

Nhận dạng người dựa trên hình ảnh khớp ngón tay sử dụng Raspberry Pi

Personal verification based on finger knuckle print recognition system using Raspberry Pi

Trần Thị Thảo^{1*}, Phạm Văn Trường¹

¹Khoa Tự động hóa, Trường Điện-Điện tử, Đại học Bách Khoa Hà Nội

* Corresponding author E-mail: thao.tranthi@hust.edu.vn

Tóm tắt

Bài báo đề xuất một hệ thống nhận diện khớp ngón tay người (finger knuckle recognition-FKR)) sử dụng máy tính nhúng Raspberry Pi 4 Model B và camera Pi. Nghiên cứu bắt đầu bằng việc khảo sát các phương pháp trích xuất đặc trưng và đối sánh gần đây, sau đó đánh giá về tốc độ xử lý và độ chính xác nhận dạng để từ đó lựa chọn phương pháp phù hợp. Quá trình phân tích cho thấy phương pháp sử dụng đặc trưng về vector đồng bộ định hướng nhị phân BOCV- vốn được đề xuất trong các nghiên cứu trước đây về nhận dạng lòng bàn tay, cho kết quả tốt khi thử nghiệm trên bài toán nhận diện khớp ngón tay. Bên cạnh đánh giá trên các dữ liệu ảnh tự thu thập, để đánh giá kết quả phương pháp đề xuất, chúng tôi đã đánh giá thuật toán trên bộ dữ liệu mở của Đại học Bách Khoa HongKong. Kết quả cho thấy, mô hình BOCV cho độ chính xác trên 90% mà vẫn đáp ứng thời gian thực.

Keywords: Nhận dạng khớp ngón tay; Định danh người; Raspberry PI; Sinh trắc học; BOCV;

Abbreviations

FKP	Finger Knuckle Print
OCV	Orientation Cooccurrence Vector
CSDL	Cơ sở dữ liệu
ROI	Region Of Interest

Abstract

This paper proposes an approach for personal verification based on finger knuckle recognition (FKR) system using Raspberry Pi 4 and Pi camera. We evaluate and compare some advanced methods for biometric feature extraction and matching in terms of recognition accuracy and computational time. From the evaluation and analysis, we found that the Binary Orientation Cooccurrence Vector (BOCV)-which is previously often used for palmprint recognition, is the method of choice for our low-cost and real time FKR system. Along with evaluating our data, we also assess some recent methods on the Hong Kong Polytechnic University Contactless Finger Knuckle Images Database. Experiments show that the BOCV method obtains the accuracy over 90% and offers realtime recognition.

1. Đặt vấn đề

Trong những thập kỷ gần đây, sinh trắc học nổi lên như một công nghệ đáng tin cậy cho các hệ thống xác thực cá

nhân với mức độ bảo mật cao hơn so với một số phương pháp truyền thống khác như nhập mã pin, mật khẩu. Trong số các đặc điểm sinh trắc học được sử dụng, bàn tay con người là hình thức công nghệ sinh trắc học được ra đời từ lâu, phát triển và được áp dụng rộng rãi. Các cấu trúc có thể được trích xuất từ bàn tay bao gồm hình học bàn tay, dấu vân tay, lòng bàn tay, khớp ngón tay. Những tính chất liên quan đến bàn tay có tính ổn định cao, không thay đổi theo thời gian và đáng tin cậy. Khi một người đã đến tuổi trưởng thành, cấu trúc và cấu hình bàn tay vẫn tương đối ổn định trong suốt cuộc đời của người đó. Ngoài ra, công nghệ quét các cấu trúc của tay được coi là không xâm lấn so với các hệ thống quét mống mắt hoặc võng mạc. Người dùng không cần phải nhận thức về cách họ tương tác với hệ thống. Những lợi thế này đã tạo điều kiện thuận lợi cho việc triển khai các đặc tính của tay trong các ứng dụng sinh trắc học.

Trong các bài toán nhận dạng sử dụng các thông tin về bàn tay con người thì nhận diện khớp ngón tay người là một trong những vấn đề đang thu hút nhiều quan tâm trong nghiên cứu và phát triển, với rất nhiều ứng dụng trong thực tế cuộc sống cũng như trong khoa học công nghệ. Nhận diện khớp ngón tay có nhiều ưu điểm nổi trội. Cụ thể là, so với nhận dạng lòng bàn tay, bài toán nhận diện khớp ngón tay có những ưu điểm như có tính duy nhất cao hơn, việc thu nhận hình ảnh khớp ngón tay dễ dàng thực hiện hơn, và rõ nét hơn. So với vân tay, hình ảnh của khớp ngón tay ít bị mờ do bị tác động trong quá trình sử dụng các công cụ lao động.

