

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN TỬ - VIỄN THÔNG

ĐỒ ÁN TỐT NGHIỆP

NGÀNH: KỸ THUẬT ĐIỆN TỬ VIỄN THÔNG

CHUYÊN NGÀNH: KỸ THUẬT ĐIỆN TỬ

ĐỀ TÀI:

**GIẢI PHÁP CHĂM CÔNG CHO DOANH NGHIỆP
ỨNG DỤNG NHẬN ĐIỆN KHUÔN MẶT**

Người hướng dẫn: TS. NGUYỄN DUY NHẬT VIỄN

Sinh viên thực hiện: NGUYỄN ĐÌNH KHÁNH

HÀ PHƯỚC PHÚC

Số thẻ sinh viên: 106200056

106200065

Lớp: 20DT2

Đà Nẵng, 06/2025

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN TỬ - VIỄN THÔNG

ĐỒ ÁN TỐT NGHIỆP

NGÀNH: KỸ THUẬT ĐIỆN TỬ VIỄN THÔNG

CHUYÊN NGÀNH: KỸ THUẬT ĐIỆN TỬ

ĐỀ TÀI:

GIẢI PHÁP CHẤM CÔNG CHO DOANH NGHIỆP DỤNG NHẬN ĐIỆN KHUÔN MẶT

Người hướng dẫn: TS. NGUYỄN DUY NHẬT VIỄN

Sinh viên thực hiện: NGUYỄN ĐÌNH KHÁNH

HÀ PHƯỚC PHÚC

Số thẻ sinh viên: 106200056

106200065

Lớp: 20DT2

Đà Nẵng, 06/2025

NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP

I. Thông tin chung:

1. Họ và tên sinh viên: **NGUYỄN ĐÌNH KHÁNH**
2. Lớp: **20DT2** Số thẻ SV: **106200056**
3. Tên đề tài: **Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt.**
4. Người hướng dẫn: **NGUYỄN DUY NHẬT VIỄN** Học hàm/ học vị: **Tiến sĩ**

II. Nhận xét, đánh giá đồ án tốt nghiệp:

1. Về tính cấp thiết, tính mới, khả năng ứng dụng của đề tài: (điểm tối đa là 2đ)
.....
.....
2. Về kết quả giải quyết các nội dung nhiệm vụ yêu cầu của đồ án: (điểm tối đa là 4đ)
.....
.....
3. Về hình thức, cấu trúc, bố cục của đồ án tốt nghiệp: (điểm tối đa là 2đ)
.....
.....
4. Đề tài có giá trị khoa học/ có bài báo/ giải quyết vấn đề đặt ra của doanh nghiệp hoặc nhà trường: (điểm tối đa là 1đ)
.....
.....
5. Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa:
.....
.....

III. Tinh thần, thái độ làm việc của sinh viên: (điểm tối đa 1đ)

.....

IV. Đánh giá:

1. Điểm đánh giá: /10 (lấy đến 1 số lẻ thập phân)
2. Đề nghị: ☐ Được bảo vệ đồ án ☐ Bổ sung để bảo vệ ☐ Không được bảo vệ

Đà Nẵng, ngày tháng năm 2025

Người hướng dẫn

NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP

I. Thông tin chung:

1. Họ và tên sinh viên: **HÀ PHƯỚC PHÚC**
2. Lớp: **20DT2** Số thẻ SV: **106200065**
3. Tên đề tài: **Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt.**
4. Người hướng dẫn: **NGUYỄN DUY NHẬT VIỄN** Học hàm/ học vị: **Tiến sĩ**

II. Nhận xét, đánh giá đồ án tốt nghiệp:

1. Về tính cấp thiết, tính mới, khả năng ứng dụng của đề tài: (điểm tối đa là 2đ)
-
-
2. Về kết quả giải quyết các nội dung nhiệm vụ yêu cầu của đồ án: (điểm tối đa là 4đ)
-
-
3. Về hình thức, cấu trúc, bố cục của đồ án tốt nghiệp: (điểm tối đa là 2đ)
-
-
4. Đề tài có giá trị khoa học/ có bài báo/ giải quyết vấn đề đặt ra của doanh nghiệp hoặc nhà trường: (điểm tối đa là 1đ)
-
-
5. Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa:
-
-

III. Tinh thần, thái độ làm việc của sinh viên: (điểm tối đa 1đ)

.....

IV. Đánh giá:

1. Điểm đánh giá: /10 (lấy đến 1 số lẻ thập phân)
2. Đề nghị: ☐ Được bảo vệ đồ án ☐ Bổ sung để bảo vệ ☐ Không được bảo vệ

Đà Nẵng, ngày tháng năm 2025

Người hướng dẫn

NHẬN XÉT PHẢN BIỆN ĐỒ ÁN TỐT NGHIỆP

I. Thông tin chung:

1. Họ và tên sinh viên: **NGUYỄN ĐÌNH KHÁNH**
2. Lớp: **20DT2** Số thẻ SV: **106200056**
3. Tên đề tài: **Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt.**
4. Người hướng dẫn: **NGUYỄN DUY NHẬT VIỄN** Học hàm/ học vị: **Tiến sĩ**

II. Nhận xét, đánh giá đồ án tốt nghiệp:

TT	Các tiêu chí đánh giá	Điểm tối đa	Điểm đánh giá
1	Sinh viên có phương pháp nghiên cứu phù hợp, giải quyết đủ nhiệm vụ đồ án được giao	80	
1a	- Tính mới (nội dung chính của ĐATN có những phần mới so với các ĐATN trước đây). - Đề tài có giá trị khoa học, công nghệ; có thể ứng dụng thực tiễn.	15	
1b	- Kỹ năng giải quyết vấn đề; hiểu, vận dụng được kiến thức cơ bản, cơ sở, chuyên ngành trong vấn đề nghiên cứu. - Chất lượng nội dung ĐATN (thuyết minh, bản vẽ, chương trình, mô hình, ...).	50	
1c	- Có kỹ năng vận dụng thành thạo các phần mềm ứng dụng trong vấn đề nghiên cứu; - Có kỹ năng đọc, hiểu tài liệu bằng tiếng nước ngoài ứng dụng trong vấn đề nghiên cứu; - Có kỹ năng làm việc nhóm;	15	
2	Kỹ năng viết:	20	
2a	- Bố cục hợp lý, lập luận rõ ràng, chặt chẽ, lời văn súc tích	15	
2b	- Thuyết minh đồ án không có lỗi chính tả, in ấn, định dạng	5	
3	Tổng điểm đánh giá theo thang 100:		
	Quy về thang 10 (lấy đến 1 số lẻ)		

- Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa:

.....

- Câu hỏi đề nghị sinh viên trả lời trong buổi bảo vệ:

.....

.....

- Đề nghị: ☐ Được bảo vệ đồ án ☐ Bổ sung để bảo vệ ☐ Không được bảo vệ

Đà Nẵng, ngày tháng năm 2025

Người phản biện

NHẬN XÉT PHẢN BIỆN ĐỒ ÁN TỐT NGHIỆP

III. Thông tin chung:

1. Họ và tên sinh viên: **HÀ PHƯỚC PHÚC**
2. Lớp: **20DT2** Số thẻ SV: **106200065**
3. Tên đề tài: **Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt.**
4. Người hướng dẫn: **NGUYỄN DUY NHẬT VIỄN** Học hàm/ học vị: **Tiến sĩ**

IV. Nhận xét, đánh giá đồ án tốt nghiệp:

TT	Các tiêu chí đánh giá	Điểm tối đa	Điểm đánh giá
1	Sinh viên có phương pháp nghiên cứu phù hợp, giải quyết đủ nhiệm vụ đồ án được giao	80	
1a	- Tính mới (nội dung chính của ĐATN có những phần mới so với các ĐATN trước đây). - Đề tài có giá trị khoa học, công nghệ; có thể ứng dụng thực tiễn.	15	
1b	- Kỹ năng giải quyết vấn đề; hiểu, vận dụng được kiến thức cơ bản, cơ sở, chuyên ngành trong vấn đề nghiên cứu. - Chất lượng nội dung ĐATN (thuyết minh, bản vẽ, chương trình, mô hình, ...).	50	
1c	- Có kỹ năng vận dụng thành thạo các phần mềm ứng dụng trong vấn đề nghiên cứu; - Có kỹ năng đọc, hiểu tài liệu bằng tiếng nước ngoài ứng dụng trong vấn đề nghiên cứu; - Có kỹ năng làm việc nhóm;	15	
2	Kỹ năng viết:	20	
2a	- Bố cục hợp lý, lập luận rõ ràng, chặt chẽ, lời văn súc tích	15	
2b	- Thuyết minh đồ án không có lỗi chính tả, in ấn, định dạng	5	
3	Tổng điểm đánh giá theo thang 100:		
	Quy về thang 10 (lấy đến 1 số lẻ)		

- Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa:

.....

- Câu hỏi đề nghị sinh viên trả lời trong buổi bảo vệ:

.....

.....

- Đề nghị: ☐ Được bảo vệ đồ án ☐ Bổ sung để bảo vệ ☐ Không được bảo vệ

Đà Nẵng, ngày tháng năm 2025

Người phản biện

TÓM TẮT

Tên đề tài: Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt.

Sinh viên thực hiện: Nguyễn Đình Khánh

Số thẻ SV: 106200056 Lớp: 20DT2

Sinh viên thực hiện: Hà Phước Phúc

Số thẻ SV: 106200065 Lớp: 20DT2

Đồ án tốt nghiệp “Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt” xây dựng một hệ thống chấm công tự động dựa trên công nghệ thị giác máy tính và trí tuệ nhân tạo. Hệ thống sử dụng Mediapipe để phát hiện khuôn mặt, FaceNet để trích xuất đặc trưng khuôn mặt và Support Vector Machine (SVM) để phân loại và nhận diện nhân viên. Tất cả được triển khai trên nền tảng phần cứng Raspberry Pi 4B, với giao diện người dùng trực quan bằng Tkinter.

Dữ liệu nhân viên và thời gian chấm công được lưu trữ dưới dạng CSV và đồng bộ liên tục với cơ sở dữ liệu MySQL. Hệ thống hỗ trợ các chức năng quản lý nhân sự như thêm, sửa, xóa thông tin nhân viên, yêu cầu đăng nhập quản trị viên bằng mật khẩu được mã hóa SHA256. Chức năng gửi mã OTP qua email được tích hợp để xác thực khi thay đổi mật khẩu admin, tăng cường tính bảo mật.

Kết quả thực nghiệm cho thấy hệ thống đạt độ chính xác cao trong các điều kiện thay đổi biểu cảm, đeo kính, thay đổi ánh sáng, tuy nhiên hiệu quả giảm khi khuôn mặt bị che khuất. Đây là giải pháp phù hợp với các doanh nghiệp vừa và nhỏ tại Việt Nam, góp phần hiện đại hóa quy trình quản lý nhân sự.

Để thực hiện được đề tài này đồ án sẽ bao gồm 4 chương như sau:

Chương 1: Tổng quan về đề tài

Chương 2: Cơ sở lý thuyết

Chương 3: Thiết kế và triển khai

Chương 4: Kết quả và đánh giá

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

TT	Họ tên sinh viên	Số thẻ SV	Lớp	Ngành
1	Nguyễn Đình Khánh	106200056	20DT2	Kỹ thuật Điện tử - Viễn thông
2	Hà Phước Phúc	106200065	20DT2	Kỹ thuật Điện tử - Viễn thông

1. Tên đề tài đồ án:

Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt.

2. Đề tài thuộc diện: ☐ Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện

3. Các số liệu và dữ liệu ban đầu:

- Dữ liệu hình ảnh khuôn mặt được thu thập từ các nguồn công khai như GitHub, Kaggle, kết hợp với một phần dữ liệu do nhóm tự thu thập. Thông tin chi tiết về các bộ dữ liệu này được trình bày cụ thể trong Chương 4.
- Thông tin về mô hình nhận diện khuôn mặt: Mediapipe để phát hiện và tiền xử lý khuôn mặt, mô hình Pretrained Fcaenet để trích xuất đặc trưng, thuật toán SVM cho mô hình phân loại.
- Thông tin nhân sự: Tên, mã nhân viên, vị trí làm việc.
- Phần mềm và công nghệ sử dụng: Python, OpenCV, TensorFlow/Keras, scikit-learn, Mediapipe, Tkinter, ...

4. Nội dung các phần thuyết minh và tính toán:

a. Phần chung:

- Tìm hiểu, phân tích yêu cầu đề tài
- Xác định kiến trúc hệ thống
- Nghiên cứu Raspberry Pi và các thư viện nhận diện khuôn mặt.
- Thiết kế sơ đồ khối chức năng và luồng dữ liệu hệ thống
- Kiểm thử giao diện hiển thị và kết nối các chức năng
- Thiết kế, xây dựng trang web
- Thiết kế và xây dựng giao diện hiển thị bằng Tkinter
- Kiểm thử và sửa lỗi hệ thống
- Viết báo cáo và slide trình bày

b. Phần riêng:

TT	Họ tên sinh viên	Nội dung
1	Nguyễn Đình Khánh	Xây dựng tính năng đăng nhập cho giao diện Tkinter
2		Triển khai truy cập từ xa bằng Cloudflare Tunnel
3		Đánh giá mô hình
4		Xây dựng tính năng tự đồng bộ dữ liệu MySQL theo thời gian thực cho Web App
5		Thiết kế giao diện Tkinter
6	Hà Phước Phúc	Xây dựng mô hình nhận diện khuôn mặt
7		Phát triển các chức năng cho hệ thống nhận diện
8		Xây dựng và kết nối cơ sở dữ liệu MySQL
9		Xây dựng Web App
10		Đồng bộ dữ liệu từ CSV sang MySQL

5. Các bản vẽ, đồ thị (ghi rõ các loại và kích thước bản vẽ):

- Sơ đồ khối hệ thống
- Các lưu đồ thuật toán
- Ma trận nhầm lẫn của mô hình

6. Họ tên người hướng dẫn: TS. Nguyễn Duy Nhật Viễn

7. Ngày giao nhiệm vụ đồ án: 03/03/2025

8. Ngày hoàn thành đồ án: 28/06/2025

Đà Nẵng, ngày tháng 06 năm 2025

Trưởng Bộ môn.....

Người hướng dẫn

TS. Nguyễn Duy Nhật Viễn

LỜI NÓI ĐẦU

Trong hành trình học tập và nghiên cứu tại Trường Đại học Bách khoa – Đại học Đà Nẵng, nhóm đã có cơ hội tiếp cận với nhiều kiến thức chuyên ngành quý giá, cũng như rèn luyện kỹ năng làm việc nhóm và giải quyết vấn đề thực tiễn. Đồ án tốt nghiệp với đề tài “*Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt*” là một cột mốc quan trọng đánh dấu sự kết hợp giữa lý thuyết và ứng dụng công nghệ vào đời sống, đặc biệt trong lĩnh vực quản lý nhân sự hiện đại.

Hệ thống được thiết kế nhằm mang lại một phương thức chấm công tiện lợi, nhanh chóng và an toàn cho doanh nghiệp, góp phần nâng cao hiệu quả làm việc và giảm thiểu gian lận trong quản lý thời gian lao động. Đây là một hướng đi phù hợp với xu thế chuyển đổi số đang diễn ra mạnh mẽ trong các doanh nghiệp hiện nay.

Nhóm xin gửi lời cảm ơn chân thành đến toàn thể quý Thầy/Cô Khoa Điện tử - Viễn thông, những người đã truyền đạt không chỉ kiến thức chuyên môn mà còn là tinh thần học tập nghiêm túc và trách nhiệm trong suốt những năm qua.

Đặc biệt, nhóm xin bày tỏ lòng biết ơn sâu sắc đến TS. Nguyễn Duy Nhật Viễn – người đã luôn tận tình hướng dẫn, hỗ trợ về mặt chuyên môn cũng như định hướng xuyên suốt quá trình thực hiện đồ án. Những góp ý và chỉ dẫn quý báu từ thầy đã giúp nhóm nhóm hoàn thiện sản phẩm một cách rõ ràng và thực tế hơn.

Dù đã cố gắng hết mình, nhưng do hạn chế về thời gian và kinh nghiệm, đồ án vẫn có thể còn một số thiếu sót. Rất mong nhận được ý kiến đóng góp từ quý Thầy Cô và các bạn để tiếp tục cải thiện và phát triển hệ thống trong thời gian tới.

Nhóm xin chân thành cảm ơn!

CAM ĐOAN

Nhóm xin cam đoan rằng đề tài “*Giải pháp chấm công cho doanh nghiệp ứng dụng nhận diện khuôn mặt*” là kết quả nghiên cứu và thực hiện của chính nhóm. Các tài liệu tham khảo, hình ảnh và nội dung được trích dẫn trong quá trình thực hiện đều đã được ghi rõ nguồn gốc trong phần Tài liệu tham khảo. Mọi số liệu, kết quả và phân tích được trình bày trong báo cáo là trung thực và phản ánh đúng quá trình làm việc của nhóm.

Nhóm hoàn toàn chịu trách nhiệm trước nhà trường về tính trung thực và nội dung của bản báo cáo này.

Sinh viên thực hiện

Nguyễn Đình Khánh

Hà Phước Phúc

MỤC LỤC

Tóm tắt	
Nhiệm vụ đồ án tốt nghiệp	
Lời nói đầu.....	i
Cam đoan.....	ii
Mục lục	iii
Danh sách các bảng, hình vẽ	vii
Danh sách các ký hiệu, chữ viết tắt	ix
	Trang
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI.....	2
1.1. Giới thiệu chương	2
1.2. Tính cấp thiết	2
1.3. Mục tiêu đề tài	2
1.3.1. Mục tiêu tổng quát.....	2
1.3.2. Mục tiêu cụ thể	3
1.4. Đối tượng và phạm vi nghiên cứu	3
1.4.1. Đối tượng nghiên cứu	3
1.4.2. Phạm vi nghiên cứu	3
1.5. Phương pháp nghiên cứu	3
1.6. Phân tích đề tài	4
1.6.1. Bài toán chấm công	4
1.6.2. Công nghệ nhận diện khuôn mặt	4
1.7. Kết luận chương.....	4
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	5
2.1. Giới thiệu chương	5
2.2. Các công nghệ nền tảng cho nhận diện khuôn mặt.....	5
2.2.1. Phát hiện khuôn mặt bằng Mediapipe	5

2.2.2. Trích xuất đặc trưng bằng FaceNet	6
2.2.3. Phân loại và nhận diện khuôn mặt bằng thuật toán SVM	8
2.3. Tổng quan về phần cứng.....	9
2.3.1. Raspberry Pi 4 Model B	9
2.3.2. Webcam Camera HD 1080P	11
2.3.3. Màn hình FPT Elead F18RW	12
2.3.4. Bàn phím cơ Biojee	13
2.4. Các phần mềm và dịch vụ sử dụng trong hệ thống.....	14
2.4.1. XAMPP	14
2.4.2. Cloudflare Tunnel.....	15
2.4.3. Email OTP	16
2.5. Các công cụ phát triển giao diện	17
2.5.1. Tkinter	17
2.5.2. Ngôn ngữ lập trình PHP	17
2.6. Kết luận chương.....	18
CHƯƠNG 3: THIẾT KẾ VÀ TRIỂN KHAI.....	19
3.1. Giới thiệu chương	19
3.2. Các yêu cầu về tính năng và kỹ thuật.....	19
3.3. Sơ đồ khối chức năng	19
3.4. Thiết kế và xây dựng mô hình huấn luyện	20
3.4.1. Mô hình trích xuất đặc trưng FaceNet:.....	20
3.4.2. Mô hình phân loại SVC:.....	21
3.4.3. Các thư viện cần thiết	21
3.4.4. Thu thập và tiền xử lý dữ liệu.....	22
3.4.5. Xây dựng và huấn luyện mô hình.....	22
3.5. Thiết kế khối thiết bị	23
3.5.1. Xây dựng lưu đồ chính của hệ thống.....	24
3.5.2. Lưu đồ con chương trình Khởi tạo	26

3.5.3. Lưu đồ con chương trình Kết nối và đồng bộ MySQL	27
3.5.4. Lưu đồ con chương trình Đăng nhập.....	28
3.5.5. Lưu đồ con chương trình Chấm công.....	30
3.5.6. Lưu đồ con chương trình Thêm nhân viên	32
3.5.7. Lưu đồ con chương trình Xóa nhân viên.....	34
3.5.8. Lưu đồ con chương trình Sửa thông tin.....	36
3.5.9. Lưu đồ con chương trình Đổi mật khẩu	38
3.6. Thiết kế và triển khai khối Web App	40
3.6.1. Web App.....	40
3.6.2. Cloudflare Tunnel.....	45
3.7. Kết luận chương.....	46
CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ.....	47
4.1. Giới thiệu chương	47
4.2. Giao diện người dùng.....	47
4.2.1 Tab Chấm Công.....	49
4.2.2. Tab Thêm Nhân Viên	50
4.2.3. Tab Xóa Nhân Viên.....	50
4.2.4. Tab Sửa Thông Tin.....	51
4.2.5. Tab Đổi mật khẩu	52
4.3 Giao diện web.....	54
4.4. Kết quả thao tác Xóa nhân viên	57
4.5. Kết quả thao tác Thêm nhân viên	59
4.6. Kết quả thao tác Sửa thông tin.....	61
4.7. Kết quả thao tác Chấm công	62
4.8. Kết quả thao tác Đổi mật khẩu	63
4.9. Đánh giá.....	65
4.9.1. Đánh giá mô hình huấn luyện.....	65
4.9.2. Đánh giá các yếu tố thay đổi cùng một người.....	67

4.10. Kết luận chương.....	68
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	69
TÀI LIỆU THAM KHẢO	70
PHỤ LỤC	

DANH SÁCH CÁC BẢNG, HÌNH VẼ

Bảng 2.1 Đánh giá độ chính xác giữa hai mô hình	7
Bảng 4.1 Đánh giá độ chính xác các yếu tố thay đổi cùng một người.....	67
Hình 2.1 Tạo Vector từ khuôn mặt [5]	6
Hình 2.2 Mô hình Inception Resnet V1 [6].....	6
Hình 2.3 Thuật toán Triplet loss mô phỏng quá trình nhận diện [8].....	8
Hình 2.4 Raspberry Pi 4 Model B [9].....	10
Hình 2.5 Webcam Camera HD 1080P	11
Hình 2.6 Màn hình FPT Elead F18RW	12
Hình 2.7 Bàn phím cơ Biojee	13
Hình 2.8 Bảng điều khiển XAMPP	14
Hình 2.9 Sơ đồ quy trình hoạt động của Cloudflare Tunnel	15
Hình 2.10 Dịch vụ xác thực OTP	16
Hình 3.1 Sơ đồ khối của hệ thống	20
Hình 3.2 Lưu đồ chính của hệ thống	24
Hình 3.3 Lưu đồ chương trình con khởi tạo của hệ thống	26
Hình 3.4 Lưu đồ chương trình con kết nối và đồng bộ MySQL	27
Hình 3.5 Lưu đồ chương trình con đăng nhập	29
Hình 3.6 Lưu đồ chương trình con chấm công.....	31
Hình 3.7 Lưu đồ chương trình con thêm nhân viên	33
Hình 3.8 Lưu đồ chương trình con xóa nhân viên.....	35
Hình 3.9 Lưu đồ chương trình con sửa thông tin nhân viên	37
Hình 3.10 Lưu đồ chương trình con đổi mật khẩu	39
Hình 3.11 Lưu đồ quy trình chính của trang web.....	41
Hình 3.12 Lưu đồ quy trình đăng nhập	42
Hình 3.13 Lưu đồ con của các tab thông tin nhân viên và điểm danh	43
Hình 3.14 Lưu đồ quy trình đổi mật khẩu	43
Hình 3.15 Các bảng trong cơ sở dữ liệu MySQL.....	44
Hình 3.16 Các vị trí cần cấu hình trên trang web của Cloudflare	45
Hình 3.17 Dòng lệnh được cấp bởi hệ thống của Cloudflare.....	46
Hình 4.1 Giao diện chính của hệ thống	47

Hình 4.2 Giao diện đăng nhập	48
Hình 4.3 Giao diện tab Chấm Công	49
Hình 4.4 Giao diện tab Thêm Nhân Viên.....	50
Hình 4.5 Giao diện tab Xóa Nhân Viên	51
Hình 4.6 Giao diện tab Sửa Thông Tin	52
Hình 4.7 Giao diện nhập mật khẩu cũ, mới.....	53
Hình 4.8 Giao diện nhập mã OTP	53
Hình 4.9 Giao diện trang đăng nhập.....	54
Hình 4.10 Giao diện tab thông tin nhân viên	54
Hình 4.11 Giao diện tab ghi nhận chấm công	55
Hình 4.12 Giao diện nhập mật khẩu cũ	56
Hình 4.13 Giao diện nhập mã OTP và mật khẩu mới	56
Hình 4.14 Thư mục lưu thông tin ảnh của Nhân viên	57
Hình 4.15 Thông tin Nhân viên trong CSV và MySQL.....	57
Hình 4.16 Kiểm tra thông tin của Nhân viên 103	57
Hình 4.17 Kiểm tra thông tin của Nhân viên 103 sau khi xóa	58
Hình 4.18 Thông báo Log khi xóa thành công Nhân viên 103	58
Hình 4.19 Thư mục lưu thông tin ảnh của Nhân viên sau khi xóa 103.....	58
Hình 4.20 Thông tin Nhân viên trong CSV và MySQL sau khi xóa 103	59
Hình 4.21 Thao tác Thêm nhân viên mới 1032.....	59
Hình 4.22 Thông báo Log khi thêm thành công Nhân viên 1302	60
Hình 4.23 Kiểm tra thông tin của Nhân viên mới 1302	60
Hình 4.24 Thư mục lưu thông tin ảnh của Nhân viên sau khi thêm 1032	60
Hình 4.25 Thông tin Nhân viên trong CSV và MySQL sau khi thêm nhân viên 1032	61
Hình 4.26 Thông tin trước và sau khi chỉnh sửa của Nhân viên 1032.	61
Hình 4.27 Thông báo Log sau khi sửa thông tin nhân viên 1302	61
Hình 4.28 Thư mục lưu thông tin ảnh của Nhân viên 1032 đã sửa thông tin	62
Hình 4.29 Thông tin Nhân viên trong CSV và MySQL sau khi sửa.....	62
Hình 4.30 Các thông tin hiển thị kết quả chấm công trên giao diện	63
Hình 4.31 Thông tin chấm công trong file ‘timekeeping.csv’	63
Hình 4.32 Thông tin chấm công ở bảng ‘timekeeping’ trong MySQL	63
Hình 4.33 Ô nhập mật khẩu cũ và mới.....	64
Hình 4.34 Mã OTP được gửi về Email	64
Hình 4.35 Ô chờ nhập OTP trên giao diện Tkinter	64
Hình 4.36 Ma trận nhầm lẫn của mô hình	66

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Chữ viết tắt	Giải thích
SVC	Support Vector Classifier
AI	Artificial Intelligence
SHA-256	Secure Hash Algorithm 256-bit
SQL	Structured Query Language
PHP	Hypertext Preprocessor
SSD	Single Shot Detector
SVM	Support Vector Machine
KNN	K-Nearest Neighbors
HDMI	High-Definition Multimedia Interface
VGA	Video Graphics Array
USB	Universal Serial Bus
OTP	One-Time Password
GUI	Graphical User Interface
JSON	JavaScript Object Notation
CSV	Comma Separated Values
ID	Identification

MỞ ĐẦU

Trong thời đại công nghệ 4.0, việc ứng dụng các giải pháp thông minh vào công tác quản lý nhân sự đang trở thành xu hướng tất yếu. Một trong những vấn đề được nhiều doanh nghiệp quan tâm là làm thế nào để nâng cao hiệu quả và tính chính xác trong việc chăm công của nhân viên. Thực tế cho thấy, các phương pháp chăm công truyền thống vẫn còn tồn tại nhiều hạn chế về độ bảo mật, tính linh hoạt cũng như khả năng kiểm soát. Do đó, nhu cầu phát triển một hệ thống chăm công hiện đại, tự động và thuận tiện là hết sức cần thiết.

Mục tiêu chính của đề tài là xây dựng thành công một hệ thống chăm công tự động sử dụng công nghệ nhận diện khuôn mặt, trong đó tích hợp các thuật toán như Mediapipe để phát hiện khuôn mặt, FaceNet để trích xuất đặc trưng, và SVC để phân loại nhân viên. Hệ thống được thiết kế với giao diện trực quan bằng Tkinter, cho phép quản lý nhân viên (thêm, sửa, xóa) và bảo vệ dữ liệu bằng cơ chế đăng nhập quản trị viên sử dụng mã hóa SHA-256.

Phạm vi nghiên cứu của đề tài tập trung vào việc phát triển hệ thống chăm công tự động dựa trên công nghệ nhận diện khuôn mặt, không mở rộng sang các ứng dụng khác như giám sát an ninh hay thanh toán. Đối tượng nghiên cứu bao gồm các thuật toán trí tuệ nhân tạo được áp dụng trong nhận diện khuôn mặt, hệ thống lưu trữ dữ liệu nhân viên, cũng như trải nghiệm của người dùng cuối trong quá trình sử dụng hệ thống.

