

Xây dựng hệ thống nhận dạng khuôn mặt sử dụng Raspberry Pi 4

Bùi Xuân Vinh*, Lê Thị Ngọc Hà*

*ThS. Trường Đại học Sư phạm Kỹ thuật Vinh

Received: 16/7/2024; Accepted: 23/7/2024; Published: 31/7/2024

Abstract: Face recognition is a challenging field in computer vision and machine learning. Most existing face recognition systems rely on powerful computational resources such as DSPs or general-purpose computers, making them difficult to implement in medium and small-scale projects like biometric identification for home security systems, timekeeping, and employee management in companies. We introduce a new embedded hardware platform for image processing, specifically the Raspberry Pi 4, utilizing Intel's open-source image processing library OpenCV. We employ Haar-like features for face detection and Principal Component Analysis (PCA) for face recognition, all implemented on the Raspberry Pi 4 board. The system is designed with limited hardware resources, low cost, and low energy consumption, ensuring good recognition performance and speed.

Keywords: Raspberry Pi; PCA; face recognition.

1. Giới thiệu

Nhận dạng mặt người là một lĩnh vực nghiên cứu của ngành Computer Vision, và cũng được xem là một lĩnh vực nghiên cứu của ngành Biometrics (tương tự như nhận dạng vân tay - Fingerprint recognition, hay nhận dạng mống mắt - Iris recognition). Xét về nguyên tắc chung, nhận dạng mặt có sự tương đồng rất lớn với nhận dạng vân tay và nhận dạng mống mắt, tuy nhiên sự khác biệt nằm ở bước trích chọn đặc trưng của mỗi lĩnh vực. Trong khi nhận dạng vân tay và mống mắt đã đạt tới độ chín, tức là có thể áp dụng trên thực tế một cách rộng rãi thì nhận dạng mặt người vẫn còn nhiều thách thức và vẫn là một lĩnh vực nghiên cứu thú vị với nhiều người. So với nhận dạng vân tay và mống mắt, nhận dạng mặt có nguồn dữ liệu phong phú hơn và ít đòi hỏi sự tương tác có kiểm soát hơn.

Nhận dạng khuôn mặt được ứng dụng rất rộng rãi trong đời sống hằng ngày của con người, như các hệ thống bảo mật, an ninh, giám sát, quản lý vào ra, tìm kiếm thông tin, ứng dụng trong mạng xã hội như Facebook, google... Vì vậy nhận dạng khuôn mặt được nhiều nhóm nghiên cứu, công ty công nghệ lớn trên thế giới quan tâm. Thêm nữa cùng với sự ra đời và phát triển bùng nổ của khoa học máy tính, nhận dạng khuôn mặt đang đòi hỏi nhiều yêu cầu cao về kết quả, độ tin cậy, tính bảo mật.

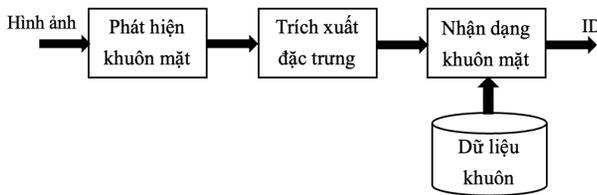
Hiện nay, tất cả các thuật toán nhận diện khuôn mặt được phát triển phần lớn đều dựa trên nền tảng hệ thống xử lý tín hiệu số DSP (Digital Signal

Processing) với lượng tài nguyên tính toán rất lớn và giá thành cao, vì vậy khó có thể tích hợp vào các hệ thống nhận dạng yêu cầu giá thành thấp. Để giải quyết vấn đề này, bài báo hướng đến việc tối ưu hóa thiết kế trên phần cứng là máy tính nhúng Raspberry Pi, với đặc trưng là tài nguyên tính toán giới hạn nhỏ gọn, giá thành thấp nhưng vẫn đảm bảo độ chính xác nhận dạng.

Bài báo trình bày việc thực thi và đánh giá hệ thống nhận dạng khuôn mặt trên máy tính nhúng Raspberry Pi 4 thông qua các bộ thư viện khác nhau, từ đó thu thập bộ thông số phù hợp nhất cho ứng dụng trên Raspberry Pi. Việc giải quyết bài toán này có thể mở ra một hướng mới trong ứng dụng công nghệ nhận diện khuôn mặt cho điều tra tội phạm, kiểm tra hành khách ở sân bay, điều khiển giao thông hay các hệ thống bảo mật khác, cho phép giảm giá thành, thời gian đáp ứng vừa phải, độ tin cậy cao.

