**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**



**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC SINH VIÊN**

***ĐỀ TÀI*: NGHIÊN CỨU THIẾT KẾ CHẾ TẠO KHÓA CỬA TỰ ĐỘNG ỨNG DỤNG CÔNG NGHỆ XỬ LÝ ẢNH KHUÔN MẶT**

Thuộc nhóm ngành : Kỹ thuật điều khiển và tự động hóa

Sinh viên thực hiện : Dương Xuân Ngọc Tự Động Hóa 3-K62

Lê Anh Dũng Tự Động Hóa 3-K62

Nguyễn Trọng Bảo An Tự Động Hóa 3-K62

Phan Văn Long Tự Động Hóa 3-K62

Cao Minh Khánh Tự Động Hóa 3-K62

Giảng viên hướng dẫn : TS. Phí Văn Lâm

**HÀ NỘI, 5/2024**

**MỤC LỤC**

**MỤC LỤC HÌNH ẢNH**

[Hình 2.1: Khóa điện 12V 9](#_Toc165585229)

[Hình 2.2: Máy tính nhúng Raspberry pi 3 model B+ và màn hình cảm ứng LCD3.5inch 9](#_Toc165585230)

[Hình 2.3: Module relay 5v 10A 10](#_Toc165585231)

[Hình 2.4: Mica Trung Quốc 5mm 10](#_Toc165585232)

[Hình 2.5: Wecam HD 11](#_Toc165585233)

[Hình 2.6: Logo phần mềm Visual studio code 12](#_Toc165585234)

[Hình 2.7: Logo phần mềm Pycharm professional 2024.1 12](#_Toc165585235)

[Hình 2.8: Logo phần mềm Raspberry Pi Imager 12](#_Toc165585236)

[Hình 2.9: Logo phần mềm PuTTy 13](#_Toc165585237)

[Hình 2.9: Logo phần mềm RealVNC Viewer 13](#_Toc165585238)

[Hình 3.1: Resize ảnh, tạo ra các bản copy với nhiều kích thước 25](#_Toc165585239)

[Hình 3.2: Mạng P-net 25](#_Toc165585240)

[Hình 3.3: phép NMS, giữ lại box có confident (độ tin cậy) cao nhất 26](#_Toc165585241)

[Hình 3.4: R-Net 27](#_Toc165585242)

[Hình 3.5: O-Net 27](#_Toc165585243)

[Hình 3.6: Hình ảnh sau khi thông qua 3 mạng lưới xử lý 28](#_Toc165585244)

[Hình 3.7: Mạng inception V1 29](#_Toc165585245)

[Hình 3.8: Các bước training của facenet 30](#_Toc165585246)

[Hình 4.1: Giao diện người dùng 33](#_Toc165585247)

[Hình 4.2: Thông báo đăng nhập thành công 34](#_Toc165585248)

[Hình 4.3: Thông báo đăng nhập thất bại 34](#_Toc165585249)

[Hình 4.4: Cần đăng nhập tài khoản đã được cấp trước đó 35](#_Toc165585250)

[Hình 4.5: Nhập thông tin mới 35](#_Toc165585251)

[Hình 4.6: Bắt đầu quá trình thu thập dữ liệu khuôn mặt 36](#_Toc165585252)

[Hình 3.9: Facenet phân tích đặc điểm khuôn mặt con người 40](#_Toc165585253)

[Hình 3.10: Biểu diễn dữ liệu trong hệ tọa độ descartes 41](#_Toc165585254)

[Hình 3.11: Tạo vector ngẫu nhiên từ bộ dữ liệu 41](#_Toc165585255)

[Hình 3.12: Các dữ liệu được phân tán ngẫu nhiên 42](#_Toc165585256)

[Hình 3.13: Chọn ngẫu nhiên 1 ảnh làm Anchor 43](#_Toc165585257)

[Hình 3.14: So sánh với ví dụ tích cực (positive example) 43](#_Toc165585258)

[Hình 3.15: So sánh với ví dụ tiêu cực (negative example) 44](#_Toc165585259)

[Hình 3.16: Điều chỉnh thông số của dữ liệu 44](#_Toc165585260)

[Hình 3.17: Nhóm các dữ liệu với nhau 45](#_Toc165585261)

[Hình 3.18: Ví dụ tích cực gần với anchor hơn và ngược lại 45](#_Toc165585262)

[Hình 3.20: Phân tích khuôn mặt dưới dạng vector 46](#_Toc165585263)

**KÍ HIỆU VÀ VIẾT TẮT**

**LỜI NÓI ĐẦU**

Những năm gần đây, công nghệ sinh trắc học tiếp tục phát triển nhanh chóng, mang lại sự tiện lợi và an toàn hơn cho cuộc sống. Bằng cách sử dụng các đặc điểm đặc trưng của một cá nhân, hầu như không thể sao chép, ngành công nghệ này dần chứng tỏ là sự lựa chọn tối ưu hơn so với các phương pháp bảo mật truyền thống.

Mặc dù đã tồn tại trong hàng thập kỷ, chủ yếu thông qua dấu vân tay, mẫu sinh học (ADN, máu, nước bọt, ...) và thậm chí cả quét mống mắt, nhưng cho đến khi công nghệ nhận dạng khuôn mặt ra đời mới đánh dấu bước đi vượt bậc của nền công nghệ này.

Hiện nay, rất nhiều ứng dụng và thiết bị đã được trang bị công nghệ nhận diện khuôn mặt để đáp ứng nhu cầu tiện lợi và tính bảo mật. Nắm bắt được su thế này và tình trạng thực tế hàng ngày nhiều người bị rơi mất chìa khóa cửa, thẻ từ (thẻ mở cửa) hoặc có thể quên mật khẩu cửa nhà. Nhóm tôi quyết định đưa ra 1 sản phẩm khóa cửa tự động sử dụng công nghệ xử lý ảnh khuôn mặt. Loại khóa này sẽ giải quyết những lỗi lo như quên hoặc mất chìa khóa, quên mật khẩu, …hơn nữa lại có tính bảo mật rất cao. Nhóm tôi hướng đến 1 sản phẩm đáp ứng được nhu cầu của số đông, ứng dụng công nghệ cao nhưng lại có giá thành hợp lý để nhiều đối tượng khách hàng có thể sử dụng.

Mục tiêu của nghiên cứu này là tìm hiểu và đánh giá hiệu suất của cửa thông minh nhận diện khuôn mặt, tập trung vào các khía cạnh bảo mật, tiện ích và hiệu suất. Đầu tiên, nghiên cứu sẽ đánh giá khả năng của hệ thống nhận diện khuôn mặt trong việc bảo vệ an ninh của ngôi nhà hoặc doanh nghiệp, bằng cách đo lường độ chính xác của việc nhận dạng khuôn mặt và khả năng chống lại các hình thức tấn công giả mạo. Thứ hai, nghiên cứu sẽ đánh giá mức độ tiện ích và thuận tiện của cửa thông minh nhận diện khuôn mặt đối với người dùng, bao gồm thời gian phản hồi, sự dễ dàng trong việc sử dụng và khả năng tích hợp với các hệ thống khác như hệ thống nhà thông minh. Cuối cùng, nghiên cứu sẽ thử nghiệm và đánh giá hiệu suất của hệ thống nhận diện khuôn mặt trong các điều kiện ánh sáng khác nhau, góc nhìn khác nhau và đối với các nhóm người dùng khác nhau. Với triển vọng phát triển trong tương lai, các hệ thống cửa thông minh nhận diện khuôn mặt có tiềm năng để trở thành một phần không thể thiếu của môi trường sống và làm việc của chúng ta. Công nghệ này có thể tiếp tục phát triển để tích hợp các tính năng thông minh như giao tiếp giữa các thiết bị, quản lý người dùng và truy cập từ xa, cũng như tối ưu hóa hiệu suất và bảo mật. Hy vọng rằng nghiên cứu này sẽ cung cấp thông tin quý báu để cải thiện và phát triển các giải pháp cửa thông minh nhận diện khuôn mặt, từ đó tạo ra một môi trường sống và làm việc an toàn, thông minh và tiện nghi hơn cho mọi người.