Phương pháp nghiên cứu được thực hiện theo hướng thực nghiệm, bao gồm thu thập dữ liệu khuôn mặt (với sự đồng ý của người tham gia) để huấn luyện mô hình, phát triển hệ thống sử dụng Python kết hợp với các thư viện mã nguồn mở như Mediapipe, TensorFlow, và xây dựng giao diện Tkinter với dữ liệu được lưu trữ trên CSV và MySQL. Đồng thời, hệ thống được đánh giá qua các chỉ số như độ chính xác nhận diện, thời gian xử lý và tỷ lệ lỗi trong điều kiện triển khai thực tế.

Cấu trúc của đồ án gồm 4 chương chính:

- Chương 1: Tổng quan về đề tài
- Chương 2: Cơ sở lý thuyết
- Chương 3: Thiết kế và triển khai
- Chương 4: Kết quả và đánh giá

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Giới thiệu chương

Chương 1 sẽ trình bày tổng quan về đề tài, bao gồm tính cấp thiết, mục tiêu, đối tượng, phạm vi nghiên cứu, và phương pháp nghiên cứu, từ đó đặt nền tảng cho các phân tích và giải pháp được đề xuất trong các chương tiếp theo.

1.2. Tính cấp thiết

Hệ thống chấm công là một phần không thể thiếu trong quản lý nhân sự của doanh nghiệp, ảnh hưởng trực tiếp đến hiệu quả vận hành và chi phí. Tuy nhiên, các phương pháp chấm công truyền thống tại Việt Nam hiện nay còn tồn tại nhiều hạn chế:

- Gian lận thời gian: Chấm công hộ (qua thẻ từ) vẫn phổ biến, gây khó khăn trong việc đảm bảo tính minh bạch.
- Phí bảo trì: Hệ thống vân tay hoặc thẻ từ đòi hỏi bảo trì định kỳ, đặc biệt khi thiết bị bị hỏng hoặc dữ liệu bị lỗi.
- Hạn chế kỹ thuật: Trong điều kiện ánh sáng yếu hoặc môi trường đông người, các hệ thống cũ thường hoạt động kém hiệu quả.

Theo các phân tích gần đây, thị trường công nghệ sinh trắc học toàn cầu đang tăng trưởng nhanh. Một báo cáo cho biết thị trường đạt khoảng 14,6 tỷ USD vào năm 2018 và dự kiến tăng trưởng với tốc độ $\sim 16,2\%$ hằng năm để đạt khoảng 55,4 tỷ USD vào năm 2027 [1]. Sự tăng trưởng này được thúc đẩy bởi nhu cầu cao trong các ứng dụng về an ninh, giám sát cũng như trong bán lẻ và quản lý nhân sự. Chẳng hạn, hàng ngàn doanh nghiệp đã áp dụng hệ thống chấm công sinh trắc học (dùng vân tay hoặc nhận diện khuôn mặt) để nâng cao độ chính xác và tính bảo mật [2]. Đặc biệt, công nghệ nhận diện khuôn mặt (hoàn toàn không tiếp xúc) ngày càng được chú ý vì dễ tích hợp vào các hệ thống hiện đại và mang lại trải nghiệm liền mạch, không cần chạm.

Vì vậy, việc nghiên cứu và triển khai giải pháp chấm công ứng dụng nhận diện khuôn mặt là một nhu cầu cấp thiết, góp phần nâng cao hiệu quả quản lý và khả năng cạnh tranh của doanh nghiệp trong thời đại công nghệ 4.0.

1.3. Mục tiêu đề tài

1.3.1. Mục tiêu tổng quát

Phát triển một hệ thống chấm công tự động, chính xác, và bảo mật cho các doanh nghiệp vừa và nhỏ tại Việt Nam, sử dụng công nghệ nhận diện khuôn mặt.

1.3.2. Mục tiêu cụ thể

- Xây dựng hệ thống nhận diện khuôn mặt dựa trên Mediapipe (phát hiện khuôn mặt) và FaceNet (trích xuất đặc trưng), kết hợp thuật toán SVM để phân loại nhân viên.
- Phát triển giao diện quản lý nhân viên (thêm, xóa, sửa thông tin) với bảo mật thông qua đăng nhập admin sử dụng mật khẩu mã hóa SHA-256.
- Triển khai và thử nghiệm hệ thống trong môi trường thực tế.

1.4. Đối tượng và phạm vi nghiên cứu

1.4.1. Đối tượng nghiên cứu

a. Đối tượng chính

Công nghệ nhận diện khuôn mặt và ứng dụng trong hệ thống chấm công tự động cho doanh nghiệp.

b. Đối tượng phụ

- Các thuật toán AI như Mediapipe (phát hiện khuôn mặt), FaceNet (trích xuất đặc trưng), và SVC (phân loại).
- Quy trình quản lý chấm công và dữ liệu nhân viên (lưu trữ trong CSV và MySQL).

1.4.2. Phạm vi nghiên cứu

- Nghiên cứu tập trung vào phát triển hệ thống chấm công sử dụng nhận diện khuôn mặt, không mở rộng sang các ứng dụng khác như an ninh hay thanh toán.
- Sử dụng các công nghệ mã nguồn mở (Mediapipe, TensorFlow, OpenCV) và phần cứng phổ thông (webcam, máy tính nhúng) để đảm bảo chi phí triển khai hợp lý.

1.5. Phương pháp nghiên cứu

- Phương pháp thực nghiệm:

- Thu thập dữ liệu khuôn mặt (với sự đồng ý của người tham gia) để huấn luyện mô hình nhận diện khuôn mặt.
- Phát triển hệ thống sử dụng Python, Mediapipe, FaceNet, TensorFlow, và Tkinter; lưu trữ dữ liệu trong CSV và MySQL.

- Đo lường các chỉ số như độ chính xác nhận diện, thời gian xử lý, và tỷ lệ lỗi khi nhận diện của hệ thống.

1.6. Phân tích đề tài

1.6.1. Bài toán chấm công

Chấm công là quá trình ghi nhận và theo dõi thời gian làm việc của người lao động trong một tổ chức hoặc doanh nghiệp.

Mục đích của chấm công:

- Là cơ sở để tính lương, phụ cấp, thưởng hoặc kỷ luật theo thời gian làm việc thực tế.
- Hỗ trợ bộ phận nhân sự trong việc thống kê ngày công, nghỉ phép, làm thêm giờ, đi muộn – về sớm.
- Giúp doanh nghiệp kiểm soát nhân sự, nâng cao tính minh bạch và hạn chế gian lận trong quản lý thời gian làm việc.
- Tăng hiệu quả quản lý trong môi trường sản xuất có số lượng lao động lớn hoặc làm việc theo ca.

Các thông tin cơ bản cần có khi chấm công bao gồm thông tin của nhân viên như ‘Mã nhân viên’, ‘Tên nhân viên’, ‘Vị trí làm việc’ và ‘Thời gian ghi nhận chấm công (ngày, giờ vào, giờ ra)’.

1.6.2. Công nghệ nhận diện khuôn mặt

Công nghệ nhận diện khuôn mặt là một phương pháp xác định danh tính người dùng thông qua đặc điểm khuôn mặt. Đây là một giải pháp sinh trắc học không tiếp xúc, giúp tự động hóa quá trình xác thực với độ chính xác và tốc độ cao.

Trong hệ thống chấm công, công nghệ này được sử dụng để thay thế các phương pháp truyền thống như quét thẻ hay vân tay, nhằm tăng tính minh bạch, giảm gian lận và nâng cao hiệu quả quản lý nhân sự.

Nhận diện khuôn mặt là quá trình xác định danh tính của một cá nhân bằng cách phân tích các đặc trưng hình học và kết cấu trên khuôn mặt. Quy trình gồm 3 bước chính:

- Xác định vị trí khuôn mặt trong hình ảnh hoặc video
- Trích xuất vector đặc trưng từ khuôn mặt.
- So sánh vector đặc trưng với cơ sở dữ liệu để xác định danh tính.

1.7. Kết luận chương

Như vậy, trong chương này đã trình bày cái nhìn tổng quan về đề tài, nhấn mạnh tính cấp thiết của việc ứng dụng công nghệ nhận diện khuôn mặt trong chấm công cho các doanh nghiệp vừa và nhỏ tại Việt Nam. Các mục tiêu, đối tượng, phạm vi, và phương pháp nghiên cứu đã được xác định rõ ràng, tạo tiền đề cho các phân tích chuyên sâu và phát triển hệ thống trong các chương tiếp theo.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Giới thiệu chương

Trong chương 2 sẽ trình bày cơ sở lý thuyết và kỹ thuật cho hệ thống chăm công ứng dụng nhận diện khuôn mặt, đặt nền tảng cho việc thiết kế và triển khai giải pháp. Nội dung bao gồm tổng quan về phần cứng (Raspberry Pi 4, webcam, ...), các công nghệ nhận diện khuôn mặt (Mediapipe, FaceNet, SVC), phần mềm và dịch vụ (Visual Studio Code, XAMPP, Cloudflare Tunnel) và công cụ phát triển giao diện (Tkinter, PHP).

Các lý thuyết và công nghệ được trình bày gắn liền với hệ thống được phát triển, sử dụng các công cụ mã nguồn mở và phần cứng phổ thông để đảm bảo tính khả thi và chi phí hợp lý cho các doanh nghiệp vừa và nhỏ tại Việt Nam.

2.2. Các công nghệ nền tảng cho nhận diện khuôn mặt

2.2.1. Phát hiện khuôn mặt bằng Mediapipe

MediaPipe là nền tảng mã nguồn mở của Google, cung cấp các giải pháp xử lý thị giác máy tính thời gian thực. Giải pháp **MediaPipe Face Detection** sử dụng thuật toán BlazeFace – một mô hình học sâu nhẹ được tối ưu cho GPU di động [3]. BlazeFace được thiết kế để chạy real-time trên thiết bị di động, đạt tốc độ xử lý lên đến 200–1000+ khung hình/giây (tương đương độ trễ dưới 1 ms) trên các thiết bị hàng đầu [4]. Mô hình này không chỉ phát hiện hộp bao quanh khuôn mặt mà còn dự đoán 6 điểm mốc quan trọng trên khuôn mặt (hai mắt, hai vị trí vành tai, mũi và miệng) cho mỗi khuôn mặt [3] [4]. Hiệu suất siêu nhanh của BlazeFace cho phép nó được dùng trong nhiều ứng dụng thực tế ảo tăng cường (AR) yêu cầu khuôn mặt phải được xác định tức thời trước khi chuyển sang các bước xử lý tiếp theo.

- **Mạng nơ-ron nhẹ:** BlazeFace sử dụng kiến trúc mạng trích xuất đặc trưng nhỏ gọn, lấy cảm hứng từ mạng MobileNetV1/V2 [3]. Thiết kế này giúp giảm số phép tính và phù hợp để chạy real-time trên các thiết bị di động và máy tính cấu hình yếu.
- **SSD (Single Shot Detector):** BlazeFace áp dụng khung phát hiện SSD để suy đoán vị trí hộp bao quanh khuôn mặt và độ tin cậy [3]. Cụ thể, các tầng đầu ra của mạng sẽ dự đoán đồng thời tọa độ khung và điểm tin cậy trong một lần duy nhất, giúp giảm độ phức tạp so với các phương pháp dò tìm theo từng bước.
- **Giới hạn anchor box:** Khác với SSD truyền thống dùng hàng chục anchor box ở nhiều kích thước và tỷ lệ khác nhau, BlazeFace chỉ sử dụng tổng cộng 6 anchor

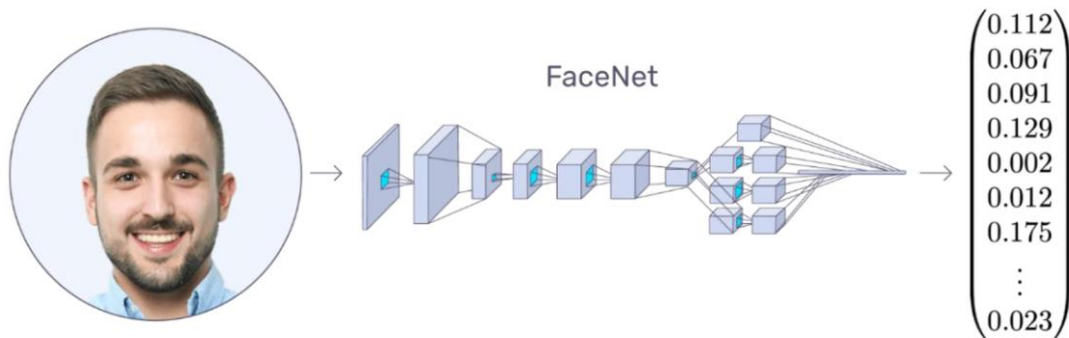
box ở tầng đặc trưng kích thước 8×8 (với tỷ lệ khung cố định 1:1) [4]. Thiết kế này vừa giảm thiểu tính toán vừa duy trì độ chính xác cao do khuôn mặt người thường có tỷ lệ ổn định.

- **Dự đoán điểm đặc trưng:** Mỗi lần phát hiện, BlazeFace trả về 6 điểm mốc trên khuôn mặt (đầu mũi, trung tâm miệng, hai trung tâm mắt và hai vị trí vành tai) [4]. Những điểm mốc này giúp ước lượng chính xác hơn vị trí và hướng của khuôn mặt, hỗ trợ tốt cho các bước xử lý tiếp theo như ước lượng chiều quay 3D của khuôn mặt.

2.2.2. Trích xuất đặc trưng bằng FaceNet

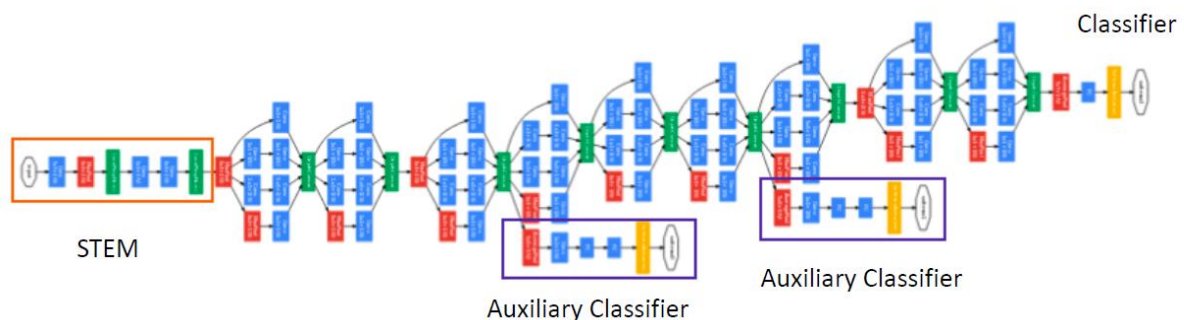
a. Kiến trúc mô hình

FaceNet là một mạng nơ-ron sâu được sử dụng để trích xuất các đặc điểm từ hình ảnh khuôn mặt của một người. Mô hình được xuất bản vào năm 2015 bởi các nhà nghiên cứu Schroff và cộng sự của Google [5].



Hình 2.1 Tạo Vector từ khuôn mặt [5].

Hình 2.1 Mô tả cách FaceNet hoạt động bằng cách lấy hình ảnh từ khuôn mặt người làm input và sẽ trả ra một vector output có 512 giá trị số đại diện cho những nét đặc trưng có trên khuôn mặt. Trong máy học, vector này được gọi là vector nhúng (embedding vector). Lý do các vector được gọi như vậy là vì tất cả các thông tin quan trọng trên khuôn mặt được “nhúng” vào vector này. Theo lý thuyết, vector nhúng của các khuôn mặt thuộc cùng một người sẽ tương đương với nhau [5].



Hình 2.2 Mô hình Inception Resnet V1 [6].

Hình 2.2 Mô tả cách FaceNet được xây dựng trên mô hình Inception Resnet V1. Inception có điểm đặc trưng là có thêm 2 output phụ (auxiliary classifier). Hai output phụ này không ảnh hưởng quá nhiều tới chất lượng của mạng lưới trong khi huấn luyện những epoch đầu. Chúng chỉ giúp việc huấn luyện mô hình diễn ra nhanh hơn bằng cách tối ưu những layer đầu dựa trên các output phụ (trong những epoch đầu). Có thể hiểu một cách khác là trong những epoch đầu, các layer càng về cuối sẽ chưa được tối ưu ngay. Sau một thời gian tối ưu các layer đầu rồi mới tối ưu các layer tiếp theo dựa trên final output. Việc này cải thiện khả năng tính toán và tốc độ huấn luyện lên nhiều lần.

b. Mô hình Pretrain

Hiện nay có

Bảng 2.1 Đánh giá độ chính xác giữa hai mô hình [7].

Model name	LFW accuracy	Training dataset
20180408-102900 (111MB)	0.9905	CASIA-Webface
20180402-114759 (107MB)	0.9965	VGGFace2

Có thể tăng kết quả dự đoán của mô hình nhận diện khuôn mặt bằng cách sử dụng thêm hàm loss function như Triplet Loss và Pairwise Loss trong khi huấn luyện mô hình. Từ đó giúp mô hình đưa các vector nhúng cho khuôn mặt hiệu quả hơn từ đó tăng khả năng nhận diện khuôn mặt chính xác cho mô hình.

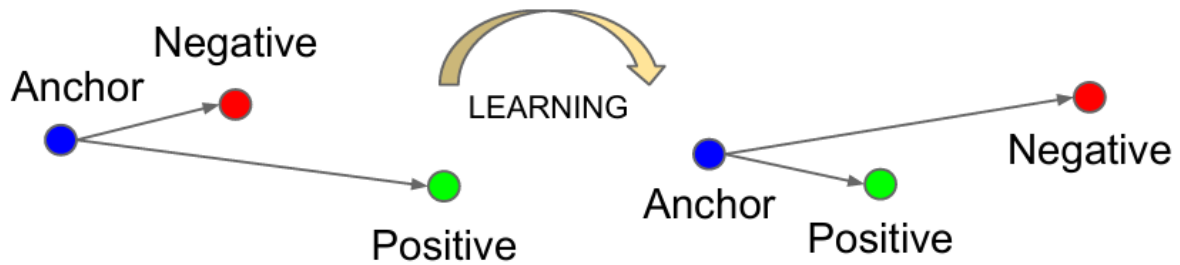
Hàm Triplet Loss là một hàm loss function sử dụng một bộ 3 ảnh để huấn luyện mô hình. Hàm được định nghĩa như sau:

$$L_{triplet} = \sum_{\forall T} \max \left(0, 1 - \frac{\|F(I_R) - F(I_N)\|_2}{\|F(I_R) - F(I_P)\|_2 + m} \right) \quad (2.1)$$

Với $F(I_R)$, $F(I_P)$, và $F(I_N)$ lần lượt là các vector nhúng của các ảnh tham chiếu (I_R), positive (I_P) và negative (I_N). Từ các vector nhúng có thể tính được khoảng cách Euclidean giữa cặp negative $\|F(I_R) - F(I_N)\|_2$ và cặp positive $\|F(I_R) - F(I_P)\|_2$. Giá trị $L_{triplet}$ trong quá trình huấn luyện được tối ưu sao cho tỉ lệ giữa hai khoảng cách này lớn nhất. Đồng nghĩa với việc khoảng cách giữa cặp positive sẽ giảm đi, trong khi khoảng cách giữa cặp negative sẽ được tăng lên.

Tuy nhiên sau khi huấn luyện bằng hàm Triplet Loss, mặc dù khoảng cách giữa các cặp dữ liệu được nằm trong một khoảng tỉ lệ giá trị nhất định, nhưng giá trị khoảng cách tuyệt đối giữa chúng lại không hề đồng nhất. Vì vậy cần dùng thêm hàm Pairwise Loss để tối thiểu toàn bộ khoảng cách giữa các cặp positive đã được chọn trong hàm Triplet Loss T.

$$L_{pairs} = \sum_{(I_R, I_P) \in T} \|F(I_R) - F(I_P)\|_2^2 \quad (2.2)$$



Hình 2.3 Thuật toán Triplet loss mô phỏng quá trình nhận diện [8].

Hình 2.3 Mô tả sự thay đổi khoảng cách giữa các cặp dữ liệu sau khi sử dụng hai hàm loss function trong quá trình huấn luyện mô hình. Quá trình huấn luyện sẽ tối thiểu khoảng cách tuyệt đối giữa các cặp vector positive này. Từ đó giới hạn lại được khoảng cách tuyệt đối giữa chúng.

c. Sử dụng mô hình để trích xuất đặc trưng

Để chương trình có thể nhận diện được khuôn mặt hiệu quả nhất, ta cần có một cụm vector nhúng của khuôn mặt ở nhiều vị trí góc độ khác nhau bằng cách quay lại video. Để tăng khả năng nhận diện khuôn mặt chính xác của mô hình, ta có thể căn chỉnh thẳng hàng lại các ảnh. Sau khi tiền xử lý xong khuôn mặt, tiếp theo ta sẽ mở mô hình bằng pretrain FaceNet bằng TensorFlow Sau đó ta khai báo input cho mạng lưới, chuyển đổi ảnh thành các vector nhúng và lưu vào file pickle.

2.2.3. Phân loại và nhận diện khuôn mặt bằng thuật toán SVM

Để nhận diện được khuôn mặt, ta cần trích xuất các đặc trưng của khuôn mặt thành một vector nhúng. Từ vector nhúng đó ta có thể nhận diện danh tính khuôn mặt chủ yếu thông qua hai cách dưới đây:

- Dùng các thuật toán phân loại như KNN, SVM hoặc Random Forest.
- So sánh khoảng cách Euclidean, Manhattan, Cosine, ... giữa hai vector nhúng.

Sử dụng phương pháp so sánh khoảng cách sẽ tốn nhiều thời gian hơn vì cần phải thực hiện tính toán với tất cả các vector nhúng đã được xác định sẵn. Phương pháp sử dụng thuật toán phân loại sẽ tốn ít tài nguyên hơn do chỉ cần xác định vị trí của vector nhúng mới nằm ở trong cụm nào. Tuy vậy phương pháp phân loại cần phải xác định biên giới cho từng cụm vector nhúng sao cho các vector này ít nằm ở sai cụm càng tốt. Trên thực tế cần rất nhiều thời gian cũng như chất lượng đầu vào tốt để xác định được ranh giới này tối ưu và hiệu quả nhất.

Vì vậy đối với trường hợp số lượng danh tính cần nhận diện với số lượng vừa và nhỏ, ta sẽ sử dụng phương pháp so sánh khoảng cách. Với trường hợp danh tính được lưu với số lượng lớn, ta sẽ sử dụng phương pháp phân loại để tăng tốc thời gian xử lý.

Sau khi trích xuất được vector nhúng (embedding vector) từ khuôn mặt bằng FaceNet, bước tiếp theo trong quy trình nhận diện khuôn mặt là phân loại vector này để xác định danh tính (identity) của người dùng.

Trong chương trình, Mô hình SVC được dùng để triển khai thuật toán sử dụng để thực hiện nhiệm vụ phân loại. SVM là một thuật toán học máy giám sát (supervised machine learning) mạnh mẽ, được ứng dụng rộng rãi trong các bài toán phân loại, bao gồm nhận diện khuôn mặt.

SVM hoạt động bằng cách tìm một siêu phẳng (hyperplane) tối ưu trong không gian đa chiều để tách biệt các lớp dữ liệu (ở đây là các danh tính khác nhau). Trong trường hợp nhận diện khuôn mặt, mỗi danh tính (ID) được xem như một lớp (class), và vector nhúng 512 chiều từ FaceNet là dữ liệu đầu vào để phân loại.

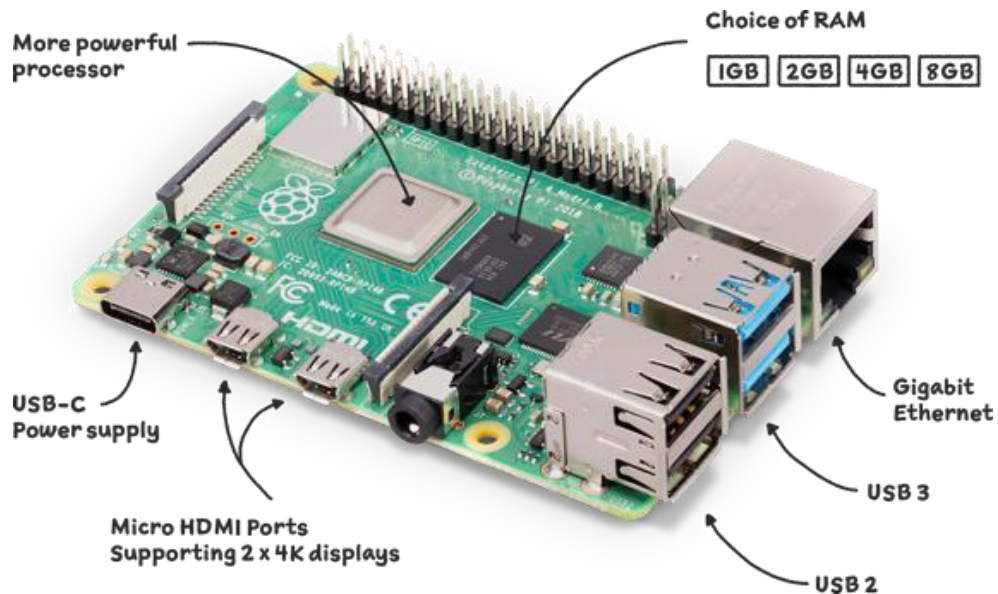
Mô hình SVC được huấn luyện trên dữ liệu khuôn mặt đã căn chỉnh, lưu vào file facemodel.pkl, và được tải lại khi nhận diện.

2.3. Tổng quan về phần cứng

2.3.1. Raspberry Pi 4 Model B

Raspberry Pi là một máy tính mini có kích thước nhỏ gọn nhưng đầy đủ chức năng, được thiết kế để phục vụ cho giáo dục, nghiên cứu và phát triển các ứng dụng công nghệ, phù hợp cho các ứng dụng AI và xử lý hình ảnh.

Raspberry Pi 4 được chọn nhờ chi phí thấp (khoảng 50-70 USD), khả năng mở rộng, và cộng đồng hỗ trợ lớn. Hệ điều hành Raspbian (dựa trên Linux) cung cấp môi trường phát triển thân thiện, tích hợp dễ dàng với Python và các thư viện AI, khả năng lập trình linh hoạt và dễ tích hợp với các thiết bị phần cứng như cảm biến, camera, màn hình, ... Raspberry Pi đã trở thành lựa chọn phổ biến trong nhiều lĩnh vực như IoT, tự động hóa, điều khiển nhúng, và trí tuệ nhân tạo. Thiết bị này có thể hoạt động liên tục, tiêu thụ điện năng thấp và dễ dàng triển khai trong các dự án thực tế.



Hình 2.4 Raspberry Pi 4 Model B [9].

Hình 2.4 Mô tả Raspberry Pi 4 Model B - Máy tính đơn board (SBC – Single Board Computer) ra mắt vào tháng 6 năm 2019.

Thông số kỹ thuật của Raspberry Pi 4 Model B dùng trong đề tài:

- Bộ vi xử lý: Broadcom BCM2711, 4 nhân Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- RAM: 4GB
- Kết nối không dây: Wi-Fi băng tần kép 2.4GHz và 5.0GHz IEEE 802.11ac, Bluetooth 5.0, BLE
- Cổng mạng: Gigabit Ethernet
- Cổng USB: 2 cổng USB 3.0 và 2 cổng USB 2.0
- Cổng hiển thị: 2 cổng Micro HDMI hỗ trợ độ phân giải lên tới 4K
- Cổng giao tiếp khác: MIPI DSI, MIPI CSI, cổng AV 4 chân
- Đồ họa: Hỗ trợ H.265 (4Kp60 decode), H.264 (1080p60 decode, 1080p30 encode), OpenGL ES 3.0
- Lưu trữ: Khe cắm thẻ MicroSD cho hệ điều hành và dữ liệu
- Nguồn điện: 5V – 3A DC qua cổng USB-C hoặc qua chân GPIO (tối thiểu 3A)

Với hiệu năng ổn định, khả năng tương thích cao và dễ dàng tích hợp vào các hệ thống thực tế, Raspberry Pi 4 Model B hoàn toàn đáp ứng được yêu cầu của đề tài chăm công bằng nhận diện khuôn mặt. Thiết bị này không chỉ giúp đơn giản hóa quá trình triển khai, mà còn đảm bảo tính linh hoạt, tiết kiệm và hiệu quả trong quá trình vận hành lâu dài.