Hiện nay, sự phát triển mạnh mẽ của ngành khoa học máy tính, xử lý ảnh và các công nghệ hỗ trợ khác đã tạo môi

trường thuận lợi cho bài toán nhận diện khớp ngón tay từ ảnh số. Một số hệ thống nhận diện khớp ngón tay FKR đã ra đời với độ tin cậy cao [1, 2]. Trong các ứng dụng này, hầu hết các thiết bị thu nhận cấu trúc hình học của tay đều dựa trên thiết kế dựa trên cảm ứng. Tức là người dùng được yêu cầu chạm vào thiết bị hoặc giữ một số chốt hướng dẫn hoặc ngoại vi để chụp ảnh bàn tay của họ [3].

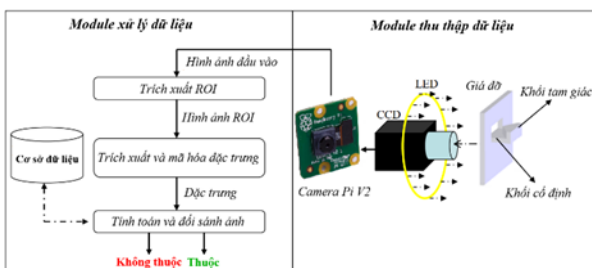
Tuy vậy, nhận diện khớp ngón tay vẫn là một bài toán khó. Có thể kể đến những khó khăn của bài toán này như góc chụp và điều kiện của ảnh. Cụ thể là, ảnh chụp khớp ngón tay có thể thay đổi rất nhiều bởi góc chụp giữa camera và bàn tay. Chẳng hạn như: chụp thẳng, chụp chéo bên trái, bên phải ở các góc khác nhau...). Bên cạnh đó, ảnh được chụp trong các điều kiện khác nhau về chiều sáng, về tính chất camera (máy kỹ thuật số, máy ảnh hồng ngoại...) ảnh hưởng rất nhiều đến chất lượng hình ảnh chụp khớp ngón tay. Bên cạnh đó, trong các phương pháp nhận diện khớp ngón tay phổ biến hiện nay, thường dựa trên các thuật toán có khối lượng tính toán lớn như nhóm các phương pháp dựa trên biểu diễn thưa (sparse representation) hay học sâu (deep learning) [4, 5] dẫn đến yêu cầu cần có thiết bị tính song song GPU giá thành cao, khó thực hiện trên máy tính nhúng....

Trong nghiên cứu này, chúng tôi hướng tới mục tiêu xây dựng một hệ thống với giá thành thấp nhưng vẫn đảm bảo độ chính xác chấp nhận được. Cụ thể là, nghiên cứu, đề xuất dùng phương pháp sử dụng đặc trưng về vector đồng bộ định hướng nhị phân BOCV- vốn được đề xuất trong các nghiên cứu trước đây về nhận dạng lòng bàn tay, cho bài toán nhận diện khớp ngón tay. Thêm vào đó, phương pháp trong nghiên cứu này có thể triển khai trên hệ thống giá thành thấp sử dụng máy tính nhúng Raspberry Pi 4 Model B mà vẫn đáp ứng thời gian thực.

2. Hệ thống nhận dạng khớp ngón tay

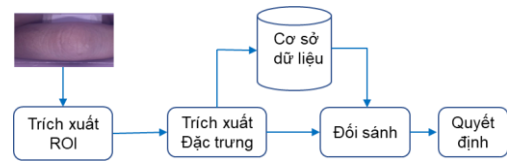
2.1. Cấu trúc chung của hệ thống

Hệ thống nhận diện khớp ngón tay cơ bản được chia làm hai module chính như minh họa ở Hình 1. Thứ nhất là module thu thập dữ liệu phụ trách việc thêm dữ liệu vào cơ sở dữ liệu và lấy dữ liệu để đối sánh với cơ sở dữ liệu có sẵn. Module thứ hai phụ trách việc xử lý dữ liệu đầu vào và so sánh với cơ sở dữ liệu để đưa ra kết quả người đó có thuộc cơ sở dữ liệu có sẵn hay là không. Mô hình này xây dựng dựa trên bài báo [3]. Ở đây chúng tôi đề xuất giải pháp để thích nghi cho thực hiện trên máy tính Pi với một phương pháp trích xuất đặc trưng và đối sánh khác với tốc độ xử lý nhanh hơn rất nhiều so với bài báo gốc.