# Chương 1: Tổng Quan

* 1. **Tổng quan đề tài nghiên cứu.**

Hiện nay, bài toán nhận dạng khuôn mặt tập trung sự quan tâm của nhiều nhà khoa học bởi đây là một trong những thông tin đặc thù và riêng biệt của con người và có thể khai thác vào nhiều ứng dụng khác nhau như tương tác người máy (Human Machine-Interaction: HCI) , định danh , định danh lại , điều khiển và tự động hóa , bảo mật , giám sát …. Tuy nhiên, bài toán này cũng còn đang phải đối mặt với một số thách thức khi triển khai hệ thống thực, tính bảo mật hệ thống, độ chính xác tùy ứng dụng,… Bài toán nhận dạng khuôn mặt có thể chia thành các bước như: Phát hiện khuôn mặt, biểu diễn khuôn và nhận dạng. Ở bước đầu tiên là phát hiện khuôn mặt đã có nhiều phương pháp được đề xuất như Haarlike Cascade , Dlib hay phương pháp sử dụng các mạng học sâu Yolo. trong nghiên cứu này chúng tôi sử dụng YoloV8n. Trong khâu biểu diễn khuôn mặt, chúng tôi sẽ sử dụng các giải pháp biểu diễn điểm đặc trưng có đề xuất kết hợp giải pháp chống giả mạo dựa trên một số khung hình liên tiếp nhau. Khâu nhận dạng khuôn mặt chúng tôi sử dụng FaceNet để nhận dạng và đánh giá sự hiệu quả của phương pháp nhận diện. Đề tài này dựa trên deep learning (học sâu) để nhận diện khuôn mặt đã được đăng ký trước.

* 1. **Tóm tắt hoạt động nghiên cứu.**

Thiết Lập Môi Trường Thử Nghiệm: Tạo một môi trường thử nghiệm chính xác và đại diện cho các tình huống thực tế mà cửa thông minh sẽ được triển khai bao gồm các điều kiện ánh sáng khác nhau, góc nhìn khác nhau và các biến thể về mặt vật lý của môi trường. Thu Thập Dữ Liệu Mẫu: Sử dụng camera hoặc thiết bị ghi hình để thu thập dữ liệu mẫu của các người dùng tiềm năng. Đảm bảo rằng dữ liệu mẫu đủ đa dạng để đại diện cho các điều kiện khác nhau mà hệ thống có thể gặp phải trong thực tế. Bảo Mật và Sự Đồng Ý: Đảm bảo tuân thủ các quy định về bảo mật và quyền riêng tư. Trước khi thu thập dữ liệu, hãy đảm bảo rằng người tham gia đã đồng ý tham gia vào việc thu thập dữ liệu và hiểu rõ về mục đích của nghiên cứu. Xử Lý và Lưu Trữ Dữ Liệu: Sau khi thu thập, dữ liệu mẫu cần được xử lý và lưu trữ một cách an toàn và có tổ chức. Điều này đảm bảo rằng dữ liệu có sẵn để phân tích và đánh giá hiệu suất của hệ thống nhận diện khuôn mặt.

# Chương 2: Vật tư và phần mềm sử dụng

### Vật tư phần cứng

* Khóa điện 12V

A metal door lock with a wire

Description automatically generated

Hình 2.1: Khóa điện 12V

* Máy tính nhúng Raspberry pi 3 model B+ và màn hình cảm ứng LCD3.5inchA small electronic device with a screen

  Description automatically generated

Hình 2.2: Máy tính nhúng Raspberry pi 3 model B+ và màn hình cảm ứng LCD3.5inch

* Module relay 5v 10A

A blue and black electronic device

Description automatically generated

Hình 2.3: Module relay 5v 10A

* Tấm mica 5mm

A clear box with a lock

Description automatically generated

Hình 2.4: Mica Trung Quốc 5mm

* Webcam HD



Hình 2.5: Wecam HD

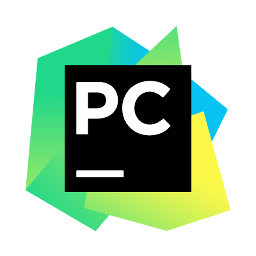
**2.2 Phần mềm sử dụng**

* Visual studio code



Hình 2.6: Logo phần mềm Visual studio code

* Pycharm professional 2024.1



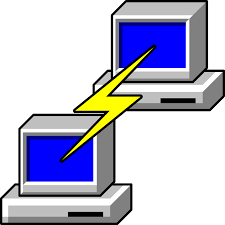
Hình 2.7: Logo phần mềm Pycharm professional 2024.1

* Raspberry Pi Imager



Hình 2.8: Logo phần mềm Raspberry Pi Imager

* PuTTy



Hình 2.9: Logo phần mềm PuTTy

* RealVNC Viewer



Hình 2.9: Logo phần mềm RealVNC Viewer

# Chương 3: Cơ sở lý thuyết

**3.1 Machine learning, Deep learning**

**3.1.1 Machine learning**

* Khái niệm:
* Machine learning (ML) hay học máy là một nhánh của trí tuệ nhân tạo (AI), nó là một lĩnh vực nghiên cứu cho phép máy tính có khả năng cải thiện chính bản than chúng dựa trên dữ liệu mẫu (training data) hoặc dựa vào kinh nghiệm (những gì đã được học). ML có thể tự dự đoán hoặc đưa ra quyết định mà không cần được lập trình cụ thể.
* Bài toán ML thường được chia làm 2 loại là dự đoán (prediction) và phân loại (classification)
* Phân loại ML:

ML có khá nhiều cách phân loại, thông thường sẽ được phân làm các loại sau:

* Supervised learning (học có giám sát): sử dụng tập dữ liệu đã được người dung gắn nhãn trước và phân loại để cho phép thuật toán xem hiệu suất của nó đến mức nào
* Unsupervised learning (học không giám sát): sử dụng tập dữ liệu thô không được gắn nhãn và thuật toán xác định các mẫu và mối quan hệ trong dữ liệu mà không cần sự trợ giúp từ người dung
* Semi-supervised learning (học bán giám sát): tập dữ liệu chứa dữ liệu có cấu trúc và không cấu trúc, hướng dẫn thuật toán đưa ra kết quả độc lập. Sự kết hợp của hai loại dữ liệu trong một tập dữ liệu huấn luyện cho phép các thuật toán ML học cách gắn nhãn dữ liệu chưa được gắn nhãn
* Reinforce learning (học củng cố/tăng cường): tập dữ liệu sử dụng hệ thống “thưởng/phạt”, cung cấp phản hồi cho thuật toán để học hỏi từ kinh nghiệm của chính nó bằng phương pháp thử và sai
* Deep learning (DL): học sâu về 1 vấn đề chuyên biệt nào đó
* Phương thức hoạt động:
* ML Workflow: quy trình làm việc của ML gồm các bước
* Data collection – thu thập dữ liệu: để máy tính có thể học được, cần có một bộ dữ liệu (dataset), người lập trình có thể tự thu thập chúng hoặc lấy các bộ dữ liệu có sẵn đã được công bố trước đó. Lưu ý cần thu thập dữ liệu từ nguồn chính thống để đảm bảo độ chính xác của dữ liệu và máy có thể học một cách đúng đắn và hiệu quả
* Preprocessing – tiền xử lý: bước này chuẩn hóa dữ liệu, loại bỏ các thuộc tính không cần thiết, gán nhãn dữ liệu, mã háo một số đặc trưng, rút gọn dữ liệu nhưng vẫn đảm bảo kết quả… Bước này tốn thời gian tỷ lệ thuận với số lượng dữ liệu hiện có
* Training model – huấn luyện mô hình: huấn luyện cho mô hình hay chính là cho nó học trên bộ dữ liệu đã được thu thập và xử lý ở 2 bước trên
* Evaluating model – đánh giá mô hình: sau khi đã huấn luyện xong, ta cần dùng các độ đo để đánh giá mô hình, tùy vào từng độ đo khác nhau mà mô hình cũng được đánh giá tốt hay không khác nhau. Độ chính xác của mô hình đạt trên 85% thường được coi là tốt
* Improve – cải thiện: sau khi đánh giá mô hình, các mô hình hoạt động không tốt thì cần được training lại, liên tục cho đến khi đạt độ chính xác như kỳ vọng
* Các thuật toán phổ biến:
* Neural networks: mô phỏng cách thức hoạt động của bộ não con người, với một số lượng khổng lồ các nút xử lý được liên kết với nhau. Neurals networks là thuật toán được dung trong việc nhận dạng các mẫu và đóng via trò quan trọng trong các ứng dụng bao gồm dịch ngôn ngữ tự nhiên, nhận dạng hình ảnh, giọng nói và tạo hình ảnh
* Linear regression: được sử dụng để dự đoán các giá trị số, dựa trên mối quan hệ tuyến tính giữa các giá trị khác nhau
* Logistic regression: giúp đưa ra những dự đoán cho các biến phản hồi phân loại, chẳng hản như câu trả lời “có/không” cho các câu hỏi, nó có thể được sử dụng cho các ứng dụng như phân loại thư rác và kiểm soát chất lượng trên dây chuyền sản xuất
* Clustering: các thuật toán phân cụm có thể xác định các mẫu trong dữ liệu để nó có thể được nhóm lại. Máy tính có thể giúp các nhà khoa học dữ liệu bằng cách xác định sự khác biệt giữa các mục dữ liệu mà con người đã bỏ qua
* Decision trees: được sử dụng dể dự đoán giá trị số (hồi quy) và phân loại dữ liệu. Decision trees sử dụng một chuỗi phân nhanh của các quyết định được liên kết có thể được biểu diễn bằng một sơ đồ cây. Một trong những ưu điểm của thuật toán này là chúng dễ xác thực và kiểm tra, không như Neural networks
* Random forests: dự đoán một giá trị hoặc danh sách mục bằng cách kết hợp các kiểu quả từ một số cây quyết định
* Một số khái niệm cơ bản:
* Dataset: là tập dữ liệu dạng nguyên thủy chưa qua xử lý mà bạn đã thu thập được ở bước data collection. Một dataset gồm nhiều data point
* Data point: là điểm dữ liệu, mỗi điểm dữ liệu biểu diễn cho một quan sát. Mỗi data point có nhiều đặc trưng hay thuộc tính khác nhau, được chia làm hai loại: dữ liệu số (numerical) và dữ liệu không phải số (ví dụ như chuỗi) (non-numerical/categorical). Data point được biểu diễn thành dòng tương ứng, mỗi dòng có thể có 1 hoặc nhiều dữ liệu (chính là các đặc trưng)
* Training data và test data: dataset thường sẽ được chia làm 2 tập này, training data dùng để huấn luyện cho mô hình, test data dùng để dự đoán kết quả và đánh giá mô hình. Có bài toán người ta sẽ cho sẵn hai tập này thì bạn không cần phải chia nữa, đối với bài toán chỉ cho mỗi dataset thôi thì phải chia ra. Thường tỷ lệ giữa tập train và test sẽ là 8/2.
* Features vector: là vector đặc trưng, mỗi vector này sẽ biểu diễn cho một điểm dữ liệu trong dataset. Mỗi vector có n chiều biểu diễn các đặc trưng của điểm dữ liệu, mỗi đặc trưng là một chiều và phải là dữ liệu số. Các mô hình chỉ có thể huấn luyện được từ các vector đặc trưng này, do đó dataset cần phải chuyển về dạng một tập các vector đặc trưng (features vectors).
* Model: là các mô hình được dùng để training trên một training data theo thuật toán của mô hình đó. Sau đó mô hình có thể dự đoán hoặc đưa ra các quyết định dựa trên những gì chúng đã được học.
* Ứng dụng:
* Speech regconition: nhận dạng giọng nói tự động (ASR), nhận dạng giọng nói máy tính hoặc chuyển giọng nói thành văn bản
* Customer service: chatbots trực tuyến thay thế các tác nhân con người trong hành trình của khách hàng, thay đổi cách chúng ta nghĩ về sự tương tác của khách hàng trên các nền tảng xã hội
* Computer vision: cho phép máy tính lấy thông tin có ý nghĩa từ video, hình ảnh kĩ thuật số và các đầu vào trực quan khác, sau đó thực thi hành động thích hợp
* Recommendation engines: sử dụng dữ liệu hành vi tiêu dùng trong quá khứ, các thuật toán **AI learning** có thể giúp khám phá các xu hướng dữ liệu có thể được sử dụng để phát triển các chiến lược cross-sell hiệu quả hơn
* Automated stock trading: Được thiết kế để tối ưu hóa danh mục đầu tư chứng khoán, các nền tảng giao dịch tần suất cao do AI điều khiển để hàng triệu giao dịch mỗi ngày mà không cần đến sự can thiệp của con người
* Fraud detection: Các ngân hàng và các tổ chức tài chính có thể sử dụng máy học để phát hiện các giao dịch đáng ngờ

**3.1.2 Deep learning**

* Khái niệm:
* Mô hình deep learning - học sâu là một nhánh nhỏ của AI - artificial intelligence (trí tuệ nhân tạo). Nó hoạt động dựa trên mạng nơ-ron nhân tạo (neural networks) để phân tích, xử lý dữ liệu và mô phỏng bộ não của con người, về cơ bản là một mạng nơ-ron có ba lớp trở lên. Những mạng lưới thần kinh này cố gắng mô phỏng hành vi của não người cho phép deep learning “học” từ một lượng lớn dữ liệu. Trong khi mạng nơ-ron có một lớp vẫn có thể đưa ra các dự đoán gần đúng, các lớp ẩn bổ sung có thể giúp tối ưu hóa và tinh chỉnh để có độ chính xác.
* Mạng lưới thần kinh sâu (DNN - Deep neural networks) bao gồm nhiều lớp tế bào thần kinh khác nhau, có khả năng thực hiện các tính toán rất phức tạp. Deep Learning hiện đang phát triển rất nhanh và được coi là một trong những bước đột phá lớn nhất trong Machine Learning.
* DL thúc đẩy nhiều ứng dụng và dịch vụ trí tuệ nhân tạo (AI) nhằm cải thiện tự động hóa, thực hiện các tác vụ phân tích và vật lý mà không cần sự can thiệp của con người.
* Cách thức hoạt động:

DL hoạt động bằng cách khám phá các cấu trúc phức tạp trong dữ liệu mà chúng trải nghiệm. Cụ thể là xây dựng các mô hình tính toán bao gồm nhiều lớp xử lý, mạng có thể tạo ra nhiều mức trừu tượng để biểu diễn dữ liệu. Mạng thần kinh nhân tạo trong Deep Learning được xây dựng để mô phỏng khả năng tư duy của bộ não con người.

* Một mạng nơ-ron bao gồm nhiều lớp (layer) khác nhau, càng nhiều lớp thì mạng sẽ càng “sâu”. Trong mỗi lớp là các node (nút mạng) và được liên kết với các lớp khác bên cạnh.
* Mỗi kết nối giữa các nút sẽ có một trọng số tương ứng, trọng số càng cao thì mức độ ảnh hưởng của kết nối này đến mạng nơ-ron càng lớn.
* Mỗi nơ-ron sẽ có một chức năng kích hoạt, về cơ bản chịu trách nhiệm "chuẩn hóa" đầu ra từ nơ-ron này.
* Dữ liệu được người dùng nhập vào mạng thần kinh đi qua tất cả các lớp và trả về kết quả ở layer cuối cùng, được gọi là lớp đầu ra (output layer).
* Trong quá trình đào tạo mô hình mạng nơ-ron, các trọng số sẽ được thay đổi và nhiệm vụ của mô hình là tìm tập giá trị của các trọng số sao cho phán đoán đúng nhất.
* Đối với nhiều tác vụ, chẳng hạn như thị giác máy tính, nhận dạng giọng nói còn được gọi là xử lý ngôn ngữ tự nhiên, dịch máy và robot, hiệu suất của các hệ thống deep learning vượt xa so với các hệ thống machine learning.
* Các hệ thống Deep Learning yêu cầu phần cứng rất mạnh để có thể xử lý lượng lớn dữ liệu và thực hiện các phép tính phức tạp. Nhiều mô hình Deep Learning có thể mất hàng tuần hoặc thậm chí hàng tháng để triển khai trên phần cứng tiên tiến nhất hiện nay.
* Ưu nhược điểm:

Một số ưu điểm nổi bật của Deep Learning bao gồm:

* Cấu trúc neural networks linh hoạt, dễ dàng thay đổi để phù hợp với nhiều thuật toán khác nhau.
* Có khả năng giải nhiều vấn đề phức tạp với độ chính xác rất cao.
* **Tự động hóa các tính năng**
* Có khả năng thực hiện tính toán song song, hiệu năng tốt, xử lý lượng dữ liệu lớn.
* **Tương thích tốt với kiểu dữ liệu phi cấu trúc**

Bên cạnh ưu điểm, Deep Learning vẫn còn tồn tại nhiều hạn chế:

* Cần khối lượng dữ liệu khổng lồ để khai thác tối đa khả năng của Deep Learning.
* Chi phí tính toán cao vì phải xử lý nhiều mô hình phức tạp.
* Không có nền tảng lý thuyết vững chắc để chọn các công cụ tối ưu cho Deep Learning.
* Các yếu tố quyết định việc sử dụng:
* Mục tiêu và độ phức tạp của dự án: Lợi thế của deep learning là giải quyết các vấn đề phức tạp bằng cách đưa ra các phân tích trong mối quan hệ ẩn trong dữ liệu. Đặc biệt mô hình này phù hợp khi áp dụng vào việc xử lý dữ liệu ở nhiều dạng khác như ngôn ngữ, hình ảnh, nhận diện giọng nói, v.v.
* Tài nguyên: Một khối lượng lớn dữ liệu của doanh nghiệp cần được xử lý thông qua mô hình deep learning sẽ dễ dàng hơn. Tuy nhiên, quá trình xử lý vô cùng phức tạp và tốn kém do đó tùy vào khối lượng dữ liệu mà doanh nghiệp sẽ đưa ra quyết định nên lựa chọn mô hình deep learning hay machine learning.
* Số lượng lớn dữ liệu: Mô hình deep learning chỉ ra các mối quan hệ ẩn sâu bên trong bộ dữ liệu. Tuy nhiên, điều này cũng đồng nghĩa với việc dữ liệu đầu vào phải lớn hơn nhiều so với thuật toán của machine learning. Do đó, đối với lượng dữ liệu lớn việc sử dụng deep learning rất phù hợp.
* Ứng dụng của DL:
* Người máy: Sự phát triển của khoa học công nghệ trong thời gian gần đây, đặc biệt trong lĩnh vực người máy được thúc đẩy bởi những tiến bộ trong AI và deep learning.
* Nông nghiệp: Ngày nay, deep learning cho phép nông dân triển khai thiết bị có thể nhìn thấy và phân biệt giữa cây trồng và cỏ dại. Khả năng này cho phép máy làm cỏ phun thuốc diệt cỏ một cách có chọn lọc lên cỏ dại và để các cây khác không bị ảnh hưởng.
* Hình ảnh y tế và chăm sóc sức khỏe: Học sâu đang có những bước tiến đáng kể vào việc cải thiện chất lượng, chăm sóc sức khỏe bằng cách dự đoán các sự kiện y tế từ dữ liệu hồ sơ sức khỏe điện tử.

**3.2 Công nghệ nhận diện khuôn mặt**

Trình phân tích khuôn mặt là phần mềm xác định hoặc xác nhận danh tính của một người qua khuôn mặt của họ. Công nghệ này hoạt động bằng cách xác định và đo lường các đặc điểm khuôn mặt trong hình ảnh. Công nghệ nhận dạng khuôn mặt có thể xác định khuôn mặt người trong hình ảnh hoặc video, xác định xem khuôn mặt xuất hiện trong hai hình ảnh có phải là cùng một người không hoặc tìm kiếm khuôn mặt trong một bộ sưu tập đồ sộ các hình ảnh hiện có. Các hệ thống bảo mật sinh trắc học sử dụng công nghệ nhận dạng khuôn mặt để nhận dạng cá nhân độc nhất trong lúc triển khai người dùng hoặc đăng nhập, cũng như để tăng cường cho hoạt động xác thực người dùng. Các thiết bị di động và cá nhân cũng thường sử dụng công nghệ phân tích khuôn mặt để bảo mật thiết bị.

* Cách thức hoạt động:

Nhận diện khuôn mặt hoạt động theo ba bước: phát hiện, phân tích và nhận diện.

* Phát hiện: là quá trình tìm khuôn mặt trong hình ảnh. Nhận diện khuôn mặt được thị giác máy tính hỗ trợ có thể phát hiện và xác định khuôn mặt của cá nhân từ hình ảnh có một hoặc nhiều khuôn mặt người. Công nghệ này có thể phát hiện dữ liệu khuôn mặt ở cả nét mặt nhìn từ phía trước và từ phía bên.
* Thị giác máy tính:

Thiết bị máy sử dụng thị giác máy tính để xác định người, địa điểm và mọi thứ trong hình ảnh với độ chính xác bằng hoặc cao hơn mức của con người và với hiệu quả và tốc độ cao hơn nhiều. Nhờ sử dụng công nghệ trí tuệ nhân tạo (AI) phức tạp, thị giác máy tính tự động hóa hoạt động trích xuất, phân tích, phân loại và nắm bắt thông tin hữu ích từ dữ liệu hình ảnh. Dữ liệu hình ảnh tồn tại ở nhiều dạng, ví dụ:

* + Hình ảnh đơn lẻ
  + Cảnh video
  + Góc nhìn từ nhiều camera
  + Dữ liệu ba chiều
* Phân tích:

Sau hệ thống nhận diện khuôn mặt phân tích hình ảnh khuôn mặt. Hệ thống này ánh xạ và đọc hình dạng khuôn mặt cũng như biểu cảm trên khuôn mặt. Hệ thống xác định các điểm quan trọng trên khuôn mặt, những điểm này đóng vai trò là chìa khóa để phân biệt khuôn mặt với những vật thể khác. Công nghệ nhận diện khuôn mặt thường tìm kiếm những đặc điểm sau:

* + Khoảng cách giữa hai mắt
  + Khoảng cách từ trán đến cằm
  + Khoảng cách giữa mũi và miệng
  + Độ sâu của hốc mắt
  + Hình dạng của gò má
  + Đường viền môi, tai và cằm

Sau đó, hệ thống sẽ chuyển đổi dữ liệu nhận dạng khuôn mặt thành một dãy số hoặc điểm được gọi là dấu khuôn mặt (faceprint). Mỗi người sẽ có một dấu khuôn mặt độc nhất, tương tự như dấu vân tay. Thông tin được nhận dạng khuôn mặt sử dụng cũng có thể được sử dụng ngược lại để tái cấu trúc khuôn mặt của một người thành phiên bản kỹ thuật số.

* Nhận diện

Nhận diện khuôn mặt có thể xác định một người bằng cách so sánh các khuôn mặt trong hai hoặc nhiều hình ảnh và đánh giá khả năng khớp khuôn mặt. Ví dụ: nhận diện khuôn mặt có thể xác minh khuôn mặt có trong ảnh tự chụp bằng camera của di động khớp với khuôn mặt trong hình ảnh giấy tờ tùy thân do chính phủ cấp như giấy phép lái xe hoặc hộ chiếu, cũng như xác minh khuôn mặt có trong ảnh tự chụp không khớp với khuôn mặt trong tập hợp các khuôn mặt được chụp trước đó.

* Ứng dụng:
* Phát hiện gian lận
* An ninh mạng và bảo mật
* Kiểm soát sân bay và biên giới
* Thực hiện các giao dịch trực tuyến
* Chăm sóc sức khỏe

**3.3 OpenCV**

Project OpenCV được bắt đầu từ Intel năm 1999 bởi Gary Bradsky. OpenCV viết tắt cho Open Source Computer Vision Library. OpenCV là một thư viện mã nguồn mở lớn cho thị giác máy tính, học máy và xử lý hình ảnh, và hiện có thêm tính năng tăng tốc GPU cho các hoạt động theo real-time.

Hiện nay, nó đóng một vai trò quan trọng trong việc hoạt động thời gian thực, điều này rất quan trọng trong các hệ thống hiện nay. Bằng cách sử dụng nó, người ta có thể xử lý hình ảnh và video để nhận diện đối tượng, khuôn mặt, hoặc thậm chí là chữ viết tay của con người

OpenCV được phát hành theo giấy phép BSD, do đó nó miễn phí cho cả học tập và sử dụng với mục đích thương mại. Nó có trên các giao diện C++, C, Python và Java và hỗ trợ Windows, Linux, Mac OS, iOS và Android. OpenCV được thiết kế để hỗ trợ hiệu quả về tính toán và chuyên dùng cho các ứng dụng real-time (thời gian thực). Nếu được viết trên C/C++ tối ưu, thư viện này có thể tận dụng được bộ xử lý đa lõi (multi-core processing).