2. Các kỹ thuật nhận dạng khuôn mặt

Sơ đồ khối tổng quát của hệ thống nhận dạng khuôn mặt được trình bày ở sơ đồ 2.1. Trong sơ đồ này, quá trình huấn luyện hệ thống với dữ liệu huấn luyện và quá trình nhận dạng với dữ liệu kiểm tra được trình bày đồng thời, trong đó các công đoạn phát hiện khuôn mặt, tiền xử lý, trích chọn đặc trưng và đối sánh được thực hiện tương tự ở cả hai quá trình huấn luyện và kiểm tra. Trong quá trình huấn luyện, các đặc trưng được trích chọn sẽ được lưu vào cơ sở dữ liệu để thực hiện nhận dạng thông qua đối sánh trong quá trình kiểm tra.



Sơ đồ 2.1. Kiến trúc tổng quan của hệ thống nhận dạng khuôn mặt.

Các kỹ thuật tương ứng được chúng tôi sử dụng trong hệ thống nhận dạng khuôn mặt là: kỹ thuật phát hiện khuôn mặt với đặc trưng Haar-like, Kỹ thuật trích chọn đặc trưng phương pháp phân tích thành phần chính PCA và kỹ thuật đối sánh mẫu sử dụng khoảng cách Euclidian.

2.1. Kỹ thuật phát hiện khuôn mặt với đặc trưng Haar-like

Mỗi đặc trưng Haar-like là các hình chữ nhật được phân thành các vùng trắng đen khác nhau, có thể chứa 2, 3 hoặc 4 miền hình học khác màu nhau. Để phát hiện khuôn mặt, việc đầu tiên là phải huấn luyện dựa vào các đặc trưng Haar-like. Ảnh đầu vào trước hết phải được xám hóa, sau đó cho một hoặc nhiều hình chữ nhật chứa đặc trưng Haar-like chạy khắp bức ảnh, những chỗ khác biệt sẽ được lưu trữ lại. Tập hợp hàng nghìn điểm khác biệt của hàng trăm khuôn mặt khác nhau sẽ cho ta một tập dữ liệu dùng cho việc phát hiện mặt người.

Trong quá trình huấn luyện, bộ phân loại phải duyệt qua tất cả các đặc trưng của các mẫu trong tập *training*, vì vậy sẽ tốn rất nhiều thời gian. Thực tế là trong các mẫu đưa vào, không phải mẫu nào cũng thuộc loại khó nhận dạng, có những mẫu *background* rất dễ nhận ra (đây là những mẫu *background* đơn giản). Đối với những mẫu này, chỉ cần xét một hay vài đặc trưng đơn giản là có thể nhận diện được chứ không cần xét tất cả các đặc trưng. Một kỹ thuật được sử dụng để rút ngắn thời gian xử lý, giảm thiểu tỉ lệ sai lầm là *Cascade of Classifiers*. Trong kỹ thuật này, *Cascade tree* gồm nhiều *stage* (hay còn gọi là *layer*), mỗi *stage* của cây sẽ là một *stage classifier*. Một mẫu để được phân loại là đối tượng thì nó cần phải đi qua hết tất cả các *stages* của cây. Các *stage classifiers* ở *stage* sau được huấn luyện bằng những mẫu *negative* mà *stage classifier* trước nó nhận dạng sai, tức là nó sẽ tập trung học từ các mẫu *background* khó hơn, do đó sự kết hợp các *stage classifiers* này lại sẽ giúp bộ phân loại có *false alarm* thấp. Với cấu trúc này, những mẫu *background* dễ nhận diện sẽ bị loại ngay từ những *stages* đầu tiên, giúp đáp ứng tốt nhất đối

với độ phức tạp gia tăng của các mẫu đưa vào, đồng thời giúp rút ngắn thời gian xử lý.

2.2. Kỹ thuật trích chọn đặc trưng phương pháp phân tích thành phần chính PCA

Phương pháp phân tích thành phần chính PCA (Principle Components Analysis), là kỹ thuật hữu ích trong các ứng dụng nhận dạng và nén ảnh, và là một kỹ thuật phổ biến để tìm mẫu trong các dữ liệu nhiều chiều.