2.3.2. Webcam Camera HD 1080P

Webcam là thiết bị ghi hình video được sử dụng phổ biến trong nhiều lĩnh vực như hội nghị trực tuyến, học tập từ xa, livestream và giám sát an ninh. Thiết bị này giúp truyền tải hình ảnh và âm thanh qua máy tính hoặc các thiết bị điện tử khác một cách dễ dàng và tiện lợi.



Hình 2.5 Webcam Camera HD 1080P

Hình 2.5 Mô tả Webcam Camera HD 1080P là thiết bị ghi hình với độ phân giải Full HD, mang đến hình ảnh sắc nét và chất lượng cao. Thiết bị có thiết kế nhỏ gọn, dễ dàng kết nối qua cổng USB, cho phép truyền tải hình ảnh mượt mà và chân thực. Webcam còn tích hợp micro thu âm chất lượng, giúp đảm bảo âm thanh rõ ràng trong quá trình sử dụng.

Thông số kỹ thuật của Webcam Camera HD 1080P dùng trong đề tài:

- Độ phân giải video: Full HD 1920×1080 pixel
- Tốc độ khung hình: 30 fps
- Ống kính: Thủy tinh nhiều lớp, giúp tăng độ sắc nét
- Góc nhìn: 70° – 90°
- Lấy nét: Cố định hoặc tự động (tùy mẫu)
- Hiệu chỉnh ánh sáng: Tự động cân bằng trắng và điều chỉnh ánh sáng
- Kết nối: USB-A
- Tương thích hệ điều hành: Windows, macOS, Linux
- Tính năng bổ sung: Nắp che bảo vệ quyền riêng tư, chân đế xoay linh hoạt

- Kích thước: Khoảng 95×68×72 mm
- Trọng lượng: Khoảng 130g

Các tính năng như độ phân giải Full HD, góc nhìn rộng, cùng khả năng tự động điều chỉnh ánh sáng giúp nâng cao độ chính xác trong quá trình nhận dạng. Kết nối USB tiện lợi và tính tương thích với nhiều hệ điều hành cũng tạo điều kiện thuận lợi cho việc tích hợp và vận hành hệ thống trong thực tế.

2.3.3. Màn hình FPT Elead F18RW

Màn hình FPT Elead F18RW là sản phẩm màn hình rời được thiết kế dành cho các ứng dụng văn phòng và công nghiệp nhẹ. Với kích thước vừa phải, thiết bị này đáp ứng tốt nhu cầu hiển thị hình ảnh rõ nét và màu sắc trung thực, phục vụ các hệ thống tự động hóa như chấm công nhận diện khuôn mặt. Sản phẩm có thiết kế nhỏ gọn, giúp tiết kiệm không gian lắp đặt, đồng thời dễ dàng kết nối với các thiết bị xử lý trung tâm qua cổng HDMI.



Hình 2.6 Màn hình FPT Elead F18RW

Hình 2.6 là màn hình FPT Elead F18RW. Dưới đây là một số thông số chính của nó:

- Kích thước màn hình: 18.5 inch
- Loại màn hình: LED
- Độ phân giải: 1366 x 768 pixels
- Tỷ lệ khung hình: 16:9
- Độ sáng: 250 cd/m²
- Thời gian đáp ứng: 5 ms
- Cổng kết nối: 1 x VGA, 1 x HDMI

- Nguồn điện sử dụng: 12V – 2.5A (nguồn rời)

Trong đề tài, màn hình FPT Elead F18RW đảm nhận vai trò thiết bị hiển thị chính. Màn hình giúp người dùng quan sát hình ảnh nhận diện khuôn mặt trực tiếp, đồng thời cung cấp các thông báo kết quả chấm công như thành công hoặc lỗi nhận diện. Ngoài ra, màn hình còn hỗ trợ người quản trị thực hiện các chức năng khác của hệ thống như thêm, xoá, sửa thông tin cho nhân viên. Đây là sản phẩm có giá thành hợp lý, dễ bảo trì và thay thế, phù hợp với ngân sách cũng như yêu cầu vận hành lâu dài của dự án.

2.3.4. Bàn phím cơ Biojee

Bàn phím cơ Biojee là sản phẩm bàn phím chất lượng cao, thường được sử dụng trong các hệ thống cần thao tác nhập liệu nhanh, chính xác và bền bỉ như hệ thống chấm công hoặc điều khiển máy tính công nghiệp. Với thiết kế chắc chắn và phím cơ cho phản hồi tốt, bàn phím giúp người dùng thao tác thuận tiện, giảm mỏi tay khi sử dụng liên tục trong thời gian dài.



Hình 2.7 Bàn phím cơ Biojee

Hình 2.7 Bàn phím cơ Biojee với thông số kỹ thuật:

- Kiểu bàn phím: Bàn phím cơ (Mechanical Keyboard)
- Loại switch: Blue switch (hoặc tương tự, tùy model)
- Số phím: 104 phím chuẩn full-size
- Kết nối: USB có dây
- Chất liệu keycap: ABS
- Độ bền phím: Trên 50 triệu lần nhấn
- Kích thước: Khoảng 44 x 13 x 3.8 cm

Trong đề tài, bàn phím cơ Biojee được dùng để nhập liệu thủ công khi cần thiết, hỗ trợ người quản lý trong việc cấu hình phần mềm, thêm hoặc chỉnh sửa thông tin người dùng, cũng như xử lý các trường hợp ngoại lệ khi nhận diện khuôn mặt gặp lỗi. Bàn phím cơ cho độ bền cao, đáp ứng nhanh và phản hồi rõ ràng giúp tăng hiệu suất làm việc của nhân viên vận hành hệ thống.

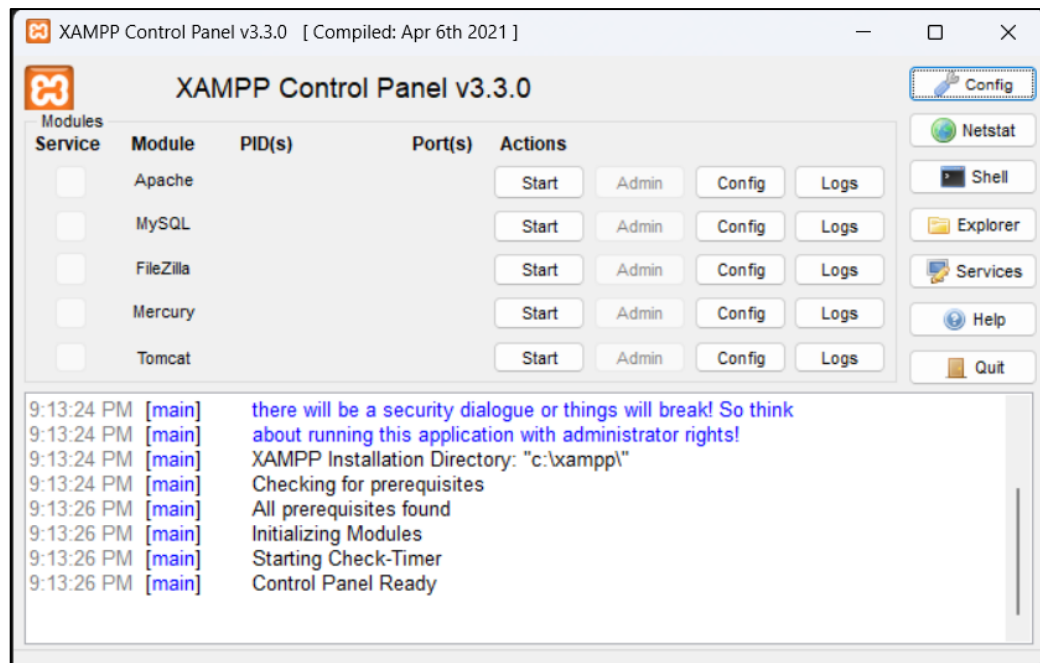
2.4. Các phần mềm và dịch vụ sử dụng trong hệ thống

2.4.1. XAMPP

XAMPP là một phần mềm máy chủ web mã nguồn mở, được phát triển bởi Apache Friends. Tên gọi "XAMPP" là viết tắt của các thành phần chính:

- X: Hệ điều hành bất kỳ
- A: Apache – máy chủ web
- M: MySQL – hệ quản trị cơ sở dữ liệu
- P: PHP – ngôn ngữ lập trình phía máy chủ
- P: Perl – ngôn ngữ lập trình đa mục đích

XAMPP được thiết kế nhằm mục đích đơn giản hóa quá trình cài đặt môi trường máy chủ cục bộ, hỗ trợ các nhà phát triển web và lập trình viên xây dựng, kiểm thử và triển khai ứng dụng một cách nhanh chóng mà không cần cấu hình thủ công nhiều thành phần riêng lẻ.



Hình 2.8 Bảng điều khiển XAMPP

Hình 2.8 là giao diện bảng điều khiển của XAMPP – một giao diện đơn giản, trực quan, dễ thao tác. Các ưu điểm chính của XAMPP:

- Gói tích hợp đầy đủ các thành phần cần thiết chỉ với một trình cài đặt duy nhất.
- Giao diện điều khiển trực quan cho phép quản lý và giám sát các dịch vụ như Apache, MySQL, FileZilla, Tomcat, ...
- Tương thích với Windows, macOS và Linux.

- Tạo môi trường localhost để phát triển ứng dụng web mà không cần máy chủ thật.
- Cung cấp phpMyAdmin để quản lý MySQL hoặc MariaDB thông qua giao diện đồ họa.
- Hỗ trợ đầy đủ các phiên bản PHP hiện đại, phù hợp cho phát triển web động và xử lý phía máy chủ.

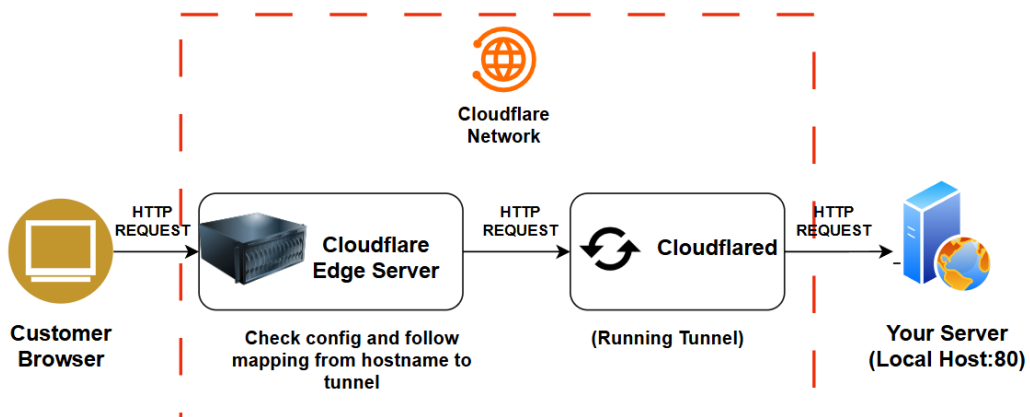
Trong đề tài, XAMPP được sử dụng làm nền tảng máy chủ cục bộ để tiếp nhận và lưu trữ dữ liệu điểm danh. Thành phần Apache trong XAMPP đóng vai trò là máy chủ web, giúp nhận dữ liệu từ thiết bị nhận diện khuôn mặt gửi đến. Trong khi đó, MySQL được dùng để xây dựng cơ sở dữ liệu lưu trữ thông tin người dùng và thời gian điểm danh. Việc sử dụng XAMPP giúp hệ thống hoạt động ổn định, dễ quản lý và thuận tiện trong việc phát triển, kiểm thử trước khi triển khai thực tế.

2.4.2. Cloudflare Tunnel

Cloudflare Tunnel là một dịch vụ do Cloudflare cung cấp, cho phép kết nối an toàn giữa máy chủ cục bộ và Internet mà không cần mở cổng hay cấu hình địa chỉ IP công khai. Thông qua công cụ cloudflared, người dùng có thể dễ dàng tạo một đường hầm (tunnel) từ máy chủ nội bộ đến mạng lưới của Cloudflare, từ đó truy cập dịch vụ nội bộ qua một tên miền an toàn được bảo vệ bởi Cloudflare.

Các đặc điểm nổi bật:

- Không cần cấu hình mở cổng trên router giúp giảm thiểu rủi ro bảo mật liên quan đến việc chuyển tiếp cổng hoặc việc công khai địa chỉ IP nội bộ ra Internet.
- Chỉ cần cài đặt cloudflared và cấu hình vài dòng lệnh là có thể đưa ứng dụng cục bộ ra Internet.
- Có thể sử dụng tên miền riêng được quản lý bởi Cloudflare để truy cập dịch vụ nội bộ.
- Hỗ trợ duy trì kết nối ổn định, tự động phục hồi khi có sự cố.



Hình 2.9 Sơ đồ quy trình hoạt động của Cloudflare Tunnel

Hình 2.9 Mô tả cách hoạt động của Cloudflare Tunnel: Người dùng truy cập ứng dụng qua một địa chỉ duy nhất, tại đó Cloudflare Edge Server sẽ kiểm tra và xác thực yêu cầu, đồng thời lọc bỏ các mối nguy cơ tấn công hoặc truy cập trái phép. Sau đó, yêu cầu được chuyển tiếp an toàn qua Tunnel đến máy chủ nội bộ nơi ứng dụng đang chạy. Máy chủ xử lý yêu cầu và gửi phản hồi trở lại qua Tunnel, rồi Cloudflare chuyển tiếp phản hồi đến người dùng.

Trong đề tài, Cloudflare Tunnel được sử dụng để xuất bản máy chủ web cục bộ (chạy bằng XAMPP) ra Internet, giúp người quản lý có thể truy cập trang quản lý điểm danh từ xa một cách an toàn và tiện lợi. Giải pháp này đặc biệt hữu ích khi hệ thống được triển khai trên Raspberry Pi trong mạng nội bộ không có IP tĩnh hoặc không thể cấu hình router.

2.4.3. Email OTP

Email OTP (One-Time Password) là một phương pháp bảo mật hiện đại được sử dụng để xác minh danh tính người dùng thông qua việc gửi mã xác thực dùng một lần đến địa chỉ email đã đăng ký. Mã OTP này giúp người dùng hoàn tất quá trình đăng nhập hoặc thực hiện các giao dịch quan trọng một cách an toàn và hiệu quả.



Hình 2.10 Dịch vụ xác thực OTP

Hình 2.10 Mô tả về dịch vụ xác thực OTP. Khi người dùng yêu cầu xác thực, ví dụ như đăng nhập tài khoản, hệ thống sẽ tự động tạo ra một mã OTP duy nhất, ngẫu nhiên và có thời hạn sử dụng. Mã này sau đó được gửi trực tiếp đến địa chỉ email mà người dùng đã đăng ký trước đó. Người dùng sẽ nhập mã OTP này vào hệ thống để hoàn tất quá trình xác thực, đảm bảo chỉ những người sở hữu email hợp lệ mới có thể truy cập vào tài khoản hoặc thực hiện giao dịch.

Công nghệ Email OTP mang lại nhiều lợi ích nổi bật. Trước hết, mã OTP là duy nhất và có thời hạn, điều này làm cho việc giả mạo hoặc sử dụng lại mã trở nên rất khó khăn, nâng cao mức độ bảo mật cho hệ thống. Thứ hai, phương thức này rất tiện lợi khi người dùng chỉ cần truy cập vào hộp thư email của mình để nhận mã, giúp quá trình xác thực

diễn ra nhanh chóng và dễ dàng. Ngoài ra, Email OTP thường được kết hợp cùng với mật khẩu trong xác thực hai yếu tố, từ đó tăng cường mức độ an toàn cho tài khoản và dữ liệu cá nhân.

Trong đề tài, Email OTP được ứng dụng để tăng cường bảo mật khi người dùng thực hiện thao tác đổi mật khẩu trên cả giao diện chăm công và trang web quản lý. Khi người dùng yêu cầu đổi mật khẩu, hệ thống sẽ gửi một mã OTP duy nhất đến địa chỉ email đã đăng ký của họ. Người dùng phải nhập chính xác mã này mới có thể hoàn tất việc thay đổi mật khẩu. Việc này giúp ngăn chặn các hành vi thay đổi mật khẩu trái phép, bảo vệ tài khoản người dùng khỏi bị xâm nhập và đảm bảo an toàn dữ liệu cá nhân

2.5. Các công cụ phát triển giao diện

2.5.1. Tkinter

Tkinter là một thư viện lập trình giao diện người dùng đồ họa (GUI - Graphical User Interface) được tích hợp sẵn trong ngôn ngữ lập trình Python. Nó sẽ giúp người dùng tương tác với chương trình thông qua các thành phần đồ họa (widget) như nút bấm, ô nhập liệu, nhãn văn bản và nhiều widget khác.

Tkinter hỗ trợ xây dựng các ứng dụng có giao diện trực quan, thân thiện, dễ sử dụng với nhiều thành phần đa dạng như:

- **Ô nhập liệu (Entry)** để người dùng nhập dữ liệu.
- **Nhãn văn bản (Label)** dùng để hiển thị thông tin hoặc hướng dẫn.
- **Nút nhấn (Button)** để kích hoạt các hành động như lưu dữ liệu, điều khiển chương trình.
- **Khung chứa (Frame)** dùng để nhóm và tổ chức các widget trong giao diện.

Với khả năng bố trí linh hoạt thông qua các phương pháp như pack(), grid() và place(), Tkinter giúp thiết kế giao diện phù hợp với nhiều loại ứng dụng khác nhau. Thư viện này cũng tương thích tốt với các thư viện Python khác, giúp dễ dàng kết nối với cơ sở dữ liệu, xử lý hình ảnh hoặc các thiết bị ngoại vi.

Trong đề tài, Tkinter được chọn làm công cụ phát triển giao diện người dùng nhằm cung cấp môi trường quản lý và điều khiển thuận tiện, giúp người dùng dễ dàng thao tác và giám sát quá trình chăm công. Nhờ đó, việc vận hành hệ thống trở nên hiệu quả và thân thiện hơn.

2.5.2. Ngôn ngữ lập trình PHP

PHP là một ngôn ngữ lập trình kịch bản phía máy chủ, mã nguồn mở và miễn phí, được sử dụng rộng rãi để phát triển các ứng dụng web động. PHP tương thích đa nền tảng, chạy hiệu quả trên các hệ điều hành phổ biến như Windows, MacOS và Linux.

PHP chủ yếu được dùng để tạo ra các trang web có nội dung động, quản lý dữ liệu người dùng qua form, xử lý file trên server, thao tác với cookies, cũng như tương tác với cơ sở dữ liệu như MySQL. PHP có thể chạy dưới dạng mô-đun của máy chủ web hoặc sử dụng CGI, giúp nó linh hoạt trong việc triển khai trên nhiều môi trường khác nhau.

Ưu điểm của PHP trong phát triển web:

- PHP có khả năng nhúng trực tiếp vào mã HTML, giúp tạo giao diện người dùng linh hoạt, tương tác và dễ dàng tùy chỉnh.
- Có cú pháp đơn giản, rõ ràng, giúp rút ngắn thời gian phát triển ứng dụng và nâng cao hiệu quả công việc.
- Tương thích cao với hầu hết các trình duyệt web phổ biến, đảm bảo trải nghiệm người dùng mượt mà và ổn định.
- Hiệu suất xử lý nhanh, phù hợp cho các ứng dụng web có lượng truy cập lớn và quản lý dữ liệu trực tuyến hiệu quả.
- Dễ dàng tùy biến và mở rộng nhờ mã nguồn mở cùng với cộng đồng phát triển rộng lớn, giàu tài nguyên hỗ trợ.
- Tích hợp tốt với các hệ quản trị cơ sở dữ liệu phổ biến như MySQL, giúp quản lý và xử lý dữ liệu hiệu quả.

PHP được lựa chọn cho đề tài nhờ khả năng nhúng linh hoạt vào HTML, tạo nên giao diện thân thiện và dễ tương tác. Với tính tương thích cao, hiệu suất xử lý nhanh và tính mở rộng vượt trội từ mã nguồn mở, PHP đáp ứng tốt yêu cầu phát triển ứng dụng quản lý dữ liệu. Đồng thời, các cơ chế bảo mật đa dạng cùng sự tích hợp hiệu quả với MySQL giúp bảo đảm an toàn và ổn định cho hệ thống.

2.6. Kết luận chương

Như vậy, trong chương này đã trình bày các cơ sở lý thuyết và công nghệ nền tảng cho hệ thống chăm công ứng dụng nhận diện khuôn mặt. Những kiến thức được đề cập không chỉ giúp định hướng thiết kế tổng thể mà còn đảm bảo tính khả thi trong quá trình triển khai thực tế. Đây chính là nền tảng quan trọng để triển khai, kiểm thử và đánh giá toàn diện giải pháp trong các chương sau.

CHƯƠNG 3: THIẾT KẾ VÀ TRIỂN KHAI

3.1. Giới thiệu chương

Chương 3 trình bày quá trình tính toán và thiết kế hệ thống chấm công ứng dụng nhận diện khuôn mặt, chuyển từ nền tảng lý thuyết (Chương 2) sang triển khai thực tế. Nội dung bao gồm các yêu cầu hệ thống, kiến trúc tổng thể, thiết kế khối thiết bị và thiết kế khối Web App (giao diện web, cơ sở dữ liệu, và Cloudflare Tunnel). Các thiết kế được xây dựng dựa trên mã nguồn thực tế, bao gồm ứng dụng cục bộ (Python, Tkinter) và giao diện web (PHP, XAMPP).

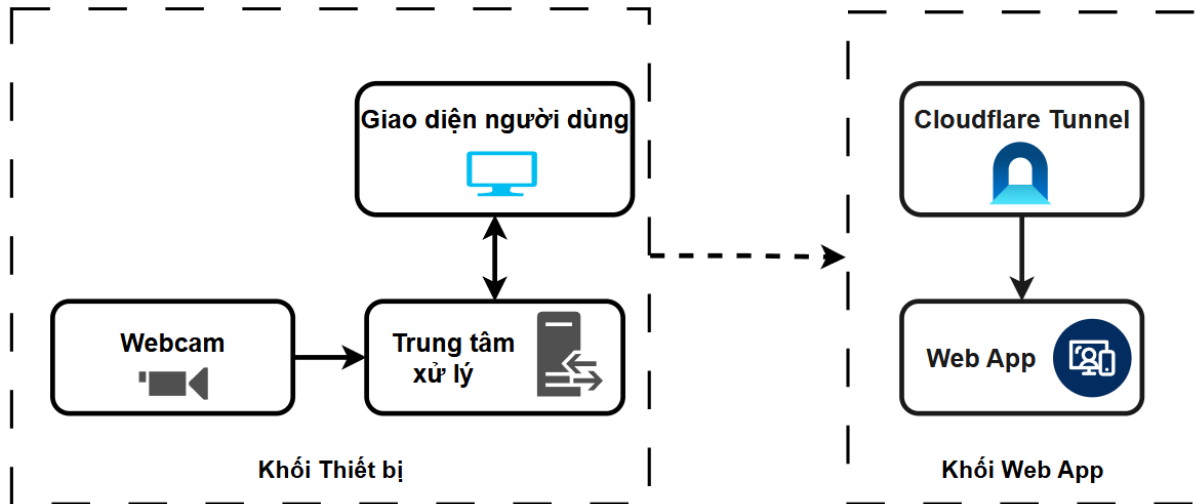
3.2. Các yêu cầu về tính năng và kỹ thuật

Hệ thống chấm công được thiết kế nhằm thay thế các phương pháp truyền thống bằng một giải pháp tự động, nhanh chóng và chính xác, sử dụng công nghệ nhận diện khuôn mặt. Hệ thống cần đáp ứng các yêu cầu về tính năng và kỹ thuật như sau:

- Nhận diện khuôn mặt theo thời gian thực từ webcam, MediaPipe để phát hiện và FaceNet để trích đặc trưng, kết hợp thuật toán SVC để nhận diện và phân loại.
- Xác suất nhận diện cần đạt trên 80% để được ghi nhận chấm công.
- Ghi nhận thông tin chấm công (mã nhân viên, họ tên, chức vụ, thời gian) vào cả MySQL và file CSV, tránh trùng lặp bằng cách kiểm tra khoảng cách thời gian giữa các lần chấm công.
- Hệ thống có khả năng tự kết nối lại cơ sở dữ liệu khi bị gián đoạn và xử lý đa luồng.
- Đồng bộ dữ liệu giữa CSV và MySQL để đảm bảo toàn vẹn và tránh mất mát.
- Giao diện thân thiện (Tkinter), chia thành các tab: Chấm công, Thêm/Xoá/Sửa nhân viên và Đổi mật khẩu, có hiển thị ảnh và thông tin nhận diện.
- Cho phép linh hoạt mở rộng (thêm/xoá/sửa nhân viên, huấn luyện lại mô hình).
- Các chức năng quản trị yêu cầu đăng nhập bằng mật khẩu mã hóa SHA-256 (lưu trong JSON).
- Hỗ trợ gửi mã OTP qua email khi đổi mật khẩu để tăng bảo mật.

3.3. Sơ đồ khối chức năng

Sơ đồ khối chức năng được xây dựng để mô tả các yêu cầu về tính năng và kỹ thuật cho hệ thống, bao gồm 2 khối chính là ‘Khối thiết bị’ và ‘Khối Web App’.



Hình 3.1 Sơ đồ khối của hệ thống

Hình 3.1 Mô tả sơ đồ khối của hệ thống chăm công sử dụng công nghệ nhận diện khuôn mặt. Camera có nhiệm vụ ghi lại hình ảnh khuôn mặt của nhân viên và truyền về trung tâm xử lý, nơi thực hiện nhận diện và lưu trữ dữ liệu chăm công. Giao diện đồ họa được giúp người quản lý dễ dàng thao tác như thêm, xóa nhân viên hoặc theo dõi kết quả nhận diện. Dữ liệu sau khi xử lý cũng được đồng bộ lên Web App, cho phép truy cập từ xa thông qua kết nối bảo mật Cloudflare Tunnel.

3.4. Thiết kế và xây dựng mô hình huấn luyện

3.4.1. Mô hình trích xuất đặc trưng FaceNet:

Trong Đồ án này, nhóm sử dụng Pretrained model FaceNet (20180402-114759.pb). Là một trong những pretrained model FaceNet phổ biến, được phát triển bởi nhóm nghiên cứu của Google và được cộng đồng tái sử dụng rộng rãi (bao gồm dự án facenet-pytorch). Dưới đây là các thông tin chi tiết về mô hình này, dựa trên tài liệu gốc và nguồn mở:

3.1.4.1. Nguồn gốc và huấn luyện:

FaceNet được giới thiệu bởi nhóm nghiên cứu tại Google trong bài báo "FaceNet: A Unified Embedding for Face Recognition and Clustering" (Schroff et al., 2015, CVPR).

Mô hình 20180402-114759.pb được huấn luyện trên tập dữ liệu VGGFace2 (3.31 triệu ảnh và 9131 danh tính) hoặc CASIA-WebFace (453,453 ảnh và 10,575 danh tính).

Mô hình sử dụng hàm mất mát Triplet Loss, giúp tối ưu hóa khoảng cách giữa các embedding: các khuôn mặt cùng danh tính được kéo gần nhau, các khuôn mặt khác danh tính được đẩy xa nhau trong không gian Euclidean.

3.4.1.2. Cấu trúc mô hình:

FaceNet dựa trên kiến trúc mạng Inception-ResNet-v1.

Số lượng tham số: Khoảng 7.5 triệu tham số

Đầu ra: Mô hình tạo ra embedding 512 chiều

3.4.1.3. Độ chính xác:

Đánh giá trên LFW trong quá trình đào tạo (Labeled Faces in the Wild), FaceNet đạt độ chính xác $99.65\% \pm 0.252\%$.

Độ chính xác này được đo lường trên các cặp ảnh (giống hoặc khác nhau), sử dụng khoảng cách Euclidean giữa các embedding.

Mô hình không được huấn luyện lại trong Đồ án này, do pretrained model đã được tối ưu hóa cho bài toán nhận diện khuôn mặt và đủ đáp ứng yêu cầu của hệ thống chăm công.

3.4.1.4. Ưu điểm:

Độ chính xác cao, hoạt động tốt với các góc độ và điều kiện ánh sáng khác nhau.

Sử dụng pretrained model giúp tiết kiệm thời gian và tài nguyên, không cần huấn luyện lại.

3.4.2. Mô hình phân loại SVC:

Mô hình SVC được dùng để triển khai thuật toán SVM, sau khi cắt và resize khuôn mặt, ảnh đó được truyền vào mô hình FaceNet để lấy vector và SVC chỉ xử lý những vector 512 chiều do FaceNet tạo ra.

Mô hình SVC là một mô hình học máy cổ điển (classical ML). SVC không học lại từ ảnh gốc – nó chỉ học cách phân biệt các vector đầu ra từ FaceNet.

Mô hình Pretrained được huấn luyện trên các dữ liệu lên tới hàng triệu ảnh, do đó khả năng mã hóa rất tốt các bức ảnh trên không gian 512 chiều. Vậy trong Đồ án này sẽ sử dụng lại mô hình Pretrained, tính toán embedding vector và huấn luyện embedding vector bằng thuật toán SVM để phân loại các khuôn mặt.