OpenCV cho phép bạn thực hiện các hoạt động khác nhau trên hình ảnh.

* Đọc hình ảnh: OpenCV giúp bạn đọc hình ảnh từ tệp hoặc trực tiếp từ máy ảnh để làm cho nó truy cập được để tiếp tục xử lý.
* Tăng cường hình ảnh: Bạn sẽ có thể tăng cường hình ảnh bằng cách điều chỉnh độ sáng, độ nét hoặc độ tương phản của hình ảnh. Điều này hữu ích để trực quan hóa chất lượng của hình ảnh.
* Phát hiện đối tượng: Như bạn có thể thấy trong hình ảnh dưới đây, đối tượng cũng có thể được phát hiện bằng cách sử dụng OpenCV, vòng đeo cổ, đồng hồ, mẫu, khuôn mặt có thể được phát hiện. Điều này cũng có thể bao gồm nhận diện khuôn mặt, hình dạng hoặc thậm chí là các đối tượng khác.
* Lọc hình ảnh: Bạn có thể thay đổi hình ảnh bằng cách áp dụng các bộ lọc khác nhau như làm mờ hoặc làm sắc nét.
* Vẽ hình ảnh: OpenCV cho phép vẽ văn bản, đường và bất kỳ hình dạng nào trên hình ảnh.
* Lưu hình ảnh đã thay đổi: Sau khi xử lý, bạn có thể lưu hình ảnh đã được chỉnh sửa để phân tích trong tương lai.
* Tính năng của OpenCV:
* Đầu vào/đầu ra hình ảnh/video, xử lý, hiển thị (core, imgproc, highgui): Bao gồm các hoạt động cơ bản như đọc và ghi hình ảnh/video, xử lý chúng (ví dụ: thay đổi kích thước, cắt, lọc), và hiển thị chúng. Chức năng cơ bản xử lý cấu trúc dữ liệu và thuật toán cơ bản, imgproc tập trung vào các hoạt động xử lý hình ảnh, và highgui xử lý các nhiệm vụ liên quan đến giao diện người dùng đồ họa.
* Phát hiện đối tượng/đặc điểm (objdetect, features2d, nonfree): Các module này giúp phát hiện đối tượng hoặc đặc điểm trong hình ảnh. objdetect chủ yếu được sử dụng cho phát hiện đối tượng, features2d xử lý việc phát hiện và mô tả đặc điểm, và nonfree bao gồm các thuật toán cho việc phát hiện đặc điểm không phải mã nguồn mở.
* Thị giác máy tính dựa trên hình ảnh mắt đơn hoặc kép dựa trên hình ảnh (calib3d, stitching, videostab): Bộ các module này xử lý các khía cạnh hình học của thị giác máy tính. calib3d được sử dụng cho việc hiệu chỉnh camera và tái tạo 3D, stitching tập trung vào việc nối ảnh (ví dụ: tạo ảnh toàn cảnh), và videostab cung cấp các công cụ cho ổn định hình ảnh video.
* Nhiếp ảnh tính toán (photo, video, superres): Các module này được dành riêng cho các kỹ thuật xử lý hình ảnh tiên tiến được sử dụng trong nhiếp ảnh tính toán. Chúng bao gồm các chức năng để cải thiện hình ảnh, xử lý video, và thực hiện siêu phân giải.
* Học máy & phân cụm (ML, flann): OpenCV cung cấp hỗ trợ cho các nhiệm vụ học máy và phân cụm. Module ML cung cấp các thuật toán học máy khác nhau cho các nhiệm vụ phân loại, hồi quy, và phân cụm, trong khi flann cung cấp các thuật toán để tìm kiếm gần nhất xấp xỉ nhanh chóng.
* Tăng tốc CUDA (gpu): Module này cho phép tăng tốc GPU của một số hoạt động OpenCV thông qua CUDA, nền tảng tính toán song song và mô hình lập trình của NVIDIA.

Các mô-đun này bao gồm một loạt các chức năng, cho phép các nhà phát triển triển khai các nhiệm vụ thị giác máy tính phức tạp một cách hiệu quả bằng cách sử dụng OpenCV.

* Ứng dụng:

Có rất nhiều ứng dụng được giải quyết bằng OpenCV, dưới đây là một số trong số đó:

* Nhận diện khuôn mặt
* Kiểm tra và giám sát tự động
* Đếm số lượng người (lưu lượng đi lại trong trung tâm mua sắm, v.v.)
* Đếm số lượng phương tiện trên cao tốc cùng với tốc độ của chúng
* Các cài đặt nghệ thuật tương tác
* Phát hiện bất thường (lỗi) trong quá trình sản xuất (các sản phẩm lỗi lạ)
* Ghép ảnh từ góc đường
* Tìm kiếm và truy xuất video/hình ảnh
* Điều hướng và điều khiển xe robot và xe không người lái
* Nhận diện đối tượng
* Phân tích hình ảnh y tế
* Phim - Cấu trúc 3D từ chuyển động

## 3.4 MTCNN (Multi-task Cascaded Convolutional Networks) và FaceNet

## Nhận diện khuôn mặt (Face Recognition) là một trong những thách thức lớn mà các nhà nghiên cứu về Học máy - Học sâu đã và đang phải đối mặt. Bài toán này có thể được áp dụng ở rất nhiều lĩnh vực khác nhau, đặc biệt trong những lĩnh vực yêu cầu độ chính xác và bảo mật cao như eKYC trong E-Comercial và nhận diện danh tính qua surveillance camera (CCTV). Ta sẽ chia bài toán này ra thành 2 vấn đề chính: Phát hiện khuôn măt (Face Detection) và Phân biệt khuôn mặt (Face Verification). Để đi sâu hơn về quá trình hiểu và áp dụng hoàn chỉnh một bài Face Recognition, ta cần tìm hiểu và suy luận 2 mạng nổi tiếng cho 2 vấn đề trên: MTCNN (Multi-task Cascaded Convolutional Networks) và FaceNet.

## 3.4.1 MTCNN

## 3.4.1.1 Khái niệm cơ bản

## MTCNN hay Multi-Task Cascaded Convolutional Neural Networks là một mạng nơ-ron nhận dạng khuôn mặt và các điểm đặc trưng trên hình ảnh. Nó được công bố vào năm 2016 bởi Zhang và đồng nghiệp.

## MTCNN là một trong những công cụ phát hiện khuôn mặt phổ biến và chính xác nhất hiện nay. Nó bao gồm 3 mạng nơ-ron được kết nối theo một chuỗi liên tiếp tượng trưng cho 3 stage chính là P-Net, R-Net và O-Net (PRO!)

## 3.4.1.2 Ba lớp mạng đặc biệt

## Stage 1: P-Net

## Trước hết, một bức ảnh thường sẽ có nhiều hơn một người - một khuôn mặt. Ngoài ra, những khuôn mặt thường sẽ có kích thước khác nhau. Ta cần một phương thức để có thể nhận dạng toàn bộ số khuôn mặt đó, ở các kích thước khác nhau. MTCNN đưa cho chúng ta một giải pháp, bằng cách sử dụng phép Resize ảnh, để tạo một loạt các bản copy từ ảnh gốc với kích cỡ khác nhau, từ to đến nhỏ, tạo thành 1 Image Pyramid.

## A person in a suit Description automatically generated

Hình 3.1: Resize ảnh, tạo ra các bản copy với nhiều kích thước

## Với mỗi một phiên bản copy-resize của ảnh gốc, ta sử dụng kernel 12x12 pixel và stride = 2 để đi qua toàn bộ bức ảnh, dò tìm khuôn mặt. Vì các bản copies của ảnh gốc có kích thước khác nhau, cho nên mạng có thể dễ dàng nhận biết được các khuôn mặt với kích thước khác nhau, mặc dù chỉ dùng 1 kernel với kích thước cố định (Ảnh to hơn, mặt to hơn; Ảnh nhỏ hơn, mặt nhỏ hơn). Sau đó, ta sẽ đưa những kernels được cắt ra từ trên và truyền qua mạng P-Net (Proposal Network). Kết quả của mạng cho ra một loạt các bounding boxes nằm trong mỗi kernel, mỗi bounding boxes sẽ chứa tọa độ 4 góc để xác định vị trí trong kernel chứa nó (đã được normalize về khoảng từ (0,1)) và điểm confident (Điểm tự tin) tương ứng.

