

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐIỆN ĐIỆN TỬ

BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG

ĐỒ ÁN TỐT NGHIỆP

**HỆ THỐNG ĐIỂM DANH NHÂN VIÊN
KHI RA VÀO CÔNG SỬ DỤNG NHẬN
DIỆN KHUÔN MẶT**

NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH

Sinh viên: **CAO TÚ TÀI**

MSSV: 15119124

NGUYỄN ĐĂNG KHOA

MSSV: 15119097

TP. HỒ CHÍ MINH – 06/2019

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐIỆN ĐIỆN TỬ

BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG

ĐỒ ÁN TỐT NGHIỆP

**HỆ THỐNG ĐIỂM DANH NHÂN VIÊN
KHI RA VÀO CÔNG SỬ DỤNG NHẬN
DIỆN KHUÔN MẶT**

NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH

Sinh viên: **CAO TÚ TÀI**

MSSV: 15119124

NGUYỄN ĐĂNG KHOA

MSSV: 15119097

Hướng dẫn: **ThS. ĐẬU TRỌNG HIỂN**

TP. HỒ CHÍ MINH – 06/2019

BẢN NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP

(Dành cho giảng viên hướng dẫn)

Đề tài: Ứng dụng điểm danh nhân viên sử dụng nhận diện khuôn mặt

Sinh viên thực hiện: 1. Cao Tú Tài
2. Nguyễn Đăng Khoa

MSSV: 15119124

MSSV: 15119097

Giảng viên hướng dẫn: Ths. Đậu Trọng Hiền

Nhận xét bao gồm các nội dung sau đây:

1. Tính hợp lý trong cách đặt vấn đề và giải quyết vấn đề; ý nghĩa khoa học và thực tiễn:

Đặt vấn đề rõ ràng, mục tiêu cụ thể; đề tài có tính mới, cấp thiết; đề tài có khả năng ứng dụng, tính sáng tạo.

hợp lý

2. Phương pháp thực hiện/ phân tích/ thiết kế:

Phương pháp hợp lý và tin cậy dựa trên cơ sở lý thuyết; có phân tích và đánh giá phù hợp; có tính mới và tính sáng tạo.

phù hợp

3. Kết quả thực hiện/ phân tích và đánh giá kết quả/ kiểm định thiết kế:

Phù hợp với mục tiêu đề tài; phân tích và đánh giá / kiểm thử thiết kế hợp lý; có tính sáng tạo/ kiểm định chặt chẽ và đảm bảo độ tin cậy.

phân tích

4. Kết luận và đề xuất:

Kết luận phù hợp với cách đặt vấn đề, đề xuất mang tính cải tiến và thực tiễn; kết luận có đóng góp mới mẻ, đề xuất sáng tạo và thuyết phục.

thực tiễn

5. Hình thức trình bày và bố cục báo cáo:

Văn phong nhất quán, bố cục hợp lý, cấu trúc rõ ràng, đúng định dạng mẫu; có tính hấp dẫn, thể hiện năng lực tốt, văn bản trau chuốt.

hợp lý

6. Kỹ năng chuyên nghiệp và tính sáng tạo:

Thể hiện các kỹ năng giao tiếp, kỹ năng làm việc nhóm, và các kỹ năng chuyên nghiệp khác trong việc thực hiện đề tài.

7. Tài liệu trích dẫn

Tính trung thực trong việc trích dẫn tài liệu tham khảo; tính phù hợp của các tài liệu trích dẫn; trích dẫn theo đúng chỉ dẫn APA.

8. Đánh giá về sự trùng lặp của đề tài

Cần khẳng định đề tài có trùng lặp hay không? Nếu có, đề nghị ghi rõ mức độ, tên đề tài, nơi công bố, năm công bố của đề tài đã công bố.

không trùng lặp

9. Những nhược điểm và thiếu sót, những điểm cần được bổ sung và chỉnh sửa*

10. Nhận xét tinh thần, thái độ học tập, nghiên cứu của sinh viên

Tốt

Đề nghị của giảng viên hướng dẫn

Ghi rõ: "Báo cáo đạt/ không đạt yêu cầu của một khóa luận tốt nghiệp kỹ sư, và được phép/ không được phép bảo vệ khóa luận tốt nghiệp"

Đạt yêu cầu

Tp. HCM, ngày 03 tháng 06 năm 2019

Người nhận xét

(Ký và ghi rõ họ tên)

Đậu Trọng Hiền

LỜI CẢM ƠN

Tác giả xin tỏ lòng kính trọng và cảm ơn tới Thầy Ths. Đậu Trọng Hiền đã tận tình hướng dẫn, giúp đỡ tác giả trong suốt quá trình thực hiện đồ án tốt nghiệp này.

Đồng thời, tác giả xin gửi lời cảm ơn tới thầy cô trong Khoa Điện – Điện tử đã tận tình giảng dạy, truyền đạt cho tác giả những kiến thức quý báu trong suốt những năm học qua, giúp tác giả có nền tảng kiến thức để vận dụng vào thực hiện đồ án tốt nghiệp cũng như áp dụng vào thực tiễn.

Tác giả cũng rất trân trọng sự giúp đỡ của các bạn bè trong và ngoài lớp đã tận tình giúp đỡ trong thời gian thực hiện luận văn.

Trong quá trình thực hiện đề tài, tuy rất cố gắng nhưng vì thời gian tìm hiểu có hạn, kiến thức và kinh nghiệm còn hạn hẹp nên không tránh khỏi nhiều thiếu sót. Tác giả rất mong nhận được những ý kiến đóng góp của thầy và các bạn để đề tài được hoàn thiện hơn.

Một lần nữa, tác giả xin chân thành cảm ơn!

TÓM TẮT

Ngày nay với sự phát triển mạnh mẽ của các công nghệ hiện đại, vấn đề bảo mật và lưu trữ thông tin ngày càng đòi hỏi tính chính xác và độ bảo mật ngày càng cao, chính vì vậy công nghệ sinh trắc học đã ra đời. Công nghệ sinh trắc học là công nghệ nhận dạng từ những đặc tính sinh trắc học của con người nó mang tính duy nhất và không thay đổi theo thời gian vì vậy đáp ứng được nhu cầu của bảo mật của xã hội hiện nay. Với mong muốn tìm hiểu về công nghệ này tác giả thực hiện nghiên cứu và xây dựng chương trình thực hiện nhận dạng khuôn mặt để phát triển ứng dụng điểm danh và chấm công nhân viên cho các cơ quan, cá nhân, tổ chức.

Trong thời buổi ngày nay, khi mà kinh tế ngày càng phát triển thì công nghệ nhận dạng khuôn mặt trở nên phổ biến hơn bao giờ hết. Con người có quyền kì vọng nhiều lợi ích mà nó mang lại trong tương lai.

Một trong những điểm tuyệt vời nhất mà công nghệ nhận dạng khuôn mặt đem lại cho con người là nó có thể hoạt động ở khoảng cách xa. Điều này đồng nghĩa với việc không phải tiếp xúc trực tiếp với bất kì hệ thống bảo mật nào kể cả máy quét vân tay xưa nay ta vẫn sử dụng. Hệ thống nhận diện khuôn mặt hoạt động bằng cách tạo ra một bản thiết kế các dạng khuôn mặt khác nhau. Điều này được thực hiện bằng cách chụp lại bằng camera và sau đó đo khoảng cách trên gương mặt, được gọi là các điểm nút, bao gồm giữa mắt và chiều rộng của mũi. Hình ảnh này sau đó có thể được sử dụng như tài liệu tham khảo. Điều này cho phép hệ thống được sử dụng cho mục đích xác thực.

MỤC LỤC

DANH MỤC HÌNH	VIII
DANH MỤC BẢNG	X
CÁC TỪ VIẾT TẮT.....	XI
CHƯƠNG 1 GIỚI THIỆU	1
1.1 GIỚI THIỆU	1
1.2 MỤC TIÊU ĐỀ TÀI	2
1.3 GIỚI HẠN ĐỀ TÀI	2
1.4 PHƯƠNG PHÁP NGHIÊN CỨU	2
1.5 ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU	3
1.6 BỐ CỤC QUYỀN BÁO CÁO.....	3
CHƯƠNG 2 CƠ SỞ LÝ THUYẾT.....	5
2.1 TÌNH HÌNH NGHIÊN CỨU	5
2.2 CÁC HƯỚNG TIẾP CẬN MẶT NGƯỜI.....	5
2.3 PHÁT HIỆN MẶT NGƯỜI SỬ DỤNG THUẬT TOÁN VIOLA - JONES.....	6
2.3.1 ĐẶC TRƯNG HAAR - LIKE	6
2.3.2 ADABOOST.....	9
2.3.3 SƠ ĐỒ KHỞI QUÁ TRÌNH TÌM KIẾM MẶT NGƯỜI.....	10

2.4 KỸ THUẬT MÃ HÓA TẠO RA NHỮNG ĐẶC TRƯNG CHO NHẬN DIỆN	
KHUÔN MẶT	11
2.4.1 THƯ VIỆN DLIB	11
2.4.2 THUẬT TOÁN HISTOGRAM OF ORIENTED GRADIENTS(HOG)	12
2.4.3 THUẬT TOÁN SUPPORT VECTOR MACHINE(SVM).....	13
2.4.4 DATASET LABELED FACE IN WILD (LFW)	14
2.4.5 TRÍCH ĐẶC TRƯNG VÀ HUẤN LUYỆN	15
2.5 NHẬN DẠNG KHUÔN MẶT NGƯỜI.....	16
CHƯƠNG 3 THIẾT KẾ HỆ THỐNG	17
3.1 YÊU CẦU HỆ THỐNG	17
3.2 MÔ HÌNH HỆ THỐNG	18
3.3 THIẾT KẾ PHẦN CỨNG	18
3.3.1 SƠ ĐỒ KHỐI PHẦN CỨNG	18
3.3.2 LỰA CHỌN PHẦN CỨNG	19
3.3.3 KẾT NỐI PHẦN CỨNG	23
3.4 THIẾT KẾ PHẦN MỀM	23
3.4.1 CHỨC NĂNG HOẠT ĐỘNG CỦA PHẦN MỀM	23
3.4.2 QUY TRÌNH XỬ LÝ CỦA HỆ THỐNG	23
3.4.3 LƯU ĐỒ HOẠT ĐỘNG.....	24
3.4.4 CÁC BƯỚC THỰC HIỆN	27
CHƯƠNG 4 KẾT QUẢ.....	33
4.1 MÔ HÌNH THỰC TẾ CỦA HỆ THỐNG	33
4.2 HOẠT ĐỘNG CỦA HỆ THỐNG	33
CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	41
5.1 KẾT LUẬN	41
5.2 HƯỚNG PHÁT TRIỂN	42

TÀI LIỆU THAM KHẢO.....	44
--------------------------------	-----------

DANH MỤC HÌNH

Hình 2.1: Đặc trưng Haar-like cơ bản.....	6
Hình 2.2: Đặc trưng Haar-like mở rộng.....	7
Hình 2.3: Công thức Integral Image của ảnh	8
Hình 2.4: Công thức tính mức xám của vùng D.....	8
Hình 2.5: Lưu đồ hợp nhất weak classifiers để xác định khuôn mặt	9
Hình 2.6 Quá trình hình thành strong classifiers.....	10
Hình 2.7: Sơ đồ khối quá trình tìm kiếm khuôn mặt người	10
Hình 2.8: Kết quả sau khi tính toán độ dốc.....	12
Hình 2.9: Kết quả sau khi lấy phiếu bầu	13
Hình 2.10: SVM phân chia các Support Vectors thành 2 phần riêng biệt.....	14
Hình 2.11: Định lượng khuôn mặt sử dụng vector đặc trưng 128-D.....	15
Hình 2.12: Mô hình Convnet tượng trưng chuyển ảnh thành vector 4096 chiều	15
Hình 2.13: Quá trình đào tạo trong bài toán triplet	16
Hình 2.14: Quá trình nhận diện mặt người của toàn bộ hệ thống.....	17
Hình 3.1: Sơ đồ mô hình hệ thống.....	18
Hình 3.2: Sơ đồ khối hệ thống.....	18
Hình 3.3: Raspberr Pi Camera Module V1.....	19
Hình 3.4: Raspberry Pi 3 Model B.....	20
Hình 3.5: Loa VSP-CK4	22
Hình 3.6: Adapter 5V/2A.....	22
Hình 3.7: Cách kết nối phần cứng.....	23
Hình 3.8: Lưu đồ phát hiện khuôn mặt.....	25
Hình 3.9: Lưu đồ huấn luyện.....	25
Hình 3.10: Lưu đồ nhận diện khuôn mặt.....	26

<i>Hình 3.11: Cấu trúc phần mềm</i>	<i>27</i>
<i>Hình 3.12: Kết quả sau khi build OpenCV</i>	<i>30</i>
<i>Hình 3.13: Kết quả sau khi biên dịch OpenCV</i>	<i>31</i>
<i>Hình 4.1: Mô hình hoàn chỉnh của hệ thống</i>	<i>33</i>
<i>Hình 4.2: Giao diện của hệ thống</i>	<i>34</i>
<i>Hình 4.3: Cảnh báo người dùng.....</i>	<i>34</i>
<i>Hình 4.4: Giao diện khi lưu ảnh nhân viên</i>	<i>35</i>
<i>Hình 4.5: Trích đặc trưng của tất cả các ảnh hiện có</i>	<i>35</i>
<i>Hình 4.6: Sau khi đã trích xong tất cả.....</i>	<i>35</i>
<i>Hình 4.7: Kết quả sau khi nhận diện sẽ hiển thị trên màn hình</i>	<i>36</i>
<i>Hình 4.8: Lưu thông tin giờ vào, giờ ra của nhân viên.....</i>	<i>37</i>
<i>Hình 4.9: Kết quả nhận diện của một nhân viên khác</i>	<i>38</i>
<i>Hình 4.10: Kết quả sau khi nhận diện ngược sáng</i>	<i>38</i>
<i>Hình 4.11: Kết quả nhận diện trong môi trường ánh sáng chói</i>	<i>39</i>
<i>Hình 4.12: Kết quả khi nhận diện với góc nghiêng.....</i>	<i>39</i>

DANH MỤC BẢNG

<i>Bảng 1: Bảng thống kê tỉ lệ chính xác qua các lần thử.....</i>	40
---	----

CÁC TỪ VIẾT TẮT

HOG: Histogram of Oriented Gradients

SVM: Support Vector Machine

LFW: Labeled Face in Wild

Chương 1 **GIỚI THIỆU**

1.1 GIỚI THIỆU

Ngày nay, cùng với những tiến bộ vượt bậc của khoa học kỹ thuật nói chung, bộ môn khoa học xử lý ảnh đã và đang thu được những thành tựu lớn lao và chứng tỏ vai trò không thể thiếu với những ứng dụng sâu rộng trong khoa học kỹ thuật cũng như đời sống xã hội. Con người rất giỏi việc nhận diện khuôn mặt và các hình mẫu phức tạp. Ngay cả khi một thời gian trôi qua vẫn không ảnh hưởng đến khả năng này và con người muốn máy tính trở nên mạnh mẽ như họ trong việc nhận diện khuôn mặt. Hơn một thập kỷ qua, có rất nhiều công trình nghiên cứu về bài toán xác định khuôn mặt người từ ảnh đen trắng, ảnh xám, đến ảnh màu. Các nghiên cứu đi từ bài toán đơn giản là ảnh chỉ chứa một khuôn mặt người nhìn thẳng vào thiết bị thu hình và đầu ở tư thế thẳng đứng, cho đến ảnh màu với nhiều khuôn mặt người trong cùng ảnh, khuôn mặt có quay một góc nhỏ, hay bị che khuất một phần, và với ảnh nền của ảnh phức tạp (ảnh chụp ngoài thiên nhiên) nhằm đáp ứng nhu cầu thật sự cần thiết của con người. Bài toán xác định khuôn mặt người là một kỹ thuật máy tính để xác định các vị trí và các kích thước của khuôn mặt trong ảnh bất kỳ (ảnh kỹ thuật số). Kỹ thuật này nhận biết các đặt trưng khuôn mặt và bỏ qua những thứ khác như: toà nhà, cây cối, cơ thể... Nó có thể giải quyết các bài toán như giám sát tự động phục vụ trong cơ quan, ngân hàng, kho bạc... hoặc trong việc giám sát giao thông tự động phục vụ tại bãi đỗ xe, trạm thu phí tự động hoặc việc phát hiện và nhận dạng mặt người phục vụ trong công tác quân sự, an ninh v.v...