Hình 1: Cấu trúc của một hệ thống định danh/xác thực người dựa trên hình ảnh khớp ngón tay sử dụng máy tính nhúng PI. Mô hình xây dựng dựa trên bài báo [3]

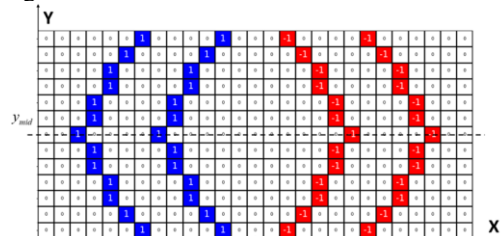
Các bước chính của một bài toán nhận dạng khớp ngón tay người gồm có 3 bước cơ bản như ở hình 2, bao gồm: trích xuất vùng quan tâm (ROI), trích chọn đặc trưng, và đối sánh ảnh. Khi hình ảnh ngón tay của người dùng được chụp, các vùng quan tâm ROI của ngón tay sẽ được trích xuất. Sau đó, các tính năng phân biệt trong ROI có thể được trích xuất bằng cách sử dụng các kỹ thuật và các thuật toán đa dạng để xác định, nhận diện khớp ngón tay đó là của ai, nhóm người quan tâm nào theo từng yêu cầu cụ thể của mỗi bài toán.



Hình 2: Các bước chính của một bài toán nhận dạng khớp ngón tay người.

2.2. Trích xuất vùng quan tâm ROI

Khi hình ảnh ngón tay của người dùng được chụp, các vùng quan tâm ROI (Region of Interest) của ngón tay sẽ được trích xuất, ví dụ khớp chính, khớp phụ ngón tay, móng tay. ROI chứa thông tin quan trọng của khớp ngón tay được sử dụng để nhận dạng. Ở nghiên cứu này chúng tôi lựa chọn sử dụng phần khớp chính – major FKG để nhận diện. ROI được xử lý trước để kết cấu của khớp ngón tay có thể phân biệt được với nền. Sau đó, các tính năng phân biệt trong ROI có thể được trích xuất bằng cách sử dụng các kỹ thuật và các thuật toán đa dạng để xác định, nhận diện khớp ngón tay đó là của ai, nhóm người quan tâm nào theo từng yêu cầu cụ thể của mỗi bài toán. Cụ thể trong bước trích xuất vùng ROI, chúng tôi thực hiện các bước chính sau: Đầu tiên ta sẽ tiến xử lý ảnh bằng các phương pháp làm mịn Gaussian, giảm độ phân giải. Bước tiếp theo là xác định trục X của vùng quan tâm và cắt ảnh thu thập để chỉ lần miền gần với vùng xác định. Sở dĩ ta xác định trục X trước vì thông thường đường biên dưới cùng của ngón tay tách biệt với nền có thể áp dụng phân ngưỡng với toàn bộ hình ảnh FKR thu thập được. Bước tiếp theo là dò cạnh, ở đây chúng tôi dùng phương pháp Canny để dò cạnh với thuật toán đã nhúng trong gói OpenCV [6], giúp các vân của khớp ngón tay hiện lên. Từ cạnh dò được, ta cần mã hóa đường vân lồi của khớp ngón tay, như minh họa ở Hình 3 để từ đó xác định tâm- coi như trung tâm của vùng lồi về hai bên như ở trên hình vẽ, đồng thời xác định trục Y dựa trên thông tin về vùng đường vân lồi. Cuối cùng là cắt vùng ROI dựa trên tọa độ X, Y cũng như các thông tin về vân lồi ở các bước trên.



Hình 3: Minh họa sơ đồ mã hóa đường vân lồi của khớp ngón tay.

