

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN
BỘ MÔN CÔNG NGHỆ TRI THỨC**

**LUẬN VĂN TỐT NGHIỆP
CỬ NHÂN TIN HỌC**

ĐỀ TÀI :

**NHẬN DẠNG NGƯỜI
DỰA VÀO THÔNG TIN KHUÔN MẶT
XUẤT HIỆN TRÊN ẢNH**

**GIÁO VIÊN HƯỚNG DẪN
TS LÊ HOÀI BẮC**

SINH VIÊN THỰC HIỆN
TRẦN PHƯỚC LONG 9912606
NGUYỄN VĂN LƯỢNG 9912608

TP. HỒ CHÍ MINH, 07/ 2003

LỜI CẢM ƠN



Xin chân thành cảm ơn các thầy, các cô khoa Công Nghệ Thông Tin, Đại Học Khoa Học Tự Nhiên đã tận tình dạy dỗ, truyền đạt cho chúng tôi nhiều kiến thức quý báu.

Xin tỏ lòng biết ơn sâu sắc đến thầy Lê Hoài Bắc, người đã tận tình giúp đỡ và truyền đạt nhiều kinh nghiệm để đề tài có thể được thực hiện và hoàn thành.

Xin chân thành cảm ơn thầy Võ Đức Khánh, anh Phạm Nam Trung, anh Nguyễn Đức Hoàng Hạ, anh Hoàng Thân Anh Tuấn đã giúp đỡ, động viên chúng tôi rất nhiều trong quá trình thực hiện đề tài.

Lời cảm ơn sâu sắc nhất xin dành cho bố mẹ vì ơn sinh thành và giáo dưỡng.

Xin cảm ơn tất cả.

TP. Hồ Chí Minh tháng 07 năm 2003.

Trần Phước Long

Nguyễn Văn Lượng

LỜI MỞ ĐẦU

Trong những năm gần đây, các ứng dụng về trí tuệ nhân tạo ngày càng phát triển và được đánh giá cao. Một lĩnh vực đang được quan tâm của trí tuệ nhân tạo nhằm tạo ra các ứng dụng thông minh, có tính người đó là nhận dạng. Đối tượng cho việc nghiên cứu nhận dạng cũng rất phong phú và đa dạng. Trong đề tài này chúng tôi chọn đối tượng là khuôn mặt.

Khuôn mặt đóng vai trò quan trọng trong quá trình giao tiếp giữa người với người, và cũng mang một lượng thông tin giàu có, chẳng hạn có thể xác định giới tính, tuổi tác, trạng thái cảm xúc của người đó, ... hơn nữa khảo sát chuyển động của các đường nét trên khuôn mặt có thể biết được người đó muốn nói gì. Do đó, nhận dạng khuôn mặt là điều quan trọng và cần thiết trong xã hội loài người. Đó là lý do chúng tôi chọn đề tài :

“NHẬN DẠNG NGƯỜI DỰA VÀO THÔNG TIN KHUÔN MẶT XUẤT HIỆN TRÊN ẢNH”

Để có hệ thống nhận dạng khuôn mặt với chất lượng tốt, chúng tôi đã tiếp cận bằng hai mô hình xử lý được đánh giá là mạnh trong lĩnh vực trí tuệ nhân tạo, đó là mô hình phân cách với thuật toán SVM và mô hình thống kê với thuật toán HMM làm công cụ xử lý chính cho việc nhận dạng người dựa vào thông tin khuôn mặt trên ảnh.

Đề tài được tổ chức thành chín chương với nội dung :

- ✧ **Chương 1:** Phát biểu bài toán nhận dạng người dựa vào thông tin khuôn mặt xuất hiện trên ảnh.
- ✧ **Chương 2:** Mô tả dữ liệu.
- ✧ **Chương 3:** Dò tìm khuôn mặt.
- ✧ **Chương 4:** Rút trích đặc trưng từ khuôn mặt.
- ✧ **Chương 5:** Phương pháp SVM và ứng dụng nhận dạng khuôn mặt.
- ✧ **Chương 6:** Phương pháp Mô hình Markov ẩn và ứng dụng nhận dạng khuôn mặt.
- ✧ **Chương 7:** Thiết kế chương trình và hướng dẫn sử dụng.
- ✧ **Chương 8:** Thực nghiệm và kết quả.
- ✧ **Chương 9:** Nhận xét và hướng phát triển.

MỤC LỤC

Chương 1 PHÁT BIỂU BÀI TOÁN NHẬN DẠNG NGƯỜI DỰA VÀO THÔNG TIN KHUÔN MẶT XUẤT HIỆN TRÊN ẢNH.....	1
1.1 Tổng quan và các khái niệm liên quan đến nhận dạng khuôn mặt.....	2
1.1.1 Hệ thống sinh trắc học.....	2
1.1.2 Hệ thống nhận dạng khuôn mặt	2
1.1.3 Hệ thống xác minh hay xác thực khuôn mặt là gì?	2
1.1.4 Những thách thức trong bài toán nhận dạng khuôn mặt	3
1.2 Tổng quan về các ứng dụng tương tác người máy (Human computer interactive) liên quan đến khuôn mặt.....	4
1.3 Các hướng tiếp cận chính trong lĩnh vực nhận dạng khuôn mặt	7
1.3.1 Các công trình nghiên cứu về phương pháp nhận dạng và kiểm chứng chất lượng cho một hệ thống nhận dạng khuôn mặt	7
1.3.2 Hướng tiếp cận được thử nghiệm trong luận văn.....	10
Chương 2 MÔ TẢ DỮ LIỆU	11
2.1 Thu thập dữ liệu	12
2.2 Biểu diễn dữ liệu khuôn mặt trong máy tính	14
Chương 3 ĐÒ TÌM KHUÔN MẶT	15
3.1 Giới thiệu	16
3.1.1 Các thách thức trong việc dò tìm khuôn mặt	16
3.1.2 Tiếp cận theo khung nhìn kết hợp mạng nơron.....	18
3.1.3 Dò tìm khuôn mặt bằng phương pháp mạng neural.....	20
3.2 Chuẩn bị dữ liệu cho hệ thống dò tìm khuôn mặt.....	21
3.2.1 Giới thiệu.....	21
3.2.2 Gán nhãn và canh biên các đặc trưng khuôn mặt.....	21
3.2.3 Tiền xử lý về độ sáng và độ tương phản trên tập mẫu học	25
3.3 Phương pháp dò tìm khuôn mặt thẳng.....	27
3.3.1 Giới thiệu.....	27
3.3.2 Huấn luyện dò tìm khuôn mặt.....	28
3.3.2.1 Ảnh huấn luyện khuôn mặt.....	30
3.3.2.2 Ảnh huấn luyện không phải khuôn mặt.....	30
3.3.2.3 Phương pháp huấn luyện chủ động	31

3.3.3 Phương pháp cải tiến chất lượng dò tìm khuôn mặt	34
3.3.3.1 Các Heuristic loại bỏ thông tin thừa.....	34
3.3.3.2 Hệ thống Mạng Kết Hợp	37
Chương 4 RÚT TRÍCH ĐẶC TRƯNG TỪ KHUÔN MẶT.....	39
4.1 Tiếp cận theo phương pháp phân tích thành phần chính (Principal Component Analysis hay PCA)	40
4.1.1 Vector riêng, Trị riêng và sự chéo hoá của ma trận.....	40
4.1.2 Kỳ vọng và phương sai trong thống kê đa chiều	41
4.1.3 Kỹ thuật rút trích đặc trưng bằng phương pháp phân tích thành phần chính	42
4.2 Tiếp cận theo phương pháp Biến đổi Cosine rời rạc	47
4.2.1 Ý nghĩa phép biến đổi DCT	47
4.2.2 Các khái niệm quan trọng	47
4.2.3 Kỹ thuật mã hoá hệ số DCT	49
4.2.4 Quét Zigzag.....	53
Chương 5 SVM VÀ ỨNG DỤNG NHẬN DẠNG KHUÔN MẶT	54
5.1 Cơ sở lý thuyết của SVM.....	55
5.1.1 Các khái niệm nền tảng.....	55
5.1.1.1 Đường bao tổng quát cho một hệ máy học.....	55
5.1.1.2 Chiều VC (VC-dimension).....	56
5.1.1.3 Phân hoạch tập dữ liệu bằng các siêu mặt có hướng.....	56
5.1.1.4 Cực tiểu đường bao lỗi trên cơ sở cực tiểu chiều VC	57
5.1.1.5 Cực tiểu hoá lỗi theo cấu trúc (SRM).....	58
5.1.2 SVM tuyến tính.....	58
5.1.2.1 Trường hợp dữ liệu có thể phân cách được.....	58
5.1.2.2 Điều kiện tối ưu Karush-Kuhn-Tucker.....	61
5.1.2.3 Trường hợp dữ liệu không thể phân cách được.....	61
5.1.3 SVM phi tuyến	64
5.1.4 Chiều VC của SVM.....	68
5.1.5 Hạn chế của phương pháp SVM	68
5.2 Nhận dạng khuôn mặt người với SVM.....	69
5.2.1 Nhận dạng đa lớp dùng SVM với cây nhị phân	69
5.2.2 Nhận dạng khuôn mặt dùng SVM.....	71
5.2.2.1 Giai đoạn huấn luyện hệ thống.....	71
5.2.2.1.1 Huấn luyện SVM cho bài toán nhận dạng khuôn mặt	71
5.2.2.1.2 Vector hoá tập mẫu khuôn mặt thô.....	72