Từ những ứng dụng trên của công nghệ nhận diện khuôn mặt, tác giả đã chọn đề tài nghiên cứu: *“Hệ thống điểm danh nhân viên khi ra vào cổng sử dụng nhận diện khuôn mặt”* để giải quyết vấn đề chấm công cho nhân viên, vấn đề mà hầu hết tất cả các tổ chức hoạt động đều cần phải dùng đến.

1.2 MỤC TIÊU ĐỀ TÀI

Đề tài: “*Hệ thống điểm danh nhân viên khi ra vào cổng sử dụng nhận diện khuôn mặt*” được thực hiện với mục tiêu ban đầu là:

- Thực hiện thêm ảnh nhận diện cho các nhân viên trong công ty kèm theo mã số cho mỗi nhân viên đó.
- Tiến hành nhận diện khuôn mặt cho nhân viên khi thực hiện quét khuôn mặt trên camera và sau đó xuất ra giờ vào, giờ ra của nhân viên đó.
- Chương trình xử lý nhận diện khuôn mặt với độ chính xác cao.
- Xây dựng giao diện tiện lợi, dễ sử dụng cho người dùng.

1.3 GIỚI HẠN ĐỀ TÀI

Do tính chất của các đặc trưng và thuật toán nhận diện khuôn mặt quá nhiều và khó nên đề tài này chỉ tập trung vào phần ứng dụng phát triển thực tế.

Do các khó khăn của bài toán nhận diện khuôn mặt như: tư thế góc chụp, các thành phần xuất hiện trên khuôn mặt, độ sáng của ảnh... vì thế trong luận văn này tác giả đưa ra các ràng buộc và giả định sau để làm giảm độ phức tạp của bài toán nhận diện khuôn mặt:

- Người được nhận diện phải nhìn thẳng vào camera, hoặc góc nhìn so với camera không đáng kể (nhỏ hơn 15°).
- Camera có độ phân giải quá kém sẽ ảnh hưởng đến kết quả xử lý không được tốt nhất có thể.
- Khuôn mặt không bị che khuất.
- Khoảng cách người nhận diện đến camera trong khoảng 30cm đến 60cm.

1.4 PHƯƠNG PHÁP NGHIÊN CỨU

Tìm hiểu phương pháp phát hiện khuôn mặt Thuật Toán Viola-Jones và phương pháp trích đặc trưng dựa vào thư viện dlib).

Tìm hiểu các lý thuyết có liên quan như ngôn ngữ Python, Raspberry Pi và thư viện OpenCV.

Phương pháp thực nghiệm: xây dựng chương trình nhận diện khuôn mặt trên máy tính nhúng Raspberry Pi để điểm danh cho nhân viên công ty.

1.5 ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

Đối tượng nghiên cứu:

- Các phương pháp, giải thuật phát hiện và nhận dạng khuôn mặt người trên ảnh.
- Máy tính nhúng Raspberry Pi, camera Pi, loa.
- Ngôn ngữ lập trình Python, thư viện OpenCV và dlib.

Phạm vi nghiên cứu:

- Việc xử lý và nhận dạng ảnh khuôn mặt thỏa mãn các điều kiện sau:
 - + Ánh sáng đều, không có chiều sáng, không có ánh sáng mạnh.
 - + Góc ảnh: trực diện hoặc gần như trực diện.
 - + Khuôn mặt không bị che khuất.
 - + Ảnh chất lượng cao.
 - + Khoảng cách người nhận diện đến camera trong khoảng 40cm – 60cm.

1.6 BỐ CỤC QUYỀN BÁO CÁO

Nội dung chính của đề tài được trình bày với năm chương như sau:

Chương 1: Giới thiệu

Trong chương này nêu ra được tình hình nghiên cứu hiện nay, lý do và mục tiêu chọn đề tài, đối tượng và phạm vi nghiên cứu của đề tài, phương pháp nghiên cứu và giới hạn của đề tài.

Chương 2: Cơ sở lý thuyết

Các lý thuyết chính liên quan đến phương pháp phát hiện và nhận diện khuôn mặt người.

Chương 3: Thiết kế hệ thống

Trong chương này mục đích là thiết kế phần cứng và xây dựng phần mềm cho hệ thống với những yêu cầu đặt ra:

- Thiết kế phần cứng: Từ sơ đồ khối tổng quát và sơ đồ khối chi tiết để tiến hành lựa chọn các linh kiện cho các khối.

- Thiết kế phần mềm: Xây dựng chương trình nhận diện và phát âm thanh trên Raspberry Pi sử dụng ngôn ngữ lập trình Python và thư viện OpenCV, Dlib.

Chương 4: Kết quả

Trình bày về kết quả thi công phần cứng, phần mềm và đánh giá ưu nhược điểm của hệ thống.

Chương 5: Kết luận và hướng phát triển

Đưa ra các kết luận về những vấn đề mà trong quá trình nghiên cứu đã đạt được.

Chương 2 CƠ SỞ LÝ THUYẾT

2.1 TÌNH HÌNH NGHIÊN CỨU

Hiện nay, xu hướng phát triển công nghệ được ứng dụng trong nhiều lĩnh vực, để đáp ứng được nhu cầu về phát triển thiết bị phần mềm như hiện nay, trí tuệ thông minh nhân tạo AI (viết tắt của Artificial Intelligence) đã ra đời. AI là công nghệ mô phỏng các quá trình suy nghĩ và học tập của con người cho máy móc, đặc biệt là các hệ thống máy tính. Các quá trình này bao gồm việc học tập (thu thập thông tin và các quy tắc sử dụng thông tin), lập luận (sử dụng các quy tắc để đạt được kết luận gần đúng hoặc xác định) và tự sửa lỗi. Các ứng dụng đặc biệt của AI bao gồm các hệ thống chuyên gia, nhận dạng tiếng nói và thị giác máy tính (nhận diện khuôn mặt, vật thể hoặc chữ viết). Các công ty công nghệ lớn như Facebook, SenseTime, Google... đã kế thừa sự phát triển của AI để xây dựng hệ thống tự nhận diện khuôn mặt với độ chính xác ngày càng cao vào thiết bị và phần mềm.

Các hệ thống quản lý nhân viên ngày càng được trang bị những hệ thống giám sát chất lượng cao với sự giúp sức của AI nhằm nâng cao sự chính xác cũng như tiện lợi cho người sử dụng. Hệ thống quản lý sử dụng nhận diện khuôn mặt là một trong những hệ thống đang được chú ý và phát triển từng ngày, vì nó khắc phục được những hạn chế nhất định mà các hệ thống truyền thống trước kia không thể đáp ứng.

2.2 CÁC HƯỚNG TIẾP CẬN MẶT NGƯỜI

Dựa vào tính chất của các phương pháp xác định mặt người trên ảnh, các phương pháp này được chia thành bốn loại chính, tương ứng với bốn hướng tiếp cận khác nhau:

- Hướng tiếp cận dựa trên tri thức: đây là hướng tiếp cận theo kiểu top-down. Dựa vào các thuật toán, mã hóa các đặc trưng và quan hệ giữa các đặc trưng của khuôn mặt.

- Hướng tiếp cận dựa trên đặc trưng bất biến: đây là hướng tiếp cận theo kiểu bottom – up. Dựa vào các đặc trưng bất biến: khuôn mặt, kết cấu, màu sắc và nhiều đặc trưng kết hợp.

- Hướng tiếp cận dựa trên so sánh mẫu: dùng các mẫu khuôn mặt và đặc trưng chuẩn (các mẫu đã được lưu trữ) để so sánh các khuôn mặt hay các đặc trưng của khuôn mặt (các mẫu này được chọn tách biệt theo tiêu chuẩn đã được các tác giả đề ra để so sánh).

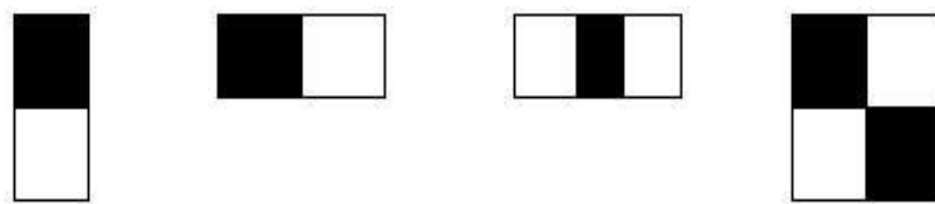
- Hướng tiếp cận dựa trên ngoại hình: trái ngược với hướng tiếp cận dựa trên khuôn mẫu, các mô hình (hay các mẫu) sẽ được học từ một tập ảnh huấn luyện mà thể hiện tính chất tiêu biểu của sự xuất hiện của mặt người trong ảnh. Sau đó hệ thống (mô hình) sẽ xác định mặt người. Phương pháp này còn được biết đến với tên gọi tiếp cận theo các phương pháp học máy.

- Ngoài ra cũng có rất nhiều nghiên cứu mà phương pháp xác định mặt người không chỉ dựa vào một hướng mà có liên quan đến nhiều hướng.

2.3 PHÁT HIỆN MẶT NGƯỜI SỬ DỤNG THUẬT TOÁN VIOLA – JONES

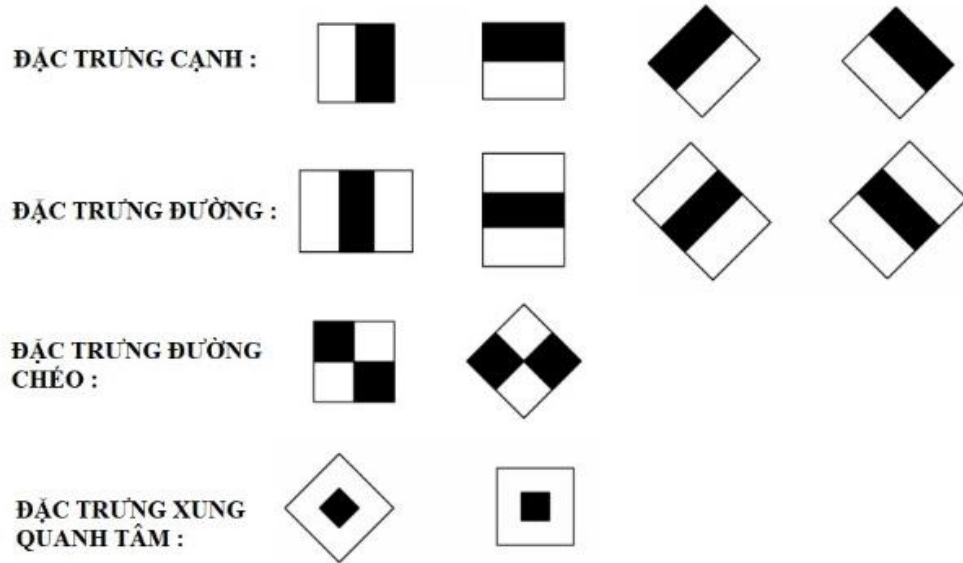
2.3.1 ĐẶC TRƯNG HAAR - LIKE

Các đặc trưng Haar-like do Viola và Jones công bố. Chúng gồm 4 đặc trưng cơ bản để xác định khuôn mặt người. Mỗi đặc trưng Haar-like là sự kết hợp của hai hay ba hình chữ nhật "trắng" hay "đen" như trong hình sau:



Hình 2.1: Đặc trưng Haar-like cơ bản

Để sử dụng các đặt trưng này vào việc xác định khuôn mặt người, 4 đặt trưng Haar-like cơ bản được mở rộng ra, và được biểu diễn trong hình 2.2:



Hình 2.2: Đặc trưng Haar-like mở rộng

Dùng các đặc trưng trên, ta có thể tính được giá trị của đặc trưng Haar-like là sự chênh lệch giữa tổng của các pixel của các vùng đen và các vùng trắng như trong công thức sau:

$$f(x) = \text{Tổng vùng đen} - \text{Tổng vùng trắng (mức xám các pixel)} \quad (1)$$

Sử dụng giá trị này so sánh với các giá trị của các giá trị pixel thô, các đặc trưng Haar-like có thể tăng/giảm sự thay đổi in-class/out-of-class (bên trong hay bên ngoài lớp khuôn mặt người), do đó sẽ làm cho bộ phân loại dễ hơn.

