

BỘ GIAO THÔNG VẬN TẢI BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC HÀNG HẢI VIỆT NAM



NGUYỄN THỊ THANH VÂN

**NGHIÊN CỨU NHẬN DẠNG BIỂU CẢM MẶT
NGƯỜI TRONG TƯƠNG TÁC NGƯỜI MÁY**

LUẬN VĂN THẠC SĨ NGÀNH KỸ THUẬT

HẢI PHÒNG - 2016

BỘ GIAO THÔNG VẬN TẢI BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC HÀNG HẢI VIỆT NAM

NGUYỄN THỊ THANH VÂN

**NGHIÊN CỨU NHẬN DẠNG BIỂU CẢM MẶT
NGƯỜI TRONG TƯƠNG TÁC NGƯỜI MÁY**

LUẬN VĂN THẠC SĨ NGÀNH KỸ THUẬT

NGÀNH: CÔNG NGHỆ THÔNG TIN MÃ SỐ: 60480201

CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN

Người hướng dẫn khoa học: TS. Hồ Thị Hương Thơm

HẢI PHÒNG - 2016

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Các số liệu, kết quả nêu trong Luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Tôi xin cam đoan rằng các thông tin trích dẫn trong Luận văn đã được chỉ rõ nguồn gốc.

Học viên

Nguyễn Thị Thanh Vân

LỜI CẢM ƠN

Tôi xin chân thành gửi lời cảm ơn sâu sắc đến các thầy cô khoa Công nghệ thông tin – Trường Đại học Hàng Hải đã giảng dạy và hướng dẫn tôi trong quá trình học tập tại trường. Đặc biệt là TS. Hồ Thị Hương Thơm đã tận tình hướng dẫn và giúp đỡ tôi trong suốt quá trình thực hiện đề tài:

“NGHIÊN CỨU NHẬN DẠNG BIỂU CẢM MẶT NGƯỜI TRONG TƯƠNG TÁC NGƯỜI MÁY”.

Tôi xin chân thành cảm ơn các thầy cô trong Viện đào tạo sau Đại Học- Đại học Hàng Hải, đã tạo điều kiện tốt nhất cho tôi được học tập và nghiên cứu tại trường.

Cuối cùng, tôi muốn gửi lời cảm ơn vô hạn tới gia đình, bạn bè và những người thân yêu luôn bên cạnh, động viên tôi trong suốt quá trình thực hiện luận văn tốt nghiệp.

Xin chân thành cảm ơn!

Học viên

Nguyễn Thị Thanh Vân

MỤC LỤC

LỜI CAM ĐOAN.....	i
LỜI CẢM ƠN	ii
MỤC LỤC	iii
DANH MỤC CÁC CHỮ VIẾT TẮT VÀ KÝ HIỆU	vi
DANH MỤC CÁC BẢNG.....	vii
DANH MỤC CÁC HÌNH	viii
LỜI MỞ ĐẦU	1
CHƯƠNG 1. ĐẶC TRƯNG CỦA MẶT NGƯỜI VÀ BIỂU CẢM KHUÔN MẶT	3
1.1 Đặc trưng của mặt người	3
1.1.1 Lời giới thiệu	3
1.1.2 Phương pháp tiếp cận đặc trưng khuôn mặt.....	6
1.2 Biểu cảm khuôn mặt	6
1.2.1 Các loại cảm xúc	6
1.2.2 Nhận dạng cảm xúc bằng nét mặt	7
1.3 Một số phương pháp phát hiện biểu cảm khuôn mặt	8
1.3.1 Một số phương pháp dựa trên đặc trưng của mặt.....	8
1.3.2 Phương pháp sử dụng Action Units	9
1.3.3 Phương pháp dùng mô hình AAM kết hợp tương quan điểm.....	10
1.3.4 Mô hình tổng quan	11
1.3.5 Thách thức trong vấn đề nhận dạng cảm xúc khuôn mặt.....	11
1.3.6 Một số vấn đề khác.....	11
CHƯƠNG 2. MỘT SỐ PHƯƠNG PHÁP PHÁT HIỆN BIỂU CẢM KHUÔN MẶT	14
2.1 Cơ sở lý thuyết	14
2.1.1 Giới thiệu về mạng Noron	14
2.1.1.1 Mạng Perceptron nhiều tầng (MPL – Multi Perceptron Layer)	15

2.1.1.2	Ánh xạ lan truyền tiến.....	15
2.1.1.3	Hàm sigmoid.....	18
2.1.1.4	Thuật toán lan truyền ngược	19
2.1.2	Giới thiệu về PCA.....	25
2.1.2.1	Một số khái niệm toán học:	25
2.1.2.2	Ma trận đại số	29
2.1.2.3	Eigenvector (Vector riêng)	29
2.1.2.4	Eigenvalue (Giá trị riêng)	30
2.1.2.5	Phân tích thành phần chính của PCA	30
2.1.3	Giới thiệu về Eigenface.....	31
2.1.3.1	Tính Eigenface	32
2.1.3.2	Sử dụng khuôn mặt đặc trưng để phân loại hình ảnh khuôn mặt	35
2.1.3.3	Tóm tắt phương pháp nhận dạng đặc trưng khuôn mặt.....	39
2.1.3.4	Tìm và phát hiện khuôn mặt	40
2.1.3.5	Các vấn đề khác	44
2.2	Phương pháp phát hiện biểu cảm khuôn mặt.....	46
2.2.1	Nhận dạng cảm xúc dựa trên phương pháp PCA	47
2.2.2	Nhận dạng cảm xúc dựa trên PCA kết hợp các thuật toán học	49
CHƯƠNG 3: CÀI ĐẶT VÀ THỬ NGHIỆM.....		52
3.1	Môi trường cài đặt.....	52
3.2	Giao diện chương trình	52
3.2.1	Tạo tập ảnh cơ sở	53
3.2.2	Nhận dạng biểu cảm.....	54
3.2.3	Ứng dụng để chụp ảnh	56
3.3	Thử nghiệm và đánh giá.....	56
3.3.1	Thử nghiệm	56
3.3.2	Đánh giá	57
3.3.2.1	Phương pháp PCA truyền thống.....	57
3.3.2.2	Phương pháp sử dụng mạng nơron.....	58
3.3.2.3	Phương pháp sử dụng cây quyết định.....	58
3.3.2.4	Tổng kết	59

KẾT LUẬN	60
TÀI LIỆU THAM KHẢO.....	61

DANH MỤC CÁC CHỮ VIẾT TẮT VÀ KÝ HIỆU

<i>Chữ viết tắt</i>	<i>Giải thích</i>
PCA	Viết tắt của thuật ngữ “ <i>Principal Component Analysis</i> ”
Input/hidden/output layer	Lớp đầu vào/ẩn/ đầu ra
MLP	“ <i>MultiLayer Perceptrons</i> ”: Mạng noron nhiều tầng truyền thẳng
MIT	Massachusetts Institute of Technology
AU	Action Units
JAFFE	Japanese Female Facial Expression
AAM	Active Appearance Model
MPL	Multi Perceptron Layer
Perceptron	Bộ phân loại tuyến tính cho phân loại dữ liệu
Hàm Sigmoid	Là trường hợp đặc biệt của hàm lôgit
EigenFace	Đặc trưng khuôn mặt
EigenValue	Trị riêng
EigenVector	Vector riêng
Gradient descent	tiến dần tới cực tiểu địa phương
Happy	Hạnh phúc
Disgust	Ghê tởm
Sad	Buồn
Anger	Giận dữ
Neutral	Trung lập

DANH MỤC CÁC BẢNG

<i>Số bảng</i>	<i>Tên bảng</i>	<i>Trang</i>
2.1	Tập dữ liệu 1	26
2.2	Tập dữ liệu 2	27

DANH MỤC CÁC HÌNH

<i>Số hình</i>	<i>Tên hình</i>	<i>Trang</i>
1.1	Tập các khuôn mặt biểu cảm đặc trưng	9
1.2	Các đặc trưng AU	9
1.3	Các điểm đánh dấu trên khuôn mặt	10
1.4	Mô hình nhận dạng cảm xúc	11
1.5	Biểu cảm khuôn mặt trong không gian 3D	12
1.6	Cảm xúc trong video	13
2.1	Mô hình mạng lan truyền tiến	16
2.2	Đồ thị hàm truyền sigmoid	19
2.3	Lan truyền ngược	21
2.4	Minh họa việc tính δ_j cho việc tính nút ẩn j	24
2.5	Ví dụ về non-eigenvector và eigenvector	29
2.6	Ví dụ về eigenvector có tỉ lệ khác vẫn là eigenvector	29
2.7	Các hình ảnh khuôn mặt được sử dụng làm tập huấn luyện	32
2.8	Khuôn mặt trung bình	34
2.9	Bảy trong số các khuôn mặt đặc trưng được tính từ những hình ảnh đầu vào của hình 2.7	35
2.10	Hình ảnh khuôn mặt ban đầu và hình chiếu của nó lên không gian mặt được xác định bởi các eigenfaces của hình 2.9	36
2.11	Ba ảnh và hình chiếu của nó vào không gian mặt, được xác định bởi các eigenface.	38
2.12	Dò tìm vị trí của đầu và hệ thống định vị	41
2.13	(a) Ảnh ban đầu . (b) Bản đồ mặt tương ứng.	42

2.14	Vị trí của đầu đã được xác định	43
2.15	Năm vùng của khuôn mặt được xem xét trong nghiên cứu này	46
2.16	Hai thành phần Đầu tiên của vector vùng mặt thấp	47
2.17	Ví dụ về trích chọn đặc trưng bằng PCA	47
	Hình ảnh đại diện bởi tập các giá trị riêng	48
	Mô hình mạng nơron	49
	Cây quyết định	51
3.1	Giao diện chính của chương trình	53
3.2	Tạo tập huấn luyện ảnh ban đầu	54
3.3	Nhận dạng và hiển thị kết quả	55
3.4	Ứng dụng của nhận dạng biểu cảm	56
3.5	Tập huấn luyện ảnh khuôn mặt	57

LỜI MỞ ĐẦU

1. Tính cấp thiết của đề tài

Trong những năm gần đây, cùng với sự phát triển của khoa học công nghệ, tương tác người máy đã trở thành một lĩnh vực nổi bật nhằm cung cấp cho con người khả năng phục vụ máy móc. Điều này bắt nguồn từ khả năng máy móc có thể tương tác được với con người. Máy móc cần các kỹ thuật để có thể trao đổi thông tin với con người và một trong các kỹ năng đó là khả năng hiểu được cảm xúc. Cách tốt nhất để một người biểu thị cảm xúc là qua khuôn mặt.

Ứng dụng của nhận dạng cảm xúc trong cuộc sống hàng ngày là rất lớn, các hệ thống phát hiện trạng thái buồn ngủ dựa vào cảm xúc trên khuôn mặt được phát triển để cảnh báo cho người lái xe khi thấy dấu hiệu buồn ngủ, mệt mỏi. Các hệ thống kiểm tra tính đúng đắn của thông tin, các phần mềm điều kiện dựa vào cảm xúc, các thiết bị hỗ trợ người tàn tật...

2. Mục tiêu của đề tài

Mục tiêu của đề tài là nghiên cứu 1 số phương pháp nhận dạng cảm xúc khuôn mặt dựa trên ảnh tĩnh và trực diện.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu của đề tài: Nghiên cứu một số kỹ thuật nhận dạng biểu cảm khuôn mặt như: Sử dụng các đơn vị vận động trên khuôn mặt (Action Units – AU), sử dụng PCA, mô hình AAM kết hợp tương quan điểm.

Phạm vi nghiên cứu: dựa trên cơ sở lý thuyết của một số phương pháp nghiên cứu tìm hiểu được, áp dụng xây dựng chương trình nhận dạng biểu cảm khuôn mặt dựa trên ảnh tĩnh (lưu dưới dạng BITMAP), hoặc trên ảnh chụp từ webcam

4. Phương pháp nghiên cứu

Phương pháp nghiên cứu của đề tài là nghiên cứu cơ sở lý thuyết của một số phương pháp phát hiện biểu cảm khuôn mặt. Dựa trên nền tảng lý thuyết đó xây dựng một chương trình thử nghiệm với ảnh tĩnh hoặc video, từ đó đưa ra một số nhận xét đánh giá của phương pháp tìm hiểu được để ứng dụng trong thực tế.

5. Dự kiến kết quả đạt được

Đưa ra được kết quả nghiên cứu về mặt lý thuyết một số cơ sở của phương pháp nhận dạng biểu cảm khuôn mặt, xây dựng được chương trình với một số kỹ thuật biểu cảm khuôn mặt dựa trên đầu vào là ảnh tĩnh hoặc video, từ đó nêu ra một số vấn đề ứng dụng trong thực tế.

CHƯƠNG 1. ĐẶC TRƯNG CỦA MẶT NGƯỜI VÀ BIỂU CẢM KHUÔN MẶT

1.1 Đặc trưng của mặt người

1.1.1 Lời giới thiệu

Khuôn mặt là trọng tâm chính trong mối quan hệ giao tiếp trong xã hội, đóng vai trò quan trọng trong việc truyền tải bản sắc và cảm xúc. Chúng ta có thể nhận ra khuôn mặt của rất nhiều người trong suốt cuộc đời, và việc nhận diện khuôn mặt quen thuộc chỉ trong nháy mắt thậm chí sau nhiều năm không gặp. Điều này khá là rõ nét, bất chấp những thay đổi lớn về thị giác, biểu hiện, lão hóa, hoặc những thay đổi về kiểu tóc, về kính... Ngoài ra, ảnh khuôn mặt trong thực tế còn chứa đựng rất nhiều vấn đề như: độ sáng, độ nhòe/mờ, độ nhiễu, độ phân giải, góc ảnh...

Mô hình tính toán nhận dạng cảm xúc khuôn mặt là một hướng nghiên cứu được rất nhiều nhà khoa học lớn quan tâm, và có nhiều ứng dụng trong thực tế. Máy tính nhận diện biểu cảm khuôn mặt có thể được áp dụng cho một loạt các vấn đề bao gồm cả việc xác định tội phạm, hệ thống an ninh, hình ảnh, xử lý phim và sự tương tác của con người với máy tính. Ở các trường Đại học hàng đầu về Công Nghệ Thông Tin như *MIT* (Massachusetts Institute of Technology), *Stanford*, *Berkeley* và các công ty lớn như Microsoft, Apple, Google, Facebook đều có các trung tâm về sinh trắc học (*Biometric Center*) và nghiên cứu về nhận dạng biểu cảm khuôn mặt là một trong những lĩnh vực nghiên cứu chính cho đến nay.

Luận văn dựa trên một phương pháp tiếp cận lý thuyết thông tin, phân tích hình ảnh khuôn mặt, sau đó đưa những hình ảnh đó vào một tập hợp nhỏ các ảnh có tính năng đặc trưng gọi là “eigenfaces”, đây chính là tập huấn luyện ban đầu của hình ảnh khuôn mặt. Quá trình nhận dạng được thực hiện bằng cách chiếu một hình ảnh mới vào không gian con mở rộng bởi các eigenfaces (“không gian mặt”), và sau đó phân loại mặt bằng cách so sánh vị trí của nó trong không gian mặt với các vị trí được biết của khuôn mặt.

Tự động học và sau đó nhận ra các gương mặt mới là mục đích chính để phát hiện ra biểu cảm khuôn mặt. Nhận dạng trong các điều kiện khác nhau được thực hiện bằng cách huấn luyện trên một số giới hạn các điểm đặc trưng như: nhìn thẳng, nhìn một góc 45 độ và nhìn nghiêng. Cách tiếp cận này có lợi thế hơn các chương trình nhận dạng khuôn mặt khác ở tốc độ, sự đơn giản, khả năng học và sự ổn định với những thay đổi nhỏ hoặc dần dần trong hình ảnh khuôn mặt.

Nhiều phương pháp trong nhận dạng khuôn mặt đã tập trung vào phát hiện các tính năng riêng biệt như mắt, mũi, miệng, đầu và xác định một số kiểu khuôn mặt bằng vị trí, kích thước, và mối quan hệ. Kết quả nhận dạng của nhiều hệ thống cũng rất khác nhau, tùy thuộc vào dữ liệu kiểm thử. Trong khuôn khổ của luận văn này, một phương pháp rất nổi tiếng và kinh điển đó là *EigenFace* được nghiên cứu và tìm hiểu sâu, từ đó phát hiện ra biểu cảm khuôn mặt rõ nét hơn. Ngoài ra, giải pháp dùng eigen (bao gồm eigenvalue, eigenvector) để tìm ra không gian con *PCA* (Principal Component Analysis) đóng vai trò rất quan trọng trong lĩnh vực thị giác máy tính.

Vào năm 1966, Bledsoe là người đầu tiên đã thử xây dựng chương trình nhận dạng khuôn mặt tự động kết hợp với hệ thống máy tính, bằng cách phân loại khuôn mặt trên cơ sở mốc chuẩn được nhập vào bằng tay. Các thông số để phân loại là khoảng cách chuẩn, tỉ lệ giữa các điểm như góc, mắt, miệng, chóp mũi và chóp cằm. Sau này, tại Bell Labs đã phát triển một vector với 21 tính năng, và khuôn mặt được phát hiện bằng cách sử dụng kỹ thuật phân loại tiêu chuẩn mẫu. Các tính năng được lựa chọn đánh giá chủ yếu là: màu tóc, chiều dài của đôi tai, độ dày môi...

