

LỜI NÓI ĐẦU

Cảm biến (Sensor) là thiết bị dùng để cảm nhận biến đổi các đại lượng vật lý và các đại lượng không có tính chất điện cần đo thành các đại lượng điện có thể đo và xử lý được. Hiện nay, chúng được sử dụng rộng rãi trong các lĩnh vực xe hơi, sản xuất công nghiệp, dân dụng, văn phòng, môi trường, ... Cảm biến quang là một loại cảm biến dựa trên các tính chất đặc trưng của ánh sáng như tính chất hạt và sóng. Một trong những ứng dụng quan trọng của cảm biến quang là cảm biến vân tay dùng để nhận dạng vân tay, được sử dụng rộng rãi trong các khóa cửa vân tay, bảo mật máy tính xách tay, máy chấm công, các thiết bị trong khoa học hình sự, ... nhờ tính bảo mật và tiện lợi cao. Nghiên cứu và ứng dụng nhận dạng vân tay vào cuộc sống là một trong những xu hướng quan trọng và có thực tiễn cao. Vấn đề này đã được thế giới tiếp cận rất sớm và đã thu được những thành tựu đáng chú ý, tuy nhiên nó vẫn còn tương đối mới mẻ ở Việt Nam.

Nhóm Cảm biến vân tay gồm 9 thành viên của lớp ĐH Cơ điện tử - K2, ĐH Công nghiệp Hà Nội (DS cụ thể ở phần phụ lục 2) được sự giúp đỡ của thầy Nhữ Quý Thor đã nhận và thực hiện đề tài “**Cảm biến nhận dạng vân tay**”. Do nhiều nguyên nhân chủ quan và khách quan nên nhóm Cảm biến đã thu được một số kết quả quan trọng trong việc nghiên cứu và nhận dạng vân tay, nhưng vẫn còn nhiều hạn chế và thiếu sót của đề tài mà nhóm Cảm biến vẫn chưa khắc phục được. Bài báo cáo này mới chỉ tập trung vào việc nhận dạng vân tay bằng phương pháp trích điểm đặc trưng Minutiae của vân tay, còn nhiều vấn đề cần phải giải quyết để hoàn thiện đề tài này, vì vậy nhóm Cảm biến xin cảm ơn và mong muốn sự chung tay góp sức của các thành viên trong nhóm và sự giúp đỡ của các thầy cô giáo cùng các bạn.

Hà Nội, tháng 5 năm 2010

MỤC LỤC

Lời nói đầu	1
Mục lục	2
Chương 1 – GIỚI THIỆU CHUNG	4
1. Công nghệ sinh trắc và vấn đề bảo mật	4
2. Lịch sử nhận dạng vân tay	4
3. Tình hình ứng dụng công nghệ nhận dạng vân tay trên thế giới và Việt Nam	6
Chương 2 – NHẬN DẠNG VÂN TAY	7
1. Hệ thống nhận dạng vân tay	7
2. Cơ sở nhận dạng và phân loại vân tay	9
a. Cơ sở nhận dạng vân tay	9
b. Phân loại vân tay	12
3. Phương pháp nhận dạng vân tay	13
a. Thuật toán xử lý ảnh	13
b. Phương pháp trích các điểm đặc trưng	13
c. Nâng cao chất lượng ảnh	19
d. Phát hiện điểm đặc trưng	23
e. Lọc điểm đặc trưng bị lỗi	25
f. Đối sánh vân tay	26
Chương 3 – CHƯƠNG TRÌNH NHẬN DẠNG VÂN TAY	29
1. Giới thiệu chung về chương trình	29
a. Phần mềm	29
b. Phần cứng	30
2. Lưu đồ giải thuật	34
3. Giao diện chương trình	36
4. Hướng dẫn sử dụng chương trình	36
a. Lấy ảnh vân tay từ thiết bị <i>U.are.U 4500 Fingerprint Reader</i>	37
b. Lấy mẫu (Tạo cơ sở dữ liệu)	38
c. Xử lý vân tay cần nhận dạng	39
d. Thực hiện nhận dạng vân tay	40
Chương 4 – KẾT LUẬN	44

1. Đánh giá và hướng phát triển của đề tài	44
a. Đánh giá kết quả đề tài	44
b. Hướng phát triển của đề tài	44
2. Lời kết	45
Phụ lục	46
1. Hàm chính GDNDVT.m	46
2. Danh sách nhóm Cảm biến nhận dạng vân tay	46
Tài liệu tham khảo	54

CHƯƠNG I

GIỚI THIỆU CHUNG

1. Công nghệ sinh trắc và vấn đề bảo mật

Trong thời đại ngày nay, sự phát triển không ngừng của khoa học kỹ thuật đã giúp cho con người thuận tiện hơn trong các công việc hằng ngày. Với sự bùng nổ về công nghệ thông tin, quá trình toàn cầu hóa diễn ra nhanh chóng, sự bảo mật riêng tư thông tin cá nhân cũng như để nhận biết một người nào đó trong hàng tỉ người trên trái đất đòi hỏi phải có một tiêu chuẩn, hệ thống đảm nhận các chức năng đó. Công nghệ sinh trắc ra đời và đáp ứng được các yêu cầu trên.

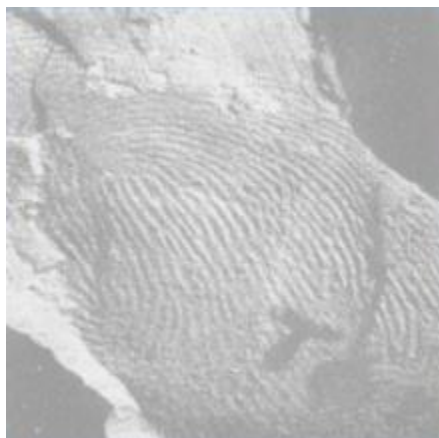
Nhiều công nghệ sinh trắc đã và đang được phát triển, một số chúng đang được sử dụng trong các ứng dụng thực tế và phát huy hiệu quả cao. Các đặc trưng sinh trắc thường được sử dụng là vân tay, gương mặt, móng mắt, tiếng nói. Mỗi đặc trưng sinh trắc có điểm mạnh và điểm yếu riêng, nên việc sử dụng đặc trưng sinh trắc cụ thể là tùy thuộc vào yêu cầu của mỗi ứng dụng nhất định. Các đặc trưng sinh trắc có thể được so sánh dựa vào các yếu tố sau: tính phổ biến, tính phân biệt, tính ổn định, tính thu thập, hiệu quả, tính chấp nhận. Trong yêu cầu về bảo mật và tìm kiếm, tính phân biệt (hai người khác nhau thì đặc trưng sinh trắc này phải khác nhau) và ổn định (đặc trưng sinh trắc này không thay đổi theo từng giai đoạn thời gian tương ứng với hạng mục đối sánh nhất định) được quan tâm nhiều hơn cả. Vân tay đã được biết tới với tính phân biệt (tính chất cá nhân) và ổn định theo thời gian cao nhất, vì vậy nó là đặc trưng sinh trắc được sử dụng rộng rãi nhất. Nhận dạng sinh trắc đề cập đến việc sử dụng các đặc tính hành vi và thể chất (ví dụ: vân tay, gương mặt, chữ kí...) có tính chất khác biệt để nhận dạng một người một cách tự động. Nhận dạng vân tay được xem là một trong những kỹ thuật nhận dạng hoàn thiện và đáng tin cậy nhất.

Trong các tổ chức, cơ quan an ninh, quân sự, hành chính, khoa học... luôn có nhu cầu kiểm tra và trả lời các câu hỏi: “người này có phải là đối tượng đó hay không?”, “người này có được quyền truy cập và sử dụng thiết bị đó?”, “người này có được biết những thông tin đó?”... Phương pháp dựa vào thẻ bài truyền thống (ví dụ dùng chìa khóa...), phương pháp dựa vào trí thức (ví dụ dùng mật khẩu và PIN – Personal Identification Number) đã được sử dụng phổ biến nhưng thực tế đã chứng minh là không hiệu quả vì tính an toàn không cao và khó nhớ. Người ta nhận thấy các đặc trưng sinh trắc không thể dễ dàng bị thay thế, chia sẻ hay giả mạo..., chúng được xem là đáng tin cậy hơn trong nhận dạng một người so với các phương pháp trên. Vân tay là một trong những đặc điểm khá đặc biệt của con người bởi vì tính đa dạng của nó, mỗi người sở hữu một dấu vân tay khác nhau, rất ít trường hợp những người có dấu vân tay trùng nhau. Bằng việc sử dụng vân tay và mật mã, việc xác nhận một người có thể được thực hiện bằng một hệ thống nhận dạng vân tay an toàn và nhanh chóng.

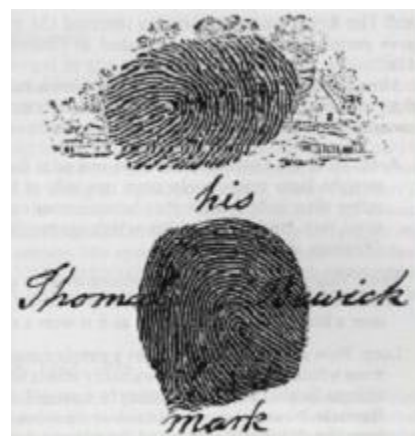
2. Lịch sử nhận dạng vân tay

Từ xa xưa, con người đã nhận ra tính cá nhân của vân tay nhưng chưa có bất kỳ một cơ sở khoa học nào. Đến thế kỷ 16, các kỹ thuật vân tay khoa học hiện đại mới xuất hiện và từ đó các lý thuyết và chương trình mô tả, nhận dạng vân tay mới phát triển mau chóng. Năm 1664,

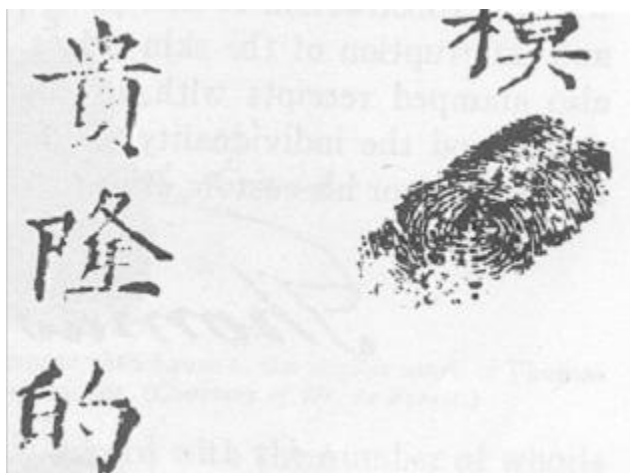
Nehemiah Grew nhà sinh thái học thực vật xuất bản những trang sách đầu tiên các nghiên cứu có tính hệ thống của ông về vân tay. Năm 1788, Mayer đã mô tả chi tiết thông tin giải phẫu của vân tay để đặc tính hóa, nhận dạng các đặc tính vân tay. Năm 1809, Thomas Bewick bắt đầu sử dụng vân tay của mình như là biểu tượng đăng kí thương mại – đã tạo ra một cột mốc quan trọng trong nghiên cứu khoa học về nhận dạng vân tay. Năm 1880, Henry Fauld đã đưa ra giả thuyết khoa học khẳng định tính cá nhân của vân tay dựa vào các nhận thức kinh nghiệm. Năm 1888, Francis Galton giới thiệu các đặc trưng chi tiết phục vụ cho đối sánh vân tay.



Vân tay trên đèn của người Paléttin (400 A.D)



Con dấu thương mại của Berwick (1809)



Chữ kí bằng vân tay của người Trung Quốc khi buôn bán (1839)

Hình 1: Một số bằng chứng vân tay tìm được thời xưa

Đầu thế kỉ 20, cấu trúc của vân tay mới được mô tả một cách khá đầy đủ. Các nguyên lý sinh học của vân tay được tổng kết như sau: Biểu bì vân có các đặc tính khác nhau trên các vân tay khác nhau (nguyên lý này là cơ sở cho nhận dạng vân tay); cấu hình vân tay có sự thay đổi trên từng cá nhân, nhưng sự thay đổi nhỏ này vẫn cho phép phân loại một cách có hệ thống các vân tay

(nguyên lý này là cơ sở để tiến hành phân loại vân tay); các chi tiết và cấu hình của mỗi đường vân là ổn định và không thay đổi

Cũng từ đầu thế kỉ 20, nhận dạng vân tay chính thức được chấp nhận như một phương pháp nhận dạng cá nhân có giá trị và trở thành chuẩn trong pháp luật. Ví dụ, năm 1924 FBI đã thiết lập một cơ sở dữ liệu có 810.000 thể vân tay.

3. Tình hình ứng dụng công nghệ nhận dạng vân tay trên thế giới và Việt Nam hiện nay

Hơn 100 năm qua so sánh dấu vân tay vốn được coi là một phương tiện hữu hiệu hỗ trợ cho các nhà điều tra trong quá trình phá án và xét xử. Người ta có thể tìm ra tung tích tội phạm cũng như nạn nhân thông qua dấu vân tay ở trên hiện trường. Tuy nhiên phương pháp này vẫn bộc lộ một vài khuyết điểm do tác động của các yếu tố khách quan như môi trường thời tiết, hiện trường sau khi khảo sát,... và các yếu tố chủ quan gây nhiễu. Nếu chỉ đơn thuần dựa vào yếu tố kỹ thuật mà bỏ qua một loạt các biện pháp nghiệp vụ khác, sai số này có thể lên tới 10%. Mặc dù vậy, phương pháp nhận dạng vân tay hiện vẫn còn phổ biến ở nhiều nơi và nhiều quốc gia, mặc nhiên phương pháp nhận dạng vân tay vẫn được sử dụng trong việc điều tra phá án của cảnh sát vì thế việc nâng cao sự chính xác khi nhận dạng vân tay là một vấn đề thiết yếu.

Ngày nay, người ta cũng lợi dụng các đặc điểm riêng biệt của vân tay để xây dựng các hệ thống bảo mật các thông tin riêng tư cho người sở hữu chúng, từ việc dùng các ổ khóa vân tay thay thế cho các ổ khóa thông thường cho đến việc dùng vân tay thay thế mật khẩu đã quá phổ biến trong thời đại công nghệ thông tin. Người ta chỉ cần quét dấu vân tay của mình qua các thiết bị chức năng là có thể mở được một cánh cửa, đăng nhập vào hệ thống máy vi tính, qua một phòng bí mật hay các trạm bảo vệ bí mật. Đó là giải pháp an ninh tuyệt đối cho những yêu cầu bảo mật của con người trong nhiều lĩnh vực như: Kiểm soát an ninh trong các cơ quan của Chính phủ, trong quân đội, ngân hàng, trung tâm lưu trữ dữ liệu... hoặc để kiểm soát ra vào của nhân viên tại các trung tâm thương mại, các tập đoàn, các đại sứ quán...

Trong lĩnh vực quản lý nhân sự, phương pháp nhận dạng vân tay còn hỗ trợ đắc lực cho việc quản lý và chấm công tại các nhà máy, xí nghiệp, công ty bằng máy các máy chấm công vân tay. Tuy nhiên, phổ biến nhất có lẽ là dấu vân tay của chúng ta qua mặt sau của chứng minh thư để xác định một cách nhanh nhất các đặc điểm, hồ sơ của một công dân đã được lưu trong cơ sở dữ liệu.

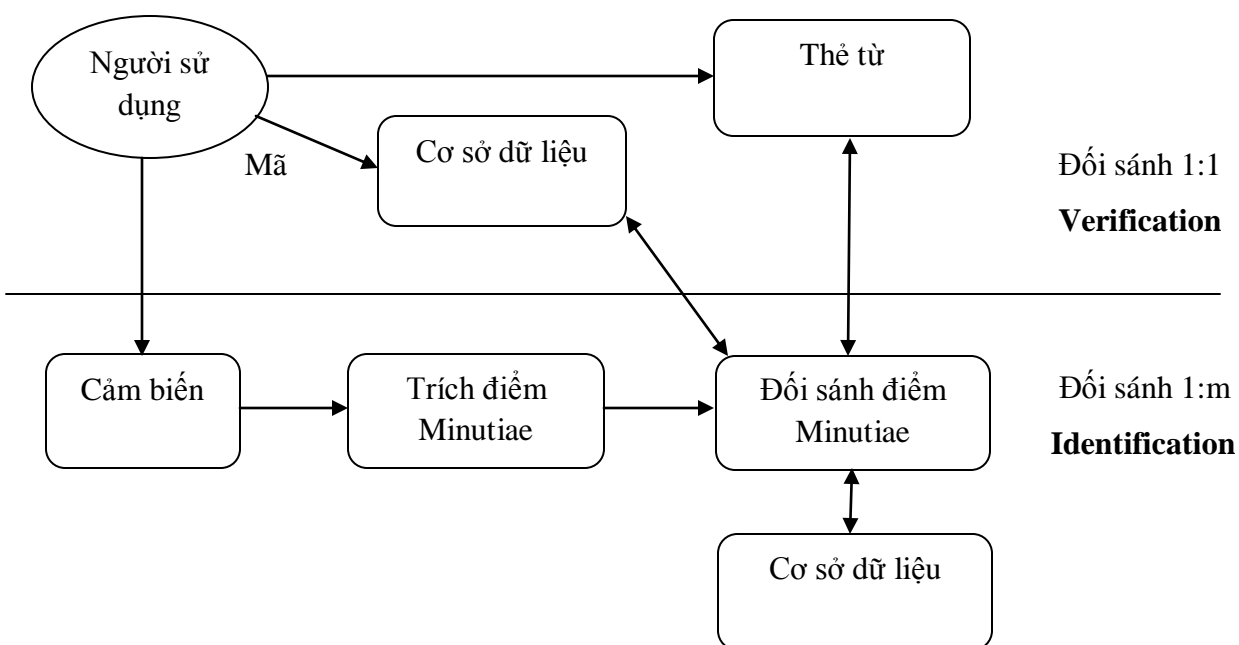
Trên thế giới hiện nay đã xuất hiện nhiều sản phẩm công nghệ cao sử dụng phương pháp nhận dạng vân tay như khóa vân tay, máy chấm công vân tay, máy tính xác tay,... Tuy nhiên đây vẫn là vấn đề còn chưa được nghiên cứu nhiều ở Việt Nam. Ở nước ta, phương pháp này mới chỉ phổ biến ở việc quản lý nhân sự thông qua chứng minh thư nhân dân và phục vụ điều tra phá án. Các sản phẩm công nghệ cao nói trên chúng ta vẫn phải nhập khẩu với giá thành khá cao, do đó chúng vẫn chưa được phổ biến rộng rãi.

CHƯƠNG II

NHẬN DẠNG VÂN TAY

1. Hệ thống nhận dạng vân tay

Hệ thống nhận dạng: là hệ thống xác thực một cá nhân bằng cách tìm kiếm và đối sánh đặc tính sinh trắc của người này với toàn bộ các mẫu sinh trắc được lưu giữ trong cơ sở dữ liệu. Hình 2 là cấu trúc cơ bản của một hệ thống nhận dạng vân tay.