Như vậy ta có thể thấy rằng, để tính các giá trị của đặc trưng Haar-like, ta phải tính tổng của các vùng pixel trên ảnh. Nhưng để tính toán các giá trị của các đặc trưng Haar-like cho tất cả các vị trí trên ảnh đòi hỏi chi phí tính toán khá lớn, không đáp ứng được cho các ứng dụng đòi hỏi tính run-time. Do đó Viola và Jones đưa ra một khái niệm gọi là Integral Image, là một mảng 2 chiều với kích thước bằng với kích của ảnh cần tính các đặc trưng Haar-like, với mỗi phần tử của mảng

này được tính bằng cách tính tổng của điểm ảnh phía trên (dòng-1) và bên trái (cột-1) của nó. Bắt đầu từ vị trí trên, bên trái đến vị trí dưới, phải của ảnh, việc tính toán này đơn thuần chỉ dựa trên phép cộng số nguyên đơn giản, do đó tốc độ thực hiện rất nhanh.



$$P(x,y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (2)$$

Hình 2.3: Công thức Integral Image của ảnh

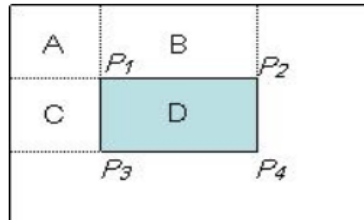
Sau khi đã tính được Integral Image, việc tính tổng các giá trị mức xám của một vùng bất kỳ nào đó trên ảnh thực hiện rất đơn giản theo cách sau:

Giả sử ta cần tính tổng các giá trị mức xám của vùng D như trong hình 2.4, ta có thể tính như sau:

$$D = A + B + C + D - (A+B) - (A+C) + A \quad (3)$$

Với $A + B + C + D$ chính là giá trị tại điểm P4 trên Integral Image, tương tự như vậy $A+B$ là giá trị tại điểm P2, $A+C$ là giá trị tại điểm P3, và A là giá trị tại điểm P1. Vậy ta có thể viết lại biểu thức tính D ở trên như sau:

$$D = \underbrace{(x_4, y_4)}_{A+B+C+D} - \underbrace{(x_2, y_2)}_{A+B} - \underbrace{(x_3, y_3)}_{A+C} + \underbrace{(x_1, y_1)}_A \quad (4)$$



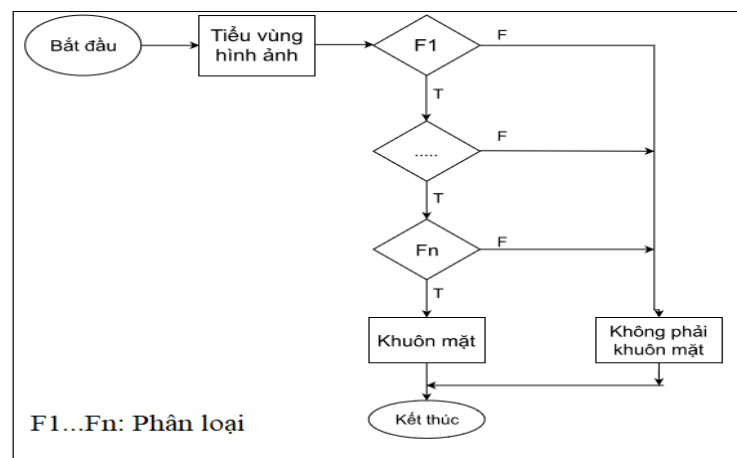
Hình 2.4: Công thức tính mức xám của vùng D

Tiếp theo, để chọn các đặc trưng Haar-like dùng cho việc thiết lập ngưỡng, Viola và Jones sử dụng một phương pháp máy học được gọi là AdaBoost. AdaBoost sẽ kết hợp các bộ phân loại yếu để tạo thành một bộ phân loại mạnh. Với bộ phân loại yếu chỉ cho ra câu trả lời chính xác chỉ hơn viện đoán một cách ngẫu nhiên một chút, còn bộ phân loại mạnh có thể đưa ra câu trả lời chính xác trên 60%.

2.3.2 ADABOOST

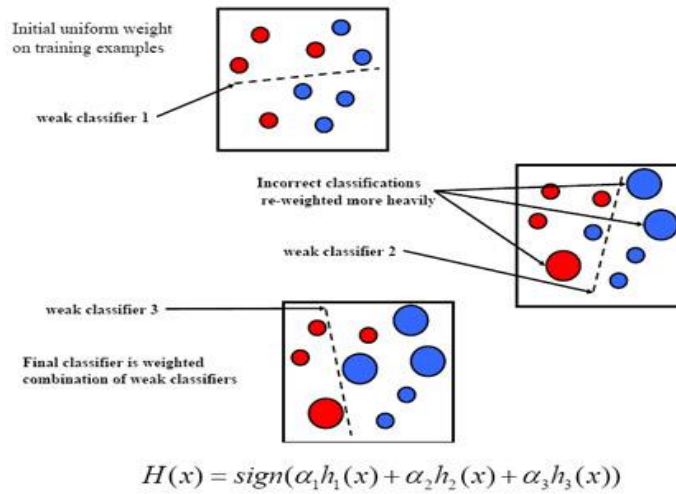
AdaBoost là một thuật toán boosting dùng để xây dựng bộ phân lớp (classifier). Adaboost cũng hoạt động trên nguyên tắc kết hợp tuyến tính các weak classifiers để hình thành một strong classifier.

Là một cải tiến của tiếp cận boosting, AdaBoost sử dụng thêm khái niệm trọng số (weight) để đánh dấu các mẫu khó nhận dạng. Trong quá trình huấn luyện, cứ mỗi weak classifiers được xây dựng, thuật toán sẽ tiến hành cập nhật lại trọng số để chuẩn bị cho việc xây dựng weak classifier kế tiếp: tăng trọng số của các mẫu bị nhận dạng sai và giảm trọng số của các mẫu được nhận dạng đúng bởi weak classifier vừa xây dựng. Bằng cách này weak classifier sau có thể tập trung vào các mẫu mà các weak classifiers trước nó làm chưa tốt. Sau cùng, các weak classifiers sẽ được kết hợp tùy theo mức độ tốt của chúng để tạo nên strong classifier. Viola và Jones dùng AdaBoost kết hợp các bộ phân loại yếu sử dụng các đặc trưng Haar-like theo mô hình phân tầng (cascade) như sau:



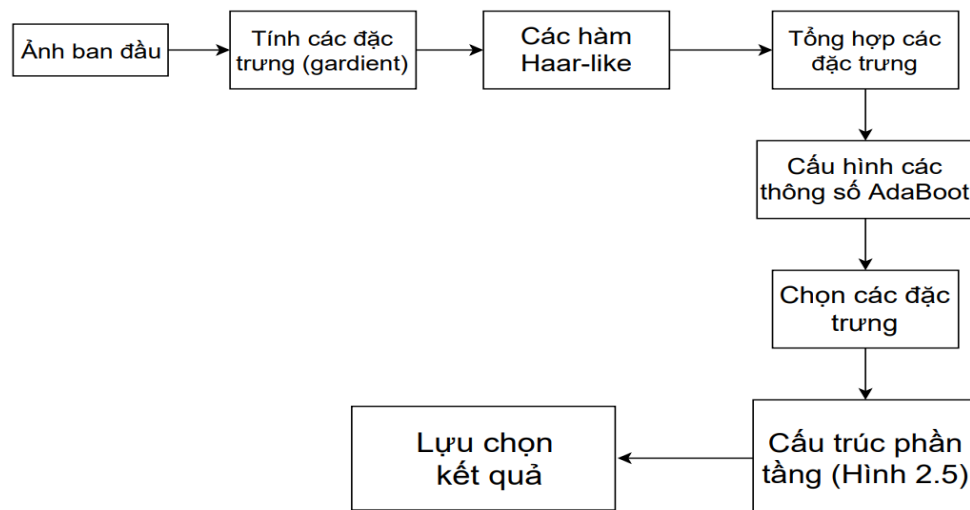
Hình 2.5: Lưu đồ hợp nhất weak classifiers để xác định khuôn mặt

AdaBoost sẽ kết hợp các bộ phân loại yếu thành bộ phân loại mạnh như sau:



Hình 2.6 Quá trình hình thành strong classifiers

2.3.3 SƠ ĐỒ KHỞI QUÁ TRÌNH TÌM KIẾM MẶT NGƯỜI



Hình 2.7: Sơ đồ khởi quá trình tìm kiếm khuôn mặt người

Như trong hình, từ ảnh gốc ban đầu, ta sẽ được tính Integral Image, là mảng 2 chiều với phần tử (x, y) sẽ được tính bằng tổng của các phần tử (x', y') với $x' < x$ và $y' < y$, mục đích là để tính nhanh tổng của các giá trị mức xám của một vùng hình

chữ nhật bất kỳ trên ảnh gốc. Các vùng ảnh con này sẽ được đưa qua các hàm Haar cơ bản để ước lượng đặc trưng, kết quả ước lượng sẽ được đưa qua bộ điều chỉnh AdaBoost để loại bỏ nhanh các đặc trưng không có khả năng là đặc trưng của khuôn mặt người. Chỉ có một tập nhỏ các đặc trưng mà bộ điều chỉnh AdaBoost cho là có khả năng là đặc trưng của khuôn mặt người mới được chuyển sang cho bộ quyết định kết quả (là tập các bộ phân loại yếu có cấu trúc như trong Hình 2.5). Bộ quyết định sẽ tổng hợp kết quả là khuôn mặt người nếu kết quả của các bộ phân loại yếu trả về là khuôn mặt người.

Mỗi bộ phân loại yếu sẽ quyết định kết quả cho một đặc trưng Haar-like, được xác định ngưỡng đủ nhỏ sao cho có thể vượt được tất cả các bộ dữ liệu mẫu trong tập dữ liệu huấn luyện (số lượng ảnh khuôn mặt trong tập huấn luyện có thể rất lớn). Trong quá trình xác định khuôn mặt người, mỗi vùng ảnh con sẽ được kiểm tra với các đặc trưng trong chuỗi các đặc trưng Haar-like, nếu có một đặc trưng Haar-like nào cho ra kết quả là khuôn mặt người thì các đặc trưng khác không cần xét nữa. Thứ tự xét các đặc trưng trong chuỗi các đặc trưng Haar-like sẽ được dựa vào trọng số (weight) của đặc trưng đó do AdaBoost quyết định dựa vào số lần và thứ tự xuất hiện của các đặc trưng Haar-like.

2.4 KỸ THUẬT MÃ HOÁ TẠO RA NHỮNG ĐẶC TRƯNG CHO NHẬN DIỆN KHUÔN MẶT

2.4.1 THƯ VIỆN DLIB

Dlib là thư viện phần mềm mã nguồn mở chạy được trên nhiều nền tảng được viết bằng ngôn ngữ C++ bởi Davis King. Dlib hỗ trợ nhiều công việc khác nhau như threading, networking, numerical operations, machine learning, computer vision, compression. Trong OpenCV dlib được sử dụng để hỗ trợ người dùng trong việc xác định khuôn mặt. Thuật toán mà Dlib sử dụng đó là HOG (Histogram of Oriented Gradients) và SVM (Support Vector Machine), đây chính là lý do tại sao dlib có thời

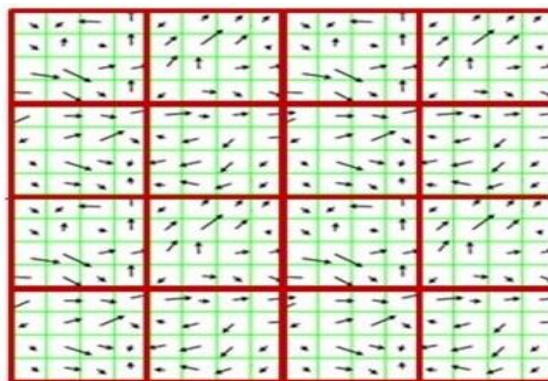
gian chạy rất nhỏ và có thể sử dụng trong các hệ thống không được trang bị những bộ xử lý mạnh hoặc GPU nhưng vẫn ổn định khi chạy trong thời gian thực.

2.4.2 THUẬT TOÁN HISTOGRAM OF ORIENTED GRADIENTS(HOG)

HOG (histogram of oriented gradients) là một thuật toán được sử dụng trong computer vision và xử lý hình ảnh, dùng để phát hiện một đối tượng. Các khái niệm về HOG được nêu ra từ năm 1986 tuy nhiên cho đến năm 2005 HOG mới được sử dụng rộng rãi sau khi Navneet Dalal và Bill Triggs công bố những bổ sung về HOG. HOG được tính toán trên một lưới dày đặc các ô và chuẩn hóa sự tương phản giữa các khối để nâng cao độ chính xác. HOG được sử dụng chủ yếu để mô tả hình dạng và sự xuất hiện của một đối tượng trong ảnh. Bài toán tính toán HOG thường gồm 5 bước:

- + Chuẩn hóa hình ảnh trước khi xử lý: Bước chuẩn hóa này hoàn toàn không bắt buộc, nhưng trong một số trường hợp, bước này có thể cải thiện hiệu suất của bộ mô tả HOG. Có ba phương pháp chuẩn hóa chính mà chúng ta có thể xem xét: chuẩn hoá gamma/năng lượng, chuẩn hoá góc vuông, chuẩn hoá phương sai.

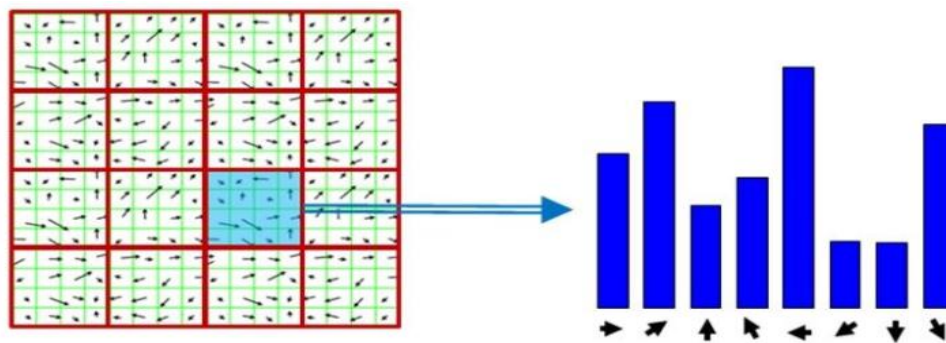
- + Tính toán độ dốc theo cả hướng x và y: để lấy được hình ảnh độ dốc, chúng ta sẽ sử dụng tích chập giữa hình ảnh đầu vào với hai bộ lọc chiều x và chiều y, sau khi có các ảnh độ dốc chúng ta có thể tính toán cường độ độ dốc của hình ảnh.



Hình 2.8: Kết quả sau khi tính toán độ dốc

- + Lấy phiếu bầu cùng trọng số trong các ô: bây giờ chúng ta cần chia hình ảnh của chúng ta thành các cell và block. Một cell là một vùng hình chữ nhật được xác

định bởi số điểm ảnh thuộc mỗi cell. Ví dụ: nếu ta có một hình ảnh 128×128 với $\text{pixel_per_cell} = 4 \times 4$ thì sẽ có $32 \times 32 = 1024$ cell, $\text{pixel_per_cell} = 32 \times 32$, sẽ có $4 \times 4 = 16$ cell. Với mỗi cell trong bức ảnh, ta cần xây dựng 1 biểu đồ cường độ dốc. Mỗi pixel sẽ được lấy phiếu bầu vào vào biểu đồ, trọng số của mỗi phiếu bầu chính là cường độ dốc tại pixel đó. Cuối cùng, mỗi pixel đóng góp một phiếu bầu có trọng số vào biểu đồ. Lúc này, chúng ta có thể thu thập và ghép các biểu đồ này để tạo ra vector đặc trưng cuối cùng. Tuy nhiên, ta sẽ chuẩn hóa các block để có được kết quả tốt hơn.