Bài báo đầu tiên được viết bởi Fischler và Elschlager (1973) tự động phát hiện ra các đặc tính. Họ đã mô tả một thuật toán có sử dụng mẫu đặc trưng cục bộ kết hợp với một thông số phù hợp để tìm và xác định đặc điểm khuôn mặt. Cách tiếp cận này đã được tiếp tục phát triển bởi Yuille, Cohen, và Hallinan (1989). Chiến lược của họ là dựa trên "các mẫu bất biến", đó là mô hình tham số của

khuôn mặt và tính năng của nó, trong đó các giá trị tham số được xác định bởi sự tương tác với ảnh. Kohonen (1989) và Lahtio (1981) đã mô tả một mạng lưới liên kết với một thuật toán đơn giản có thể nhận ra (phân loại) hình ảnh khuôn mặt, và gọi lại một hình ảnh khuôn mặt từ hình ảnh đầu vào chưa đầy đủ hoặc bị nhiễu. Fleming và Cottrell (1990) mở rộng các ý tưởng sử dụng các đơn vị phi tuyến tính, huấn luyện hệ thống bằng cách lan truyền ngược. Hệ thống WISARD Stonham (1986) là một thiết bị nhận dạng mẫu học có mục đích chung dựa trên các nguyên tắc mạng nơron. Nó được áp dụng thành công với một số ảnh khuôn mặt nhị phân, nhận dạng được cả tình trạng và cách biểu cảm. Hầu hết các hệ thống phân chia khuôn mặt xử lý hình ảnh đầu vào như là một mô hình 2-D nói chung, và có thể không cần sử dụng các cấu hình thuộc tính của khuôn mặt. Hơn nữa, các hệ thống này đòi hỏi một số lượng quá mức các ví dụ huấn luyện để đạt được một hiệu suất như mong muốn.

Nhiều nhà khoa học đã tiếp cận nhận dạng khuôn mặt bằng cách mô tả mặt bởi một tập hợp các thông số hình học, và thực hiện mô hình nhận dạng dựa trên các thông số đó, (ví dụ: Kaya & Kobayashi, 1972; Cannon, Jones, Campbell, & Morgan, 1986; Craw, Ellis, & Lishman, 1987; Wong, Law, & Tsaug, 1989). Kanade's (1973). Hệ thống tính toán một tập các tham số trên khuôn mặt từ một hình ảnh khuôn mặt đơn và sử dụng một kỹ thuật phân loại để tìm khuôn mặt phù hợp từ tập đã có. Cách tiếp cận này hoàn toàn phụ thuộc chủ yếu vào phân tích biểu đồ cục bộ và các tỉ lệ màu xám.

Nghiên cứu gần đây của Burt (1988) sử dụng một phương pháp tiếp cận “cảm biến thông minh” dựa trên đa giải pháp. Cách này sử dụng một máy tính đặc biệt để tính toán hình chớp một cách nhanh chóng, và được chứng minh kết quả xác định là tức thời. Hệ thống này hoạt động tốt trong những điều kiện hạn chế, nhưng phải chịu một số vấn đề về mối quan hệ tương quan, bao gồm cả độ nhiễu và kích thước của ảnh.

1.1.2 Phương pháp tiếp cận đặc trưng khuôn mặt

Nhiều nghiên cứu trước đây về nhận diện khuôn mặt thường bỏ qua vấn đề về diện mạo khuôn mặt, tuy nhiên đây là điều rất quan trọng để nhận biết biểu cảm khuôn mặt. Điều này gợi ý cho chúng ta rằng cách tiếp cận lý thuyết thông tin mã hóa và giải mã hình ảnh khuôn mặt, có thể cung cấp thông tin chính xác về ảnh khuôn mặt, nhấn mạnh sự quan trọng của vùng và tính năng toàn cục. Tính năng này có thể hoặc không liên quan trực tiếp đến khái niệm về trực giác của khuôn mặt như: mắt, mũi, môi và tóc.

Trong ngôn ngữ của lý thuyết thông tin, chúng ta muốn trích xuất các thông tin có liên quan đến hình ảnh khuôn mặt, mã hóa nó một cách hiệu quả nhất có thể, và so sánh khuôn mặt được mã hóa với một cơ sở dữ liệu tương tự đã được mã hóa. Một phương pháp đơn giản để giải nén các thông tin chứa hình ảnh của khuôn mặt là nắm bắt được sự thay đổi trong tập các hình ảnh khuôn mặt, tính đối lập của các trạng thái biểu cảm, sử dụng nó để mã hóa và so sánh với hình ảnh khuôn mặt khác.

Mặt khác, chúng ta tìm các thuộc tính chủ yếu của khuôn mặt, hoặc các vector đặc trưng riêng của ma trận phương sai của tập các hình ảnh khuôn mặt, xử lý hình ảnh đó như là một điểm (hay vector) trong không gian đa chiều. Các vector đặc trưng riêng được sắp xếp, mỗi một vector đó có sự biến đổi khác nhau giữa các hình ảnh khuôn mặt.

Các vector đặc trưng có thể coi như một tập hợp các đại lượng đặc trưng cho sự biến đổi giữa các hình ảnh khuôn mặt. Mỗi vùng ảnh góp phần cho mỗi vector đặc trưng, bởi vậy có thể hiển thị các vector đặc trưng như một ma trận mà chúng ta gọi là eigenface.

1.2 Biểu cảm khuôn mặt

1.2.1 Các loại cảm xúc

Sự tương tác giữa con người và máy tính sẽ tự nhiên hơn nếu máy tính có thể nhận biết và đáp ứng giao tiếp phi ngôn ngữ của con người như cảm xúc. Cảm

xúc của con người có thể phân loại thành năm loại cảm xúc: hạnh phúc, ghê tởm, giận dữ, buồn và trung lập. Trạng thái chuyển động của khuôn mặt đóng một vai trò quan trọng trong việc thể hiện những cảm xúc này. Các cơ mặt có thể thay đổi, và con người có thể nhận ra những tín hiệu ngay cả khi chúng được hiển thị một cách tinh tế, đồng thời xử lý thông tin có được bằng tai và mắt. Dựa trên các nghiên cứu tâm lý, kết luận rằng một số cảm xúc được xác định tốt hơn với âm thanh như buồn và sợ hãi, và những người khác thì với video, chẳng hạn như sự giận dữ và hạnh phúc.

1.2.2 Nhận dạng cảm xúc bằng nét mặt

Nét mặt cung cấp các đầu mối quan trọng về cảm xúc. Do đó, một số phương pháp đã được đề xuất để nhận biết các trạng thái cảm xúc của con người. Các tính năng được sử dụng thường dựa trên vị trí không gian cục bộ hoặc di chuyển các điểm cụ thể và vùng của khuôn mặt, không giống như các phương pháp tiếp cận dựa trên âm thanh, sử dụng số liệu thống kê toàn diện của âm thanh.

Mase đề xuất một hệ thống nhận dạng cảm xúc sử dụng chính các chuyển động của cơ mặt. Với 11 cửa sổ nằm ở mặt, các cử động cơ được chiết xuất bằng cách sử dụng dòng chảy quang. Đối với phân loại K-láng giềng gần nhất, quy tắc này đã được sử dụng với độ chính xác tới 80% với năm loại cảm xúc: hạnh phúc, ghê tởm, giận dữ, buồn và trung lập. *Yacoob* và cộng sự đề xuất một phương pháp tương tự, thay vì sử dụng hoạt động trên cơ mặt, họ đã xây dựng một từ điển để chuyển đổi chuyển động kết hợp với khóe của miệng, mắt và lông mày. Họ phân loại năm cảm xúc cơ bản, sử dụng hệ thống dựa trên nguyên tắc với độ chính xác là 88%.

Black và cộng sự sử dụng mô hình tham số để trích xuất các hình dạng và chuyển động của miệng, mắt và lông mày. Nhóm cũng xây dựng một đại lượng trung và cấp cao của các hoạt động trên khuôn mặt bằng cách sử dụng một cách tiếp cận tương tự với độ chính xác 89%. Nhóm cộng sự *Tian* cố gắng để nhận ra đơn vị hành động (AU), sử dụng tạm thời các tính năng trên khuôn mặt như môi, mắt và nếp nhăn.

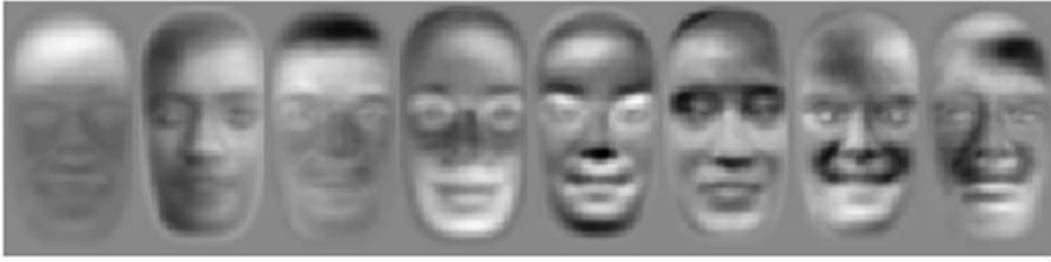
Mô hình này được sử dụng để xác định vị trí các hình dạng và sự xuất hiện của các tính năng trên khuôn mặt, đã đạt được 96% độ chính xác. Essa và cộng sự, phát triển một hệ thống định lượng trên khuôn mặt dựa trên các mô hình tham số của các nhóm cơ trên khuôn mặt độc lập. Họ đã mô hình hóa khuôn mặt bằng cách sử dụng một phương pháp dòng quang cùng với hình học, vật lý và chuyển động dựa trên mô hình, tạo ra các mẫu không gian-thời gian được sử dụng. Trong luận văn này, việc phát hiện các đặc tính trên khuôn mặt được thực hiện bằng cách đánh dấu.

1.3 Một số phương pháp phát hiện biểu cảm khuôn mặt

Hiện nay có rất nhiều phương pháp phát hiện biểu cảm khuôn mặt, phổ biến là một số phương pháp sau: Sử dụng các đơn vị vận động trên khuôn mặt (Action units – AU), sử dụng PCA, AAM kết hợp tương quan điểm, sử dụng các phương pháp học,... Mỗi phương pháp đều có ưu và nhược điểm riêng. Đối với các phương pháp sử dụng PCA kết hợp mạng nơron, cần một tập dữ liệu chuẩn để huấn luyện. Việc xây dựng các tập huấn luyện này cũng tương đối khó khăn và tốn kém vì cần nhiều người làm mẫu, những người này phải có khả năng diễn đạt cảm xúc tốt, ngoài ra còn cần sự đánh giá của các chuyên gia tâm lý. Hiện nay có một số tập huấn luyện chuẩn thường được dùng như JAFFE (Japanese Female Facial Expression) hay Cohn-kanade.

1.3.1 Một số phương pháp dựa trên đặc trưng của mặt

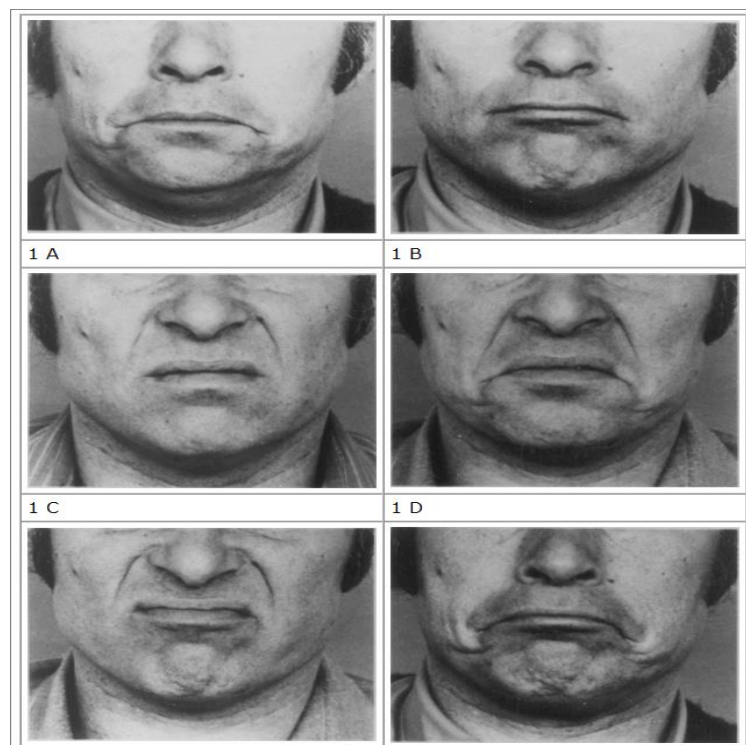
Các kỹ thuật sử dụng trong phương pháp này là phân tích thành phần chính PCA, sau đó huấn luyện bằng các thuật toán học. PCA được Karl Pearson tạo ra năm 1901. Những năm 80, Sirovich và Kirby đã phát triển kỹ thuật này để thể hiện khuôn mặt một cách hiệu quả, đưa ra sự giống nhau giữa nhiều hình ảnh khuôn mặt khác nhau, kỹ thuật này tìm ra những thành phần cơ bản của sự phân bố trên khuôn mặt, thể hiện bằng các vector đặc trưng. Từng khuôn mặt trong một tập hợp các khuôn mặt sau đó có thể tính xấp xỉ bằng sự kết hợp tuyến tính giữa những vector đặc trưng lớn nhất, được biết tới như là đặc trưng khuôn mặt.



Hình 1.1. Tập các khuôn mặt biểu cảm đặc trưng

1.3.2 Phương pháp sử dụng Action Units

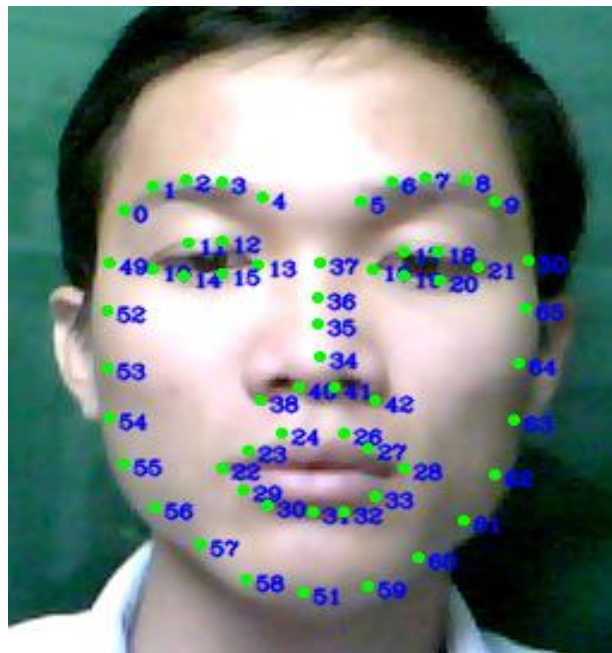
Phương pháp này nhận dạng cảm xúc dựa trên các đơn vị chuyển động của khuôn mặt (AU). Có tất cả 64 AU, mỗi AU là sự kết hợp của một số cơ trên khuôn mặt. Cảm xúc được nhận dạng bằng cách phát hiện tại một thời điểm có bao nhiêu AU xuất hiện trên khuôn mặt và với các AU xuất hiện cùng nhau tương ứng với một cảm xúc.



Hình 1.2. Các đặc trưng AU

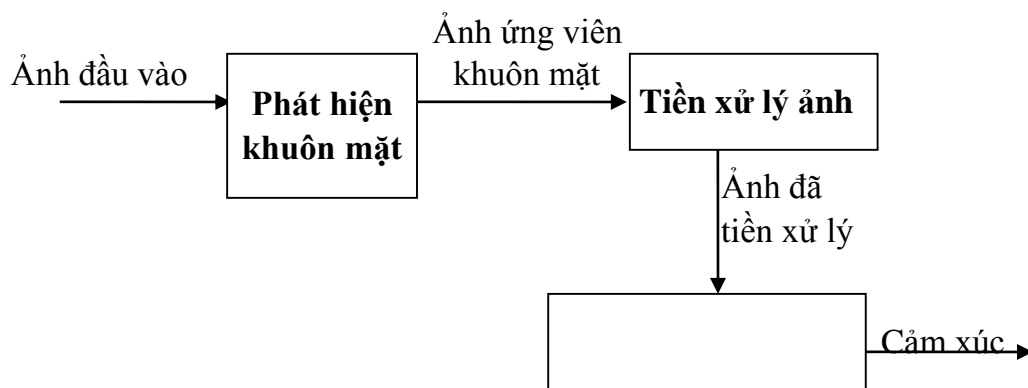
1.3.3 Phương pháp dùng mô hình AAM kết hợp tương quan điểm

Phương pháp này sử dụng mô hình AAM để phát hiện khuôn mặt, sau đó dựa vào tỷ lệ giữa hai mắt, lông mày, miệng, mũi, ... để nhận dạng cảm xúc. Hạn chế của phương pháp này là việc xác định tỉ lệ ngưỡng để xác định cảm xúc. Tuy nhiên phương pháp này có ưu điểm về tốc độ, do đó thường được ứng dụng trong nhận dạng cảm xúc thời gian thực.



Hình 1.3. Các điểm đánh dấu trên khuôn mặt

1.3.4 Mô hình tổng quan



Hình 1.4. Mô hình nhận dạng cảm xúc

1.3.5 Thách thức trong vấn đề nhận dạng cảm xúc khuôn mặt

Xác định cảm xúc khuôn mặt là một bài toán khó bởi vì con người ngoài 5 cảm xúc cơ bản, còn rất nhiều cảm xúc đa dạng khác. Hơn nữa, vì nhận dạng cảm xúc dựa trên các đặc điểm của khuôn mặt nên thực tế không thể biết được cảm xúc đó là đúng hay không. Về phương pháp nhận dạng, cũng gặp khó khăn khi ảnh khuôn mặt không chính diện, quá nhỏ, hay trong điều kiện ánh sáng không tốt.