3.4.3. Các thư viện cần thiết

- **tensorflow**: Tải và chạy mô hình FaceNet đã huấn luyện trước (file 20180402-114759.pb) để trích xuất vector đặc trưng (embedding) từ ảnh khuôn mặt.
- **cv2 (OpenCV)**: Đọc ảnh từ file (cv2.imread), chuyển đổi định dạng màu, resize ảnh, và xử lý ảnh đầu vào trước khi đưa vào FaceNet.
- **mediapipe**: Phát hiện khuôn mặt trong ảnh đầu vào (sử dụng mô hình **FaceDetection**) để xác định vùng cần crop.
- **sklearn.svm.SVC**: Khởi tạo và huấn luyện mô hình **Support Vector Classifier (SVC)** trên tập vector embedding để phân loại khuôn mặt.

- **sklearn.preprocessing.LabelEncoder**: Mã hóa tên người (dạng chuỗi) thành các số nguyên để mô hình SVC có thể huấn luyện và dự đoán; đồng thời dùng để giải mã ngược lại kết quả.
- **numpy**: Xử lý mảng số liệu (array) như ảnh sau khi chuyển đổi, embedding vector, v.v. Hỗ trợ tính toán hiệu quả với dữ liệu lớn.
- **os**: Duyệt thư mục chứa ảnh khuôn mặt, hỗ trợ thao tác file, đường dẫn.
- **pickle**: Tuân tự hóa (serialize) và lưu mô hình SVC cùng LabelEncoder vào file .pkl để có thể load lại khi cần mà không huấn luyện lại từ đầu.

3.4.4. Thu thập và tiền xử lý dữ liệu

3.4.4.1. Mô tả dữ liệu

Trong Đồ án này Mô hình SVC sử dụng dữ liệu hình ảnh khuôn mặt ít nhất từ 2 người trở lên để thực hiện việc phân loại, với mỗi người thì cần ít nhất 30 ảnh được lấy từ các góc mặt nghiêng trái/phải, đưa mặt lên/xuống, thay đổi các góc nhìn của mắt, hình dáng khuôn miệng, ...v.v...

3.4.4.2. Tiền xử lý

Tiền xử lý dữ liệu là một trong những bước quan trọng nhất giúp nâng cao hiệu quả của bất kỳ vấn đề phân loại nào, vì nó giúp chuyển đổi dữ liệu thành các định dạng lý tưởng và dễ hiểu hơn.

Chuẩn hóa ảnh đầu vào (kích thước, màu sắc, vùng mặt) để đảm bảo đầu vào có chất lượng tốt, đồng nhất, đảm bảo FaceNet nhận ảnh đúng định dạng, cải thiện độ chính xác của quá trình trích xuất đặc trưng và phân loại.

Dựa trên bounding box, cắt phần khuôn mặt ra khỏi ảnh gốc để loại bỏ nền thừa và nhiễu không cần thiết, điều chỉnh ảnh khuôn mặt về kích thước chuẩn 160x160 pixel, phù hợp với đầu vào của mô hình FaceNet, tính và chuẩn hóa toàn bộ pixel về trung bình 0, độ lệch chuẩn 1, chuyển ảnh sang kiểu float32 và mở rộng chiều để có dạng [1, 160, 160, 3] – đúng định dạng đầu vào của TensorFlow.

3.4.5. Xây dựng và huấn luyện mô hình

3.4.5.1. Xây dựng mô hình

- Duyệt dữ liệu ảnh khuôn mặt được lưu theo từng thư mục đại diện cho mỗi người
- Tiền xử lý ảnh:
 - Phát hiện khuôn mặt bằng Mediapipe.
 - Crop ảnh khuôn mặt, resize về kích thước 160x160.
 - Chuẩn hóa (prewhiten) để giảm ảnh hưởng ánh sáng và độ tương phản.
- Trích xuất vector đặc trưng từ ảnh khuôn mặt đã xử lý được, sau đó đưa vào mô hình FaceNet để thu được vector 512 chiều.

- Nhãn tên của mỗi người được mã hóa thành số bằng LabelEncoder

3.4.5.2. Huấn luyện mô hình

- Sau khi có tập dữ liệu gồm: **Input**: các vector đặc trưng (data), **Label**: tên người
- Tiến hành huấn luyện mô hình SVC như sau:

```
model = SVC(kernel='rbf', C=20.0, gamma=0.01, probability=True)
model.fit(data, y)
```

- Trong đó:

- kernel='rbf': sử dụng kernel phi tuyến Gaussian để phân biệt các lớp không tuyến tính.
- C=20.0: hệ số điều chỉnh giữa margin và lỗi phân loại.
- gamma=0.01: ảnh hưởng của từng điểm dữ liệu lên ranh giới phân loại.
- probability=True: cho phép mô hình xuất ra xác suất dự đoán (phục vụ đánh giá độ tin cậy khi nhận diện).
- data: danh sách vector đặc trưng từ ảnh khuôn mặt.
- y: nhãn tương ứng, đã được mã hóa số từ tên người.

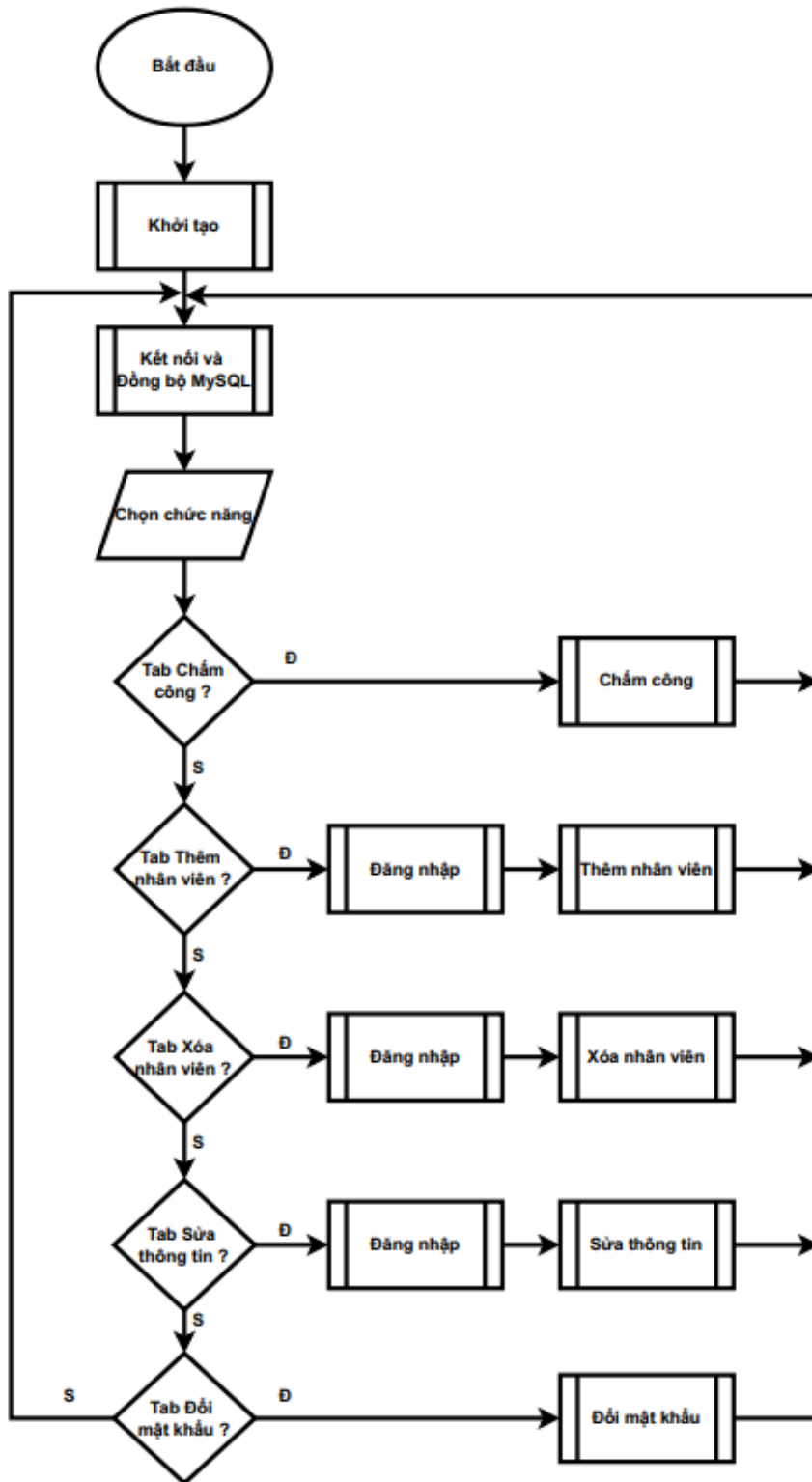
- Sau khi huấn luyện xong, hệ thống sẽ lưu lại mô hình cùng với bộ mã hóa nhãn vào file facemodel.pkl để dùng lại khi nhận diện.

3.5. Thiết kế khối thiết bị

Trung tâm xử lý với phần cứng là Máy tính nhúng Raspberry Pi 4B, sẽ thực hiện toàn bộ chức năng từ thu thập dữ liệu hình ảnh khuôn mặt, tiền xử lý hình ảnh, huấn luyện mô hình, thực hiện phân loại và nhận diện, ghi thông tin chấm công vào file CSV và MySQL, ...v.v...

Các chức năng được Trung tâm xử lý đảm nhiệm sẽ được mô tả cụ thể thông qua lưu đồ thuật toán của hệ thống chấm công ở Hình 3.2

3.5.1. Xây dựng lưu đồ chính của hệ thống



Hình 3.2 Lưu đồ chính của hệ thống

Hình 3.2 mô tả lưu đồ chính của hệ thống. Khi hệ thống được khởi động, bước đầu tiên là khởi tạo hệ thống, nơi các thành phần cần thiết được thiết lập như camera, mô hình nhận diện khuôn mặt.

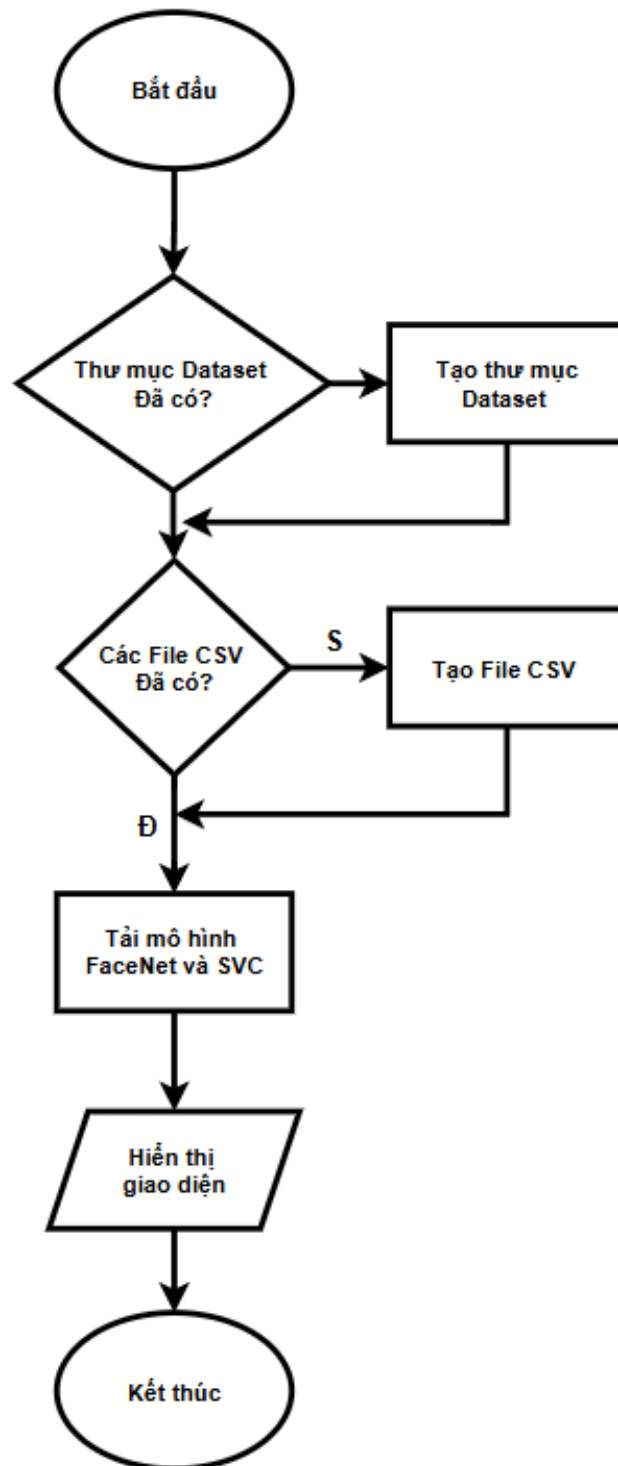
Tiếp theo là bước kết nối và đồng bộ MySQL. Trong bước này, hệ thống kiểm tra trạng thái kết nối đến cơ sở dữ liệu MySQL, tạo các file và các bảng nếu chưa có, và tiến hành đồng bộ dữ liệu từ các tệp CSV sang cơ sở dữ liệu để đảm bảo tính nhất quán.

Sau khi khởi tạo xong, hệ thống chuyển sang bước chọn chức năng cần dùng. Người dùng có thể chọn một trong các chức năng chính sau:

- **Chấm công** – Nhận diện khuôn mặt để ghi nhận thời gian làm việc.
- **Thêm nhân viên** – Nhập thông tin và thu thập ảnh khuôn mặt của nhân viên mới.
- **Xóa nhân viên** – Tìm và xóa thông tin nhân viên khỏi hệ thống.
- **Sửa thông tin** – Cập nhật thông tin hoặc ID của nhân viên.
- **Đổi mật khẩu** – Thay đổi mật khẩu tài khoản quản trị.

Ba chức năng Thêm, Xóa và Sửa yêu cầu xác thực tài khoản quản trị trước khi thao tác.

3.5.2. Lưu đồ con chương trình Khởi tạo



Hình 3.3 Lưu đồ chương trình con khởi tạo của hệ thống

Hình 3.3 mô tả lưu đồ quy trình khởi tạo hệ thống. Khi bắt đầu, hệ thống thực hiện kiểm tra các điều kiện ban đầu để đảm bảo môi trường hoạt động đã sẵn sàng.

Bước đầu tiên là kiểm tra sự tồn tại của thư mục Dataset – đây là nơi lưu trữ dữ liệu khuôn mặt. Nếu thư mục này chưa tồn tại, hệ thống sẽ tiến hành tạo thư mục Dataset mới. Nếu thư mục đã có sẵn, hệ thống tiếp tục sang bước kế tiếp.

Tiếp theo, hệ thống kiểm tra sự tồn tại của các tệp CSV cần thiết, bao gồm employees.csv và timekeeping.csv. Nếu các tệp này chưa có, hệ thống sẽ tự động tạo mới các file CSV để lưu trữ thông tin nhân viên và dữ liệu chấm công.

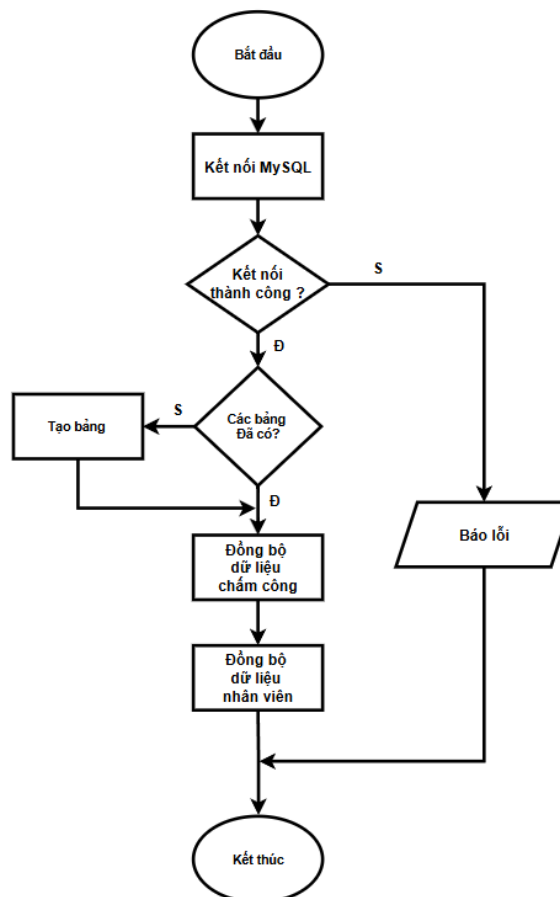
Sau khi hoàn tất việc kiểm tra và tạo dữ liệu đầu vào, hệ thống tiến hành tải mô hình FaceNet và SVC. Đây là các thành phần chính giúp hệ thống có thể nhận diện khuôn mặt và phân loại nhân viên dựa trên dữ liệu đã huấn luyện.

Cuối cùng, sau khi các tài nguyên đã được khởi tạo và tải đầy đủ, hệ thống sẽ hiển thị giao diện người dùng, sẵn sàng để người dùng thao tác. Kết thúc quá trình khởi tạo, hệ thống chuyển sang chế độ chờ sử dụng các chức năng chính.

3.5.3. Lưu đồ con chương trình Kết nối và đồng bộ MySQL

Các thư viện Python cần dùng:

- **mysql.connector:** Kết nối, truy vấn, thao tác với cơ sở dữ liệu MySQL, bắt và xử lý lỗi khi thao tác với MySQL.
- **pandas:** Đọc, ghi, xử lý file CSV để đồng bộ dữ liệu giữa CSV và MySQL.
- **os:** Kiểm tra, thao tác file/thư mục khi đồng bộ dữ liệu (ví dụ: kiểm tra file CSV tồn tại).
- **dotenv:** Đọc biến môi trường (thông tin cấu hình kết nối MySQL) từ file .env.



Hình 3.4 Lưu đồ con chương trình con kết nối và đồng bộ MySQL

Hình 3.4 mô tả lưu đồ quy trình kết nối và đồng bộ dữ liệu với cơ sở dữ liệu MySQL. Quá trình bắt đầu từ nút "Bắt đầu", sau đó hệ thống tiến hành thực hiện thao tác kết nối đến cơ sở dữ liệu MySQL để chuẩn bị cho các bước xử lý tiếp theo.

Ở giai đoạn đầu tiên, hệ thống cố gắng thiết lập kết nối với máy chủ MySQL. Ngay sau khi gửi yêu cầu kết nối, hệ thống kiểm tra xem kết nối này có thành công hay không. Nếu kết nối được thiết lập thành công, hệ thống kiểm tra xem đã có các bảng trong cơ sở dữ liệu hay chưa. Nếu chưa có, hệ thống tiến hành tạo các bảng.

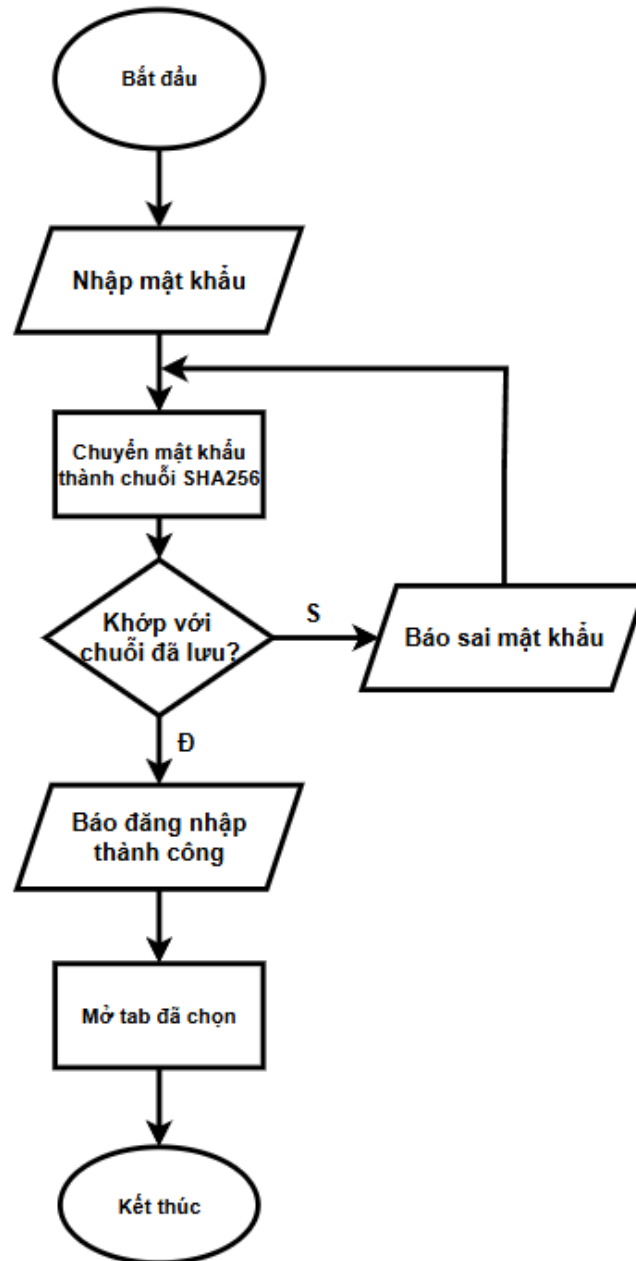
Tiếp theo, chuyển sang bước đồng bộ dữ liệu. Đầu tiên, hệ thống thực hiện đồng bộ dữ liệu chăm công từ hệ thống vào cơ sở dữ liệu, đảm bảo cập nhật đầy đủ thông tin chăm công mới nhất. Sau đó, hệ thống tiến hành đồng bộ dữ liệu nhân viên, giúp đảm bảo mọi thông tin về nhân sự cũng được cập nhật chính xác và kịp thời trên cơ sở dữ liệu.

Trong trường hợp việc kết nối đến MySQL không thành công, hệ thống sẽ chuyển sang bước xử lý lỗi. Tại đây, hệ thống hiển thị thông báo lỗi để người dùng biết về sự cố kết nối, đồng thời quy trình sẽ kết thúc mà không thực hiện các thao tác đồng bộ tiếp theo nhằm đảm bảo an toàn dữ liệu.

3.5.4. Lưu đồ con chương trình Đăng nhập

Các thư viện Python cần dùng:

- **json:** đọc file admin password.json chứa mật khẩu admin đã mã hóa.
- **hashlib:** mã hóa mật khẩu người dùng nhập vào (SHA-256) để so sánh với mật khẩu đã lưu.



Hình 3.5 Lưu đồ chương trình con đăng nhập

Hình 3.5 trình bày lưu đồ quy trình đăng nhập dành cho tài khoản quản trị. Quy trình bắt đầu khi người dùng truy cập vào một chức năng yêu cầu xác thực, như thêm, xóa hoặc sửa thông tin nhân viên.

Bước đầu tiên, hệ thống yêu cầu người dùng nhập mật khẩu. Sau khi nhập, mật khẩu sẽ được chuyển đổi thành chuỗi mã hóa SHA256 nhằm đảm bảo an toàn và bảo mật thông tin.

Hệ thống sau đó thực hiện so sánh chuỗi mã hóa vừa tạo với chuỗi mã hóa mật khẩu đã lưu sẵn trước đó trong hệ thống. Nếu hai chuỗi không khớp, hệ thống sẽ thông báo

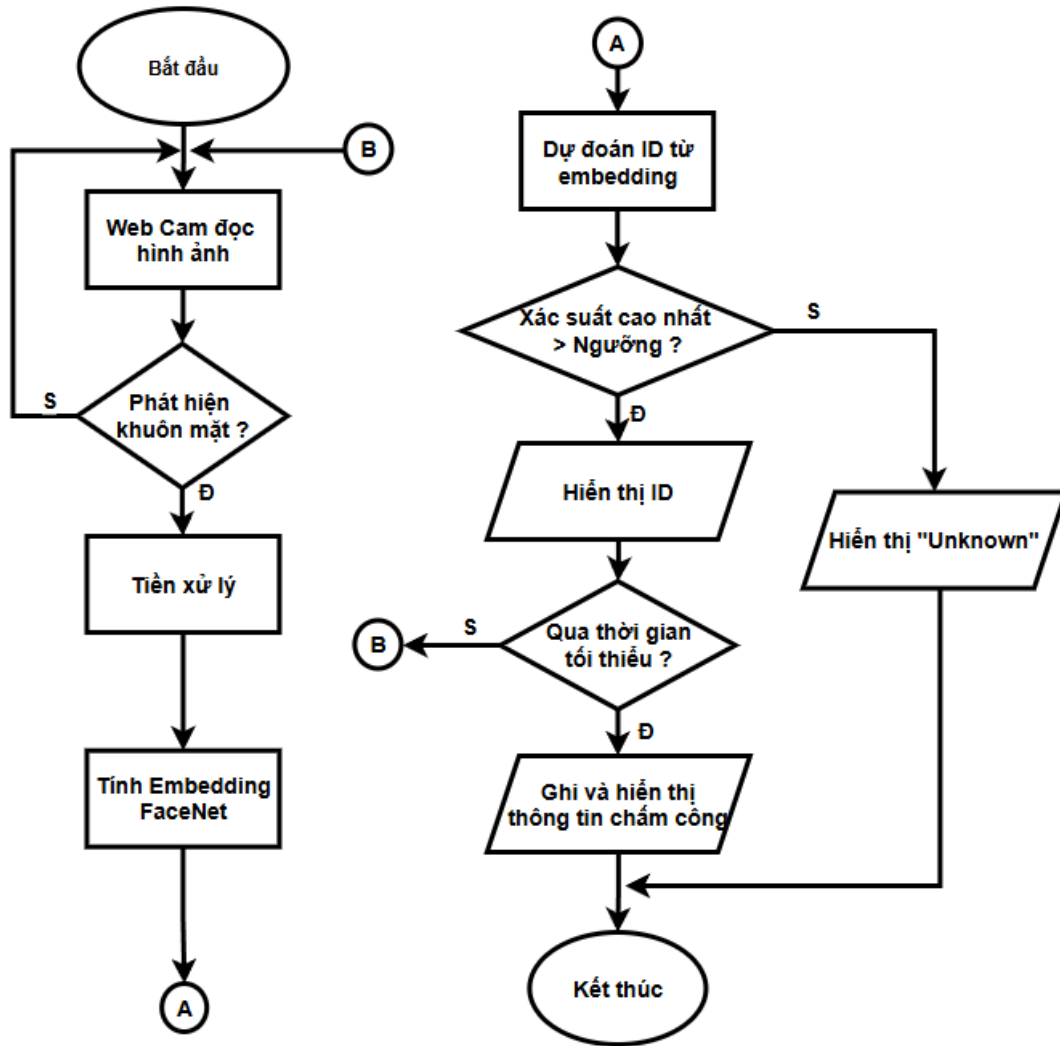
sai mặt khẩu, cho phép người dùng nhập lại. Ngược lại, nếu chuỗi khớp, hệ thống sẽ báo đăng nhập thành công.

Khi đăng nhập thành công, hệ thống sẽ mở tab chức năng mà người dùng đã chọn trước đó (ví dụ: thêm nhân viên, xóa nhân viên, sửa thông tin), và tiếp tục các thao tác bên trong tab tương ứng. Kết thúc quá trình xác thực, người dùng được phép thực hiện các chức năng yêu cầu quyền quản trị.

3.5.5. Lưu đồ con chương trình Chấm công

Các thư viện Python cần dùng:

- **cv2:** Thiết lập và lấy hình ảnh từ camera, xử lý ảnh, cắt và lưu ảnh khuôn mặt.
- **imutils:** Hỗ trợ resize, thao tác ảnh thuận tiện khi xử lý video từ camera.
- **mediapipe:** Phát hiện khuôn mặt trong ảnh/video (face detection).
- **facenet:** Tiền xử lý ảnh, trích xuất đặc trưng khuôn mặt (embedding) phục vụ nhận diện.
- **tensorflow:** Chạy mô hình nhận diện khuôn mặt (FaceNet) để sinh embedding.
- **sklearn.svm:** Phân loại khuôn mặt bằng mô hình SVC.
- **numpy:** Xử lý dữ liệu số, mảng ảnh, tính toán embedding.
- **pandas:** Đọc, ghi, xử lý file CSV để lưu và truy xuất dữ liệu chấm công.
- **datetime:** Lấy ngày, giờ hiện tại khi chấm công.
- **threading:** Đảm bảo các thao tác ghi dữ liệu, xử lý ảnh diễn ra song song, không bị treo giao diện.



Hình 3.6 Lưu đồ chương trình con chấm công

Hình 3.6 trình bày lưu đồ chi tiết của quy trình chấm công tự động bằng nhận diện khuôn mặt. Quy trình bắt đầu khi hệ thống khởi chạy chức năng chấm công.

Đầu tiên, Webcam liên tục đọc hình ảnh từ môi trường xung quanh. Mỗi khung hình được phân tích để xác định xem có khuôn mặt xuất hiện hay không. Nếu không phát hiện khuôn mặt, hệ thống quay trở lại bước đọc ảnh tiếp theo. Nếu có phát hiện khuôn mặt, hệ thống sẽ chuyển sang giai đoạn xử lý.