2.3. Trích xuất đặc trưng và đối sánh ảnh

Sau khi đã trích xuất được vùng quan tâm (ROI), việc tiếp theo là tiến hành sử dụng hình ảnh ROI thu được để nhận

diện khớp ngón tay người. Hiện nay có rất nhiều phương pháp trích xuất đặc trưng trong sinh trắc học, như Imcomp-Code&MagCode [3], CompCode [1], Localized Radon Transform – LRT [7], Binary Orientation Co-occurrence Vector– BOCV [8]. Trong nghiên cứu này, chúng tôi sử dụng phương pháp trích xuất thông tin đặc trưng của khớp ngón tay sử dụng phương pháp nhị phân hóa sơ đồ mã hóa định hướng (BOCV) [8] do chi phí tính toán thấp, phù hợp với hệ thống thời gian thực sử dụng máy tính nhúng Raspberry.

2.3.1. Biến đổi sóng Gabor

Trong bài toán nhận diện khớp ngón tay, để trích xuất đặc trưng, người ta thường thực hiện một phép tích chập giữa ảnh với một hàm lọc Gabor. Hàm Gabor được đề xuất trong bài toán về sinh trắc học được tóm tắt như dưới đây. Công thức Gabor, như giới thiệu trong công bố của Tai Sing Lee [9], được cấu thành từ hàm Gaussian nhân với hàm sin(cos) được trình bày một cách tổng quan như sau:

$$\psi(x, y, \omega_0, \theta) = \frac{\omega_0}{\sqrt{2\pi\kappa}} e^{-\frac{\omega_0^2}{8\kappa^2}(4x'^2 + y'^2)} \left(e^{i\omega_0 x'} - e^{-\frac{\kappa^2}{2}} \right) \quad (1)$$

với

$$x' = x \cos \theta + y \sin \theta \quad (2)$$

$$y' = -x \sin \theta + y \cos \theta \quad (3)$$

trong đó:

ω_0 là tần số xuyên tâm tính theo đơn vị radian trên một đơn vị chiều dài.

θ là góc định hướng của phép biến đổi sóng Gabor, tính theo đơn vị radian.

κ là một hằng số, được lựa chọn tùy thuộc vào băng thông tần số của hệ thống.

Các thành phần trong phương trình bao gồm:

x' là tích vô hướng của vector (x, y) và vector $\overline{Oa} = (\cos \theta, \sin \theta)$.

Tương tự, y' là tích vô hướng của vector (x, y) và vector $\overline{Ob} = (-\sin \theta, \cos \theta)$.

Nhận xét rằng: 2 vector \overline{Oa} và \overline{Ob} vuông góc với nhau (có tích vô hướng bằng 1). Như vậy có thể hiểu rằng (x', y') chính là điểm (x, y) sau khi đã được biến đổi từ hệ tọa độ góc Oxy sang hệ tọa độ mới $O'xy$ với \overline{Oa} và \overline{Ob} chính là hai vector đơn vị. Khi $\theta = 0$ thì $x' = x, y' = y$, hệ tọa độ $O'xy \equiv Oxy$.

Công thức Gabor được biểu diễn dưới dạng số phức, bao gồm hai phần là phần thực và phần ảo. Sử dụng công thức Euler, $e^{ix} = \cos x + i \sin x$, ta có thể biểu diễn phần thực và ảo của phép biến đổi sóng Gabor tương ứng theo các công thức sau:

$$\psi_R(x, y, \omega_0, \theta) = \frac{\omega_0}{\sqrt{2\pi\kappa}} e^{-\frac{\omega_0^2}{8\kappa^2}(4x'^2 + y'^2)} \left(\cos(\omega_0 x') - e^{-\frac{\kappa^2}{2}} \right) \quad (4)$$

$$\psi_I(x, y, \omega_0, \theta) = \frac{\omega_0}{\sqrt{2\pi\kappa}} e^{-\frac{\omega_0^2}{8\kappa^2}(4x'^2 + y'^2)} \left(\sin(\omega_0 x') - e^{-\frac{\kappa^2}{2}} \right) \quad (5)$$

2.3.2. Phương pháp BOCV

Thuật toán Binary Orientation Co-occurrence Vector (BOCV) vốn được đề xuất trong bài báo [8] cho bài toán nhận diện lòng bàn tay. Trong nghiên cứu này chúng tôi áp dụng cho bài toán nhận diện khớp ngón tay KFR. Phương pháp BOCV sử dụng phép biến đổi sóng Gabor để trích xuất thông số định hướng của khớp ngón tay để mã hóa hình ảnh ROI. Các bước chính trong việc dùng BOCV trong bài toán KFR được trình bày như phân dưới đây:

Bước 1: Tính toán vector OCV

Sau khi đã định nghĩa công thức của phép biến đổi sóng Gabor và lựa chọn thông số, tiếp đến ta sẽ tính toán một vector 6 chiều bằng cách chuẩn hóa theo 6 hướng của phép biến đổi sóng Gabor, gọi là vector định hướng (OCV-orientation co-occurrence) theo công thức sau:

$$OCV = \{G_j(x, y), j = \overline{0, 5}\} \quad (6)$$

với:

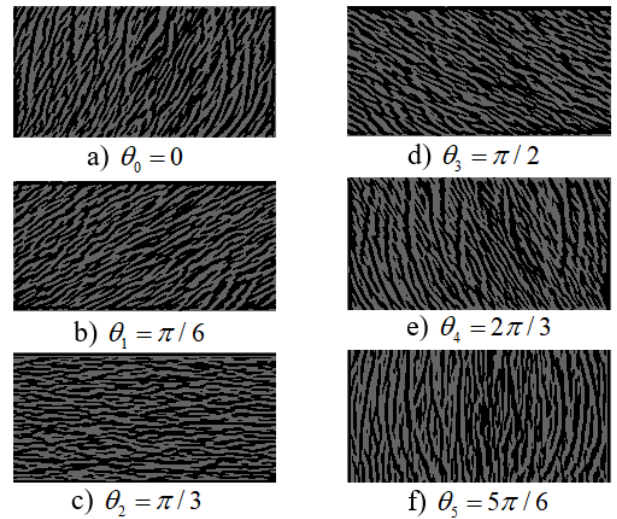
$$G_j(x, y) = I_{ROI}(x, y) * \psi_R(x, y, \omega_0, \theta), \theta_j = \frac{j\pi}{6} \quad (7)$$

Bước 2: Chuẩn hóa nhị phân vector định hướng OCV

Theo phương trình (6) và phương trình (7), được định nghĩa ở trên, các phần tử của vector OCV đều là số thực. Để tăng tốc độ tính toán trong quá trình đối sánh ảnh, ta sẽ chuẩn hóa vector OCV theo mã nhị phân 6 bit, gọi là Binary OCV (BOCV).

Công thức chuẩn hóa như sau:

$$P'_j = \begin{cases} 1, & G_j(x, y) \\ 0, & otherwise \end{cases} \quad (8)$$



Hình 4: Minh họa các đặc trưng BOCV với các hướng $\theta = 0, \pi/6, \pi/3, \pi/2, 2\pi/3, 5\pi/6$.

Bước 3: Xác định Mask cho hình ảnh đầu vào.

Thuật toán này ngoài sơ đồ mã hóa vector BOCV, chúng tôi cũng sử dụng thêm hình ảnh Mask tham khảo từ bài báo [10], để xác định những pixel không nằm trong vùng khớp ngón tay, ví dụ như khi ngón tay đặt lệch khỏi trục.

Bước 4: Đối sánh ảnh.

Để đối sánh và nhận diện ảnh, chúng tôi sử dụng vector mã hóa BOCV và mask của ảnh ROI đã được trích xuất.

Giả sử cần đối sánh hai ảnh ROI1 và ROI2 có cùng kích thước $X \times Y$, qua bước trích xuất dữ liệu ta thu được cặp vector mã hóa $BOCV1 = \{P_j^b, j = \overline{0,5}\}$

$BOCV2 = \{Q_j^b, j = \overline{0,5}\}$ và cặp mask là (P_M, Q_M) của 2 ảnh ROI1 và ROI2.

Như vậy, để đối sánh 2 ảnh ROI1 và ROI2, ta sẽ thực hiện tính toán khoảng cách để đối sánh 2 cặp ma trận $(BOVC1, BOVC2)$ và (P_M, Q_M) .

Khoảng cách đối sánh giữa hai ảnh ROI1 và ROI2 được tính theo công thức sau:

$$D(R_1, R_2) = \frac{\sum_{x=1}^X \sum_{y=1}^Y \sum_{j=0}^5 (P_j^b(x, y) \otimes Q_j^b(x, y)) \cap (P_M(x, y) \cap Q_M(x, y))}{6 \sum_{x=1}^X \sum_{y=1}^Y P_M(x, y) \cap Q_M(x, y)} \quad (9)$$

trong đó phép \otimes là phép logic XOR giữa 2 bit 0 và 1.

Nhận xét: theo phương trình (8) trên, ta thấy vector mã hóa BOCV và ma trận mask chứa các phần tử thuộc tập $\{0,1\}$, như vậy giá trị của $D(R_1, R_2)$ nhận được thuộc trong khoảng $(0,1)$.