Hình 2: Cấu trúc cơ bản của hệ thống nhận dạng vân tay

Hệ thống này gồm 2 phần:

- Verification (Xác nhận dấu vân tay): Đầu tiên một người sẽ cung cấp dấu vân tay cùng với thông tin hoặc đặc điểm cá nhân của người đó như họ tên, ngày sinh, quê quán... (trong chứng minh thư) hoặc là Username, tên tài khoản, các quyền hạn của người đó,...(trong bảo mật). Bước này nhằm tạo ra một cơ sở dữ liệu tương ứng dấu vân tay và các đặc điểm liên quan. Nguyên lý cơ bản của hệ thống này là sử dụng các diot phát sáng để truyền các tia gần hồng ngoại (Near Infrared NIR) tới ngón tay và chúng sẽ được hấp thụ lại bởi hồng cầu trong máu. Vùng các tia bị hấp thụ trở thành vùng tối trong hình ảnh và được chụp lại bởi camera CCD. Sau đó, hình ảnh được xử lý và tạo ra mẫu vân tay. Mẫu vân tay được chuyển đổi thành tín hiệu số và là dữ liệu để nhận dạng người sử dụng chỉ trong vòng chưa đến 2 giây. Công nghệ truyền ánh sáng của Hitachi cho phép ghi lại rõ nét sơ đồ vân nhờ độ tương phản cao và khả năng tương thích với mọi loại da tay, kể cả da khô, da dầu hay có vết bẩn, vết nhăn hoặc bị khiếm khuyết do tạo hoá trên bề mặt của các ngón tay. Lượng dữ liệu nhỏ đó là căn cứ cho việc nhận dạng và tạo nên một hệ thống nhỏ gọn, an toàn, thân thiện và nhanh nhất trên thế giới. Hệ thống này có thể lưu trữ từ

6.000 - 8.000 ngón tay trong một máy và mỗi người có thể được nhận dạng bởi 1 trong 5 ngón tay khác nhau đã đăng ký trước đó. Ưu điểm vượt trội của hệ thống này là chỉ tương tác với cơ thể sống nên việc bắt chước, giả mạo hoặc ăn cắp dữ liệu là điều hoàn toàn bất khả thi. FVB ra đời hồi đầu năm 2006, đã nhanh chóng thành công tại thị trường Nhật Bản, Singapor, Trung Quốc... Hiện nay, trên thị trường thế giới đã có bán nhiều loại thiết bị chụp vân tay (fingerprint reader, fingerprint scanner) với các chất lượng khác nhau. Một số ảnh vân tay được chụp từ các thiết bị này trong hình 3.



Hình 3: Ảnh vân tay được chụp từ các thiết bị tương ứng

a) Biometrika FX2000; b) Digital Persona UareU2000; c) Identix DFR200; d) Ethentica TactilSense T-FPM; e) STMicroelectronics TouchChip TCS1AD; f) Veridicom FPS110; g) Atmel FingerChip AT77C101B; h) Authentec AES4000.

- **Identification (Nhận diện dấu vân tay):** Dấu vân tay sẽ được đưa thu thập từ một sensor để đối chiếu với database chứa các vân tay để truy ra các đặc điểm muốn truy xuất. Việc đối sánh ảnh vân tay cần nhận dạng chỉ cần được tiến hành trên các vân tay (có trong cơ sở dữ liệu) thuộc loại đã được xác định nhờ quá trình phân loại. Đây là giai đoạn quyết định xem hai ảnh vân tay có hoàn toàn giống nhau hay không và đưa ra kết quả nhận dạng, tức là ảnh vân tay cần nhận dạng tương ứng với vân tay của cá thể nào đã được lưu trữ trong cơ sở dữ liệu.

Để đánh giá một hệ thống nhận dạng vân tay ta cần phân tích hai loại lỗi đó là: Lỗi từ chối nhầm (False Reject Rate: FRR) và lỗi chấp nhận nhầm (False Accept Rate: FAR)

$$FRR = \frac{\text{Số lỗi từ chối nhầm của các vân tay khác nhau}}{\text{Tổng số lần đối sánh của các vân tay khác nhau}}$$

$$FAR = \frac{\text{Số lỗi chấp nhận nhầm của các vân tay khác nhau}}{\text{Tổng số lần đối sánh của các vân tay khác nhau}}$$

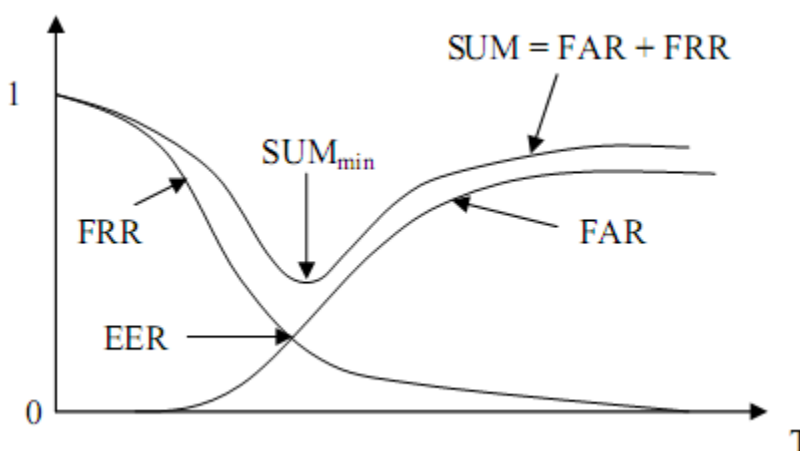
Giá trị của hai loại lỗi này có mối quan hệ với nhau thông qua giá trị ngưỡng đối sánh T (threshold) là sai lệch cho phép giữa mẫu cần đối sánh với mẫu được lưu trong cơ sở dữ liệu. Khi chọn giá trị ngưỡng thấp thì lỗi từ chối nhầm sẽ tăng, lỗi chấp nhận nhầm sẽ giảm và ngược lại.

Hệ thống thường được đánh giá theo hai cách:

- Tỷ lệ cực tiểu $SUM_{\min} = (FAR + FRR)_{\min}$: Theo quan điểm dù là loại lỗi gì cũng là lỗi, do đó tỷ lệ lỗi cực tiểu SUM_{\min} là hệ số lỗi nhỏ nhất mà hệ thống có thể đạt được.

- Mức độ lỗi cân bằng (Equal Error Rate: EER): Đó là điểm mà FAR và FRR bằng nhau.

Mối quan hệ giữa FAR, FRR, SUM và EER theo ngưỡng T được thể hiện trong hình 4.



Hình 4: Mối quan hệ giữa FAR, FRR, SUM và EER theo ngưỡng T

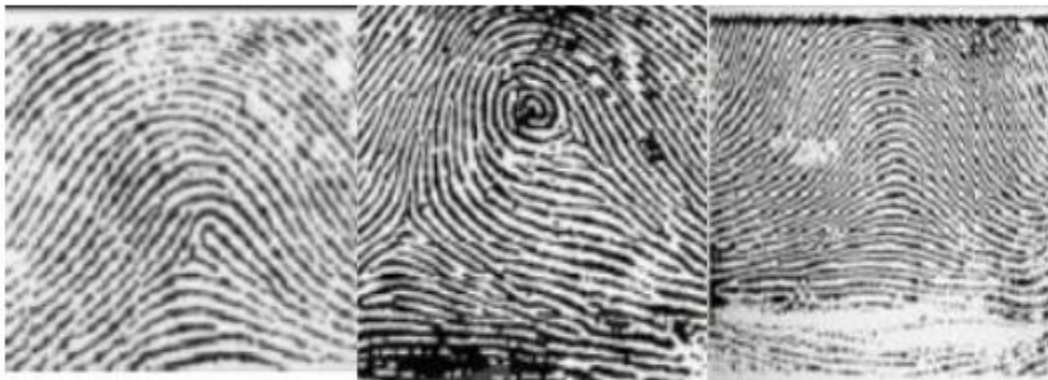
2. Cơ sở nhận dạng và phân loại vân tay

a. Cơ sở nhận dạng vân tay

Như đã nói ở trên, cơ sở nhận dạng vân tay là những đặc điểm riêng biệt trong cấu tạo của các vân tay khác nhau. Dấu vân tay của mỗi cá nhân là độc nhất. Xác suất hai cá nhân - thậm chí ngay cả anh em (hoặc chị em) sinh đôi cùng trứng - có cùng một bộ dấu vân tay là 1 trên 64 tỉ. Ngay cả các ngón trên cùng bàn tay cũng có vân khác nhau. Dấu vân tay của mỗi người là không đổi trong suốt cuộc đời. Người ta có thể làm phẫu thuật thay da ngón tay, nhưng chỉ sau một thời gian dấu vân tay lại được hồi phục như ban đầu.

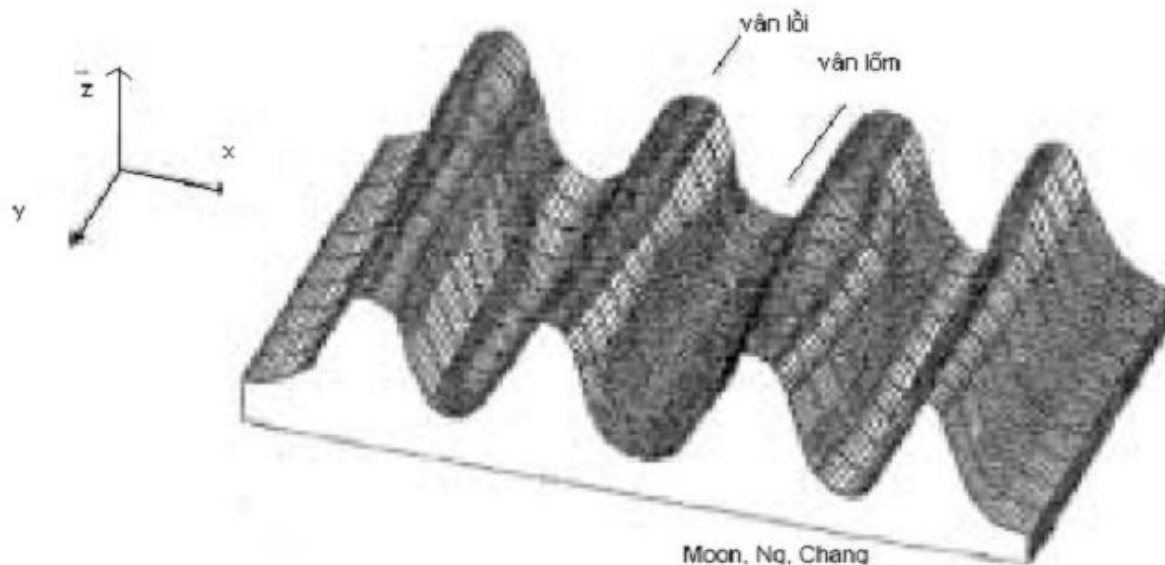
Vân tay là những đường có dạng dòng chảy có trên ngón tay người. Nó là một tham số sinh học bất biến theo tuổi tác đặc trưng cho mỗi cá thể. Cấu trúc dễ nhận thấy nhất của vân tay là các vân lồi (đường gợn - ridge) và vân lõm (luống - valley); trong ảnh vân tay, vân lồi có màu tối trong khi vân lõm có màu sáng. Vân lồi có độ rộng từ 100 μm đến 300 μm . Độ rộng của một cặp vân

lồi lõm cạnh nhau là 500 μm . Các chấn thương như bong nhẹ, mòn da.. không ảnh hưởng đến cấu trúc bên dưới của vân, khi da mọc lại sẽ khôi phục lại đúng cấu trúc này. Vân lồi và vân lõm thường chạy song song với nhau; chúng có thể rẽ thành hai nhánh, hoặc kết thúc.



Hình 5: Vân tay

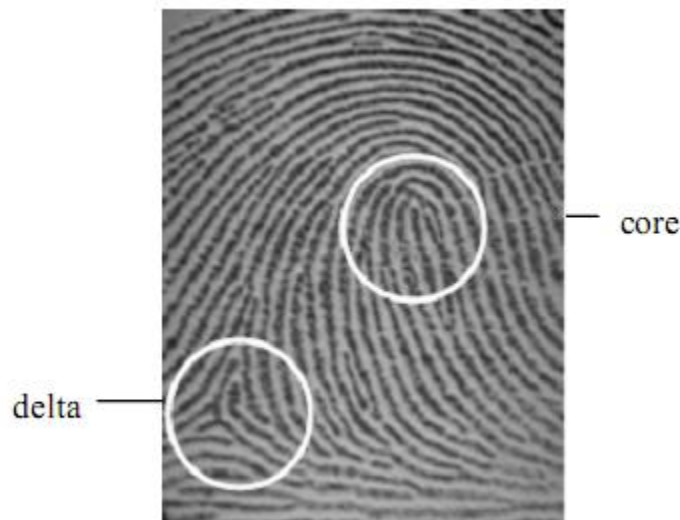
Hình ảnh vân tay thường được biểu diễn như là một bề mặt hai chiều. Kí hiệu I là ảnh vân tay cấp xám với cấp xám g . $I[x,y]$ là cấp xám của điểm ảnh $[x,y]$. Kí hiệu $z = S(x,y)$ là bề mặt rời rạc tương ứng với ảnh I : $S(x,y) = I[x,y]$. Bằng cách chọn các điểm ảnh màu sáng có cấp xám là 0, và các điểm ảnh có màu tối có cấp xám là $g-1$, thì các đường vân (xuất hiện có màu tối trong I tương ứng với bề mặt vân lồi còn khoảng không gian giữa các vân lồi (có màu sáng) tương ứng là bề mặt vân lõm.



Hình 6: Bề mặt S của một vùng vân tay

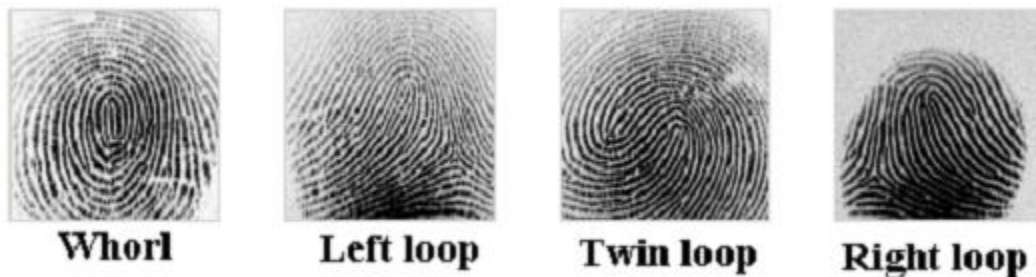
Trên các ảnh vân tay có các điểm đặc trưng (là những điểm đặc biệt mà vị trí của nó không trùng lặp trên các vân tay khác nhau) được phân thành hai loại: Singularity và Minutiae.

- **Singularity:** Trên vân tay có những vùng có cấu trúc khác thường so với những vùng bình thường khác (thường có cấu trúc song song), những vùng như vậy gọi là Singularity. Có hai loại Singularity là Core và Delta.



Hình 7: Điểm Core và Delta

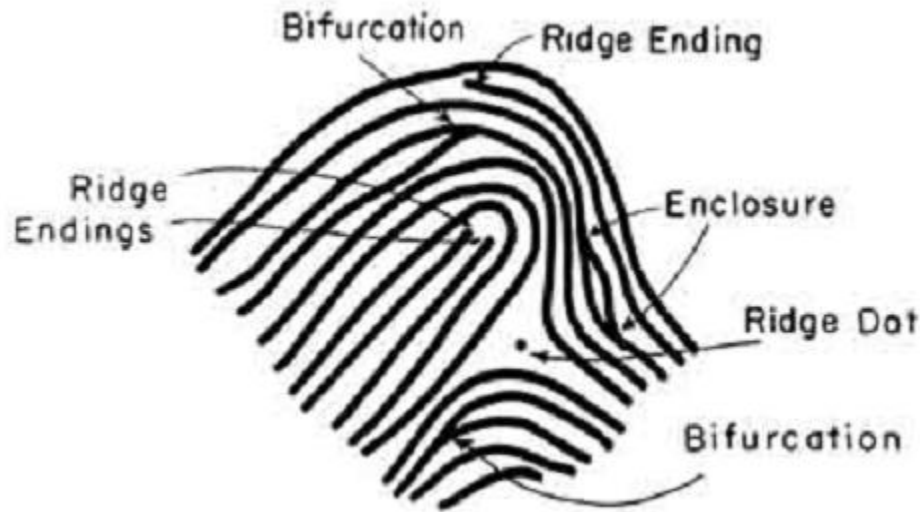
Điểm Core thường có một số dạng như sau:



Hình 8: Một số loại Core thường gặp

- **Minutiae:** Khi dò theo từng đường vân ta sẽ thấy có những điểm đường vân kết thúc (Ridge Ending) hoặc rẽ nhánh (Bifurcation), những điểm này được gọi chung là Minutiae.

Phương pháp trích các điểm Singularity và Minutiae được trình bày ở phần **3.b – Phương pháp trích các điểm đặc trưng**.



Hình 9: Các điểm Minutiae Ridge Ending (điểm kết thúc) và Bifurcation (điểm rẽ nhánh)

b. Phân loại vân tay

Các ảnh vân tay được phân loại nhằm tăng tốc độ tìm kiếm trong cơ sở dữ liệu vân tay trong quá trình nhận dạng. Và các phương pháp phân loại vân tay hiện nay đều dựa trên hai loại đặc điểm chung nhất của mọi vân tay, đó là tâm (hay điểm nhân) và tam phân điểm (hay còn được gọi là delta).

Dựa trên các thông tin về số lượng tam phân điểm và vị trí của chúng là ta hoàn toàn có thể xác định được loại của vân tay. Vì vậy, vấn đề trích chọn tâm và tam phân điểm là khâu không thể thiếu được trong quá trình phân loại vân tay.

Sau đây là một số phương pháp phân loại vân tay đã được nghiên cứu và công bố có sử dụng phương pháp trích chọn tâm và tam phân điểm.

- Phương pháp phân loại Henry: Đây là phương pháp phân loại cổ điển và phổ biến nhất, được sử dụng chủ yếu khi nhận dạng vân tay một cách thủ công. Henry đã định nghĩa điểm tâm là “điểm nằm về phía bắc nhất của đường vân nằm trong cùng nhất”. Thực tế, điểm tâm là điểm trung tâm của vùng loop nằm về phía bắc nhất. Nếu vân tay không chứa các vùng loop hay whorl thì điểm tâm là điểm mà tại đó độ cong của đường vân là lớn nhất. Các tâm và tam phân điểm được nhận biết bằng mắt thường và vân tay được phân loại dựa trên số lượng đường vân bị cắt bởi đường nối tâm và tam phân điểm.

- Các phương pháp phân loại dựa trên các đặc điểm tổng thể: Việc phân loại vân tay trong phần lớn các hệ AFIS hiện nay đều dựa trên các đặc điểm tổng thể. Việc trích chọn tâm và tam phân điểm có thể được thực hiện trực tiếp trên ảnh vân tay theo phương pháp xử lý ảnh theo từng điểm, nhưng nhược điểm của phương pháp này là tốc độ xử lý chậm. Sau khi tách hướng các vùng, ta nhận được một ảnh định hướng đặc trưng cho vân tay.

- + Phương pháp 2: Mầu phân bố hướng chuẩn được định nghĩa là một mẫu hai chiều mô tả phân bố của các hướng lần xung quanh một điểm đặc trưng. Bằng nghiên cứu thống kê trên nhiều vân

tay, các tác giả đã định nghĩa đặc trưng tâm tam phân điểm bằng các mẫu phân bố hướng chuẩn. Việc trích chọn tâm và tam phân điểm được qui về việc tìm kiếm trên ảnh định hướng các vectơ phân bố hướng có dạng giống với mẫu phân bố hướng chuẩn bằng các đối sánh các mẫu phân bố hướng tại các điểm có khả năng là đặc điểm với các mẫu phân bố hướng chuẩn.

+ Phương pháp 3: Hướng của các vùng được lượng tử hóa theo 8 hướng trong khoảng từ 0° đến 180° . Các vùng đặc điểm tâm và tam phân điểm được định vị trên ảnh định hướng bằng cách kiểm tra chỉ số Poincare trên một đường cong nhỏ khép kín xung quanh một điểm. Chỉ số Poincare được xác định ở mục tiếp theo.

3. Phương pháp nhận dạng vân tay

Hai phương pháp nhận dạng vân tay thường được sử dụng là:

- Phương pháp 1: Dựa vào các đặc tính cụ thể của dấu vân tay, như điểm cuối, điểm rẽ nhánh của các vân trên tay.

- Phương pháp 2: So sánh toàn bộ đặc tính của dấu vân tay.

