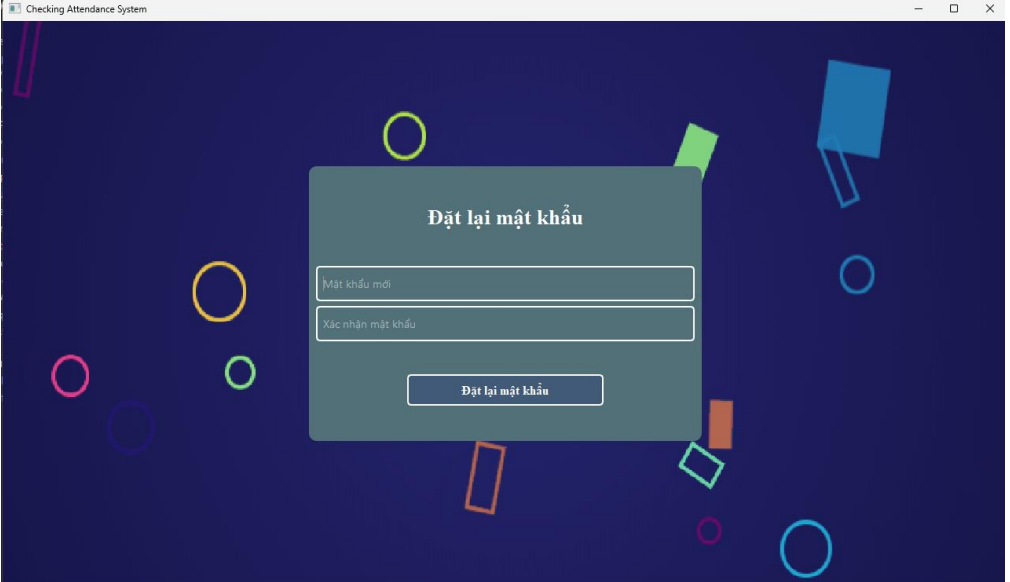
Hình 4.3. Giao diện quên mật khẩu.

Tương tự với giao diện đăng nhập tài khoản, khi quên mật khẩu, người dùng sẽ sử dụng chức năng quên mật khẩu để chuyển đến giao diện này, sau đó người dùng cần nhập Email đã được đăng ký trong hệ thống, nếu nhập đúng thì sẽ có một mã OTP gồm 6 số được gửi đến email. Người dùng có thể cập nhật lại mật khẩu mởi nếu như mã xác thực được nhập chính xác.



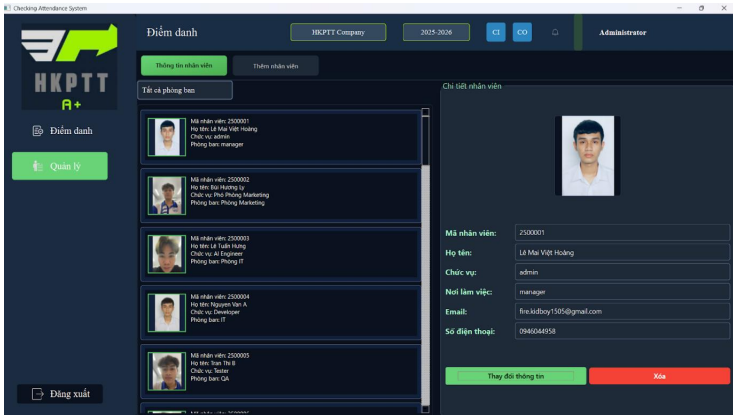
Hình 4.4. Giao diện xác thực OTP.

Giao diện xác thực mã OTP gồm 6 số, được lấy từ Email đã đăng ký nhằm xác minh tài khoản, khi đã nhập xong thì tài khoản vừa đăng ký sẽ được xác 62 thực hoặc yêu cầu điền lại mật khẩu được chấp nhận, sau đó sẽ chuyển đến giao diện đăng nhập hoặc giao diện đặt lại mật khẩu mới nên như mã OTP chính xác



Hình 4.5. Giao diện đặt lại mật khẩu.