5.2.2.1.3	Rút trích đặc trưng khuôn mặt	73
5.2.2.1.4	Tạo các bộ phân loại nhị phân	75
5.2.2.1.5	Huấn luyện cho mỗi bộ phân loại nhị phân từ các tập mẫu nhị phân hoá hai lớp khuôn mặt với nhau	76
5.2.2.1.6	Khởi tạo kiến trúc cây nhị phân	87
5.2.2.2	Giai đoạn nhận dạng khuôn mặt	87
5.2.2.2.1	Nhận dạng khuôn mặt dùng SVM	87
5.2.2.2.2	Kỹ thuật nhận dạng khuôn mặt SVM	87
5.2.2.2.2.1	Vector hoá tập mẫu khuôn mặt thô	87
5.2.2.2.2.2	Rút trích đặc trưng khuôn mặt	87
5.2.2.2.2.3	Đưa mẫu thử nghiệm khuôn mặt x vào cấu trúc nhị phân và thực hiện đối sánh trên từng mô hình nhị phân SVMs	87
5.2.2.2.3	Mô phỏng quá trình nhận dạng khuôn mặt	90
5.2.3	Nhận xét và hướng phát triển tương lai	92
5.2.3.1	Ưu điểm	92
5.2.3.2	Khuyết điểm và hạn chế	93
5.2.3.3	Những đề xuất và cải tiến	93
5.2.3.3.1	Về mặt thuật toán học	93
5.2.3.3.2	Về mặt chương trình ứng dụng	94
Chương 6 MÔ HÌNH MAKOV ẨN VÀ ỨNG DỤNG NHẬN DẠNG KHUÔN MẶT		95
6.1	Giới thiệu mô hình Markov ẩn	96
6.1.1	Mô hình Markov	96
6.1.2	Mô hình Markov ẩn	97
6.1.2.1	Xác suất của chuỗi quan sát	98
6.1.2.1.1	Thủ tục tiến	99
6.1.2.1.2	Thủ tục lùi	100
6.1.2.2	Dãy trạng thái tối ưu	101
6.1.2.3	Hiệu chỉnh các tham số của mô hình	103
6.2	ỨNG DỤNG MÔ HÌNH MARKOV ẨN NHẬN DẠNG KHUÔN MẶT NGƯỜI	104
6.2.1	Ý tưởng	104
6.2.2	Nhận dạng khuôn mặt bằng mô hình Markov ẩn	105
6.2.2.1	Giai đoạn huấn luyện hệ thống	105
6.2.2.1.1	Ảnh khuôn mặt huấn luyện	105
6.2.2.1.2	Biểu diễn dữ liệu khuôn mặt theo mô hình Markov	106

6.2.2.1.3	Kỹ thuật trích đặc trưng trên mẫu khuôn mặt	109
6.2.2.1.4	Huấn luyện HMM	112
6.2.2.1.5	Đồ thị biểu diễn tác vụ học qua các vòng lặp và cực đại xác suất ước lượng mô hình từ dữ liệu quan sát.	113
6.2.2.2	Giai đoạn nhận dạng khuôn mặt.....	131
6.2.3	Nhận xét và hướng phát triển tương lai.....	131
6.2.3.1	Ưu điểm	131
6.2.3.2	Khuyết điểm	132
Chương 7	THIẾT KẾ CHƯƠNG TRÌNH VÀ HƯỚNG DẪN SỬ DỤNG..	133
7.1	Giới thiệu	134
7.2	Thiết kế và cài đặt chương trình	134
7.3	Giao diện màn hình và hướng dẫn sử dụng	135
Chương 8	THỰC NGHIỆM VÀ KẾT QUẢ.....	140
8.1	Dữ liệu và phương pháp thử nghiệm nhận dạng khuôn mặt	141
8.2	Kết quả Kết quả theo tiếp cận HMM.....	143
8.2.1	Thực nghiệm trên từng bộ tham số	143
8.2.2	Nhận xét	148
8.3	Kết quả theo tiếp cận SVM.....	148
8.3.1	Thực nghiệm trên từng bộ tham số	148
8.3.2	Nhận xét	155
8.4	So sánh kết quả HMM và SVM.....	156
Chương 9	NHẬN XÉT VÀ HƯỚNG PHÁT TRIỂN.....	158
9.1	Thuận lợi.....	159
9.2	Khó khăn.....	160
9.3	Hướng phát triển tương lai.....	161
9.4	Tổng kết.....	163

DANH SÁCH CÁC HÌNH

Hình 1-1 So sánh tác vụ nhận dạng khuôn mặt và xác minh khuôn.....	3
Hình 1-2 Mô phỏng hệ thống nhận dạng khuôn mặt	10
Hình 2-1 Dữ liệu gồm 30 người được gán nhãn theo thứ tự từ 1 đến 30.	13
Hình 2-2 Dữ liệu gồm 10 người được gán nhãn theo thứ tự từ 1 đến 10	13
Hình 2-3 Kích thước chuẩn hoá của một mẫu khuôn mặt trong tập học	14
Hình 3-1 Sơ đồ luồng xử lý các bước chính trong tiến trình dò tìm khuôn mặt.....	20
Hình 3-2 Trái: Mẫu khuôn mặt chuẩn. Phải: Các vị trí đặc trưng khuôn mặt chuẩn (tròn trắng), và phân phối của các vị trí đặc trưng thực (sau khi canh biên) từ mọi mẫu (các điểm đen).	23
Hình 3-3 Ví dụ ảnh khuôn mặt thẳng được canh biên.	23
Hình 3-4 Các bước trong việc tiền xử lý window. Đầu tiên, xây dựng hàm ánh xạ tuyến tính với các giá trị mật độ trong window, và sau đó trừ đi nó, để hiệu chỉnh về độ sáng. Tiếp theo, áp dụng cân bằng lược đồ, để hiệu chỉnh đầu vào camera khác nhau và cải thiện độ tương phản. Trong mỗi bước, việc ánh xạ được tính với các pixel bên trong hình tròn, và được áp dụng với toàn window.	26
Hình 3-5 Thuật toán dò tìm khuôn mặt.....	28
Hình 3-6 Trong khi huấn luyện, hệ thống đã huấn luyện một phần được áp dụng với các ảnh phong cảnh không chứa khuôn mặt (như bên trái). Bất kỳ vùng nào trong ảnh được dò là khuôn mặt là lỗi, và được thêm vào tập mẫu huấn luyện âm.	32
Hình 3-7 Ảnh mẫu để thử nghiệm đầu ra của bộ dò tìm thẳng.....	32
Hình 3-8 Đầu ra của mạng dò tìm.....	33
Hình 3-9 Kết quả áp dụng $\text{threshold}(4,2)$ với các ảnh trong Hình 3-8.	34
Hình 3-10 Kết quả áp dụng chồng lấp với các ảnh của Hình 9.....	35
Hình 3-11 Cơ cấu trộn nhiều dò tìm từ một mạng đơn: A) Các dò tìm được ghi trong chóp “đầu ra”. B) tính số dò tìm trong lân cận của mỗi dò tìm. C) Bước cuối cùng là kiểm tra các vị trí khuôn mặt đã đưa ra về tính chồng lấp, và D) loại bỏ các dò tìm chồng lấp nếu tồn tại.	36
Hình 3-12 AND các đầu ra từ hai mạng trên các vị trí và tỷ lệ khác nhau có thể cải thiện độ chính xác dò tìm.	37
Hình 4-1 Hai trục tương ứng với hai thành phần quan trọng nhất và ít quan trọng nhất đối với tập mẫu có hai cluster như trên.....	44
Hình 4-2 Các hàm cơ sở của phép biến đổi Cosine rời rạc, Miền quang phổ của phép biến đổi Cosine rời rạc bao gồm một mảng hai chiều 8’8, mỗi phần tử trong	

mảng là giá trị biên độ của một trong 64 hàm cơ sở.....	50
Hình 4-3 Quá trình mã hoá DCT trên một khối 8×8.....	52
Hình 4-4 Vẽ khối zigzag dạng 1	53
Hình 4-5 Vẽ khối zigzag dạng 2	53
Hình 5-1 Ba điểm trong R^2	57
Hình 5-2 Độ tin cậy VC là hàm đơn điệu theo h	57
Hình 5-3 Các tập hàm học lồng vào nhau được sắp thứ tự theo chiều VC.....	58
Hình 5-4 Siêu mặt phân cách tuyến tính cho trường hợp phân cách được và kí hiệu các support vector chính là các điểm được bao bằng viên tròn	59
Hình 5-5 Siêu mặt phân cách tuyến tính cho trường hợp không phân cách được.	63
Hình 5-6 Ảnh, trong H, với hình vuông $[-1,1] \times [-1,1] \in R^2$ dưới ánh xạ Φ	65
Hình 5-7 Trái: Cấu trúc cây nhị phân với số lớp bằng số mũ của 2. Phải: số lớp không bằng số mũ của 2.....	70
Hình 5-8 Các tác vụ huấn luyện hệ thống SVMs nhận dạng khuôn mặt.....	71
Hình 5-9 Vector hoá mẫu khuôn mặt	72
Hình 5-10 Mô phỏng phân lớp khuôn mặt giữa hai người bằng hàm tuyến tính	77
Hình 5-11 Biểu diễn số liệu bảng 1 lên đồ thị.....	79
Hình 5-12 Mô phỏng phân lớp khuôn mặt giữa hai người quá nhiều đặc trưng tương đương hay biến động.	80
Hình 5-13 Biểu diễn số liệu bảng 1(Linear), bảng 2(Poly-2), bảng 3(Poly-3), bảng 4 (Poly-4) trên cùng một đồ thị.....	84
Hình 5-14 Các tác vụ nhận dạng khuôn mặt.....	87
Hình 5-15 Mô phỏng cách ghép thành từng cặp nhị phân từ các Node lá của cây nhị phân.....	88
Hình 5-16 Kết xuất phân loại mẫu x ở cấp 1.	88
Hình 5-17 Kết quả mẫu x được nhận dạng với nhãn thuộc về khuôn mặt của người “Lớp1”.....	89
Hình 5-18 Mô phỏng cách ghép thành từng cặp nhị phân từ các Node lá của cây nhị phân.....	90
Hình 5-19 Quá trình xây dựng cây nhị phân từ cấp có L-1 cặp đến cấp có 2K/2 cặp phân loại nhị phân.....	90
Hình 5-20 Nhận dạng Mẫu thử nghiệm chưa được quan sát thuộc về Người 1 là đúng.....	91
Hình 6-1 Mô hình Markov ba trạng thái biểu diễn thời tiết.....	96
Hình 6-2 Mô phỏng mô hình Markov ẩn rời rạc bằng mô hình bình ban.....	97
Hình 6-3 Tính toán theo thủ tục tiến ở một thời điểm	99