Thực tế đây là hai mức độ của nhận dạng và dễ thấy rằng phương pháp 2 đã bao gồm phương pháp 1. Tuy nhiên do đặc điểm của vân tay, nếu ta không phải so sánh quá nhiều (cơ sở dữ liệu không quá lớn) các đặc điểm đặc biệt trên dấu vân tay đủ để ta nhận dạng ra dấu vân tay đó của ai. Phương pháp 2 là một công việc phức tạp đòi hỏi tính toán nhiều nhưng tất nhiên là cho kết quả với độ tin cậy cao hơn. Do trình độ và điều kiện hoàn cảnh không cho phép nên nhóm cảm biến quyết định sử dụng phương pháp thứ nhất và tập trung vào việc trích chọn điểm Minutiae để nhận dạng vân tay.

a. Thuật toán xử lý ảnh

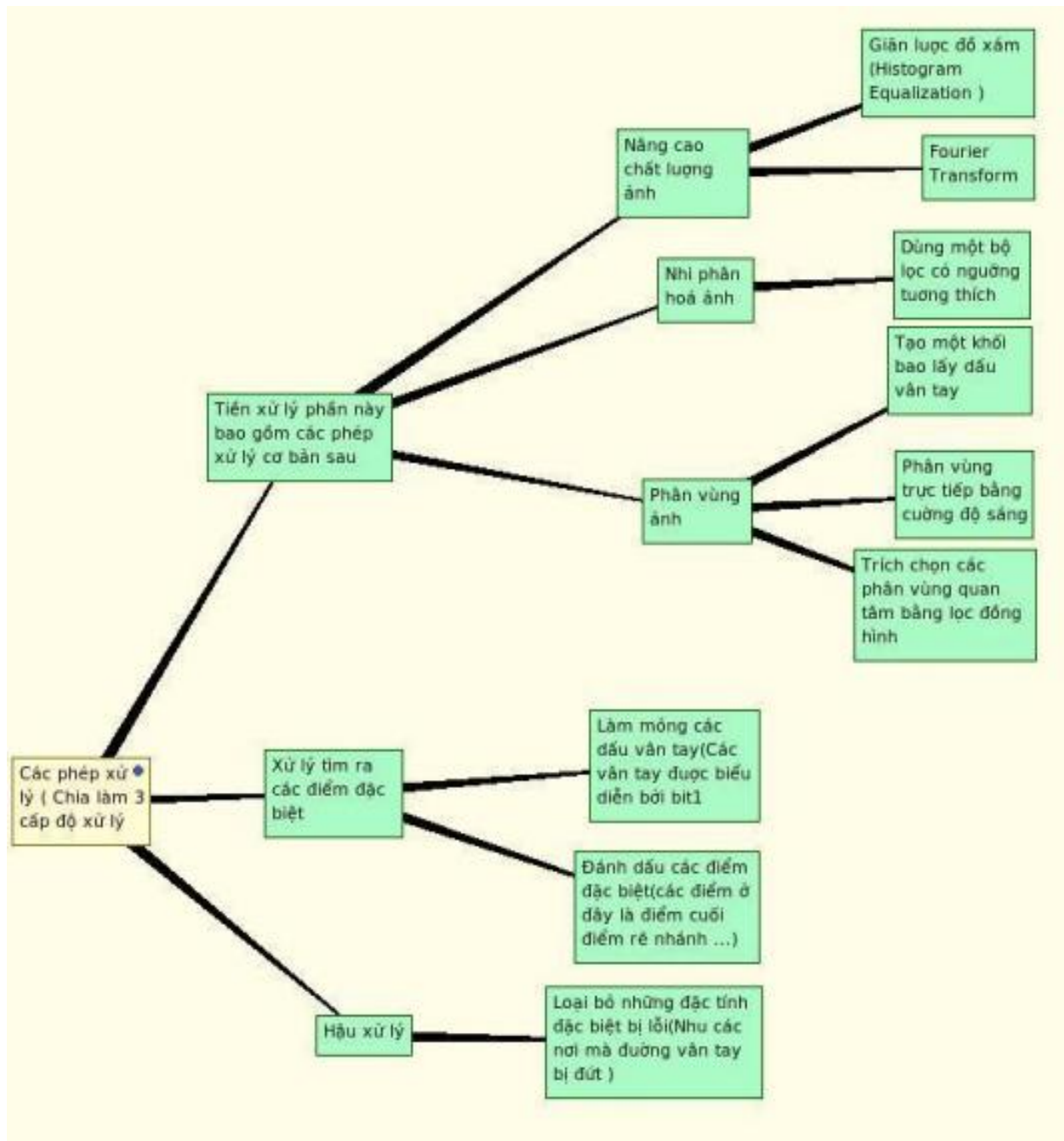
Quá trình nhận dạng vân tay thực chất là quá trình xử lý ảnh vân tay. Thuật toán xử lý ảnh vân tay được thể hiện trên hình 10. Do điều kiện chuyên môn nên bài báo cáo này không đi sâu vào môn xử lý ảnh mà chỉ tập trung vào việc ứng dụng nó vào việc trích được các điểm đặc trưng từ một bức ảnh dấu vân tay.

b. Phương pháp trích các điểm đặc trưng

*** Trường định hướng (orientation field)**

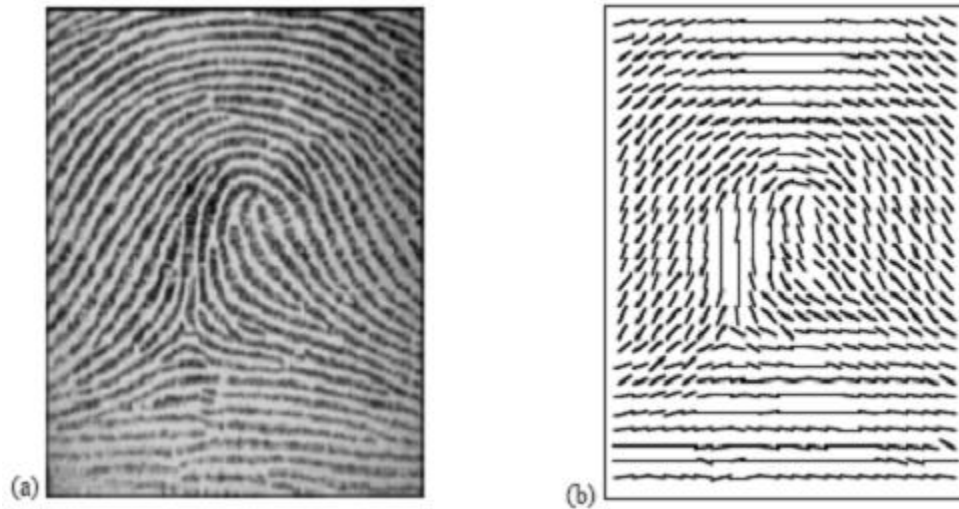
Ảnh vân tay là ảnh định hướng, các đường vân là các đường cong theo các hướng xác định. Góc định hướng θ_{xy} hợp bởi phương của một điểm (x,y) trên đường vân với phương ngang được gọi là hướng của điểm đó, nó nằm trong đoạn $[0^\circ;180^\circ]$. Thay vì tính góc định hướng tại mỗi điểm ảnh, hầu hết các phương pháp trích chọn đặc trưng và xử lý vân tay ước lượng góc định hướng tại các vị trí rời rạc (để làm giảm gánh nặng tính toán và cho phép thực hiện các ước lượng còn lại nhờ phép nội suy). Tập hợp các hướng của các điểm trên ảnh vân tay gọi là trường định hướng của ảnh vân tay đó (xem hình 11). Ảnh định hướng vân tay là một ma trận D mà mỗi phần tử mang thông tin về góc định hướng của các đường vân. Mỗi phần tử θ_{ij} , tương ứng với nốt $[i,j]$ trong lưới ô vuông chứa điểm ảnh $[x_i, y_j]$, biểu diễn hướng trung bình của đường vân trong lân cận của $[x_i, y_j]$. Người ta thêm vào một giá trị r_{ij} liên kết với θ_{ij} để biểu diễn tính tin cậy (hay toàn

vény) của hướng. Giá trị của r_{ij} là nhỏ ở các vùng bị nhiễu và hư hại, có giá trị lớn ở các vùng có chất lượng tốt (xem hình 12).

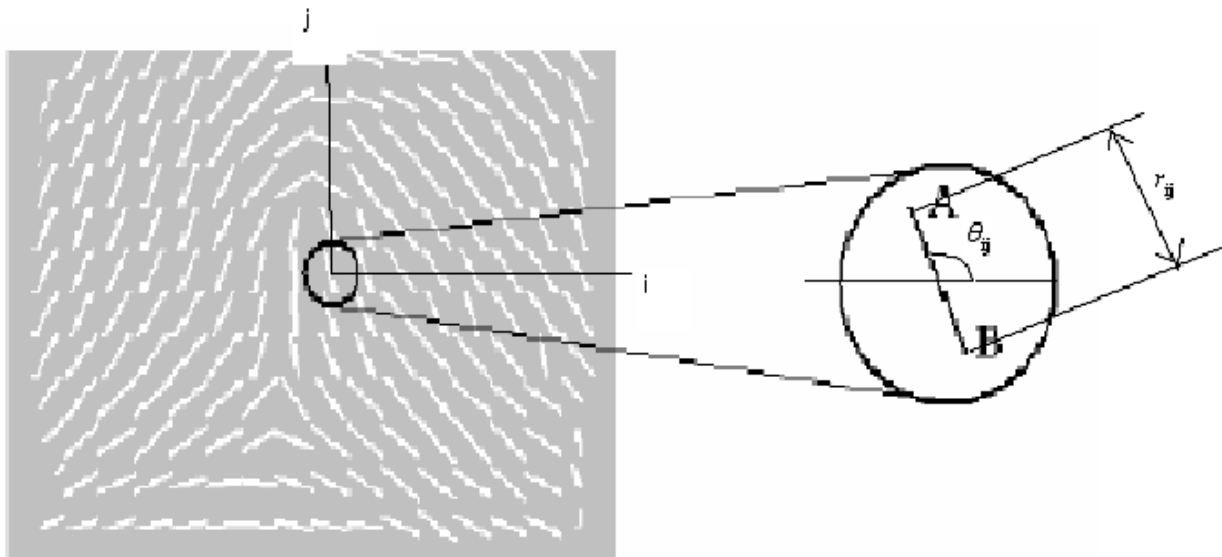


Hình 10: Thuật toán xử lý ảnh vân tay

Để tính góc định hướng, phương pháp đơn giản nhất là tính toán gradient trên ảnh vân tay. Gradient $\nabla(x_i, y_j)$ ở điểm $[x_i, y_j]$ của I là một véc tơ hai chiều $[\nabla_x(x_i, y_j), \nabla_y(x_i, y_j)]$ trong đó thành phần ∇_x và ∇_y là đạo hàm theo x và y của I tại điểm $[x_i, y_j]$ tương ứng với hướng x và y . Góc pha gradient biểu thị hướng thay đổi mật độ điểm ảnh lớn nhất. Vì vậy, hướng θ_{ij} của một góc giả định qua vùng có tâm tại $[x_i, y_j]$ là trực giao với góc pha gradient tại $[x_i, y_j]$.



Hình 11: Ảnh vân tay và trường định hướng của nó



Hình 12: Một ảnh định hướng vân tay được tính trên một lưới 16x16

Ratha, Chen và Jain (1995) đã tính góc định hướng θ_{ij} bằng cách kết hợp nhiều ước lượng gradient trong một cửa sổ 17x17 có tâm tại $[x_i, y_j]$.

$$\theta_{ij} = 90 + \frac{1}{2} \arctan \left[\frac{2G_{xy}}{G_{xx} - G_{yy}} \right]$$

$$G_{xy} = \sum_{h=-8}^8 \sum_{k=-8}^8 \nabla_x(x_i + h, y_j + k) \nabla_y(x_i + h, y_j + k)$$

$$G_{xx} = \sum_{h=-8}^8 \sum_{k=-8}^8 (\nabla_x (x_i + h, y_j + k))^2$$

$$G_{yy} = \sum_{h=-8}^8 \sum_{k=-8}^8 (\nabla_y (x_i + h, y_j + k))^2$$

Trong đó, ∇_x và ∇_y là các thành phần gradient theo hướng x và y được tính qua mặt nạ Sobel.

*** Xác định các điểm Singularity bằng chỉ số Poincare (Poincare index)**

Giả sử (i,j) là một điểm bất kỳ trên ảnh vân tay, C là một đường cong khép kín xung quanh (i,j) thì chỉ số Poincare tại (i,j) là tổng đại số các độ sai lệch hướng của các điểm liên tiếp nhau trên đường cong C.

$$Poincare(i, j) = \sum_{k=0}^{N_p-1} \Delta(k)$$

$$\Delta(k) = \begin{cases} d(k) & |d(k)| < \pi/2 \\ d(k) + \pi & d(k) \leq -\pi/2 \\ d(k) - \pi & \end{cases}$$

$$d(k) = \theta(x_{k+1}, y_{k+1}) - \theta(x_k, y_k)$$

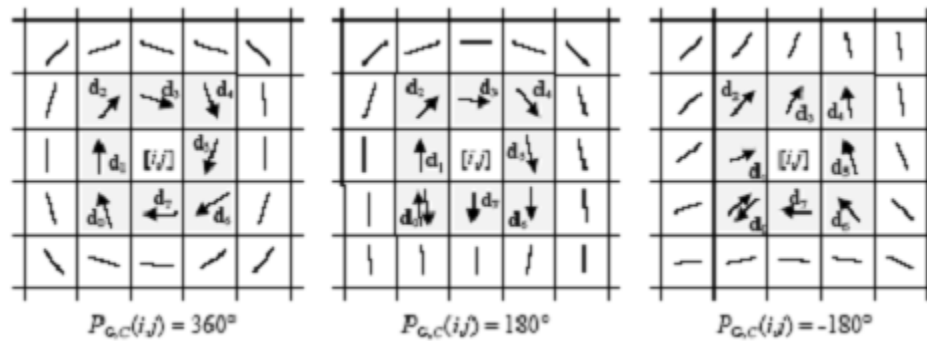
Trong đó: N_p là tổng số điểm trên đường cong “số” C

$\theta(x,y)$ là hướng tại điểm (x,y)

Dựa vào chỉ số Poincare ta có thể xác định các điểm singularity như sau:

$$Poincare(i, j) = \begin{cases} 0^0 & (i,j) \text{ không phải là điểm singularity} \\ 360^0 & (i,j) \text{ là điểm whorl} \\ 180^0 & (i,j) \text{ là điểm loop} \\ -180^0 & (i,j) \text{ là điểm delta} \end{cases}$$

Hình 13 minh họa cho cách tính chỉ số Poincare tại điểm (i,j) với số điểm trên đường cong “số” $N_p = 8$.

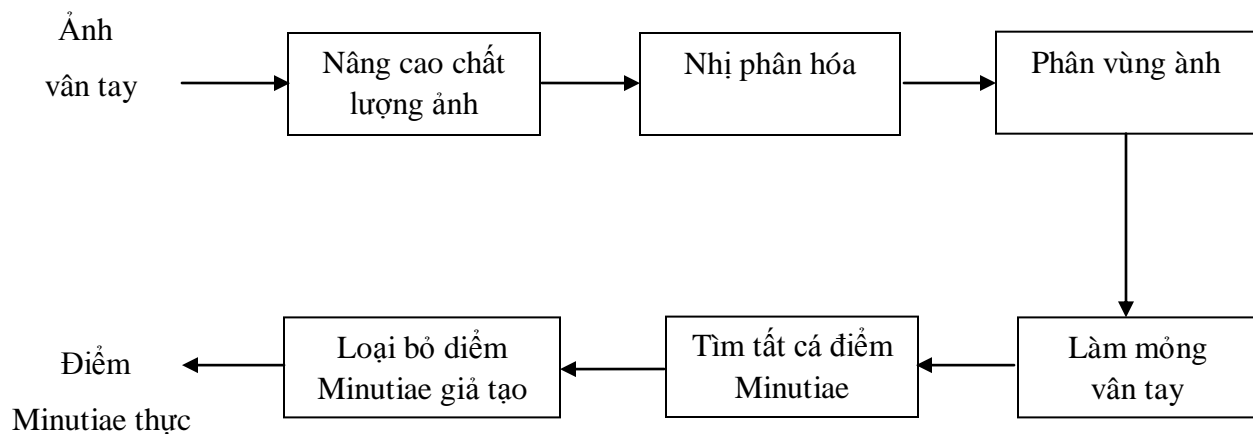


Hình 13: Cách tính chỉ số poicare tại điểm (i,j) với $N_p = 8$

*** Trích các điểm Minutiae**

Có hai phương pháp chính để tìm các điểm minutiae: trích các điểm minutiae từ ảnh nhị phân (binary) và trích các điểm minutiae trực tiếp từ ảnh xám. Tong đề tài này, nhóm cảm biến tập trung vào phương pháp thứ nhất.

+ Trích các điểm minutiae từ ảnh nhị phân (binary):



Hình 14: Sơ đồ thuật toán trích các điểm Minutiae từ ảnh xám

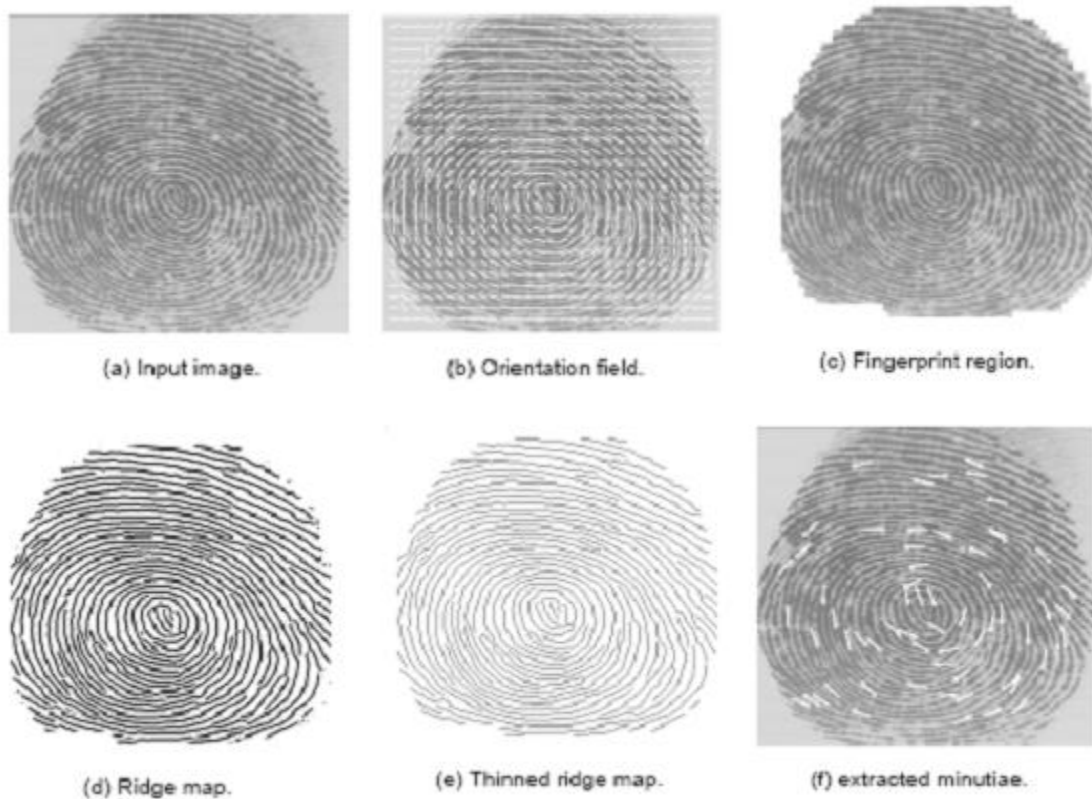
Ý tưởng chính của phương pháp này là từ ảnh xám ban đầu ta sử dụng các bộ lọc thích hợp để phát hiện và làm mảnh đường vân dưới dạng một pixel (ridge detection), biến đổi ảnh xám ban đầu thành ảnh binary (có giá trị là 0 hoặc 1) tương ứng.