Sau khi thực hiện xác thực OTP thông qua chức năng quên mật khẩu, người dùng thực hiện nhập mật khẩu mới, đồng thời nhập thêm mật khẩu một lần nữa để chắc chắn rằng mật khẩu mới là chính xác. Khi bấm nút đặt lại mật khẩu, mật khẩu sẽ được thay đổi và được mã hoá rồi lưu vào hệ thống để đảm bảo tính bảo mật thông tin tài khoản.



Hình 4.6. Giao diện hiển thị thông tin của các nhân viên.

Đây là giao diện được hiển thị ngay sau khi đăng nhập, giao diện bao gồm thanh Sidebar, thanh Header và phần nội dung chính.

Header: Là phần phía trên của trang, bao gồm logo của ứng dụng và các công cụ như bật tắt camera, thông tin của quản lý. Header được cố định tại vị trí trên cùng của giao diện.

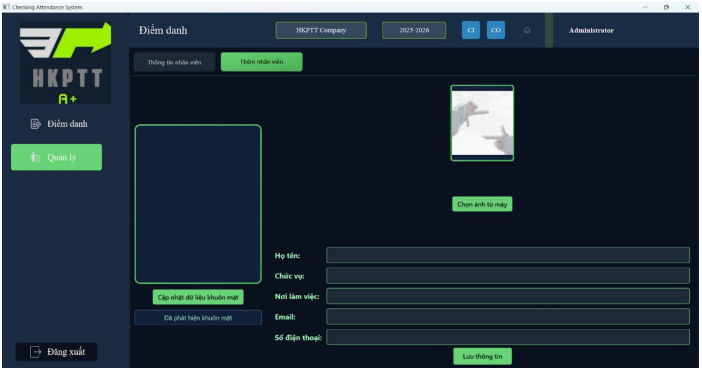
Sidebar: Là phần nằm phía bên trái chứa các chức năng của ứng dụng để có thể chọn là chức năng điểm danh và chức năng quản lý.

Phần nội dung chính được hiển thị dựa vào lựa chọn xem thông tin quản lý hoặc là xem thông tin điểm danh ở phần Sidebar.

Ở giao diện này, tất cả nhân viên sẽ được hiển thị, quản lý sẽ xem chi tiết thông tin của nhân viên bao gồm mã nhân viên, Họ tên, chức vụ, nơi làm việc, email và số điện thoại liên hệ của nhân viên đó bằng cách bấm vào một nhân viên mà quản lý muốn xem thông tin chi tiết. Bên cạnh đó còn có một bộ lọc để có thể lọc nhân viên theo nơi làm việc (phòng ban) để tiện hơn cho việc tìm kiếm nhân viên muốn xem thông tin.

Khi muốn thay đổi hoặc cập nhật thông tin của nhân viên thì quản lý sẽ thực hiện bấm chọn vào người muốn thay đổi. Phía bên phải của giao diện sẽ hiển thị các thông tin có thể thay đổi, bấm vào các trường cần thay đổi và nhập thông tin mới vào, ngoại trừ Mã nhân viên và Email là không thể thay đổi. Sau đó, khi bấm vào nút “Thay đổi thông tin” thì sẽ cập nhật thông tin mới của nhân viên đó vào hệ thống, các thông tin mới sẽ ngay lập tức hiển thị.

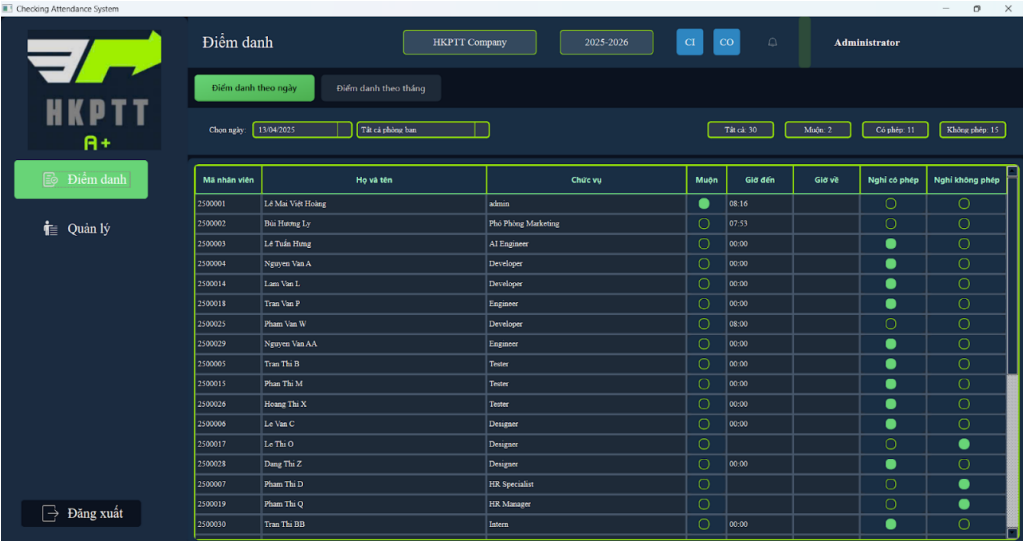
Ngoài ra, khi muốn xoá nhân viên được chọn, quản lý thực hiện bấm nút “Xoá” trên giao diện, sẽ có một thông báo xác nhận có xoá hay không và 64 khi bấm xác nhận, các thông tin của nhân viên đó sẽ biến mất bao gồm cả dữ liệu điểm danh của nhân viên đó trong những ngày trước đó.