Tại bước tiền xử lý, ảnh sẽ được xử lý chuẩn hóa (như cân bằng sáng, resize...) để chuẩn bị cho việc trích xuất đặc trưng. Sau đó, hệ thống thực hiện cắt và giữ vùng ảnh chứa khuôn mặt.

Kế tiếp, khuôn mặt đã được cắt sẽ được đưa vào mô hình FaceNet để tính toán embedding, tức là một vector đặc trưng đại diện cho khuôn mặt đó. Hệ thống sử dụng embedding để dự đoán ID của khuôn mặt bằng mô hình phân loại đã được huấn luyện. Sau khi dự đoán, hệ thống kiểm tra xác suất dự đoán cao nhất có vượt ngưỡng tin cậy

hay không. Nếu không vượt ngưỡng, hệ thống hiển thị "Unknown" và kết thúc quy trình. Nếu đáp ứng ngưỡng, hệ thống hiển thị ID đã nhận diện được và kết thúc quy trình.

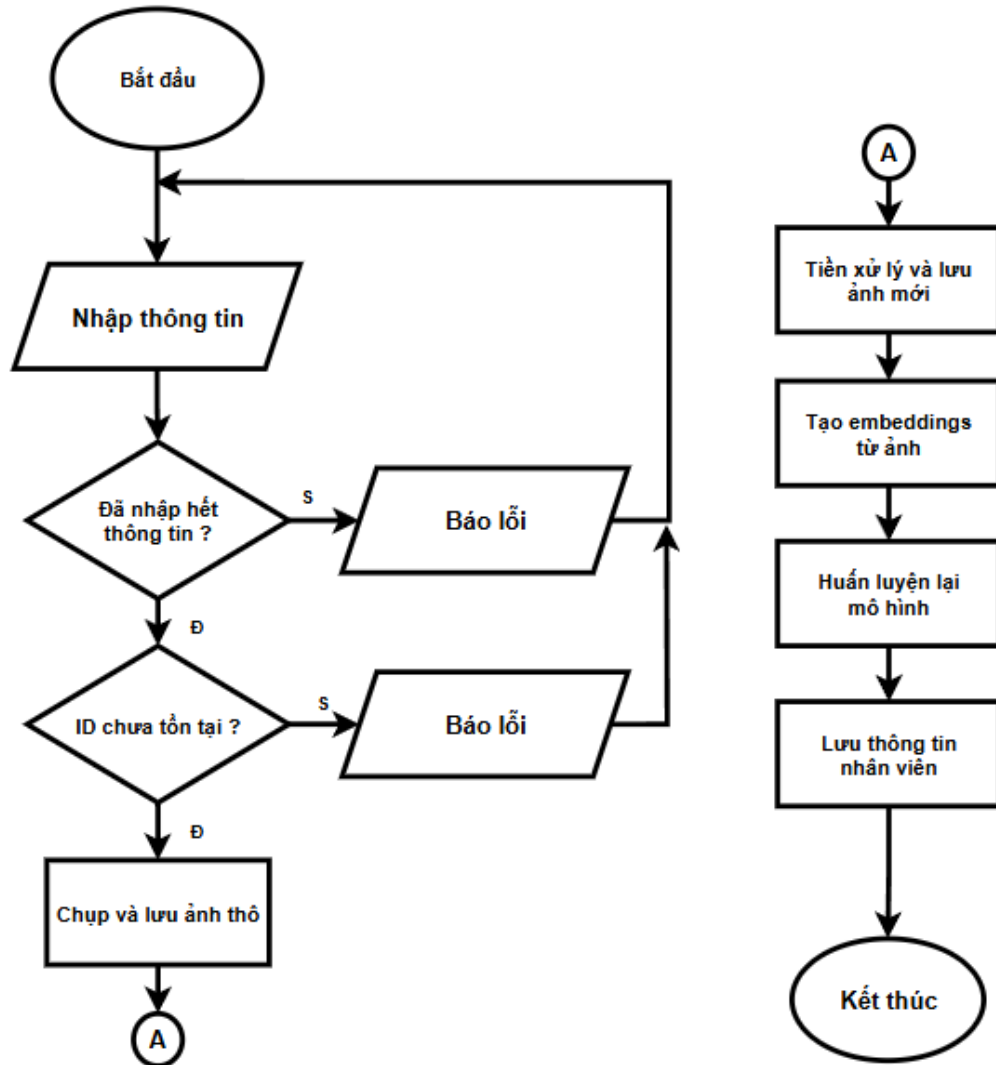
Tiếp theo, hệ thống kiểm tra xem đã đủ thời gian tối thiểu kể từ lần chấm công trước hay chưa (nhằm tránh ghi trùng liên tục). Nếu chưa đủ thời gian, hệ thống quay lại tiếp tục đọc ảnh. Nếu đã đủ thời gian, hệ thống tiến hành ghi thông tin chấm công (bao gồm ID, tên, chức vụ, ngày giờ) vào hệ thống và hiển thị kết quả cho người dùng.

Cuối cùng, sau khi ghi nhận thành công, hệ thống kết thúc chu trình chấm công hoặc tiếp tục lặp lại để nhận diện người tiếp theo. Quy trình này được thực hiện liên tục trong thời gian chức năng chấm công đang hoạt động.

3.5.6. Lưu đồ con chương trình Thêm nhân viên

Các thư viện Python cần dùng:

- **os:** Kiểm tra, tạo thư mục lưu dữ liệu khuôn mặt của nhân viên mới.
- **pandas:** Đọc, ghi, cập nhật file CSV lưu thông tin nhân viên.
- **threading:** Chạy song song quá trình thu thập dữ liệu khuôn mặt, tránh treo giao diện.
- **cv2:** Chụp và lưu ảnh khuôn mặt từ camera khi thu thập dữ liệu.
- **mediapipe:** Phát hiện khuôn mặt trong ảnh khi thu thập dữ liệu.
- **facenet:** Tiền xử lý ảnh khuôn mặt, chuẩn bị dữ liệu cho huấn luyện.



Hình 3.7 Lưu đồ chương trình con thêm nhân viên

Hình 3.7 trình bày lưu đồ quy trình thêm nhân viên vào hệ thống chăm công bằng nhận diện khuôn mặt. Quy trình bắt đầu khi người dùng truy cập chức năng “Thêm nhân viên”. Bước đầu tiên, hệ thống yêu cầu người dùng nhập đầy đủ các thông tin cần thiết bao gồm: mã nhân viên (ID), họ tên, và chức vụ.

Sau khi nhập thông tin, hệ thống thực hiện kiểm tra tính đầy đủ. Nếu người dùng chưa nhập đủ các trường thông tin, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng bổ sung đầy đủ. Việc này nhằm đảm bảo dữ liệu đầu vào chính xác và nhất quán.

Tiếp theo, hệ thống tiến hành kiểm tra tính hợp lệ của mã ID nhân viên. Nếu mã ID này đã tồn tại trong hệ thống, một thông báo lỗi sẽ được hiển thị để ngăn chặn việc trùng lặp dữ liệu. Người dùng sẽ cần nhập lại ID chưa được sử dụng. Nếu ID là duy nhất, quy trình sẽ tiếp tục.

Tại bước này, hệ thống tiến hành kích hoạt camera để chụp và lưu các ảnh thô của khuôn mặt nhân viên. Các ảnh này sẽ được lưu vào thư mục riêng theo ID nhân viên, để phục vụ cho quá trình huấn luyện mô hình.

Sau khi thu thập xong ảnh, hệ thống thực hiện bước tiền xử lý ảnh, bao gồm việc cắt vùng khuôn mặt, resize ảnh, ... Sau đó, mỗi ảnh sẽ được đưa vào mô hình FaceNet để tính toán vector đặc trưng (embedding) đại diện cho khuôn mặt.

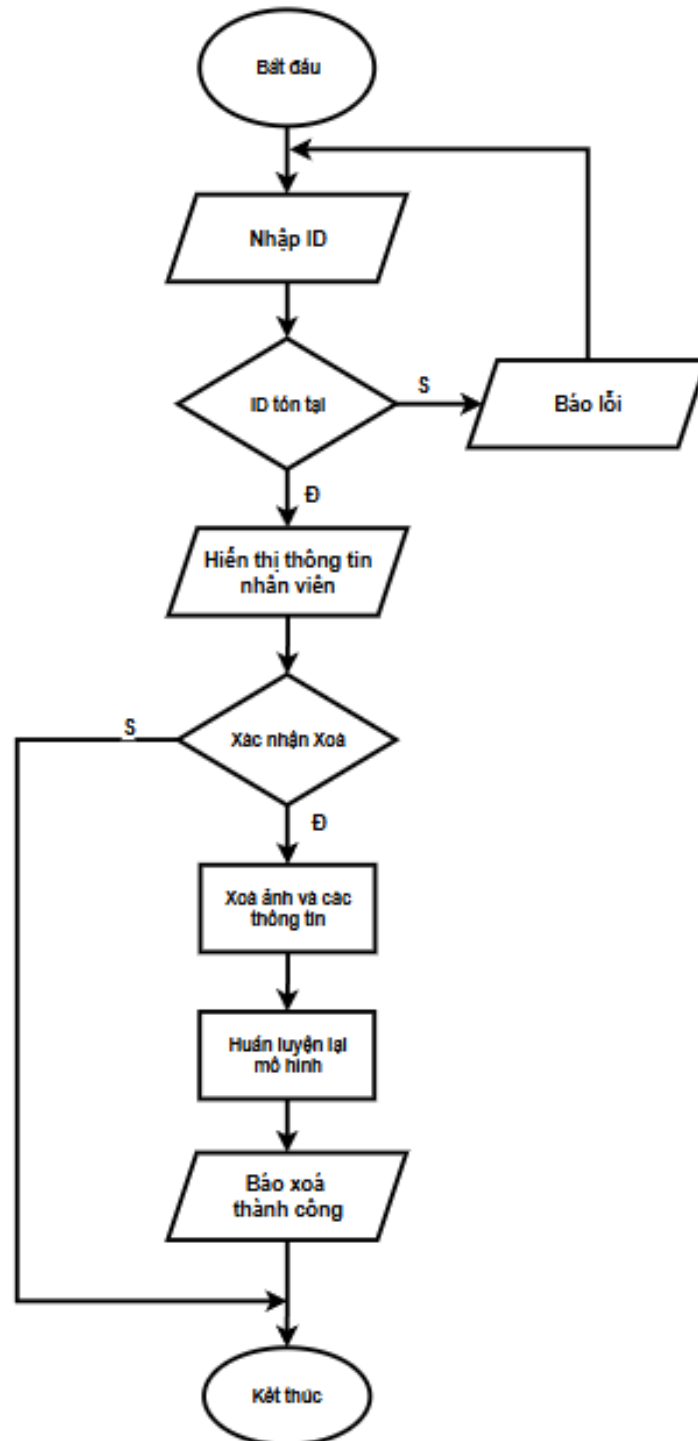
Khi đã có embedding, hệ thống tiến hành huấn luyện lại mô hình phân loại (SVC) với toàn bộ dữ liệu hiện có, bao gồm cả nhân viên mới vừa được thêm. Việc huấn luyện giúp hệ thống cập nhật và sẵn sàng nhận diện nhân viên mới trong các lần chấm công sau.

Cuối cùng, hệ thống thực hiện bước lưu thông tin nhân viên mới vào hệ thống quản lý để đảm bảo dữ liệu được cập nhật đầy đủ. Khi quá trình hoàn tất, hệ thống kết thúc quy trình thêm nhân viên và trở về trạng thái sẵn sàng cho các thao tác tiếp theo.

3.5.7. Lưu đồ con chương trình Xóa nhân viên

Các thư viện Python cần dùng:

- **os:** Kiểm tra, xóa thư mục và file dữ liệu khuôn mặt của nhân viên trên ổ đĩa.
- **pandas:** Đọc, ghi, cập nhật file CSV để xóa thông tin nhân viên và lịch sử chấm công.
- **threading:** Thực hiện thao tác xóa dữ liệu song song, tránh treo giao diện.



Hình 3.8 Lưu đồ chương trình con xóa nhân viên

Hình 3.8 trình bày lưu đồ quy trình xóa nhân viên trong hệ thống chăm công bằng nhận diện khuôn mặt.

Quy trình bắt đầu khi người dùng truy cập vào chức năng “Xóa nhân viên”. Hệ thống yêu cầu người dùng nhập ID của nhân viên cần xóa để xác định đối tượng cần xử lý.

Sau khi nhập ID, hệ thống thực hiện kiểm tra sự tồn tại của ID đó trong hệ thống. Nếu ID không tồn tại, hệ thống sẽ hiển thị thông báo lỗi, đồng thời kết thúc hoặc yêu cầu nhập lại.

Ngược lại, nếu ID tồn tại, hệ thống sẽ hiển thị thông tin chi tiết của nhân viên, bao gồm các trường như họ tên và chức vụ, để người dùng xác nhận lại.

Tại bước tiếp theo, người dùng được yêu cầu xác nhận việc xóa nhân viên. Nếu người dùng không xác nhận, quy trình sẽ dừng và quay lại bước ban đầu. Nếu người dùng đồng ý xóa, hệ thống tiến hành các bước xử lý tiếp theo.

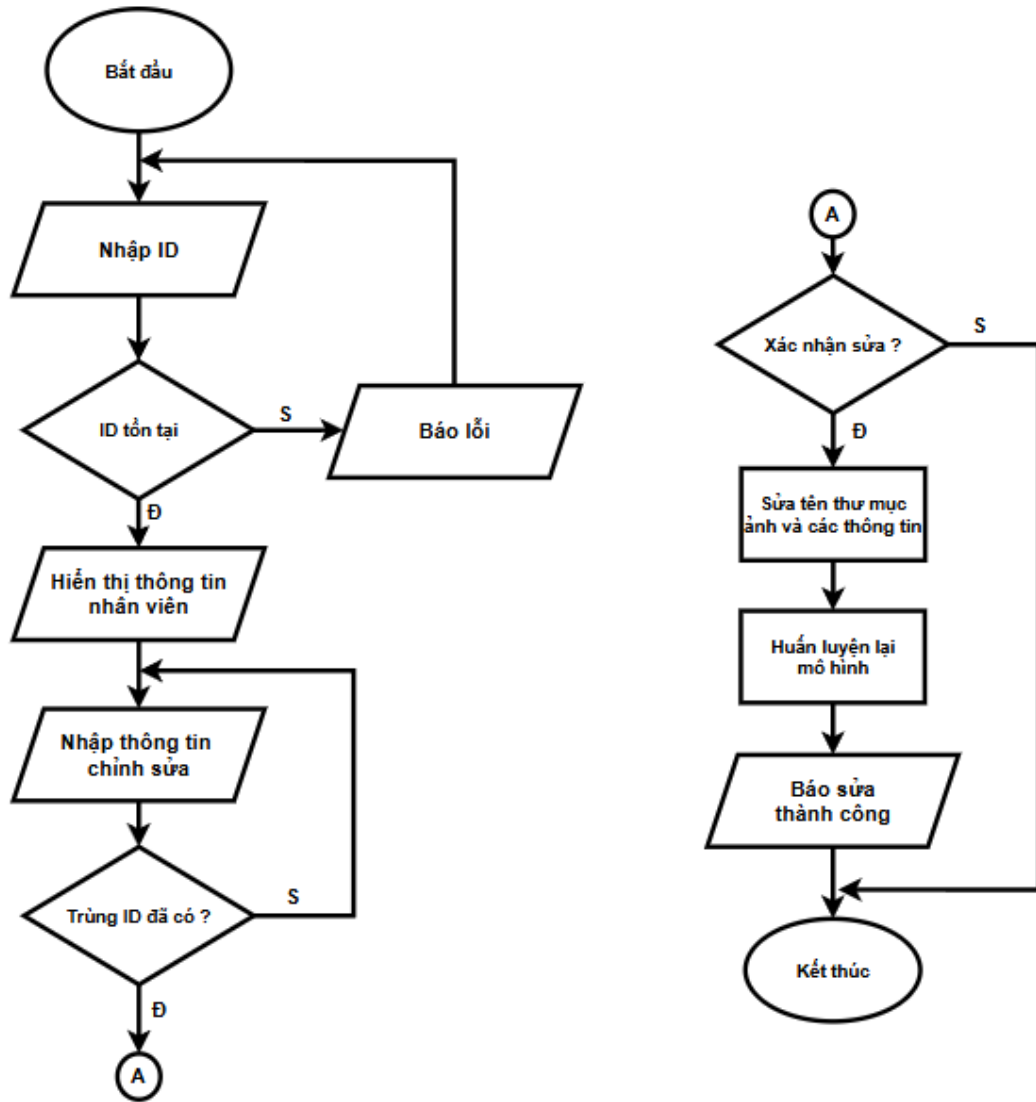
Trước tiên, hệ thống xóa toàn bộ thông tin và dữ liệu liên quan đến nhân viên đó, bao gồm ảnh khuôn mặt và dữ liệu nhận dạng. Sau đó, hệ thống huấn luyện lại mô hình nhận diện khuôn mặt, nhằm loại bỏ ảnh hưởng của nhân viên vừa bị xóa và đảm bảo độ chính xác trong các lần nhận diện tiếp theo.

Khi quá trình xóa hoàn tất, hệ thống hiển thị thông báo xóa thành công, kết thúc quy trình và sẵn sàng cho thao tác mới.

3.5.8. Lưu đồ con chương trình Sửa thông tin

Các thư viện Python cần dùng:

- **os:** Đổi tên, kiểm tra, thao tác thư mục và file dữ liệu khuôn mặt khi sửa ID nhân viên.
- **pandas:** Đọc, ghi, cập nhật file CSV để sửa thông tin nhân viên và lịch sử chăm công.
- **threading:** Chạy lại huấn luyện mô hình nhận diện sau khi sửa thông tin, tránh treo giao diện.



Hình 3.9 Lưu đồ chương trình con sửa thông tin nhân viên

Hình 3.9 mô tả lưu đồ quy trình sửa thông tin nhân viên trong hệ thống chăm công bằng nhận diện khuôn mặt.

Quy trình sửa thông tin nhân viên bắt đầu khi người dùng truy cập vào chức năng “Sửa thông tin” trên giao diện hệ thống. Tại đây, hệ thống yêu cầu người dùng nhập mã ID của nhân viên cần chỉnh sửa. Đây là bước đầu tiên để xác định chính xác đối tượng cần cập nhật thông tin.

Sau khi người dùng nhập ID, hệ thống tiến hành kiểm tra xem ID đó có tồn tại trong hệ thống hay không. Nếu ID không tồn tại, hệ thống sẽ hiển thị thông báo lỗi và quy trình quay lại bước nhập ID để thử lại. Ngược lại, nếu ID hợp lệ, hệ thống sẽ hiển thị đầy đủ thông tin hiện tại của nhân viên đó, bao gồm ID, tên, chức vụ.

Tiếp theo, người dùng sẽ nhập các thông tin cần chỉnh sửa, có thể bao gồm tên, chức vụ hoặc mã ID mới. Sau khi nhập xong, hệ thống kiểm tra ID mới có trùng với một nhân viên khác đang có trong hệ thống hay không. Nếu ID mới bị trùng, hệ thống sẽ báo lỗi

và yêu cầu người dùng sửa lại. Nếu không có lỗi, hệ thống yêu cầu người dùng xác nhận hành động sửa thông tin. Nếu người dùng không xác nhận, quy trình sẽ dừng và quay lại giao diện ban đầu. Nếu xác nhận sửa, hệ thống sẽ tiến hành cập nhật lại thông tin nhân viên, bao gồm cả việc đổi tên thư mục ảnh nhận diện khuôn mặt nếu ID bị thay đổi, cũng như cập nhật thông tin trong hệ thống quản lý.

Sau đó, hệ thống thực hiện huấn luyện lại mô hình nhận diện khuôn mặt, nhằm đảm bảo các thay đổi về nhân viên được cập nhật vào mô hình phân loại hiện tại. Việc này giúp mô hình tiếp tục hoạt động chính xác trong các lần nhận diện sau.

Cuối cùng, hệ thống hiển thị thông báo sửa thành công, và kết thúc quy trình. Hệ thống sau đó trở về trạng thái sẵn sàng cho các thao tác tiếp theo.

3.5.9. Lưu đồ con chương trình Đổi mật khẩu

Các thư viện Python cần dùng:

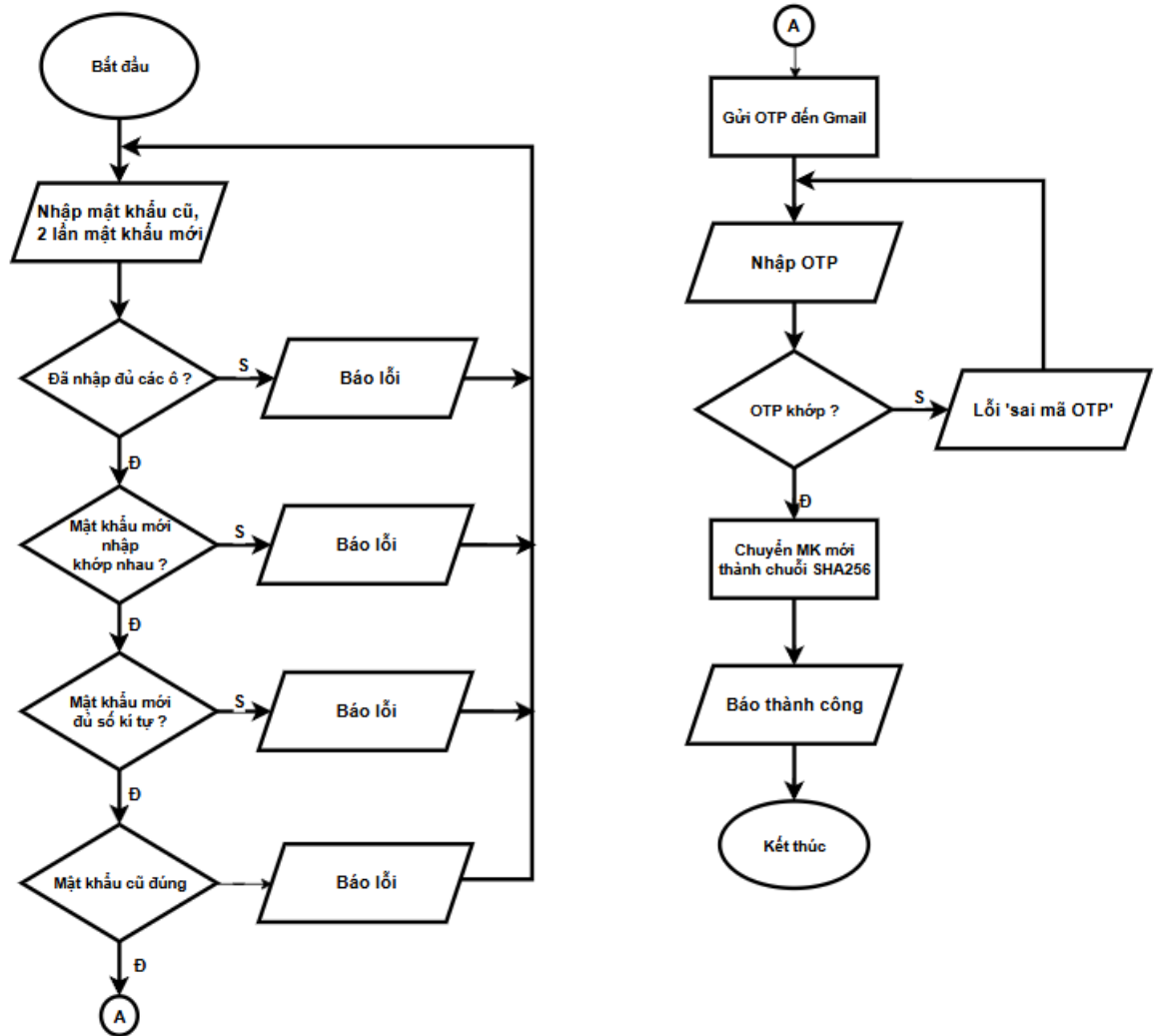
json: Đọc và ghi file admin password.json để lưu mật khẩu admin đã mã hóa.

hashlib: Mã hóa mật khẩu mới bằng SHA-256 trước khi lưu.

random: Sinh mã OTP ngẫu nhiên gửi qua email.

smtplib: Gửi email chứa mã OTP xác thực đổi mật khẩu.

email.mime.text, email.mime.multipart: Tạo nội dung email OTP



Hình 3.10 Lưu đồ chương trình con đổi mật khẩu

Hình 3.10 mô tả lưu đồ quy trình đổi mật khẩu trong hệ thống chăm công bằng nhận diện khuôn mặt.

Quy trình đổi mật khẩu bắt đầu khi người dùng truy cập vào chức năng “Đổi mật khẩu” trên giao diện hệ thống. Hệ thống yêu cầu người dùng nhập mật khẩu cũ và nhập mật khẩu mới hai lần để xác nhận tính chính xác.

Ngay sau đó, hệ thống tiến hành kiểm tra tính hợp lệ của thông tin đã nhập. Nếu người dùng chưa nhập đầy đủ các ô, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại. Nếu đã nhập đủ, hệ thống tiếp tục kiểm tra hai mật khẩu mới có khớp nhau không. Nếu không khớp, hệ thống tiếp tục thông báo lỗi.

Tiếp theo, hệ thống kiểm tra độ dài tối thiểu của mật khẩu mới. Nếu mật khẩu mới không đạt yêu cầu về số ký tự hệ thống sẽ báo lỗi. Nếu hợp lệ, hệ thống kiểm tra tính đúng đắn của mật khẩu cũ bằng cách so sánh với mật khẩu hiện tại đang lưu. Nếu sai, hệ thống sẽ hiển thị lỗi. Ngược lại, nếu đúng, hệ thống thực hiện bước gửi mã OTP về

địa chỉ Gmail đã cấu hình trước đó. Người dùng sẽ kiểm tra hộp thư và nhập mã OTP vào ô xác thực.

Hệ thống tiếp tục kiểm tra mã OTP nhập vào có khớp với mã đã gửi không. Nếu sai, hệ thống hiển thị thông báo lỗi “Sai mã OTP” và yêu cầu nhập lại. Nếu đúng, hệ thống tiến hành chuyển đổi mật khẩu mới thành chuỗi SHA256 để đảm bảo bảo mật, và ghi nhận thay đổi này vào hệ thống.

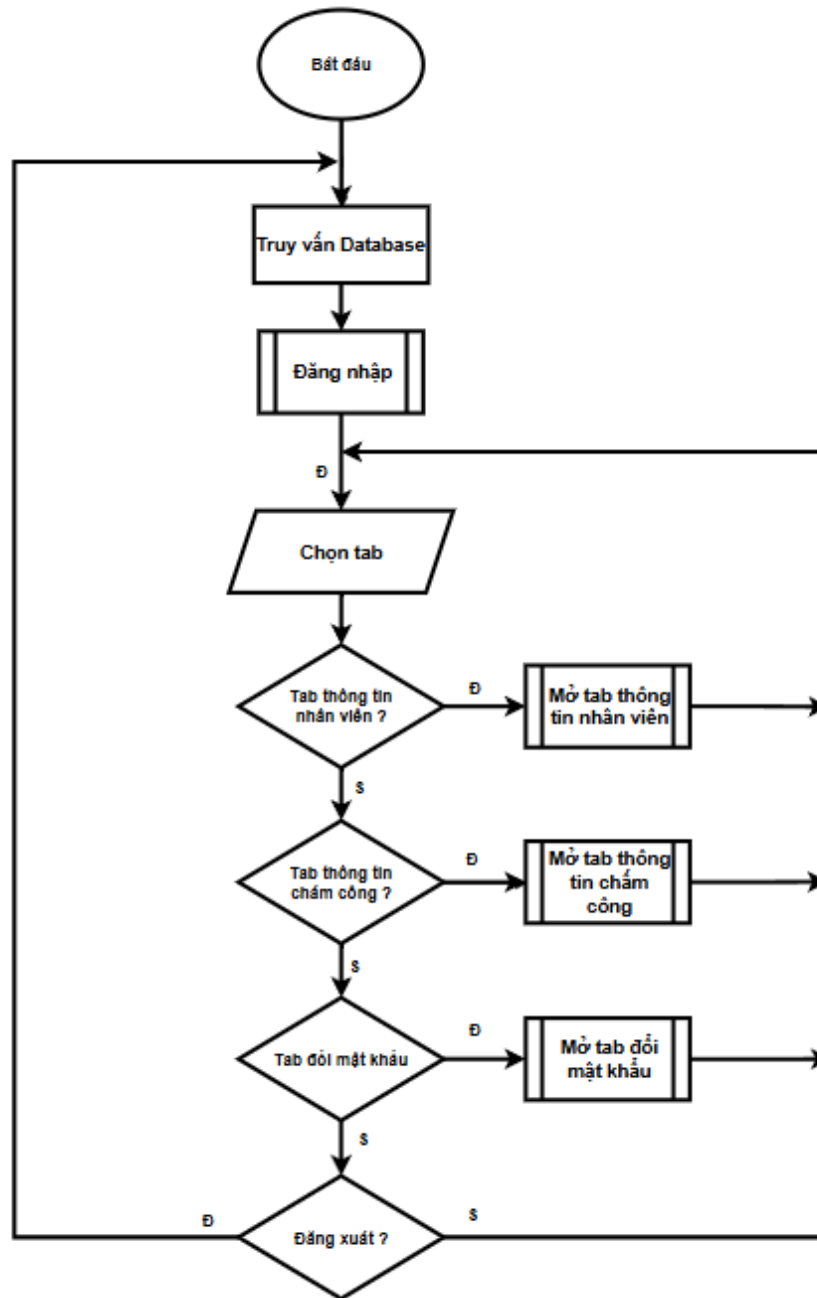
Cuối cùng, hệ thống hiển thị thông báo đổi mật khẩu thành công, kết thúc quy trình.