Hình 2.9: Kết quả sau khi lấy phiếu bầu

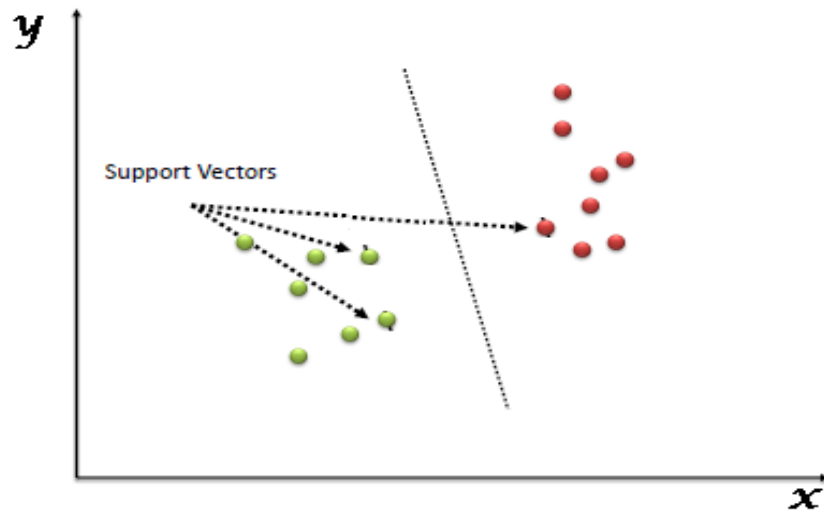
+ Chuẩn hóa các khối: một lần nữa, ta cần chia các khối giống như chia ô ở phía trên. Đơn vị của ta không còn là các điểm ảnh nữa mà là các ô. Người ta thường sử dụng hoặc 2×2 hoặc 3×3 cell_per_block có được độ chính xác hợp lý trong hầu hết các trường hợp. Các khối này sẽ chồng lên nhau, tiếp đến ta sẽ tiến hành thu thập và ghép các biểu đồ của ô trong khối.

+ Thu thập tất cả các biểu đồ cường độ hướng độ dốc để tạo ra vector đặc trưng cuối cùng.

2.4.3 THUẬT TOÁN SUPPORT VECTOR MACHINE(SVM)

SVM là một thuật toán giám sát, nó có thể sử dụng cho cả việc phân loại hoặc đệ quy. Tuy nhiên nó được sử dụng chủ yếu cho việc phân loại. Trong thuật toán này, chúng ta vẽ đồ thị dữ liệu là các điểm trong n chiều (ở đây n là số lượng các tính năng ta có) với giá trị của mỗi tính năng sẽ là một phần liên kết. Sau đó chúng

ta thực hiện tìm một con đường để phân chia các lớp. Con đường này là một đường thẳng có thể phân chia các lớp ra thành hai phần riêng biệt. Support Vectors hiểu một cách đơn giản là các đối tượng trên đồ thị tọa độ quan sát, SVM là một biên giới để chia hai lớp tốt nhất.



Hình 2.10: SVM phân chia các Support Vectors thành 2 phần riêng biệt

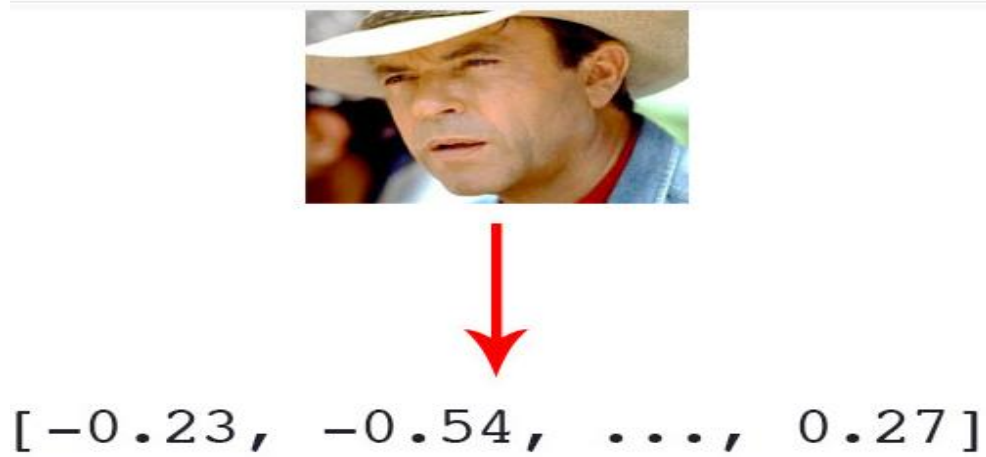
Trong SVM, rất dễ dàng để có một siêu phẳng tuyến tính (linear hyper-plane) để chia thành hai lớp. Nói một cách đơn giản nó thực hiện một số biết đổi dữ liệu phức tạp, sau đó tìm ra quá trình tách dữ liệu dựa trên các nhãn hoặc đầu ra mà chúng ra đã xác định trước.

2.4.4 DATASET LABELED FACE IN WILD (LFW)

LFW là một cơ sở dữ liệu các bức ảnh khuôn mặt được thiết kế để nghiên cứu vấn đề nhận dạng khuôn mặt không bị giới hạn. Tập dữ liệu chứa hơn 13.000 hình ảnh khuôn mặt được thu thập từ web. Mỗi khuôn mặt đã được dán nhãn với tên của người trong hình. 1680 người trong hình có hai hoặc nhiều ảnh khác nhau trong bộ dữ liệu. Những khuôn mặt này chúng được phát hiện bởi giải thuật tìm khuôn mặt Viola-Jones.

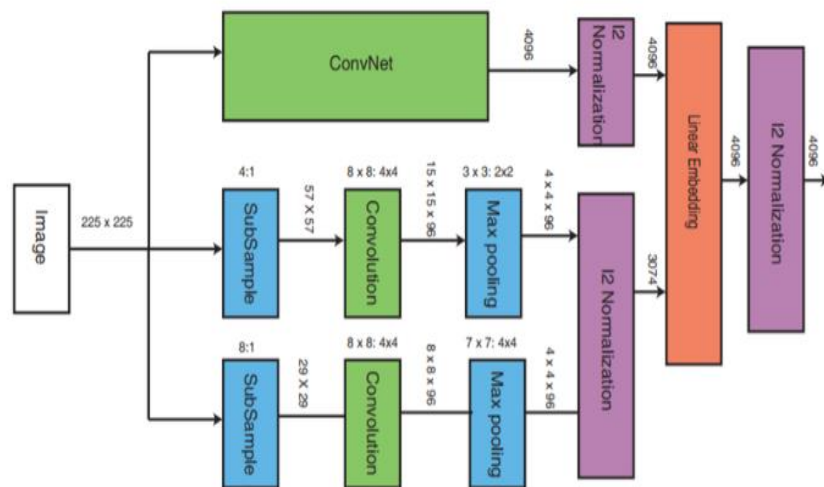
2.4.5 TRÍCH ĐẶC TRƯNG VÀ HUẤN LUYỆN

Đối với mạng nhận dạng khuôn mặt dlib, vector đặc trưng đầu ra là 128-D (tức là danh sách 128 số có giá trị thực) được sử dụng để định lượng khuôn mặt.



Hình 2.11: Định lượng khuôn mặt sử dụng vector đặc trưng 128-D

Việc tạo ra các vector đặc trưng từ ảnh đầu vào được thực hiện trên mô hình deep learning ConvNet sử dụng hàm Triplet.

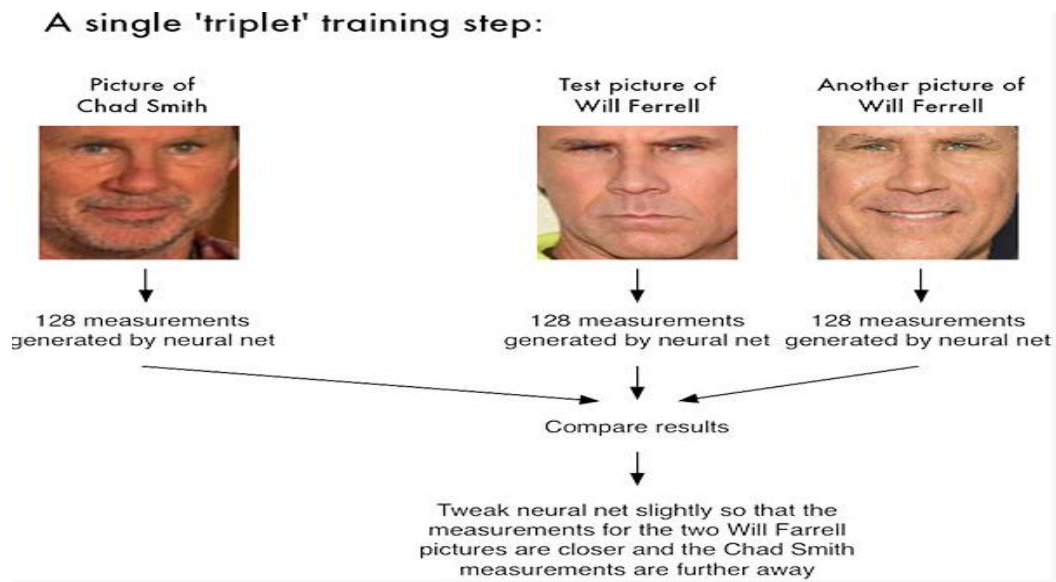


Hình 2.12: Mô hình Convnet tương trưng chuyển ảnh thành vector 4096 chiều

Đây là một deep learning network với cấu trúc 3 nhánh. Với 1 ảnh đưa vào, chúng ta sẽ thu được 1 vector cuối cùng đầu ra (Trong ảnh là 4096 chiều, còn dlib sẽ

đưa về 128 chiều). Điểm khác ở đây là ảnh sẽ được phân tích theo 3 mô hình khác nhau, mô hình đã được chứng minh hiệu quả trong nhiều bài toán xử lý ảnh.

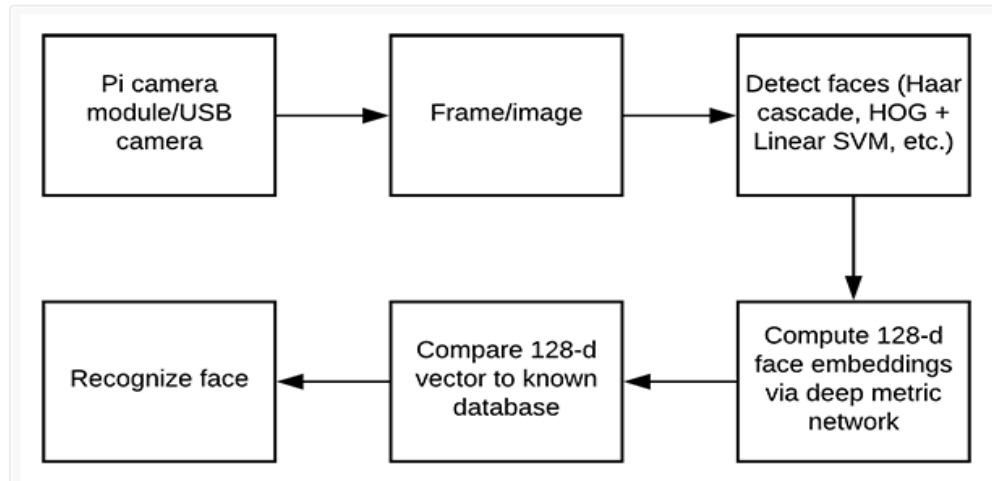
Việc đào tạo mạng được thực hiện bằng cách sử dụng bài toán Triplet (bộ ba): bộ ba của chúng ta bao gồm: 1 ảnh mặt của 1 người bất kỳ, 1 ảnh mặt khác của người đó, 1 ảnh mặt của người khác. Với việc huấn luyện mô hình như thế, chúng ta sẽ có thêm thông tin về mối quan hệ giữa các ảnh, điều này giúp mô hình chúng ta phù hợp hơn nhiều với bài toán. Mạng nhận dạng khuôn mặt dlib đã được Davis King đào tạo trên bộ dữ liệu khoảng 3 triệu hình ảnh. Trên LFW dataset mạng so sánh và cho ra kết quả chính xác tới 99,38% so với các phương pháp tiên tiến hiện đại khác.



Hình 2.13: Quá trình đào tạo trong bài toán triplet

2.5 NHẬN DẠNG KHUÔN MẶT NGƯỜI

Nguyên tắc chung của các phương pháp nhận dạng khuôn mặt hiện này là thực hiện trích chọn đặc trưng của ảnh cần nhận dạng và đối chiếu với đặc trưng của tập ảnh mẫu đã huấn luyện. Ở mạng nhận dạng khuôn mặt dlib đặc trưng để so sánh sẽ là các vector đặc trưng 128-D. Nếu khoảng cách giữa các vector đặc trưng là thỏa mãn điều kiện quy định trước thì khuôn mặt sẽ nhận diện thành công.



Hình 2.14: Quá trình nhận diện mặt người của toàn bộ hệ thống

Chương 3 THIẾT KẾ HỆ THỐNG

3.1 YÊU CẦU HỆ THỐNG

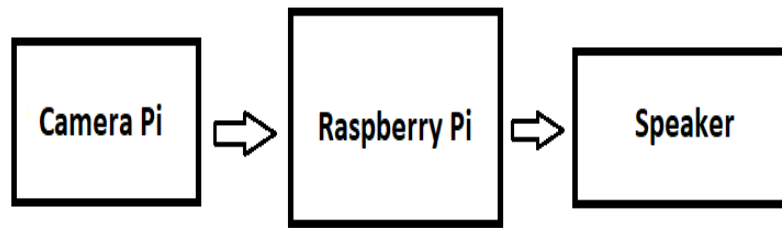
Chức năng:

- Thêm hình ảnh nhân viên trực tiếp.
- Nhận diện khuôn mặt để thực hiện điểm danh.
- Thông báo cho nhân viên biết đã điểm danh thành công.
- Ghi dữ liệu thời gian vào, ra của nhân viên.

Đặc tính:

- Hệ thống dễ thi công.
- Hệ thống có tính linh hoạt cao.
- Hệ thống sử dụng các linh kiện rẻ, dễ dàng sử dụng.

3.2 MÔ HÌNH HỆ THỐNG



Hình 3.1: Sơ đồ mô hình hệ thống

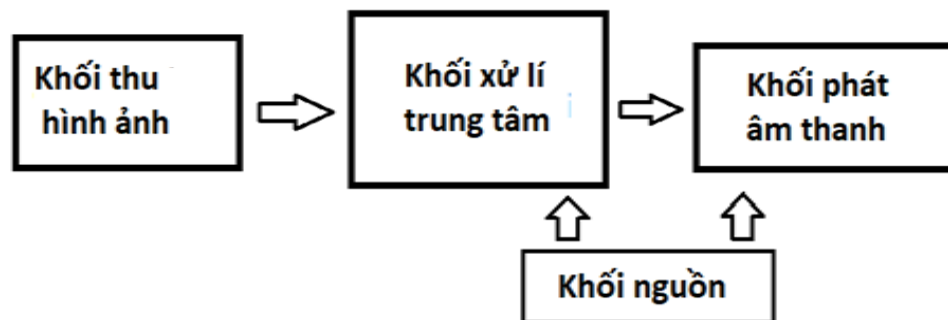
Với mục tiêu tạo ra một hệ thống hoạt động ổn định về phần cứng và tương tác dễ dàng với người dùng ở phần mềm, nên trước khi đưa ra những giải pháp cụ thể, tác giả đã xây dựng khái quát mô hình hoạt động chung của hệ thống như hình sau:

Theo như mô hình đã đề ra, hệ thống sẽ có bốn khối chính:

- Camera Pi sẽ có nhiệm vụ chụp ảnh và quay video.
- Raspberry Pi sẽ liên tục xử lý hình ảnh, video nhận từ camera truyền về, từ đó so sánh với dữ liệu hình ảnh của hệ thống và đưa ra kết quả.
- Speaker sẽ phát ra âm thanh khi nhận diện thành công.