1.3.6 Một số vấn đề khác

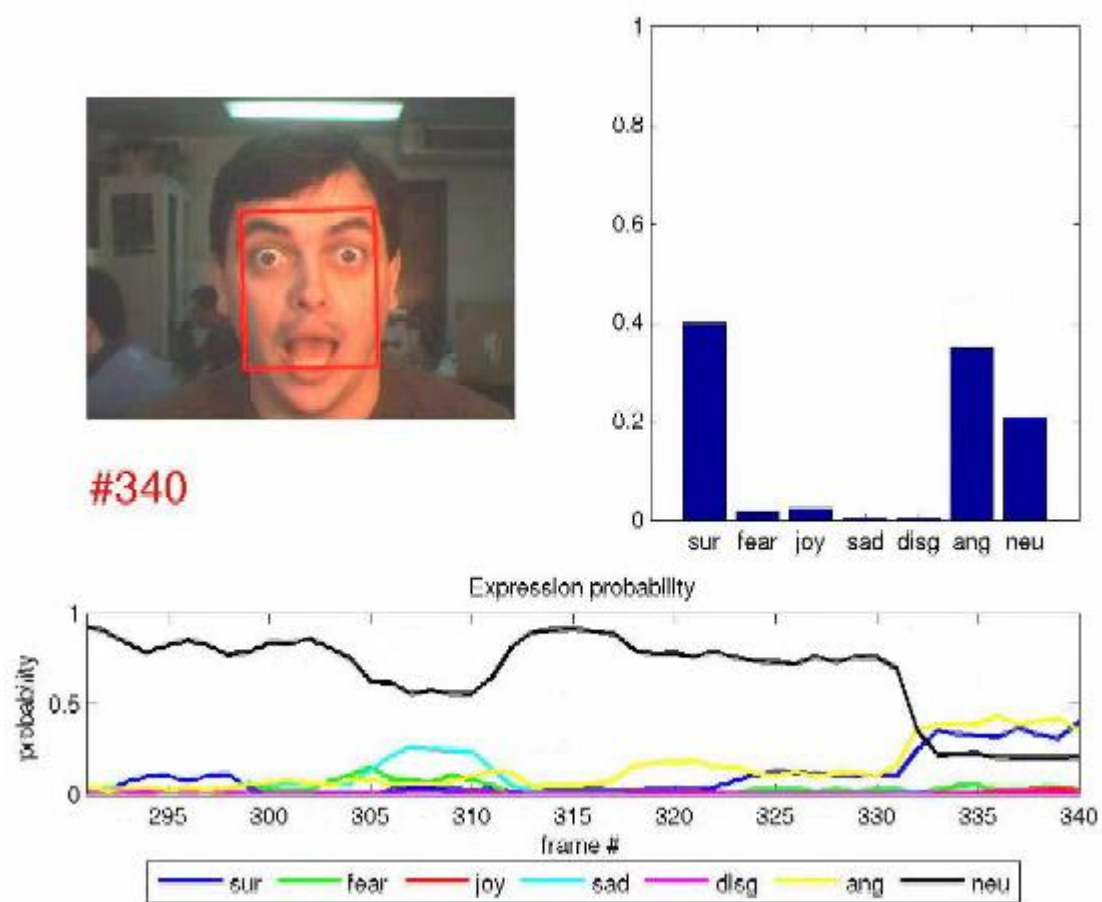
Bên cạnh việc nhận dạng cảm xúc trong không gian 2D còn có một số vấn đề liên quan mật thiết.

Nhận dạng cảm xúc trong không gian 3D: Đây là vấn đề rất gần gũi với nhận dạng cảm xúc trong không gian 2D, tuy nhiên trong không gian 3D chúng ta có nhiều thông tin hơn, ngoài màu sắc, đặc trưng còn có hình dáng của khuôn mặt,...



Hình 1.5. Biểu cảm khuôn mặt trong không gian 3D

Nhận dạng cảm xúc trong video: Vấn đề này dễ dàng hơn vì chúng ta có rất nhiều thông tin về khuôn mặt dựa vào các khung hình liên tiếp, và vấn đề này cũng thực tiễn hơn nhiều so với nhận dạng cảm xúc trong không gian 2D.



Hình 1.6. Cảm xúc trong video

CHƯƠNG 2. MỘT SỐ PHƯƠNG PHÁP PHÁT HIỆN BIỂU CẢM KHUÔN MẶT

2.1 Cơ sở lý thuyết

2.1.1 Giới thiệu về mạng Noron

Có thể nói, hiện nay, không có một định nghĩa chính thức nào cho mạng noron. Tuy nhiên phần lớn mọi người đều đồng tình rằng mạng noron là một mạng bao gồm rất nhiều bộ xử lý đơn giản (gọi là các unit), mỗi unit có vùng nhớ riêng của mình. Các unit được kết nối với nhau thông qua kênh thông tin (gọi là các connection), thường mang dữ liệu số (không phải là các ký hiệu), và được mã hóa theo một cách nào đấy. Unit chỉ xử lý trên bộ dữ liệu của riêng nó và trên các đầu vào được đưa tới thông qua các liên kết. hạn chế của phép xử lý cục bộ này là nó thường ở trạng thái nghỉ trong suốt quá trình học.

Một số mạng noron là mô hình mạng noron sinh học, một số thì không, nhưng từ trước tới nay tất cả các lĩnh vực của mạng noron đều được nghiên cứu xây dựng xuất phát từ các yêu cầu xây dựng hệ thống nhận tạo rất phức tạp, hay các phép xử lý “thông minh” tương tự như bộ não con người.

Hầu hết các mạng noron đều có một vài quy tắc học nào đó mà thông qua đó trọng số của các liên kết được điều chỉnh dựa trên dữ liệu. Nói cách khác, mạng noron “học” và các ví dụ và dựa trên các dữ liệu đó thì nó có khả năng tổng quát tri thức và đưa ra “nhận thức của mình”.

Mạng noron là mô hình mạng ứng dụng các phương pháp xử lý song song và các thành phần mạng xử lý hoàn toàn độc lập với nhau. Người ta xem khả năng xử lý song song số lượng lớn và tính liên kết cao của mạng noron là các tính chất đặc trưng của nó. Tuy nhiên với những yêu cầu như thế thì sẽ không có những mô hình đơn giản, ví dụ như mô hình hồi quy tuyến tính đơn giản, một mô hình được ứng dụng rất rộng rãi của mạng noron.

Mạng nơron có thể áp dụng trong mọi trường hợp khi tồn tại một mối liên hệ giữa các biến độc lập (inputs) và các biến phụ thuộc (outputs), thậm chí ngay cả khi mối quan hệ đó trở nên phức tạp. Một số lĩnh vực mà mạng nơron đã áp dụng thành công như dự đoán triệu chứng y học, dự đoán thị trường chứng khoán, đánh giá độ tin cậy tài chính, điều chỉnh điều kiện của cơ cấu máy móc.

2.1.1.1 Mạng Perceptron nhiều tầng (MPL – Multi Perceptron Layer)

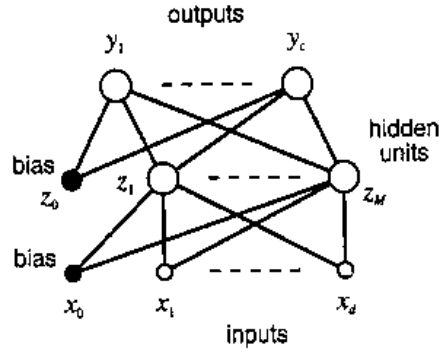
MPL là một loại mạng lan truyền tiến được huấn luyện theo kiểu học có giám sát. Mạng là một cấu trúc gồm nhiều lớp trọng số. Ở đây ta chỉ xét đến loại mạng lan truyền khả vi. Đây là loại mạng có thể áp dụng phương pháp tính toán khá hiệu quả và mạnh gọi là lan truyền ngược lỗi, để xác định đạo hàm hàm lỗi theo các trọng số và độ dốc trong mạng. Đây là một tính chất rất quan trọng của những mạng kiểu này bởi các đạo hàm này đóng vai trò trung tâm trong các giải thuật học của các mạng đa lớp. Vấn đề lan truyền ngược sẽ trình bày chi tiết ở phần sau.

2.1.1.2 Ánh xạ lan truyền tiến

Trong phần này ta sẽ nghiên cứu mô hình mạng nơron lan truyền tiến như là một khung tổng quát đại diện cho các hàm ánh xạ phi tuyến giữa tập các biến đầu vào và tập các biến đầu ra.

Mạng phân lớp

Các mạng đơn lớp được xây dựng dựa trên sự kết hợp tuyến tính các biến đầu vào được chuyển đổi từ một hàm truyền phi tuyến.



Hình 2.1 Mô hình mạng lan truyền tiến

Các nút không phải là các nút nhập và nút xuất được gọi là các nút ẩn. Trong mô hình chúng ta nghiên cứu ở đây, có d nút nhập, M nút ẩn và c nút xuất.

Kết quả của nút ẩn thứ j được tính như sau:

$$a_j = \sum_{i=1}^d w_{ji}^{(1)} x_i + w_{j0}^{(1)} \quad (2.1)$$

Trong đó w là trọng số của lớp đầu tiên, từ nút nhập i đến nút ẩn j , và x là trọng ngưỡng của nút ẩn j .

Giả sử đặt một biến cố định $x_0 = 1$. Từ đó công thức (2.1) có thể được viết lại:

$$a_j = \sum_{i=0}^d w_{ji}^{(1)} x_i \quad (2.2)$$

Sau đó z_k của nút ẩn j được tính toán bằng cách chuyển đổi tổng tuyến tính (2.2) sử dụng hàm truyền $g(\cdot)$, tức là: $z_k = g(a_j)$ (2.3)

Kết xuất của mạng được tính bằng cách chuyển đổi độ hoạt động của các nút ẩn sử dụng một lớp các nút thứ 2. Với mỗi nút xuất k , ta có:

$$a_k = \sum_{j=1}^M w_{kj}^{(2)} z_j + w_{k0}^{(2)} \quad (2.4)$$

Đặt $z_0 = 1$ ta có:

$$a_k = \sum_{j=0}^M w_{kj}^{(2)} z_j \quad (2.5)$$

Sau đó giá trị này được cho hàm truyền phi tuyến, cho ta kết xuất đầu ra của nút xuất k: $y_k = \tilde{g}(a_k)$ (2.6)

Ở đây, sử dụng kí hiệu để biểu diễn hàm truyền của các nút xuất nhằm chỉ ra rằng hàm này có thể không trùng với hàm đã được sử dụng trong lớp ẩn.

Kết hợp (2.2), (2.3), (2.5), (2.6) ta có công thức chung cho mô hình mạng trong hình trên:

$$y_k = \tilde{g} \left(\sum_{j=0}^M w_{kj}^{(2)} g \left(\sum_{i=0}^d w_{ji}^{(1)} x_i \right) \right) \quad (2.7)$$

Kiến trúc mạng tổng quát

Ta có thể xây dựng ánh xạ mạng tổng quát hơn bằng cách nghiên cứu sơ đồ mạng phức tạp hơn. Tuy nhiên ở đây ta chỉ giới hạn nghiên cứu trong phạm vi các mạng lan truyền tiến.

Mạng lan truyền tiến là mạng không có một kết nối quay lui nào trong mạng.

Theo Bishop (1995): Về mặt tổng quát, một mạng được gọi là lan truyền tiến nếu nó có thể gán các số liên tục cho tất cả các nút nhập, tất cả các nút ẩn và nút xuất sao cho mỗi nút chỉ có thể nhận được kết nối từ các nút nhập hoặc các nút được gán số nhỏ hơn.

Với những mạng có tính chất như thế, kết xuất của mạng là các hàm quyết định đầu vào, và vì thế toàn bộ mạng được gọi là một *ánh xạ hàm phi tuyến đa biến*.

Kết xuất của nút k tính được như sau:

$$z_k = g \left(\sum_j w_{kj} z_j \right) \quad (2.8)$$

Trong đó $g(.)$ là một hàm truyền phi tuyến, và j thuộc tập tất cả các nút nhập và các nút gửi, kết nối tới nút k (Tham số trọng ngưỡng cũng đã được bao hàm ở trong phép tính này).

Với một tập cho trước các giá trị đầu vào, áp dụng liên tục công thức (2.8) sẽ cho phép kích hoạt tất cả các nút trong mạng được ước lượng, bao gồm cả kích hoạt các nút xuất. Quá trình này được gọi là lan truyền tiến các tín hiệu qua mạng.

Nếu như hàm truyền của tất cả các nút ẩn trong mạng là tuyến tính, thì với những mạng như thế sẽ luôn luôn tìm được một mô hình mạng tương đương mà không có một nút ẩn nào. Mạng này được gọi là mạng tuyến tính đa lớp và vì thế không được đi sâu nghiên cứu, mà chỉ chủ yếu nghiên cứu các mạng đa lớp với các hàm truyền của các nút ẩn là phi tuyến.

2.1.1.3 Hàm sigmoid

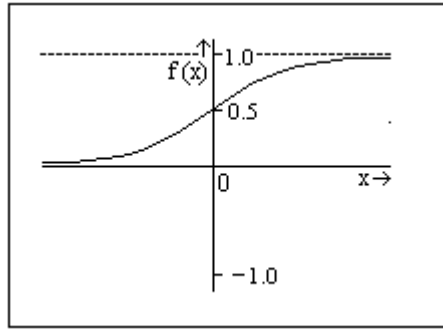
Bây giờ, ta sẽ xem xét hàm truyền logistic dạng S, trong đó đầu ra của nó nằm trong khoảng $(0,1)$, có phương trình như sau:

$$g(a) = \frac{1}{1 + \exp(-a)} \quad (2.9)$$

Hình vẽ dưới đây biểu diễn hàm truyền sigmoid cho các nút trong mạng. Đây là hàm mũ có đặc tính vô cùng quan trọng vì : khi x chạy từ vô cùng lớn đến vô cùng bé thì $f(x)$ luôn chạy trong khoảng từ 0 đến 1. Giải thuật học sẽ điều chỉnh trọng số kết nối giữa các nút để hàm ánh xạ giá trị của x sang dạng nhị phân, thông thường:

$$f(x) > 0.9 : f(x) = 1 \quad (2.10)$$

$$f(x) < 0.1 : f(x) = 0. \quad (2.11)$$



Hình 2.2 Đồ thị hàm truyền sigmoid

Sử dụng hàm sigmoid tại đầu ra sẽ giới hạn phạm vi có thể xảy ra với các nút xuất (giá trị kết xuất là từ 0 tới 1), và trong một số trường hợp thì điều này có thể không mong muốn. Thậm chí ngay cả khi giá trị xuất mong muốn nằm trong giới hạn của hàm sigmoid thì ta vẫn phải chú ý rằng hàm sigmoid $g(.)$ là một hàm đơn điệu tăng, do đó có thể lấy nghịch đảo. Do vậy một giá trị xuất y mong muốn đối với mạng có nút xuất thuộc dạng sigmoid, thì tương đương với một giá trị xuất $g^{-1}(y)$ đối với mạng có nút xuất tuyến tính.

Một nút ẩn thuộc dạng sigmoid có thể xấp xỉ một nút ẩn tuyến tính bất kì. Điều này đạt được bằng cách thiết kế cho tất cả trọng số các cung đầu vào của nút, cũng như trọng ngưỡng, sao cho rất nhỏ để tổng của giá trị nhập phải nằm trên phần tuyến tính của đường cong sigmoid, gần đúng với đường thẳng nguyên thủy. Nút ẩn dạng sigmoid có thể tạo ra xấp xỉ một hàm bậc thang (step) bằng việc đặt giá trị cho trọng số và trọng ngưỡng rất lớn.

Do đó, những mạng neural với nhiều tầng nút xử lý có khả năng xấp xỉ hoá vì chúng đã chứa đựng trong đó mạng neural hai tầng như một trường hợp đặc biệt. Điều này cho phép các tầng còn lại được sắp xếp để thực hiện những biến đổi tuyến tính như trên, và sự biến đổi đồng nhất chính là một trường hợp đặc biệt của một phép biến đổi tuyến tính.

2.1.1.4 Thuật toán lan truyền ngược

Trước tiên ta xem xét trường hợp mạng sử dụng hàm ngưỡng. nếu một nút đầu ra tạo một đáp số sai lệch thì ta phải xem nút ẩn nào tạo sự sai lệch đó, cũng

chính là việc quyết định trọng số nào cần phải điều chỉnh và điều chỉnh là bao nhiêu.

Để giải quyết vấn đề gán trọng số này, hãy xem xét một mạng với các hàm truyền phân biệt. Nếu coi hàm lỗi là một hàm riêng biệt cho giá trị xuất của mạng thì bản thân nó cũng chính là một hàm phân biệt của trọng số. Do đó, có thể tính toán đạo hàm hàm lỗi theo các trọng số, và giá trị đạo hàm này có thể dùng cực tiểu hoá hàm lỗi bằng cách sử dụng phương pháp giảm gradient (gradient descent) hoặc các phương pháp tối ưu hoá khác.

Giải thuật ước lượng đạo hàm hàm lỗi gọi là *lan truyền ngược*, nó tương đương với việc lan truyền ngược lỗi trong mạng. Kỹ thuật về lan truyền ngược được biết đến rất rộng rãi và chi tiết qua các bài báo cũng như các cuốn sách của Rumelhart, Hinton và Williams (1986). Tuy nhiên gần đây một số ý tưởng tương tự cũng được một số nhà nghiên cứu phát triển như Werbos (1974) và Parker (1985).