3.6. Thiết kế và triển khai khối Web App

3.6.1. Web App

3.6.1.1. Xây dựng lưu đồ quy trình hoạt động của Web App

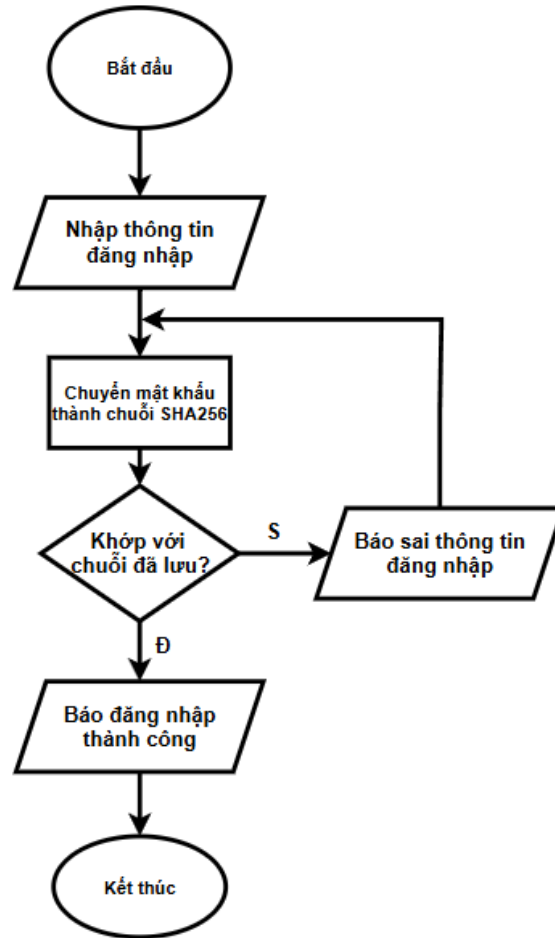
Các lưu đồ dưới đây thể hiện các chức năng chính trong quá trình sử dụng hệ thống, bao gồm đăng nhập, chọn chức năng, lọc và xuất dữ liệu và đổi mật khẩu.



Hình 3.11 Lưu đồ quy trình chính của trang web

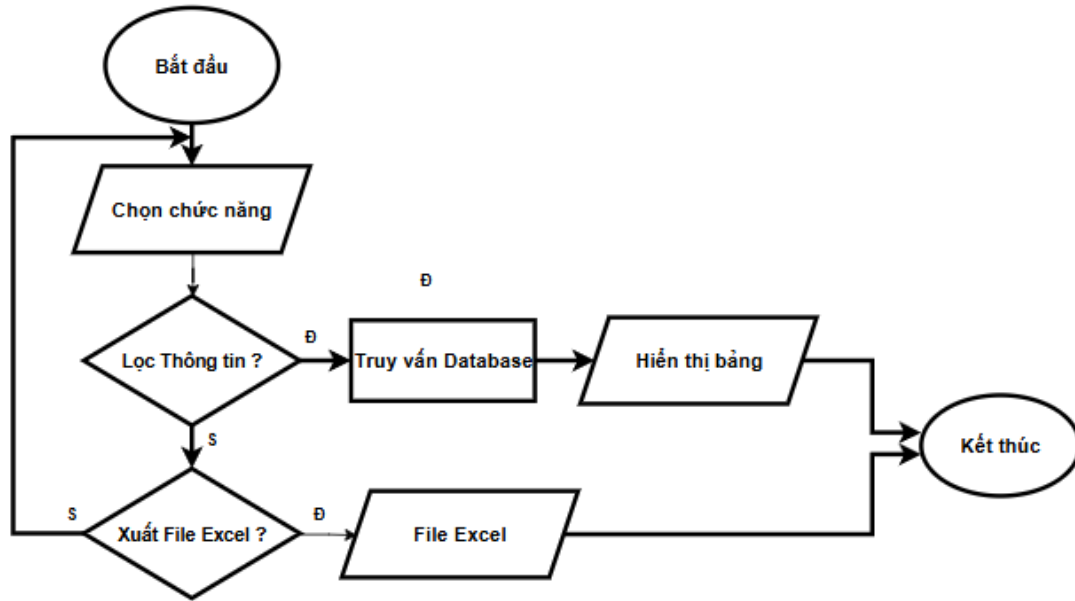
Hình 3.11 mô tả quy trình hệ thống hoạt động bằng cách truy vấn đến cơ sở dữ liệu và cho phép người dùng thực hiện đăng nhập. Sau khi đăng nhập thành công, người dùng có thể lựa chọn các tab chức năng để xem và thao tác như: "Thông tin nhân viên", "Thông tin chấm công", hoặc "Đổi mật khẩu".

Mỗi lựa chọn sẽ dẫn đến việc hiển thị giao diện tương ứng. Nếu người dùng muốn kết thúc phiên làm việc, họ có thể chọn "Đăng xuất" để quay lại trang đăng nhập.

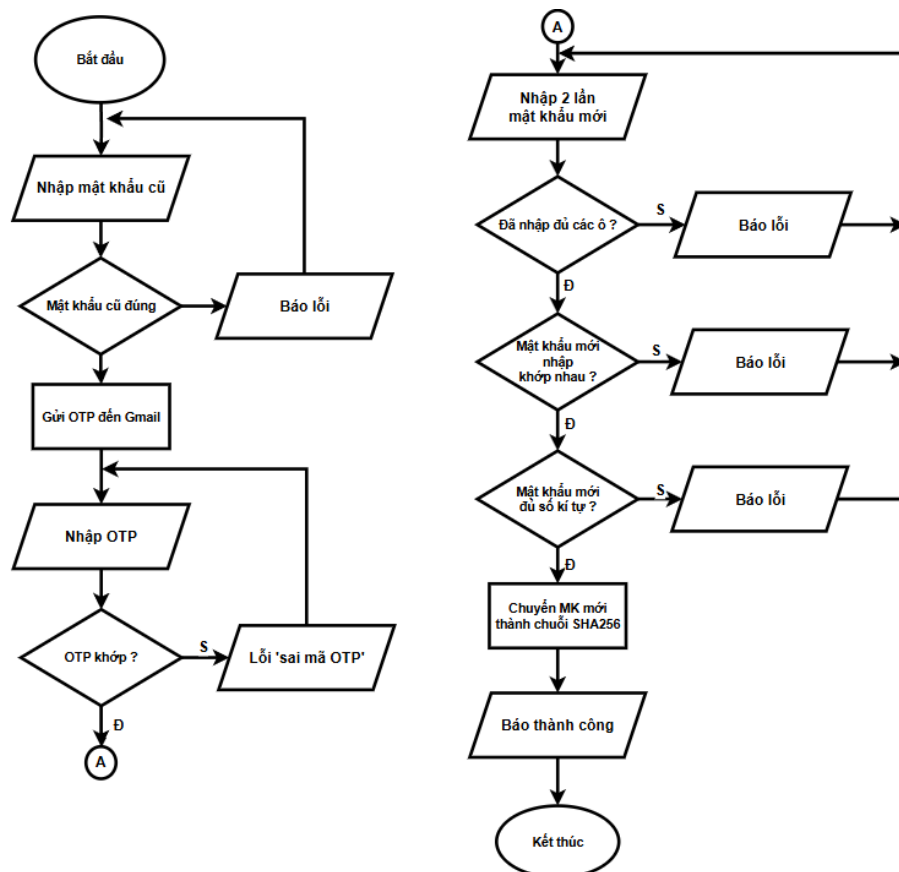


Hình 3.12 Lưu đồ quy trình đăng nhập

Hình 3.12 mô tả quy trình xác thực đăng nhập trong hệ thống. Người dùng bắt đầu bằng việc nhập tên đăng nhập và mật khẩu. Ngay sau đó, mật khẩu được hệ thống mã hóa bằng thuật toán SHA256 nhằm tăng cường bảo mật. Chuỗi mã hóa này được so sánh với mật khẩu đã lưu trong cơ sở dữ liệu. Nếu hai chuỗi khớp nhau, hệ thống xác nhận thông tin hợp lệ, cho phép người dùng truy cập và chuyển đến giao diện chính. Ngược lại, nếu không trùng khớp, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng nhập lại thông tin.



Hình 3.13 Lưu đồ con của các tab thông tin nhân viên và điểm danh



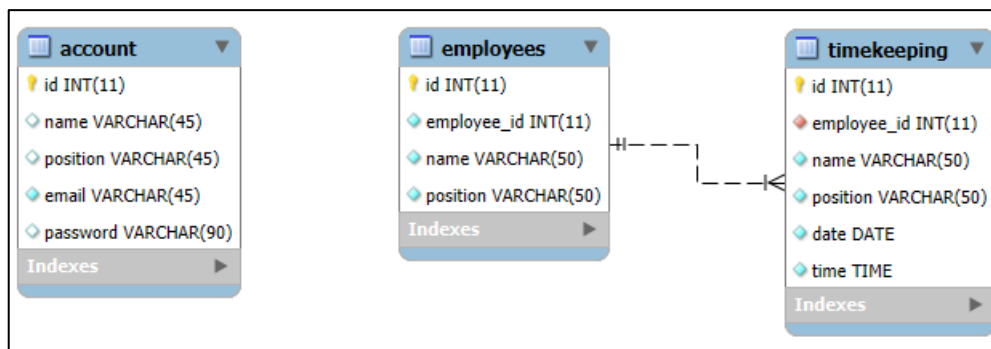
Hình 3.14 Lưu đồ quy trình đổi mật khẩu

Hình 3.13 mô tả chức năng lọc và xuất dữ liệu. Sau khi bắt đầu, người dùng lựa chọn chức năng xử lý. Nếu người dùng chọn lọc thông tin, hệ thống sẽ truy vấn đến cơ sở dữ

liệu và hiển thị bảng dữ liệu phù hợp. Sau đó, nếu người dùng có nhu cầu, họ có thể tiếp tục xuất dữ liệu ra file Excel. File này sẽ được hệ thống tạo và cung cấp cho người dùng trước khi kết thúc quá trình.

Quy trình đổi mật khẩu được trình bày rõ ràng trong **hình 3.14**, bao gồm bước xác thực mật khẩu cũ và mã OTP gửi về Gmail. Nếu người dùng nhập đúng mật khẩu cũ và mã OTP, họ sẽ được chuyển đến bước nhập mật khẩu mới. Hệ thống kiểm tra xem các ô đã được điền đầy đủ chưa, mật khẩu mới có trùng khớp và đủ độ dài quy định không. Nếu tất cả điều kiện đều thỏa mãn, mật khẩu mới sẽ được mã hóa bằng thuật toán SHA256 và thông báo đổi mật khẩu thành công.

3.6.1.2. Xây dựng cơ sở dữ liệu



Hình 3.15 Các bảng trong cơ sở dữ liệu MySQL

Cơ sở dữ liệu của hệ thống được tổ chức thành ba bảng chính như **hình 3.15**: account, employees và timekeeping, với mối quan hệ chặt chẽ nhằm hỗ trợ quản lý tài khoản người dùng, thông tin nhân viên và dữ liệu chấm công.

Bảng account lưu trữ thông tin đăng nhập bao gồm các trường như:

- id là mã định danh tài khoản
- name là tên tài khoản quản trị
- position thể hiện chức vụ
- email dùng để xác thực và nhận OTP
- password là mật khẩu đã được mã hóa theo chuẩn SHA256

Tiếp theo là bảng employees, nơi lưu thông tin chi tiết của nhân viên như:

- id là khóa chính
- employee_id là mã nhân viên
- name là tên nhân viên
- position là chức vụ

- employees_id có thể dùng để liên kết với các bảng khác hoặc thể hiện cấu trúc tổ chức.

Cuối cùng, bảng timekeeping ghi lại dữ liệu chấm công, bao gồm các trường:

- id là mã định danh bản ghi chấm công
- employee_id để liên kết với bảng employees
- name và position được lưu trữ nhằm thuận tiện theo dõi
- date là ngày chấm công
- time là giờ chấm công cụ thể

Bảng timekeeping có mối liên kết trực tiếp với bảng employees qua trường employee_id, cho phép mỗi bản ghi chấm công gắn với một nhân viên cụ thể.

Thiết kế cơ sở dữ liệu này đơn giản nhưng hiệu quả, đảm bảo đáp ứng các chức năng quan trọng như đăng nhập hệ thống, quản lý thông tin nhân viên và theo dõi chấm công hàng ngày một cách chính xác và có hệ thống.

3.6.2. Cloudflare Tunnel

Cấu hình trên trang web của Cloudflare:

The screenshot shows the Cloudflare Tunnel configuration interface. It includes a section 'Name your tunnel' with a text input for 'Tunnel name' containing 'chamcongsql'. Below this is a section 'Edit public hostname for chamcongsql' with fields for 'Subdomain' (containing 'websql'), 'Domain' (a dropdown menu showing 'dinhkhanh.online'), 'Service Type' (a dropdown menu showing 'HTTP'), and 'URL' (containing 'localhost:80'). Red boxes and numbers 1 through 5 highlight these specific fields.

Hình 3.16 Các vị trí cần cấu hình trên trang web của Cloudflare

Hình 3.16 mô tả các vị trí cần điền trong ảnh cấu hình Cloudflare Tunnel:

1. Tunnel name: Tên cho tunnel của cần dùng (trong đề tài dùng tên: chamcongsql). Tên này dùng để quản lý và nhận diện tunnel trong Cloudflare.
2. Subdomain: Nhập tên subdomain muốn sử dụng (trong đề tài: websql). Subdomain này sẽ kết hợp với domain để tạo thành địa chỉ truy cập công khai.

3. Domain: Chọn hoặc nhập domain chính mà bạn đã cấu hình trên Cloudflare (trong đề tài: dinhkhanh.online). Subdomain ở trên sẽ được gắn vào domain này.

4. Type: Chọn loại dịch vụ mà tunnel sẽ chuyển tiếp. Ở đây chọn HTTP để chuyển tiếp các kết nối web.

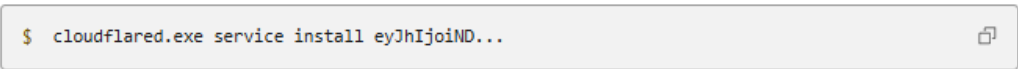
5. URL: Nhập địa chỉ và cổng của dịch vụ nội bộ mà tunnel sẽ kết nối đến. Ví dụ: localhost:80, tương ứng với máy chủ web đang chạy trên cổng 80 của máy cục bộ.

Cấu hình trên máy chủ cục bộ:

Sau khi cài đặt Cloudflared trên máy cục bộ, mở Terminal trên máy với quyền quản trị viên. Tiếp theo, chuyển đến thư mục chứa mã nguồn web.

Sau khi tạo tunnel trên giao diện web của Cloudflare, hệ thống sẽ cung cấp một dòng lệnh để cài đặt tunnel dưới dạng dịch vụ như **hình 3.17**.

4. Run the following command:



```
$ cloudflared.exe service install eyJhIjoND...
```

Hình 3.17 Dòng lệnh được cấp bởi hệ thống của Cloudflare

Sao chép và dán lệnh trên vào Terminal và chạy, dịch vụ sẽ tự động chạy mỗi khi máy tính được khởi động lại.

Kết quả: Người dùng từ bất kỳ đâu có thể truy cập dịch vụ nội bộ thông qua địa chỉ dạng: <https://websql.dinhkhanh.online>

3.7. Kết luận chương

Như vậy, chương này đã trình bày quá trình chuyển đổi từ lý thuyết sang thực tiễn trong việc thiết kế hệ thống chấm công ứng dụng nhận diện khuôn mặt. Các nội dung bao gồm xác định yêu cầu hệ thống, xây dựng sơ đồ khối, thiết kế khối xử lý, cùng với khối Web App. Hệ thống được triển khai dựa trên mã nguồn thực tế bằng Python (Tkinter) cho ứng dụng cục bộ và PHP (XAMPP) cho giao diện web, kết hợp Cloudflare Tunnel để hỗ trợ truy cập từ xa. Những nền tảng thiết kế này là bước quan trọng chuẩn bị cho giai đoạn xây dựng, kiểm thử và đánh giá hệ thống hoàn chỉnh trong chương tiếp theo.

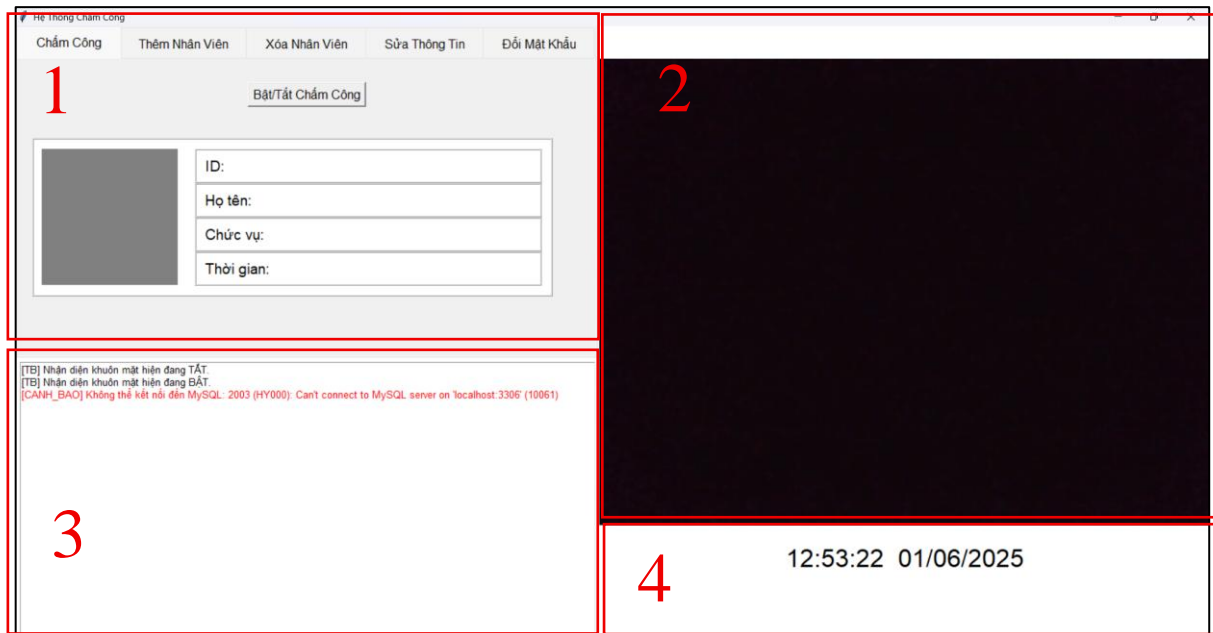
CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ

4.1. Giới thiệu chương

Chương này trình bày kết quả hoạt động của hệ thống chăm công khuôn mặt, bao gồm giao diện người dùng và các chức năng chính như thêm, xóa, sửa nhân viên, chăm công và đổi mật khẩu. Bên cạnh đó, chương còn đánh giá hiệu suất nhận diện khuôn mặt trong các điều kiện khác nhau, từ đó làm rõ ưu điểm và hạn chế của hệ thống.

4.2. Giao diện người dùng

Giao diện người dùng được thiết kế bằng Tkinter với bố cục rõ ràng. Các thành phần giao diện được sắp xếp hợp lý, đảm bảo tính thẩm mỹ và thuận tiện. Màu sắc và kiểu chữ được chọn phù hợp, tạo cảm giác dễ chịu cho người dùng. Giao diện mang phong cách tối giản, tập trung vào sự gọn gàng và dễ quan sát, giúp người dùng tiếp cận nhanh chóng mà không cần nhiều hướng dẫn. Kích thước cửa sổ và các thành phần được tối ưu để phù hợp với nhiều độ phân giải màn hình khác nhau, đảm bảo tính linh hoạt khi sử dụng trên nhiều thiết bị.

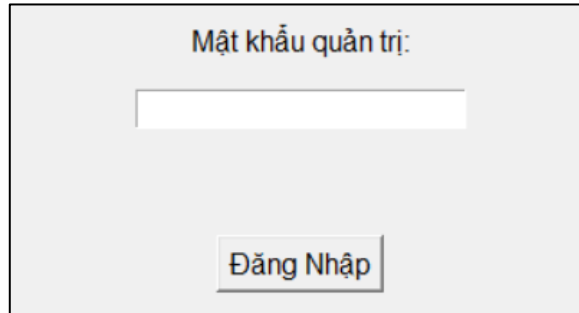


Hình 4.1 Giao diện chính của hệ thống

Hình 4.1 Mô tả giao diện chính của hệ thống với 4 khung:

Khung 1: Bao gồm các tab chức năng của hệ thống, là nơi hiển thị thông tin chăm công, các ô nhập liệu và các thành phần giao diện liên quan. Hầu hết các thao tác của người dùng, như nhập dữ liệu, xác nhận, chỉnh sửa hoặc xem thông tin, đều được thực hiện tại khu vực này.

Ngoài ra, giao diện đăng nhập ở **Hình 4.2** cũng xuất hiện tại khung này khi người dùng truy cập vào các tab quản trị như Thêm Nhân Viên, Xóa Nhân Viên hoặc Sửa Thông Tin. Giao diện này gồm một ô nhập liệu để người dùng điền mật khẩu và một nút "Đăng Nhập" bên dưới.

The image shows a login form with a light gray background. At the top, the text "Mật khẩu quản trị:" is displayed in a dark blue font. Below this text is a white rectangular input field for the password. At the bottom of the form is a button with the text "Đăng Nhập" in a dark blue font, enclosed in a thin black border.

Hình 4.2 Giao diện đăng nhập

Khi nhập đúng mật khẩu quản trị và đăng nhập thành công, giao diện đăng nhập sẽ tự động ẩn đi, thay thế bằng giao diện chức năng tương ứng của tab đó. Nếu nhập sai, hệ thống sẽ hiển thị thông báo lỗi ngay trên giao diện. Phần đăng nhập này giúp đảm bảo an toàn, ngăn chặn truy cập trái phép vào các tính năng quan trọng của hệ thống.


Khung 2: Hiển thị hình ảnh từ webcam theo thời gian thực, phục vụ nhận diện khuôn mặt. Khung này sử dụng một widget tk.Label để liên tục cập nhật và trình chiếu các khung hình (frame) từ camera, giúp người dùng quan sát trực tiếp quá trình nhận diện khuôn mặt hoặc thu thập dữ liệu khuôn mặt. Khi bật chức năng nhận diện, khung camera sẽ hiển thị hình ảnh kèm theo các khung nhận diện khuôn mặt, thông tin nhận diện (ID, xác suất, trạng thái). Ngoài ra, khung này còn hỗ trợ hiển thị các hiệu ứng khi thu thập dữ liệu hoặc khi nhận diện thành công, giúp người dùng dễ dàng theo dõi và thao tác với hệ thống.

Khung 3: Hiển thị các thông báo, trạng thái hoạt động, lỗi hoặc cảnh báo cho người dùng. Khung này được xây dựng bằng widget tk.Text và nằm ở phía dưới bên trái giao diện chính. Mỗi khi có sự kiện quan trọng như thêm, xóa, sửa nhân viên, chấm công, lỗi kết nối, hoặc các thao tác hệ thống khác, thông tin sẽ được ghi vào khung log này. Các dòng log được phân biệt màu sắc: thông báo lỗi hoặc cảnh báo hiển thị màu đỏ, còn các thông báo bình thường hiển thị màu đen. Khung log giúp người dùng dễ dàng theo dõi lịch sử hoạt động, phát hiện và xử lý sự cố kịp thời trong quá trình sử dụng hệ thống.

Khung 4: Một thành phần giao diện nằm ở phía trên bên phải cửa sổ chính, ngay dưới khung camera. Đồng hồ này sử dụng widget tk.Label để hiển thị giờ, phút, giây và ngày tháng hiện tại. Thời gian được cập nhật tự động mỗi giây nhờ hàm after của tkinter, giúp người dùng luôn theo dõi được thời gian thực khi sử dụng hệ thống. Đồng hồ góp phần tăng tính trực quan và tiện lợi cho giao diện chấm công.

4.2.1 Tab Chấm Công

Giao diện tab Chấm Công được thiết kế trực quan, giúp người dùng dễ dàng thực hiện và theo dõi quá trình chấm công bằng khuôn mặt.



Hình 4.3 Giao diện tab Chấm Công

Hình 4.3 Giao diện tab Chấm Công của hệ thống. Cụ thể:

Nút "Bật/Tắt Chấm Công": Được đặt nổi bật phía trên, cho phép người dùng chủ động kích hoạt hoặc tạm dừng chức năng nhận diện khuôn mặt để chấm công. Khi nhấn nút này, hệ thống sẽ bắt đầu hoặc dừng việc nhận diện khuôn mặt từ camera.

Khung thông tin nhân viên:

- Bên trái: Là ô hiển thị ảnh khuôn mặt của nhân viên vừa được nhận diện. Nếu chưa có nhận diện, ô này sẽ hiển thị màu xám mặc định.
- Bên phải: Là các dòng thông tin được trình bày rõ ràng, bao gồm:
 - ID: Hiển thị mã số nhân viên vừa chấm công.
 - Họ tên: Hiển thị tên đầy đủ của nhân viên.
 - Chức vụ: Hiển thị chức vụ của nhân viên.
 - Thời gian: Hiển thị thời điểm chấm công (ngày, giờ) tại thời điểm đó.

Khi hệ thống nhận diện thành công một khuôn mặt, toàn bộ thông tin và ảnh sẽ được cập nhật ngay lập tức lên khung này, giúp người dùng xác nhận lại kết quả chấm công. Các thành phần được căn chỉnh hợp lý, dễ quan sát, đảm bảo thao tác nhanh chóng và thuận tiện cho người dùng trong môi trường thực tế. Giao diện này giúp quá trình chấm công trở nên minh bạch, trực quan và giảm thiểu sai sót khi nhận diện nhân viên.

4.2.2. Tab Thêm Nhân Viên

Tab Thêm Nhân Viên là nơi cho phép quản trị viên nhập thông tin để thêm một nhân viên mới vào hệ thống.

Chấm Công	Thêm Nhân Viên	Xóa Nhân Viên	Sửa Thông Tin	Đổi Mật Khẩu
<div>ID:</div> <div><input type="text"/></div> <div>Họ tên:</div> <div><input type="text"/></div> <div>Chức vụ:</div> <div><input type="text"/></div> <div>Xác nhận thêm</div>				

Hình 4.4 Giao diện tab Thêm Nhân Viên

Hình 4.4 Giao diện tab Thêm Nhân Viên của hệ thống với các thành phần chính:

Các trường nhập liệu:

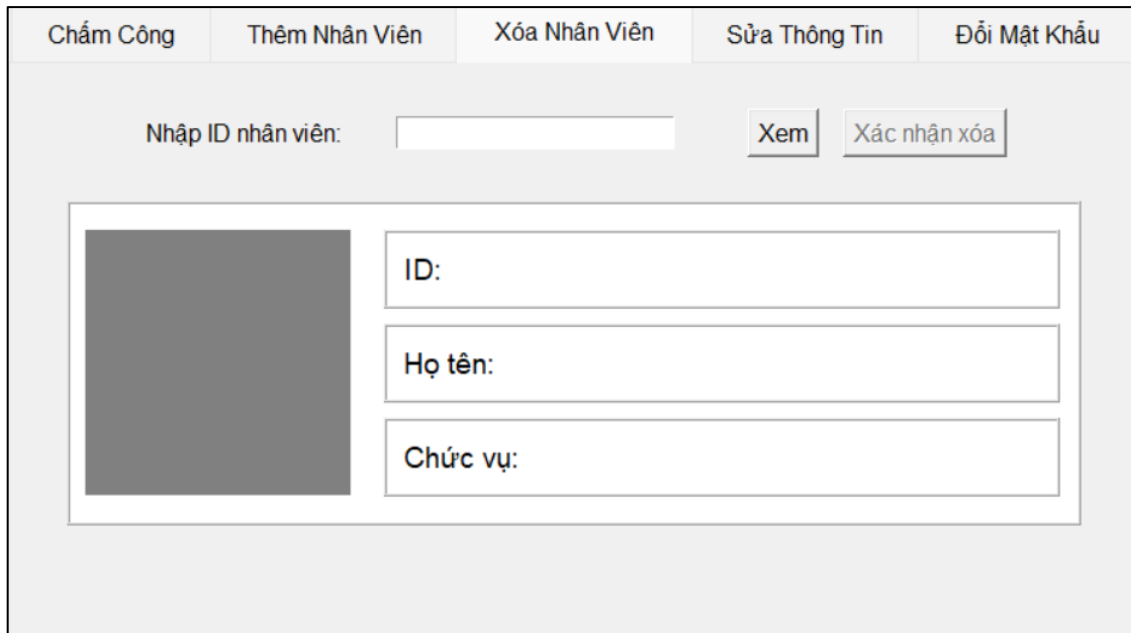
- ID: Ô nhập để điền mã số nhân viên.
- Họ tên: Ô nhập để điền họ tên đầy đủ của nhân viên.
- Chức vụ: Ô nhập để điền chức vụ của nhân viên.