3.3 THIẾT KẾ PHẦN CỨNG

3.3.1 SƠ ĐỒ KHỐI PHẦN CỨNG



Hình 3.2: Sơ đồ khối hệ thống

- Chức năng các khối:

+ Khối thu hình ảnh: lấy hình ảnh từ môi trường gửi về khối xử lý trung tâm.

+ Khối xử lý trung tâm: nhận dữ liệu ảnh từ khối thu hình ảnh, xử lý nhận diện khuôn mặt. Sau khi nhận diện xong sẽ xuất tín hiệu điều khiển đến khối phát âm thanh.

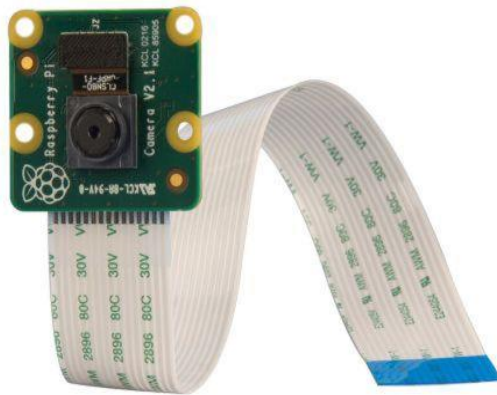
+ Khối phát âm thanh: nhận tín hiệu điều khiển từ khối xử lý trung tâm và phát ra âm thanh báo hiệu cho hệ thống.

+ Khối nguồn: cấp nguồn cho khối xử lý trung tâm và khối phát âm thanh.

3.3.2 LỰA CHỌN PHẦN CỨNG

3.3.2.1 Camera

Trên thị trường hiện nay có rất nhiều loại Camera có thể sử dụng trong nghiên cứu này như Camera IP, Camera IP Wifi..., tuy nhiên, kích thước của chúng khá lớn, cũng như giá thành cao. Vì vậy, nghiên cứu sử dụng Raspberry Pi Camera Module V1, có giá vừa phải, kích thước nhỏ, rất phù hợp để xây dựng mô hình, và đồng thời giá thành lại rất phù hợp để làm nghiên cứu.



Hình 3.3: Raspberr Pi Camera Module V1

Raspberry Pi Camera Module V1 là loại Module chất lượng cao 5 megapixel. Sử dụng cảm biến hình ảnh OV5647 tùy chỉnh được thiết kế thêm vào bảng cho Raspberry Pi, sử dụng tương thích với tất cả các dòng Raspberry Pi từ trước đến

nay, chất lượng hình ảnh tốt, độ phân giải cao và có khả năng quay phim ở chất lượng HD. Nó gắn với Raspberry Pi bằng một trong những lỗ nhỏ trên bề mặt bảng và sử dụng giao diện CSI dành riêng, được thiết kế đặc biệt để kết nối với máy ảnh.

- Tính năng, đặc điểm:

- + Kích thước: 25x24x9 mm.
- + Máy ảnh 5 megapixel có khả năng chụp ảnh 2592 x 1944 pixel.
- + Chụp video với độ phân giải 1080P 30, 720P 60 và VGA 640x480P 60.
- + Lens: Fixed Focus.
- + Connector: Ribbon connector.
- + Tất cả phần mềm được hỗ trợ trong phiên bản mới nhất của Hệ điều hành Raspbian.

3.3.2.2 Khối xử lý trung tâm

Có rất nhiều lựa chọn để làm khối xử lý, ví dụ như: Laptop, Raspberry Pi,.... Tuy nhiên, nghiên cứu sử dụng Raspberry Pi làm nhiệm vụ xử lý ảnh nhận được từ Raspberry Pi Camera để điều khiển phát âm thanh và nhận diện khuôn mặt vì giá thành rẻ và kích thước nhỏ, phù hợp với mô hình dự kiến.



Hình 3.4: Raspberry Pi 3 Model B

Raspberry Pi là một trong những kit nhúng được ứng dụng nhiều nhất như một máy tính mini có thể lập trình điều khiển ngõ ra theo ý muốn của người lập trình.

Raspberry Pi rất thích hợp cho người sử dụng muốn phát triển nhúng có kết hợp đồ họa, âm thanh...

Raspberry Pi 3 Model B là thế hệ thứ 3 của gia đình Raspberry Pi, nó ra đời vào tháng 2 năm 2016. Thông tin cấu hình Raspberry Pi 3:

- + Broadcom BCM2837 chipset running at 1.2 GHz.
- + 64-bit quad-core ARM Cortex-A53.
- + 802.11 b/g/n Wireless LAN.
- + Bluetooth 4.1 (Classic & Low Energy).
- + Dual core Videocore IV® Multimedia co-processor.
- + 1 GB LPDDR2 memory.
- + Supports all the latest ARM GNU/Linux distributions and Windows 10 IoT.
- + MicroUSB connector for 2.5 A power supply.
- + 1 x 10/100 Ethernet port.
- + 1 x HDMI video/audio connector.
- + 1 x RCA video/audio connector.
- + 4 x USB 2.0 ports.
- + 40 GPIO pins.
- + Chip antenna.
- + DSI display connector.
- + MicroSD card slot.
- + Dimensions: 85 x 56 x 17 mm.

3.3.2.3 Khối âm thanh

Sử dụng speaker VSP-CK4 với thiết kế nhỏ gọn, độ bền cao, dễ sử dụng rất thích hợp để sử dụng cho đề tài :

+ Loa sử dụng nguồn input là 5V, vì vậy có thể cấp nguồn cho loa thông qua cổng USB của máy tính rất tiện dụng. Ngoài ra, có thể sử dụng sạc cho điện thoại để làm nguồn cho loa hoạt động trong những chuyến dã ngoại hay sinh hoạt ngoài trời rất cơ động.

+ Kết nối với thiết bị phát âm thanh thông qua jack 3.5 tiện dụng, phù hợp với hầu hết các thiết bị nghe nhạc, thiết bị di động MP3/MP4, iPod, iPad, iPhone, máy nghe đĩa CD,DVD, PC, laptop và đặc biệt là Raspberry cũng sử dụng jack 3.5.



Hình 3.5: Loa VSP-CK4

3.3.2.4 Khối nguồn

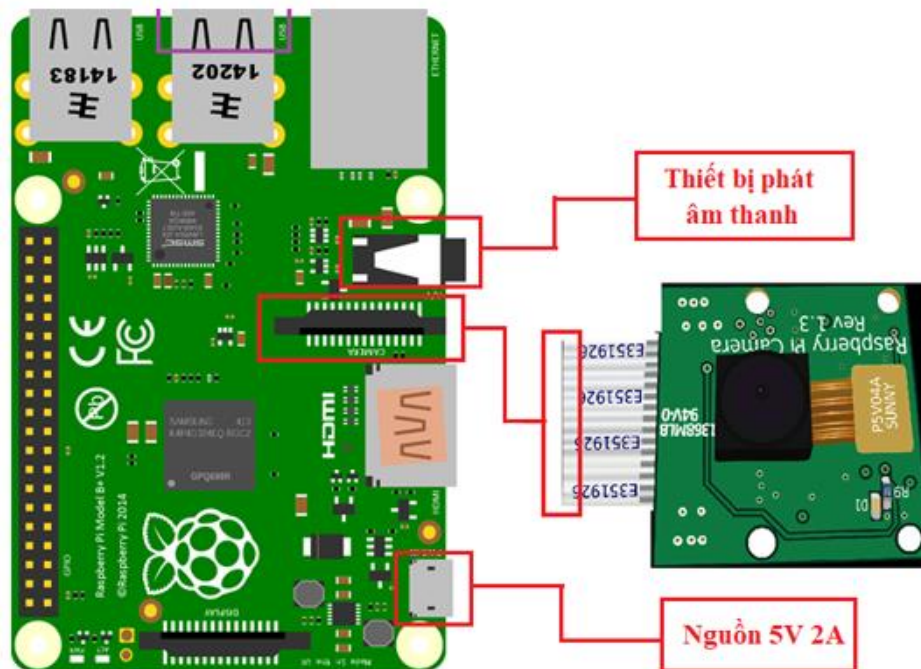
Sử dụng Adapter 5V/2A để cấp nguồn cho Raspberry Pi và loa thông qua jack USB. Nguồn này sẽ đáp ứng đủ công suất cho cả Raspberry và loa hoạt động bình thường.



Hình 3.6: Adapter 5V/2A

3.3.3 KẾT NỐI PHẦN CỨNG

Phần cứng được thiết kế gồm một Raspberry Pi 3, một camera Raspberry camera V1, một loa phát âm thanh) Sau đây là hình ảnh sau khi kết nối phần cứng:



Hình 3.7: Cách kết nối phần cứng

3.4 THIẾT KẾ PHẦN MỀM

3.4.1 CHỨC NĂNG HOẠT ĐỘNG CỦA PHẦN MỀM

Phát hiện khuôn mặt trong ảnh thu từ camera trong thời gian thực, nhận diện khuôn mặt được phát hiện sau đó đưa ra tín hiệu điều để phát ra âm thanh thông qua loa để báo hiệu nhận diện thành công và ghi thời gian vào bảng tính để lưu lại giờ làm việc của từng nhân viên theo từng ngày trong tháng.

3.4.2 QUY TRÌNH XỬ LÝ CỦA HỆ THỐNG

Hệ thống được xử lý theo quy trình sau:

Chụp ảnh từ camera để tạo dataset cho từng nhân viên trong công ty với mỗi nhân viên sẽ gồm 10 ảnh khuôn mặt có góc chụp khác nhau và lưu vào cùng một thư mục với tên thư mục là mã số nhân viên.

Xử lý phát hiện khuôn mặt để mã hoá khuôn mặt của từng bức hình thành vector đặc trưng theo phương pháp HOG kết hợp SVM được thực hiện bởi thư viện dlib. Sau khi đã trích đặc trưng xong sẽ tiến hành nhận diện khuôn mặt.

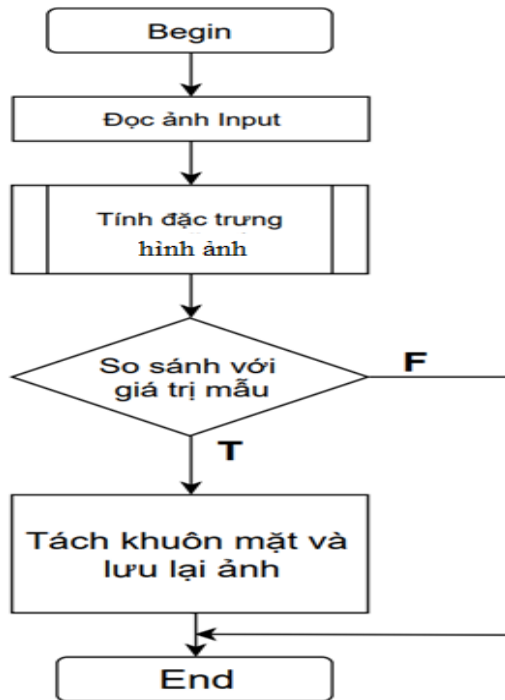
Xử lý nhận diện khuôn mặt sử dụng phương pháp phát hiện khuôn mặt của Viola-Jones vì Raspberry có cấu hình khá thấp, để có thể chạy thời gian thực đề tài đã tính toán và lựa chọn thuật toán này để tách khuôn mặt khi phát hiện người trong tầm nhìn của camera sau đó tiến hành tính toán đặc trưng của khuôn mặt hiện tại dùng thư viện dlib. Cuối cùng là so sánh đặc trưng của khuôn mặt hiện tại với từng đặc trưng trong dataset của hệ thống nếu trùng khớp sẽ tiến hành lưu thời gian vào bảng tính.

3.4.3 LƯU ĐỒ HOẠT ĐỘNG

3.4.3.1 Lưu đồ phát hiện khuôn mặt

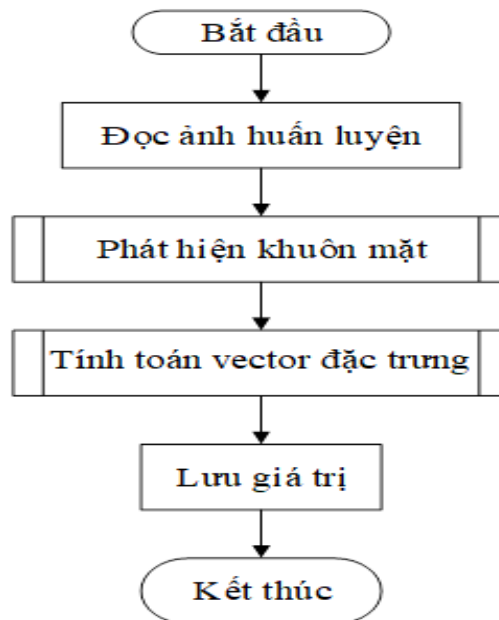
Phát hiện khuôn mặt là bước đầu tiên trong quá trình nhận diện khuôn mặt. Bản chất của phát hiện khuôn mặt là tìm kiếm và định vị trí khuôn mặt trên khung ảnh. Phương pháp phát hiện khuôn mặt được sử dụng là phương pháp sử dụng các đặc trưng Haar-like.

Ảnh thu vào từ camera sẽ được chuyển đổi từ không gian ảnh màu BGR sang ảnh xám, sau đó được tiến hành cân bằng histogram trên ảnh xám để thuận tiện cho việc xử lý. Sau khi tìm được khuôn mặt, ta sẽ đánh dấu vị trí khuôn mặt trên khung ảnh. Sau cùng ta sẽ lưu ảnh khuôn mặt tìm được để sử dụng cho bước tiếp theo.