## A diagram of a block diagram Description automatically generated

Hình 3.2: Mạng P-net

## Để loại trừ bớt các bounding boxes trên các bức ảnh và các kernels, ta sử dụng 2 phương pháp chính là lập mức Threshold confident - nhằm xóa đi các box có mức confident thấp và sử dụng NMS (Non-Maximum Suppression) để xóa các box có tỷ lệ trùng nhau (Intersection Over Union) vượt qua 1 mức threshold tự đặt nào đó. Hình ảnh dưới đây là minh họa cho phép NMS, những box bị trùng nhau sẽ bị loại bỏ và giữ lại 1 box có mức confident cao nhất.

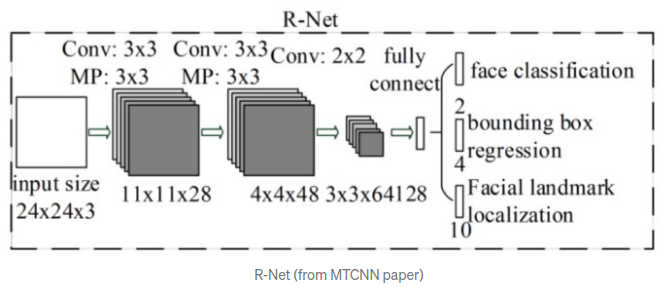
## A screenshot of a computer Description automatically generated

Hình 3.3: phép NMS, giữ lại box có confident (độ tin cậy) cao nhất

## Sau khi đã xóa bớt các box không hợp lý, ta sẽ chuyển các tọa độ của các box về với tọa độ gốc của bức ảnh thật. Do tọa độ của box đã được normalize về khoảng (0,1) tương ứng như kernel, cho nên công việc lúc này chỉ là tính toán độ dài và rộng của kernel dựa theo ảnh gốc, sau đó nhân tọa độ đã được normalize của box với kích thước của của kernel và cộng với tọa độ của các góc kernel tương ứng. Kết quả của quá trình trên sẽ là những tọa độ của box tương ứng ở trên ảnh kích thước ban đầu. Cuối cùng, ta sẽ resize lại các box về dạng hình vuông, lấy tọa độ mới của các box và feed vào mạng tiếp theo, mạng R.

## Stage 2: R-Net

## Mạng R (Refine Network) thực hiện các bước như mạng P. Tuy nhiên, mạng còn sử dụng một phương pháp tên là padding, nhằm thực hiện việc chèn thêm các zero-pixels vào các phần thiếu của bounding box nếu bounding box bị vượt quá biên của ảnh. Tất cả các bounding box lúc này sẽ được resize về kích thước 24x24, được coi như 1 kernel và feed vào mạng R. Kết quả sau cũng là những tọa độ mới của các box còn lại và được đưa vào mạng tiếp theo, mạng O.

Hình 3.4: R-Net

## Stage 3: O-Net

## Cuối cùng là mạng O (Output Network), mạng cũng thực hiện tương tự như việc trong mạng R, thay đổi kích thước thành 48x48. Tuy nhiên, kết quả đầu ra của mạng lúc này không còn chỉ là các tọa độ của các box nữa, mà trả về 3 giá trị bao gồm: 4 tọa độ của bounding box (out[0]), tọa độ 5 điểm landmark trên mặt, bao gồm 2 mắt, 1 mũi, 2 bên cánh môi (out[1]) và điểm confident của mỗi box (out[2]). Tất cả sẽ được lưu vào thành 1 dictionary với 3 keys kể trên.

## A diagram of a graph Description automatically generated

Hình 3.5: O-Net

## A person with red eyes Description automatically generated

Hình 3.6: Hình ảnh sau khi thông qua 3 mạng lưới xử lý

## 3.4.2 FaceNet ****(FaceNet: A Unified Embedding for Face Recognition and Clustering)****

## **FaceNet là một mạng nơ-ron sâu được sử dụng để trích xuất các đặc trưng từ hình ảnh của khuôn mặt của một người. Nó được công bố vào năm 2015 bởi các nhà nghiên cứu của Google, gồm cả Schroff và đồng nghiệp.**

## ****3.4.2.1 Khái niệm cơ bản****

## **Embedding Vector (vecto nhúng): Là một vector với dimension cố định (thường có chiều nhỏ hơn các Feature Vector bình thường), đã được học trong quá trình train và đại diện cho một tập các feature có trách nhiệm trong việc phân loại các đối tượng trong chiều không gian đã được biến đổi. Embedding rất hữu dụng trong việc tìm các Nearest Neighbor trong 1 Cluster cho sẵn, dựa theo khoảng cách-mối quan hệ giữa các embedding với nhau.**

## **Inception V1: Một cấu trúc mạng CNN được giới thiệu vào năm 2014 của Google, với đặc trưng là các khối Inception. Khổi này cho phép mạng được học theo cấu trúc song song, nghĩa là với 1 đầu vào có thể được đưa vào nhiều các lớp Convolution khác nhau để đưa ra các kết quả khác nhau, sau đó sẽ được Concatenate vào thành 1 output. Việc học song song này giúp mạng có thể học được nhiều chi tiết hơn, lấy được nhiều feature hơn so với mạng CNN truyền thống. Ngoài ra, mạng cũng áp dụng các khối Convolution 1x1 nhằm giảm kích thước của mạng, khiến việc train trở nên nhanh hơn.**

## A diagram of a process Description automatically generated

Hình 3.7: Mạng inception V1

## **Triplet Loss: Thay vì sử dụng các hàm loss truyền thống, khi mà ta chỉ so sánh giá trị đầu ra của mạng với ground truth thực tế của dữ liệu, Triplet loss đưa ra một công thức mới bao gồm 3 giá trị đầu vào gồm anchor (mỏ neo)** : ảnh đầu ra của mạng; positive : ảnh cùng là 1 người với anchor; negative : ảnh không cùng là 1 người với anchor

## 

## là margin (lề them) giữa cặp positive với negative, độ sai lệch cần thiết tối thiểu giữa 2 miền giá trị, chính là embedding của . Công thức trên cho ta thấy mong muốn về khoảng cách giữa 2 embeddings là và sẽ phải nhỏ hơn ít nhất giá trị so với cặp và . Việc của ta là làm cho sự chênh lệch giữa 2 phía của công thức trở nên lớn nhất có thể, hay nói cách khác phải là minium hay và phải là maximum. Và để mạng học được nhiều hơn thì điểm positive được chọn phải nằm xa nhất có thể so với anchor, còn điểm negative được chọn phải nằm gần nhất có thể so với anchor, nhằm khiến cho mạng gặp những trường hợp xấu nhất. Hàm loss tổng quát như sau:

## 

## ****3.4.2.2 Cách thức hoạt động****

## **Quá trình Training (huấn luyện):**

## **Sử dụng một tập Dataset với rất nhiều các cá thể người khác nhau, mỗi cá thể có một số lượng ảnh nhất định.**

## **Xây dựng một mạng DNN dùng để làm Feature Extractor cho Dataset trên, kết quả là 1 embedding 128-Dimensions. Trong paper có 2 đại diện mạng là Zeiler&Fergus và InceptionV1.**

## **Huấn luyện mạng DNN để kết quả embedding có khả năng nhận diện tốt, bao gồm 2 việc là sử dụng** **normalization (Khoảng cách Euclide) cho các embeddings đầu ra và tối ưu lại các parameters trong mạng màng bằng Triplet Loss.**

## **Hàm Triplet Loss sẽ sử dụng phương pháp Triplet Selection, lựa chọn các embeddings sao cho việc học diễn ra tốt nhất.**

## A grey rectangular sign with black text Description automatically generated

Hình 3.8: Các bước training của facenet

## **Quá trình Inference (suy luận):**

## **Truyền ảnh mặt cần classify vào trong mạng Feature Extractor, thu được 1 embedding.**

## **Tiến hành sử dụng hàm** **và so sánh với các embedding khác trong tập embeddings đã có. Việc classify sẽ giống như thuật toán k-NN với k = 1.**

## 3.4.2.3 Chi tiết hơn về cách thức hoạt động của FaceNet (phụ lục)

## 3.5 Máy tính nhúng Raspberry Pi 3 B+

## 3.5.1 Sự ra đời

## Raspberry Pi là các máy tính bo mạch đơn (hay còn gọi là máy tính nhúng) kích thước chỉ bẳng một thẻ tín dụng, được phát triển tại Anh bởi Raspberry Pi Foundation với mục đích ban đầu là thúc đẩy việc giảng dạy về khoa học máy tính cơ bản trong các trường học và các nước đang phát triển, được giới thiệu lần đầu tiên vào tháng 2 năm 2012 bởi Eben Upton và nhóm phát triển tại Đại học Cambridge.