Cần nói thêm rằng giải thuật lan truyền ngược được sử dụng trong mạng nơron có ý nghĩa rất lớn, như kiến trúc của mạng nhiều tầng cũng thường được gọi là mạng lan truyền ngược. Khái niệm *lan truyền ngược cũng thường được sử dụng để mô tả quá trình huấn luyện của mạng nhiều tầng sử dụng phương pháp gradient descent áp dụng trên hàm lỗi dạng sai số bình phương trung bình*. Để làm rõ hơn về thuật ngữ này ta cần xem xét quá trình huấn luyện mạng. Phần lớn các giải thuật huấn luyện mạng đều liên quan đến một thủ tục được lặp đi lặp lại nhằm làm giảm thiểu hàm lỗi, bằng cách điều chỉnh trọng số trong một chuỗi các bước.

Tại mỗi bước như vậy, có thể chia thành hai bước phân biệt:

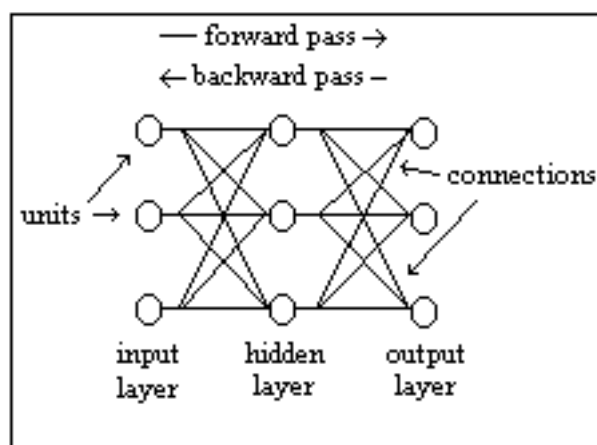
- Bước thứ nhất, cần phải tính đạo hàm hàm lỗi theo các trọng số. Ta biết rằng đóng góp rất quan trọng của kỹ thuật lan truyền ngược đó là việc cung cấp một phương pháp hết sức hiệu quả về mặt tính toán trong việc đánh giá các đạo hàm. Tại bước này lỗi sẽ được lan truyền

ngược nên ta sử dụng khái niệm lan truyền ngược để đặc trưng riêng cho việc đánh giá đạo hàm này.

- Bước thứ hai, đạo hàm sẽ được sử dụng trong việc tính toán điều chỉnh đối với trọng số. Kỹ thuật đơn giản nhất được sử dụng ở đây là kỹ thuật gradient descent, kỹ thuật này được Rumelhart và cộng sự (1986) đưa ra lần đầu tiên.

Điều quan trọng cần phân biệt là hai bước này độc lập nhau. Do đó, quá trình xử lý đầu tiên được biết đến là quá trình lan truyền ngược các lỗi vào trong mạng để đánh giá đạo hàm, có thể áp dụng đối với rất nhiều loại mạng khác nhau chứ không chỉ đối với riêng mạng nhiều tầng. Cũng có thể áp dụng với các loại hàm lỗi khác chứ không chỉ là hàm tính sai số bình phương cực tiểu, và để đánh giá các đạo hàm khác này có thể sử dụng các phương pháp khác như phương pháp ma trận Jacobian và Hessian mà ta sẽ xem xét ở phần sau. Cũng tương tự như vậy tại bước thứ hai, việc điều chỉnh trọng số sử dụng các đạo hàm đã được tính trước đó có thể thực hiện với nhiều phương pháp tối ưu hoá khác nhau, và rất nhiều trong số các phương pháp đó cho kết quả tốt hơn phương pháp gradient descend.

Lan truyền ngược:



Hình 2.3 Lan truyền ngược

Áp dụng giải thuật lan truyền ngược cho bất kỳ một mạng nơron có cấu hình lan truyền tiến tùy ý, sử dụng các hàm truyền phi tuyến tùy ý, và cả hàm lỗi có

dạng tùy ý. Để minh họa ta sẽ dùng một mạng có cấu trúc một tầng nút ẩn dạng sigmoid và hàm lỗi là hàm tính theo sai số trung bình bình phương.

Trong các mạng lan truyền tiến nói chung mỗi nút đều tính tổng đầu vào của nó theo công thức:

$$a_j = \sum_i w_{ji} z_i \quad (2.12)$$

Với z_i là giá trị nhập hoặc là giá trị xuất của một nút có cung kết nối với nút j và w_{ji} chính là trọng số của cung kết nối đó. Giá trị tổng này được tính trên tất cả các nút có kết nối trực tiếp với nút j . Ta biết rằng, trọng ngưỡng của nút được đưa vào trong tổng bằng cách tạo ra thêm một giá trị nhập cố định bằng 1. Tổng trong (2.12) lại được biến đổi thông qua một hàm truyền phi tuyến $g(.)$ để đưa ra được giá trị xuất z_i của nút j theo công thức:

$$z_i = g(a_j) \quad (2.13)$$

Bây giờ chúng ta cần phải xác định giá trị của các trọng số trong mạng thông qua việc tối thiểu hoá hàm lỗi.

Ở đây ta sẽ coi hàm lỗi được viết như một tổng của các lỗi tại mỗi mẫu riêng biệt. Tổng này sẽ được tính trên các mẫu của tập huấn luyện

$$E = \sum_n E^n \quad (2.14)$$

Với n là nhãn của từng mẫu.

Giả định rằng lỗi E^n có thể được thể hiện như một hàm riêng của các biến đầu ra, có nghĩa là :

$$E^n = E^n(y_c, \dots, y_c) \quad (2.15)$$

Mục đích ở đây là tìm ra một hàm để tính được đạo hàm của hàm lỗi theo các trọng số và trọng ngưỡng của mạng.

Đối với từng mẫu, ta sẽ coi như đã cung cấp một vector nhập tương ứng là đầu vào và tính được giá trị xuất của các nút ẩn cũng như nút xuất theo các công

thức (2.12), (2.13). Quá trình này thường được gọi là quá trình lan truyền tiến trong mạng.

Hãy xem xét việc tính đạo hàm của E^n theo các trọng số w_{ji} . Giá trị xuất của các nút sẽ phụ thuộc vào từng mẫu nhập n . Tuy nhiên để dễ nhìn, quy ước sẽ bỏ qua việc viết kí tự n trên các biến nhập và xuất. Cần chú ý rằng E^n phụ thuộc vào trọng số w_{ji} thông qua tổng giá trị nhập a_i của nút j . Do đó, có thể đưa ra công thức tính các đạo hàm riêng như sau:

$$\frac{\partial E^n}{\partial w_{ji}} = \frac{\partial E^n}{\partial a_j} * \frac{\partial a_j}{\partial w_{ji}} \quad (2.16)$$

Từ (2.16) ta có:

$$\frac{\partial a_j}{\partial w_{ji}} = z_i \quad (2.17)$$

Như vậy suy ra:

$$\frac{\partial E^n}{\partial w_{ji}} = \delta_j z_i \quad (2.18)$$

Trong đó:

$$\delta_j \equiv \frac{\partial E^n}{\partial a_j} \quad (2.19)$$

Từ công thức (2.18) thấy rằng để tính được đạo hàm ta chỉ cần tính giá trị cho mỗi nút ẩn và nút xuất trong mạng và sau đó áp dụng công thức (2.18).

Với các nút xuất thì việc tính δ_k là hết sức đơn giản.

Ta có:

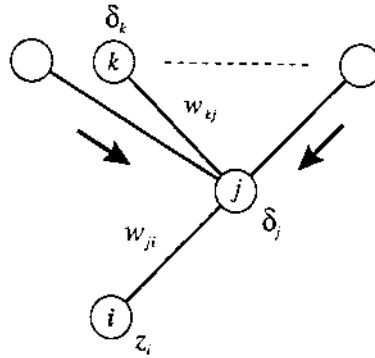
$$\delta_k \equiv \frac{\partial E^n}{\partial a_k} = g'(a_k) \frac{\partial E^n}{\partial y_k} \quad (2.20)$$

Để tính ra (2.20) cần tìm ra công thức tính $g'(a)$ và $\frac{\partial E^n}{\partial y}$.

Để tính được δ cho các nút ẩn, cần sử dụng công thức tính đạo hàm riêng:

$$\delta_j \equiv \frac{\partial E^n}{\partial a_j} = \sum_k \frac{\partial E^n}{\partial a_k} \frac{\partial a_k}{\partial a_j} \quad (2.21)$$

Trong đó giá trị tổng được tính trên các nút k mà nút j kết nối đến. Việc sắp xếp các nút cũng như các trọng số được minh hoạ trong hình 2.4



Hình 2.4: Minh họa việc tính δ_j cho việc tính nút ẩn j

Chú ý rằng các nút có nhãn k này có thể bao gồm cả nút nhập và nút xuất.

Ta có công thức lan truyền ngược như sau:

$$\delta_j \equiv g'(a_j) \sum_k w_{kj} \delta_k \quad (2.22)$$

Công thức này nói lên rằng giá trị của δ đối với một nút ẩn có thể được tính từ việc lan truyền ngược các giá trị δ của các nút ẩn cao hơn trong mạng, hình 2.4. Ta đã biết giá trị δ của các nút xuất nên có thể áp dụng công thức (2.22) một cách đệ quy nhằm tính ra giá trị δ cho tất cả nút ẩn trong mạng, mà không quan tâm đến cấu hình của nó.

Ta có thể tổng kết lại giải thuật lan truyền ngược nhằm tính đạo hàm hàm lỗi E^n theo các trọng số trong 4 bước:

- Đưa vector nhập x^n vào mạng và lan truyền tiến nó trong mạng để tìm ra giá trị xuất cho tất cả các nút ẩn cũng như nút xuất.

➤ Tính δ cho tất cả các nút xuất sử dụng công thức (2.22)

➤ Lan truyền ngược các d bằng công thức để thu được δ cho mỗi nút ẩn trong mạng.

➤ áp dụng $\frac{\partial E^n}{\partial w_{ji}} = \delta_j z_i$ để tính các đạo hàm.

Đạo hàm của lỗi tổng E có thể thu được bằng cách lặp đi lặp lại các bước trên đối với từng mẫu trong tập huấn luyện, sau đó tính tổng trên các lỗi.

Trong quá trình tính đạo hàm trên, giả định rằng mỗi nút ẩn cũng như xuất đều có chung một hàm truyền $g(\cdot)$. Tuy nhiên điều này hoàn toàn có thể tính được với trường hợp mỗi nút khác nhau đều có các hàm truyền riêng, đơn giản bằng cách đánh hàm $g(\cdot)$ ứng với từng nút.

2.1.2 Giới thiệu về PCA

Phần này giúp hiểu được phép phân tích thành phần chính (PCA). PCA là một kỹ thuật hữu ích trong các ứng dụng nhận dạng mặt và nén ảnh, và là một kỹ thuật phổ biến để tìm mẫu trong các dữ liệu nhiều chiều.

Giới thiệu về các khái niệm toán học sẽ được sử dụng trong PCA. Các khái niệm đó bao gồm: Độ lệch chuẩn (Standard deviation), phương sai (variance), hiệp phương sai (covariance), vec tơ riêng (eigenvector), giá trị riêng (eigenvalue).

2.1.2.1 Một số khái niệm toán học:

Độ lệch chuẩn:

Để hiểu độ lệch chuẩn, chúng ta cần một tập dữ liệu. Giả sử ta có tập

$$X = [1 \ 2 \ 4 \ 6 \ 12 \ 15 \ 25 \ 45 \ 68 \ 67 \ 65 \ 98]$$

\bar{X} là ký hiệu đại diện cho tập số, mỗi số riêng biệt được ký hiệu X_i (Ví dụ $X_3 = 4$). Phần tử đầu tiên là X_1 và n là số lượng phần tử của tập hợp. Khi đó trung bình của mẫu có công thức:

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} \quad (2.23)$$

\bar{X} Là ký hiệu trung bình của mẫu, tuy nhiên trung bình mẫu không nói lên được nhiều điều ngoại trừ cho ta biết nó là một điểm giữa. Ví dụ với 2 tập dữ liệu

[0 8 12 20] và [8 9 11 12]

có trung bình mẫu bằng nhau nhưng lại khá khác nhau. Sự khác biệt ở đây là khoảng cách của dữ liệu, và độ lệch chuẩn là đại lượng để đo khoảng cách này. Ta có thể hiểu độ lệch chuẩn là khoảng cách trung bình từ trung bình mẫu đến các điểm của dữ liệu. Ta có công thức:

$$s = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)}} \quad (2.24)$$

Tập hợp 1

X	$(X - \bar{X})$	$(X - \bar{X})^2$
0	-10	100
8	-2	4
12	2	4
20	10	100
Tổng		208
Chia cho (n-1)		69.333
Căn bậc hai		8.3266

Bảng 2.1 Tập dữ liệu 1

Tập hợp 2

X_i	$(X_i - \bar{X})$	$(X_i - \bar{X})^2$
8	-2	4
9	-1	1
11	1	1
12	2	4
Tổng		10
Chia cho (n-1)		3.333
Căn bậc hai		1.8257

Bảng 2.2 Tập dữ liệu 2

Dễ dàng nhận thấy tập dữ liệu 1 có độ lệch chuẩn lớn hơn tập dữ liệu 2.

Phương sai

Phương sai là một đại lượng khác dùng để đo khoảng cách của dữ liệu. Ta có công thức:

$$s^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)} \quad (2.25)$$

Dễ thấy phương sai chính là bình phương độ lệch chuẩn.

Hiệp phương sai

Hai đại lượng độ lệch chuẩn và phương sai chỉ sử dụng được trong một chiều, trong thực tế dữ liệu có thể có rất nhiều chiều. Một ví dụ đơn giản ta có dữ liệu về cân nặng và điểm số của toàn bộ sinh viên trong lớp K51-KHMT. Đối với dữ liệu này, độ lệch chuẩn và phương sai chỉ tính được trên từng chiều riêng biệt và không thấy được mối liên hệ giữa 2 chiều này.

Tương tự phương sai, hiệp phương sai là đại lượng đo sự biến thiên giữa 2 chiều. Nếu tính hiệp phương sai giữa 1 chiều với chính nó ta được phương sai của chiều đó. Nếu tập dữ liệu có 3 chiều x, y, z ta có thể tính hiệp phương sai của từng cặp chiều (x, y), (y, z), (z, x). Công thức của hiệp phương sai tương tự công thức của phương sai. Công thức của phương sai được khai triển như sau:

$$var(X) = \frac{\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})}{(n-1)} \quad (2.26)$$

Và công thức của hiệp phương sai:

$$cov(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)} \quad (2.27)$$

Từ công thức hiệp phương sai ta thấy, nếu $cov(X, Y)$ dương thì X, Y đồng biến, $cov(X, Y)$ âm thì X, Y nghịch biến, nếu bằng 0 thì X, Y độc lập.

Ma trận hiệp phương sai

Hiệp phương sai có sự biến thiên giữa 2 chiều, do đó đối với tập dữ liệu có n chiều có $\frac{n!}{(n-2)! \cdot 2}$ giá trị hiệp phương sai khác nhau. Để thuận tiện cho việc tính toán, biểu diễn các giá trị này thông qua một ma trận gọi là ma trận hiệp phương sai. Định nghĩa của ma trận như sau:

$$C^{n \times n} = (c_{i,j}, c_{i,j} = cov(Dim_i, Dim_j)) \quad (2.28)$$

Trong đó $C^{n \times n}$ là 1 ma trận với n hàng, n cột và Dim_x là chiều thứ x. Ví dụ ma trận hiệp phương sai của 1 tập dữ liệu có 3 chiều x, y, z:

$$C = \begin{pmatrix} \text{cov}(x, x) & \text{cov}(x, y) & \text{cov}(x, z) \\ \text{cov}(y, x) & \text{cov}(y, y) & \text{cov}(y, z) \\ \text{cov}(z, x) & \text{cov}(z, y) & \text{cov}(z, z) \end{pmatrix} \quad (2.29)$$

2.1.2.2 Ma trận đại số

Phần này giới thiệu về hai khái niệm là nền tảng được sử dụng trong PCA đó là vector riêng (eigenvector) và giá trị riêng (eigenvalue).

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 3 \end{pmatrix} = \begin{pmatrix} 11 \\ 5 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 12 \\ 8 \end{pmatrix} = 4 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

Hình 2.5 Ví dụ về non-eigenvector và eigenvector

$$2 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 6 \\ 4 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 6 \\ 4 \end{pmatrix} = \begin{pmatrix} 24 \\ 16 \end{pmatrix} = 4 \times \begin{pmatrix} 6 \\ 4 \end{pmatrix}$$

Hình 2.6 Ví dụ về eigenvector có tỉ lệ khác vẫn là eigenvector

2.1.2.3 Eigenvector (Vector riêng)

Ta có thể nhân 2 ma trận với điều kiện kích cỡ phù hợp và eigenvector là một trường hợp đặc biệt của phép nhân này. Quan sát hai phép nhân ma trận với vector trên hình (2.7) (2.8). Ở ví dụ thứ nhất vector kết quả không phải là một bội số của vector gốc, trong khi ở ví dụ thứ hai vector kết quả bằng 4 lần vector gốc. Ta thấy $\begin{pmatrix} 3 \\ 2 \end{pmatrix}$ vector biểu diễn một mũi tên từ điểm (0, 0) đến điểm (3, 2) và ma trận

còn lại được hiểu là ma trận chuyển đổi. Nếu ta nhân ma trận này về bên trái của vector thì vector mới nhận được chính là vector cũ bị tịnh tiến đi 1 lượng. Đó là tính biến đổi của vector riêng.

Các tính chất của vector riêng:

- Chỉ các ma trận vuông ($n \times n$) mới có vector riêng.
- Không phải mọi ma trận vuông đều có vector riêng.
- Nếu 1 ma trận vuông ($n \times n$) có vector riêng thì sẽ có n vector riêng.
- Nếu nhân vector riêng với 1 số thì kết quả sau khi nhân với ma trận chuyển đổi, vector kết quả vẫn là vector ban đầu
- Tất cả các vector riêng của 1 ma trận đều trực giao với nhau

2.1.2.4 Eigenvalue (Giá trị riêng)

Giá trị riêng là một khái niệm liên quan chặt chẽ đến vector riêng. Thực tế một giá trị riêng nhân với hai vector riêng bằng nhau được gọi là giá trị riêng ứng với một vector riêng. Giá trị riêng và vector riêng luôn đi với nhau thành 1 cặp.