Hình 4.7. Giao diện thêm một nhân viên mới.

Giao diện này được sử dụng để lưu vector embedding của nhân viên được thêm mới, với những thông tin của nhân viên. Khi chọn chức năng thêm nhân viên mới, thì ta cần cập nhật ảnh đại diện, các thông tin cá nhân của nhân viên mới và cả dữ liệu về vector embedding. Các hành động thêm nhân viên mới được thực hiện như sau:

* Bấm vào nút cập nhật dữ liệu khuôn mặt để mở camera quét vector embedding, nhân viên mới sẽ để khuôn mặt mình vào phần camera, nếu Camera hiển thị được khuôn mặt thì sẽ có dòng chữ “Đã phát hiện được khuôn mặt” ở dưới, còn nếu khống, dòng chữ “Chưa phát hiện được khuôn mặt” xuất hiện.
* Nhập các dữ liệu cá nhân của nhân viên và bấm lưu thông tin, lúc này Camera sẽ chụp một bức ảnh có chứa khuôn mặt đã phát hiện của nhân viên và chuyển dữ liệu sang thành dạng vector embedding, lưu vào trong hệ thống, đồng thời các thông tin khác cũng được lưu vào cơ sở dữ liệu.

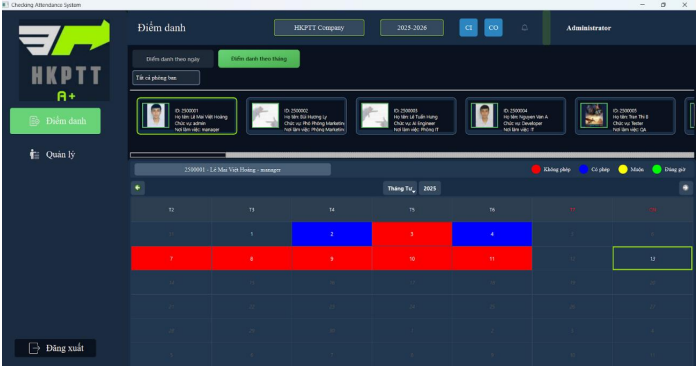


Hình 4.8. Giao diện hiển thị danh sách chấm công theo ngày

Giao diện này bao gồm khả năng tính tổng số nhân viên, hiển thị thông tin điểm danh của tất cả nhân viên bao gồm giờ có mặt, giờ ra về, có muộn hay không, nghỉ có phép hay nghỉ không phép.

Giao diện này cũng hỗ trợ xử lý các yêu cầu nghỉ phép hoặc các yêu cầu khác liên quan đến dữ liệu điểm danh cần phải thay đổi để đảm bảo tính minh bạch.

Quản lý sẽ chọn một ngày, và trong ngày đó sẽ hiển thị danh sách các thông tin của nhân viên bao gồm mã nhân viên, họ tên và chức vụ, bên cạnh đó là trạng thái điểm danh của các nhân viên trong ngày đó. Đồng thời còn có một bộ lọc phòng ban để xem được trạng thái có mặt của các nhân viên trong phòng ban được chọn.



Hình 4.9. Giao diện hiển thị thông tin điểm danh của một nhân viên trong một tháng.