Sau đó, các điểm minutiae sẽ được trích như sau: giả sử (x,y) là một điểm trên đường vân đã được làm mảnh và N_0, N_1, \dots, N_7 là 8 điểm xung quanh nó thì:

- (x,y) là một điểm kết thúc nếu $\sum_{i=0}^7 N_i = 1$

- (x,y) là một điểm rẽ nhánh nếu $\sum_{i=0}^7 N_i > 2$

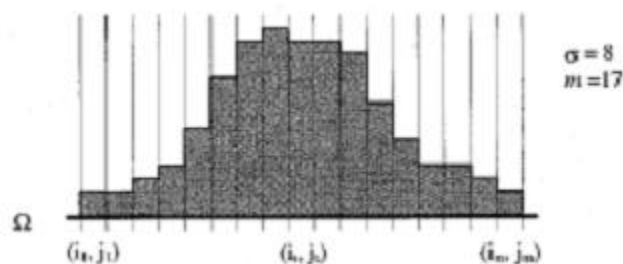
Phương pháp nhị phân hóa ảnh vân tay được trình bày cụ thể ở phần **3.d - Phát hiện điểm đặc trưng**.



Hình 15: Các kết quả của thuật toán

+ Trích các điểm minutiae trực tiếp từ ảnh xám

Giả sử I là một ảnh xám có kích thước $m \times n$, coi chiều thứ ba z là mức xám tại điểm (i, j) (xem hình 6). Theo quan điểm toán học, đường vân là tập hợp các điểm cực đại dọc theo một hướng xác định. Việc xác định các điểm Minutiae trực tiếp từ ảnh xám dựa vào thuật toán dò theo đường vân. Thuật toán này dựa vào việc xác định các điểm cực đại dọc theo đường vân. Điểm cực đại có thể được xác định bằng cách so sánh mức xám giữa các điểm trong thiết diện Ω .

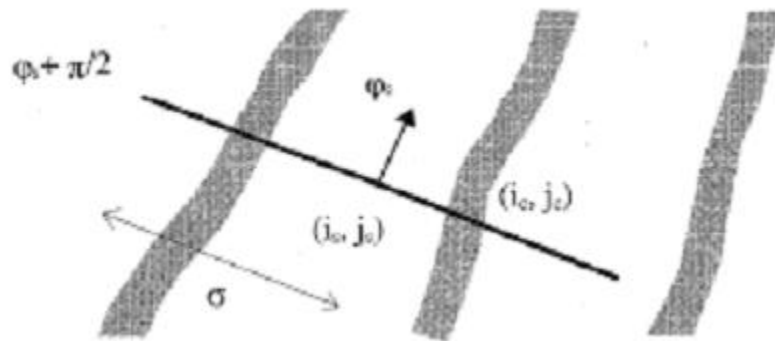


Hình 16: Thiết diện của đường vân (i_s, j_s)

Thuật toán tìm các điểm Minutiae bằng cách dò theo đường vân được thực hiện như sau:

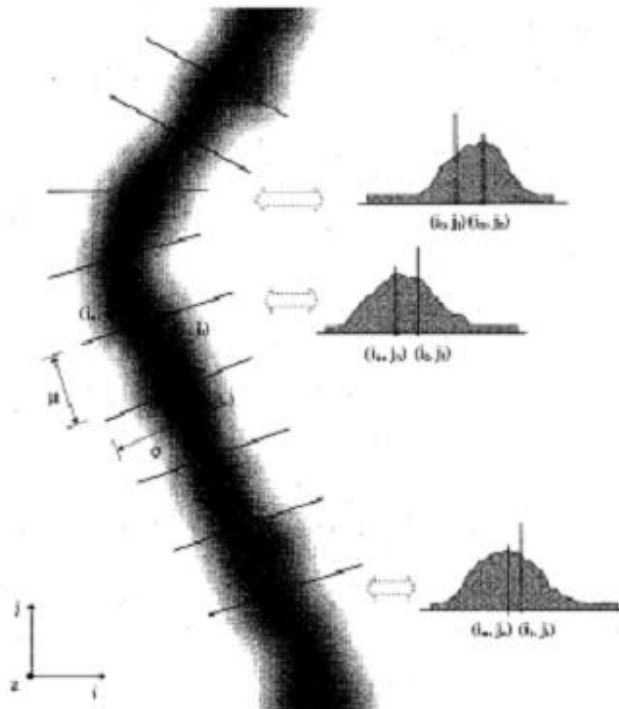
- Lấy một điểm bất kỳ (i_s, j_s) trên ảnh I .
- Tìm hướng φ_s tại điểm (i_s, j_s) .

- Tìm điểm cực đại (i_c, j_c) gần (i_s, j_s) nhất.



Hình 17: Điểm cực đại (i_c, j_c) gần (i_s, j_s) nhất

- Tìm hướng φ_c tại điểm (i_c, j_c) .
- Dịch chuyển theo hướng φ_c một đoạn μ .

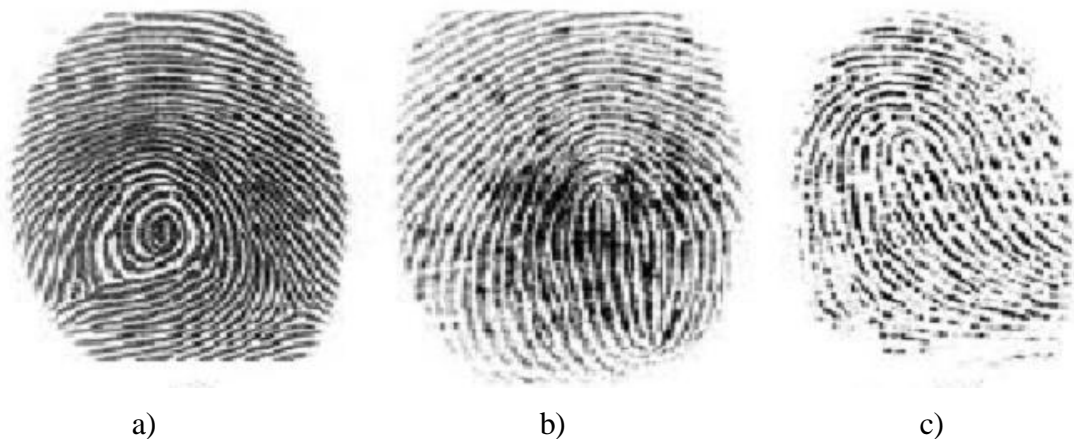


Hình 18: Dịch chuyển theo đường vân từng đoạn μ

- Tiếp tục quá trình này để dò theo theo đường vân (ridge following) cho đến khi không phát hiện được điểm cực đại (i_c, j_c) thì đó là điểm Ridge Ending hoặc chạm vào một đường vân khác thì đó là điểm Bifurcation (mỗi đường vân sau khi được dò theo được gán nhãn).
- Tiếp theo chọn một điểm (i_s, j_s) khác và thực hiện lại quá trình trên cho đến khi dò hết tất cả các đường vân.

c. Nâng cao chất lượng ảnh

Hiệu quả sử dụng của các thuật toán trích chọn điểm đặc trưng và các kỹ thuật nhận dạng vân tay khác phụ thuộc rất lớn vào chất lượng của ảnh vân tay đầu vào. Trong trường hợp ảnh vân tay có chất lượng tốt, vân lõm và vân lồi thay thế nhau và hướng theo một hướng cố định, các vân có thể dễ dàng được phát hiện và các chi tiết có thể xác định một cách chính xác trên ảnh (hình 19a). Nhưng trong thực tế, do điều kiện da (như khô hay ướt, bị cắt...), nhiễu cảm biến, ấn vân tay không đúng, và các ngón tay chất lượng thấp, một phần không nhỏ các ảnh vân tay (khoảng 10%) là có chất lượng thấp như trong hình 19b và 19c.



Hình 19: Một số ảnh vân tay

Trong nhiều trường hợp, một ảnh vân tay chứa nhiều vùng gồm có cả chất lượng tốt, trung bình và xấu. Các dạng mất giá trị liên hệ với vân tay bao gồm: các vân không liên tục, có vài nếp đứt; các vân song song không tách biệt rõ ràng do tồn tại của nhiều liên kết các vân song song, khiến chúng tách biệt kém; bị cắt, có nếp gấp, hay thâm sẹo. Ba dạng khiến vân mất giá trị này làm cho việc trích chọn điểm đặc trưng cực kì khó khăn do nảy sinh các vấn đề sau: trích chọn các chi tiết sai lệch; bỏ qua các chi tiết đúng; gây lỗi về hướng và vị trí của chi tiết. Vì vậy, để bảo đảm hiệu quả tốt trong các thuật toán trích chọn chi tiết trên các ảnh vân tay chất lượng kém, cần một thuật toán tăng cường để nâng cao sự rõ ràng trong cấu trúc vân.

Nói chung, với một ảnh vân tay cho trước, các vùng vân tay đã được phân đoạn có thể chia vào ba loại:

- Vùng được định nghĩa tốt: các vân được phân biệt rõ ràng với các vân khác.
- Vùng có khả năng phục hồi: các vân bị hư hỏng bởi các đường đứt gãy nhỏ, thâm sẹo... nhưng chúng vẫn có khả năng nhìn được và các vùng xung quanh cung cấp thông tin đủ để khôi phục cấu trúc ban đầu của chúng.
- Vùng không thể phục hồi: nơi các vân bị hư hại bởi các nhiễu nghiêm trọng, không có vân nào nhìn thấy được và các vùng xung quanh không cho phép chúng được xây dựng lại.

Các vùng chất lượng tốt, có thể phục hồi và không thể phục hồi có thể được nhận dạng qua vài tiêu chuẩn: độ tương phản, tính đầy đủ của hướng, tần suất vân, và các đặc tính cục bộ khác có thể kết hợp để định nghĩa chỉ số chất lượng. Mục đích của một thuật toán tăng cường là để cải

thiện tính rõ ràng của các cấu trúc vân trong các vùng có khả năng khôi phục và đánh dấu các vùng không thể khôi phục vì quá nhiều cho các xử lý tiếp theo.

Có rất nhiều phương pháp để nâng cao chất lượng vân trong đó có một phương pháp đã sử dụng và đạt hiệu quả rất tốt đó là sử dụng bộ lọc Gabor, được Hong, Wan và Jain (1998) đưa ra. Bộ lọc Gabor có các thuộc tính chọn tần suất và chọn hướng và có độ phân giải tùy chọn trong cả miền không gian và miền tần số (Daugman (1995), Jain và Farrokhnia (1991)). Như đã trong hình 6, một bộ lọc Gabor được định nghĩa bởi một sóng phẳng hình sin. Một bộ lọc đối xứng hai chiều Gabor có dạng sau:

$$g(x, y : \theta, f) = \exp \left\{ -\frac{1}{2} \left[\frac{x_{\theta}^2}{\sigma_x^2} + \frac{y_{\theta}^2}{\sigma_y^2} \right] \right\} \cdot \cos(2\pi f \cdot x_{\theta})$$

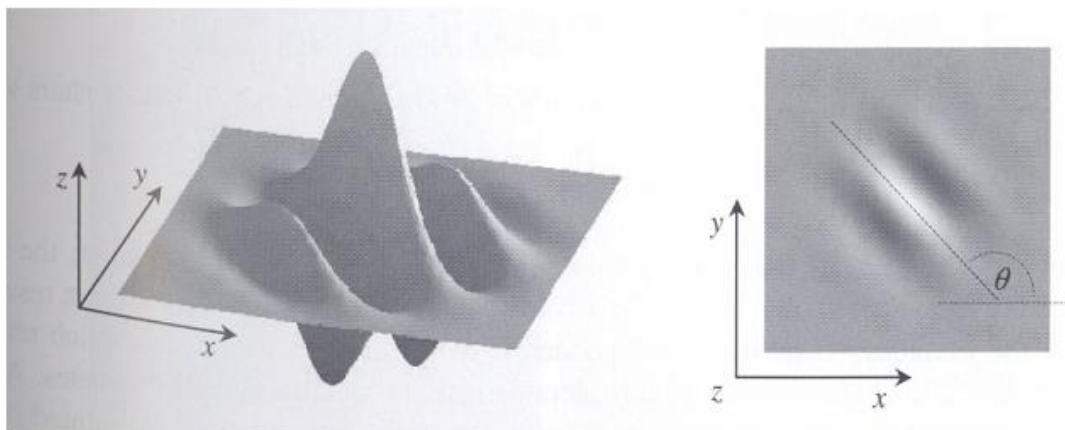
$$\begin{pmatrix} x_{\theta} \\ y_{\theta} \end{pmatrix} = \begin{pmatrix} \cos(90^\circ - \theta) & \sin(90^\circ - \theta) \\ -\sin(90^\circ - \theta) & \cos(90^\circ - \theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \sin \theta & \cos \theta \\ -\cos \theta & \sin \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Trong đó: + θ là hướng của bộ lọc.

+ $[x_0, y_0]$ là ảnh của $[x, y]$ sau khi quay quanh trục Cartesian một góc $(90^\circ - \theta)$.

+ f là tần suất của sóng phẳng hình sin.

+ σ_x, σ_y là độ lệch chuẩn Gauss tương ứng dọc theo trục x và trục y .



Hình 20: Biểu diễn đồ họa của bộ lọc Gabor xác định bởi các tham số $\theta = 90^\circ$, $f = 1/5$, $\sigma_x = \sigma_y = 3$

Để áp dụng các bộ lọc Gabor tới một ảnh, cần xác định bốn tham số ($\theta, f, \sigma_x, \sigma_y$). Tần suất của bộ lọc hoàn toàn được quyết định bởi tần suất vân cục bộ và hướng của bộ lọc được quyết định bởi hướng vân cục bộ. Tuy nhiên, do việc tính toán tần suất vân tay cục bộ tại mỗi điểm ảnh là rất phức tạp nên có thể ước lượng giá trị của f phù hợp ($1/5, 1/7, 1/9, \dots$). Việc chọn các giá trị σ_x và σ_y có thể hoán đổi cho nhau. Nếu chọn giá trị lớn thì bộ lọc sẽ chịu nhiễu nhiều hơn, nhưng lại

tạo ra sự nhầm lẫn giữa vân lồi và vân lõm. Ngược lại nếu chọn giá trị nhỏ, thì các bộ lọc ít nhầm lẫn giữa vân lồi và vân lõm nhưng sau đó chúng sẽ ít hiệu quả trong việc loại bỏ các nhiễu. Dựa trên dữ liệu kinh nghiệm, Hong, Wan và Jain đặt $\sigma_x = \sigma_y = 4$.

* **Thuật toán tăng cường chất lượng ảnh bằng bộ lọc Gabor như sau:**

+ Chuẩn hóa mức xám: Đặt $I(x,y)$ là mức xám tại điểm (x,y) của ảnh I . Đầu tiên tính kỳ vọng của ảnh theo công thức:

$$m = \frac{\sum_{x=1}^{width} \sum_{y=1}^{height} I_{(x,y)}}{width * height}$$

Sau đó tính giá trị phương sai của ảnh theo công thức:

$$v = \frac{\sum_{x=1}^{width} \sum_{y=1}^{height} (I_{(x,y)} - m)^2}{width * height}$$

Cuối cùng tính giá trị mức xám mới theo công thức:

$$I' [x, y] = \begin{cases} m_0 + \sqrt{(I[x,y]-m)^2 \cdot v_0 / v} & \text{khi } I[x,y] > m \\ m_0 - \sqrt{(I[x,y]-m)^2 \cdot v_0 / v} & \text{khi } I[x,y] \leq m \end{cases}$$

Trong đó: m_0 , v_0 là kỳ vọng và phương sai mong muốn, thường được chọn là 100.

Nếu mức xám của các vùng khác nhau trên ảnh I không đồng đều nhau thì có thể chia I thành các khối nhỏ và chuẩn hóa theo từng khối.

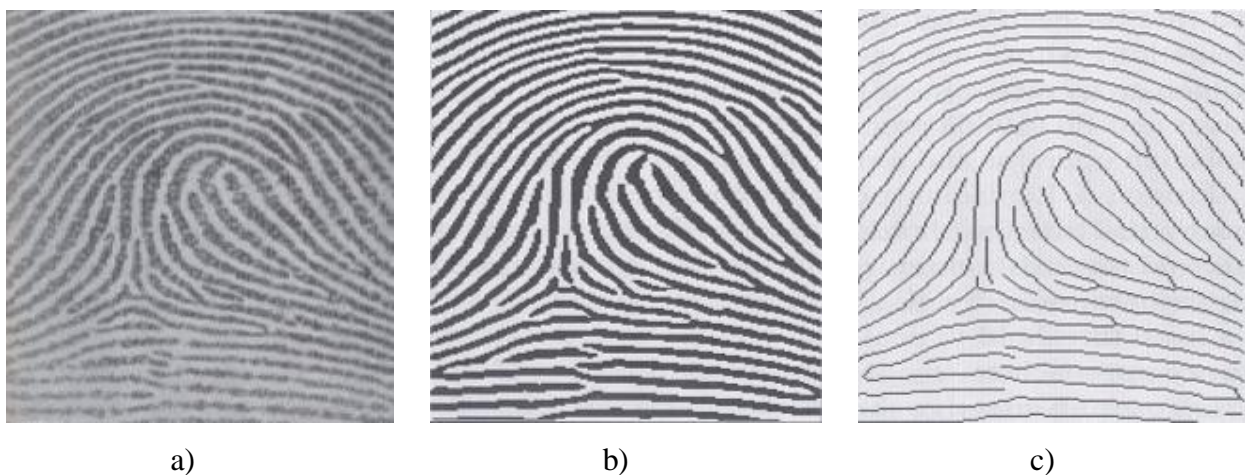


Hình 21: Ảnh vân tay ban đầu (a), ảnh chuẩn hóa của nó (b), ảnh sau khi lọc Gabor (c)

- + Xác định trường định hướng của vân theo phương pháp đã trình bày ở trên.
- + Sử dụng hàm lọc Gabor cho ảnh đã chuẩn hóa trong miền tần số
- Chia ảnh cần lọc thành từng khối nhỏ kích thước $W \times W$
- Xác định hướng của khối (dựa vào trường định hướng)
- Hướng θ của bộ lọc là hướng của khối
- Sử dụng phép biến đổi Fourier FFT và Fourier ngược IFFT cho từng khối ảnh và hàm Gabor

d. Phát hiện điểm đặc trưng

Hầu hết các hệ thống tự động so sánh các vân tay dựa trên đối sánh các điểm đặc trưng; vì vậy việc trích chọn điểm đặc trưng đáng tin cậy là một nhiệm vụ cực kì quan trọng, nhiều nghiên cứu đã được tiến hành theo hướng này. Dù các phương pháp khá khác nhau nhưng hầu hết chúng đều yêu cầu ảnh cấp xám vân tay được chuyển vào ảnh nhị phân. Trong các bước chuẩn hóa đã chuẩn bị một số giai đoạn để làm thuận tiện quá trình nhị phân hóa về sau. Một vài thuật toán chuẩn hóa cung cấp đầu ra là ảnh nhị phân, vì vậy sự phân biệt giữa chuẩn hóa và nhị phân hóa đôi khi bị xóa nhòa. Ảnh nhị phân nhận được bởi quá trình nhị phân hóa thường được qua giai đoạn làm mảnh làm cho độ dày của đường vân giảm xuống một điểm ảnh (hình 22). Cuối cùng qua quá trình quét ảnh cho phép phát hiện các điểm ảnh tương ứng với các điểm đặc trưng.