2.1.2.5 Phân tích thành phần chính của PCA

PCA là 1 phương pháp để nhận dạng các mẫu trong dữ liệu và biểu diễn dữ liệu bằng cách làm nổi bật sự giống và khác nhau. Khi các mẫu trong dữ liệu rất khó nhận ra trong không gian đa chiều thì PCA là một công cụ mạnh để phân tích chúng.

Các bước cơ bản trong PCA:

Bước 1: Lấy dữ liệu (Get data)

Bước 2: Trừ trung bình mẫu.

Với mỗi chiều dữ liệu giả sử ở chiều x , đều có một trung bình mẫu, công việc trong bước này là trừ tất cả giá trị trong chiều x cho trung bình mẫu x . Kết thúc bước này sẽ có trung bình mẫu ở tất cả các chiều là 0.

Bước 3: Tính ma trận hiệp phương sai

Bước 4: Tính các vector riêng và giá trị riêng của ma trận hiệp phương sai.

Bước 5: Chọn các thành phần chính

Đây là bước cuối cùng trong PCA. Trong bước này, tùy thuộc vào số lượng thành phần chính cần lấy, ta lấy lần lượt các thành phần (vector riêng) tương ứng với các giá trị riêng cao nhất.

2.1.3 Giới thiệu về Eigenface

Ý tưởng sử dụng eigenfaces đã được phát triển bởi Sirovich, Kirby (1987), Kirby và Sirovich (1990) tìm các hình ảnh của khuôn mặt bằng cách phân tích thành phần chính trên khuôn mặt. Bắt đầu với tập các hình ảnh khuôn mặt ban đầu, họ đã tính toán một hệ thống tọa độ tốt nhất cho ảnh, mỗi tọa độ là một hình ảnh gọi là Eigenpicture. Tác giả lập luận rằng, tập các hình ảnh khuôn mặt có thể là khoảng cách gần đúng được khôi phục bằng cách lưu trữ một tập nhỏ các trọng số cho từng khuôn mặt và một tập nhỏ các hình ảnh chuẩn (eigenpictures). Trọng số từng khuôn mặt được tính bằng cách chiếu hình ảnh khuôn mặt vào eigenpicture.

Nhận diện khuôn mặt gồm các hoạt động khởi tạo sau:

1. Khởi tạo tập các hình ảnh khuôn mặt (tập huấn luyện).
2. Tính eigenfaces từ tập huấn luyện, chỉ giữ lại những hình ảnh M tương ứng với giá trị đặc trưng cao nhất. Hình ảnh M xác định không gian mặt, nếu là khuôn mặt mới, eigenfaces có thể được cập nhật hoặc tính toán lại.
3. Tính toán phân bố tương ứng trong không gian trọng lượng M -chiều cho mỗi khuôn mặt đã nhận dạng, bằng cách chiếu hình ảnh khuôn mặt vào “không gian mặt”.

2.1.3.1 Tính Eigenface



Hình 2.7 Các hình ảnh khuôn mặt được sử dụng làm tập huấn luyện

Chiếu hình ảnh khuôn mặt $I(x, y)$ vào không gian 2 chiều N của mảng A'' (8-bit) giá trị cường độ. Hình ảnh được coi như là một vector có kích thước N^2 , do đó hình ảnh có kích thước 256×256 sẽ trở thành vector có kích thước 65.536 hoặc tương đương với một điểm trong không gian.

Hình ảnh khuôn mặt giống hình oval không được phân loại trong không gian ảnh lớn này, nên được mô tả bởi không gian con nhỏ hơn. Ý tưởng chính là tìm vector tốt nhất cho việc phân bố hình ảnh khuôn mặt trong toàn bộ không gian ảnh. Vector xác định không gian con của hình ảnh khuôn mặt, gọi là “không gian mặt”. Mỗi vector có chiều dài N^2 là sự kết hợp tuyến tính của các hình ảnh khuôn mặt ban đầu. Vì vector là vector đặc trưng của ma trận phương sai ứng với hình ảnh khuôn mặt ban đầu. Hình 2.7 là tập huấn luyện các hình ảnh khuôn mặt.

Tập huấn luyện các hình ảnh khuôn mặt là $\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M$. Khuôn mặt trung bình của tập hợp được xác định bởi $\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n$. Mỗi khuôn mặt khác nhau có giá trị trung bình $\Phi_i = \Gamma_i - \Psi$, khuôn mặt trung bình Ψ được thể hiện trong Hình 2.8. Đây là tập lớn các vector phụ thuộc vào việc phân tích thành phần chính, để tìm kiếm một tập \mathbf{M} vector trực chuẩn \mathbf{u}_n , mô tả sự phân bố dữ liệu. Vector \mathbf{u}_k được tính như:

$$\lambda_k = \frac{1}{M} \sum_{n=1}^M (\mathbf{u}_k^T \Phi_n)^2 \quad (2.30)$$

Vector này lớn nhất phụ thuộc vào

$$\mathbf{u}_l^T \mathbf{u}_k = \delta_{lk} = \begin{cases} 1, & \text{if } l = k \\ 0, & \text{otherwise} \end{cases} \quad (2.31)$$

Vector \mathbf{u}_k và vô hướng λ_k là vector đặc trưng và vector trị riêng, tương ứng của ma trận phương sai

$$\mathbf{C} = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = \mathbf{A} \mathbf{A}^T \quad (2.32)$$

Trong đó ma trận $\mathbf{A} = [\Phi_1 \Phi_2 \dots \Phi_M]$. Tuy nhiên xác định vector đặc trưng N^2 và giá trị đặc trưng là công việc khó khăn đối với kích cỡ các ảnh. Như vậy ta cần một phương pháp tính toán để tìm ra vector đặc trưng

Nếu số lượng các điểm dữ liệu trong không gian ảnh là ít hơn so với kích thước của không gian ($M < N^2$), thì sẽ chỉ có $M-1$, thay vì N^2 vector đặc trưng. (Các vector đặc trưng còn lại sẽ kết hợp với giá trị đặc trưng). Có thể tìm ra phương án cho vector đặc trưng kích thước N^2 trong trường hợp này bằng cách, đầu tiên ta chiếu vector đặc trưng lên ma trận \mathbf{M} của \mathbf{M} , (như ta tính ma trận 16×16 chứ không phải ma trận $16,384 \times 16,384$), sau đó dùng tổ hợp tuyến tính thích hợp của hình ảnh khuôn mặt Φ , tính toán vector đặc trưng \mathbf{v}_i của $\mathbf{A}^T \mathbf{A}$ như sau :

$$\mathbf{A}^T \mathbf{A} \mathbf{v}_i = \mu_i \mathbf{v}_i \quad (2.33)$$

Ta có :

$$AA^T Av_i = \mu_i Av_i \quad (2.34)$$

Ta thấy rằng Av_i là vector đặc trưng của $C = AA^T$



Hình 2.8 Khuôn mặt trung bình

Xây dựng ma trận M của M , $L = A^T A$ khi $L_{mn} = \Phi_m^T \Phi_n$ và tìm các vector đặc trưng M , v_l của L . Những vector xác định tổ hợp tuyến tính của tập huấn luyện hình ảnh khuôn mặt M để tạo thành vector đặc trưng u_l

$$u_l = \sum_{k=1}^M v_{lk} \Phi_k, \quad l = 1, \dots, M \quad (2.35)$$

Trong thực tế, tập huấn luyện của hình ảnh khuôn mặt sẽ tương đối nhỏ ($M < M^2$), và bài toán trở nên dễ dàng hơn. Các giá trị đặc trưng liên quan cho phép ta xếp các vector đặc trưng theo tính hữu dụng của nó trong việc mô tả sự biến đổi giữa các hình ảnh. Hình 2.9 cho ta thấy 7 vector đặc trưng bắt nguồn từ những hình ảnh đầu vào của Hình 2.7



Hình 2.9 Bảy trong số các khuôn mặt đặc trưng được tính từ những hình ảnh đầu vào của hình 2.7

2.1.3.2 Sử dụng khuôn mặt đặc trưng để phân loại hình ảnh khuôn mặt

Hình ảnh khuôn mặt đặc trưng được tính từ vector đặc trưng L trải rộng trên tập mô tả về những hình ảnh khuôn mặt. Sirovich và Kirby (1987) đã dự đoán rằng khoảng 40 eigenfaces là đủ cho một tập mô tả hình ảnh khuôn mặt, vậy $M' = 40$ eigenfaces .

Trong đó M' là điều kiện đủ để nhận dạng. Các eigenfaces trải trong không gian con M' chiều của không gian hình ảnh ban đầu N^2 . Vector đặc trưng M' của ma trận L là vector có giá trị đặc trưng lớn nhất. Trong nhiều trường hợp thử nghiệm, dựa trên $M=16$ hình ảnh khuôn mặt thì tính được $M'=7$ eigenfaces

Một hình ảnh khuôn mặt mới (Γ) đưa vào “không gian mặt” được tính như sau

$$\omega_k = u_k^T (\Gamma - \Psi) \quad (2.36)$$

Với $k=1, \dots, M'$

Hình 2.10 cho ta thấy một bức ảnh và hình chiếu của nó vào không gian bảy chiều.



Hình 2.10: Hình ảnh khuôn mặt ban đầu và hình chiếu của nó lên không gian mặt được xác định bởi các eigenfaces của hình 2.9

Vector $\Omega^T = [\omega_1, \omega_2, \dots, \omega_M]$ mô tả thuộc tính của eigenface, xử lý các eigenfaces như tập cơ sở cho hình ảnh khuôn mặt. Các vector được sử dụng trong thuật toán nhận dạng mẫu chuẩn để tìm ra một số lớp khuôn mặt được xác định trước. Phương pháp đơn giản để xác định lớp khuôn mặt tốt nhất là tìm lớp khuôn mặt đó bằng khoảng cách tối thiểu Euclide

$$\epsilon_k^2 = \|\Omega - \Omega_k\|^2 \quad (2.37)$$

Trong đó Ω_k là vector mô tả lớp khuôn mặt k th. Lớp khuôn mặt Ω_i được tính bằng trung bình kết quả các eigenface đặc trưng trên một số lượng nhỏ các hình ảnh khuôn mặt của một người. Một khuôn mặt được phân loại thuộc lớp k , khi ϵ_k là nhỏ nhất đến ngưỡng θ_ϵ . Ngược lại khuôn mặt được phân loại là ‘không biết’ và tùy chọn sử dụng để tạo ra một lớp mặt mới.

Việc tạo các vector trọng lượng tương đương với chiếu hình ảnh khuôn mặt ban đầu lên không gian mặt, nhiều hình ảnh sẽ chiếu vào vector mẫu nhất định. Đây không phải là vấn đề đối với hệ thống, tuy nhiên khi khoảng cách ϵ giữa ảnh

và không gian mặt chỉ đơn giản là khoảng cách bình phương giữa các hình ảnh đầu vào

$$\Phi = \Gamma - \Psi \text{ and } \Phi_f = \sum_{i=1}^{M'} \omega_i l \quad (2.38)$$

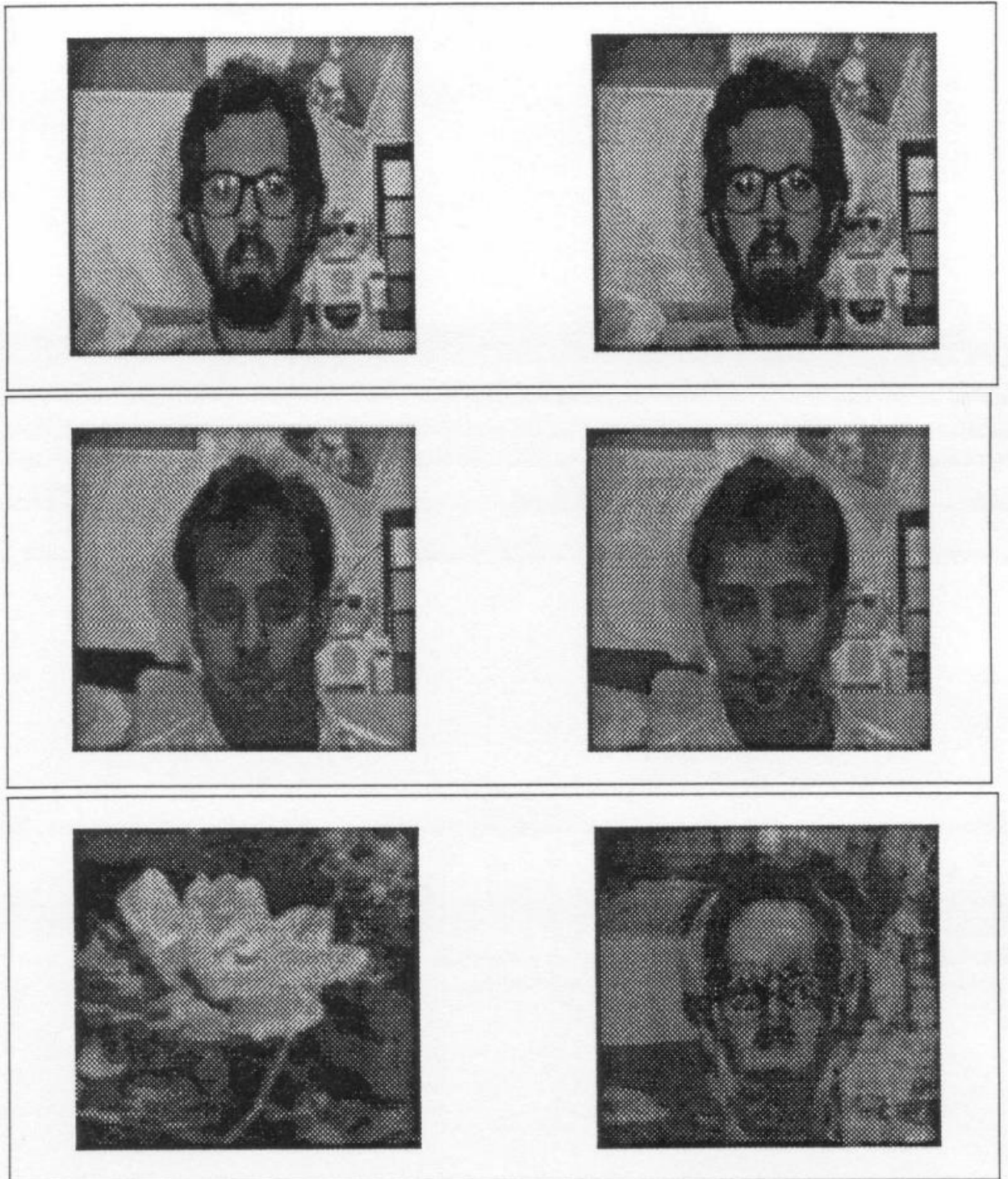
Hình chiếu của nó lên không gian mặt

$$\epsilon^2 = \|\Phi - \Phi_f\|^2 \quad (2.39)$$

Như vậy có bốn khả năng cho hình ảnh đầu vào và vector mô hình của nó:

- Gần không gian mặt và gần một lớp mặt.
- Gần không gian mặt nhưng không ở gần một lớp mặt đã biết.
- Xa không gian mặt và gần một lớp mặt.
- Xa không gian mặt và xa lớp mặt.

Trong trường hợp đầu tiên, mặt người được nhận dạng và xác định. Trường hợp thứ hai chỉ ra rằng hình ảnh không phải là hình ảnh khuôn mặt. Trường hợp ba được thể hiện như một đại lượng dương tính trong hệ thống nhận dạng. Hình 4 cho thấy một số giải pháp đo lường về khoảng cách giữa không gian mặt.



Hình 2.11 Ba ảnh và hình chiếu của nó vào không gian mặt, được xác định bởi các eigenface. Tính tương đối của khoảng cách từ không gian mặt là (a) 29.8 , (b) 58.5 , (c) 5217.4

2.1.3.3 Tóm tắt phương pháp nhận dạng đặc trưng khuôn mặt

Sử dụng eigenface để nhận diện khuôn mặt bao gồm các bước sau:

- Tạo tập các hình ảnh khuôn mặt đặc trưng, tập này gồm một số hình ảnh của một số người, với thay đổi trong biểu cảm và ánh sáng (bốn ảnh cho 10 người, do đó $M=40$)
- Tính ma trận L (40×40), tìm vector đặc trưng và giá trị đặc trưng của nó, chọn vector đặc trưng M' với các giá trị đặc trưng liên kết cao nhất (trong ví dụ này $M'=10$)
- Kết hợp huấn luyện tập các hình ảnh theo phương Eq. Kết quả ($M'=10$) eigenfaces u_k
- Với mỗi hình ảnh được biết, tính toán lớp vector Ω_k bằng trung bình mẫu vector eigenface Ω tính từ bốn ảnh cá nhân ban đầu. Chọn ngưỡng θ_k xác định khoảng cách lớn nhất từ bất kỳ lớp mặt, và một ngưỡng θ_k xác định khoảng cách tối đa từ không gian mặt.
- Với mỗi hình ảnh khuôn mặt mới được xác định, tính toán vector mô hình của nó Ω , các khoảng cách ε_i của mỗi lớp, và khoảng cách ε đến không gian mặt. Nếu khoảng cách tối thiểu $\varepsilon_k < \theta_\varepsilon$ và khoảng cách $\varepsilon < \theta_\varepsilon$ thì phân loại khuôn mặt đầu vào. Nếu khoảng cách tối thiểu $\varepsilon_k > \theta_\varepsilon$ nhưng $\varepsilon < \theta_\varepsilon$, thì hình ảnh được phân loại là “không rõ”.
- Nếu ảnh mới được phân loại là ảnh khuôn mặt đã biết, thì ảnh này được thêm vào tập ban đầu gồm các hình ảnh khuôn mặt quen thuộc, và eigenface được tính toán lại. Do đó có thể thay đổi không gian mặt khi hệ thống gặp nhiều trường hợp là các khuôn mặt đã biết.