Hình 3.8: Lưu đồ phát hiện khuôn mặt

3.4.3.2 Lưu đồ huấn luyện



Hình 3.9: Lưu đồ huấn luyện

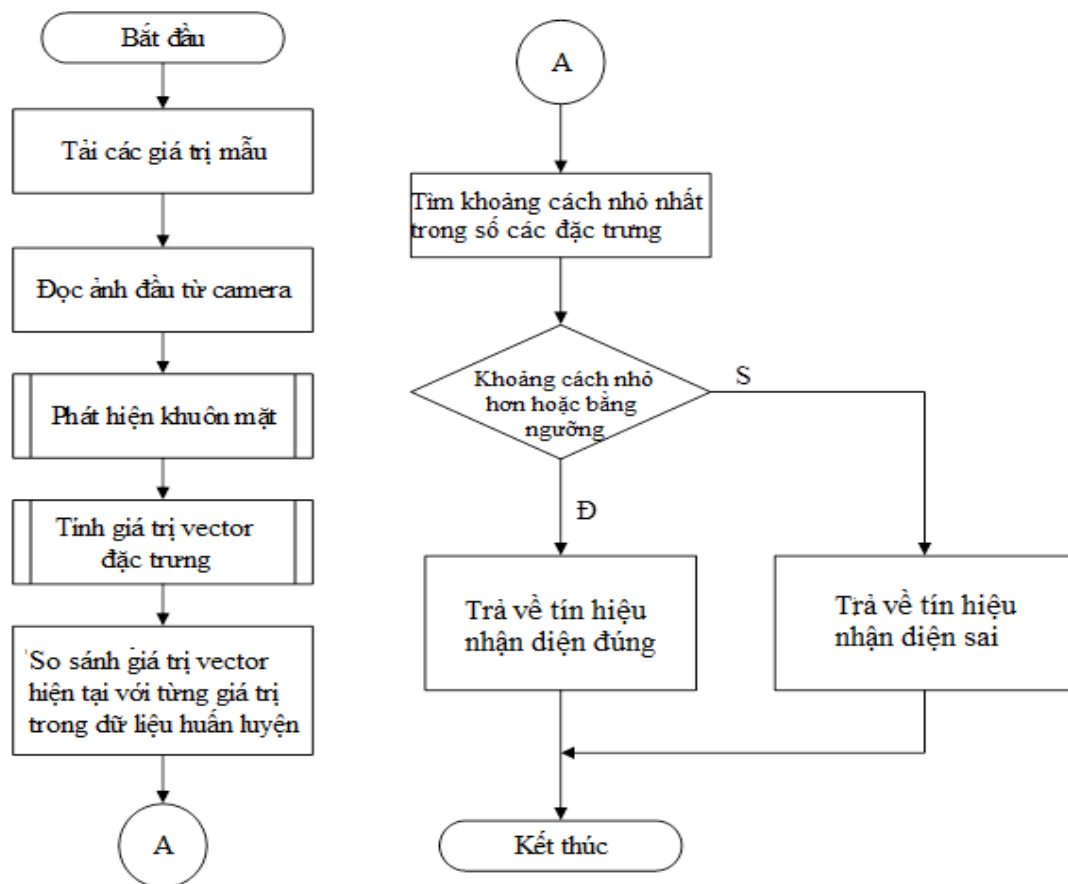
Ảnh huấn luyện được đọc lần lượt từng ảnh. Với mỗi ảnh sẽ được thực hiện các bước sau:

- Phát hiện khuôn mặt: chương trình sẽ dò tìm các khuôn mặt trên ảnh, sau đó đánh dấu khuôn mặt lớn nhất trong ảnh (ảnh có nhiều khuôn mặt) sau đó tách khuôn mặt vừa tìm được.

- Ảnh khuôn mặt sẽ được xử lí tính toán , mỗi ảnh sẽ tương ứng với các vector đặc trưng riêng.

- Sau khi tính giá trị vector đặc trưng của từng ảnh ta sẽ lưu tất cả các giá trị đó vào một tập tin dữ liệu để phục vụ cho việc nhận dạng.

3.4.3.3 Lưu đồ nhận diện khuôn mặt



Hình 3.10: Lưu đồ nhận diện khuôn mặt

Quá trình nhận diện bắt đầu bằng việc tải tập cơ sở dữ liệu lên, dữ liệu mẫu chứa các giá trị vector đặc trưng của các ảnh mẫu có trong cơ sở dữ liệu. Sau đó sẽ đọc ảnh từ camera để lấy ảnh người cần nhận diện.

Tiếp tục tìm khuôn mặt có trong ảnh, sau đó căn chỉnh và tách khuôn mặt. Khuôn mặt sau khi tách sẽ được xử lý để tính giá trị vector đặc trưng.

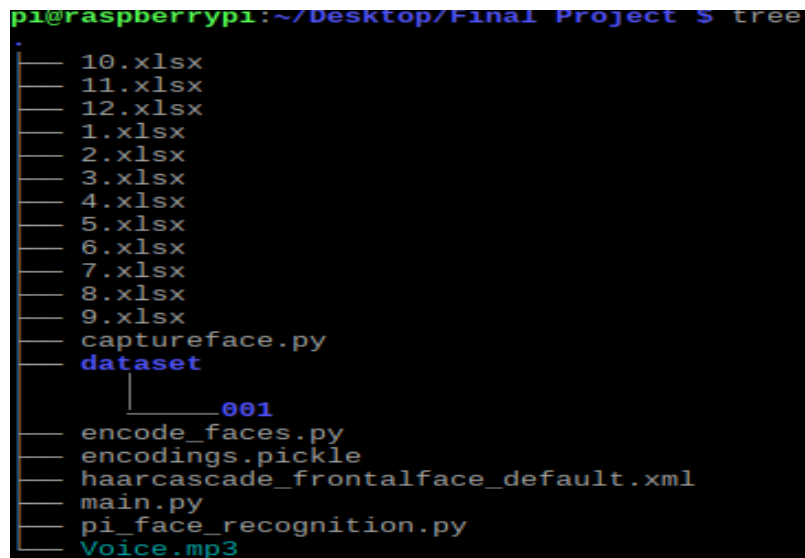
Với giá trị đặc trưng vừa tìm được sẽ mang đi tính khoảng cách với các đặc trưng có trong tập cơ sở dữ liệu. Khi có được tất cả các giá trị khoảng cách, ta sẽ tìm khoảng cách nhỏ nhất trong đó. Khoảng cách càng nhỏ biểu thị cho độ giống với ảnh trong dữ liệu mẫu càng lớn, và ngược lại khoảng cách càng lớn thì độ giống càng thấp.

Mang giá trị khoảng cách nhỏ nhất so sánh với giá trị ngưỡng, ở đây nhóm chọn ngưỡng bằng 0.3. Nếu nhỏ hơn ngưỡng thì ta kết luận nhận diện đúng người trong dữ liệu hệ thống. Ngược lại, nếu lớn hơn ngưỡng ta kết luận nhận diện không đúng người trong dữ liệu hệ thống.

3.4.4 CÁC BƯỚC THỰC HIỆN

3.4.4.1 Cấu trúc phần mềm của hệ thống

Phần mềm được xây dựng với cấu trúc như hình :



Hình 3.11: Cấu trúc phần mềm

- Đầu tiên là folder dataset: đây là nơi chứa các thư mục con bao gồm hình chụp khuôn mặt của mỗi nhân viên trong hệ thống.
 - + Thư mục 001: chứa hình của nhân viên có mã số 001.
- Captureface.py là chương trình dùng để chụp lại khuôn mặt của nhân viên khi ta cần thêm vào hệ thống.
- Encode_faces.py là chương trình dùng để trích vector đặc trưng cho từng khuôn mặt có trong dataset.
- Encodings.pickle là file lưu lại đặc trưng tương ứng cho mỗi khuôn mặt của mỗi nhân viên trong hệ thống.
- Pi_face_recognition.py là chương trình dùng để nhận diện khuôn mặt nhân viên và ghi thời gian chấm công vào excel file.
- Haarcascade_frontalface_default.xml: là file chứa đặc trưng dùng để phát hiện khuôn mặt của OpenCV.
- Main.py là chương trình giao diện của hệ thống.
- Voice.mp3 là file âm thanh dùng để báo hiệu khi nhận diện đúng khuôn mặt có trong hệ thống.
- 12 file excel đại diện cho 12 tháng trong năm. Hệ thống sẽ ghi thời gian chấm công tương ứng với từng tháng trong năm vào các file excel tương ứng.

3.4.4.2 Cài đặt công cụ và thư viện cần thiết

Trước khi cài đặt các công cụ và thư viện cần thiết ta cần cập nhật và nâng cấp driver cho raspberry pi với câu lệnh :

```
sudo apt-get update && sudo apt-get upgrade
```

Tiếp theo ta tiến hành cài đặt các công cụ phát triển, bao gồm cả Cmake nó giúp ta cấu hình OpenCV khi tiến hành build.

```
sudo apt-get install build-essential cmake pkg-config
```

Một số packages I/O cho việc định dạng ảnh, cho phép tải các định dạng ảnh khác nhau từ ổ đĩa. Ví dụ như JPEG, PNG, TIFF...


```
sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev
```

Một số packages I/O định dạng các video, cho phép đọc các định dạng video khác nhau từ ổ đĩa và làm việc trực tiếp với các luồng video.

```
sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev  
libxvidcore-dev libx264-dev
```

Một số gói phục vụ cho thư viện OpenCV khác.

```
sudo apt-get install libgtk2.0-dev libgtk-3-dev libatlas-base-dev gfortran
```

Nếu raspberry đã có python3 thì bỏ qua bước cài đặt python.

```
sudo apt-get install python3-dev
```

Tải OpenCV về và giải nén bao gồm opencv và opencv_contrib. Phải đảm bảo rằng phiên bản các cả 2 là giống nhau.

```
wget -O opencv.zip https://github.com/Itseez/opencv/archive/3.3.0.zip
```

```
unzip opencv.zip
```

```
wget -O opencv_contrib.zip
```

```
https://github.com/Itseez/opencv_contrib/archive/3.3.0.zip
```

```
unzip opencv_contrib.zip
```

Tiếp theo là cài đặt pip. Dùng để quản lí các package trên python, giúp cho việc quản lí và cài đặt các packages trên python dễ dàng hơn.

```
wget https://bootstrap.pypa.io/get-pip.py
```

```
sudo python3 get-pip.py
```

Cài Numpy package, gói này giúp việc xử lí số dễ dàng trên python.

```
pip3 install numpy
```

Tiến hành biên dịch và cài đặt OpenCV. Trước khi cài đặt ta cần tăng kích thước không gian hoán đổi cho raspberry để đảm bảo nó không bị treo khi biên dịch vì vấn đề bộ nhớ. Mở file dphys-swapfile tại đường dẫn /etc/dphys-swapfile và chuyển `CONF_SWAPSIZE=1024` sau đó lưu lại và restart swap service.

```
sudo /etc/init.d/dphys-swapfile stop
```

```
sudo /etc/init.d/dphys-swapfile start
```

Tạo thư mục build bên trong folder chứa thư viện opencv và tiến hành build với cmake.

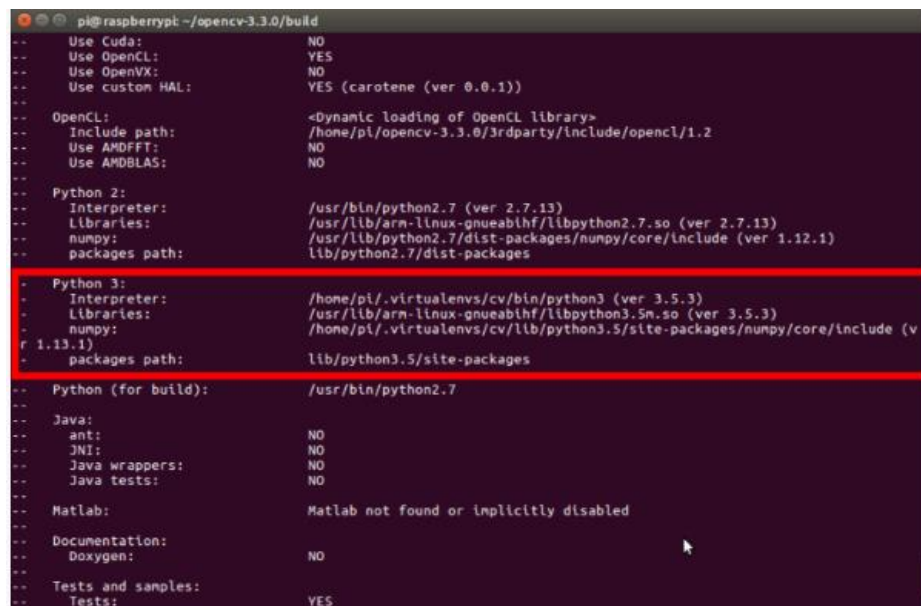
```
cd ~/opencv-3.3.0/
```

```
mkdir build
```

```
cd build
```

```
cmake -D CMAKE_BUILD_TYPE=RELEASE \
-D CMAKE_INSTALL_PREFIX=/usr/local \
-D INSTALL_C_EXAMPLES=ON \
-D INSTALL_PYTHON_EXAMPLES=ON \
-D OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib/modules \
-D CPU_BASELINE=SSE4_2 \-D CPU_DISPATCH= ... \
-D BUILD_NEW_PYTHON_SUPPORT=ON \
-D BUILD_opencv_python3=ON \
-D HAVE_opencv_python3=ON \
-D PYTHON_DEFAULT_EXECUTABLE=$(which python3) \
-D BUILD_EXAMPLES=ON ..
```

Sau khi build xong sẽ nhận được kết quả như hình:

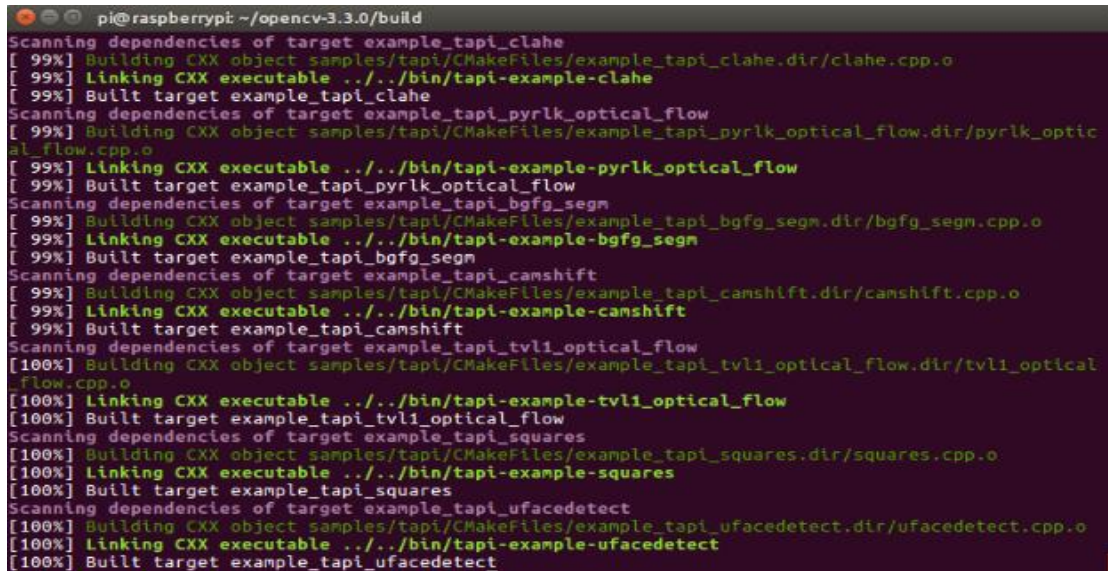


```
pi@raspberrypi:~/opencv-3.3.0/build
-- Use Cuda: NO
-- Use OpenCL: YES
-- Use OpenVX: NO
-- Use custom HAL: YES (carotene (ver 0.0.1))
--
-- OpenCL:
--   Include path: /home/pi/opencv-3.3.0/3rdparty/include/opencl/1.2
--   Use AMDFFT: NO
--   Use AMDBLAS: NO
--
-- Python 2:
--   Interpreter: /usr/bin/python2.7 (ver 2.7.13)
--   Libraries: /usr/lib/arm-linux-gnueabihf/libpython2.7.so (ver 2.7.13)
--   numpy: /usr/lib/python2.7/dist-packages/numpy/core/include (ver 1.12.1)
--   packages path: lib/python2.7/dist-packages
--
-- Python 3:
--   Interpreter: /home/pi/.virtualenvs/cv/bin/python3 (ver 3.5.3)
--   Libraries: /usr/lib/arm-linux-gnueabihf/libpython3.5m.so (ver 3.5.3)
--   numpy: /home/pi/.virtualenvs/cv/lib/python3.5/site-packages/numpy/core/include (v
--   1.13.1)
--   packages path: lib/python3.5/site-packages
--
-- Python (for build): /usr/bin/python2.7
--
-- Java:
--   ant: NO
--   JNI: NO
--   Java wrappers: NO
--   Java tests: NO
--
-- Matlab: Matlab not found or implicitly disabled
--
-- Documentation:
--   Doxygen: NO
--
-- Tests and samples:
--   Tests: YES
```

Hình 3.12: Kết quả sau khi build OpenCV

Sau khi build thành công tiến hành biên dịch OpenCV sử dụng lệnh.