Một bức ảnh khuôn mặt 2D có kích thước $R \times C$, có thể xem như 1 vectơ có chiều $N = R \times C$. Toàn bộ bức ảnh được biểu diễn bởi 1 tập hợp các điểm trong 1 không gian rất rộng lớn. Ý tưởng được đặt ra là làm thế nào có thể biểu diễn khuôn mặt bằng các vectơ trong một không gian con khác nhưng vẫn chứa được các đặc trưng nhất của bức ảnh, không gian này được gọi là “không gian mặt người”. Mỗi vectơ này có chiều dài $R \times C$, mô tả bức ảnh $R \times C$ là sự kết hợp tuyến tính của bức ảnh gốc. Bởi vì các vectơ này là các *eigenvector* của ma trận *covariance* của bức ảnh gốc và chúng giống mặt người khi biểu diễn nên được gọi là các *eigenface*. Các bước trích chọn đặc trưng với PCA:

- **Bước 1:** Tạo một tập S gồm M ảnh (ảnh học). Mỗi ảnh có kích thước $R \times C$. Mỗi ảnh được chuyển thành một vector $N = R \times C$ chiều:

$$S = \{\Gamma_1, \Gamma_2, \dots, \Gamma_M\} \quad (1)$$

- **Bước 2:** Tính ảnh trung bình Ψ của tập hợp trên:

$$\Psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i \quad (2)$$

- **Bước 3:** Tính sai lệch của các ảnh đầu vào so với trung bình: $\Phi_i = \Gamma_i - \Psi$ (3)

- **Bước 4:** Tính ma trận hiệp phương sai C:

$$C = \frac{1}{M} \sum_{i=1}^M \Phi_i \Phi_i^T = A \cdot A^T \quad (4)$$

C có kích thước $N \times N$. Vấn đề về tìm vectơ riêng (*eigenvector*) u_i của ma trận C khó thực hiện được vì kích thước quá lớn.

- **Bước 5:** Tìm *eigenvector* u_i của C ta thực hiện:

Giả sử v_i là vectơ riêng của ma trận $A^T A$, tức là:

$$A^T A v_i = \mu_i v_i \quad (5)$$

Nhân 2 vế của (5) với ma trận A ta được:

$$A A^T A v_i = \mu_i A v_i \quad (6)$$

Như vậy $A v_i$ là *eigenvector* của C.

- **Bước 6:** Tìm *eigenvector* và *eigenvalue* của ma trận L: $L = A^T A$ (7)

Khi đó L là ma trận có kích thước là $m \times m$, kích thước này nhỏ hơn nhiều so với $n \times n$ nên việc tính *eigenvector* sẽ nhanh chóng hơn.

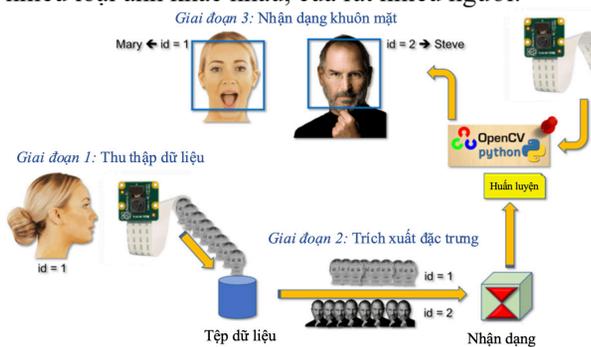
Giả sử v_i là *eigenvector* của L tính được:

$$u_i = \sum_{k=1}^M v_{ik} \Phi_k \quad (8)$$

Sau khi tìm được các *eigenface*, một ảnh trong cơ sở dữ liệu huấn luyện hay một ảnh mới đầu vào được chiếu lên không gian các *eigenface* này để tạo ra vectơ đặc tính. Vectơ này có kích thước nhỏ hơn nhiều so với kích thước ảnh nhưng vẫn mang nhiều nhất thông tin chứa trong ảnh.

3. Xây dựng hệ thống nhận dạng khuôn mặt sử dụng Raspberry Pi 4

Quá trình nhận dạng khuôn mặt trên Raspberry Pi 4 như sơ đồ 2.2. Đây là quá trình nhận dạng khuôn mặt trên máy tính nhúng Raspberry Pi 4 sử dụng thư viện OpenCV. Với dữ liệu là hình ảnh khuôn mặt có sẵn, được tổng hợp lại trong bộ cơ sở dữ liệu với nhiều loại ảnh khác nhau, của rất nhiều người.



Sơ đồ 2.2. Quá trình nhận dạng khuôn mặt trên Raspberry Pi 4.