Hình 6-4 Tính toán theo thủ tục lùi ở một thời điểm	100
Hình 6-5 Huấn luyện khuôn mặt bằng mô hình Markov ẩn rời rạc.....	105
Hình 6-6 Mẫu khuôn mặt cho việc huấn luyện mô hình Markov ẩn rời rạc với kích thước chuẩn 32x32 (pixels).....	106
Hình 6-7 Tách mẫu huấn luyện HxW thành một chuỗi các khối con PxW.	106
Hình 6-8 Mẫu khuôn mặt sẽ được tách thành 7 khối theo thứ tự từ trái sang phải với mỗi khối là 32x8(pixels)	108
Hình 6-9 Mẫu khuôn mặt được tách thành 7 khối theo thứ tự từ trên xuống dưới với mỗi khối là 32x8(pixels)	109
Hình 6-10 Khối đầu tiên trong 7 khối cần được lượng hoá thành vector quan sát.	110
Hình 6-11 Tách khối 8x8 (pixels)	110
Hình 6-12 Chuỗi quan sát từ người thứ nhất được gán nhãn “Người 1”	114
Hình 6-13 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với N = 4.....	116
Hình 6-14 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với N = 6.....	118
Hình 6-15 Các tiến trình huấn luyện HMM cho tập khuôn mặt “Người 1” với N = 8.....	120
Hình 6-16 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với N = 10.....	121
Hình 6-17 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với M = 2	124
Hình 6-18 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với M = 4.....	126
Hình 6-19 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với M = 6	128
Hình 6-20 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với M = 8	129
Hình 6-21 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với M = 10nh Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với M = 10	131

DANH SÁCH CÁC BẢNG

Bảng 4-1 Dữ liệu trên Matrxận hai chiều 8x8	51
Bảng 4-2 Dữ liệu qua phép biến đổi 2D-DCT	52
Bảng 5-1 Số Vector hỗ trợ tính được từ 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “lớp 1” với 29 lớp khuôn mặt khác.....	79
Bảng 5-2 Kết quả của việc huấn luyện từ 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “Lớp 1” với các khuôn mặt của 29 người còn lại bằng SVM phi tuyến có dạng đa thức bậc 2 (Poly-2).	83
Bảng 5-3 Kết quả của việc huấn luyện từ 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “Lớp 1” với các khuôn mặt của 29 người còn lại bằng SVM phi tuyến có dạng đa thức bậc 2 (Poly-3).	83
Bảng 5-4 Kết quả của việc huấn luyện từ 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “Lớp 1” với các khuôn mặt của 29 người còn lại bằng SVM phi tuyến có dạng đa thức bậc 2 (Poly-4).	83
Bảng 6-1 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc với số trạng thái là 4 và hệ số Mixture thay đổi từ 2→20.....	116
Bảng 6-2 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc với số trạng thái là 6 và hệ số Mixture thay đổi từ 2→12.....	118
Bảng 6-3 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc với số trạng thái là 8 và hệ số Mixture thay đổi từ 2→16.....	119
Bảng 6-4 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc với số trạng thái là 10 và hệ số Mixture thay đổi từ 2→10.....	121
Bảng 6-5 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 2 và hệ số trạng thái thay đổi từ 4→10.....	123
Bảng 6-6 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 4 và hệ số trạng thái thay đổi từ 4→10.....	125
Bảng 6-7 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 6 và hệ số trạng thái thay đổi từ 4→10.....	127
Bảng 6-8 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 8 và hệ số trạng thái thay đổi từ 4→10.....	128
Bảng 6-9 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 10 và hệ số trạng thái thay đổi từ 4→10.....	129
Bảng 8-1 Mô tả dữ liệu thử nghiệm thu thập từ mỗi người trong hệ thống nhận dạng	142

Bảng 8-2 Kết quả hệ thống nhận dạng theo mô hình Markov với số trạng thái $N = 4$	144
Bảng 8-3 Kết quả hệ thống nhận dạng theo mô hình Markov với số trạng thái $N = 6$	145
Bảng 8-4 Kết quả hệ thống nhận dạng theo mô hình Markov với số trạng thái $N = 8$	146
Bảng 8-5 Kết quả hệ thống nhận dạng theo mô hình Markov với số trạng thái $N = 10$	147
Bảng 8-6 Kết quả nhận dạng tốt nhất với phương pháp mô hình Markov tại $N = 6$ và $M = 10$	148
Bảng 8-7 Kết quả nhận dạng với phương pháp SVMs với $C = 30$	150
Bảng 8-8 Kết quả nhận dạng với phương pháp SVMs với $C = 50$	151
Bảng 8-9 Kết quả nhận dạng với phương pháp SVMs với $C = 100$	152
Bảng 8-10 Kết quả nhận dạng với phương pháp SVMs với $C = 200$	153
Bảng 8-11 Kết quả nhận dạng với phương pháp SVMs với $C = 400$	154
Bảng 8-12 Kết quả nhận dạng tốt nhất với phương pháp SVMs tại $C = 400$ và K là hàm xử lý chính dạng đa thức bậc 3.....	155

Chương 1 PHÁT BIỂU BÀI TOÁN NHẬN
DẠNG NGƯỜI DỰA VÀO THÔNG TIN
KHUÔN MẶT XUẤT HIỆN TRÊN ẢNH

Khoa CNTT - ĐH KHTN TP.HCM

1.1 Tổng quan và các khái niệm liên quan đến nhận dạng khuôn mặt

1.1.1 Hệ thống sinh trắc học

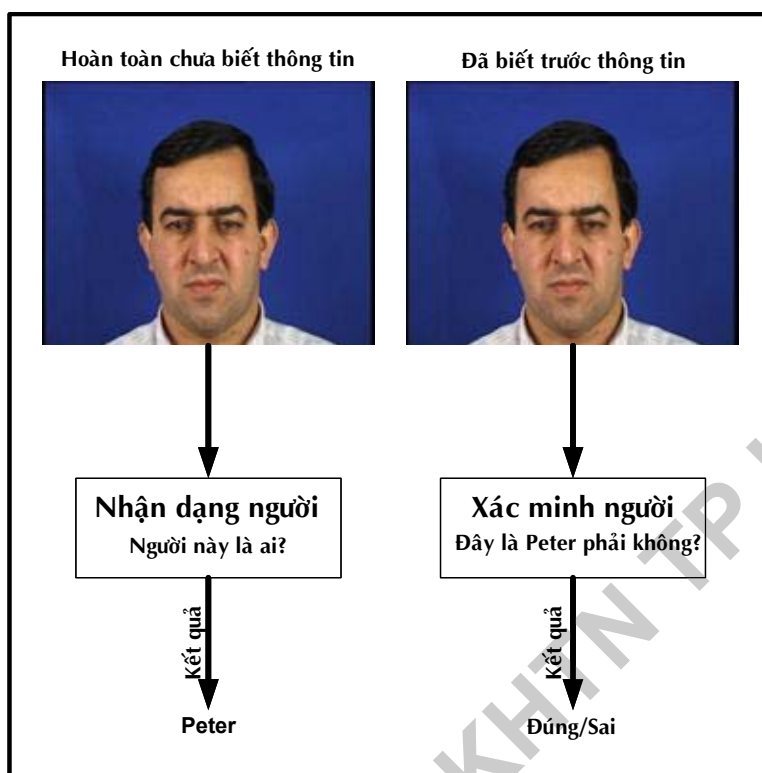
Hệ thống sinh trắc học là một hệ thống được thiết kế để xác minh và nhận dạng một người dựa vào những đặc trưng sinh học duy nhất của người đó.

1.1.2 Hệ thống nhận dạng khuôn mặt

Hệ thống nhận dạng khuôn mặt là một hệ thống được thiết kế để tìm thông tin của một người. Kỹ thuật nhận dạng là kiểm tra sự phù hợp dựa trên phép so sánh một-nhiều cụ thể là tìm ra một người là ai trong số những người đã được lưu trữ trong hệ thống dựa vào thông tin khuôn mặt.

1.1.3 Hệ thống xác minh hay xác thực khuôn mặt là gì?

Hệ thống xác minh/xác thực khuôn mặt là một hệ thống được thiết kế để xác minh thông tin của một người. Kỹ thuật xác minh là kiểm tra sự phù hợp trên phép so sánh một-một cụ thể là đối chiếu thông tin mới nhận về một người với thông tin đã lưu trữ về người này có khớp hay không dựa trên thông tin khuôn mặt.