`make -j4`



```
pi@raspberrypi: ~/opencv-3.3.0/build
Scanning dependencies of target example_tapi_clahe
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_clahe.dir/clahe.cpp.o
[ 99%] Linking CXX executable ../../bin/tapi-example-clahe
[ 99%] Built target example_tapi_clahe
Scanning dependencies of target example_tapi_pyrlk_optical_flow
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_pyrlk_optical_flow.dir/pyrlk_optical_flow.cpp.o
[ 99%] Linking CXX executable ../../bin/tapi-example-pyrlk_optical_flow
[ 99%] Built target example_tapi_pyrlk_optical_flow
Scanning dependencies of target example_tapi_bgfg_seg
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_bgfg_seg.dir/bgfg_seg.cpp.o
[ 99%] Linking CXX executable ../../bin/tapi-example-bgfg_seg
[ 99%] Built target example_tapi_bgfg_seg
Scanning dependencies of target example_tapi_camshift
[ 99%] Building CXX object samples/tapi/CMakeFiles/example_tapi_camshift.dir/camshift.cpp.o
[ 99%] Linking CXX executable ../../bin/tapi-example-camshift
[ 99%] Built target example_tapi_camshift
Scanning dependencies of target example_tapi_tvli_optical_flow
[100%] Building CXX object samples/tapi/CMakeFiles/example_tapi_tvli_optical_flow.dir/tvli_optical_flow.cpp.o
[100%] Linking CXX executable ../../bin/tapi-example-tvli_optical_flow
[100%] Built target example_tapi_tvli_optical_flow
Scanning dependencies of target example_tapi_squares
[100%] Building CXX object samples/tapi/CMakeFiles/example_tapi_squares.dir/squares.cpp.o
[100%] Linking CXX executable ../../bin/tapi-example-squares
[100%] Built target example_tapi_squares
Scanning dependencies of target example_tapi_ufacedetect
[100%] Building CXX object samples/tapi/CMakeFiles/example_tapi_ufacedetect.dir/ufacedetect.cpp.o
[100%] Linking CXX executable ../../bin/tapi-example-ufacedetect
[100%] Built target example_tapi_ufacedetect
```

Hình 3.13: Kết quả sau khi biên dịch OpenCV

Tiếp theo là cài đặt các thư viện cần thiết cho hệ thống sử dụng pip.

`pip3 install dlib`

`pip3 install face_recognition`

`pip3 install imutils`

`pip3 install openpyxl`

`pip3 install tkinter`

`pip3 install pygame`

3.4.4.3 Chức năng của các thư viện

- Thư viện `face_recognition`: là thư viện được xây từ nhận diện khuôn mặt của `dlib`. Nó cung cấp công cụ nhận diện khuôn mặt đơn giản, cho phép nhận diện khuôn mặt trong thư mục chứa hình ảnh.

- Thư viện `imutils`: có các chức năng tiện lợi để thực hiện các chức năng xử lý hình ảnh cơ bản như dịch, xoay, thay đổi kích thước, khung xương và hiển thị hình ảnh Matplotlib dễ dàng với OpenCV và Python 3.

- Thư viện openpyxl: là một thư viện Python để thao tác với các tệp Excel 2010. Bao gồm khởi tạo, đọc, ghi...
- Thư viện tkinter: là giao diện tiêu chuẩn của Python. Tkinter có sẵn trên hầu hết các nền tảng Unix, cũng như trên các hệ thống Windows.
- Thư viện pygame: chứa các hàm và lớp của python, cho phép sử dụng hỗ trợ SDL để phát cdroms, đầu ra âm thanh và video, đầu vào bàn phím, chuột và cần điều khiển.

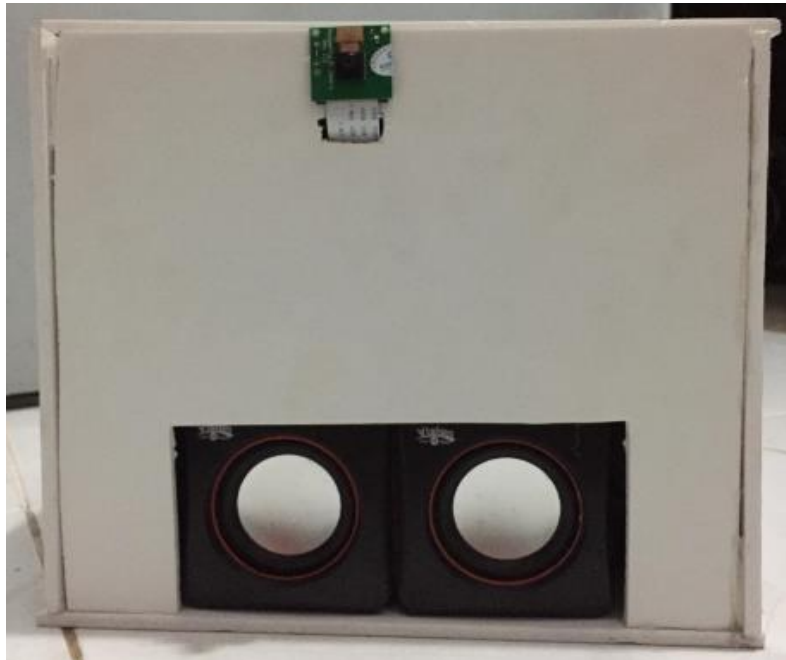
Chương 4 KẾT QUẢ

4.1 MÔ HÌNH THỰC TẾ CỦA HỆ THỐNG

Mô hình của hệ thống gồm các thành phần sau:

- Raspberry Pi (nằm bên trong hộp).
- Camera Pi: được gắn ở phía trước hộp.
- Loa: cắm trực tiếp vào Raspberry Pi đặt bên trong hộp.

Mô hình có chức năng: liên tục xử lý hình ảnh ở môi trường để liên tục ghi lại thời gian ra vào của nhân viên và báo hiệu mỗi khi ghi lại thành công.



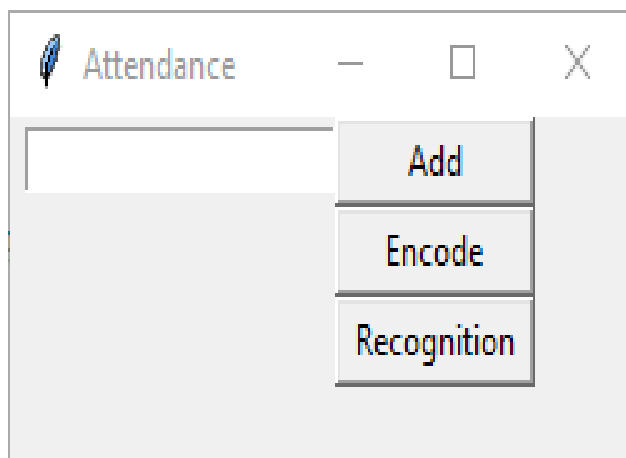
Hình 4.1: Mô hình hoàn chỉnh của hệ thống

4.2 HOẠT ĐỘNG CỦA HỆ THỐNG

Camera sẽ liên tục lấy hình ảnh từ môi trường và gửi về cho bộ xử lý để kiểm tra xem có khuôn mặt nào hay không, nếu có tiến hành tách khuôn mặt và tính toán đặc trưng của khuôn mặt đó rồi kiểm tra xem mẫu vừa lấy có tương thích với những

mẫu đã có sẵn trong dữ liệu của hệ thống hay không. Nếu mẫu vừa lấy là tương thích với những mẫu đã có trong hệ thống thì bộ xử lý sẽ tiến hành ghi dữ liệu thời gian vào bảng tính và phát ra âm thanh. Nếu mẫu vừa lấy không tương thích với những mẫu đã có trong hệ thống thì chấm công sẽ không thành công.

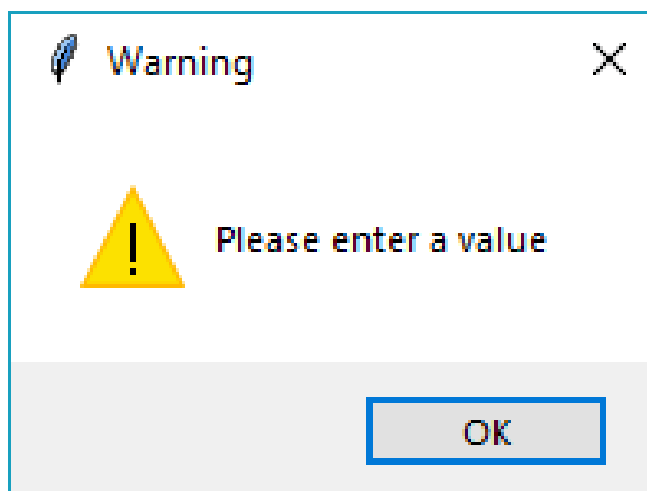
Giao diện khi khởi động hệ thống sẽ như sau:



Hình 4.2: Giao diện của hệ thống

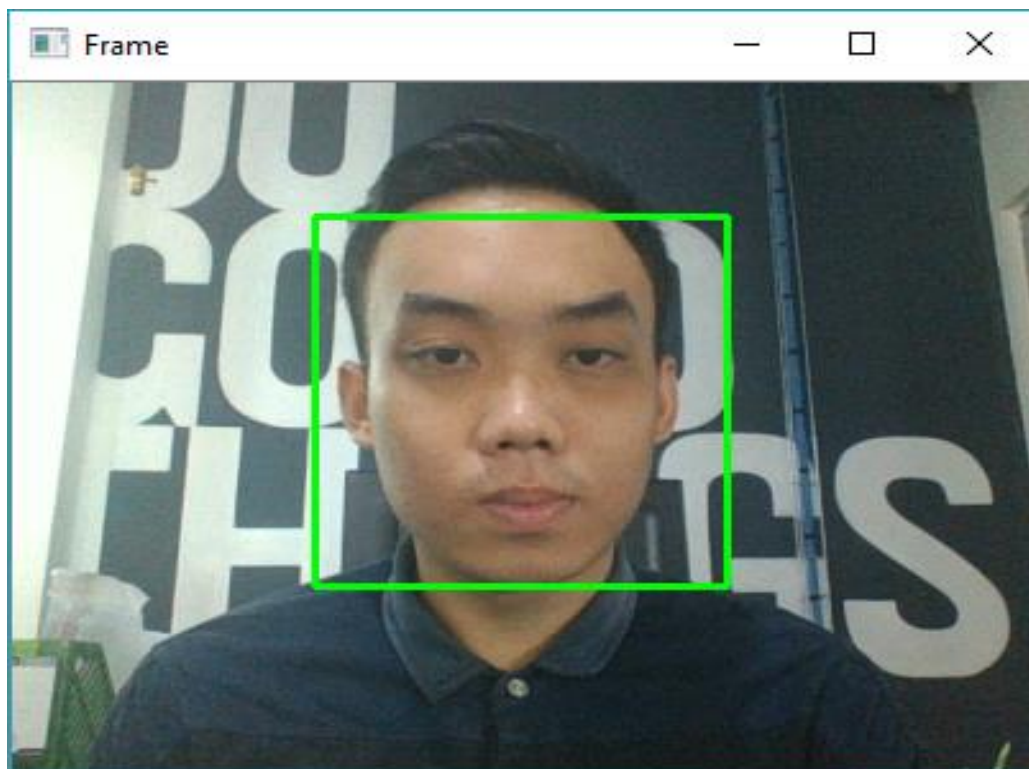
Có một ô trống được khởi tạo để người dùng có thể nhập mã số nhân viên của mình vào và tiến hành nhấn nút *Add* để thêm nhân viên vào cơ sở dữ liệu để tiến hành cho việc nhận diện điểm danh.

Nếu không nhập mã số mà ta nhấn *Add* thì hệ thống sẽ hiện ra cảnh báo và yêu cầu nhập mã số cho nhân viên.



Hình 4.3: Cảnh báo người dùng

Sau khi nhập mã số vào và nhấn nút *Add* thì sẽ hiện ra giao diện thêm ảnh cho nhân viên. Nhấn phím *K* trên bàn phím để tiến hành chụp và lưu ảnh vào thư mục chứa mã số nhân viên của nhân viên đó, nhấn phím *Q* để kết thúc.



Hình 4.4: Giao diện khi lưu ảnh nhân viên

Sau khi thêm ảnh để nhận diện cho từng nhân viên thì ta tiến hành nhấn nút *Encode* để thực hiện trích đặc trưng của từng bức ảnh đã chụp.

```
[INFO] quantifying faces...  
[INFO] processing image 1/17  
[INFO] processing image 2/17
```

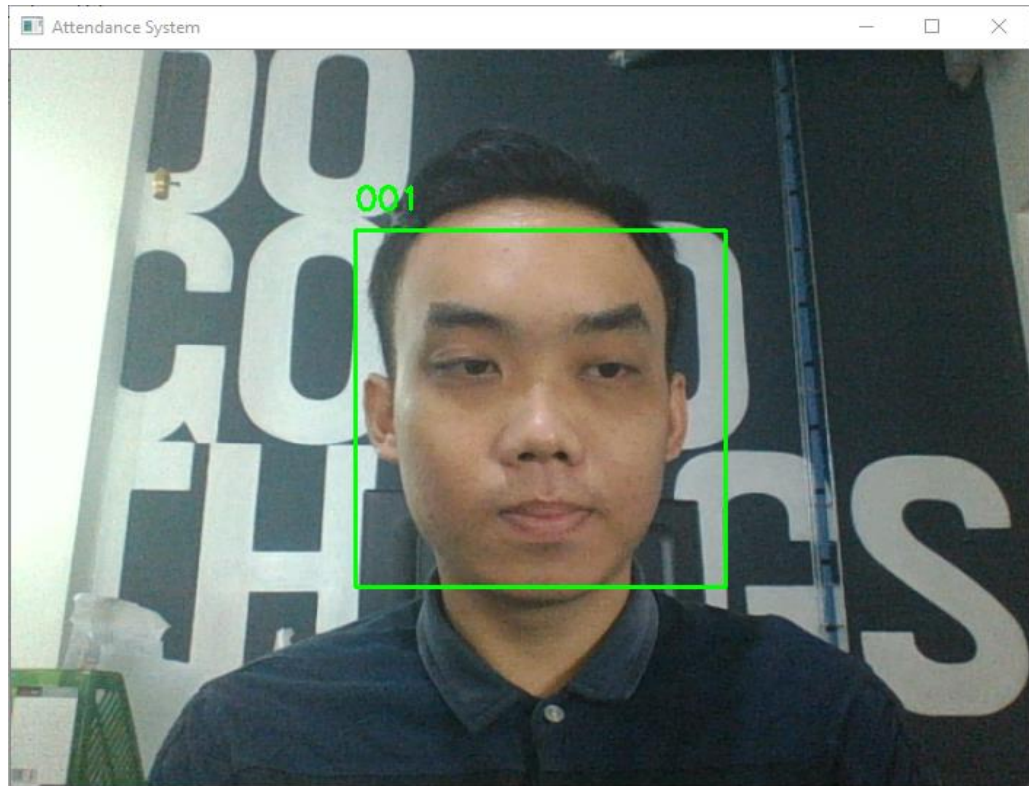
Hình 4.5: Trích đặc trưng của tất cả các ảnh hiện có

Sau khi trích hết tất cả các đặc trưng xong sẽ xuất hiện dòng này.

```
[INFO] processing image 16/17  
[INFO] processing image 17/17  
[INFO] serializing encodings...
```

Hình 4.6: Sau khi đã trích xong tất cả

Sau khi đã trích đặc trưng xong ta nhấn vào nút *Recognition* để tiến hành nhận diện khuôn mặt và điểm danh, khi nhấn vào nút *Recognition* sẽ tiến hành nhận diện nhân viên.