Quy trình được thực hiện gồm 3 giai đoạn như sau:

- *Giai đoạn 1: Phát hiện khuôn mặt và thu thập dữ liệu.*

Phát hiện khuôn mặt là phần có ý nghĩa quan trọng trong hệ thống nhận dạng khuôn mặt, hệ thống sử dụng phương pháp mạng neural tích chập nổi tầng đa nhiệm để thực hiện quá trình phát hiện và cân chỉnh khuôn mặt. Đầu ra của giai đoạn 1 là hình ảnh khuôn mặt với 5 điểm dấu gồm 2 điểm mắt, 1 điểm mũi và 2 điểm miệng của khuôn mặt.

- *Giai đoạn 2: Trích xuất đặc trưng.*

Hình ảnh khuôn mặt sau khi được phát hiện và cân chỉnh được nhúng qua một mạng Neural tích chập CNN. Hệ thống sử dụng model *MobileFaceNets* thực hiện trích xuất đặc trưng khuôn mặt. Đầu ra của giai đoạn 2 là vector 128 đặc trưng của khuôn mặt.

- *Giai đoạn 3: Nhận dạng khuôn mặt.*

Ảnh người cần nhận dạng được thu thập bởi Camera, sau đó quá trình xử lý ảnh khuôn mặt được thực hiện tương tự như giai đoạn 1 và giai đoạn 2.

Hình ảnh sau xử lý cũng được trích xuất 128 đặc trưng và được đối sánh với cơ sở dữ liệu trong hệ thống bằng cách tính khoảng cách *Euclidean*. Quá trình đối sánh tìm ra khuôn mặt trong cơ sở dữ liệu có khoảng cách *Euclidean* so với khuôn mặt cần nhận dạng nhỏ hơn một ngưỡng (siêu tham số) để hệ thống định danh được khuôn mặt đầu vào.

4. Kết luận

Bài báo này đã thiết kế và thực thi thành công một hệ thống nhận dạng mặt người trên board mạch nhúng Raspberry Pi 4. Các kết quả cho thấy hệ thống có thể đáp ứng được yêu cầu của bài toán nhận dạng mặt người ứng dụng trong bảo mật trên các hệ thống nhúng. Với nền tảng phần cứng Raspberry Pi, bài báo đã thực hiện được các thuật toán phát hiện và nhận dạng khuôn mặt người với độ chính xác cao hơn 90% khi lựa chọn sử dụng 10 ảnh cho mỗi đối tượng và 10 *eigenface* cho mỗi ảnh. So với giải pháp nhận dạng khuôn mặt trên máy tính đa mục đích, tỉ lệ nhận dạng đúng trên Raspberry Pi gần như tương đương, ngoại trừ rằng tốc độ nhận dạng chậm hơn vì những giới hạn về phần cứng của Raspberry Pi 4 (tốc độ vi xử lý, bộ nhớ). Việc xử lý các ảnh offline cho độ chính xác với thời gian nhận dạng đáp ứng thời gian thực. Kết quả nhận dạng khuôn mặt online cho thấy các cơ cấu điều khiển đã chấp hành tốt, thời gian mỗi lần nhận dạng trong khoảng 11s.

Việc xây dựng thành công mô hình nhận dạng khuôn mặt trên Raspberry Pi đã mở ra nhiều khả năng ứng dụng vào thực tế như: hệ thống kiểm soát vào ra bằng nhận dạng khuôn mặt dùng cho các cơ quan công sở, khu chung cư cao cấp hoặc biệt thự để kiểm soát được đối tượng và thời điểm ra vào. Bên cạnh đó, hệ thống chấm công không cần dùng thẻ dùng trong hệ thống quản lý, chấm công và giám sát nhân lực lao động phù hợp cho các nhà máy, khu công nghệ cao, cơ quan, đơn vị quản lý nhân lực lao động. So với giải pháp dùng máy tính thông dụng, các sản phẩm xây dựng trên Raspberry Pi 4 có thiết kế nhỏ gọn, giá thành thấp và tiêu thụ điện năng rất thấp.

Tài liệu tham khảo

- [1]. Nguyễn Mạnh Hùng, *Nhận dạng mặt người sử dụng đặc trưng PCA*, Luận văn Thạc sĩ Khoa học máy tính, Học viện Công nghệ Bưu chính Viễn thông, Hà Nội, 2013.
- [2]. Nguyễn Tất Bảo Thiện, *Lập trình hệ thống nhúng với Raspberry*, NXB Thanh Niên, Hà Nội, 2019.
- [3]. Simon Monk, *Programming the Raspberry Pi Getting started with Python*, McGraw-Hill Education. United States, 2013.