Như vậy, để nhận diện được một ảnh, ta sẽ thực hiện đối sánh ảnh này với lần lượt từng ảnh trong cơ sở dữ liệu, thu được một tập các giá trị khoảng cách:

$$distance = \{D(R, R_j), j = \overline{0, n-1}\}$$

với R là hình ảnh ROI cần nhận diện, $R_j, j = \overline{0, n-1}$ là tập những hình ảnh ROI trong cơ sở dữ liệu để đối sánh, n là số ảnh trong cơ sở dữ liệu.

Dựa vào giá trị nhỏ nhất trong tập giá trị $distance$ đó, ta sẽ đưa ra kết luận người đó có thuộc cơ sở dữ liệu hay không và đó là người nào.

3. Kết quả thực nghiệm

Trong bài toán nhận diện khớp ngón tay, vấn đề đáp ứng thời gian thực là vấn đề cần được quan tâm. Thời gian xử lý trong thuật toán tốn nhiều nhất đó là bước trích xuất đặc trưng. Do vậy trước hết, để lựa chọn phương pháp trích xuất đặc trưng phù hợp cho nghiên cứu là dùng Raspberry Pi, chúng tôi thử nghiệm trên tập dữ liệu từ Trung tâm nghiên cứu sinh trắc học thuộc Đại học Bách Khoa HongKong với các phương pháp trích xuất đặc trưng phổ biến trong sinh trắc học. Các phương pháp thử nghiệm gồm ImcompCode&MagCode[3], CompCode [1], Localized Radon Transform – LRT [7], Binary Orientation Co-occurrence Vector – BOCV [8]. Sau khi khảo sát về hiệu năng và thời gian tính toán của các phương pháp trên, sẽ lựa chọn phương pháp phù hợp để thực hiện trên máy tính nhúng.

3.1. Khảo sát và thử nghiệm trên CSDL mở

Tập data đầu tiên được tham khảo từ Trung tâm nghiên cứu sinh trắc học thuộc Đại học Bách Khoa HongKong (<http://www.comp.polyu.edu.hk/~biometrics/>). Trong tập dữ liệu này, chúng tôi sử dụng 1776 hình ảnh ngón tay phải đã được trích xuất ROI của 148 người và chia tỷ lệ 80:20 cho huấn luyện và thử nghiệm. Để đánh giá về thời gian tính toán và hiệu quả của các phương pháp trích xuất đặc trưng

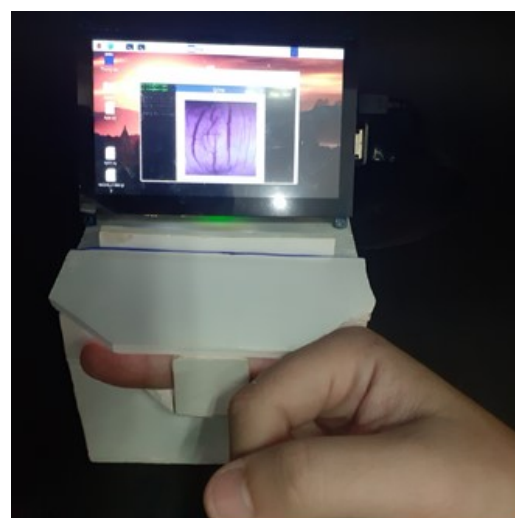
và đối sánh khác nhau, chúng tôi sử dụng máy tính có bộ xử lý song song GPU được cung cấp bởi Google Colaboratory. Kết quả so sánh được thể hiện trên Bảng 1. Qua bảng này ta thấy rằng các phương pháp CompCode, LRT, và ImcompCode & MagCode tuy cho kết quả nhận dạng cao nhưng thời gian tính toán rất lâu, khó có thể đáp ứng thời gian thực, nhất là nếu sử dụng máy tính nhúng. Vì thế chúng tôi lựa chọn phương pháp BOCV, do có thời gian tính toán nhanh nhất mà vẫn cho độ chính xác chấp nhận được.