Nút "Xác nhận thêm": Khi điền đầy đủ thông tin và nhấn nút này, hệ thống sẽ kiểm tra hợp lệ, sau đó tiến hành lưu thông tin nhân viên vào các bảng trong cơ sở dữ liệu MySQL, file CSV và dữ liệu khuôn mặt trong thư mục Dataset, sau đó bắt đầu quá trình thu thập dữ liệu khuôn mặt cho nhân viên mới.

Tab này giúp quản trị viên dễ dàng bổ sung nhân sự vào hệ thống, đảm bảo thông tin được nhập đầy đủ và chính xác trước khi lưu trữ.

4.2.3. Tab Xóa Nhân Viên

Tab Xóa Nhân Viên là nơi cho phép quản trị viên tìm kiếm và xóa thông tin của một nhân viên khỏi hệ thống.



Hình 4.5 Giao diện tab Xóa Nhân Viên

Hình 4.5 Giao diện tab Xóa Nhân Viên của hệ thống với các thành phần chính:

Ô nhập ID nhân viên: Người dùng nhập mã số nhân viên cần xóa vào ô này.

Nút "Xem": Sau khi nhập ID, nhấn nút này để tra cứu thông tin nhân viên.

Nút "Xác nhận xóa": Chỉ được kích hoạt khi tìm thấy nhân viên, cho phép xác nhận thao tác xóa.

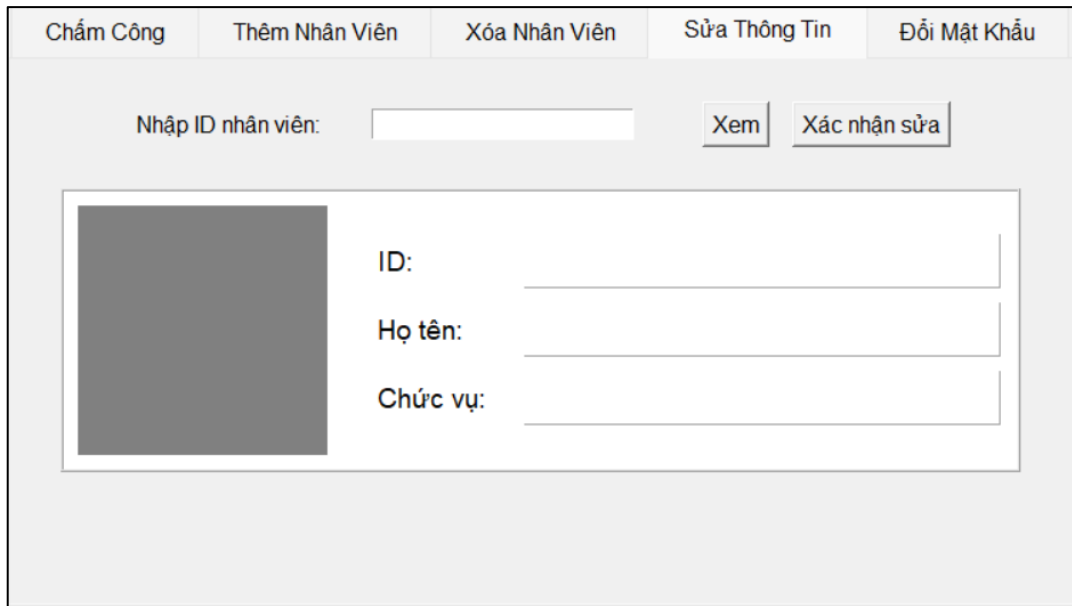
Khung thông tin nhân viên:

- Bên trái: Hiển thị ảnh khuôn mặt của nhân viên (hoặc khung xám nếu chưa có ảnh).
- Bên phải: Hiển thị các thông tin:
 - ID: Mã số nhân viên
 - Họ tên: Tên nhân viên
 - Chức vụ: Chức vụ của nhân viên

Khi xác nhận xóa, hệ thống sẽ xóa toàn bộ thông tin nhân viên khỏi các bảng trong cơ sở dữ liệu MySQL, file CSV và dữ liệu khuôn mặt trong thư mục Dataset. Tab này giúp đảm bảo việc xóa nhân viên được thực hiện chính xác, an toàn và có xác nhận rõ ràng từ người quản trị.

4.2.4. Tab Sửa Thông Tin

Tab Sửa Thông Tin là nơi cho phép quản trị viên chỉnh sửa thông tin của một nhân viên trong hệ thống.



Hình 4.6 Giao diện tab Sửa Thông Tin

Hình 4.6 Giao diện tab Sửa Thông Tin với các thành phần chính:

Ô nhập ID nhân viên: Người dùng nhập mã số nhân viên cần sửa vào ô này.

Nút "Xem": Sau khi nhập ID, nhấn nút này để tra cứu và hiển thị thông tin hiện tại của nhân viên.

Nút "Xác nhận sửa": Sau khi chỉnh sửa thông tin, nhấn nút này để lưu lại các thay đổi vào hệ thống.

Khung thông tin nhân viên:

- Bên trái: Hiển thị ảnh khuôn mặt của nhân viên (hoặc khung xám nếu chưa có ảnh).
- Bên phải: Hiển thị các trường thông tin có thể chỉnh sửa, bao gồm:
 - ID: Mã số nhân viên (có thể sửa).
 - Họ tên: Tên nhân viên (có thể sửa).
 - Chức vụ: Chức vụ của nhân viên (có thể sửa).

Khi xác nhận sửa, hệ thống sẽ cập nhật thông tin nhân viên vào các bảng trong cơ sở dữ liệu MySQL, file CSV và dữ liệu khuôn mặt trong thư mục Dataset. Tab này giúp đảm bảo việc chỉnh sửa thông tin nhân viên được thực hiện dễ dàng, chính xác và đồng bộ trên toàn hệ thống.

4.2.5. Tab **Đổi mật khẩu**

Tab Đổi Mật Khẩu là nơi cho phép quản trị viên thay đổi mật khẩu đăng nhập hệ thống một cách an toàn.

Chấm Công	Thêm Nhân Viên	Xóa Nhân Viên	Sửa Thông Tin	Đổi Mật Khẩu
<div>Mật khẩu cũ: <input type="password"/></div> <div>Mật khẩu mới: <input type="password"/></div> <div>Nhập lại mật khẩu mới: <input type="password"/></div> <div><input type="button" value="Gửi mã OTP"/></div>				

Hình 4.7 Giao diện nhập mật khẩu cũ, mới

Chấm Công	Thêm Nhân Viên	Xóa Nhân Viên	Sửa Thông Tin	Đổi Mật Khẩu
<div>Đã gửi mã OTP tới email admin. Vui lòng kiểm tra và nhập mã OTP.</div> <div>Nhập mã OTP: <input type="text"/></div> <div><input type="button" value="Xác nhận đổi mật khẩu"/></div>				

Hình 4.8 Giao diện nhập mã OTP

Giao diện tab này gồm hai phần chính:

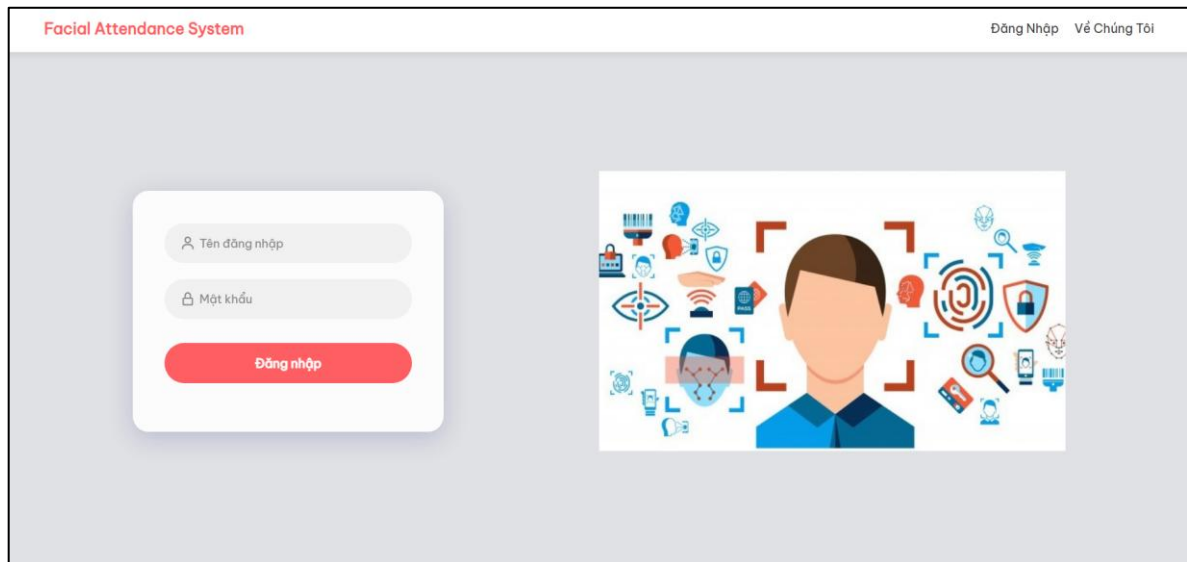
Phần 1 được mô tả trong **Hình 4.7**: Hiện thị các ô nhập liệu để quản trị viên nhập mật khẩu cũ, mật khẩu mới và xác nhận lại mật khẩu mới. Sau khi điền đầy đủ thông tin, nhấn nút "Gửi mã OTP" để nhận mã xác thực OTP gửi về email người quản trị.

Phần 2 được mô tả trong **Hình 4.8**: Khi mã OTP đã được gửi, giao diện chuyển sang ô nhập mã OTP. Quản trị viên nhập mã OTP nhận được từ email và nhấn "Xác nhận đổi mật khẩu" để hoàn tất quá trình thay đổi.

Tab này giúp đảm bảo việc đổi mật khẩu được thực hiện bảo mật, yêu cầu xác thực qua email để tránh thay đổi trái phép. Thông báo trạng thái (thành công, lỗi, hướng dẫn) sẽ hiển thị rõ ràng ngay trên giao diện.

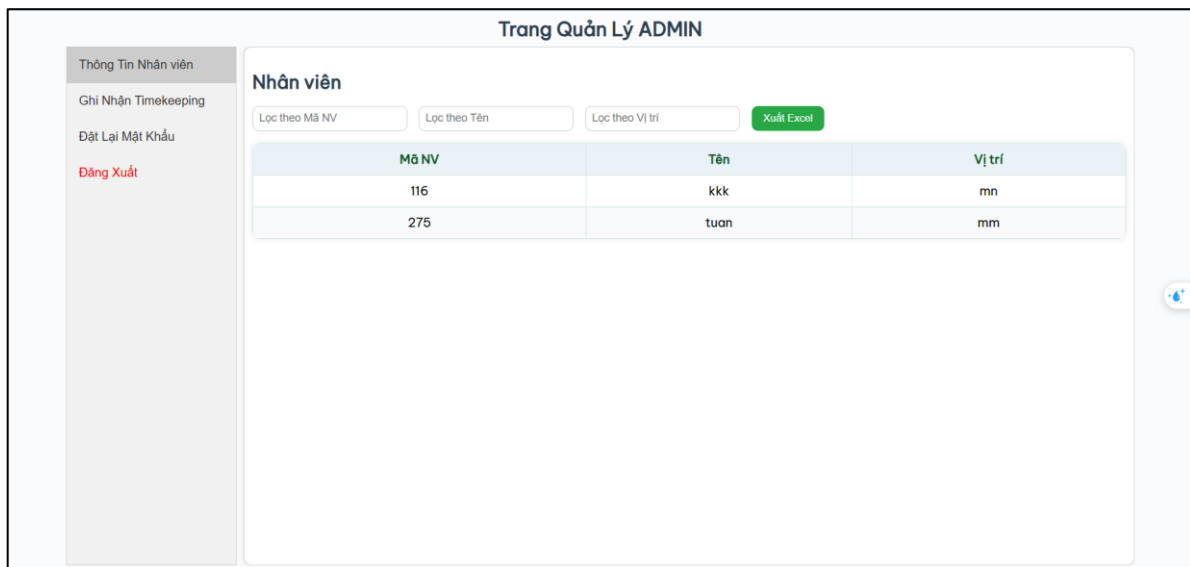
4.3 Giao diện web

Hệ thống chấm công bằng khuôn mặt được thiết kế với giao diện đơn giản, dễ sử dụng và thân thiện với người dùng.



Hình 4.9 Giao diện trang đăng nhập

Ngay từ trang đầu tiên ở **hình 4.9**, người quản lý sẽ thấy giao diện đăng nhập bao gồm hai trường nhập liệu là tên đăng nhập và mật khẩu, cùng với nút “Đăng nhập” nổi bật màu đỏ. Bên cạnh đó là hình ảnh minh họa về công nghệ nhận diện khuôn mặt, giúp thể hiện rõ mục đích và công nghệ chính của hệ thống. Phía trên cùng của giao diện có thanh menu điều hướng nhanh với các lựa chọn như “Đăng Nhập” và “Về Chúng Tôi”.



Hình 4.10 Giao diện tab thông tin nhân viên

Sau khi đăng nhập thành công với vai trò quản trị viên, người dùng sẽ được chuyển đến trang quản lý nhân viên như **hình 4.10**. Tại đây, hệ thống hiển thị danh sách nhân viên với đầy đủ thông tin như mã nhân viên, họ tên và vị trí công việc. Giao diện có tích hợp bộ lọc theo mã, tên và vị trí để thuận tiện cho việc tìm kiếm thông tin. Ngoài ra, hệ thống còn hỗ trợ xuất danh sách nhân viên ra file Excel, giúp lưu trữ và xử lý dữ liệu dễ dàng hơn.

Trang Quản Lý ADMIN

Lịch sử chấm công nhân viên

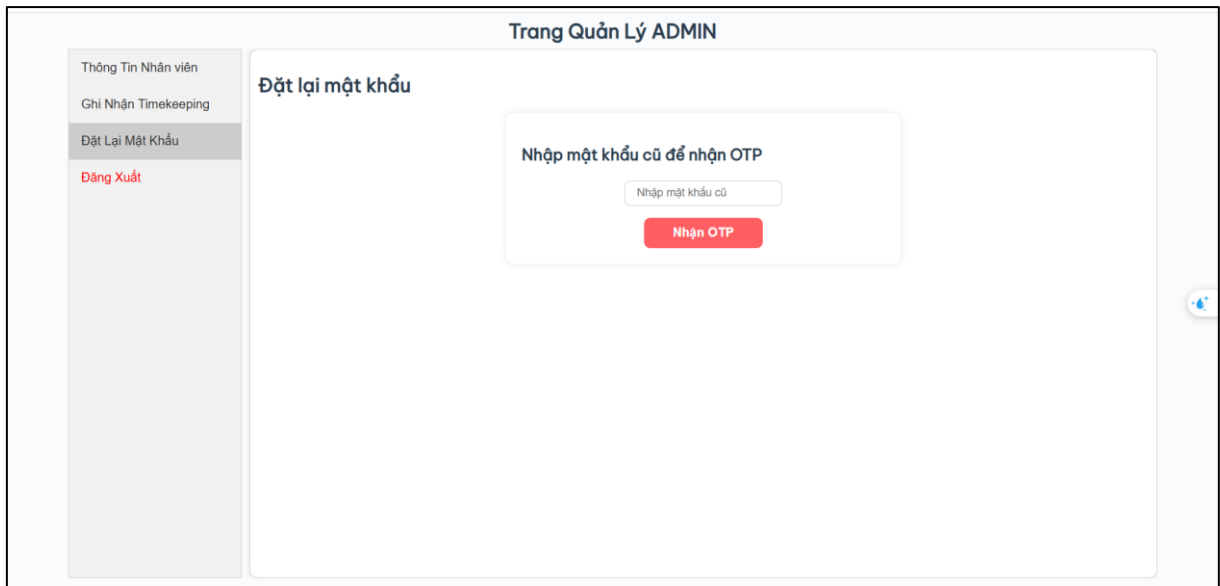
Tháng: Tháng 5 Năm: 2025

Lọc theo Mã NV Lọc theo Tên Lọc theo Vị trí **Xuất Excel**

Mã NV	Tên	Vị trí	Ngày	Check1	Check2	Check3	Check4	Check5	Check6	Check7	Check8	Check9
116	kkk	mn	23/05/2025	22:08:42	-	-	-	-	-	-	-	-
275	tuân	mm	27/05/2025	15:38:19	15:40:19	-	-	-	-	-	-	-
116	kkk	mn	27/05/2025	21:19:32	21:22:37	21:24:12	21:30:06	21:30:42	21:31:12	21:31:42	21:32:17	21:32:49

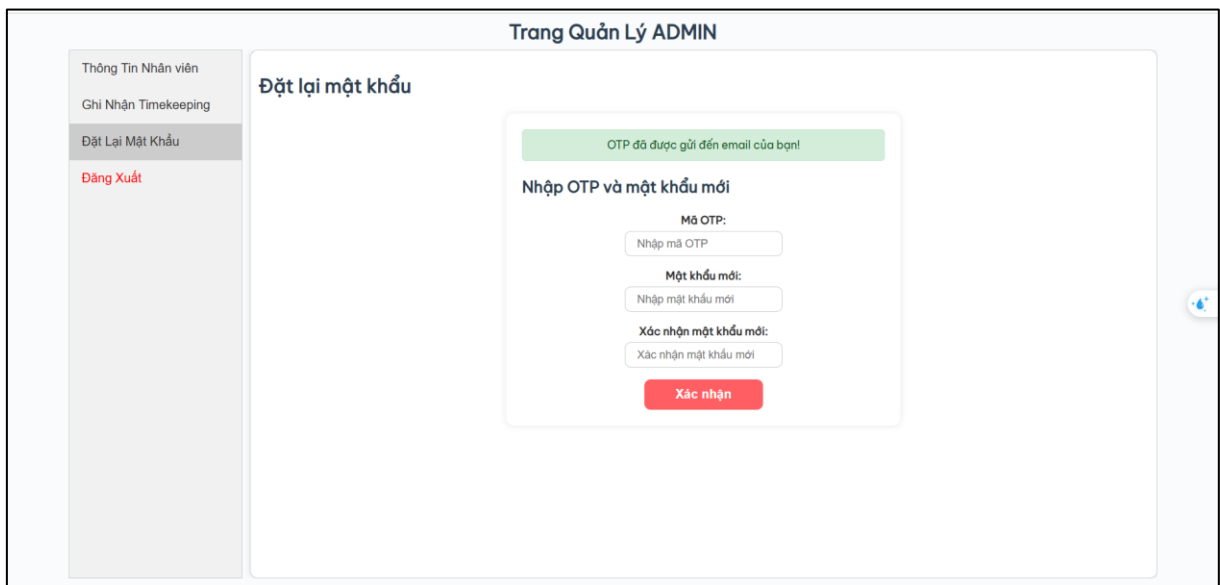
Hình 4.11 Giao diện tab ghi nhận chấm công

Trang tiếp theo là phần ghi nhận chấm công như **hình 4.11**, nơi lưu lại lịch sử chấm công của từng nhân viên. Dữ liệu được trình bày theo bảng với các thông tin như mã nhân viên, họ tên, ngày chấm công và các cột tương ứng với các mốc thời gian trong ngày. Quản trị viên có thể chọn tháng, năm và sử dụng bộ lọc theo mã, tên, hoặc vị trí để theo dõi chi tiết thời gian làm việc của từng nhân viên. Chức năng xuất Excel cũng được tích hợp để lưu trữ dữ liệu chấm công.



Hình 4.12 Giao diện nhập mật khẩu cũ

Ngoài ra, hệ thống còn cung cấp chức năng đặt lại mật khẩu nhằm tăng cường tính bảo mật. Trong bước đầu tiên ở **hình 4.12**, người dùng phải nhập mật khẩu cũ để nhận mã OTP gửi về email. Sau khi nhận được OTP, người dùng sẽ chuyển sang bước hai ở **Hình 4.13**, nơi họ nhập mã OTP, mật khẩu mới và xác nhận lại mật khẩu mới. Khi nhấn nút “Xác nhận”, mật khẩu sẽ được cập nhật thành công.



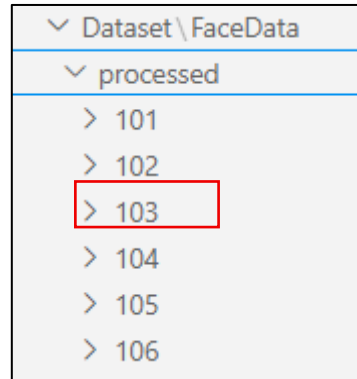
Hình 4.13 Giao diện nhập mã OTP và mật khẩu mới

Toàn bộ giao diện hệ thống được thiết kế nhất quán với tông màu nhẹ nhàng và rõ ràng. Thanh điều hướng bên trái giúp người dùng truy cập nhanh vào các chức năng như thông tin nhân viên, ghi nhận thời gian làm việc, đặt lại mật khẩu và đăng xuất. Với các chức năng đầy đủ và tiện lợi, hệ thống chăm công bằng khuôn mặt không chỉ giúp tự

động hóa quy trình điểm danh mà còn hỗ trợ hiệu quả cho công tác quản lý nhân sự trong doanh nghiệp.

4.4. Kết quả thao tác Xóa nhân viên

Trong cơ sở dữ liệu ban đầu của hệ thống chấm công có 06 thông tin của 06 Nhân viên, được thể hiện trong **hình 4.14** là nơi lưu thông tin 30 ảnh/01 Nhân viên



Hình 4.14 Thư mục lưu thông tin ảnh của Nhân viên

Khi chưa xoá, thư mục chứa các ảnh của nhân viên có ID 103 vẫn được lưu trong thư mục Dataset/FaceData/processed/103 như **hình 4.14**.

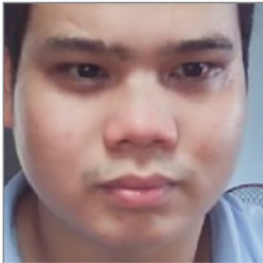
Database > employees.csv	
1 employee_id,name,position	
2 101,Phucc,iot	
3 105,khang,bv	
4 104,tuan,gd	
5 102,PhanCo,sv	
6 106,phi,cv	
7 103,khanh,ceo	

employee_id	name	position
101	Phucc	iot
102	PhanCo	sv
103	khanh	ceo
104	tuan	gd
105	khang	bv
106	phi	cv

Hình 4.15 Thông tin Nhân viên trong CSV và MySQL

Các thông tin của nhân viên đó vẫn tồn tại trong file employees.csv và trong bảng employees của MySQL như **hình 4.15**.

Nhập ID nhân viên:



ID: 103

Họ tên: khanh

Chức vụ: ceo

Hình 4.16 Kiểm tra thông tin của Nhân viên 103

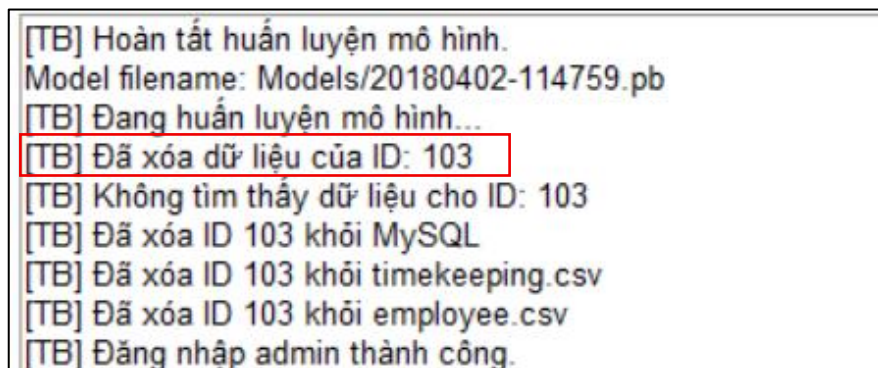
Kiểm tra thông tin của Nhân viên 103 bằng cách Nhập ID 103 và bấm xem, thông tin được hiển thị đầy đủ trên giao diện chính bao gồm ID/Họ tên/Chức vụ như **hình 4.16**.



The screenshot shows a web interface with a top navigation bar containing five tabs: "Chấm Công", "Thêm Nhân Viên", "Xóa Nhân Viên", "Sửa Thông Tin", and "Đổi Mật Khẩu". Below the tabs, there is a search section with the label "Nhập ID nhân viên:" followed by a text input field containing "103". To the right of the input field are two buttons: "Xem" and "Xác nhận xóa". Below the search section, a red error message "Không tìm thấy nhân viên!" is displayed. Below the error message, there is a form with three input fields: "ID:", "Họ tên:", and "Chức vụ:". The "ID:" field is currently empty.

Hình 4.17 Kiểm tra thông tin của Nhân viên 103 sau khi xóa

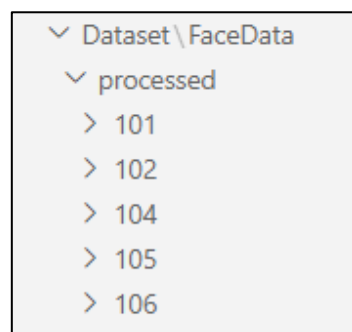
Sau khi xóa, kiểm tra thông tin của Nhân viên 103 thì hiện thông báo không tìm thấy nhân viên trên giao diện chính như **hình 4.17**.



The screenshot shows a log window with the following text: "[TB] Hoàn tất huấn luyện mô hình. Model filename: Models/20180402-114759.pb", "[TB] Đang huấn luyện mô hình...", "[TB] Đã xóa dữ liệu của ID: 103", "[TB] Không tìm thấy dữ liệu cho ID: 103", "[TB] Đã xóa ID 103 khỏi MySQL", "[TB] Đã xóa ID 103 khỏi timekeeping.csv", "[TB] Đã xóa ID 103 khỏi employee.csv", and "[TB] Đăng nhập admin thành công." The line "[TB] Đã xóa dữ liệu của ID: 103" is highlighted with a red box.


Hình 4.18 Thông báo Log khi xóa thành công Nhân viên 103

Thông báo Log hiển thị thông tin đã xóa thành công dữ liệu của Nhân viên 103 và sau đó huấn luyện lại mô hình nhận diện được hiển thị như Hình 4.18.



Hình 4.19 Thư mục lưu thông tin ảnh của Nhân viên sau khi xóa 103

Sau khi xóa thành công thì dữ liệu thông tin ảnh của Nhân viên 103 đã được xóa trong thư mục Dataset/FaceData/ chứa ảnh của nhân viên, chỉ còn thông tin của 05 Nhân viên được hiển thị như **hình 4.19**.

Database >  employees.csv	employee_id	name	position
1 employee_id,name,position	101	Phucc	iot
2 101,Phucc,iot	102	PhanCo	sv
3 105,khang,bv	104	tuan	gd
4 104,tuan,gd	105	khang	bv
5 102,PhanCo,sv	106	phi	cv
6 106,phi,cv			

Hình 4.20 Thông tin Nhân viên trong CSV và MySQL sau khi xóa 103

Các thông tin của nhân viên 103 sau khi được xóa thì ID/Họ tên/Chức vụ cũng được xóa lần lượt trong file employees.csv và bảng employees của MySQL như **hình 4.20**.

Tương tự nếu thông tin của Nhân viên 103 đã được chấm công và lưu trong file timekeeping.csv và trong bảng timekeeping củ MySQL thì cũng sẽ được xóa lần lượt.