Hình 4.7: Kết quả sau khi nhận diện sẽ hiển thị trên màn hình

Sau khi quét xong thời gian vào và ra sẽ được lưu ở file Excel tương ứng với tháng hiện tại để quản lý, thời gian đầu tiên nhận diện được nhân viên sẽ được lưu ở mục *checkin*, còn những thời gian sau đó khi nhận diện được nhân viên sẽ lưu vào mục *checkout*.

Giờ vào của nhân viên sẽ được lưu theo thứ tự mã số của nhân viên và ngày tương ứng với số cột của Excel.

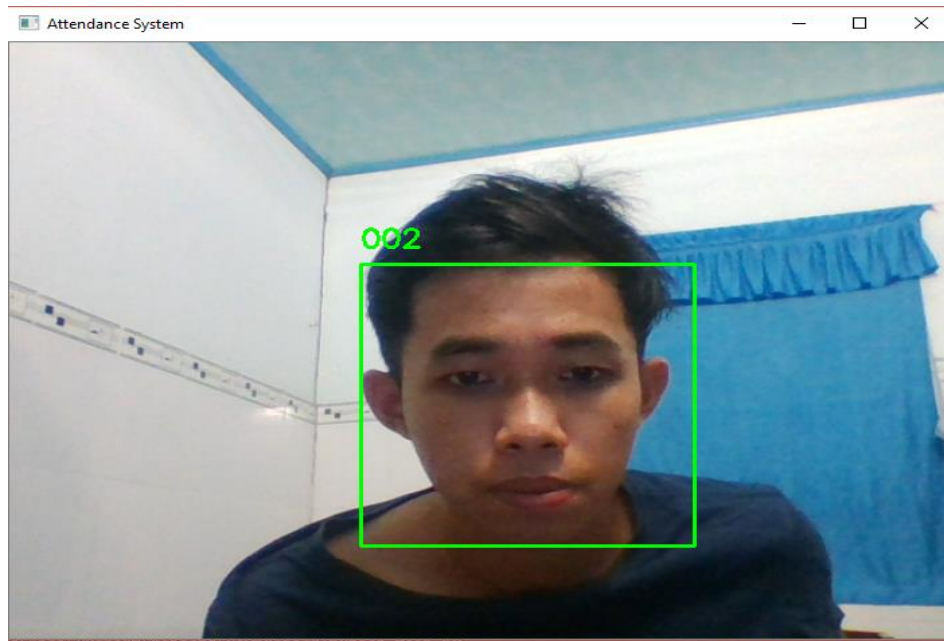
Ví dụ: hôm nay là ngày 27/5 và người có mã số 001 điểm danh thì sẽ được lưu ở hàng 1 tương ứng với mã số nhân viên và ở cột AA tương ứng với số ngày hiện tại. Ta được kết quả như hình bên dưới.

	Y	Z	AA	AB	AC		Y	Z	AA	AB	AC
1			22:17:35			1			22:18:42		
2						2					
3						3					
4						4					
5						5					
6						6					
7						7					
8						8					
9						9					
10						10					
11						11					
12						12					
13						13					
14						14					
15						15					
16						16					
17						17					
18						18					
19						19					
20						20					
21						21					
22						22					
23						23					
<div> <div> <div></div> <div></div> </div> <div> <div>Checkin</div> <div>Checkout</div> <div>+</div> </div> </div>						<div> <div> <div></div> <div></div> </div> <div> <div>Checkin</div> <div>Checkout</div> <div>+</div> </div> </div>					

Hình 4.8: Lưu thông tin giờ vào, giờ ra của nhân viên

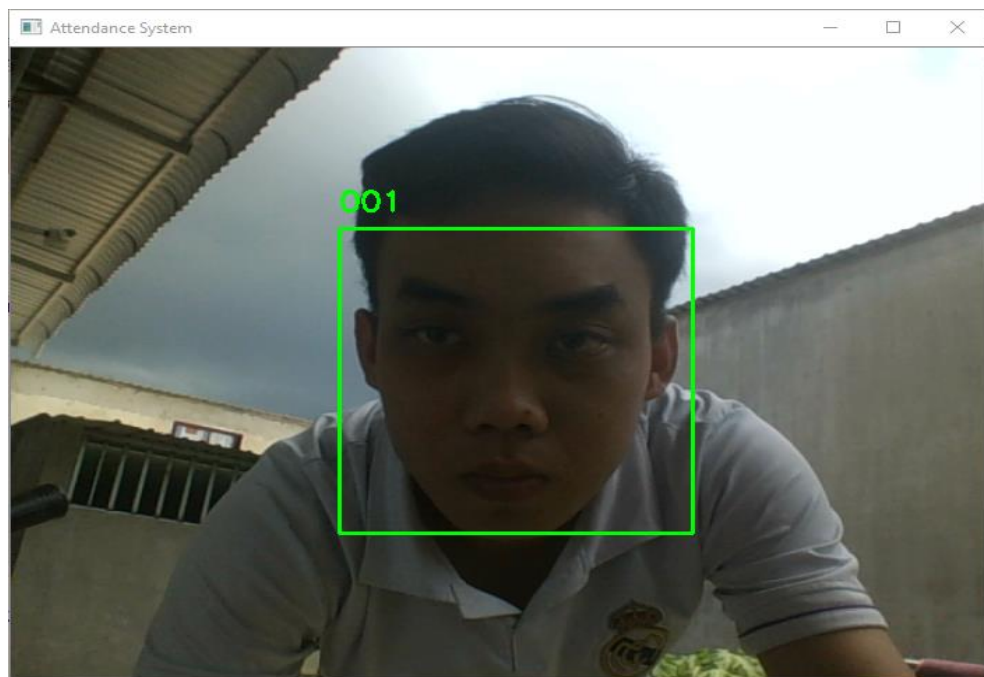
Sau khi thử nghiệm với một số người khác nhau ở một vài môi trường khác nhau, tác giả nhận được kết quả như sau:

Kết quả nhận diện vẫn chính xác sau khi thực hiện thêm ảnh cho nhân viên khác và tiến hành nhận diện.



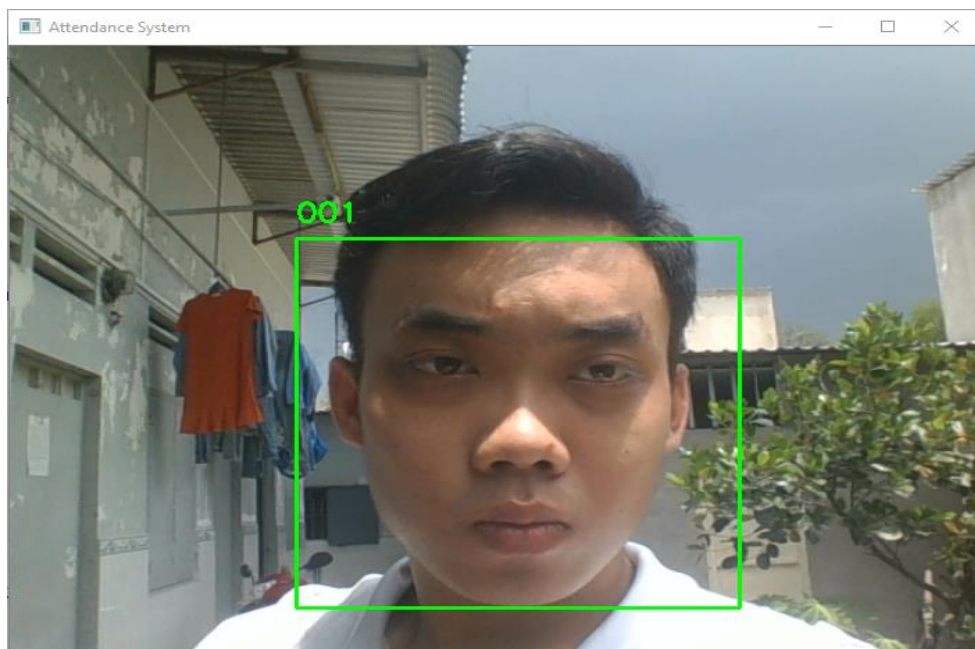
Hình 4.9: Kết quả nhận diện của một nhân viên khác

Kết quả nhận diện sau khi thực hiện nhận diện ở môi trường ngược sáng vẫn cho kết quả chính xác.



Hình 4.10: Kết quả sau khi nhận diện ngược sáng

Kết quả sau khi thực hiện nhận diện ngoài trời với điều kiện ánh sáng chói vẫn cho kết quả chính xác.



Hình 4.11: Kết quả nhận diện trong môi trường ánh sáng chói

Tuy nhiên khi thử nhận diện ở một góc nghiêng thì kết quả nhận diện lại không được như mong muốn.



Hình 4.12: Kết quả khi nhận diện với góc nghiêng

Vì vấn đề độ chính xác của đề tài được tác giả chú trọng hơn, nên tác giả đã giảm hệ số khoảng cách giữa các khuôn mặt để việc nhận diện được chính xác hơn nhưng đổi lại thì ở một số góc độ ví dụ như ở thử nghiệm trên thì việc nhận diện không cho kết quả như mong muốn.

Mã số nhân viên	Điều kiện thử nghiệm	Số lần thử	Số lần chính xác	Tỉ lệ
001	Ánh sáng chói	15	14	93.3%
001	Thiếu sáng	12	11	91.6%
002	Góc nghiêng	9	4	44.4%
002	Hệ số tolerance = 0.3	30	30	100%
001	Hệ số tolerance = 0.3	35	35	100%
002	Hệ số tolerance = 0.4	12	8	66.6%
001	Hệ số tolerance = 0.5	13	7	53.8%
Tổng				86.5%

Bảng 1: Bảng thống kê tỉ lệ chính xác qua các lần thử

Chương 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 KẾT LUẬN

Sau quá trình tìm hiểu, nghiên cứu lý thuyết về phương pháp phát hiện khuôn mặt, giải thuật nhận diện khuôn mặt, sử dụng thư viện mã nguồn mở OpenCV, dlib, ngôn ngữ lập trình Python, sau đó tiến hành thực nghiệm, đã xây dựng đề tài “Hệ thống nhận diện khuôn mặt điểm danh nhân viên”. Dựa theo mục tiêu đã thực hiện được các chức năng sau:

- Hệ thống có khả năng nhận diện được khuôn mặt người trong công ty.
- Trung bình việc nhận diện khuôn mặt và điểm danh cho một nhân viên khoảng 5-10 giây tùy vào góc nhận diện của người đó với camera và các yếu tố ngoại cảnh tác động.
- Nhóm đã thực hiện khảo sát với một số người đạt độ chính xác rất cao.
- Ghi giờ vào, giờ ra của từng nhân viên trong công ty và báo hiệu khi đã chấm công thành công.
- Hệ thống linh động, dễ dàng lắp đặt ở mọi vị trí.
- Dễ dàng sửa chữa khi gặp sự cố.
- Tỷ lệ chính xác qua những lần thử nghiệm đạt khoảng 90%

Tuy nhiên hệ thống vẫn còn tồn tại một số nhược điểm chưa khắc phục được như:

- Chưa phát hiện được những khuôn mặt ở những góc nghiêng, dẫn đến mất thời gian cho việc chấm công.
- Thời gian huấn luyện tệp chứa ảnh còn mất nhiều thời gian.

- Còn hạn chế trong việc xác định người thực hiện điểm danh là người thật hay là người trong ảnh.
- Raspberry Pi còn hạn chế về tốc độ xử lý dẫn đến việc nhận diện chưa được như mong đợi.

So với các công nghệ điểm danh bằng RFID, nhận diện vân tay hay thủ công thì việc thực hiện điểm danh bằng nhận diện khuôn mặt có những ưu và nhược điểm sau:

- Ưu điểm:
 - Thời gian thực hiện điểm danh nhanh chóng.
 - Sự chính xác của việc điểm danh được đảm bảo hơn so với các công cụ điểm danh khác.
 - Cách thực hiện điểm danh dễ dàng.
 - Không cần quản lý thẻ RFID tránh trường hợp quên hay mất thẻ.
 - Không bị các yếu tố khách quan tác động như bị mòn dấu vân tay hay đổ mồ hôi dẫn đến việc khó khăn trong việc điểm danh.
 - Không có chi phí phát sinh như khi dùng thẻ RFID.
- Nhược điểm:
 - Khi thêm nhân viên phải thực hiện trích đặc trưng của toàn bộ nhân viên.
 - Việc bảo trì sửa chữa gặp khó khăn do hệ thống cần nhiều thư viện để hỗ trợ.

5.2 HƯỚNG PHÁT TRIỂN

- Dữ liệu thời gian chấm công sẽ được đưa lên web để quản lý. Giúp việc đa truy cập trở nên dễ dàng hơn.
- Thực hiện huấn luyện ảnh trước khi cho vào Raspberry pi để rút ngắn thời gian.

- Tối ưu sự ảnh hưởng của các yếu tố: khoảng cách, ánh sáng đến sự chính xác của việc nhận diện.
- Khắc phục tình trạng điểm danh hộ bằng cách nhận diện khuôn mặt 3D tránh trường hợp dùng hình của người nào đó để thực hiện việc điểm danh.
- Nâng cao chất lượng nhận diện để đảm bảo việc nhận diện vẫn đúng và đồng thời nhận diện được nhiều góc khác nhau.

TÀI LIỆU THAM KHẢO

Tài liệu nước ngoài

- [1] Dr. Adrian Rosebrock. “*Deep Learning for Computer Vision with Python*”, 2018.
- [2] Viola. P, Jones. M, Rapid. “*Object detection using a boosted cascade of simple features*”, May 2004.
- [3] Viola. P, Jones. M, Robust. “*International Journal of Computer Vision*”, 2004.
- [4] Jason Corso. “*AdaBoost for Face Detection*”, 2014.
- [5] Adrian Rosebrock, “www.pyimagesearch.com,” [Online]. Available: www.pyimagesearch.com/2018/06/25/raspberry-pi-face-recognition.