## 3.5.2 Cấu trúc của Rasberry Pi 3 B+

## Raspberry Pi chạy trên các chip ARM và sử dụng hệ điều hành Linux, như Raspbian (một biến thể của Debian)

## Thông số kỹ thuật: Raspberry Pi 3 Model B+

## Vi xử lý: Broadcom BCM2837B0, quad-core A53 (ARMv8) 64-bit SoC @1.4GHz

## RAM: 1GB LPDDR2 SDRAM

## Kết nối: 2.4GHz and 5GHz IEEE 802.11 b/g/n/ac wireless LAN, Bluetooth 4.2, BLE, Gigabit Ethernet over USB 2.0 (Tối đa 300Mbps).

## Hỗ trợ: 40-pin GPIO, 4 cổng USB2.0

## Video và âm thanh: 1 cổng full-sized HDMI, Cổng MIPI DSI Display, cổng MIPI CSI Camera, cổng stereo output và composite video 4 chân.

## Multimedia: H.264, MPEG-4 decode (1080p30), H.264 encode (1080p30); OpenGL ES 1.1, 2.0 graphics

## Lưu trữ: MicroSD

## Điện áp hoạt động: 5V/2.5A DC cổng microUSB, 5V DC trên chân GPIO, Power over Ethernet (PoE) (yêu cầu thêm PoE HAT).

## 3.5.3 Ứng dụng:

Raspberry Pi có rất nhiều ứng dụng, từ việc sử dụng như một máy tính cá nhân cơ bản đến các dự án IoT (Internet of Things) phức tạp.

Một số ứng dụng phổ biến bao gồm:

* Học lập trình và giáo dục: Raspberry Pi là một công cụ tuyệt vời để học lập trình vì tính di động, giá cả phải chăng và khả năng tương thích với nhiều ngôn ngữ lập trình.
* Dự án IoT: Raspberry Pi có thể được sử dụng để xây dựng các hệ thống IoT, từ đo lường dữ liệu đến điều khiển các thiết bị thông qua internet.
* Media center: Raspberry Pi có thể biến thành một media center thông qua việc cài đặt các phần mềm như Kodi, cho phép người dùng xem phim, nghe nhạc và xem hình ảnh trên TV.
* Máy chơi game: Với việc cài đặt các phần mềm như RetroPie, người dùng có thể biến Raspberry Pi thành một hệ thống chơi game retro.
* Ứng dụng trong xử lý ảnh: nhận diện khuôn mặt, xử lý ảnh thời gian thực, xử lý ảnh trong y tế…..

# Chương 4: Phương pháp nghiên cứu

## 4.1 Chương trình nhận diện khuôn mặt

Chương trình Nhận diện Khuôn mặt

Chương trình này là một ứng dụng nhận diện khuôn mặt với giao diện đơn giản, cho phép người dùng đăng nhập hoặc đăng kí.

Giao diện: Giao diện của chương trình bao gồm một camera để hiển thị hình ảnh, cùng với hai nút chức năng: "Đăng nhập" và "Đăng kí".

A person taking a selfie

Description automatically generated

Hình 4.1: Giao diện người dùng

Chức năng Đăng nhập:

Khi người dùng nhấn nút "Đăng nhập", chương trình sẽ chụp 5 ảnh từ camera.

Sau đó, sử dụng mô hình FaceNet để nhận diện khuôn mặt trong 5 ảnh đó. Nếu tất cả 5 ảnh đều nhận diện cùng một người, chương trình tiếp tục tính toán tỷ lệ chính xác của 5 ảnh.

Nếu tỷ lệ chính xác trung bình của 5 ảnh lớn hơn 0,8, chương trình sẽ thông báo "Đăng nhập thành công", ngược lại sẽ yêu cầu người dùng đăng nhập lại.

A person wearing glasses

Description automatically generated

Hình 4.2: Thông báo đăng nhập thành công

A person taking a selfie

Description automatically generated

Hình 4.3: Thông báo đăng nhập thất bại

Chức năng Đăng kí:

Khi người dùng nhấn nút "Đăng kí", chương trình sẽ yêu cầu người dùng đăng nhập tài khoản.

A screenshot of a computer

Description automatically generated

Hình 4.4: Cần đăng nhập tài khoản đã được cấp trước đó

Sau khi đăng nhập thành công, người dùng sẽ nhập tên của mình vào hệ thống.

A screenshot of a computer

Description automatically generated

Hình 4.5: Nhập thông tin mới

Tiếp theo, chương trình sẽ bắt đầu quá trình lấy dữ liệu khuôn mặt của người đăng kí, thông qua việc chụp ảnh từ camera.

A person taking a selfie

Description automatically generated

Hình 4.6: Bắt đầu quá trình thu thập dữ liệu khuôn mặt

Sau khi lấy đủ dữ liệu, chương trình sẽ huấn luyện mô hình nhận diện khuôn mặt.

Khi quá trình huấn luyện hoàn tất, chương trình sẽ thông báo "Đăng kí thành công".

Chương trình Nhận diện Khuôn mặt này cung cấp một cách tiện lợi và an toàn để quản lý truy cập vào hệ thống, đồng thời đảm bảo tính bảo mật thông qua việc sử dụng công nghệ nhận diện khuôn mặt hiện đại.

## 4.2 Hệ thống phần cứng.

Hệ thống phần cứng được thiết kế đơn giản , bao gồm các thành phần sau:

* Khóa điện 12V:

Chiếc khóa điện 12V được sử dụng để khóa hoặc mở cửa một cách tự động khi nhận được tín hiệu từ hệ thống.

* Raspberry Pi :

Raspberry Pi cùng camera được sử dụng để xử lý tín hiệu nhận diện khuôn mặt và điều khiển relay để mở hoặc khóa cửa.

* Relay:

Relay được kết nối với Raspberry Pi để điều khiển việc cấp điện cho khóa điện 12V.

Khi nhận diện đúng từ hệ thống, Raspberry Pi sẽ gửi tín hiệu cho relay, kích hoạt relay và cấp điện cho khóa điện 12V.

* Hoạt động:

Khi hệ thống nhận diện đúng người dùng, Raspberry Pi sẽ gửi tín hiệu cho relay.

Relay sẽ kích hoạt và cấp điện cho khóa điện 12V.

Khi khóa điện nhận được nguồn điện, nó sẽ mở khóa và cho phép cửa mở.

# Chương 5: Kết quả nghiên cứu và những hạn chế tồn tại

## 5.1 Phương pháp kiểm chứng và kết quả đã đạt được

## Phương pháp kiểm chứng

## Xây dựng Dataset: Đầu tiên, chúng tôi đã thu thập một dataset lớn chứa ảnh khuôn mặt từ nhiều người khác nhau, bao gồm nhiều góc độ, ánh sáng và điều kiện khác nhau để đảm bảo tính đa dạng và đại diện.

## Tiền xử lý ảnh: Trước khi đưa vào huấn luyện, chúng tôi đã tiến hành các bước tiền xử lý như cân bằng sáng, cắt ảnh, phân loại và chuẩn hóa để tăng cường chất lượng và sự đồng nhất của dữ liệu.

## Huấn luyện mô hình: Sử dụng thuật toán FaceNet, chúng tôi đã tiến hành quá trình huấn luyện trên dataset đã chuẩn bị để tạo ra một mô hình nhận diện khuôn mặt chính xác và hiệu quả.