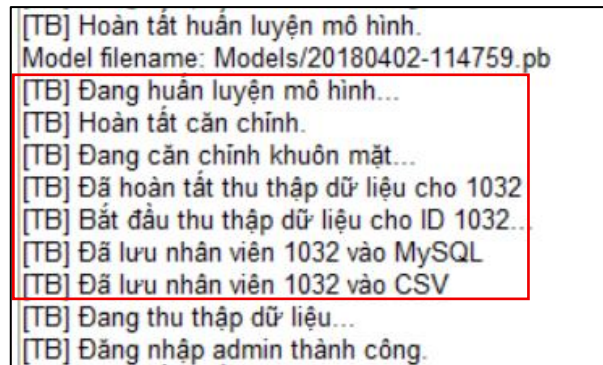
4.5. Kết quả thao tác Thêm nhân viên

Sau khi xóa nhân viên 103 ở trên, vào Tab ‘Thêm nhân viên’ để thực hiện thêm một nhân viên mới có ID 1032.

Chấm Công	Thêm Nhân Viên	Xóa Nhân Viên	Sửa Thông Tin	Đổi Mật Khẩu
<p>ID:</p> <p>1032</p> <p>Họ tên:</p> <p>khanh</p> <p>Chức vụ:</p> <p>ceo</p> <p>Xác nhận thêm</p>				

Hình 4.21 Thao tác Thêm nhân viên mới 1032

Thực hiện thao tác nhập các thông tin ID/Họ tên/Chức vụ cho Nhân viên 1032 vào các ô trong giao diện của hệ thống, được thể hiện như **hình 4.21**.

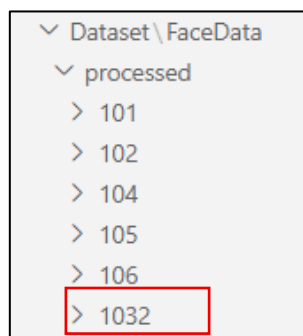


Hình 4.22 Thông báo Log khi thêm thành công Nhân viên 1302

Sau khi xác nhận thêm thông tin cho Nhân viên mới 1032, thông tin của nhân viên đó được lưu vào CSV và MySQL, đồng thời ảnh của người đó được thu thập, căn chỉnh, huấn luyện mô hình như thông báo ở **hình 4.22**.

Hình 4.23 Kiểm tra thông tin của Nhân viên mới 1302

Sau khi thêm thông tin thành công, tiến hành tìm kiếm và kiểm tra thông tin của Nhân viên mới 1032 để đối sánh với thông tin đã nhập vào như Hình 4.21, được hiển thị đầy đủ trong **hình 4.23**.



Hình 4.24 Thư mục lưu thông tin ảnh của Nhân viên sau khi thêm 1032

Database > employees.csv	
1 employee_id,name,position	
2 101,Phucc,iot	
3 105,khang,bv	
4 104,tuan,gd	
5 102,PhanCo,sv	
6 106,phi,cv	
7 1032,khanh,ceo	

employee_id	name	position
101	Phucc	iot
102	PhanCo	sv
104	tuan	gd
105	khang	bv
106	phi	cv
1032	khanh	ceo

Hình 4.25 Thông tin Nhân viên trong CSV và MySQL sau khi thêm nhân viên 1032

Dữ liệu của Nhân viên mới trong thư mục ảnh Dataset/FaceData, trong file employees.csv và bảng employees trong MySQL cũng được thêm vào đầy đủ, thể hiện trong **hình 4.24** và **hình 4.25**.

4.6. Kết quả thao tác Sửa thông tin

Sau khi thêm nhân viên mới 1032 được thực hiện ở mục 4.5, tiến hành Sửa thông tin của nhân viên này.

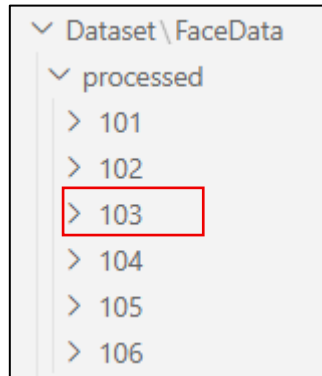
Hình 4.26 Thông tin trước và sau khi chỉnh sửa của Nhân viên 1032.

Thông tin Nhân viên 1032 (**hình 4.26 bên trái**), thông tin ID thay đổi từ ‘1032’ sang ‘103’, Họ tên thay đổi từ ‘khanh’ sang ‘khanh2’, Chức vụ thay đổi từ ‘ceo’ sang ‘ceo2’. Sau khi nhập đủ thông tin và bấm nút ‘Xác nhận sửa’ thì thông tin hiện tại được cập nhật thành công (**hình 4.26 bên phải**).

```
[TB] Hoàn tất huấn luyện mô hình.
Model filename: Models/20180402-114759.pb
[TB] Đang huấn luyện mô hình...
[TB] Đã bắt đầu huấn luyện lại mô hình sau khi sửa thông tin nhân viên 1032 thành 103
[TB] Đã đổi tên thư mục processed_path từ 1032 thành 103
[TB] Đã đổi tên thư mục raw_path từ 1032 thành 103
[TB] Đã cập nhật thông tin nhân viên 1032 trong MySQL
[TB] Đã cập nhật thông tin nhân viên 1032 trong employees.csv
[TB] Đăng nhập admin thành công.
```

Hình 4.27 Thông báo Log sau khi sửa thông tin nhân viên 1302

Sau khi xác nhận sửa thành công thông tin cho Nhân viên 1032 thành 103, thông tin của nhân viên đó được lưu vào CSV và MySQL, đồng thời huấn luyện lại mô hình như thông báo ở **hình 4.27**



Hình 4.28 Thư mục lưu thông tin ảnh của Nhân viên 1032 đã sửa thông tin

Database > employees.csv	
1 employee_id,name,position	
2 101,Phucc,iot	
3 105,khang,bv	
4 104,tuan,gd	
5 102,PhanCo,sv	
6 106,phi,cv	
7 103,khanh2,ceo2	

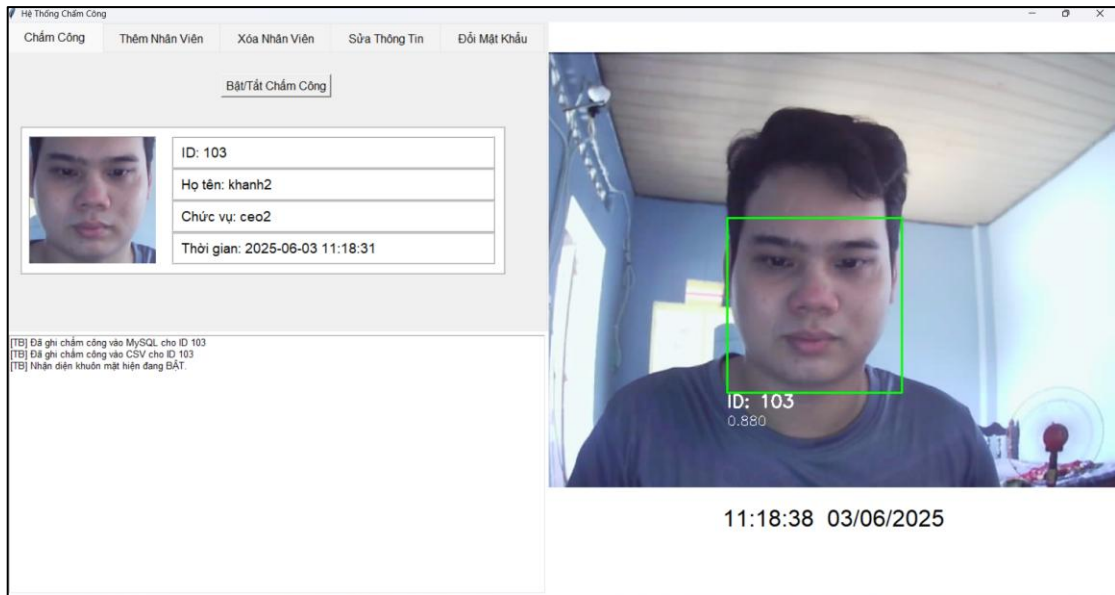
employee_id	name	position
101	Phucc	iot
102	PhanCo	sv
103	khanh2	ceo2
104	tuan	gd
105	khang	bv
106	phi	cv

Hình 4.29 Thông tin Nhân viên trong CSV và MySQL sau khi sửa

Dữ liệu thay đổi cho Nhân viên 1032 thành 103 trong thư mục ảnh Dataset/FaceData, trong file employees.csv và bảng employees trong MySQL cũng được sửa lại đầy đủ, thể hiện trong **hình 4.28** và **hình 4.29**.

4.7. Kết quả thao tác Chấm công

Tiến hành thực hiện Chấm công cho Nhân viên 103 đã được thay đổi từ trước đó.



Hình 4.30 Các thông tin hiển thị kết quả chấm công trên giao diện

Sau khi tiến hành chấm công, thông tin của nhân viên 103 được hiển thị trên giao diện như **hình 4.30** bao gồm ảnh tại thời điểm chấm công, ID, họ tên, chức vụ và thời gian chấm công. Đồng thời ở khung video, khuôn mặt nhân viên được tạo bounding box kèm theo ID và độ chính xác. Các thông báo chấm công cũng được hiển thị ở log.

Database >	timekeeping.csv
1	employee_id,name,position,date,time
2	103,khanh2,ceo2,2025-06-03,11:18:31
3	103,khanh2,ceo2,2025-06-03,11:19:08

Hình 4.31 Thông tin chấm công trong file 'timekeeping.csv'

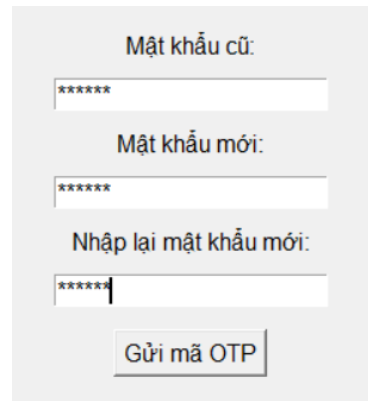
employee_id	name	position	date	time
103	khanh2	ceo2	2025-06-03	11:18:31
103	khanh2	ceo2	2025-06-03	11:19:08

Hình 4.32 Thông tin chấm công ở bảng 'timekeeping' trong MySQL

Lúc này, trong file timekeeping.csv và bảng timekeeping của MySQL cũng đã lưu thông tin chấm công của nhân viên 103 bao gồm ID, họ tên, chức vụ, ngày, giờ được thể hiện trong **hình 4.31** và **hình 4.32**.

4.8. Kết quả thao tác Đổi mật khẩu

Khi đổi mật khẩu, hệ thống yêu cầu nhập mật khẩu cũ và nhập 2 lần mật khẩu mới.



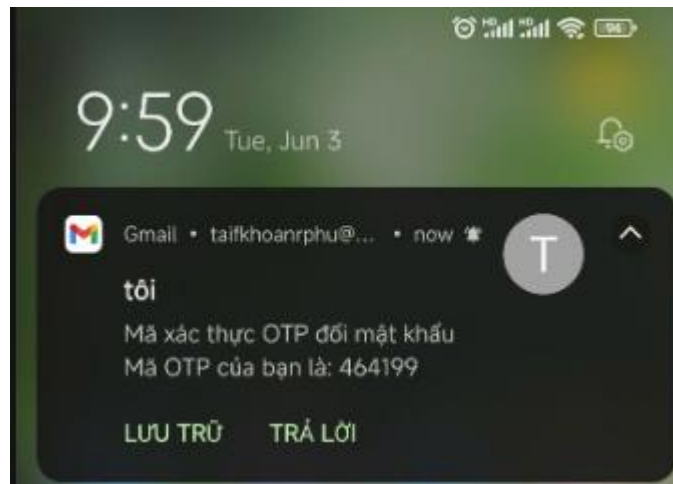
Mật khẩu cũ:

Mật khẩu mới:

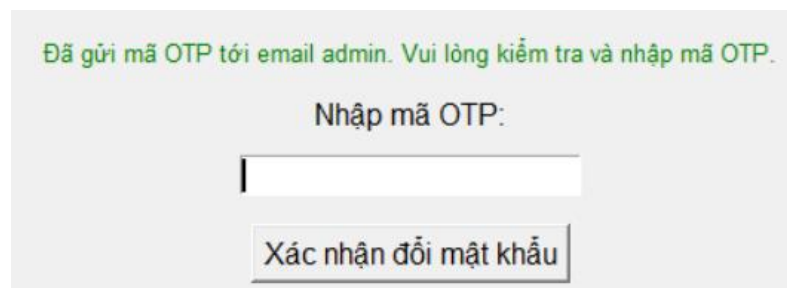
Nhập lại mật khẩu mới:

Gửi mã OTP

Hình 4.33 Ô nhập mật khẩu cũ và mới



Hình 4.34 Mã OTP được gửi về Email



Đã gửi mã OTP tới email admin. Vui lòng kiểm tra và nhập mã OTP.

Nhập mã OTP:

Xác nhận đổi mật khẩu

Hình 4.35 Ô chờ nhập OTP trên giao diện Tkinter

Sau khi người dùng hoàn thành việc nhập đầy đủ các thông tin cần thiết vào tất cả các ô trong **hình 4.33** và xác nhận, một mã OTP sẽ được gửi đến địa chỉ email đã được thiết lập trước đó như **hình 4.34**.

Nhập mã OTP như trong **hình 4.35** và hệ thống sẽ thực hiện đối sánh sự trùng khớp của mã. Sau khi mã OTP được kiểm tra đầy đủ và chính xác, người dùng chỉ cần xác nhận thao tác để hoàn tất quá trình đổi mật khẩu.

4.9. Đánh giá

4.9.1. Đánh giá mô hình huấn luyện

4.9.1.1. Mô tả tập dữ liệu đánh giá

- Dữ liệu được lấy từ 34 người khác nhau, với 28 người được lấy từ trang dữ liệu của Github [10], Kaggle [11] [12] và 6 người là nhóm tự thu thập, trong đó:

- 30 người có 30 ảnh/1 người.
- 01 người có 18 ảnh.
- 01 người có 25 ảnh.
- 02 người có 16 ảnh/1 người.

- Tổng số lượng ảnh là 975 images.

- Tên của mỗi người được chuyển định dạng thành các mã ID để thuận tiện trong quá trình xử lý. (ví dụ 101, 102, 103, ..., 134).

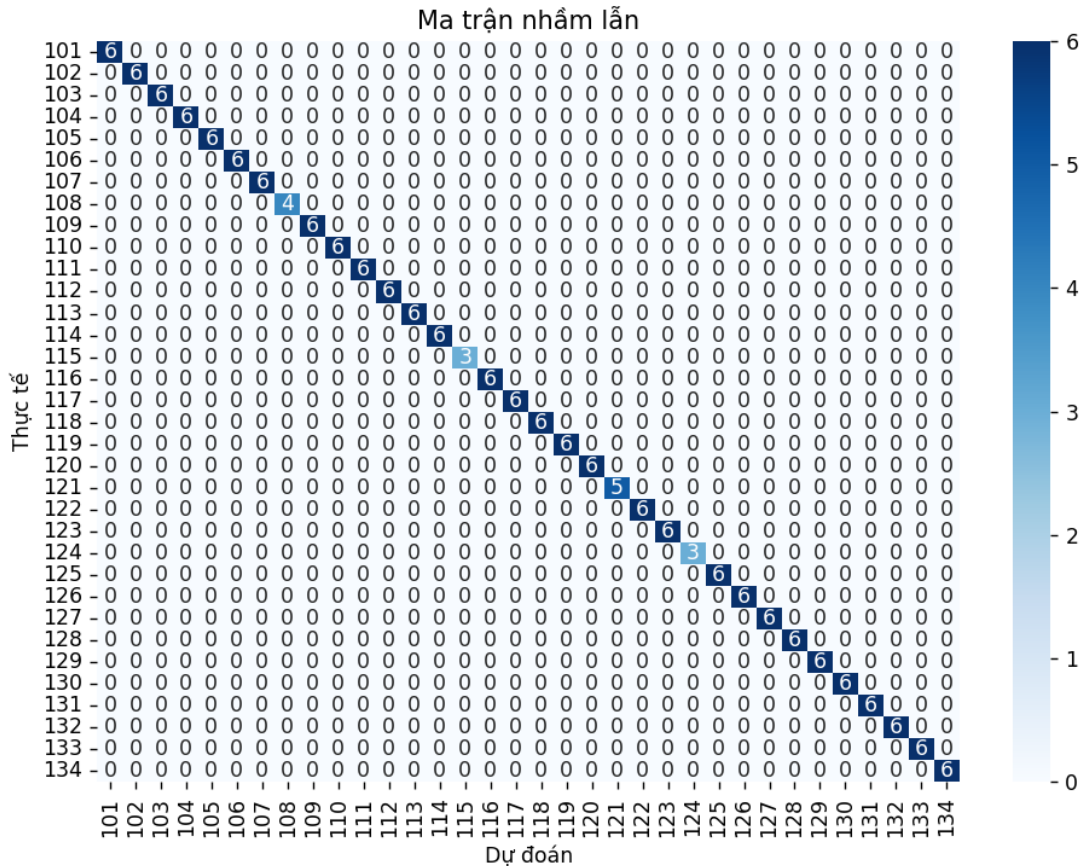
- Dữ liệu ảnh thô ban đầu được tiền xử lý, cắt tách ảnh chỉ giữ lại khuôn mặt với kích thước 160x160 px.

4.9.1.2. Cách thực hiện đánh giá

- Phương pháp thực hiện đánh giá sử dụng: Hold-out cross-validation.

- Tỷ lệ được phân chia 80% train và 20% test trên tổng 975 ảnh.

4.9.1.3. Kết quả đánh giá



Hình 4.36 Ma trận nhầm lẫn của mô hình

Ma trận nhầm lẫn thu được cho thấy mô hình nhận diện khuôn mặt hoạt động với độ chính xác rất cao. Hầu hết các giá trị đều tập trung trên đường chéo chính, chứng tỏ mô hình đã dự đoán đúng gần như tuyệt đối cho từng lớp khuôn mặt. Cụ thể, đa số các lớp đều có 6 ảnh được nhận diện chính xác – đây là số lượng ảnh tối đa của mỗi lớp trong tập kiểm thử.

Tuy nhiên, vẫn có một vài lớp có số lần nhận diện đúng nhỏ hơn 6, như lớp 108, 115, 124 hoặc 132. Nguyên nhân chủ yếu không phải do mô hình nhầm lẫn, mà chủ yếu do thiếu dữ liệu ảnh đầu vào ở những lớp này. Điều này được củng cố bởi thực tế là gần như không có giá trị nào ngoài đường chéo, cho thấy mô hình rất ít khi dự đoán sai nhãn.

Từ các kết quả trên, có thể kết luận rằng mô hình có hiệu suất nhận diện khuôn mặt rất tốt, đáng tin cậy và ổn định. Sai lệch nhỏ xuất hiện không đến từ lỗi thuật toán mà do sự thiếu đồng đều trong phân phối dữ liệu. Vì vậy, để tăng độ bao phủ và độ tin cậy chung của hệ thống, nên bổ sung thêm dữ liệu ảnh cho các lớp còn thiếu nhằm đảm bảo mỗi lớp có đầy đủ số lượng mẫu kiểm thử tương đương nhau.

4.9.2. Đánh giá các yếu tố thay đổi cùng một người

Nhóm tiến hành thử nghiệm đo ở nhiều trường hợp đối với một người để xem xét độ chính xác nhận diện.

Bảng 4.1 Đánh giá độ chính xác các yếu tố thay đổi cùng một người

Lần đo	Thay đổi biểu cảm	Đeo kính	Ngược sáng	Che mặt
1	0.80	0.77	0.79	Không thể nhận diện chính xác. Nhận nhầm sang người khác với độ chính xác từ 0.2 đến 0.4.
2	0.88	0.69	0.86	
3	0.87	0.74	0.70	
4	0.90	0.72	0.82	
5	0.76	0.73	0.83	
6	0.82	0.81	0.83	
7	0.86	0.82	0.76	
8	0.79	0.80	0.84	
9	0.84	0.79	0.74	
10	0.82	0.75	0.79	
Trung bình	0.834	0.762	0.796	

Bảng kết quả 4.1 thể hiện độ chính xác nhận diện khuôn mặt qua 10 lần chấm khi cùng một người có các yếu tố thay đổi khác nhau như thay đổi biểu cảm, đeo kính, ngược sáng và che mặt. Trong đó, thay đổi biểu cảm là yếu tố ít ảnh hưởng nhất đến hiệu suất của mô hình, với độ chính xác trung bình đạt 0.834. Điều này cho thấy mô hình nhận diện khuôn mặt vẫn duy trì được sự ổn định và chính xác ngay cả khi biểu cảm của người dùng thay đổi.

Ngược lại, khi người dùng đeo kính, độ chính xác trung bình giảm xuống còn 0.762, cho thấy kính có thể gây cản trở trong việc nhận diện do che khuất hoặc làm thay đổi đặc điểm nhận dạng trên khuôn mặt. Tương tự, điều kiện ngược sáng cũng làm giảm hiệu suất nhận diện với độ chính xác trung bình là 0.796, mặc dù mức giảm này không nghiêm trọng bằng việc đeo kính.

Đáng chú ý nhất là trường hợp có che mặt, mô hình không thể nhận diện chính xác và thường nhầm lẫn sang người khác với độ chính xác rất thấp, chỉ từ 0.2 đến 0.4. Điều này phản ánh một hạn chế lớn trong khả năng nhận diện khi khuôn mặt bị che phủ. Tuy nhiên, với một hệ thống chấm công, điều này không ảnh hưởng quá lớn.

Tổng thể, các kết quả cho thấy mô hình nhận diện khuôn mặt làm việc hiệu quả trong các điều kiện biểu cảm đa dạng và tương đối ổn định khi có ánh sáng không lý tưởng hoặc đeo kính. Tuy nhiên, việc che mặt gây ra ảnh hưởng nghiêm trọng đến hiệu suất,

cần có biện pháp bổ sung hoặc cải tiến để xử lý tốt hơn các trường hợp này nhằm đảm bảo độ chính xác và tính ứng dụng cao trong thực tế.

4.10. Kết luận chương

Như vậy, trong chương này đã tổng kết kết quả hoạt động của hệ thống chấm công khuôn mặt và đánh giá hiệu suất nhận diện khuôn mặt trong nhiều điều kiện thực tế. Những kết quả và nhận xét này là cơ sở quan trọng để hoàn thiện và phát triển hệ thống trong tương lai, hướng tới một giải pháp chấm công tự động tối ưu, chính xác và ứng dụng rộng rãi.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong quá trình thực hiện đồ án, nhóm đã xây dựng thành công một hệ thống chăm công tự động sử dụng công nghệ nhận diện khuôn mặt. Hệ thống kết hợp các thuật toán hiện đại như Mediapipe để phát hiện khuôn mặt, FaceNet để trích xuất đặc trưng, và SVM để phân loại danh tính. Giao diện người dùng được phát triển bằng Tkinter, dữ liệu được lưu trữ và đồng bộ qua MySQL và file CSV, hỗ trợ truy cập từ xa thông qua Cloudflare Tunnel.

Kết quả triển khai cho thấy hệ thống đạt độ chính xác tương đối cao, hoạt động ổn định và thân thiện với người dùng. Các chức năng như thêm, sửa, xóa nhân viên, đổi mật khẩu và xác thực OTP đều được tích hợp đầy đủ. Nhìn chung, đồ án đã hoàn thành các mục tiêu nghiên cứu đặt ra ban đầu.

Tuy nhiên, hệ thống vẫn còn một số hạn chế. Cụ thể, chưa hỗ trợ nhận diện nhiều khuôn mặt cùng lúc; chưa xử lý tốt trong điều kiện ánh sáng yếu, người dùng đeo khẩu trang hoặc thay đổi ngoại hình. Ngoài ra, việc huấn luyện lại mô hình sau mỗi lần thêm, sửa hoặc xóa thông tin nhân viên vẫn mất nhiều thời gian do phải tái huấn luyện toàn bộ mô hình từ đầu, ảnh hưởng đến hiệu suất và tính linh hoạt trong quá trình vận hành hệ thống.

Trong thời gian tới, nhóm dự định tiếp tục cải tiến hệ thống theo các hướng sau: triển khai trên các nền tảng phần cứng có hiệu năng xử lý cao như Mini PC, Orange Pi, NVIDIA Jetson, ... hoặc tích hợp thiết bị tăng tốc Coral USB Accelerator nhằm nâng cao tốc độ xử lý ảnh và tối ưu hóa hiệu suất nhận diện cũng như huấn luyện mô hình; mở rộng khả năng nhận diện trong môi trường phức tạp; đồng thời triển khai thực tế tại doanh nghiệp để tiếp nhận phản hồi và điều chỉnh hệ thống cho phù hợp với nhu cầu sử dụng thực tiễn.

TÀI LIỆU THAM KHẢO

- [1] Research and Markets, “Global Biometrics Technologies Market Report 2019–2027,” *iConnect007*, Oct. 2019. [Online]. Available: <https://iconnect007.com/index.php/article/119622/global-biometrics-technologies-market-report-2019-2027>. [Accessed: June 4, 2025].
- [2] Astute Analytica, “Vietnam Time and Attendance Software Market - Industry Dynamics, Market Size, And Opportunity Forecast To 2033,” *Astute Analytica*, Nov. 2024. [Online]. Available: <https://www.astuteanalytica.com/industry-report/vietnam-time-and-attendance-software-market>. [Accessed: June 4, 2025].
- [3] Google, “MediaPipe Face Detection,” *MediaPipe Documentation*, [Online]. Available: https://mediapipe.readthedocs.io/en/latest/solutions/face_detection.html. [Accessed: June 4, 2025].
- [4] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, “BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs,” *arXiv preprint arXiv:1907.05047*, Jul. 2019. [Online]. Available: <https://arxiv.org/abs/1907.05047>. [Accessed: June 4, 2025].
- [5] L. Dulčić, “Face recognition with FaceNet and MTCNN,” *Ars Futura Blog*, Nov. 2019. [Online]. Available: <https://arsfutura.com/blog/face-recognition-with-facenet-and-mtcnn>. [Accessed: June 4, 2025].
- [6] Seunghan96. (n.d.). “Inception v1, v2, v3: Going Deeper with Convolutions.” Available: <https://seunghan96.github.io/dl/cv/cv-Inception-v1,v2,v3>. [Accessed: June 4, 2025].
- [7] Sandberg, D. (n.d.). “FaceNet: A Unified Embedding for Face Recognition and Clustering.” GitHub. Available: <https://github.com/davidsandberg/facenet?tab=readme-ov-file>. [Accessed: June 4, 2025].
- [8] Schroff, F., Kalenichenko, D., & Philbin, J. (2015). “FaceNet: A Unified Embedding for Face Recognition and Clustering.” *arXiv preprint arXiv:1503.03832*. Available: <https://arxiv.org/pdf/1503.03832>. [Accessed: June 4, 2025].
- [9] Raspberry Pi. (n.d.). “Raspberry Pi 4 Model B.” Available: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b>. [Accessed: June 4, 2025].

- [10] J. Alex, “Celebrity Image Classifier – Image Dataset,” *GitHub Repository*, [Online]. Available: <https://github.com/John-Alex07/Celebrity-Image-Classifier>. [Accessed: June 5, 2025].
- [11] V. Gupta, “Celebrity Face Image Dataset,” *Kaggle*, 2022. [Online]. Available: <https://www.kaggle.com/datasets/vishesh1412/celebrity-face-image-dataset>. [Accessed: June 5, 2025].
- [12] R. Rizwan, “5 Celebrities Face Classification Dataset,” *Kaggle*, 2021. [Online]. Available: <https://www.kaggle.com/datasets/raahimrizwan/5-celebrities-face-classification-dataset>. [Accessed: June 5, 2025].

PHỤ LỤC

Datasheet Raspberry Pi 4 Model B:



Raspberry Pi 4 Model B Datasheet
Copyright Raspberry Pi (Trading) Ltd. 2024

2 Features

2.1 Hardware

- Quad core 64-bit ARM-Cortex A72 running at 1.5GHz
- 1, 2 and 4 Gigabyte LPDDR4 RAM options
- H.265 (HEVC) hardware decode (up to 4Kp60)
- H.264 hardware decode (up to 1080p60)
- VideoCore VI 3D Graphics
- Supports dual HDMI display output up to 4Kp60

2.2 Interfaces

- 802.11 b/g/n/ac Wireless LAN
- Bluetooth 5.0 with BLE
- 1x SD Card
- 2x micro-HDMI ports supporting dual displays up to 4Kp60 resolution
- 2x USB2 ports
- 2x USB3 ports
- 1x Gigabit Ethernet port (supports PoE with add-on PoE HAT)
- 1x Raspberry Pi camera port (2-lane MIPI CSI)
- 1x Raspberry Pi display port (2-lane MIPI DSI)
- 28x user GPIO supporting various interface options:
 - Up to 6x UART
 - Up to 6x I2C
 - Up to 5x SPI
 - 1x SDIO interface
 - 1x DPI (Parallel RGB Display)
 - 1x PCM
 - Up to 2x PWM channels
 - Up to 3x GPCLK outputs

PHỤ LỤC

Đường dẫn tới code hệ thống:

https://drive.google.com/drive/folders/1RzZcfmU15xijlJsEB_WxGg6qa9Ap6nOg