Để xây dựng một hệ thống nhận dạng khuôn mặt đòi hỏi phải đánh giá tính tổng quát, yêu cầu trung thực, và tốc độ nhận dạng. Nếu nhận dạng khuôn mặt được giới hạn trong một nhóm nhỏ người (chẳng hạn như các thành viên của một gia đình hoặc một công ty nhỏ), một tập hợp nhỏ của eigenfaces sẽ trải rộng những khuôn mặt liên quan. Nếu hệ thống là để học những gương mặt mới hoặc gương

mặt đại diện cho nhiều người, thì một tập cơ sở lớn hơn của eigenfaces sẽ được tạo. Kết quả của Sirovich và Kirby (1987); Kirby và Sirovich (1990) mã hóa hình ảnh khuôn mặt chứng minh rằng số eigenfaces là tương đối nhỏ.

2.1.3.4 Tìm và phát hiện khuôn mặt

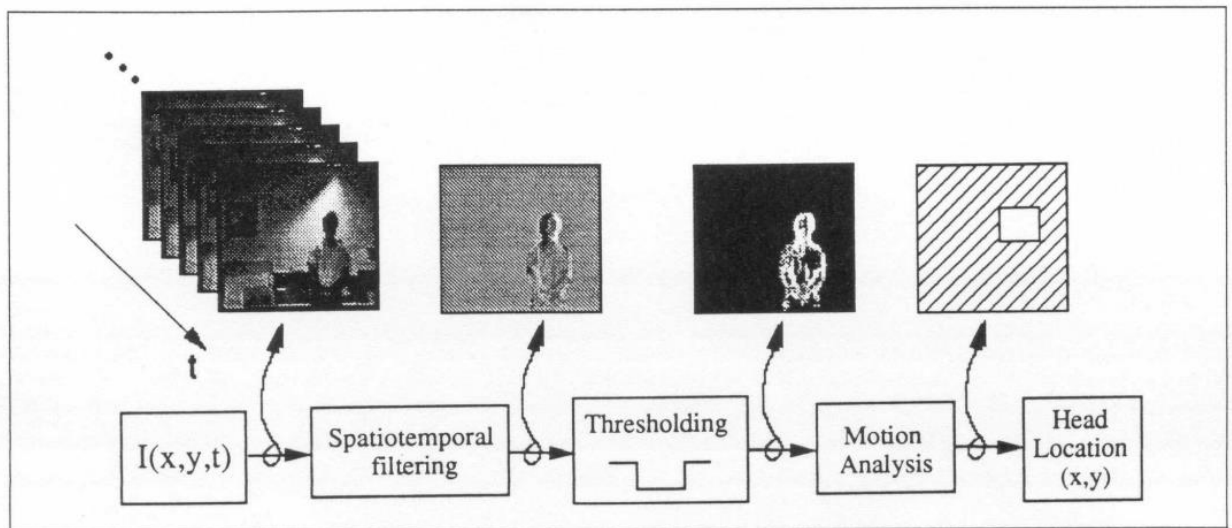
Như đã phân tích ở phần trước, giả định có hình ảnh khuôn mặt trọng tâm, kích thước giống hình ảnh khuôn mặt đã được huấn luyện và eigenfaces. Để xác định vị trí khuôn mặt trong khung cảnh, tác giả đã phát triển hai phương án xác định vị trí và theo dõi khuôn mặt bằng cách phát hiện chuyển động.

➤ Phát hiện chuyển động và theo dõi đầu

Khi chỉ có một người chuyển động trong môi trường tĩnh, phát hiện chuyển động và theo dõi thuật toán rất đơn giản. Hình 2.12 xác định vị trí và dò tìm vị trí của đầu. Không gian làm nổi bật vị trí ảnh thay đổi theo thời gian, do đó chuyển động của đầu trong ảnh “sáng lên” thì sẽ được lọc. Nếu hình ảnh sáng lên ở tất cả, thì phát hiện chuyển động và sự xuất hiện của người đó được xác định.

Hình ảnh được lọc để tạo ra chuyển động nhị phân, ta phân tích các “đốm màu chuyển động” theo thời gian để xác định xem chuyển động có được tạo ra bởi con người hay không, và định vị vị trí đầu của người đó. Một vài quy tắc đơn giản được áp dụng, chẳng hạn như “đầu là các đốm nhỏ ở phía trên và bên trên đốm lớn hơn (cơ thể)” và “chuyển động đầu tốc độ phải hợp lý và tiếp giáp” (đầu sẽ không chuyển động xung quanh hình ảnh một cách thất thường). Hình 2.14 cho thấy hình ảnh với đầu đã định vị, cùng với sự chuyển động của đầu trong khung tiếp theo.

Những hình ảnh chuyển động cho phép ước tính về quy mô, kích thước của đốm sáng, xác định kích thước của ảnh phụ để gửi đến các giai đoạn nhận dạng. Hình ảnh phụ này được thay đổi kích cỡ để phù hợp với kích thước của eigenface.

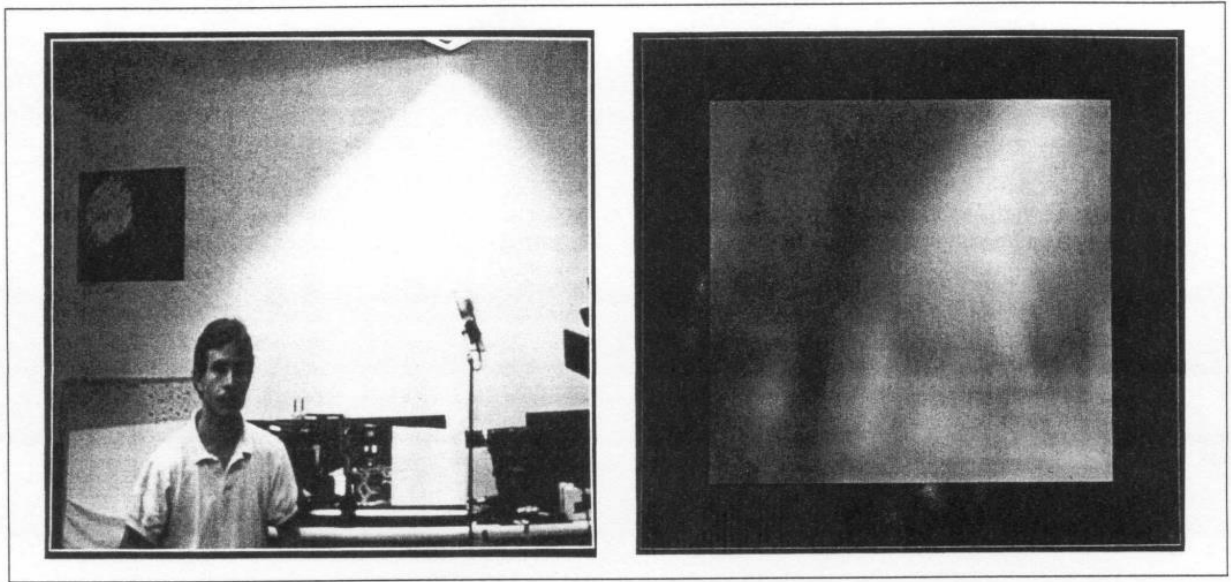


Hình 2.12. Dò tìm vị trí của đầu và hệ thống định vị

➤ Sử dụng không gian mặt để xác định vị trí mặt

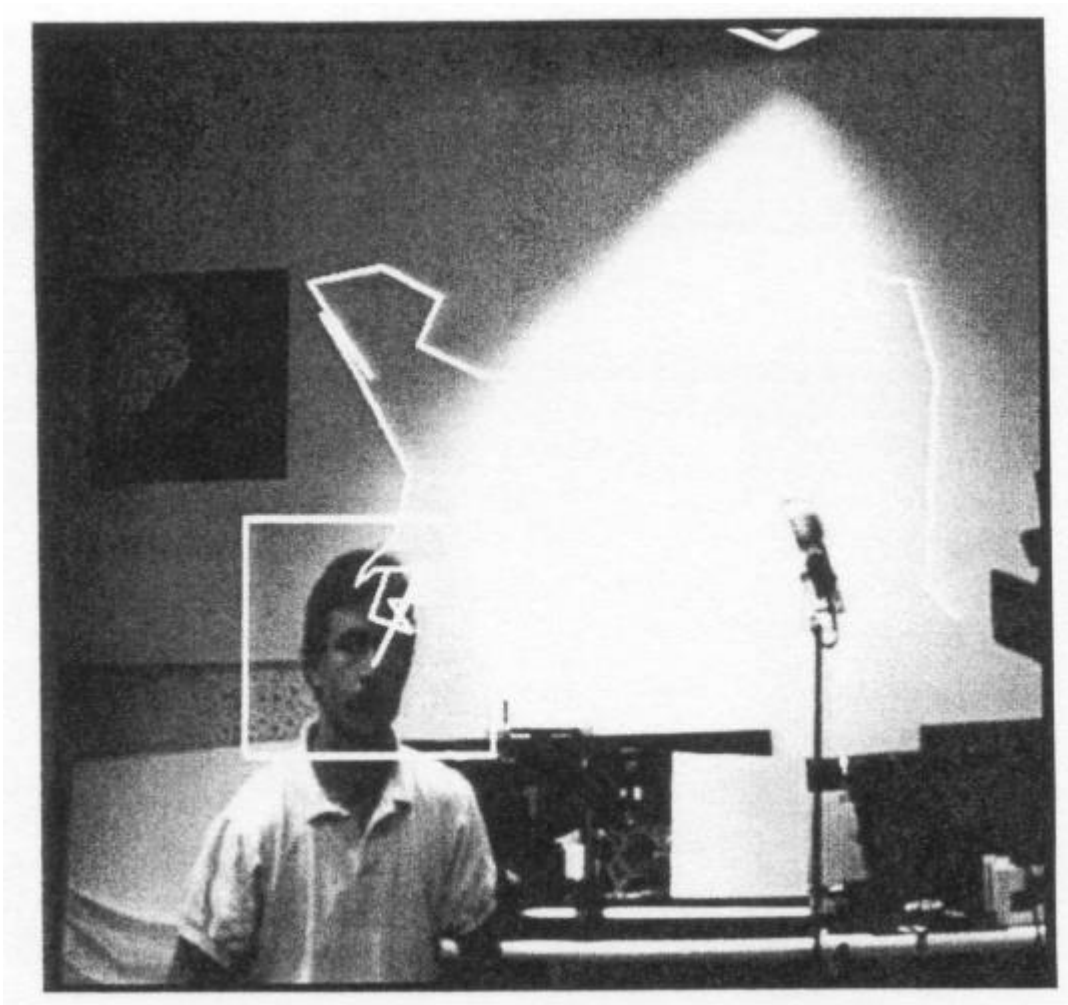
Sử dụng kiến thức của không gian mặt để xác định vị trí khuôn mặt trong ảnh đơn, là một thay thế cho vị trí khuôn mặt qua biểu cảm (ví dụ có quá ít biểu cảm hoặc nhiều đối tượng chuyển động), hoặc phương pháp có độ chính xác hơn là theo dõi biểu cảm đơn. Phương pháp này giúp nhận ra sự hiện diện khuôn mặt ngoài nhanh hơn.

Như đã thấy trong hình 2.11, hình ảnh khuôn mặt không thay đổi hoàn toàn khi chiếu vào không gian mặt, trong khi chiếu những hình ảnh không phải là khuôn mặt thì ảnh thu được rất khác biệt. Điều này sử dụng để phát hiện sự hiện diện của khuôn mặt trong một khung cảnh: tại mỗi điểm trong ảnh, tính toán khoảng cách ε giữa vị trí ảnh phụ và không gian mặt. Kết quả tính khoảng cách từ không gian mặt ở mọi điểm trong ảnh là một “bản đồ mặt” $\varepsilon(x,y)$. Hình 2.13 cho thấy một hình ảnh và bản đồ mặt giá trị thấp (vùng tối) thể hiện sự hiện diện của khuôn mặt.



Hình 2.13 (a) Ảnh ban đầu . (b) Bản đồ mặt tương ứng.

Để tính bản đồ mặt tại mỗi điểm ảnh của hình ảnh $I(x,y)$, ta cần tính khoảng cách từ tâm của ảnh phụ đến điểm ảnh trên không gian mặt, sau đó trừ đi hình chiếu gốc. Để chiếu một ảnh phụ Γ lên không gian mặt, đầu tiên ta phải trừ đi hình ảnh trung bình, kết quả $\Phi = \Gamma - \Psi$, với Φ_f là hình chiếu của Φ lên không gian mặt.



Hình 2.14 Vị trí của đầu đã được xác định

➤ Học để nhận ra gương mặt mới

Các khái niệm về không gian mặt cho phép khả năng học hỏi và nhận ra khuôn mặt mới một cách không có giám sát. Khi một hình ảnh là đủ gần để đối mặt với không gian nhưng không được phân loại là một trong những gương mặt quen thuộc, nó bắt đầu được gắn nhãn là “không rõ”. Máy tính lưu trữ vector mẫu và hình ảnh tương ứng chưa được nhận dạng. Nếu tổng hợp các vector mẫu thành cụm trong mô hình không gian, thì sự hiện diện của một khuôn mặt mới, nhưng không xác định được mặc nhiên được công nhận.

Hình ảnh tương ứng với vector mô hình trong cụm được kiểm tra bằng cách tính khoảng cách từ mỗi hình ảnh đến trung bình của các hình ảnh nhỏ hơn trong một ngưỡng được xác định trước. Nếu những hình ảnh vượt qua tính toán này, thì

giá trị trung bình của các vector đặc tính được thêm vào cơ sở dữ liệu của các khuôn mặt đã được biết. Những eigenfaces có thể được tính toán bằng cách sử dụng những hình ảnh này lưu giữ như là một tập huấn luyện mới.

2.1.3.5 Các vấn đề khác

Một số vấn đề khác cần được giải quyết để có được một hệ thống làm việc tốt. Trong phần này ta sẽ đề cập đến những vấn đề này một thời gian ngắn và chỉ ra phương pháp giải quyết.

➤ Loại bỏ nền

Trong phân tích trước đó ta đã bỏ qua ảnh hưởng của nền trong ảnh. Trong thực tế, nền có thể ảnh hưởng đáng kể đến việc nhận dạng, bởi vì phân tích eigenface như mô tả ở trên nó không nhận ra được khuôn mặt từ phần còn lại của hình ảnh. Trong các thí nghiệm được mô tả thì nền là một phần quan trọng của hình ảnh được sử dụng để phân loại các khuôn mặt.

Để đối phó với vấn đề này mà không cần phải giải quyết các vấn đề khó khăn khác (như sự phân chia mạnh mẽ của đầu) , ta đã nhân các hình ảnh khuôn mặt đầu vào bằng cửa sổ Gauss hai chiều trên khuôn mặt, do đó nền được giảm bớt và làm nổi bật khuôn mặt ở giữa. Các thí nghiệm trong chiến lược con người để nhận dạng khuôn mặt (Hay & Young, 1982) trích dẫn tầm quan trọng của tính năng khuôn mặt để nhận dạng những khuôn mặt tương tự. Bề ngoài của khuôn mặt cũng là một thực tế cần xem xét khi thay đổi kiểu tóc có thể ảnh hưởng tới việc nhận dạng.

➤ Thang chia và định hướng bất biến

Trong các thí nghiệm trên cơ sở dữ liệu của hình ảnh khuôn mặt cho thấy hiệu suất nhận dạng giảm một cách nhanh chóng như kích thước đầu, hoặc thang chia, được đánh giá sai. Kích thước đầu trong hình ảnh đầu vào phải ở trong các eigenface để hệ thống làm việc tốt. Các phân tích cảm xúc mang tới một sự đánh

giá cho kích thước đầu. Từ đó hình ảnh khuôn mặt được thay đổi kích thước phù hợp với kích thước của eigenface.

Phương pháp khác cho vấn đề về tỉ lệ, trong đó có thể được tách biệt hoặc bổ sung cho đánh giá cảm xúc, là sử dụng nhiều tỉ lệ eigenface, trong đó hình ảnh khuôn mặt của đầu vào được so sánh với eigenfaces tại một số tỷ lệ. Trong trường hợp này, hình ảnh sẽ xuất hiện để được ở gần không gian mặt của eigenface có tỉ lệ gần nhất. Tương tự, ta có thể mở rộng các hình ảnh đầu vào với nhiều kích cỡ và sử dụng các tỉ lệ mà kết quả trong các thước đo khoảng cách nhỏ nhất để đo không gian mặt.

Mặc dù phương pháp eigenfaces không phải là nhạy cảm với hướng của đầu (tức là, đầu bị nghiêng sang một bên), một cái nhìn không thẳng sẽ làm giảm hiệu suất nhận dạng. Đánh giá chính xác về trạng thái nghiêng của đầu sẽ có lợi cho việc nhận dạng. Một lần nữa, hai phương pháp đơn giản đã được xem xét và kiểm thử. Đầu tiên là tính toán hướng chuyển động của các điểm trên đầu. Phương pháp này rất ít tin cậy bởi các hình có xu hướng là một hình tròn. Tuy nhiên, thực tế cho thấy khuôn mặt là đối xứng, ít nhất là theo hướng nhìn trực diện, ta đã sử dụng toán tử đối xứng đơn giản để đánh giá hướng của đầu. Sau khi định hướng được đánh giá, hình ảnh có thể được luân chuyển để đầu ngay thẳng với các eigenface.