Bảng 1: Kết quả so sánh thời gian tính toán cho một ảnh và độ chính xác trên toàn bộ tập thử nghiệm của tập dữ liệu mở

Phương pháp	Thời gian tính toán (giây)		Độ chính xác (%)
	Trích xuất đặc trưng	Đối sánh	
CompCode	0.59	0.023	94.59
BOCV	0.51	0.001	93.24
LRT	2.70	0.290	86.48
ImcompCode&MagCode	0.90	0.034	96.62

3.2. Triển khai hệ thống FKR trên Raspberry Pi

Trong nghiên cứu này, chúng tôi lựa chọn phần cứng là Raspberry Pi với bộ xử lý trung tâm CPU là Broadcom BCM2711B0 4 nhân 1.5GHz. Tuy không có cấu hình mạnh, tốc độ tính toán không cao nhưng thiết bị này lại thích hợp sử dụng trong các dự án đòi hỏi chi phí thấp, kích thước nhỏ gọn. Trên Hình 5 là hình ảnh hệ thống nhận diện khớp ngón tay xây dựng trên Raspberry Pi. Các hình ảnh được thu thập từ Pi Camera Module V2, qua các bước tiền xử lý, trích xuất ROI, trích xuất đặc trưng và đối sánh sử dụng phương pháp BOCV.



Hình 5: Hình ảnh hệ thống FKR trên Raspberry Pi

Mỗi hình ảnh đầu vào có kích thước 380×500 , sau khi xử lý định danh thì ROI sẽ được cắt còn 280×280 pixel. Với kích thước trên, hình ảnh từ ROI phù hợp để chứa toàn bộ thông

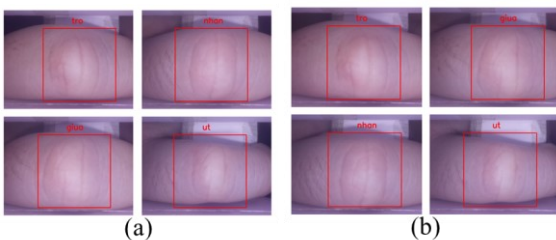
tin dùng để xử lý nhận diện hình ảnh. Về phần trích chọn vùng quan tâm, tốc độ xử lý trung bình 2.4s với một hình ảnh đầu vào kích thước 380×500. Về tốc độ trích chọn đặc trưng, mỗi ảnh từ vùng ROI kích thước 280×280 mất khoảng 5.5s. Còn bước đối sánh ảnh thì thời gian xử lý trung bình là 0.025s cho mỗi bức ảnh ROI.

Để đánh giá hệ thống, chúng tôi thu thập hình ảnh từ 12 người, mỗi người lấy 10 ảnh ngón tay giữa bên phải để đưa vào CSDL được định danh. Bên cạnh đó chúng tôi sử dụng thêm 5 ngón tay khác (không trùng với 12 ngón tay đã được định danh) chưa được định danh và chưa có trong cơ sở dữ liệu, để kiểm tra xem hệ thống có nhận diện ra được người lạ hay không. Mỗi ngón tay đều được kiểm tra 10 lần. Các hình ảnh này được chia ra thành phần huấn luyện và phần kiểm nghiệm theo tỷ lệ 8:2. Kết quả cho thấy độ chính xác 90% cho dữ liệu đã định danh, 92% cho dữ liệu chưa được định danh với tốc độ nhận diện xấp xỉ 11 giây cho lượng cơ sở dữ liệu 120 ảnh.

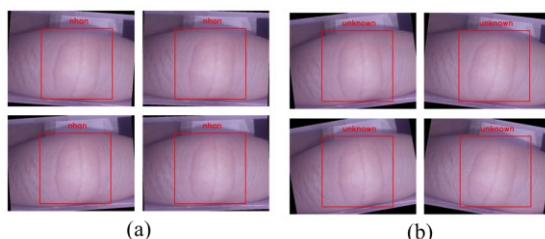
Để cải thiện độ chính xác khi nhận diện các ngón tay chưa định danh (hệ thống nhận diện là người lạ), ta có thể đặt ngưỡng nhận diện nhỏ hơn, nhưng việc này có thể làm giảm độ chính xác khi nhận diện những người đã được định danh có trong cơ sở dữ liệu. Bên cạnh đó, tốc độ xử lý của hệ thống phụ thuộc vào số lượng hình ảnh có trong tập cơ sở dữ liệu và kích thước hình ảnh đầu vào của hệ thống. Để cải thiện được tốc độ xử lý của hệ thống, có thể giảm số lượng hình ảnh định danh của một người, bên cạnh đó là giảm kích thước hình ảnh đầu vào của hệ thống.