Hình 22: Ảnh xám ban đầu (a); Ảnh sau khi nhị phân hóa (b); Ảnh sau khi làm mảnh (c)

Một số người đưa ra các phương pháp trích chọn điểm đặc trưng làm việc trực tiếp trên các ảnh cấp xám mà không cần nhị phân hóa và làm mảnh. Lựa chọn này được đưa ra do các điều kiện sau: tập tin quan trọng có thể bị mất trong quá trình nhị phân hóa, nhị phân hóa và làm mảnh rất mất thời gian, làm mảnh có thể đưa ra một lượng lớn các chi tiết sai lệch, nếu thiếu bước chuẩn hóa, hầu hết các kĩ thuật nhị phân hóa không cung cấp các kết quả tốt khi áp dụng với các ảnh chất lượng thấp.

*** Phương pháp nhị phân hóa ảnh vân tay**

Vấn đề chung của nhị phân hóa được nghiên cứu rộng rãi trong xử lý ảnh và nhận dạng mẫu. Phương pháp dễ dàng nhất sử dụng ngưỡng toàn cục t và được thực hiện bằng cách thiết lập các điểm ảnh có cấp xám nhỏ hơn t về 0 và các điểm ảnh còn lại về 1. Nói chung, các phần khác nhau của ảnh có thể được đặc tính hóa bởi độ tương phản và cường độ khác nhau, vì vậy một ngưỡng đơn là không đủ để nhị phân hóa chính xác. Vì lý do này, kỹ thuật ngưỡng cục bộ thay đổi t một cách tự động, bằng cách điều chỉnh giá trị của nó theo cường độ cục bộ trung bình. Trong trường hợp ảnh vân tay chất lượng thấp, phương pháp ngưỡng cục bộ không phải lúc nào cũng bảo đảm một kết quả chấp nhận được, và các giải pháp đặc biệt là cần thiết.

Thuật toán thay ngưỡng tự động:

Gọi $t(g)$ là số điểm ảnh có giá trị $\leq g$, $m(g)$ là giá trị trung bình của các điểm ảnh có giá trị $\leq g$:

$$t(g) = \sum_{i=0}^g h(i)$$

$$m(g) = \frac{\sum_{i=0}^g i \cdot h(i)}{t(g)}$$

Gọi G là số cấp xám được xét, P là số điểm ảnh được xét:

$$f(g) = \frac{t(g)}{P - t(g)} * [m(g) - m(G - 1)]^2 - 1$$

Giá trị ngưỡng cần tìm là θ , sao cho $f(\theta) = \max f(g)$

Rồi tách ngưỡng.

$$I(x,y) = \begin{cases} 1 & \text{nếu } I(x,y) \geq \theta \\ 0 & \text{nếu } I(x,y) < \theta \end{cases}$$

Với mục đích cải thiện chất lượng ảnh nhị phân, vài nhà nghiên cứu giới thiệu các kỹ thuật chính quy phủ đầy các lỗ, loại bỏ các đứt gãy nhỏ, loại bỏ các cầu giữa các vân. Coetzee và Botha (1993) xác định các lỗ và kẽ hở bằng cách theo dõi các đường vân từ các cửa sổ điều hợp và loại bỏ chúng bằng cách sử dụng thuật toán màu - blob. Hung (1993) sử dụng kỹ thuật lọc điều hợp để cân bằng độ rộng vân.

Một khi ảnh xương đã nhận được, một bước quét ảnh đơn giản cho phép phát hiện các điểm ảnh tương ứng với các điểm đặc trưng: trong thực tế các điểm ảnh tương ứng với các điểm đặc trưng được đặc tính hóa bằng số điểm đi qua. Số điểm đi qua $cn(p)$ của một điểm ảnh p trong ảnh nhị

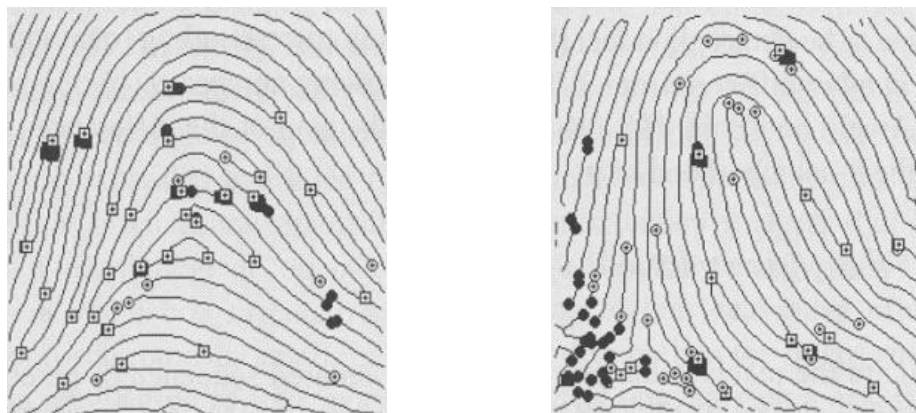
phân được xác định bằng một phần hai tổng các sai khác giữa các cặp điểm ảnh trong 8 lân cận của p.

$$cn(p) = \frac{1}{2} \sum |val(p_{i \bmod 8}) - val(p_{i-1})|$$

Trong đó: p_0, p_1, \dots, p_7 là các điểm ảnh láng giềng lân cận của điểm ảnh p và $val(p)$ thuộc $\{0, 1\}$ là giá trị của điểm ảnh. Điểm p là:

- là điểm vân trung gian nếu $cn(p) = 2$.
- là điểm kết thúc nếu $cn(p) = 1$.
- là các điểm đặc trưng phức tạp hơn (điểm rẽ nhánh, điểm giao cắt,...) nếu $cn(p) \geq 3$.

Hình 23 thể hiện hai ví dụ của quá trình trích chọn đặc tính từ ảnh nhị phân. Các vòng tròn màu trắng và các hộp trắng tương ứng là điểm kết thúc và điểm rẽ hai. Các vòng tròn màu đen và các hộp màu đen tương ứng là các điểm đặc trưng đã được lọc.



Hình 23: Hai ví dụ về phát hiện chi tiết trên xương

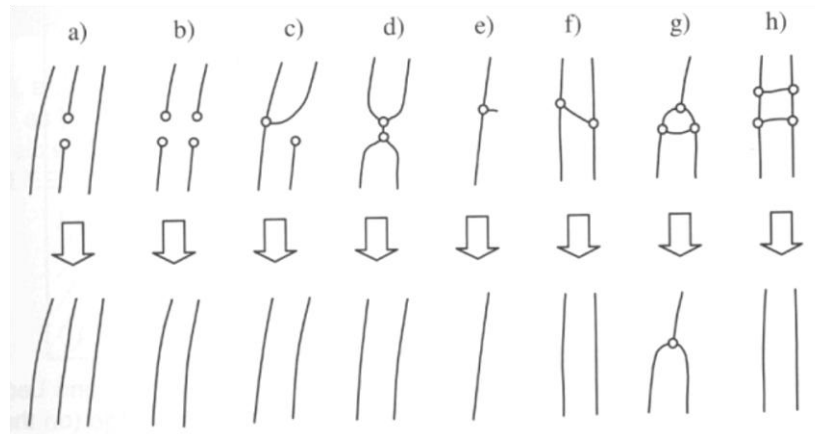
e. Lọc điểm đặc trưng bị lỗi

Để phát hiện các điểm đặc trưng sai làm ảnh hưởng đến các ảnh vân tay nhị phân được làm mạnh, có thể sử dụng vài luật cấu trúc đơn giản. Xiao (1991) xác định hầu hết các cấu trúc điểm đặc trưng sai và đưa ra phương pháp loại bỏ chúng. Thuật toán sử dụng dựa trên các luật, yêu cầu các đặc tính số học liên quan đến các điểm đặc trưng: chiều dài các vân liên quan (s), góc điểm đặc trưng, và số các điểm đặc trưng đối diện gần kề trong lân cận. Farina, Kovacs-Vajina và Leone (1999) đã đưa ra vài luật và thuật toán tiền xử lý điểm đặc trưng.

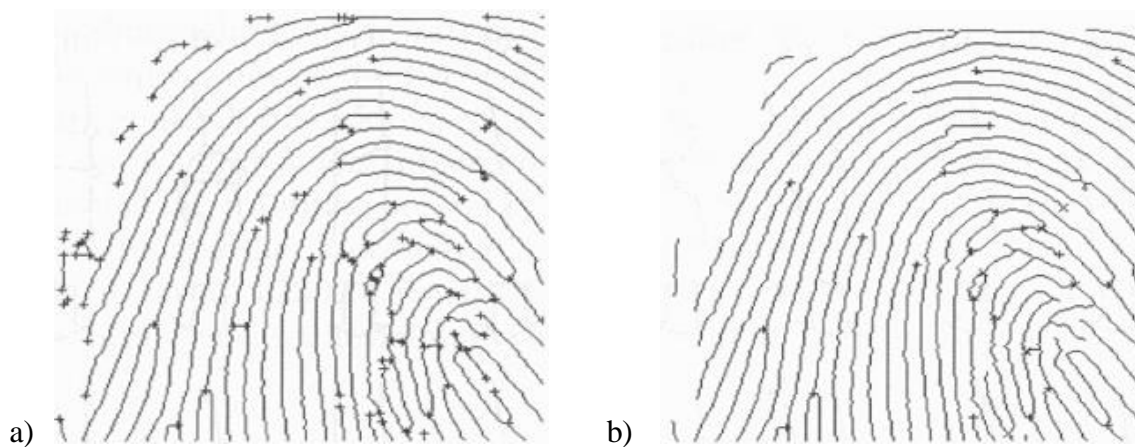
+ Các cầu bị loại bỏ khi nhìn nhận chúng trong một điểm chỉ hai sai, chỉ có hai nhánh được căn chỉnh, còn nhánh thứ ba thường vuông góc với hai nhánh còn lại.

+ Các vân quá ngắn được loại bỏ dựa vào khi so sánh chiều dài vân với khoảng cách trung bình giữa các vân.

+ Các điểm kết thúc và rẽ hai được kiểm tra: chúng được loại bỏ nếu không thỏa mãn tính hình học topo.



Hình 24: Các cấu trúc sai cơ bản (hàng đầu tiên) và cấu trúc sau khi sửa lỗi (hàng thứ hai)



Hình 25: Hậu xử lý chi tiết theo phương pháp của Farina, Kovacs-vajina và Leone (1999). Ảnh vân tay trước khi xử lý (a); ảnh vân tay sau khi xử lý (b)

f. Đối sánh vân tay

Đối sánh vân tay là công việc cuối cùng của quá trình nhận dạng vân tay, nó trả lời câu hỏi “Vân tay này là của ai trong số những người đó”. Bài toán nhận dạng vân tay (tìm kiếm một vân tay đầu vào trong một cơ sở dữ liệu có N vân tay) có thể được thực hiện như là thực hiện tuần tự đối sánh một - một giữa các cặp vân tay. Sự phân loại vân tay và các kĩ thuật đánh chỉ số thường được sử dụng để tăng tốc độ tìm kiếm trong các bài toán nhận dạng vân tay.

Một lượng lớn các thuật toán đối sánh vân tự động đã được đề nghị trong các tài liệu nhận dạng mẫu. Hầu hết các thuật toán này không gặp khó khăn trong đối sánh các ảnh vân tay chất lượng tốt. Nhưng trong đối sánh vân tay tồn tại thách thức ở các ảnh chất lượng thấp và vấn đề đối sánh từng vùng vân tay. Trong trường hợp hệ thống trợ giúp con người AFIS, một thuật toán kiểm tra chất lượng được sử dụng để chỉ lấy và chèn vào cơ sở dữ liệu các ảnh vân tay tốt. Hơn nữa, quá trình xử lý các mẫu vân khó có thể được quản lý. Dù sao, sự can thiệp là không thể trong các hệ

thống nhận dạng trực tuyến tự động - những hệ thống này đang có nhu cầu ngày càng tăng trong các ứng dụng thương mại.

Các phương pháp đối sánh vân tay có thể được phân loại thành ba họ:

+ Đối sánh dựa độ tương quan: Hai ảnh vân tay được đặt chồng lên và độ tương quan giữa các điểm ảnh tương ứng được tính với sự căn chỉnh khác nhau (ví dụ với các vị trí và độ quay khác nhau).

+ Đối sánh dựa vào điểm đặc trưng: Đây là kỹ thuật phổ biến nhất và được sử dụng rộng rãi nhất, do các chuyên gia pháp lý so sánh các vân tay và chấp nhận phương pháp như là bằng chứng định danh trong các phiên tòa ở hầu hết các quốc gia. Điểm đặc trưng được trích chọn từ hai vân tay được lưu giữ như là tập các điểm trong một bề mặt hai chiều. Đối sánh dựa vào điểm đặc trưng cơ bản bao gồm tìm kiếm sự căn chỉnh giữa tập điểm đặc trưng mẫu và tập điểm đặc trưng đầu vào dẫn tới sự so khớp lớn nhất các cặp điểm đặc trưng.

+ Đối sánh dựa đặc tính vân: Trong các ảnh vân tay chất lượng thấp, việc trích chọn chi tiết rất khó khăn. Khi các đặc trưng khác của mẫu vân tay (ví dụ: hướng và tần suất cục bộ, hình dạng vân, thông tin kết cấu) có thể được trích chọn một cách tin cậy hơn chi tiết, sự khác biệt của chúng là không cao. Các phương pháp thuộc họ này so sánh các vân tay với các đặc trưng được trích chọn từ các mẫu vân. Về nguyên lý, đối sánh dựa độ tương quan và đối sánh dựa vào chi tiết có thể xem như là một phần của đối sánh dựa đặc trưng vân, theo cách hiểu mật độ điểm ảnh, vị trí chi tiết là những đặc trưng của mẫu vân ngón tay

Nhiều kỹ thuật khác cũng đã được đề nghị chính thức, về nguyên lý, có thể được xếp vào các loại trên theo các đặc trưng được sử dụng, nhưng chúng ta đề cập để phân loại chúng tách biệt trên cơ sở các kỹ thuật đối sánh. Chúng bao gồm các phương pháp dựa mạng thần kinh và các cố gắng thực hiện đối sánh vân sử dụng các bộ xử lý song song hoặc với các kiến trúc dành riêng khác.

Trong đồ án này, nhóm cảm biến sử dụng phương pháp đối sánh điểm đặc trưng dựa biến đổi Hough.

* Thuật toán Hough

Thuật toán Hough là thuật toán tìm ra các giá trị tịnh tiến theo trục x, trục y, góc quay ngược chiều kim đồng hồ và độ nở của vân (Δx , Δy , θ , s) sao cho khi thực hiện các phép biến đổi đó thực hiện trên tập điểm đặc trưng của vân tay này sẽ thu được tập điểm đặc trưng mới mà có số lượng điểm đặc trưng trùng khớp với tập điểm đặc trưng của vân tay kia là lớn nhất.

Không gian biến đổi là không gian bốn chiều (Δx , Δy , θ , s) trong đó mỗi tham số được rời rạc hóa (kí hiệu bởi dấu $^+$) vào tập các giá trị: $\Delta x^+ \in \{\Delta x_1^+, \Delta x_2^+, \dots, \Delta x_a^+\}$; $\Delta y^+ \in \{\Delta y_1^+, \Delta y_2^+, \dots, \Delta y_b^+\}$; $\theta^+ \in \{\theta_1^+, \theta_2^+, \dots, \theta_c^+\}$; $s^+ \in \{s^+_{1}, s^+_{2}, \dots, s^+_{d}\}$.

Thuật toán phát biểu như sau:

Với mỗi m_i ($i = 1, \dots, m$), m'_j ($j = 1, \dots, n$), $\theta^+ \in \{\theta_1^+, \theta_2^+, \dots, \theta_c^+\}$, $s^+ \in \{s^+_{1}, s^+_{2}, \dots, s^+_{d}\}$ nếu khoảng cách trực tiếp $(\theta'_j + \theta^+, \theta_i) < \theta_0$

{

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix} - s^+ \begin{bmatrix} \cos \theta^+ & -\sin \theta^+ \\ \sin \theta^+ & \cos \theta^+ \end{bmatrix} \begin{bmatrix} x'_j \\ y'_j \end{bmatrix}$$

$\Delta x^+, \Delta y^+ =$ lượng tử hóa của $\Delta x, \Delta y$, tới tập gần nhất.

$$A[\Delta x^+, \Delta y^+, \theta^+, s^+] = A[\Delta x^+, \Delta y^+, \theta^+, s^+] + 1$$

}

Ở đây A chính là mảng bốn chiều mà giá trị của $A[\Delta x^+, \Delta y^+, \theta^+, s^+]$ thể hiện số cặp điểm đặc trưng ứng hộ cho $[\Delta x^+, \Delta y^+, \theta^+, s^+]$.

Ở cuối quá trình tích lũy, biến đổi có căn chỉnh tốt nhất $(\Delta x^*, \Delta y^*, \theta^*, s^*)$ nhận được như sau $(\Delta x^*, \Delta y^*, \theta^*, s^*) = \arg \max A[\Delta x^+, \Delta y^+, \theta^+, s^+]$.

Đó chính là giá trị mà ta muốn tìm.

* Thực hiện đối sánh vân tay

Kí hiệu T và I là các biểu diễn của ảnh vân tay mẫu và vân tay đầu vào. Xem mỗi điểm đặc trưng như là một một nhóm bộ ba $m = \{x, y, \theta\}$ thể hiện vị trí điểm đó ở vị trí x, y và góc chi tiết θ .

$$T = \{m_1, m_2, \dots, m_m\}; m_i = \{x_i, y_i, \theta_i\}, i = 1 \dots m$$

$$I = \{m'_1, m'_2, \dots, m'_n\}; m'_j = \{x'_j, y'_j, \theta'_j\}, j = 1 \dots n$$

Trước hết chúng ta tiến hành tịnh tiến và quay tập điểm đặc trưng của hai ảnh theo các đại lượng chúng ta thu được từ thuật toán Hough. Tiến hành đối sánh lần lượt từng điểm đặc trưng của tập điểm đặc trưng ảnh vân tay 1 với tập điểm đặc trưng thu được.

Trong đó m và n là số các điểm đặc trưng trong T và I.

Một điểm đặc trưng m'_j trong I và một điểm đặc trưng m_i trong T được xem là so khớp nếu khoảng cách không gian (sd) giữa chúng là nhỏ hơn mức dung sai cho trước r_0 và sự khác nhau về hướng (dd) giữa chúng là nhỏ hơn góc dung sai θ_0 :

$$sd(m'_j, m_i) = \sqrt{(x'_j - x_i)^2 + (y'_j - y_i)^2} \leq r_0$$

$$dd(m'_j, m_i) = \min(|\theta'_j - \theta_i|, 360^\circ - |\theta'_j - \theta_i|) \leq \theta_0$$

Đẳng thức thứ 2 lấy giá trị nhỏ nhất của $|\theta'_j - \theta_i|$ và $360^\circ - |\theta'_j - \theta_i|$ bởi vì tính chu kì của góc (sự khác nhau giữa 2° và 358° chỉ là 4°). Chúng ta cần một hộp dung sai được định nghĩa qua r_0 và θ_0 để bù vào các lỗi không thể ngăn ngừa do các thuật toán trích chọn đặc trưng và các nhiễu mềm dẻo làm cho vị trí các điểm đặc trưng thay đổi.