Hình 1-1 So sánh tác vụ nhận dạng khuôn mặt và xác minh khuôn

1.1.4 Những thách thức trong bài toán nhận dạng khuôn mặt

Những biến đổi quá lớn giữa các ảnh khuôn mặt khác nhau từ một người cần nhận dạng gồm trạng thái cảm xúc trên khuôn mặt, ánh sáng, và các thay đổi vị trí của khuôn mặt..vv.

Giới hạn về số ảnh cần thiết cho việc nhận dạng, tập học không thể bao quát được tất cả các biến đổi có thể có trên khuôn mặt của một người cần nhận dạng trong thế giới thực.

1.2 Tổng quan về các ứng dụng tương tác người máy

(Human computer interactive) liên quan đến khuôn mặt

Từ những năm 1990 trở lại đây, chúng ta đã chứng kiến sự phát triển như vũ bão của các ngành công nghiệp, đặc biệt là ngành công nghiệp chế tạo điện tử. Tuy nhiên hiện nay các thiết bị điện tử cao cấp như máy ảnh số, camera kỹ thuật số, và nhiều sản phẩm khác dường như chỉ phù hợp cho các phòng thí nghiệm, các công ty sản xuất kinh doanh, thương mại, tài chính, ngân hàng, ... Trong thời gian không xa từ 3 đến 10 năm nữa, chi phí cho các thiết bị này sẽ giảm đáng kể. Khi đó sẽ mở ra nhiều hướng nghiên cứu về thị giác máy tính, đồng thời sẽ có nhiều ứng dụng trong giao tiếp giữa người với máy tính mà trong đó hệ thống nhận dạng mặt người đóng một vai trò không nhỏ. Dưới đây chúng tôi liệt kê một số ứng dụng.

➤ Các ứng dụng chuyên biệt cho ngành hàng không

- ✓ Đảm bảo sự truy cập và tính hợp lệ trong công việc cho từng nhân viên: Mỗi nhân viên làm việc tại cảng hàng không cũng như nhân viên phi hành đoàn được cung cấp quyền truy cập để đến vị trí làm việc. Làm thế nào để xác minh nhân viên này vào đúng khu vực làm việc hay không?
- ✓ Làm sao để đảm bảo trong số những hành khách không có sự trà trộn của một số kẻ khủng bố/tội phạm quốc gia/ quốc tế?

➤ **Bảo vệ trẻ em ở nhà trẻ từ bọn bắt cóc**

- ✓ Quy định rằng, chỉ có những nhân viên của nhà trẻ mới được phép dẫn trẻ em ra ngoài và trao tận tay cho bố mẹ đón về. Nhưng trong xã hội cũng có một số trường hợp giả danh nhân viên để bắt cóc trẻ em với mục đích xấu. Làm thế nào để ngăn chặn hành vi xấu này?

➤ **Nhận dạng khuôn mặt được sử dụng kèm với thẻ quy cập**

- ✓ Trong các nước phát triển, hầu như mọi người dân đều dùng thẻ tín dụng để mua bán, rút tiền, trao đổi hàng hóa. Điều này rất nguy hiểm khi thẻ truy cập này bị người khác nhặt được hay biết được mật khẩu của sở hữu thẻ này? Làm cách nào có thể bảo đảm an toàn nhất?

Có thể dùng song mật khẩu: Có nghĩa sử dụng khuôn mặt như là một mật khẩu thứ hai để truy cập vào hệ thống cùng với thông tin từ card truy cập. Để rút được tiền

- Đưa thẻ vào hệ thống
- Đưa khuôn mặt vào để nhận dạng
- Xác minh người này có phải là chủ sở hữu của thẻ hay không?

Nếu khớp thì hệ thống cho rút tiền

Nếu không thì hệ thống không cho rút tiền.

- **Kinh doanh thương mại điện tử**
 - ✓ Với sự tiên bộ của khoa học công nghệ, nhiều hình thức kinh doanh thương mại xuất hiện, đặc biệt là thương mại điện tử. Việc buôn bán và trao đổi giữa hai bên đối tác không cần diễn ra trực tiếp (mặt đối mặt), mà chỉ cần qua mạng với hình ảnh của người đại diện. Tuy nhiên bên cạnh đó sẽ có nhiều mặt tiêu cực trên hình thức kinh doanh này, đó là các vụ lừa đảo, giả mạo, giả danh..vv. Làm sao để biết được đối tác của mình là thật hay giả?
- **Ngăn chặn việc xuất/nhập cảnh bất hợp pháp**
 - ✓ Một số người không được xuất/nhập cảnh vào nước, song họ cố tình khai gian giấy tờ để xuất/nhập cảnh bất hợp pháp. Làm sao để ngăn chặn được sự gian lận này?
- **Lần dấu vết đi tìm kẻ khủng bố**
 - ✓ Từ những bức ảnh số hay những đoạn video số đã được ghi lại tự động về hiện trường trước khi vụ khủng bố xảy ra. Cần nhận dạng những đối tượng khả nghi của vụ khủng bố này?
- **Hệ thống giám sát công nhân và chấm công tự động**
 - ✓ Hiện nay trong các khu công nghiệp hay những công ty sản xuất lớn có hàng ngàn công nhân vào ra mỗi ngày nên việc giám sát kẻ gian vào công ty cũng như công việc chấm công rất phức tạp. Vậy làm thế nào để nhận ra từng nhân viên của công ty.

- ❖ **Tóm lại:** nhu cầu sử dụng các hệ thống xử lý dùng trí tuệ nhân tạo ngày càng phát triển, mà trong đó nhận dạng khuôn mặt để mã hóa mặt khẩu cá nhân là một nhu cầu thiết yếu hiện nay và trong tương lai. Đặc biệt vụ khủng bố ngày 11-9-2001 tại Mỹ đã đánh dấu một bước ngoặt mới trong xu hướng nghiên cứu và giá trị thương mại của các hệ thống sinh trắc học để bảo vệ sự an toàn cho con người.

1.3 Các hướng tiếp cận chính trong lĩnh vực nhận dạng khuôn mặt

1.3.1 Các công trình nghiên cứu về phương pháp nhận dạng và kiểm chứng chất lượng cho một hệ thống nhận dạng khuôn mặt

Bài toán nhận dạng khuôn mặt cần xác định hai vấn đề chính: dùng thông tin nào để nhận dạng: chân mày, cặp mắt, mũi, môi, tai, hay kết hợp các thông tin trên. Và dùng phương pháp nào để huấn luyện cho máy nhận dạng dùng nguồn thông tin đó. Nhận dạng khuôn mặt trên máy tính đã trải qua nhiều bước thăng trầm với các kết quả như sau:

- Wenyi Zhao, Arvinth Krishnaswamy, Rama Chellappa, Danie L.Swets, John Weng (1998)[1] sử dụng phương pháp PCA (phân tích thành phần chính) kết hợp LDA (phân tích độc lập tuyến tính). Bước 1, chiếu ảnh khuôn mặt từ không gian ảnh thô sang không gian các không gian khuôn mặt (Mỗi lớp khuôn mặt được nhận dạng sẽ được mô hình hóa bằng một không gian khuôn mặt) dùng PCA. Bước 2, sử dụng phương pháp LDA để tạo bộ phân loại tuyến tính có khả năng phân lớp các lớp khuôn mặt.
- John Daugnman (1998)[2], đưa ra phương pháp dùng đặc trưng về tròng của mắt để phân biệt cặp (tra/gái) song sinh.
- Emmanuel Viennet và Francoise Fogelman Soulie (1998),[3] sử dụng phương pháp mạng neural nhân tạo để xử lý và nhận dạng khuôn mặt.
- Antonio J.Colmenarez và Thomas S.Huang (1998),[4] sử dụng kỹ thuật học thị giác và phù hợp mẫu 2-D. Ông quan niệm bài toán dò tìm khuôn mặt là thao tác phân loại khuôn mặt trong đó khuôn mặt thuộc về một lớp và các đối tượng khác thuộc về lớp còn lại bằng cách ước lượng mô

hình xác suất cho mỗi lớp, và việc dò tìm sử dụng luật quyết định Maximum-likelihood.

- Kazunori Okada, Johannes Steffens, Thomas Maurer, Hai Hong, Egor Elagin, Hartmut Neven, and Christoph (1998),[5] nhận dạng khuôn mặt dựa vào sóng Gabor và phương pháp phù hợp đồ thị bó. Với ý tưởng dùng đồ thị để biểu diễn khuôn mặt, ảnh khuôn mặt được đánh dấu tại các vị trí đã được xác định trước trên khuôn mặt, gọi các vị trí này chính là các vị trí chuẩn. Khi thực hiện thao tác so khớp đồ thị với một ảnh, các điểm chuẩn (Jets) sẽ trích ra từ ảnh và so sánh các điểm chuẩn này với tất cả các điểm chuẩn tương ứng trong các đồ thị khác nhau, và đồ thị nào phù hợp nhất với ảnh sẽ được chọn.
- Baback Moghaddam và Alex Pentland (1998) [6], đưa ra phương pháp phù hợp thị giác trực tiếp từ các ảnh cần sử dụng cho mục đích nhận dạng khuôn mặt và dùng độ đo xác suất để tính độ tương tự.