3.3. Kết quả thử nghiệm

Kết quả thử nghiệm cho thấy hệ thống làm việc khá tốt trong các điều kiện ánh sáng bên ngoài khác nhau, ví dụ như có nhiều ánh sáng môi trường tác động cũng như không có ánh sáng môi trường, ban đêm, như minh họa ở Hình 6.



Hình 6: Kết quả nhận diện vùng khớp ngón tay khi có ánh sáng môi trường yếu (a), và ánh sáng môi trường mạnh (b)



Hình 7: Minh họa nhận diện vùng khớp ngón tay trong điều kiện ngón tay xoay góc nhỏ (a), và xoay góc lớn (b)

Khi thực hiện thử nghiệm với điều kiện ngón tay xoay các góc nhỏ (góc nhỏ hơn 5 độ) hệ thống vẫn nhận diện tương đối chính xác, như ở Hình 7. Nhưng trong trường hợp góc

xoay quá lớn làm cho vùng quan tâm không bắt được những đường vân trên khớp ngón tay thì hệ thống sẽ không nhận diện được. Khoảng cách đối sánh trả về lớn hơn ngưỡng nên kết quả trả về là khớp ngón tay là người lạ. Để khắc phục điều này, chúng tôi đã thiết kế hệ thống với những phần khối cố định giúp cho ngón tay người không bị xoay quá nhiều trong khi sử dụng, giúp tăng độ chính xác của hệ thống.

4. Kết luận

Trong bài báo này, chúng tôi đã giới thiệu một hệ thống nhận diện khớp ngón tay người sử dụng máy tính nhúng Raspberry Pi 4 Model B và Pi camera. Chúng tôi đã khảo sát đánh giá trên cơ sở dữ liệu mở với các phương pháp trích xuất đặc trưng và đối sánh gần đây trên phương diện tính toán và độ chính xác nhận dạng. Từ đánh giá và so sánh, chúng tôi đã lựa chọn phương pháp sử dụng đặc trưng về vector đồng bộ định hướng nhị phân BOCV để thực thi trên hệ thống phần cứng giá thành hạ dùng Raspberry Pi. Các kết quả đánh giá cho thấy hệ thống cho độ chính xác trên 90% với tốc độ đáp ứng thời gian thực.

Lời cảm ơn

Nghiên cứu này được tài trợ bởi trường Đại học Bách khoa Hà Nội (HUST) trong đề tài mã số: T2021- PC-005.

Tài liệu tham khảo

- [1] L. Zhang, L. Zhang, and D. Zhang, "Finger Knuckle Print: A New Biometric Identifier," in *16th IEEE International Conference on Image Processing (ICIP)*, 2009.
- [2] A. Attia, A. Moussaoui, M. Chaa, and Y. Chahir, "Finger Knuckle Print Recognition based on Features-Level Fusion of Real and Imaginary Image," *Journal on Image and Video Processing*, vol. 8, pp. 1793-1799, 2018.
- [3] L. Zhang, L. Zhang, and H. Zhu, "Online Finger-Knuckle-Print Verification for Personal Authentication," *Pattern Recognition*, vol. 43, pp. 2560-2571, 2010.
- [4] G. Arora, A. Singh, A. Nigam, H. M. Pandey, and K. Tiwari, "FKPIndexNet: An efficient learning framework for finger-knuckle-print database indexing to boost identification," *Knowledge-Based Systems*, vol. 239, p. 108028, 2022.
- [5] H. Heidari and A. Chalechale, "Biometric authentication using a deep learning approach based on different level fusion of finger knuckle print and fingernail," *Expert Systems with Applications*, vol. 191, p. 116278, 2022.
- [6] "Canny Edge Detection," *Online*. Available: https://docs.opencv.org/3.4/da/d22/tutorial_py_canny.html.
- [7] A. Kumar and Y. Zhou, "Personal Identification using Finger Knuckle Orientation Features," *Electronics Letters*, vol. 45, pp. 1023 – 1025, 2009.
- [8] Z. Guo, D. Zhang, L. Zhang, and W. Zuo, "Palmprint verification using binary orientation co-occurrence vector," *Pattern Recognition Letters*, vol. 30, pp. 1219–1227, 2009.
- [9] T. S. Lee, "Image representation using 2D Gabor wavelets," *IEEE Transactions on Pattern Analysis and Machine Intelligence* vol. 18, pp. 959 - 971, 1996.
- [10] D. Zhang and J. You, "Online palmprint identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 1041 - 1050, 2003.