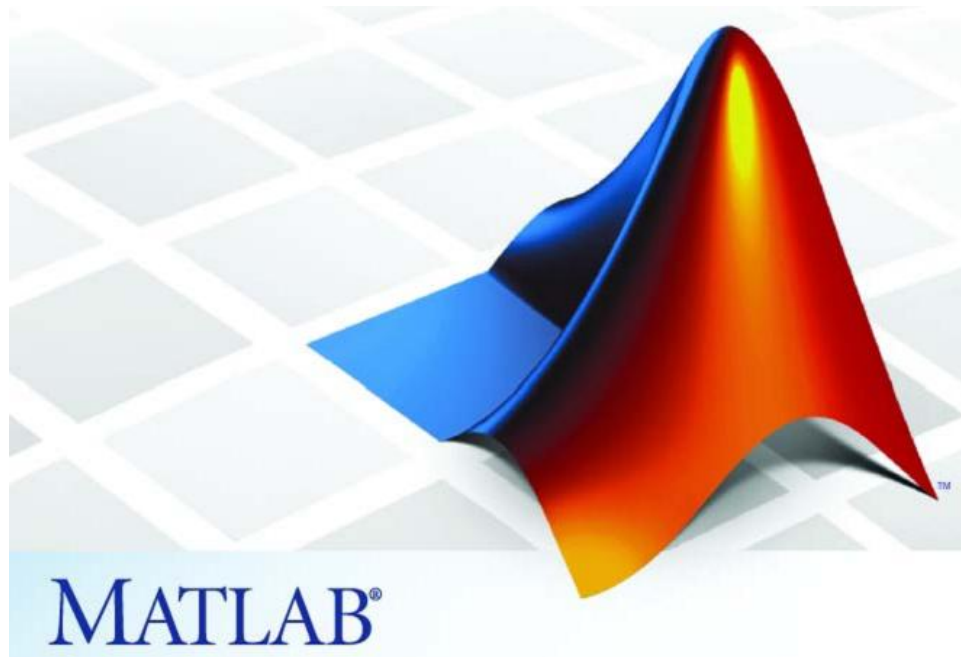
CHƯƠNG III

CHƯƠNG TRÌNH NHẬN DẠNG VÂN TAY

1. Giới thiệu chung về chương trình

Đây là chương trình nhận dạng vân tay dựa theo phương pháp trích các điểm đặc trưng Singularity và Minutiae, và như đã nói ở trên, do hạn chế về trình độ và điều kiện không cho phép nên nhóm cảm biến chỉ tập trung vào việc thực hiện trích các điểm Minutiae, lựa chọn điểm kết thúc để đối sánh vân tay. Chương trình thực hiện việc lưu trữ dữ liệu vân tay, lưu thông tin cá nhân của người có vân tay đó và nhận dạng vân tay của một người nhằm kiểm tra người đó có phải là một trong những thành viên của cơ sở dữ liệu hay không. Các dữ liệu vân tay được lưu trữ trong một file *.dat, còn thông tin các nhân được lưu trữ trong một file *.mat. Việc nhận dạng vân tay thông qua đối sánh một – một lần lượt giữa vân tay cần nhận dạng với các vân tay trong cơ sở dữ liệu, thực tế quá trình này được chuyển về việc so sánh hai file *.dat với nhau. Kết thúc việc nhận dạng, chương trình sẽ đưa ra một thông báo “Bạn là thành viên” hoặc “Bạn không phải là thành viên”. Nếu “Bạn là thành viên”, chương trình sẽ đưa ra các thông tin cá nhân đã được lưu trữ của bạn.

a. Phần mềm



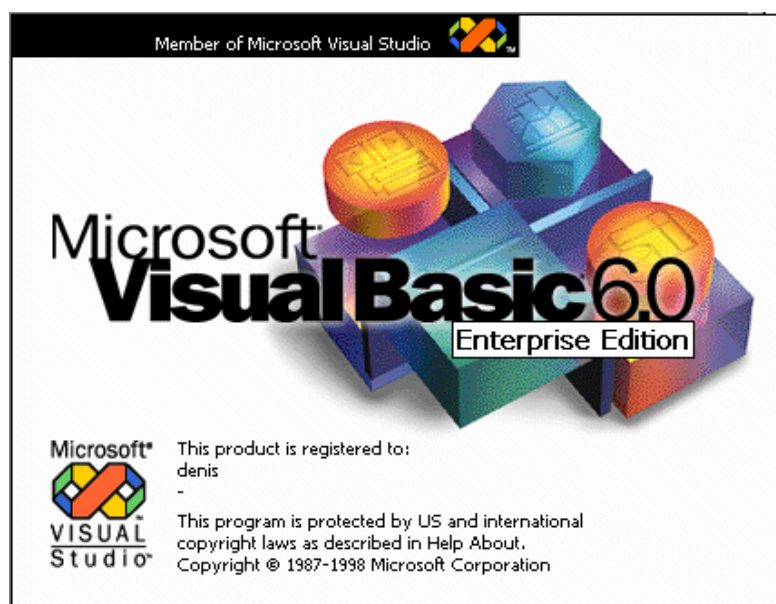
Hình 26: MATLAB – công cụ hữu hiệu để xử lý các bài toán kỹ thuật

Chương trình nhận dạng vân tay của nhóm cảm biến vân tay được xây dựng bằng phần mềm MATLAB, phiên bản 7.7.0 (R2008b). Đây là phần mềm được dùng rộng rãi trong giáo dục, phổ biến nhất là giải các bài toán số trị (cả đại số tuyến tính lẫn giải tích) trong nhiều lĩnh vực kỹ thuật. MATLAB là một môi trường tính toán số và lập trình, được thiết kế bởi công ty MathWorks. MATLAB cho phép tính toán số với ma trận, vẽ đồ thị hàm số hay biểu đồ thông

tin, thực hiện thuật toán, tạo các giao diện người dùng và liên kết với những chương trình máy tính viết trên nhiều ngôn ngữ lập trình khác. Với thư viện Toolbox, MATLAB cho phép mô phỏng tính toán, thực nghiệm nhiều mô hình trong thực tế và kỹ thuật. Giao diện của chương trình được xây dựng bằng công cụ xây dựng giao diện người dùng GUI (Graphical User Interfaces), việc xử lý ảnh vân tay cũng được thực hiện nhờ sự trợ giúp của công cụ xử lý ảnh (Image Processing Toolbox) trong MATLAB. Các bước xử lý ảnh được lập trình và lưu thành các file *.m, do số lượng file lớn và mã lập trình dài nên toàn bộ các file này cũng như các vân tay có sẵn để thử nghiệm được lưu ở địa chỉ Web

<http://galaxyz.net/data/5a686e6f686d6862/mxlavtbml.rar.glx>

(Chú ý: Chương trình này chỉ chạy được trên MATLAB phiên bản 7.3 trở lên).



Hình 27: Microsoft Visual Basic 6.0

Nhược điểm của MATLAB là tuy nó hỗ trợ được một số thiết bị ghép nối với máy tính qua cổng USB nhưng việc lập trình khá phức tạp, do đó chương trình lấy ảnh vân tay từ thiết bị lấy ảnh vân tay (phần b dưới đây) được xây dựng từ phần mềm Microsoft Visual Basic 6.0.

b. Phần cứng

*** Cảm biến vân tay**

Cảm biến vân tay là một thiết bị điện tử được sử dụng để chụp một hình ảnh kỹ thuật số của các mẫu vân tay. Hình ảnh chụp được gọi là mẫu lấy trực tiếp. Nó được xử lý kỹ thuật số để tạo ra một mẫu sinh trắc học được lưu giữ và sử dụng cho việc đối sánh. Đây là điểm chung của một số công nghệ sử dụng các bộ cảm biến vân tay.

Sự tạo ảnh vân tay quang học bao gồm việc chụp một hình ảnh kỹ thuật số của ảnh sử dụng ánh sáng nhìn thấy được. Về bản chất, loại cảm biến là máy ảnh kỹ thuật số đặc biệt. Lớp trên cùng

của cảm biến, nơi ngón tay được đặt vào, được gọi là bề mặt cảm ứng. Bên dưới lớp này là một lớp phosphor phát sáng chiếu vào bề mặt của ngón tay. Ánh sáng phản chiếu từ các ngón tay đi qua lớp phosphor thành một chuỗi các trạng thái rắn pixels thu thập hình ảnh trực quan của vân tay. Một trầy xước hoặc bề mặt cảm ứng bẩn có thể gây ra một hình ảnh vân tay chất lượng kém. Một nhược điểm của loại cảm biến này trong thực tế là khả năng hình ảnh bị ảnh hưởng lớn bởi chất lượng của da trên ngón tay. Ví dụ, một ngón tay bẩn hoặc trầy xước rất khó để có được một hình ảnh đúng. Nó cũng có thể dễ dàng bị đánh lừa bởi một hình ảnh vân tay nếu không kết hợp với một bộ dò “ngón tay sống”. Tuy nhiên, không giống như cảm biến điện dung, công nghệ cảm biến này không nhạy cảm với hiện tượng phóng điện.

*** Cơ sở của hệ thống máy quét quang học**

Có nhiều cách khác nhau để lấy hình ảnh ngón tay, trong đó cách phổ biến nhất là quét quang học và quét điện dung. Cả hai đều đưa ra cùng một hình ảnh, nhưng bằng hai phương pháp khác nhau hoàn toàn.

Trung tâm của máy quét quang học là **CCD (Charge Coupled Device)** - hệ thống cảm biến ánh sáng sử dụng trong camera kỹ thuật số. CCD là một mảng diode nhạy cảm với ánh sáng gọi là **photosite**, có nhiệm vụ tạo tín hiệu điện tương ứng với những photon ánh sáng. Mỗi **photosite** ghi lại một pixel, tức một chấm nhỏ thể hiện rằng ánh sáng đã chạm đến điểm đó. Các pixel sáng và tối sẽ tổng hợp thành một hình ảnh của vật thể được quét (như ngón tay). Thường thì Bộ chuyển đổi ADC (từ analog sang digital) trong hệ thống quét sẽ xử lý tín hiệu điện analog để tạo ra bức ảnh dạng số hóa.

Quy trình quét bắt đầu khi bạn đặt ngón tay lên một đĩa thủy tinh, và để CCD chụp ảnh. Máy quét có nguồn điện riêng, thường là một mảng diode phát sáng để tỏa sáng các đường vân trên ngón tay. Cuối cùng hệ thống CCD sẽ tạo ra hình ảnh đảo ngược của ngón tay. Bộ xử lý quét sẽ đảm bảo rằng hình ảnh thu được đủ rõ bằng cách kiểm tra độ tối pixel trung bình, hay tổng giá trị của một mẫu nhỏ, và sẽ từ chối quét hình nếu nó quá sáng hoặc quá tối. Nếu ảnh bị từ chối, máy quét sẽ điều chỉnh thời gian phơi sáng, rồi quét lại lần nữa. Còn nếu độ tối đã đủ thì hệ thống sẽ tiếp tục kiểm tra độ phân giải ảnh. Bộ xử lý sẽ quan sát một số đường thẳng di chuyển ngang dọc trên ảnh. Nếu ảnh có độ phân giải tốt, đường thẳng chạy vuông góc với vân tay sẽ gồm các phần xen kẽ gồm các pixel rất tối và rất sáng.

*** Thiết bị *U.are.U 4500 Fingerprint Reader***

Một việc rất quan trọng và cũng đầy khó khăn là lấy được ảnh vân tay. Trên thị trường Việt Nam hiện nay có nhiều thiết bị thu ảnh vân tay của nước ngoài nhưng hoặc đều đã ở dạng thành phẩm mang tính ứng dụng như máy chấm công, khóa cửa vân tay,... hoặc thu được ảnh do một bên khác viết driver thiết bị và chương trình thu ảnh. Qua nghiên cứu và tìm hiểu trên internet, nhóm cảm biến đã tìm mua và sử dụng thiết bị *U.are.U 4500 Fingerprint Reader*. Máy đọc dấu vân tay *U.are.U 4500* là máy quét dấu vân tay quang học, của hãng Digital Persona được IT WORKS độc quyền phân phối ở khu vực Châu Á Thái Bình Dương. Sản phẩm được thiết kế có kích cỡ nhỏ gọn nên tiết kiệm diện tích trên bàn làm việc và có đế cao su nên ngăn trơn trượt rất tốt. *U.are.U 4500* có ánh sáng màu xanh đen không làm chói mắt trong không gian ít ánh sáng và không làm lẫn lộn với màu đỏ của các tín hiệu cảnh báo khác. Khi cần sử dụng, người dùng chỉ đặt ngón tay trên mặt tiếp xúc, máy đọc dấu vân tay sẽ tự động quét vân tay một cách nhanh chóng. Khi đèn đỏ chớp cho thấy dấu vân tay đã được quét. Sau đó, máy sẽ nhập mật mã trước khi gửi dữ liệu qua USB. Máy đọc dấu vân tay của DigitalPersona sử dụng công nghệ quét quang

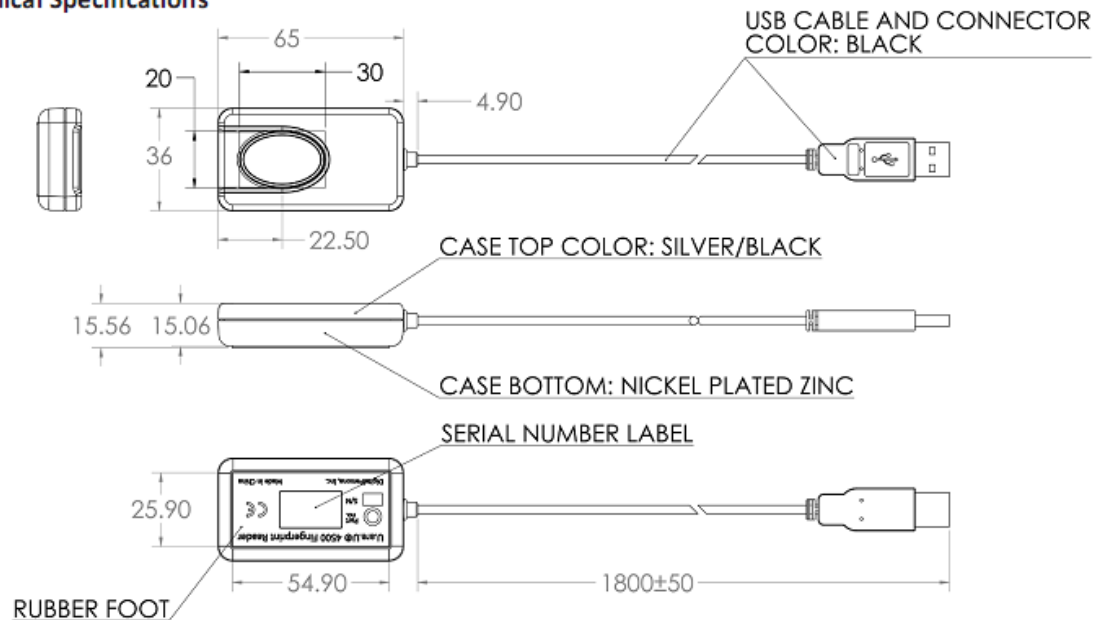
cho chất lượng hình ảnh tốt nhất. Máy đọc dấu vân tay U.are.U 4500 và công nghệ quét dấu vân tay của DigitalPersona có thể kiểm tra dấu vân tay khó đọc nhất một cách chính xác và nhanh chóng và có thể đặt ngón tay ở bất kỳ góc độ nào trên mặt tiếp xúc của máy đọc. Về giá cả, tại thời điểm tháng 4 – 2010, nhóm cảm biến đã mua thiết bị này với giá 2.400.00 VND, cộng thêm driver của hãng phân phối độc quyền IT WORKS Vietnam Ltd (157- 159 Nguyễn Đình Chiểu, quận 3, TP. Hồ Chí Minh. Tel: 0839308925. Fax: 0839302868. Email: vietnam@itworks.co.th) là 3.100.000 VND.



Hình 28: Thiết bị đọc vân tay U.are.U 4500 Fingerprint Reader

Thông tin kỹ thuật về thiết bị:

- Độ phân giải: 512 dpi
- Vùng quét ảnh: 14.6 mm (chiều rộng tính từ tâm) x 18.1 mm (chiều dài)
- Thang độ xám 8-bit (256 mức xám)
- Kích thước đầu đọc (xấp xỉ): 65 mm x 36 mm x 15.56 mm
- Tương thích với các thiết bị USB 1.0, 1.1 and 2.0 (Tốc độ cực đại)
- Sử dụng trong nhà, hộ gia đình, công sở, trường học,...

Mechanical Specifications**Hình 29: Sơ lược cấu tạo của thiết bị U.are.U 4500 Fingerprint Reader**

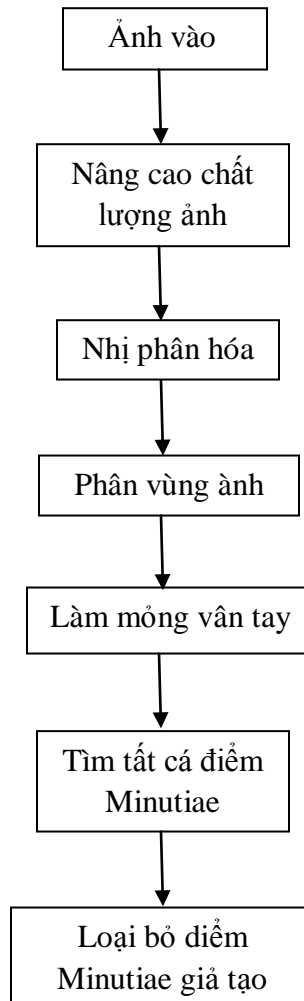
Các thông số định mức:

- Điện áp cung cấp 5.0V $\pm 5\%$ từ USB
- Dòng cung cấp – khi quét ảnh 190 mA (Tiêu chuẩn)
- Dòng cung cấp – khi chạy không 140 mA (Tiêu chuẩn)
- Dòng cung cấp – khi tạm ngừng 1.5 mA (Cực đại)
- Độ nhảy ESD >15 kV, thiết lập trong từng trường hợp
- Nhiệt độ, khi hoạt động 0 - 40°C
- Độ ẩm, khi hoạt động 20% - 80% (không ngưng tụ)
- Nhiệt độ, khi lưu trữ -10 - 60 C
- Độ ẩm, khi lưu trữ 20% - 90% (không ngưng tụ)
- Quét dữ liệu thang độ xám 8-bit
- Tuân theo tiêu chuẩn FCC Class B, CE, ICES, BSMI, MIC, USB, WHQ

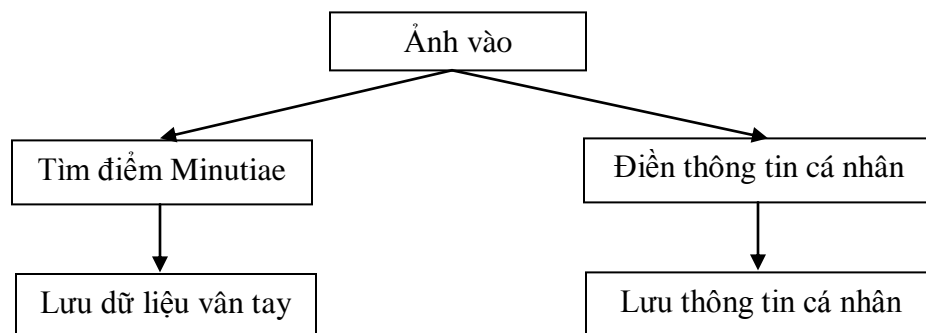
2. Lưu đồ giải thuật

Tổng quát: Lấy mẫu vân tay → Chọn vân tay để nhận dạng → Nhận dạng

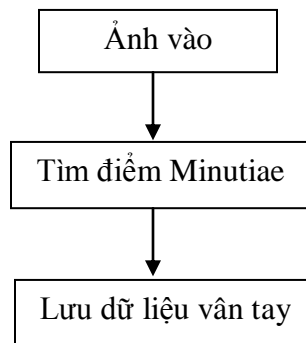
Lưu đồ giải thuật trích điểm Minutiae



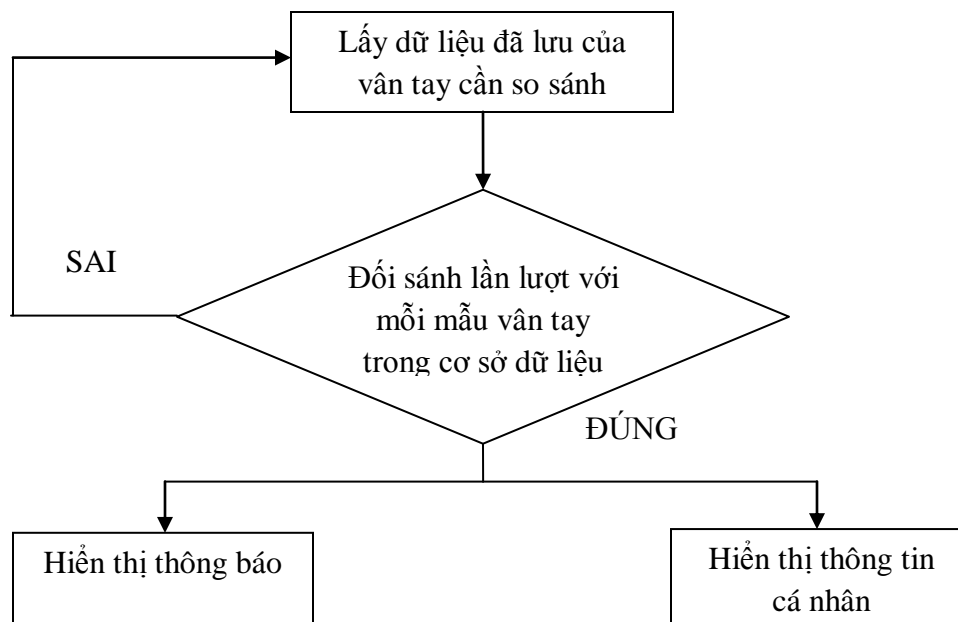
Lưu đồ giải thuật quá trình lấy mẫu vân tay



Lưu đồ giải thuật quá trình chọn vân tay để nhận dạng



Lưu đồ giải thuật quá trình nhận dạng



3. Giao diện chương trình



Hình 30: Giao diện của chương trình

Hàm xây dựng chương trình GDNDVT.m (xem phụ lục trang 41).