Khoa CNTT - ĐH KHTN TP HCM

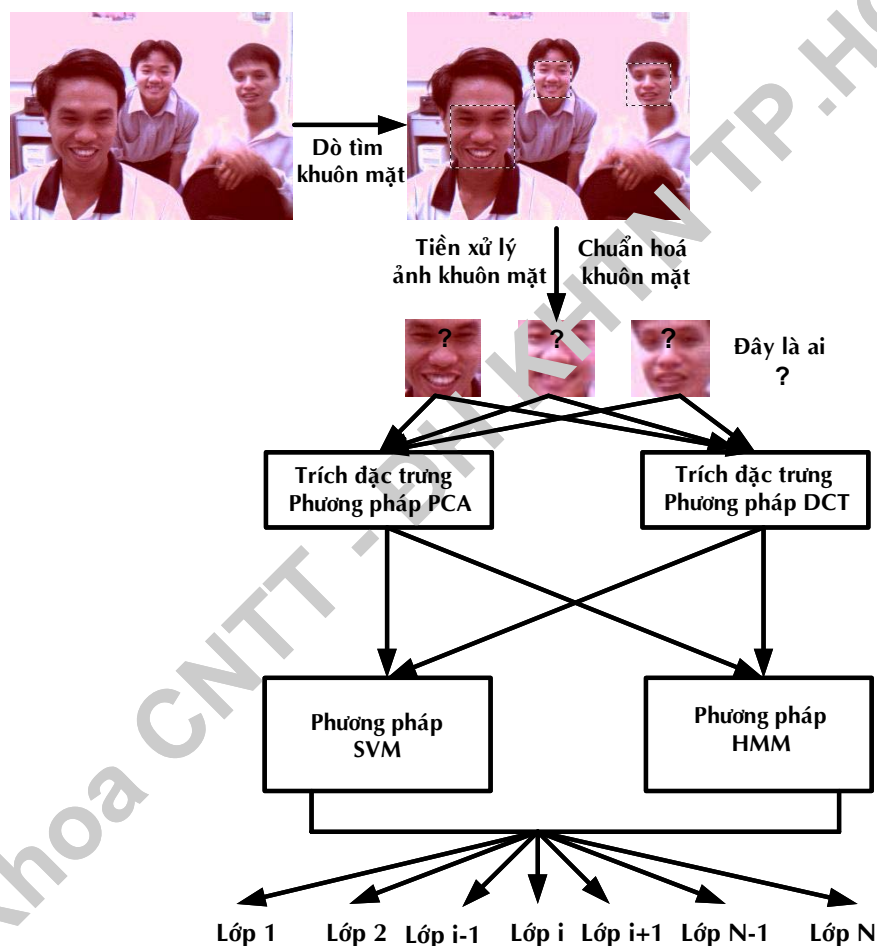
- Massimo Tistaelli và Enrico Grosso (1998) [7], đưa ra kỹ thuật thị giác động. Vì khả năng quan sát các chuyển động của khuôn mặt và xử lý các tính hướng theo dự định là thông tin rất quan trọng, từ đó nhận được mô tả đầy đủ hơn về khuôn mặt cho mục đích thu thập mẫu và nhận dạng.
- Jeffrey Huang, Chengjun Liu, và Harry Wechsler (1998)[8], đề xuất thuật toán căn cứ trên tính tiến hóa (Evolutionary computation) và di truyền (Genetic) cho các tác vụ nhận dạng khuôn mặt. Đối với cách tiếp cận này, hai mắt sẽ được dò tìm trước tiên và thông tin này được xem là vết để quan sát khuôn mặt, trình xử lý dò tiếp mắt bằng cách sử dụng một thuật toán lai để kết hợp thao tác học và tiến hóa trong quá trình học.
- Daniel Bgraham và Nigel M Allinson (1998)[9], sử dụng phương pháp được gọi là tạo bản sao không gian đặc trưng để biểu diễn và nhận dạng hướng di chuyển của khuôn mặt.
- Oi Bin Sun, Chian Prong Lam và Jian Kang Wu (1998)[10], sử dụng phương pháp tìm vùng hai chân mày, hai mắt, mũi, miệng và cằm. Ảnh khuôn mặt thẳng ban đầu được chiếu theo chiều ngang để tìm các giá trị điểm ảnh thỏa ngưỡng cho trước, đồ thị biểu diễn theo trục ngang sẽ định vị vị trí biên trên và biên dưới của hình chữ nhật bao các đặc trưng cục bộ khuôn mặt. Tương tự với chiều đứng để tìm ra đường biên bên trái và phải cho các vùng đặc trưng.
- Ara V.Nefian và Monson H.Hayes III (1998)[12] trình bày hướng tiếp cận theo mô hình mô hình Markov ẩn (HMM) trong đó ảnh mẫu khuôn mặt được lượng hóa thành chuỗi quan sát trên khuôn mặt theo quan niệm dựa trên thứ tự xuất hiện các đặc trưng khuôn mặt {hai chân mày, hai lông mi, mũi, miệng, cằm}. Trong chuỗi quan sát đó, mỗi quan sát lại là một vector nhiều chiều và mỗi vector quan sát này được sử dụng để đặc trưng cho mỗi trạng thái trong chuỗi trạng thái của HMM. Mỗi người được ước lượng bằng một mô hình của HMM.
- Guodong Guo, Stan Z.Li, Kap Luk Chan (17 January 2001) [13], dùng phương pháp SVM để nhận dạng khuôn mặt. Sử dụng chiến lược kết hợp nhiều bộ phân loại nhị phân để xây dựng bộ phân loại SVM đa lớp.

1.3.2 Hướng tiếp cận được thử nghiệm trong luận văn

Trong đề tài này chúng tôi thử nghiệm hai phương pháp nhận dạng: SVM và HMM. Hai phương pháp: PCA (phân tích thành phần chính) và DCT (biến đổi Cosine rời rạc) để rút ra các vector đặc trưng làm đầu vào cho hai bộ nhận dạng trên.

Việc cô lập khuôn mặt trong ảnh đầu vào (ảnh chứa khuôn mặt) được thực hiện với phương pháp dò tìm khuôn mặt trong ảnh dùng mạng neural.

Sơ đồ hệ thống nhận dạng khuôn mặt được minh họa trong hình sau:



Hình 1-2 Mô phỏng hệ thống nhận dạng khuôn mặt

Chương 2 MÔ TẢ DỮ LIỆU

Khoa CNTT - ĐH KHTN TP.HCM

2.1 Thu thập dữ liệu

Cơ sở dữ liệu ảnh khuôn mặt gồm 30 người được thu thập từ nhiều nguồn khác nhau. Ảnh của 10 người đầu tiên được lấy từ website <http://www.humanscan.de/support/downloads/facedb.php> của công ty Human Scan và nguồn dữ liệu này chuyên phục vụ cho bài toán dò tìm khuôn mặt, Ảnh của 3 người tiếp theo được lấy từ website <http://www.mis.atr.co.jp/~mlyons/>, Kyushu University, mỗi người gồm 20 ảnh khác nhau, và nguồn dữ liệu này chuyên phục vụ cho bài toán nhận dạng cảm xúc, 17 người còn lại từ được lấy từ website <http://cswww.essex.ac.uk/projects/vision/allfaces>, mỗi người bao gồm 20 ảnh khác nhau, và nguồn dữ liệu này chuyên phục vụ cho các ứng dụng nhận dạng khuôn mặt. Cơ sở dữ liệu này được minh hoạ trong Hình 2-1.

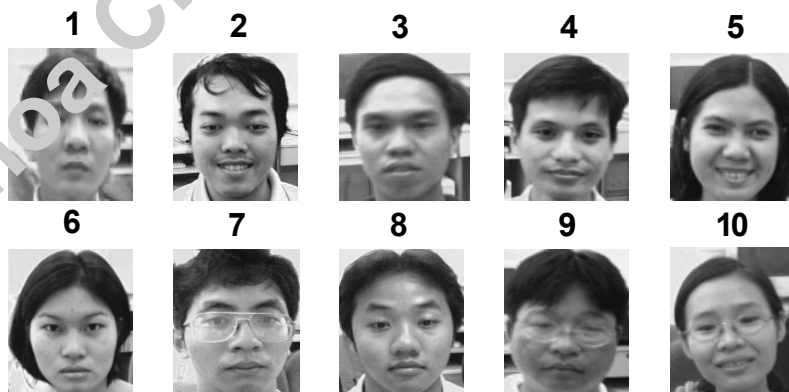
Ngoài ra, còn có tập dữ liệu do chúng tôi tạo ra trong lúc thực hiện đề tài. Đó là dữ liệu được thu thập bằng WebCam gồm 10 người khác nhau. Chính sự chủ động trong việc tạo mẫu nên số lượng ảnh khoảng trên 50 ảnh / 1 người. Tập mẫu này được minh hoạ trong Hình 2-2.

Nhận xét về tập mẫu dữ liệu: Hầu hết các khuôn mặt xuất hiện trong ảnh là khuôn mặt trực diện với mặt phẳng ảnh và mỗi khuôn mặt đều đầy đủ thông tin đặc trưng như {Hai chân mày, hai mắt, mũi, miệng, cằm}. Một số khuôn mặt quay với một góc không đáng kể.

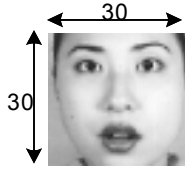
Kích thước chuẩn hoá của mỗi mẫu trong tập huấn luyện 30×30 (pixels) hoặc 32×32 (pixels) như mô tả trên Hình 2-3. Tùy thuộc vào đặc trưng xử lý của mỗi thuật toán ta sử dụng một trong hai dạng kích thước ảnh chuẩn trên.



Hình 2-1 Dữ liệu gồm 30 người được gán nhãn theo thứ tự từ 1 đến 30.