Ở giao diện này, quản lý có thể chọn một nhân viên cần để xem thông tin điểm danh theo tháng từ phòng ban, sau đó giao diện sẽ hiển thị trạng thái điểm danh của nhân viên đó theo tháng bao gồm Đúng giờ, Muốn giờ, Vắng có phép, vắng không phép. Các trạng thái này được cập nhật ngay lập tức khi có thông tin điểm danh mới được gửi về.

Khi chọn bộ lọc theo phòng ban, tất cả nhân viên của phòng ban đó sẽ hiển thị theo danh sách ngang. Phần nội dung bên dưới sẽ hiển thị dưới dạng Calendar, một trang lịch và đánh dấu trạng thái điểm danh của từng ngày khác nhau của nhân viên đó bằng các màu sắc khác nhau, giúp minh hoạ một cách trực quan tình hình chấm công của nhân viên trong 1 tháng

# KẾT LUẬN

## 1. Tổng kết kết quả đạt được

Đề tài “**Hệ thống chấm công nhận dạng khuôn mặt bằng mô hình FaceNet**” được thực hiện với mục tiêu ứng dụng công nghệ trí tuệ nhân tạo (AI), đặc biệt là học sâu (Deep Learning) và thị giác máy tính (Computer Vision), nhằm thay thế các phương pháp điểm danh truyền thống vốn còn nhiều bất cập và hạn chế. Sau quá trình nghiên cứu, triển khai và thử nghiệm, nhóm đã hoàn thành một hệ thống tương đối hoàn chỉnh và đạt được nhiều kết quả nổi bật:

* **Hệ thống chấm công nhận dạng khuôn mặt bằng mô hình FaceNet kết hợp với MTCNN**, sử dụng kiến trúc InceptionResnetV1 và huấn luyện trên tập dữ liệu Labelled Faces in the Wild (LFW). Hệ thống có khả năng nhận diện chính xác khuôn mặt trong thời gian thực, với các chỉ số đánh giá đạt mức cao: Accuracy 90%, Precision 89%, Recall 82% và F1-score 85%.
* **Tự động hóa toàn bộ quy trình điểm danh**, từ phát hiện khuôn mặt đến xác định danh tính và ghi nhận dữ liệu vào cơ sở dữ liệu, giúp tiết kiệm thời gian, giảm thiểu công sức quản lý và loại bỏ hiện tượng điểm danh hộ – một vấn đề phổ biến trong môi trường học đường và doanh nghiệp.
* **Phát triển giao diện quản lý hệ thống thân thiện và trực quan**, bao gồm các chức năng như đăng nhập/đăng ký, xác thực OTP, quản lý danh sách nhân viên, thêm/xóa nhân viên, xem lịch sử điểm danh, và xuất báo cáo. Giao diện giúp người dùng không chuyên về kỹ thuật cũng dễ dàng thao tác và sử dụng.
* **Kết nối hệ thống với thiết bị phần cứng như Raspberry Pi kết hợp với camera**, đảm bảo tính linh hoạt và khả năng ứng dụng thực tế ở những nơi có nguồn lực hạn chế. Hệ thống hoạt động ổn định trong môi trường giả lập và thực tế với dữ liệu thu thập từ lớp học hoặc nơi làm việc.

Nhìn chung, đề tài đã thành công trong việc **tích hợp các công nghệ hiện đại** để giải quyết một bài toán thực tiễn, mang tính ứng dụng cao, đồng thời củng cố kiến thức lý thuyết và kỹ năng lập trình, thiết kế hệ thống, xử lý dữ liệu cho các thành viên trong nhóm.

## 2. Hạn chế của hệ thống

Dù hệ thống đã đạt được nhiều thành tựu nhất định, trong quá trình triển khai và thử nghiệm, nhóm nhận thấy vẫn còn tồn tại một số điểm hạn chế cần khắc phục:

* **Độ chính xác nhận diện còn bị ảnh hưởng bởi điều kiện môi trường**, đặc biệt là khi ánh sáng yếu, góc quay khuôn mặt không chính diện, hoặc khi có nhiều yếu tố gây nhiễu như mũ, kính râm, hoặc che khuôn mặt. Trong những tình huống này, hệ thống đôi khi bỏ sót hoặc nhận sai khuôn mặt.
* **Khả năng nhận diện khi người dùng đeo khẩu trang chưa được hỗ trợ**, điều này là một hạn chế lớn trong bối cảnh sau đại dịch và trong các môi trường yêu cầu sử dụng khẩu trang thường xuyên như bệnh viện, nhà máy, v.v.
* **Hiệu suất của hệ thống khi triển khai trên thiết bị nhúng vẫn còn hạn chế**, do tài nguyên phần cứng bị giới hạn (RAM, CPU, GPU), nên thời gian xử lý có thể bị kéo dài khi số lượng người điểm danh lớn hoặc khi camera ghi nhận nhiều khuôn mặt cùng lúc.
* **Thiếu tính năng đồng bộ và thông báo thời gian thực**, chẳng hạn như gửi thông báo qua SMS hoặc email khi có sự kiện bất thường (vắng không phép, đến muộn), hoặc đồng bộ dữ liệu với hệ thống quản lý nhân sự tổng thể.
* **Chưa tích hợp được bảo mật nâng cao**, như mã hóa dữ liệu khuôn mặt, xác thực hai yếu tố, hoặc cơ chế phân quyền truy cập chi tiết – các yếu tố cần thiết khi hệ thống được áp dụng ở quy mô lớn hoặc môi trường nhạy cảm.

## 3. Hướng phát triển

Từ những kết quả đạt được cũng như các hạn chế còn tồn tại, nhóm định hướng tiếp tục phát triển và hoàn thiện hệ thống theo các hướng sau:

* **Nâng cao độ chính xác và khả năng tổng quát hóa của mô hình nhận diện khuôn mặt** thông qua việc thu thập thêm dữ liệu thực tế với nhiều biến thể, áp dụng các kỹ thuật tăng cường dữ liệu (data augmentation nâng cao), fine-tune mô hình trên các bộ dữ liệu chuyên biệt, và sử dụng các mô hình tiên tiến hơn như ArcFace, MobileFaceNet hoặc các mô hình transformer-based trong thị giác máy tính.
* **Tối ưu hóa hệ thống cho thiết bị nhúng** bằng cách rút gọn mô hình (model compression), sử dụng kỹ thuật như pruning, quantization, hoặc chuyển sang dùng mô hình nhẹ như MobileNetV3 để đảm bảo tốc độ xử lý và tiết kiệm tài nguyên phần cứng.
* **Phát triển các tính năng thông báo thời gian thực và đồng bộ hóa dữ liệu**, cụ thể là tích hợp gửi thông báo tự động qua SMS/email khi có trạng thái điểm danh bất thường, hoặc khi cần xác thực danh tính lại. Đồng thời, tích hợp với các hệ thống quản lý nhân sự hiện có để tạo thành một nền tảng tổng thể.
* **Mở rộng kênh truy cập hệ thống thông qua ứng dụng di động hoặc dashboard web**, cho phép người quản lý theo dõi tình hình điểm danh từ xa, thực hiện điều chỉnh, duyệt yêu cầu nghỉ phép, và xem báo cáo trực tuyến. Điều này giúp nâng cao trải nghiệm người dùng và tính linh hoạt của hệ thống.
* **Tăng cường bảo mật hệ thống** bằng cách triển khai mã hóa dữ liệu khuôn mặt, phân quyền người dùng chi tiết hơn (admin, supervisor, nhân viên), và sử dụng các biện pháp xác thực mạnh mẽ hơn như xác thực hai lớp (2FA) hoặc xác minh danh tính bằng nhiều phương thức sinh trắc học kết hợp (multimodal biometrics).
* **Nghiên cứu mở rộng mô hình sang các lĩnh vực khác** như kiểm soát truy cập ra vào, điểm danh tại sự kiện, theo dõi ca làm việc trong nhà máy, từ đó biến hệ thống điểm danh thành một giải pháp nhận diện toàn diện và đa mục đích.

# TÀI LIỆU THAM KHẢO

AI with Misa. (2021). *Mạng thần kinh nhân tạo - Artificial Neural Network.* . Retrieved from https://aiwithmisa.com/2021/03/07/aml-bai15/

Built In. (2024). *Beginner’s Guide to VGG16 Implementation in Keras*. Retrieved from https://builtin.com/machine-learning/vgg16

GeeksforGeeks. (2024). *Introduction to Padding*. Retrieved from https://www.geeksforgeeks.org/cnn-introduction-to-padding/