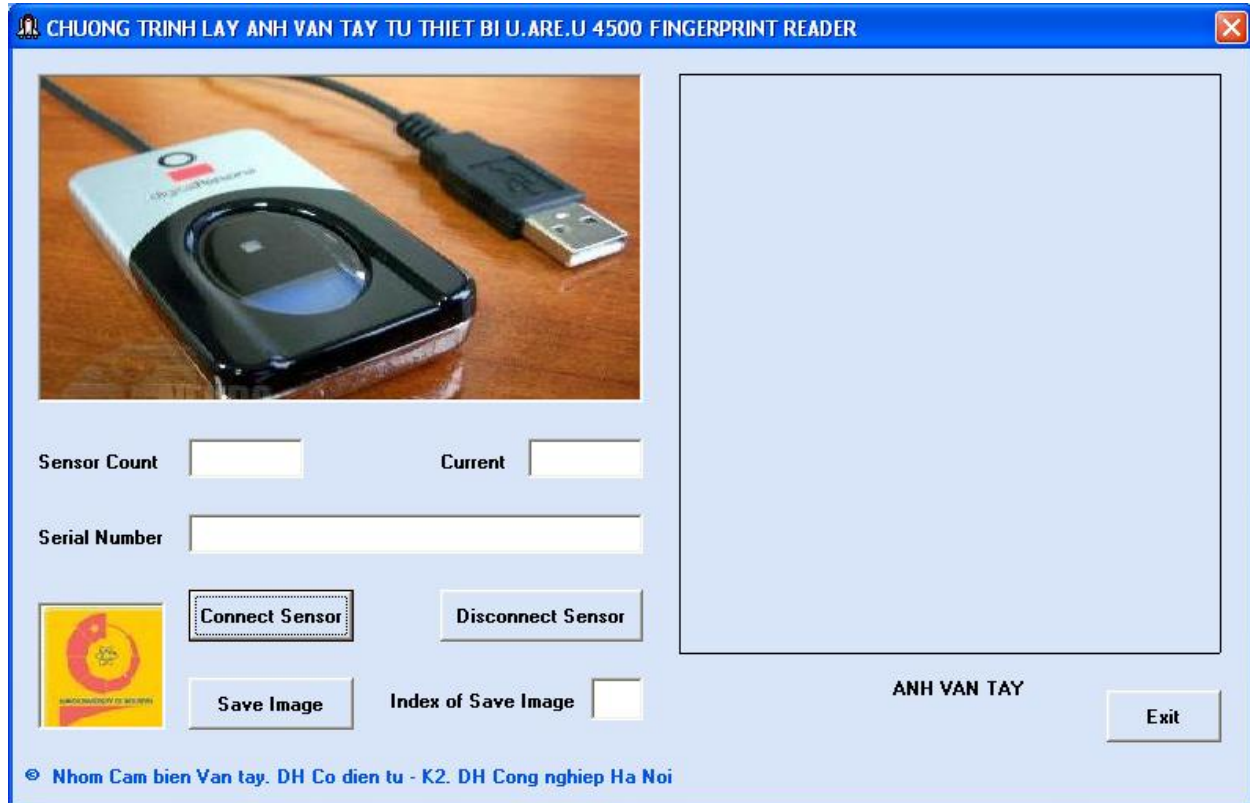
4. Hướng dẫn sử dụng chương trình

* **Chú ý:** Do yêu cầu chủ quan của chương trình nên việc lưu trữ các ảnh và lưu các file có một số quy định nhỏ như sau:

- Các ảnh vân tay dữ liệu (thư mục “vân tay du lieu”) của một người được lưu dưới tên xyz_i, trong đó: xyz là ký hiệu của người đó, i là số thứ tự mẫu vân tay của người đó ($i = 1, 2, 3, \dots$). Trong chương trình này $i = 1, 2, 3$. Tên các vân tay cần nhận dạng (thư mục “vân tay can nhan dang”) có thể dưới tên bất kỳ.
- Trong quá trình lấy mẫu, file *.dat (thư mục “co so du lieu”) cũng được lưu dưới tên xyz_i tương ứng với mẫu vân tay đưa vào. File *.mat (thư mục “thong tin thanh vien”) phải được lưu dưới tên xyz.
- Trong quá trình xử lý vân tay cần nhận dạng thì file *.dat (thư mục “du lieu can doi sanh”) được lưu dưới tên bất kỳ.

a. Lấy ảnh vân tay từ thiết bị U.are.U 4500 Fingerprint Reader

- Kết nối thiết bị với máy tính.
- Chương trình xuất hiện khi kích chuột vào nút “Lay vân tay” trong giao diện chương trình nhận dạng vân tay.



Hình 31: Giao diện chương trình lấy ảnh vân tay từ thiết bị U.are.U 4500 Fingerprint Reader

- Để sử dụng chương trình, trước hết phải kích chuột vào nút “Connect Sensor” để máy tính thực hiện việc kết nối với thiết bị và kích hoạt Cảm biến hoạt động.
- Áp ngón tay lên mặt gương của thiết bị sao cho bề mặt vân tay của ngón tay đó gần như khớp với bề mặt gương, như vậy ảnh vân tay thu được sẽ rõ nét và đầy đủ nhất, quá trình nhận dạng sẽ chính xác hơn. Chú ý: để đảm bảo tính thống nhất, vân tay được lấy là vân tay của ngón tay cái phải, tuy nhiên tùy theo tình trạng tốt xấu của vân tay từng người mà có thể lấy từ các ngón khác. Ảnh vân tay thu được hiện ra ở vị trí “ANH VAN TAY”.
- Kích chuột vào nút “Save Image” để lưu ảnh vân tay vừa thu được. Tùy theo người sử dụng muốn tạo cơ sở dữ liệu vân tay hay nhận dạng vân tay mà ảnh sẽ được lưu vào thư mục “vân tay du lieu” hay “vân tay can nhan dang” tương ứng. Để tăng độ chính xác nhận dạng, mỗi người sẽ được lấy ảnh 5 lần.
- Kết thúc việc nhận dạng, kích chuột vào nút “Disconnect Sensor” để ngắt kết nối và ngừng hoạt động của cảm biến.



Hình 32: Ảnh vân tay thu được

- Kích chuột vào nút “Exit” để thoát khỏi chương trình.

b. Lấy mẫu (Tạo cơ sở dữ liệu)

- Kích chuột vào nút “Lay mau”. Hộp thoại “Chon van tay de nhap du lieu” được mở ra, chọn một ảnh vân tay đã được lưu trong thư mục “van tay du lieu”. Ảnh vân tay được chọn sẽ hiện ra ở hình thứ nhất của phần “ANH VAN TAY”.

- Kích vào nút “Diem Minutiae” để bắt đầu quá trình phân tích ảnh vân tay. Các ảnh phân tích được thể hiện trên hình thứ hai của phần “ANH VAN TAY”. Để dễ dàng nắm được các bước xử lý, nên nhóm cảm biến đã cho các bước này lần lượt thực hiện bằng cách ấn một phím bất kỳ trên bàn phím cho đến khi hết quá trình tìm điểm Minutiae. Sau đó kích chuột vào nút “Luu du lieu” để lưu các điểm Minutiae tìm được vào một file dạng *.dat trong thư mục “co so du lieu”.



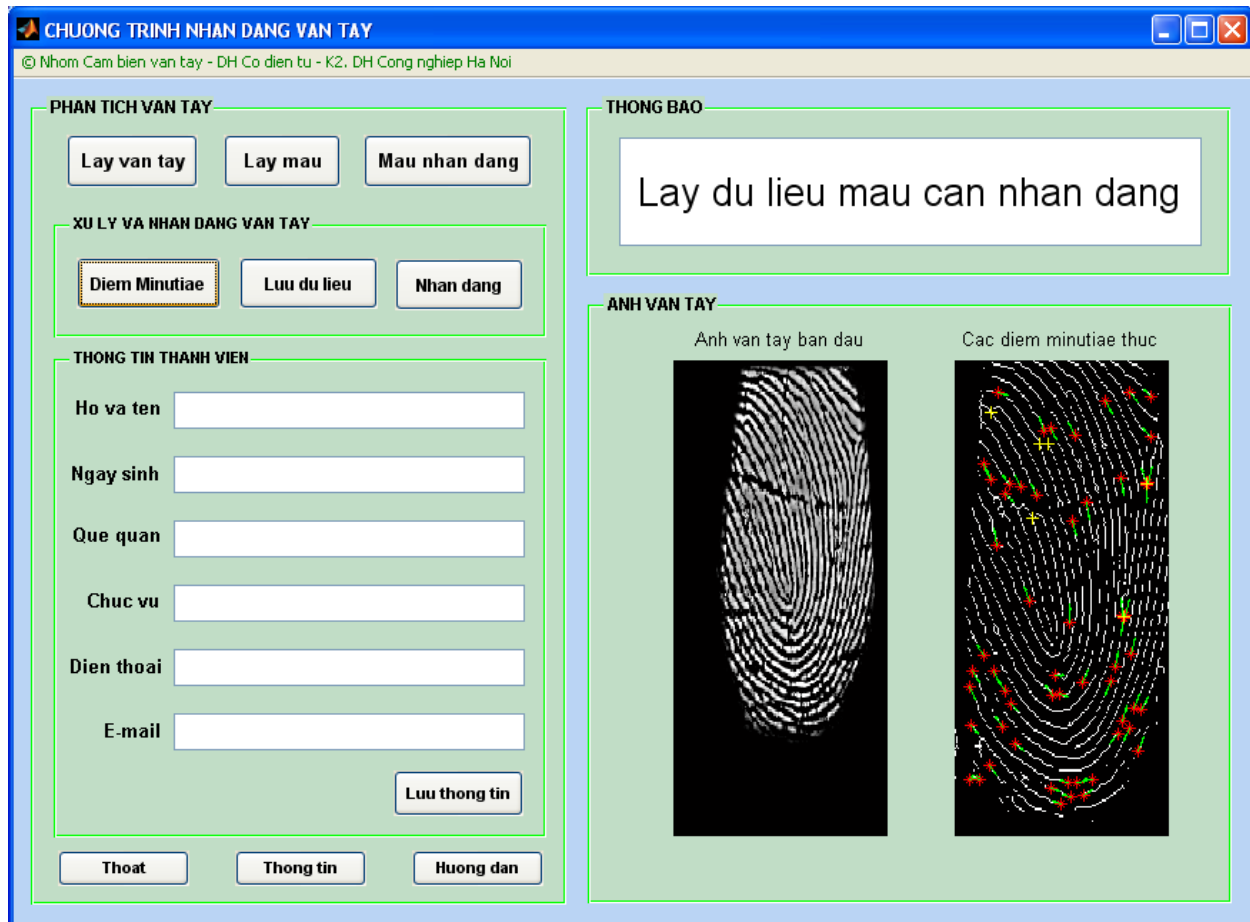
Hình 33: Quá trình lấy mẫu (tạo cơ sở dữ liệu vân tay)

- Trong phần “THÔNG TIN THÀNH VIÊN” điền các thông tin cá nhân về chủ nhân của dấu vân tay đó: họ và tên, ngày sinh, quê quán, chức vụ, điện thoại, email. Sau đó kích chuột vào nút “Luu thông tin” để lưu lại các thông tin dưới dạng file *.m trong thư mục “thông tin thành viên”.

c. Xử lý vân tay cần nhận dạng

- Kích chuột vào nút “Mau nhan dang”. Hộp thoại “Chon van tay can nhan dang” được mở ra, chọn một ảnh vân tay đã được lưu trong thư mục “van tay can nhan dang”. Ảnh vân tay được chọn sẽ hiện ra ở hình thứ nhất của phần “ANH VÂN TAY”.

- Quá trình tìm điểm Minutiae như mục a ở trên. Ở đây các file *.dat được lưu vào thư mục “du lieu can doi sanh”.



Hình 34: Quá trình xử lý vân tay cần nhận dạng – kết thúc quá trình phân tích

d. Thực hiện nhận dạng vân tay

- Kích chuột vào nút “Nhận dạng”. Hộp thoại “Chọn file dữ liệu vân tay cần nhận dạng” được mở ra, chọn một file *.dat đã được lưu ở mục b, chọn file cần nhận dạng trong thư mục “dữ liệu cần so sánh”.

Hình 35: Chương trình đang nhận dạng

Quá trình xử lý nhận dạng bắt đầu, quá trình này cũng cần mất một khoảng thời gian nhất định. Kết thúc quá trình xử lý, nếu vân tay cần nhận dạng trùng với một trong những vân tay trong cơ sở dữ liệu thì trên dòng “THÔNG BÁO” sẽ hiện “Ban la thanh vien cua nhom” và ở phần “THÔNG TIN THANH VIEN” sẽ hiện chính xác thông tin của người sở hữu vân tay đó. Ngược lại, dòng “THÔNG BÁO” sẽ hiện “Ban khong phai la thanh vien cua nhom”.

CHƯƠNG TRÌNH NHẬN DẠNG VÂN TAY

© Nhóm Cảm biến vân tay - DH Cơ điện tử - K2, DH Công nghiệp Hà Nội

PHÂN TÍCH VÂN TAY

Lay vân tay Lay mẫu Mẫu nhận dạng

XỬ LÝ VÀ NHẬN DẠNG VÂN TAY

Diem Minutiae Lưu dữ liệu Nhận dạng

THÔNG TIN THÀNH VIÊN

Họ và tên HO BA LINH

Ngày sinh 06/09/1989

Quê quán QUYNH LUU - NGHE AN

Chức vụ SINH VIEN

Điện thoại 01656085133

E-mail balinh6989@gmail.com

Lưu thông tin

THÔNG BÁO

Bạn là thành viên của nhóm

ANH VÂN TAY

Ảnh vân tay của bạn

DigitalPersona
U.are.U 4500
Fingerprint Reader

Thoát Thông tin Hướng dẫn

Hình 36: Kết thúc quá trình nhận dạng – “Bạn là thành viên của nhóm”

CHƯƠNG TRÌNH NHẬN DẠNG VÂN TAY

© Nhóm Cảm biến vân tay - DH Cơ điện tử - K2, DH Công nghiệp Hà Nội

PHÂN TÍCH VÂN TAY

Lấy vân tay Lấy mẫu Mẫu nhận dạng

XỬ LÝ VÀ NHẬN DẠNG VÂN TAY

Điểm Minutiae Lưu dữ liệu Nhận dạng

THÔNG TIN THÀNH VIÊN

Họ và tên

Ngày sinh

Quê quán

Chức vụ

Điện thoại

E-mail

Lưu thông tin

THÔNG BÁO

Bạn không phải thành viên của nhóm

ANH VÂN TAY

Matlab R2008b

The MathWorks

DigitalPersona U.are.U 4500 Fingerprint Reader

Thoát Thông tin Huong dan

Hình 37: Kết thúc quá trình nhận dạng – “Bạn là không phải thành viên của nhóm”

CHƯƠNG IV**KẾT LUẬN****1. Đánh giá kết quả và hướng phát triển của đề tài****a. Đánh giá kết quả đề tài**

Đề tài “Cảm biến nhận dạng vân tay” do nhóm Cảm biến vân tay thực hiện đã đạt được những kết quả sau:

- Tìm hiểu được các thiết bị, sản phẩm ứng dụng cảm biến quang học nói chung và cảm biến vân tay nói riêng trên thị trường. Sử dụng hiệu quả thiết bị đọc vân tay ***U.are.U 4500 Fingerprint Reader*** thể hiện qua việc kết nối với máy tính và thu thập được ảnh vân tay bằng VB6 phục vụ cho việc xử lý ảnh bằng MATLAB.
- Tìm hiểu được một số hàm và ứng dụng của nó trong công cụ xử lý ảnh và lập trình của phần mềm MATLAB.
- Nghiên cứu được đặc điểm và một số thuật toán xử lý ảnh vân tay như chuẩn hoá, tăng cường ảnh vân tay và một số thuật toán trong việc trích chọn đặc trưng vân tay như tìm ảnh định hướng, tìm ảnh nhị phân, trích chọn điểm đặc trưng Singularity và Minutiae.
- Nghiên cứu được một số kỹ thuật đối sánh vân tay như kỹ thuật dựa trên độ tương quan, kỹ thuật dựa trên điểm đặc trưng, kỹ thuật dựa trên đặc trưng vân. Trong đó chủ yếu là phương pháp đối sánh vân tay dựa trên các điểm đặc trưng.
- Viết hàm, lưu thành file *.m và thực hiện thành công một số thuật toán xử lý ảnh như chuẩn hóa, tăng cường ảnh vân và một số thuật toán trích chọn điểm đặc trưng vân tay.
- Chương trình đã có thể đối sánh hai vân tay và nhận dạng vân tay ở một mức độ nhất định.

Tuy vậy, với đề tài này, nhóm Cảm biến vẫn còn nhiều hạn chế và thiếu sót:

- Việc đối sánh, nhận dạng vân tay chỉ dựa trên một điểm đặc trưng Minutiae là... do đó mức độ chính xác và thành công đạt được là thấp.
- Cơ sở dữ liệu vân tay còn ít, tốc độ xử lý ảnh vân tay để nhận dạng chậm. Hiện tại trong cơ sở dữ liệu mới chỉ có 9x5 vân tay, tốc độ xử lý chậm nhất tới khoảng 35s.
- Chương trình chỉ chạy được trên phần mềm MATLAB, phiên bản 7.3 trở lên, không chạy được như một chương trình độc lập, do đó việc sử dụng còn phức tạp.

b. Hướng phát triển của đề tài

Với các kết quả đã đạt được cùng những hạn chế và thiếu sót trên đây đã mở ra một số hướng phát triển cho đề tài Cảm biến nhận dạng vân tay như sau:

- Nhận dạng vân tay dựa trên toàn bộ các điểm đặc trưng Singularity và Minutiae để tăng thêm độ chính xác. Đặc biệt nếu có điều kiện thì có thể áp dụng phương pháp nhận dạng dựa vào toàn bộ đặc tính vân tay.
- Nâng cao dữ liệu vân tay và tốc độ xử lý, đối sánh ảnh vân tay.