Hình 2-2 Dữ liệu gồm 10 người được gán nhãn theo thứ tự từ 1 đến 10



Hình 2-3 Kích thước chuẩn hoá của một mẫu khuôn mặt trong tập học

2.2 Biểu diễn dữ liệu khuôn mặt trong máy tính

Dữ liệu ảnh biểu diễn bên trong máy tính là cường độ sáng của điểm ảnh, tại vị trí x và y : $(I(x,y))$.

Để biểu diễn dữ liệu cho các thuật toán học nhận dạng, ta dùng hai cách tổ chức dữ liệu như sau:

- Đọc từng dòng ảnh theo thứ tự từ trên xuống, mỗi dòng ảnh được bố trí liên tục nhau trên một mảng số thực một chiều. Như vậy từ ảnh có kích thước 30×30 (pixels) ta biểu diễn thành mảng vector một chiều trong máy tính $x = (x^1, x^2, \dots, x^{900})$. Đây là cách bố trí để thí nghiệm cho phương pháp PCA và SVM.
- Đọc từng khối ảnh có kích thước 8×32 (pixel) theo thứ tự khối dưới chông lấp khối trên một nửa kích thước tính theo chiều cao, trên mỗi khối ảnh này ta lại tiếp tục tách ra mỗi khối con 8×8 liên tục nhau. Từ khối 8×8 (pixels), chúng tôi chọn ra 20 hệ số đặc trưng từ phép biến đổi trên miền tần số. Mỗi khối ảnh 8×32 sẽ được lượng hoá thành mỗi vector một chiều. Như vậy đối với ảnh mỗi khuôn mặt ta biểu diễn trong máy tính thành một chuỗi các vector một chiều liên tiếp nhau. Đây là cách bố trí để thí nghiệm cho phương pháp DCT và HMM.

Chương 3 DÒ TÌM KHUÔN MẶT

Khoa CNTT - ĐH KHTN TP.HCM

3.1 Giới thiệu

Dò tìm đối tượng là bài toán cơ bản và quan trọng trong lĩnh vực thị giác máy tính. Các kỹ thuật đã được áp dụng có thể chia thành một trong hai tiếp cận: so khớp các mô hình hình học hai, ba chiều vào ảnh [Seutens et al., 1992, Chin và Dyer, 1986, Besl và Jain, 1985], hay phương pháp so khớp các mô hình khung vào ảnh có chứa khuôn mặt cần dò tìm. Các nghiên cứu trước đây cho thấy rằng các phương pháp dựa trên khung nhìn có thể dò tìm các khuôn mặt thẳng trong nền phức tạp một cách hiệu quả.

Việc phát triển bộ dò tìm đối tượng dựa trên khung nhìn dùng máy học có ba vấn đề chính. Thứ nhất, ảnh của các đối tượng (chẳng hạn khuôn mặt) biến đổi nhiều, tùy thuộc vào độ sáng, tình trạng che lấp, tư thế, biểu hiện khuôn mặt và tính giống nhau. Thuật toán dò tìm giải quyết với càng nhiều biến đổi càng tốt. Thứ hai, một hay nhiều mạng neural được huấn luyện để giải quyết với mọi biến đổi còn lại trong việc phân biệt đối tượng (object) với không phải đối tượng (non-object). Thứ ba, đầu ra từ các bộ dò tìm phải được kết hợp lại thành một quyết định có biểu diễn đối tượng hay không.

Hai bài toán dò tìm và nhận dạng đối tượng có liên quan mật thiết. Hệ thống nhận dạng đối tượng có thể xây dựng mà không có tập bộ dò tìm đối tượng, mỗi bộ dò tìm dò một đối tượng quan tâm. Tương tự, bộ dò tìm đối tượng có thể được xây dựng mà không có hệ thống nhận dạng đối tượng; bộ nhận dạng đối tượng này cần phân biệt đối tượng mong muốn với mọi đối tượng khác có thể xuất hiện hay là lớp đối tượng chưa biết. Do đó hai bài toán là như nhau, dù trong thực hành hầu hết các hệ thống nhận dạng đối tượng ít khi giải quyết nền tùy ý, và các hệ thống dò tìm đối tượng ít khi được huấn luyện trên đủ loại đối tượng để xây dựng hệ thống nhận dạng. Điểm chú trọng khác nhau của các bài toán này dẫn đến các trình bày và thuật toán khác nhau.

Thông thường, các hệ thống nhận dạng khuôn mặt làm việc bằng cách trước hết áp dụng bộ dò tìm khuôn mặt để định vị khuôn mặt, sau đó áp dụng thuật toán nhận dạng để nhận diện khuôn mặt.

3.1.1 Các thách thức trong việc dò tìm khuôn mặt

Việc dò tìm đối tượng là bài toán xác định cửa sổ con của ảnh có thuộc về tập các ảnh của đối tượng quan tâm hay không. Do đó, đường biên quyết định của tập ảnh đối tượng phức tạp sẽ làm tăng độ khó của bài toán và có thể tăng số lỗi dò tìm.

Giả sử ta muốn dò khuôn mặt nghiêng trong mặt phẳng ảnh, ngoài các

khuôn mặt thẳng. Việc thêm các khuôn mặt nghiêng vào tập các ảnh ta muốn dò tìm làm tăng độ biến thiên của tập, và có thể làm tăng độ phức tạp của đường biên quyết định của tập ảnh. Độ phức tạp này làm bài toán dò tìm khó hơn. Việc thêm ảnh mới vào tập ảnh đối tượng có thể làm đường biên quyết định đơn giản hơn và dễ học hơn. Có thể tưởng tượng điều này là đường biên quyết định được làm trơn bằng việc thêm các ảnh vào tập.

Có nhiều nguồn biến đổi trong bài toán dò tìm đối tượng, và cụ thể trong bài toán dò tìm khuôn mặt. Có các nguồn biến đổi sau.

- ✓ *Biến đổi trong mặt phẳng ảnh*: loại biến đổi ảnh khuôn mặt đơn giản nhất có thể được biểu diễn độc lập với khuôn mặt, bằng cách quay, dịch chuyển, biến đổi tỷ lệ và soi gương ảnh.

Khoa CNTT - ĐH KHTN TP. HCM

- ✓ *Biến đổi độ sáng và ngữ cảnh*: biến đổi do đối tượng và môi trường gây ra, cụ thể các thuộc tính bề mặt của đối tượng và các nguồn sáng. Các thay đổi về nguồn sáng nói riêng có thể biến đổi hoàn toàn về bề ngoài của khuôn mặt.
- ✓ *Biến đổi nền*: Trong luận văn của mình, Sung cho rằng với kỹ thuật nhận dạng mẫu hiện nay, tiếp cận dựa trên khung nhìn để dò tìm đối tượng chỉ thích hợp cho các đối tượng có “đường biên ảnh có thể dự đoán được”. Khi đối tượng có hình dáng dự đoán được, ta có thể trích ra window chỉ chứa các pixel bên trong đối tượng, và bỏ qua nền.
- ✓ *Biến đổi hình dáng*: với khuôn mặt, loại biến đổi này bao gồm biểu lộ tình cảm khuôn mặt, miệng và mắt mở hay đóng, và hình dáng khuôn mặt của từng người.

3.1.2 Tiếp cận theo khung nhìn kết hợp mạng nơron

Hệ thống dò tìm khuôn mặt thực hiện qua bốn bước chính:

1. Ước lượng vị trí: việc dùng tiếp cận máy học, cụ thể là mạng neural, đòi hỏi việc huấn luyện mẫu. Để giảm số lượng biến đổi trong ảnh huấn luyện dương, ảnh được canh biên với các ảnh khác để cực tiểu hoá các biến đổi vị trí đặc trưng khuôn mặt. Khi thi hành chương trình, ta không biết chính xác các vị trí đặc trưng khuôn mặt, do đó không thể dùng chúng để định vị các ứng viên khuôn mặt tiềm năng. Thay vậy, ta dò tìm toàn diện ở mọi vị trí và tỷ lệ để tìm mọi vị trí ứng viên. Các cải tiến dò tìm toàn diện làm cho thuật toán nhanh hơn, với tỷ lệ dò tìm giảm 10% đến 30%.