➤ **Phân phối trong không gian mặt**

Việc phân loại khu vực gần trong mô tả trước đây giả định được phân chia bởi Gauss trong không gian mặt của một cá nhân có vector tính năng Ω . Không có lý do nào cho sự phỏng đoán phân chia riêng biệt, chúng ta muốn mô tả nó hơn là giả định nó là Gauss. Mạng phi tuyến như mô tả trong Fleming và Cottrell (1990) dường như là một cách đầy hứa hẹn để tìm hiểu sự phân bố không gian mặt.

➤ **Nhiều hướng nhìn**

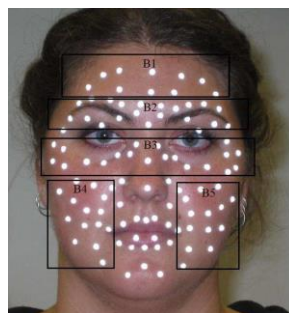
Ta đang mở rộng hệ thống để chia ra các hướng nhìn bằng cách xác định giới hạn các lớp mặt cho mỗi người đã biết tương ứng với điểm đặc trưng. Ví dụ,

một cá nhân có thể được đại diện bởi các lớp mặt tương ứng với khuôn mặt nhìn trực diện, điểm nhìn tại $\pm 45^\circ$, bên phải và bên trái hướng nhìn. Hầu hết các điều kiện nhìn thấy dường như đủ để nhận ra một khuôn mặt bất kỳ nơi nào từ hướng nhìn trực diện, vì hướng nhìn có thể xấp xỉ bằng cách thêm vào giữa các điểm cố định.

2.2 Phương pháp phát hiện biểu cảm khuôn mặt

Trong hệ thống dựa trên thông tin thị giác, dữ liệu không gian thu thập từ những điểm đánh dấu trong mỗi khung hình của video được tập hợp thành một vector tính năng bốn chiều, sau đó được sử dụng như đầu vào để phân loại. Các vùng biểu hiện trên khuôn mặt được thể hiện trong hình 2.12 dưới đây.

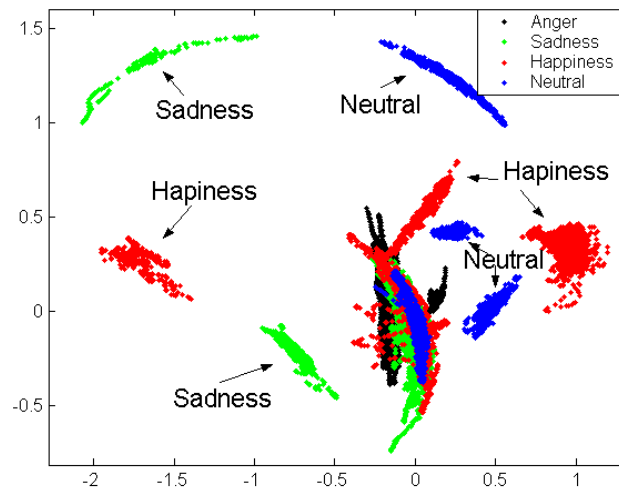
Mỗi khung dữ liệu được chia thành năm khối: trán, lông mày, mắt thấp, má phải và vùng má bên trái (xem hình 2.12). Đối với mỗi khối, phối hợp các dấu hiệu trong khối này để tạo thành một vector dữ liệu. Sau đó, phân tích thành phần phương pháp (PCA) được sử dụng để giảm số lượng các tính năng trên mỗi khung vào một vector 10 chiều cho từng khu vực, bao gồm hơn 99% sự biến đổi. Chú ý rằng những đánh dấu gần môi không được quan tâm tới, bởi những đánh dấu đó có thể nhận ra là một nụ cười, gây nhầm lẫn hệ thống nhận dạng cảm xúc.



Hình 2.12 Năm vùng của khuôn mặt được xem xét trong nghiên cứu này

Để hình dung như thế nào các vector đại diện lớp cảm xúc, hai thành phần đầu tiên của vùng mắt thấp vector được vẽ trong hình 2.13. Có thể thấy, những

cảm xúc khác nhau xuất hiện trong các cụm riêng biệt, vì vậy những đầu mối quan trọng có thể được chiết xuất từ vị trí không gian của các tính năng.



Hình 2.13 Hai thành phần Đầu tiên của vector vùng mắt thấp

2.2.1 Nhận dạng cảm xúc dựa trên phương pháp PCA

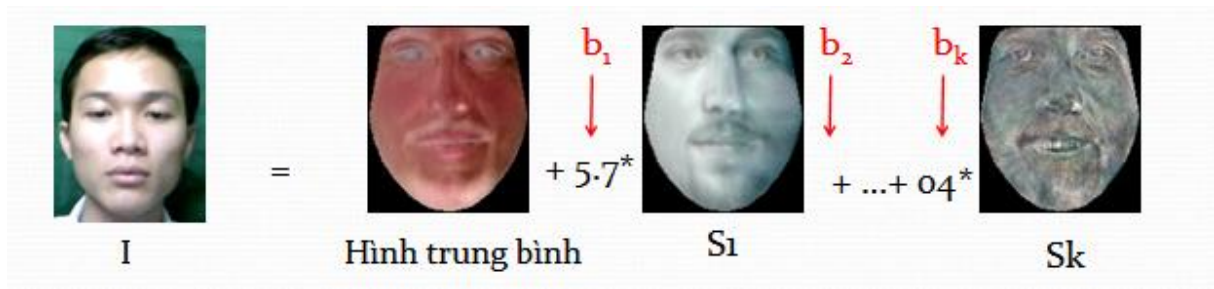
Trích chọn đặc trưng:

Xây dựng một tập các vector đặc trưng (S_1, S_2, \dots, S_k) cho mỗi hình huấn luyện sử dụng phép phân tích PCA.



Hình 2.14 Ví dụ về trích chọn đặc trưng bằng PCA

Ứng với mỗi vector đặc trưng riêng có 1 giá trị riêng. Như vậy mỗi hình huấn luyện được đại diện bởi một tập các giá trị riêng.



Hình 2.15 Hình ảnh đại diện bởi tập các giá trị riêng

Mỗi cảm xúc bao gồm 1 tập ảnh huấn luyện

Ví dụ cảm xúc vui

$$I_{(\text{Happy } 1)} = (b_{\text{Happy } 1 \ 1}, \quad b_{\text{Happy } 1 \ 2}, b_{\text{Happy } 1 \ 3} \dots b_{\text{Happy } 1 \ n})$$

$$I_{(\text{Happy } 2)} = (b_{\text{Happy } 2 \ 1}, \quad b_{\text{Happy } 2 \ 2}, b_{\text{Happy } 2 \ 3} \dots b_{\text{Happy } 2 \ n})$$

:

$$I_{(\text{Happy } m)} = (b_{\text{Happy } m \ 1}, b_{\text{Happy } m \ 2}, b_{\text{Happy } m \ 3} \dots b_{\text{Happy } m \ n})$$

Cảm xúc buồn

$$I_{(\text{Sad } 1)} = (b_{\text{Sad } 1 \ 1}, \quad b_{\text{Sad } 1 \ 2}, b_{\text{Sad } 1 \ 3} \dots b_{\text{Sad } 1 \ n})$$

$$I_{(\text{Sad } 2)} = (b_{\text{Sad } 2 \ 1}, \quad b_{\text{Sad } 2 \ 2}, b_{\text{Sad } 2 \ 3} \dots b_{\text{Sad } 2 \ n})$$

:

$$I_{(\text{Sad } m)} = (b_{\text{Sad } m \ 1}, b_{\text{Sad } m \ 2}, b_{\text{Sad } m \ 3} \dots b_{\text{Sad } m \ n})$$

Với 1 hình ảnh cần nhận dạng cảm xúc, sử dụng PCA ta được 1 tập các giá trị riêng.

$$I_{(\text{Nhan_dang})} = (b_{\text{Nhan_dang } 1}, b_{\text{Nhan_dang } 2}, b_{\text{Nhan_dang } 3} \dots b_{\text{Nhan_dang } n})$$

Quá trình nhận dạng:

Lần lượt tính khoảng cách Euclid từ ảnh cần nhận dạng đến mỗi ảnh trong tập huấn luyện

$$S_{(\text{Happy } 1)} = (S_{\text{Happy } 1,1} - b_{\text{Nhan_dang } 1})^2 + (S_{\text{Happy } 1,2} - b_{\text{Nhan_dang } 2})^2 + \dots + (S_{\text{Happy } 1,n} - b_{\text{Nhan_dang } n})^2$$

$$S_{(\text{Happy } 2)} = (S_{\text{Happy } 2,1} - b_{\text{Nhan_dang } 2})^2 + (S_{\text{Happy } 2,2} - b_{\text{Nhan_dang } 2})^2 + \dots + (S_{\text{Happy } 2,n} - b_{\text{Nhan_dang } n})^2$$

:

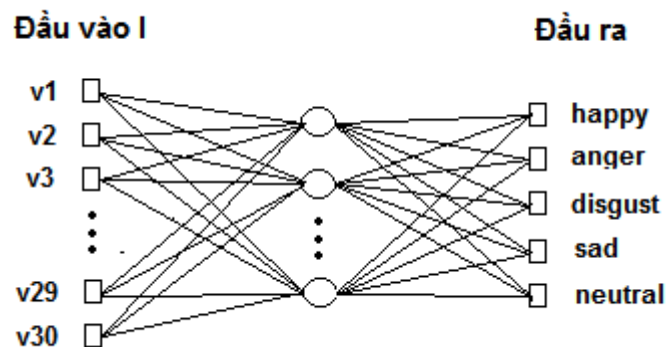
$$S_{(\text{Happy } m)} = (S_{\text{Happy } m,1} - b_{\text{Nhan_dang } 1})^2 + (S_{\text{Happy } m,2} - b_{\text{Nhan_dang } 2})^2 + \dots + (S_{\text{Happy } m,n} - b_{\text{Nhan_dang } n})^2$$

Khi đó cảm xúc của ảnh cần nhận dạng sẽ được xác định bằng cảm xúc của ảnh trong tập huấn luyện mà khoảng cách Euclid từ ảnh đó đến ảnh cần nhận dạng là bé nhất.

2.2.2 Nhận dạng cảm xúc dựa trên PCA kết hợp các thuật toán học

Mạng neuron:

Mô hình:



Hình 2.16 Mô hình mạng neuron

Hình vẽ trên cho ta mô hình của mạng neuron sử dụng trong luận văn này. Đây là mạng MLP (MultiLayer Perceptron) bao gồm 3 lớp. Lớp đầu vào gồm 30 nút là 30 giá trị riêng của 1 ảnh sau khi dùng PCA để trích chọn đặc trưng. Lớp ẩn và lớp đầu ra gồm 5 nút là 5 cảm xúc.

Trong mô hình mạng neural MPL này, sử dụng thuật toán lan truyền ngược (Backpropagation) để tiến hành học mạng, phương pháp giảm lỗi được sử dụng là phương pháp giảm gradient với hàm truyền hay hàm kích hoạt là hàm sigmoid.

Cây quyết định:

Trong lĩnh vực học máy, cây quyết định là một kiểu mô hình dự báo (*predictive model*), nghĩa là một ánh xạ từ các quan sát về một sự vật/hiện tượng tới các kết luận về giá trị mục tiêu của sự vật/hiện tượng. Mỗi một nút trong (*internal node*) tương ứng với một biến; đường nối giữa nó với nút con của nó thể hiện một giá trị cụ thể cho biến đó. Mỗi nút lá đại diện cho giá trị dự đoán của biến mục tiêu, cho trước các giá trị của các biến được biểu diễn bởi đường đi từ nút gốc tới nút lá đó. Kỹ thuật học máy dùng trong cây quyết định được gọi là học bằng cây quyết định, hay chỉ gọi với cái tên ngắn gọn là cây quyết định.

Học bằng cây quyết định cũng là một phương pháp thông dụng trong khai phá dữ liệu. Khi đó, cây quyết định mô tả một cấu trúc cây, trong đó, các lá đại diện cho các phân loại còn cành đại diện cho các kết hợp của các thuộc tính dẫn tới phân loại đó. Một cây quyết định có thể được học bằng cách chia tập hợp nguồn thành các tập con dựa theo một kiểm tra giá trị thuộc tính. Quá trình này được lặp lại một cách đệ quy cho mỗi tập con dẫn xuất. Quá trình đệ quy hoàn thành khi không thể tiếp tục thực hiện việc chia tách được nữa, hay khi một phân loại đơn có thể áp dụng cho từng phần tử của tập con dẫn xuất. Một bộ phân loại rừng ngẫu nhiên (*random forest*) sử dụng một số cây quyết định để có thể cải thiện tỉ lệ phân loại.

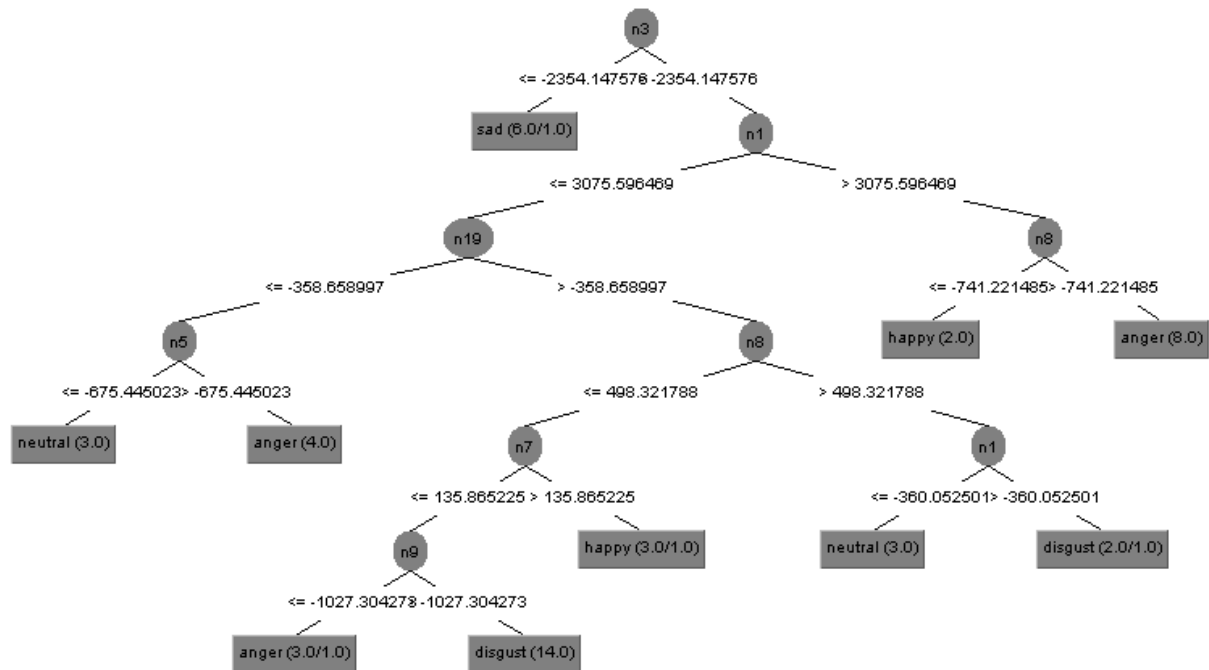
Cây quyết định cũng là một phương tiện có tính mô tả dành cho việc tính toán các xác suất có điều kiện.

Cây quyết định có thể được mô tả như là sự kết hợp của các kỹ thuật toán học và tính toán nhằm hỗ trợ việc mô tả, phân loại và tổng quát hóa một tập dữ liệu cho trước.

Dữ liệu được cho dưới dạng các bản ghi có dạng:

$$(x, y) = (x_1, x_2, x_3, \dots, x_k, y)$$

Biến phụ thuộc (*dependant variable*) y là biến mà chúng ta cần tìm hiểu, phân loại hay tổng quát hóa. $x_1, x_2, x_3 \dots$ là các biến sẽ giúp ta thực hiện công việc đó.



Hình 2.17 Cây quyết định

CHƯƠNG 3: CÀI ĐẶT VÀ THỬ NGHIỆM

Dựa trên cơ sở lý thuyết tìm hiểu được từ các chương trước, học viên sử dụng phần mềm MATLAB R2008B để cài đặt thử nghiệm chương trình. Sau đó đã đưa ra một số đánh giá nhận xét về kết quả tìm được.

3.1 Môi trường cài đặt

Chương trình nhận dạng biểu cảm khuôn mặt được cài đặt và thử nghiệm dựa trên phần mềm MATLAB R2008b.