- Xây dựng một chương trình “Nhận dạng vân tay” độc lập dựa trên nền tảng MATLAB hoặc VISUAL BASIC. Cố gắng tăng tính tự động, linh hoạt trong việc lấy mẫu, lấy vân tay kiểm tra và nhận dạng, tức là một người chỉ cần ấn tay lên cảm biến là có thể lưu được các dữ liệu vân tay của mình và người kiểm tra có thể biết được nhanh chóng mình có phải là thành viên hay không chỉ qua một bước thu ảnh mà không cần qua các bước xử lý phức tạp.

2. Lời kết

Đồ án “Cảm biến nhận dạng vân tay” được triển khai và thực hiện bởi nhóm Cảm biến vân tay trong khoảng 3 tháng. Trong thời gian này, được sự đồng thuận chung tay đóng góp vật chất và trí lực của các thành viên trong nhóm cùng với sự giúp đỡ tận tình của thầy Nhữ Quý Thợ, nhóm Cảm biến đã được một số những thành công nhất định. Nhóm Cảm biến cũng nhận thấy còn phải làm nhiều công việc nữa để hoàn thiện và phát triển đề tài. Nhóm Cảm biến xin cảm ơn và rất mong nhận được sự góp ý chỉ bảo chân thành, có giá trị của các thầy cô giáo cùng các bạn.

PHỤ LỤC

1. Hàm chính GDNDVT.m

```

function varargout = GDNDVT(varargin)
%% GDNDVT M-file for GDNDVT.fig
%% Begin initialization code - DO NOT EDIT
clc; % Xoa toan bo cua so Command Window
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',   @GDNDVT_OpeningFcn, ...
                  'gui_OutputFcn',    @GDNDVT_OutputFcn, ...
                  'gui_LayoutFcn',    [] , ...
                  'gui_Callback',     []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before GDNDVT is made visible.
function GDNDVT_OpeningFcn(hObject, eventdata, handles, varargin)
%% This function has no output args, see OutputFcn.
% Choose default command line output for GDNDVT
handles.output = hObject;
% Update handles structure
guidata(hObject, handles);
subplot(121);
imagesc(imread('hinh anh\matlab.jpg')); axis off;
subplot(122);
imagesc(imread('hinh anh\thietbi.jpg')); axis off;
% UIWAIT makes GDNDVT wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = GDNDVT_OutputFcn(hObject, eventdata, handles)
%% varargout cell array for returning output args (see VARARGOUT);
% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in laymau.
function laymau_Callback(hObject, eventdata, handles)
set(handles.xinchao,'string','Tao co so du lieu van tay'); % Goi thuoc tinh
'string' cua mot trong edittext
% Mo mot cua so de lay anh (ham "uigetfile")
[filename,pathname] = uigetfile ('*.bmp;*.BMP;*.tif;*.TIF;*.jpg','Chon van
tay de nhap du lieu','van tay du lieu');
if isequal([filename,pathname],[0,0]) % Ham "isequal": can bang cac gia tri
    return
else

```

```
    anh = imread ([pathname,filename]); % Doc mot anh tu tap tin do hoa thanh
dang ma tran so
    anh = 255-double(anh); % Ham "double": tra lai gia tri chinh xac kep
end
subplot(121); % Dinh vi tri hien thi anh len o thu nhut
imagesc(anh); % Hien thi toan bo anh
axis off; % Xoa truc toa do
title('Anh van tay ban dau'); % Dat tieu de cho anh
colormap(gray); % Tao nen cho anh (chon gray: xam de nang cao su ro net cua
anh
handles.moanh = anh; % Tao truong de luu anh, nham su dung cho cac phan sau
guidata(hObject,handles); % Luu truong vao bo nho cua MATLAB
set(handles.viethoten,'string','');
set(handles.vietngaysinh,'string','');
set(handles.vietquequan,'string','');
set(handles.vietchucvu,'string','');
set(handles.vietdienthoai,'string','');
set(handles.vietemail,'string','');
clc;

% --- Executes on button press in mauthu.
function mauthu_Callback(hObject, eventdata, handles)
set(handles.xinchao,'string','Lay du lieu mau can nhan dang');
% mo mot cua so de lay anh
[filename,pathname] = uigetfile ('*.bmp;*.BMP;*.tif;*.TIF;*.jpg','Chon van
tay can nhan dang','van tay can nhan dang');
    if isequal([filename,pathname],[0,0])
        return
    else
        anh = imread ([pathname,filename]);
        anh = 255-double(anh);
    end
subplot(121);
imagesc(anh); axis off;
title('Anh van tay ban dau');
colormap(gray);
handles.moanh = anh;
guidata(hObject,handles);
set(handles.viethoten,'string','');
set(handles.vietngaysinh,'string','');
set(handles.vietquequan,'string','');
set(handles.vietchucvu,'string','');
set(handles.vietdienthoai,'string','');
set(handles.vietemail,'string','');
clc;

% --- Executes on button press in dieminutiae.
function dieminutiae_Callback(hObject, eventdata, handles)
anh = handles.moanh; % Goi anh da lay ra o "Lay mau" hoac "Mau thu" (hinh 1)
%% Nang cao chat luong anh
% Nang cao do tuong phan su dung gian luoc do xam (Enhance contrast using
histogram equalization)
% bang viec su dung ham "histeq"
anh = histeq(uint8(anh)); % Ham "uint8": chuyen doi sang dang nguyen khong
dau co 8 bit
subplot(122); % Dinh vi tri hien anh len o thu 2
```



```

imagesc(anh); title('Nang cao chat luong anh'); axis off;
pause; % Lenh tam dung, an phim bat ky de tiep tuc
% Lam noi anh bang bien doi Furie roi rac (Enhancement by FFT)
anh = biendoiffft(anh,0.45); % Xem ham biendoiffft.m
imagesc(anh); title('Bien doi Furie roi rac'); axis off;
pause;
%% Nhi phan hoa buc anh dung mot bo loc co nguong tuong thich (Adaptive
Binarization)
anh = phannguong(double(anh),32); % Xem ham phannguong.m
imagesc(anh); title('Anh nhi phan hoa'); axis off;
pause;
%% Phan vung buc anh qua hai buoc
% Xac dinh truong dinh huong (Orientation field)
[bien1,vung1] = dinhhuong(anh,16); % Xem ham dinhhuong.m
title('Xac dinh truong dinh huong'); axis off;
pause;
% Xac dinh phan vung quan trong tu truong dinh huong (Region Of Interest)
[anh,bien2,vung2] = phanvung(anh,bien1,vung1); % Xem ham phanvung.m
title('Phan vung quan trong'); axis off;
pause;
%% Lam mong cac dau van tay (Thinned-ridge map)
% Ham "bwmorph": Xu ly hinh thai hoc buc anh nhi phan
anh = im2double(bwmorph(anh,'thin',Inf)); % gia tri 'thin'. Inf: Lam mong van
tay
imagesc(anh); title('Lam mong van tay'); axis off;
pause;
% xu ly them
anh = im2double(bwmorph(anh,'clean')); % gia tri 'clean': loai bo cac phan tu
nho co lap
imagesc(anh); title('Xu ly them (clean)'); axis off;
pause;
anh = im2double(bwmorph(anh,'hbreak')); % gia tri 'hbreak': loai bo cac phan
tu duoc lien ket cao
imagesc(anh); title('Xu ly them (hbreak)'); axis off;
pause;
anh = im2double(bwmorph(anh,'spur')); % gia tri 'spur': loai bo cac phan tu
vet
imagesc(anh); title('Xu ly them (spur)'); axis off;
pause;
%% Tim tat ca cac diem minutiae
% danh dau cac diem minutia
[diemketthuc,diemrenhanh,sodovan,dorongvan] = tatcadiem(anh,vung2,16);
bieudiendiem(anh,diemketthuc,diemrenhanh);
title('Tat ca cac diem minutiae'); axis off;
pause;
%% loai bo cac diem minutiae gia tao
[sodonhanh,diemketthucthuc,diemrenhanhthuc] =
diemthucte(anh,diemketthuc,diemrenhanh,sodovan,dorongvan);
bieudiendiem(anh,diemketthucthuc,diemrenhanhthuc);
title('Cac diem minutiae thuc'); axis off;
hold off; % nen co hold off de khong luu lai du lieu anh truoc (anh huong toi
Anh nhi phan hoa)
handles.sodonhanh = sodonhanh;
guidata(hObject,handles);
handles.diemketthucthuc = diemketthucthuc;
guidata(hObject,handles);

```

```

clc;

% --- Executes on button press in sosanh.
function sosanh_Callback(hObject, eventdata, handles)
subplot(121);
imagecsc(imread('hinh anh\matlab.jpg')); axis off;
subplot(122);
imagecsc(imread('hinh anh\thietbi.jpg')); axis off;
set(handles.xinchao,'string','Xin cac ban vui long cho doi');
set(handles.viethoten,'string','');
set(handles.vietngaysinh,'string','');
set(handles.vietquequan,'string','');
set(handles.vietchucvu,'string','');
set(handles.vietdienthoai,'string','');
set(handles.vietemail,'string','');
% Lay buc anh can kiem tra, thuc ra la mot file dat luu thong tin cua anh do
[filename pathname] = uigetfile ('*.dat','Chon anh kiem tra','du lieu can doi
sanh');
if pathname ~= 0
    str = strcat(pathname,filename);
    mauvttk = load(char(str)); % Mau van tay kiem tra
    csdl = dir('co so du lieu');
    tilept = [];
    chiso = 1;
    chisolop = 1;
    tenvantay = {' '};
    dorongvan = 10;
    % Duyệt tất cả các file .dat lưu thông tin của các bức ảnh trong csdl, sau
    % đó tính phần trăm giống nhau của bức ảnh kiểm tra và các bức ảnh trong
csld
    for i = 1:1:size(csdl,1)
        tenfile = csdl(i).name;
        if
not(strcmp(tenfile, '.')|strcmp(tenfile, '..')|strcmp(tenfile, 'Thumbs.db'))
            tenfile1 = tenfile;
            tenfile1 = strcat('co so du lieu\',tenfile1);
            mauvtss = load(char(tenfile1)); % Mau van tay so sanh
            tlpt1 = doisanh(mauvttk,mauvtss,dorongvan);
            tlpt2 = doisanh(mauvtss,mauvtkt,dorongvan);
            class_name = tenfile(1:3); % class_name = '';
            p1(chisolop) = tlpt1;
            p2(chisolop) = tlpt2;
            tenvantay(chiso) = {class_name};
            tilept(chiso) = mean(p1+p2); % Hàm "mean": tính trung bình cộng
            chiso = chiso+1;
        end
    end
end
% tilept;
% tenvantay;
[tile vitri] = max(tilept); % tìm tỷ lệ phần trăm lớn nhất
if tile > 80 % Bắt đầu quá trình hiện thị kết quả: thông tin thành viên
    set(handles.xinchao,'string','Ban la thanh vien cua nhom');
    name = tenvantay{vitri};
    thongtin = dir('thong tin thanh vien');
    anhvt = dir('van tay du lieu');

```

```

for i = 1:1:size(thongtin,1)
    tttv = thongtin(i).name; % Thông tin thành viên
    if not(strcmp(tttv, '.') | strcmp(tttv, '..') | strcmp(tttv, 'Thumbs.db'))
        tttv1 = tttv(1:3);
        if strcmp(name, tttv1)
            tttv = strcat('thong tin thanh vien\', tttv);
            tt =
load(tttv, 'hoten', 'ngaysinh', 'quequan', 'chucvu', 'dienthoai', 'email');
            set(handles.viethoten, 'string', tt.hoten);
            set(handles.vietngaysinh, 'string', tt.ngaysinh);
            set(handles.vietquequan, 'string', tt.quequan);
            set(handles.vietchucvu, 'string', tt.chucvu);
            set(handles.vietdienthoai, 'string', tt.dienthoai);
            set(handles.vietemail, 'string', tt.email);
        end
    end
end
for j = 1:1:size(anhvt,1)
    ttanh = anhvt(j).name;
    if not(strcmp(ttanh, '.') | strcmp(ttanh, '..') | strcmp(ttanh, 'Thumbs.db'))
        ttanh1 = ttanh(1:3);
        if strcmp(name, ttanh1)
            ttanh = strcat('van tay du lieu\', ttanh);
            subplot(121);
            imagesc(255-double(imread(ttanh))); axis off;
            title('Anh van tay cua ban'); colormap(gray);
        end
    end
end
else
    set(handles.xinchao, 'string', 'Ban khong phai thanh vien cua nhom');
    set(handles.viethoten, 'string', '');
    set(handles.vietngaysinh, 'string', '');
    set(handles.vietquequan, 'string', '');
    set(handles.vietchucvu, 'string', '');
    set(handles.vietdienthoai, 'string', '');
    set(handles.vietemail, 'string', '');
    name = tenvantay{vitri};
    anhvt = dir('van tay can nhan dang');
end
clc;

% --- Executes on button press in luudulieu.
function luudulieu_Callback(hObject, eventdata, handles)
    sodonhanh = handles.sodonhanh;
    diemketthucthuc = handles.diemketthucthuc;
    dhlf = get(handles.xinchao, 'string'); % dhlf: dau hieu luu file vao thu muc
    "co so du lieu" hay "du lieu can so sanh"
    if strcmp(dhlf, 'Tao co so du lieu van tay') % thi se luu vao thu muc "co so
    du lieu"
    % Luu du lieu van tay
    set(handles.xinchao, 'string', 'Luu du lieu van tay');
    [filename, pathname] = uiputfile( ...
        {'*.dat'; '*.png'}, ...
        'Luu du lieu van tay', 'co so du lieu');
    % If 'Cancel' was selected then return

```

```

    if isequal([filename,pathname],[0,0])
        return
    else
% Construct the full path and save
        file = fullfile(pathname,filename);
        save(file,'diemketthucthuc','sodonhanh','-ASCII');
    end
elseif strcmp(dhlf,'Lay du lieu mau can nhan dang') % thi se luu vao thu muc
"du lieu can so sanh"
    set(handles.xinchao,'string','Luu du lieu van tay');
    [filename,pathname] = uiputfile( ...
        {'*.dat';'*.*'}, ...
        'Luu du lieu van tay','du lieu can doi sanh');
% If 'Cancel' was selected then return
    if isequal([filename,pathname],[0,0])
        return
    else
% Construct the full path and save
        file = fullfile(pathname,filename);
        save(file,'diemketthucthuc','sodonhanh','-ASCII');
    end
end
clc;

% --- Executes on button press in luuthongtin.
function luuthongtin_Callback(hObject, eventdata, handles)
set(handles.xinchao,'string','Luu thong tin cua thanh vien');
hoten = get(handles.viethoten,'string');
ngaysinh = get(handles.vietngaysinh,'string');
quequan = get(handles.vietquequan,'string');
chucvu = get(handles.vietchucvu,'string');
dienthoai = get(handles.vietdienthoai,'string');
email = get(handles.vietemail,'string');
% Luu thong tin thanh vien co dau van tay do
[filename,pathname] = uiputfile( ...
    {'*.mat';'*.*'}, ...
    'Luu thong tin thanh vien','thong tin thanh vien');
% If 'Cancel' was selected then return
if isequal([filename,pathname],[0,0])
    return
else
% Construct the full path and save
    file = fullfile(pathname,filename);
    save(file,'hoten','ngaysinh','quequan','chucvu','dienthoai','email');
end
clc;

% --- Executes on button press in thoat.
function thoat_Callback(hObject, eventdata, handles)
Answer = questdlg('Ban co that su muon thoat khoi chuong trinh?', ...
    'Thoat', ...
    'Yes','No','Yes');
switch Answer
case 'Yes'
    close all;
case 'No'

```

```
        return
    end
    clc;

    % --- Executes on button press in thongtin.
    function thongtin_Callback(hObject, eventdata, handles)
        thongtin;

    % --- Executes on button press in huongdan.
    function huongdan_Callback(hObject, eventdata, handles)
        hdsudung;

    function viethoten_Callback(hObject, eventdata, handles)

    % --- Executes during object creation, after setting all properties.
    function viethoten_CreateFcn(hObject, eventdata, handles)
        if ispc && isequal(get(hObject, 'BackgroundColor'),
            get(0, 'defaultUicontrolBackgroundColor'))
            set(hObject, 'BackgroundColor', 'white');
        end

    function vietngaysinh_Callback(hObject, eventdata, handles)

    % --- Executes during object creation, after setting all properties.
    function vietngaysinh_CreateFcn(hObject, eventdata, handles)
        if ispc && isequal(get(hObject, 'BackgroundColor'),
            get(0, 'defaultUicontrolBackgroundColor'))
            set(hObject, 'BackgroundColor', 'white');
        end

    function vietquequan_Callback(hObject, eventdata, handles)

    % --- Executes during object creation, after setting all properties.
    function vietquequan_CreateFcn(hObject, eventdata, handles)
        if ispc && isequal(get(hObject, 'BackgroundColor'),
            get(0, 'defaultUicontrolBackgroundColor'))
            set(hObject, 'BackgroundColor', 'white');
        end

    function vietchucvu_Callback(hObject, eventdata, handles)

    % --- Executes during object creation, after setting all properties.
    function vietchucvu_CreateFcn(hObject, eventdata, handles)
        if ispc && isequal(get(hObject, 'BackgroundColor'),
            get(0, 'defaultUicontrolBackgroundColor'))
            set(hObject, 'BackgroundColor', 'white');
        end

    function vietdienthoai_Callback(hObject, eventdata, handles)

    % --- Executes during object creation, after setting all properties.
    function vietdienthoai_CreateFcn(hObject, eventdata, handles)
        if ispc && isequal(get(hObject, 'BackgroundColor'),
            get(0, 'defaultUicontrolBackgroundColor'))
            set(hObject, 'BackgroundColor', 'white');
```

```
end

function vietemail_Callback(hObject, eventdata, handles)

    % --- Executes during object creation, after setting all properties.
    function vietemail_CreateFcn(hObject, eventdata, handles)
    if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

function xinchao_Callback(hObject, eventdata, handles)

    % --- Executes during object creation, after setting all properties.
    function xinchao_CreateFcn(hObject, eventdata, handles)
    if ispc && isequal(get(hObject,'BackgroundColor'),
    get(0,'defaultUicontrolBackgroundColor'))
        set(hObject,'BackgroundColor','white');
    end

% -----
function tag1_Callback(hObject, eventdata, handles)
clc;
% -----

% --- Executes on button press in lavantay.
function layvantay_Callback(hObject, eventdata, handles)
open ctlavt.exe;
clc;
```

2. Danh sách nhóm Cảm biến nhận dạng vân tay

1. Phan Thanh Tùng (Nhóm trưởng)
2. Hồ Bá Linh (Nhóm phó)
3. Tạ Văn Toàn (Thành viên)
4. Nguyễn Thị Hường (Thành viên)
5. Nguyễn Thị Đào (Thành viên)
6. Nguyễn Mạnh Cường (Thành viên)
7. Bùi Minh Đức (Thành viên)
8. Nguyễn Văn Luật (Thành viên)
9. Nguyễn Thị Huế (Thành viên)

TÀI LIỆU THAM KHẢO

1. Khóa luận tốt nghiệp: Nghiên cứu và phát triển ứng dụng nhận dạng vân tay. (Nguyễn Đức Luân – ĐH Công nghệ. ĐH Quốc gia Hà Nội – 2006).
2. Luận văn thạc sỹ: Nhận dạng vân tay. (Nguyễn Hoàng Duy – ĐH Bách khoa. ĐH Quốc gia TP. Hồ Chí Minh – 2010).
3. An Introduction to Digital Image Processing with Matlab. (Alasdair McAndrew - School of Computer Science and Mathematics. Victoria University of Technology - Semester 1.2004).
4. BioKey Developer's Guide (Hãng ZKSoftware).
5. Product Help MATLAB 7.7.0 R2008b (© 1984-2008 The MathWorks, Inc).