## Kiểm tra và đánh giá: Cuối cùng, chúng tôi đã sử dụng một tập dữ liệu kiểm tra độc lập để đánh giá hiệu suất của hệ thống, đảm bảo rằng nó hoạt động chính xác và ổn định trên các dữ liệu mới dưới mọi góc độ, điều kiện thực tế khác nhau

## Kết quả đạt được

## Dữ liệu trả về của mô hình được huấn luyện có độ hoàn thiện tốt, mô hình nhận diện khuôn mặt đạt độ chính xác 90-95%, các chức năng giao diện hoạt động đúng như đề ra

## Giao diện người dùng đơn giản, rõ rang, dễ sử dụng, hoạt động trơn tru hiệu quả

## Sau khi nhận diện khuôn mặt là chính xác, raspberry pi có thể gửi tín hiệu điện áp điều khiển tới relay trung gian và thực hiện mở khóa điện

## 5.2 Các hạn chế tồn tại trong quá trình thực hiện

## Chưa tối ưu hóa về nguồn cấp cho khóa điện 12V: đề xuất sử dụng các nguồn năng lượng có tính lâu dài

# Kết luận và đề xuất

## Dựa trên nghiên cứu và thử nghiệm trong đề tài " Nghiên cứu thiết kế chế tạo khóa cửa tự động ứng dụng công nghệ xử lý ảnh khuôn mặt", chúng tôi đã chứng minh rằng việc tích hợp công nghệ nhận diện khuôn mặt vào khóa cửa tự động mang lại nhiều lợi ích to lớn.

## Bằng cách sử dụng thuật toán FaceNet, chúng tôi đã phát triển một hệ thống nhận diện khuôn mặt chính xác và tin cậy, giúp người dùng mở khóa cửa một cách dễ dàng và an toàn hơn bằng việc chỉ cần một cái nhìn. Hệ thống này không chỉ giúp loại bỏ hoàn toàn nhu cầu sử dụng chìa khóa vật lý mà còn tăng cường đáng kể tính bảo mật.

## Ngoài ra, chúng tôi đã thực hiện các thử nghiệm thích nghi với các điều kiện ánh sáng và góc chụp khác nhau, và kết quả cho thấy hệ thống vẫn hoạt động ổn định, độ chính xác cao và đáng tin cậy. Điều này làm tăng tính linh hoạt và sự tiện lợi của sản phẩm trong các tình huống thực tế.

## Trong tương lai, chúng tôi kỳ vọng rằng công nghệ này sẽ tiếp tục được phát triển và tích hợp vào nhiều ứng dụng khác nhau, không chỉ trong lĩnh vực bảo mật cửa nhà mà còn trong các lĩnh vực như an ninh công ty, quản lý điểm danh, và nhiều ứng dụng khác. Điều này mở ra một tương lai đầy tiềm năng cho việc ứng dụng công nghệ nhận diện khuôn mặt trong các hệ thống tự động hóa và bảo mật

# Phụ lục

## FaceNet nhận một hình ảnh của khuôn mặt của người đó làm đầu vào và đầu ra là một vector gồm 128 số, đại diện cho các đặc trưng quan trọng nhất của một khuôn mặt. Trong học máy, vector này được gọi là nhúng (embedding), tất cả các thông tin quan trọng từ một hình ảnh được nhúng vào vector này. Cơ bản, FaceNet nhận một khuôn mặt của một người và nén nó thành một vector gồm 128 số. Về lý tưởng, những nhúng của các khuôn mặt tương tự cũng sẽ tương tự.

## A computer screen shot of a computer screen Description automatically generated

Hình 3.9: Facenet phân tích đặc điểm khuôn mặt con người

## Ánh xạ dữ liệu có số chiều cao (như hình ảnh) thành biểu diễn số chiều thấp (nhúng) đã trở thành một thực hành khá phổ biến trong học máy ngày nay

## A collage of people's faces Description automatically generated

Hình 3.10: Biểu diễn dữ liệu trong hệ tọa độ descartes

## Nhúng là các vector và ta có thể hiểu các vector như là các điểm trong hệ tọa độ Descartes. Điều đó có nghĩa là chúng ta có thể vẽ một hình ảnh của một khuôn mặt trong hệ tọa độ bằng cách sử dụng các nhúng của nó. Một cách tiếp cận có thể để nhận diện một người trên một hình ảnh chưa được biết trước là tính toán nhúng của nó, tính toán khoảng cách đến các hình ảnh của những người được biết trước và nếu nhúng của khuôn mặt đó đủ gần với những nhúng của người A, chúng ta nói rằng hình ảnh này chứa khuôn mặt của người A.

## 

Hình 3.11: Tạo vector ngẫu nhiên từ bộ dữ liệu

## Để huấn luyện FaceNet, chúng ta cần một lượng lớn hình ảnh của khuôn mặt. Để giữ cho mọi thứ đơn giản, chúng ta sẽ giả định chúng ta chỉ có một vài hình ảnh từ hai người. Cùng một logic có thể được áp dụng nếu chúng ta có hàng nghìn hình ảnh của các người khác nhau. Ở đầu quá trình huấn luyện, FaceNet tạo ra các vector ngẫu nhiên cho mỗi hình ảnh, điều này có nghĩa là các hình ảnh được phân tán ngẫu nhiên khi vẽ.

## A collage of people's faces Description automatically generated

Hình 3.12: Các dữ liệu được phân tán ngẫu nhiên

## FaceNet học theo cách sau:

## Chọn một hình ảnh gốc ngẫu nhiên.

## Chọn một hình ảnh cùng một người như hình ảnh gốc (ví dụ tích cực).

## Chọn một hình ảnh của một người khác với hình ảnh gốc (ví dụ tiêu cực).

## Điều chỉnh các tham số mạng FaceNet sao cho ví dụ tích cực gần hơn với gốc hơn là ví dụ tiêu cực.

## Chúng ta lặp lại các bước này cho đến khi không còn thay đổi nào được thực hiện nữa, ta thấy tất cả các khuôn mặt của cùng một người đều gần nhau và xa hơn so với những người khác.

## Phương pháp học này với các ví dụ gốc, tích cực và tiêu cực được gọi là mất mát ba (triplet loss).

## A group of people's faces Description automatically generated

Hình 3.13: Chọn ngẫu nhiên 1 ảnh làm Anchor

## A screenshot of a computer screen Description automatically generated

Hình 3.14: So sánh với ví dụ tích cực (positive example)

## A screenshot of a computer screen Description automatically generated

Hình 3.15: So sánh với ví dụ tiêu cực (negative example)

## A screenshot of a computer screen Description automatically generated

Hình 3.16: Điều chỉnh thông số của dữ liệu

## A group of people's faces Description automatically generated

Hình 3.17: Nhóm các dữ liệu với nhau

## A group of people's faces Description automatically generated

Hình 3.18: Ví dụ tích cực gần với anchor hơn và ngược lại

## A person smiling at the camera Description automatically generated

Hình 3.20: Phân tích khuôn mặt dưới dạng vector

## Chúng ta không trực tiếp định nghĩa với FaceNet rằng các số trong vector nên đại diện cho gì trong quá trình huấn luyện, chúng ta chỉ yêu cầu rằng các vector nhúng của các khuôn mặt tương tự cũng tương tự (tức là gần nhau). Nhiệm vụ của FaceNet là tìm ra cách biểu diễn khuôn mặt bằng các vector sao cho các vector của các người cùng một nhóm là tương tự và các vector của các người khác nhau không phải là tương tự. Để thực hiện đúng, FaceNet cần xác định các đặc điểm chính của khuôn mặt của một người mà phân biệt nó với các khuôn mặt khác. FaceNet đang thử nghiệm nhiều sự kết hợp khác nhau của các đặc điểm này trong quá trình huấn luyện cho đến khi nó tìm ra những con số hoạt động tốt nhất. FaceNet (hoặc các mạng nơ-ron nói chung) không biểu diễn các đặc điểm trong một hình ảnh theo cách giống như chúng ta (khoảng cách, kích thước, v.v.). Đó là lý do tại sao việc giải thích các vector này là khó khăn, nhưng chúng ta khá chắc chắn rằng có một cái gì đó như khoảng cách giữa hai mắt ẩn sau các số trong một vector nhúng.