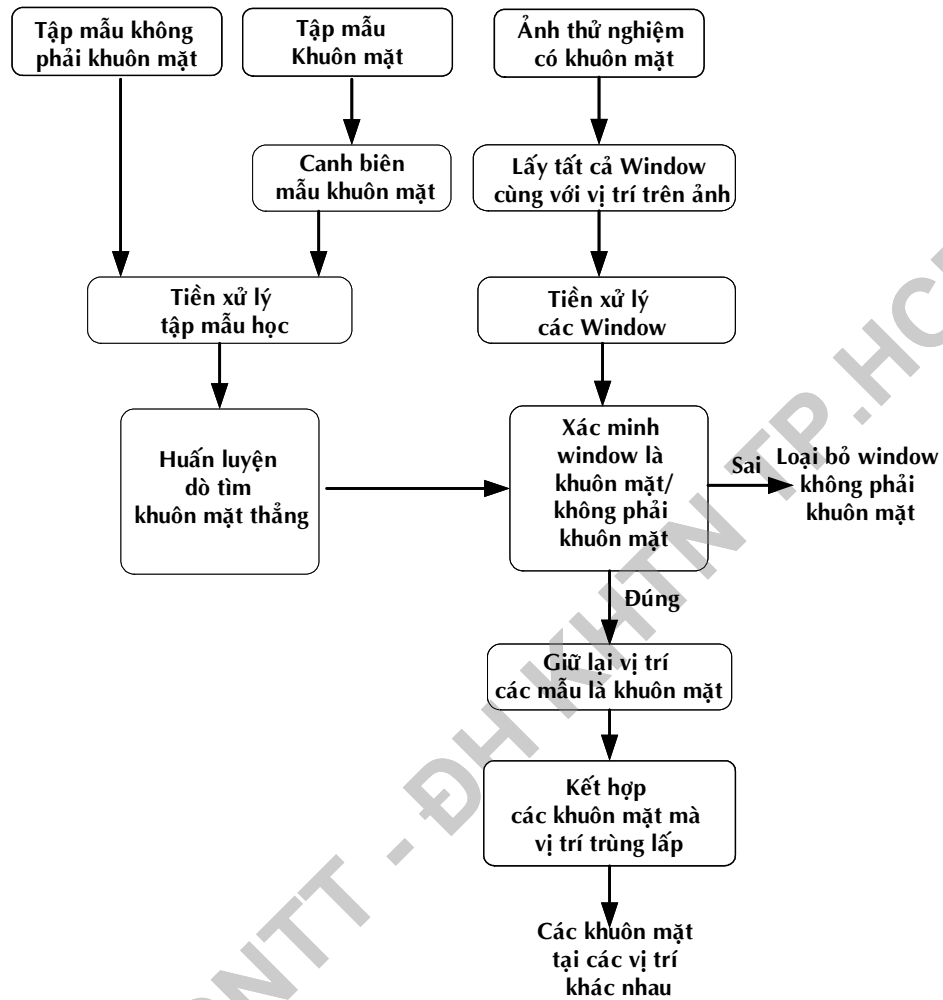
2. Tiền xử lý: để giảm các biến đổi gây ra do chiếu sáng hay camera, ảnh được tiền xử lý với các thuật toán chuẩn như cân bằng lược đồ để cải thiện độ sáng và độ tương phản trong ảnh.

3. Dò tìm: các khuôn mặt tiềm năng đã chuẩn hoá về vị trí, tư thế, và độ sáng trong hai bước đầu tiên được khảo sát để xác định chúng có thực sự là khuôn mặt hay không. Quyết định này được thực hiện bằng mạng neural đã huấn luyện với nhiều ảnh mẫu khuôn mặt và không khuôn mặt.

4. Quyết định: Kết hợp nhiều mạng để có được một quyết định khách quan nhất. Mỗi mạng học những điều khác nhau từ dữ liệu huấn luyện, và đưa ra các lỗi khác nhau. Các quyết định của chúng có thể kết hợp dùng một số heuristic đơn giản, làm tăng độ chính xác dò tìm khuôn mặt và ngăn chặn lỗi.

Khoa CNTT - ĐH KHTN TP.HCM

3.1.3 Dò tìm khuôn mặt bằng phương pháp mạng neural



Hình 3-1 Sơ đồ luồng xử lý các bước chính trong tiến trình dò tìm khuôn mặt

3.2 Chuẩn bị dữ liệu cho hệ thống dò tìm khuôn mặt

3.2.1 Giới thiệu

Dùng tiếp cận khung nhìn để dò tìm khuôn mặt, bộ dò tìm khuôn mặt theo khung nhìn phải xác định xem một cửa sổ con của một ảnh có thuộc về tập ảnh khuôn mặt hay không. Các biến đổi trong ảnh khuôn mặt có thể làm tăng độ phức tạp của đường biên quyết định để phân biệt khuôn mặt với không khuôn mặt, làm cho việc dò tìm khó khăn hơn. Phần này sẽ mô tả các kỹ thuật để làm giảm số biến đổi trong ảnh khuôn mặt.

3.2.2 Gán nhãn và canh biên các đặc trưng khuôn mặt

Bước đầu tiên trong việc giảm số các biến đổi trong ảnh khuôn mặt là canh biên các khuôn mặt này với khuôn mặt khác. Việc canh biên này sẽ làm giảm các biến đổi về vị trí, hướng, và tỷ lệ các khuôn mặt. Việc canh biên được tính trực tiếp từ các ảnh. Và nó tạo ra không gian ảnh khuôn mặt tối thiểu. Cường độ ảnh khuôn mặt có thể biến đổi nhiều, làm cho một số khuôn mặt khó canh biên với nhau.

Ta dùng giải pháp gán nhãn thủ công các mẫu khuôn mặt. Cụ thể là vị trí hai mắt, đỉnh mũi, hai góc và trung tâm miệng của mỗi khuôn mặt.

Bước tiếp theo là dùng thông tin này để canh biên các khuôn mặt với khuôn mặt khác. Trước hết định nghĩa canh biên giữa hai tập điểm đặc trưng. Đó là phép quay, biến đổi tỷ lệ, và dịch chuyển để làm cực tiểu hoá tổng bình phương khoảng cách giữa từng cặp đặc trưng tương ứng. Trong không gian hai chiều, một phép biến đổi tọa độ như vậy có thể được viết dưới dạng sau:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} s \cos \theta & -s \sin \theta \\ s \sin \theta & s \cos \theta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} = \begin{pmatrix} a & -b & t_x \\ b & a & t_y \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (3.1)$$

Nếu có nhiều tập tọa độ tương ứng, có thể viết như sau:

$$\begin{pmatrix} x_1 & -y_1 & 1 & 0 \\ y_1 & x_1 & 0 & 1 \\ x_2 & -y_2 & 1 & 0 \\ y_2 & x_2 & 0 & 1 \\ \vdots & & & \end{pmatrix} \cdot \begin{pmatrix} a \\ b \\ t_x \\ t_y \end{pmatrix} = \begin{pmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ \vdots \end{pmatrix} \quad (3.2)$$

Khi có hai hay nhiều hơn cặp điểm đặc trưng phân biệt, hệ các phương trình tuyến tính có thể được giải bằng phương pháp đảo ngược giả. Gọi ma trận

bên trái là A , vector $(a, b, t_x, t_y)^T$ là T , và bên phải là B , khi đó lời giải:

$$T = (A^T A)^{-1}(A^T B) \quad (3.3)$$

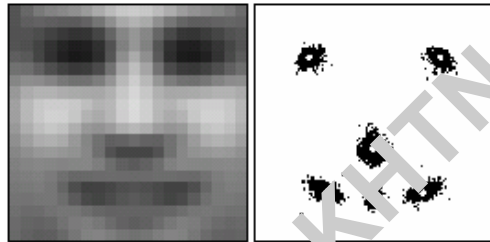
Lời giải đảo ngược giả đưa ra phép biến đổi T làm cực tiểu tổng bình phương khác biệt giữa tập tọa độ x'_i, y'_i và phiên bản đã biến đổi của x_i, y_i .

Canh biên tập các điểm đặc trưng.

1. Khởi tạo \bar{F} , vector sẽ là vị trí trung bình của mỗi đặc trưng gán nhãn trên mọi khuôn mặt, với một số vị trí đặc trưng ban đầu. Trong trường hợp canh biên các khuôn mặt thẳng, các đặc trưng này là vị trí mong muốn của hai mắt, đỉnh mũi, hai góc và trung tâm miệng của mỗi khuôn mặt trong cửa sổ đầu vào.
2. Với mỗi khuôn mặt i , dùng thủ tục canh biên để tính phép quay, dịch chuyển, và biến đổi tỷ lệ tốt nhất để canh biên các đặc trưng khuôn mặt F_i với các vị trí đặc trưng trung bình \bar{F} . Gọi vị trí đặc trưng đã canh biên F'_i .

3. Cập nhật \bar{F} bằng việc lấy trung bình các vị trí đặc trưng đã canh biên F'_i cho mỗi khuôn mặt i .
4. Tọa độ đặc trưng trong \bar{F} được quay, dịch chuyển và biến đổi để phù hợp với một số tọa độ chuẩn. Tọa độ chuẩn là tọa độ được dùng làm giá trị khởi tạo cho \bar{F} .
5. Sang bước 2.

Theo kinh nghiệm, thuật toán hội tụ trong vòng năm lần lặp, tạo cho mỗi khuôn mặt phép biến đổi để ánh xạ nó gần về vị trí chuẩn, và canh biên với mọi khuôn mặt khác. Khi đã biết các tham số để canh biên khuôn mặt, ảnh có thể được lấy mẫu lại dùng nội suy song tuyến tính. Khuôn mặt chuẩn và phân phối của các vị trí đặc trưng được cho trong Hình 3-2, và các mẫu ảnh đã được canh biên dùng kỹ thuật này được cho trong Hình 3-3.



Hình 3-2 Trái: Mẫu khuôn mặt chuẩn. Phải: Các vị trí đặc trưng khuôn mặt chuẩn (tròn trắng), và phân phối của các vị trí đặc trưng thực (sau khi canh biên) từ mọi mẫu (các điểm đen).



Hình 3-3 Ví dụ ảnh khuôn mặt thẳng được canh biên.

Trong việc huấn luyện bộ dò tìm, việc thu thập số mẫu đủ lớn là vấn đề quan trọng. Một kỹ thuật thường dùng là khung nhìn ảo, trong đó các ảnh mẫu mới được tạo ra từ ảnh thực (quay, dịch chuyển, biến đổi tỷ lệ ngẫu nhiên ảnh mẫu).

Khoa CNTT - ĐH KHTN TP.HCM

3.2.3 Tiền xử lý về độ sáng và độ tương phản trên tập mẫu học

Sau khi canh biên các khuôn mặt, vẫn còn một nguồn biến đổi chính (không kể biến đổi về bản chất giữa các khuôn mặt). Biến đổi này gây ra do độ sáng và các đặc tính máy ảnh, dẫn đến các ảnh có độ sáng tươi hay kém, hoặc ảnh có độ tương phản kém.