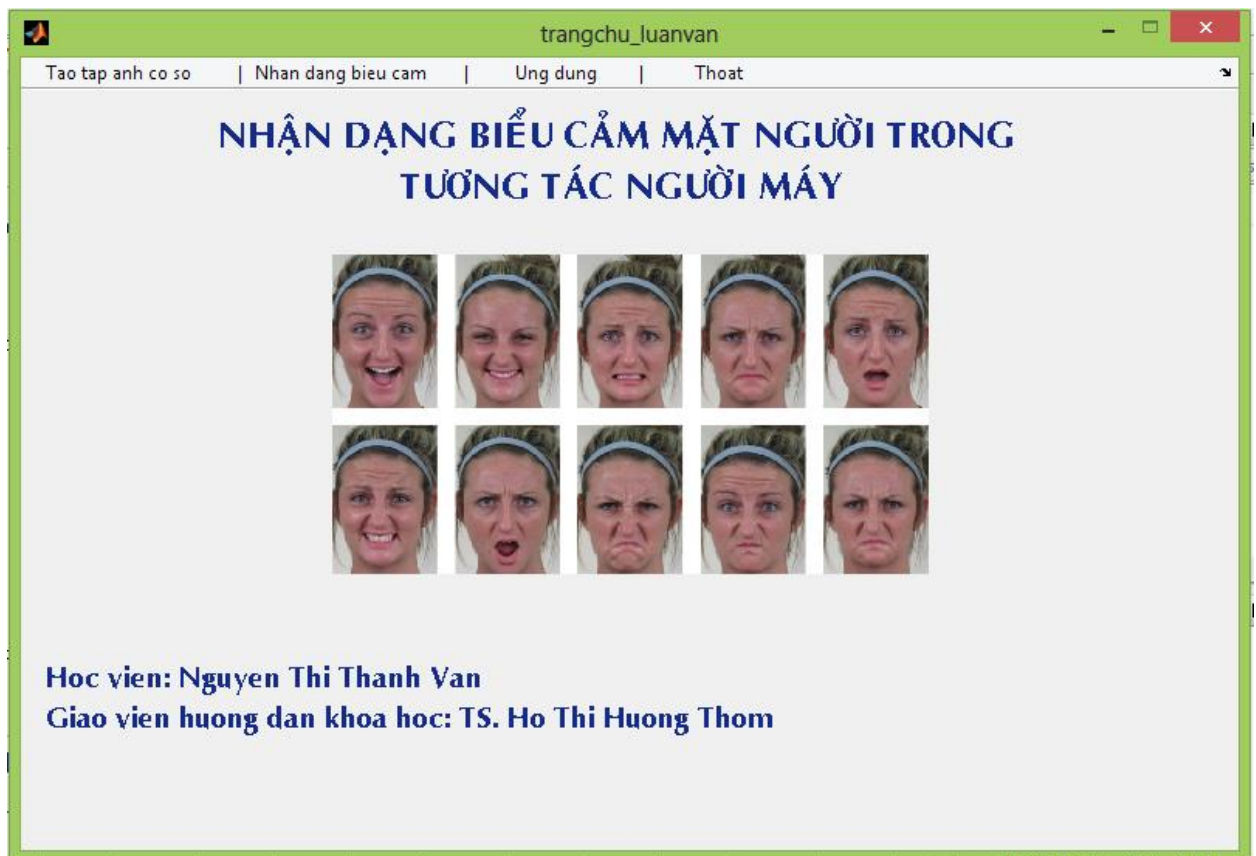
Yêu cầu phần cứng:

- Webcam 320x240 (24 bit).
- CPU: Intel Core 2 Duo P8400 2.27Ghz, 2GB RAM trở lên.

3.2 Giao diện chương trình

Dữ liệu đầu vào gồm có 50 ảnh khuôn mặt mẫu, độ phân giải 484 x 326 điểm ảnh. Tất cả các ảnh đều là khuôn mặt của một người và có độ sáng đồng đều nhau. Cảm xúc thể hiện trong mỗi ảnh khá rõ ràng. Tập dữ liệu này có 5 cảm xúc chính là: hạnh phúc, ghê tởm, giận dữ, buồn và trung lập. Trong đó có 10 ảnh bất kỳ chọn làm dữ liệu test.

Dưới đây là một số hình ảnh demo chương trình với các giao diện:



Hình 3.1 Giao diện chính của chương trình

Các chức năng tương tác gồm có: Tạo tập ảnh cơ sở, Nhận dạng biểu cảm, Ứng dụng, Thoát.

3.2.1 Tạo tập ảnh cơ sở

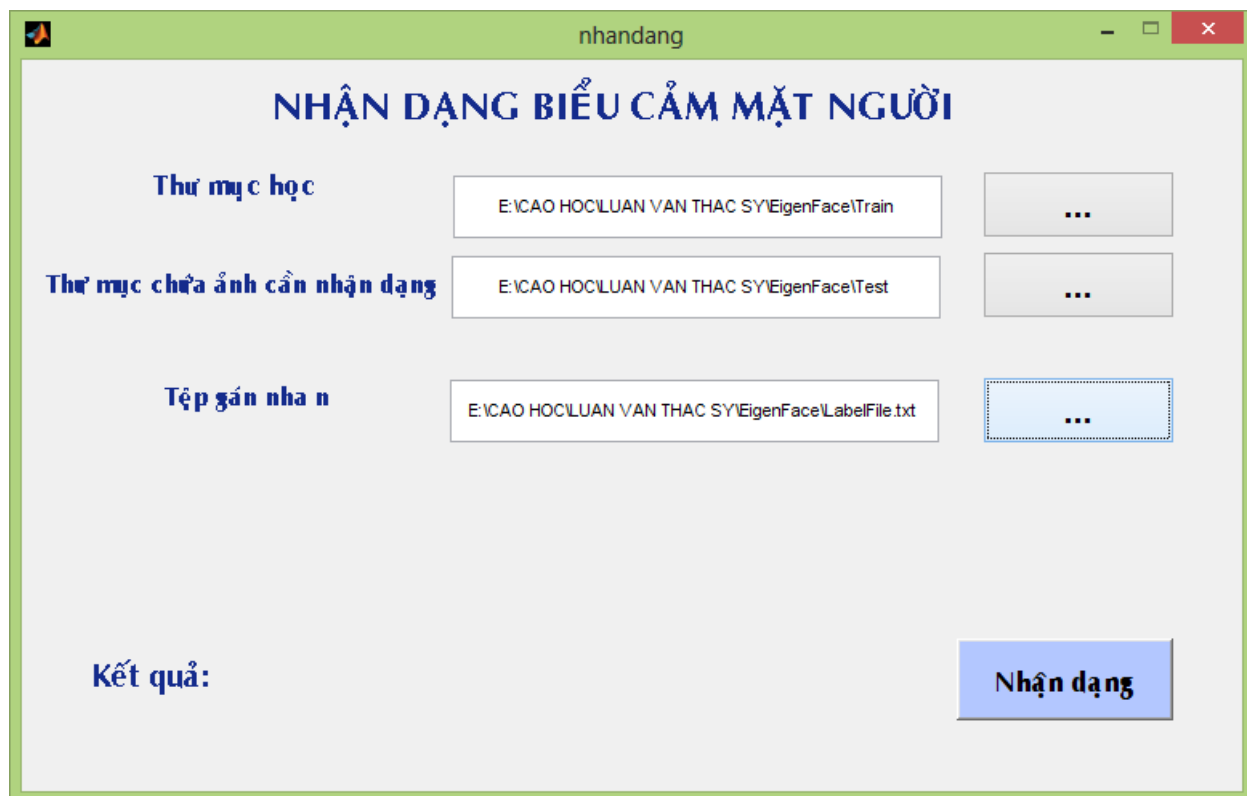
Để tạo tập huấn luyện các ảnh, vào menu “Tạo tập ảnh cơ sở” , bật webcam và thực hiện lưu ảnh như hướng dẫn vào thư mục Train. Trong thử nghiệm này, học viên lưu 50 ảnh mẫu tương ứng với 5 loại cảm xúc khác nhau: hạnh phúc, ghê tởm, giận dữ, buồn và trung lập



Hình 3.2 Tạo tập huấn luyện ảnh ban đầu

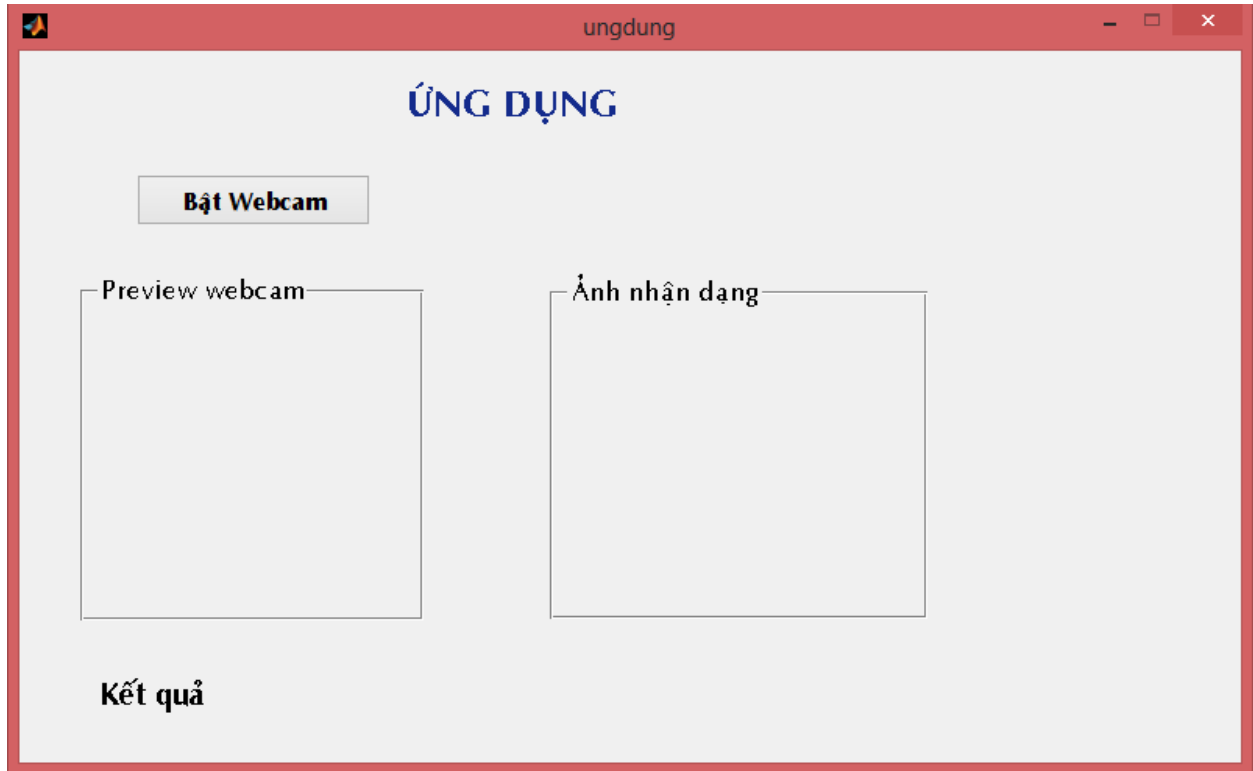
3.2.2 Nhận dạng biểu cảm

Sau khi tạo xong tập ảnh cơ sở, quá trình nhận dạng biểu cảm mặt người được thực hiện với ba bước. Bước đầu tiên chọn thư mục học chính là thư mục huấn luyện “Train” vừa tạo ở menu “Tạo tập ảnh cơ sở”. Sau đó chọn thư mục chứa ảnh cần nhận dạng là thư mục “Test”, chọn ngẫu nhiên các ảnh đã tạo hoặc ảnh khác. Cuối cùng chọn tệp gán nhãn “LabelFile.txt” và thực hiện quá trình nhận dạng



Hình 3.3 Nhận dạng và hiển thị kết quả

3.2.3 Ứng dụng để chụp ảnh

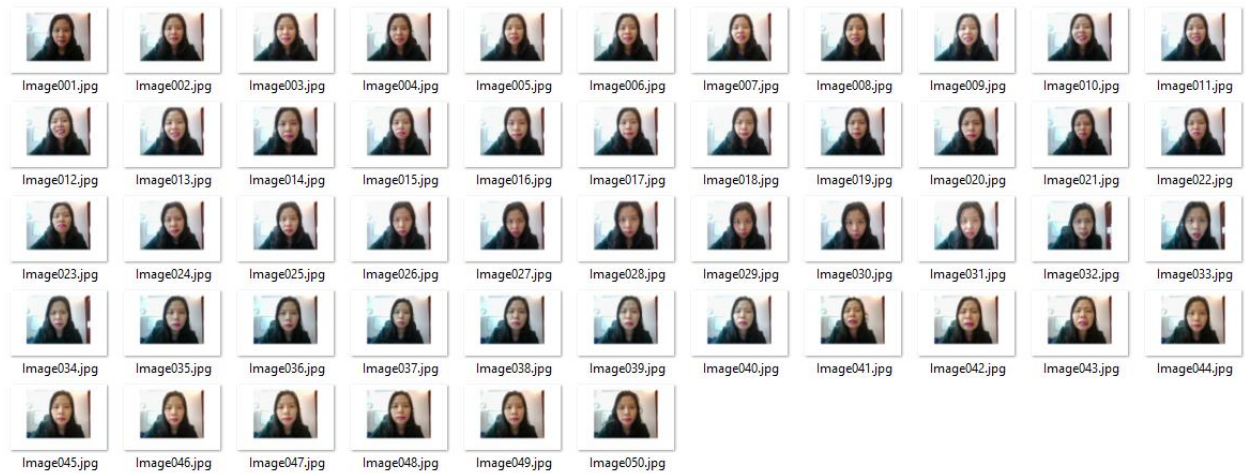


Hình 3.4 Ứng dụng của nhận dạng biểu cảm

3.3 Thử nghiệm và đánh giá

3.3.1 Thử nghiệm

Chương trình hiện tại làm việc với 50 mẫu ảnh khuôn mặt của cùng một người được lưu trong cơ sở dữ liệu, kích thước 484 x 326, ảnh được chụp với cường độ ánh sáng vừa phải, lưu dưới dạng JPG theo thứ tự từ 1 đến 50. Cảm xúc được thể hiện trong mỗi ảnh khá rõ ràng



Hình 3.5 Tập huấn luyện ảnh khuôn mặt

3.3.2 Đánh giá

3.3.2.1 Phương pháp PCA truyền thống

Với phương pháp này, kết quả nhận dạng như sau:

- Hạnh phúc: 80%
- Ghê tởm: 70%
- Giận dữ: 86%
- Buồn: 55%
- Trung lập: 84%
- Kết quả trung bình: 65%

Dựa vào kết quả, học viên nhận thấy khả năng nhận dạng của phương pháp này không cao. Tuy nhiên đây là kết quả với tập dữ liệu huấn luyện nhỏ. Nếu tập dữ liệu huấn luyện lớn hơn chắc chắn khả năng nhận dạng sẽ tăng.

Đây là phương pháp đơn giản và dễ hiểu để nhận dạng cảm xúc khuôn mặt, tuy nhiên nhược điểm lớn nhất là tốc độ xử lý chậm. Khi tập huấn luyện lớn, gồm hàng ngàn ảnh, khi đó với mỗi ảnh cần nhận dạng phải so khớp với lần lượt từng ảnh trong tập huấn luyện. Vì tốc độ chậm nên phương pháp này thường không được ứng dụng nhiều trong thực tế. Bên cạnh đó phương pháp này cũng gặp khó

khăn khi ảnh cần nhận dạng không có độ sáng tốt, hoặc khi khuôn mặt biểu cảm không chính diện.

3.3.2.2 Phương pháp sử dụng mạng nơron

Với phương pháp huấn luyện bằng mạng nơron, kết quả nhận dạng như sau:

- Hạnh phúc: 100 %
- Ghê tởm: 100%
- Giận dữ: 67%
- Buồn: 50%
- Trung lập: 80%
- Kết quả trung bình: 87%

Khi thay đổi số tầng ẩn lên lớn hơn 5 hoặc bé hơn 5, kết quả trung bình giảm xuống 83,3%. Như vậy khả năng phân loại của mạng nơron không tăng khi số lượng tầng ẩn tăng.

3.3.2.3 Phương pháp sử dụng cây quyết định

Kết quả nhận dạng như sau:

- Vui: 60 %
- Ghê tởm: 14,3%
- Giận dữ: 16,7%
- Buồn: 0%
- Trung lập: 60%
- Kết quả trung bình: 36,7%

Phương pháp này cho kết quả nhận dạng thấp, nguyên nhân có thể do tập ảnh huấn luyện ít (50 ảnh).

3.3.2.4 Tổng kết

Chương trình học viên thực hiện tốc độ nhận dạng vẫn chưa nhanh, và khi tập thử nghiệm chỉ có một ảnh cần nhận dạng thì độ chính xác của kết quả nhận dạng vẫn chưa cao.

KẾT LUẬN

Qua nghiên cứu về các phương pháp nhận dạng cảm xúc khuôn mặt, đặc biệt qua quá trình thực hiện luận văn này, học viên đã tìm hiểu được một số thuật toán và áp dụng các thuật toán này vào bài toán phân lớp để nhận dạng cảm xúc. Những kết quả chính mà luận văn đã đạt được được tổng kết như sau:

- Giới thiệu chi tiết về phương pháp trích chọn đặc trưng (PCA) và Mạng nơron nhiều tầng truyền thẳng (Multilayer Perceptron), giới thiệu sơ lược về giải thuật phân lớp cây quyết định.
- Áp dụng các giải thuật này cho bài toán nhận dạng cảm xúc khuôn mặt.
- Nhận xét và đánh giá những kết quả đạt được của các giải thuật trong bài toán nhận dạng cảm xúc.

Bên cạnh những kết quả đã đạt được, còn có những vấn đề mà thời điểm hiện tại luận văn chưa giải quyết được:

- Xây dựng tập huấn luyện lớn để đạt kết quả chính xác hơn.
- Nghiên cứu một số giải thuật trích chọn đặc trưng và phân lớp dữ liệu khác.
- Xây dựng chương trình hoàn chỉnh có giao diện tương tác với người sử dụng.

TÀI LIỆU THAM KHẢO

- [1] Lương Bá Mạnh. *Tương tác người máy*. Trường Đại học Bách Khoa Hà Nội
- [2] Nguyễn Thị Bạch Kim. *Giáo trình các phương pháp tối ưu, lý thuyết và thuật toán*. NXB Bách Khoa Hà Nội.
- [3] M. Turk, A. Pentland. *Eigenfaces for Recognition*. Journal of Cognitive Neuroscience. Vol 3, No. 1. 71-86, 1991.