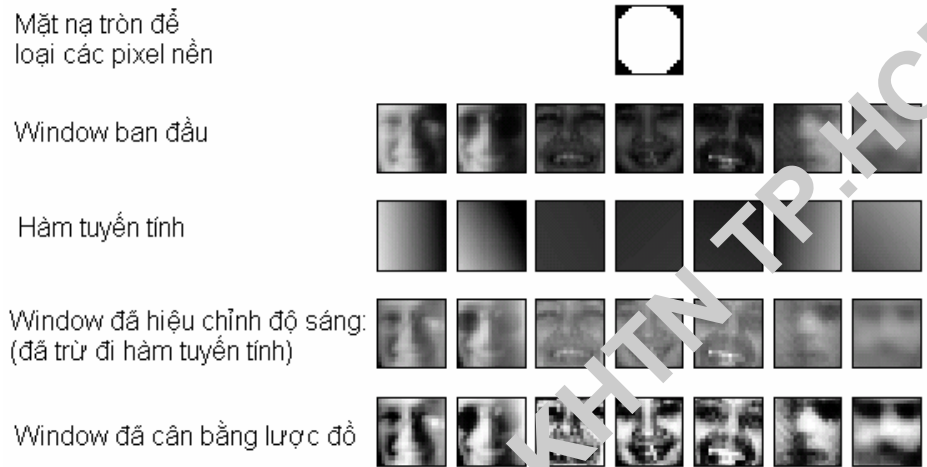
Ta xử lý vấn đề này bằng tiếp cận xử lý ảnh đơn giản. Kỹ thuật tiền xử lý trước hết cân bằng các giá trị mật độ trên toàn cửa sổ. Lập hàm biến đổi tuyến tính giá trị mật độ trong vùng tròn trong cửa sổ. Các điểm ảnh bên ngoài hình tròn có thể là nền. Nếu mật độ của pixel (x,y) là $I(x,y)$, khi đó cách biến đổi tuyến tính này được tham số hoá bởi a, b, c với:

$$(x \ y \ 1) \cdot \begin{pmatrix} a \\ b \\ c \end{pmatrix} = I(x,y)$$

(3.4)

Việc chọn cách biến đổi này là tùy ý. Nó có thể biểu diễn các khác biệt về độ sáng trên toàn ảnh. Các biến đổi được giới hạn là tuyến tính để số tham số ít và việc tạo lập hàm nhanh chóng. Tập hợp với mọi pixel trên toàn cửa sổ hình tròn ta được phương trình ma trận ràng buộc, và được giải bằng phương pháp đảo ngược giả. Phương trình tuyến tính này sẽ xấp xỉ toàn bộ độ sáng của mỗi phần của cửa sổ, và bị trừ đi với cửa sổ để cân bằng biến đổi về độ sáng.

Tiếp theo, cân bằng lược đồ, ánh xạ không tuyến tính các giá trị mật độ để mở rộng miền cường độ trong cửa sổ. Lược đồ được tính với các pixel trong vùng tròn trong cửa sổ. Việc này bù cho các khác biệt trong việc thu nhận đầu vào camera, và cũng cải thiện độ tương phản trong một số trường hợp. Các kết quả của mỗi bước được cho trong Hình 3-4.



Hình 3-4 Các bước trong việc tiền xử lý window. Đầu tiên, xây dựng hàm ánh xạ tuyến tính với các giá trị mật độ trong window, và sau đó trừ đi nó, để hiệu chỉnh về độ sáng. Tiếp theo, áp dụng cân bằng lược đồ, để hiệu chỉnh đầu vào camera khác nhau và cải thiện độ tương phản. Trong mỗi bước, việc ánh xạ được tính với các pixel bên trong hình tròn, và được áp dụng với toàn window.

3.3 Phương pháp dò tìm khuôn mặt thẳng

3.3.1 Giới thiệu

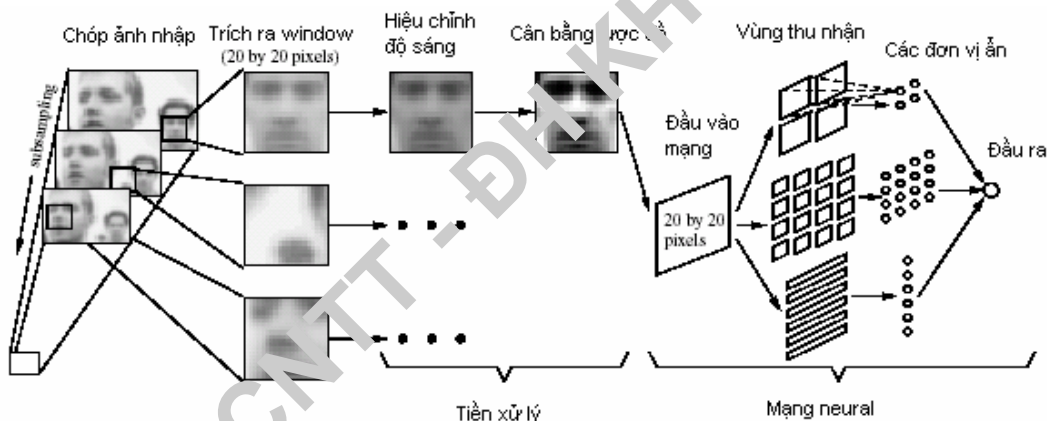
Chương này trình bày thuật toán dựa trên mạng neural để dò tìm ra các view (khung nhìn) của các khuôn mặt đứng, thẳng trong ảnh xám. Thuật toán thực hiện bằng cách áp dụng một hay nhiều mạng neural trực tiếp với các phần của ảnh đầu vào, và phân xử các kết quả của chúng. Mỗi mạng được huấn luyện để kết xuất một kết quả là có hay không có khuôn mặt.

Huấn luyện mạng neural để dò tìm khuôn mặt là một công việc đầy thách thức, vì khó khăn trong việc biểu thị các ảnh “không khuôn mặt”. Không như việc nhận dạng khuôn mặt, trong đó các lớp phân biệt là các khuôn mặt khác nhau. Hai lớp gọi là phân biệt trong dò tìm khuôn mặt là “ảnh có chứa khuôn mặt” và “ảnh không chứa khuôn mặt”. Dễ dàng lấy được mẫu ảnh chứa khuôn mặt điển hình, nhưng việc lấy mẫu ảnh không chứa khuôn mặt điển hình khó hơn rất nhiều. Ta tránh việc dùng tập huấn luyện có kích thước lớn để biểu diễn không khuôn mặt bằng việc chọn thêm ảnh vào tập huấn luyện khi tiến hành huấn luyện [Sung, 1996]. Phương pháp “bootstrap” nhằm giảm kích thước của tập huấn luyện cần thiết. Việc dùng cách thức xử lý giữa đa mạng và các heuristic để làm rõ ràng các kết quả và cải thiện đáng kể độ chính xác của bộ dò tìm.

3.3.2 Huấn luyện dò tìm khuôn mặt

Hệ thống hoạt động theo hai giai đoạn: trước hết áp dụng tập bộ dò tìm dựa trên mạng neural vào ảnh, và sau đó dùng bộ phân xử để kết hợp các đầu ra. Các bộ dò tìm riêng lẻ khảo sát mỗi vị trí trong ảnh ở một vài tỷ lệ, tìm vị trí có thể chứa khuôn mặt. Sau đó bộ phân xử hợp các dò tìm từ các mạng riêng lẻ và loại trừ các dò tìm chồng lấp.

Thành phần đầu tiên của hệ thống là mạng neural nhận đầu vào là vùng 20x20 (pixels) của ảnh và tạo đầu ra trong khoảng 1 đến -1, biểu thị có hay không có khuôn mặt. Để dò tìm mọi khuôn mặt trong ảnh, mạng được áp dụng ở mọi vị trí trong ảnh. Để dò tìm các khuôn mặt lớn hơn kích thước cửa sổ, ảnh đầu vào được giảm kích thước nhiều lần, và áp dụng bộ dò tìm ở mỗi kích thước. Mạng có một số bất biến với vị trí và kích thước. Số bất biến xác định số tỷ lệ và vị trí nó được dùng. Với bài này, ta áp dụng bộ lọc ở mọi vị trí điểm ảnh, và giảm tỷ lệ xuống 1.2 ở mỗi bước phân tích ảnh tứ phân. Chớp ảnh này được cho bên trái trong Hình 3-5.



Hình 3-5 Thuật toán dò tìm khuôn mặt

Sau khi cửa sổ 20x20 pixel được trích ra từ một vị trí và tỷ lệ nào đó trong ảnh nhập, nó được tiền xử lý dùng các bước hiệu chỉnh độ sáng và cân bằng lược đồ như đã trình bày trong Phần 2.3. Window sau khi tiền xử lý được truyền qua mạng neural. Mạng có các liên kết võng mạc đến các tầng nhập. Vùng thu nhận của các đơn vị ẩn được cho trong Hình 3-5. Window đầu vào được chia thành các mảnh nhỏ, 4 vùng 10x10 (pixels), 16 vùng 5x5 (pixels), và 6 vùng chồng lấp 20x5 (pixels). Mỗi vùng có liên kết đầy đủ với một đơn vị ẩn. Dù hình vẽ cho thấy một đơn vị ẩn cho mỗi vùng con đầu vào, nhưng các đơn vị này có thể được tái tạo. Với thử nghiệm sau, ta dùng mạng với hai và ba tập các đơn vị ẩn này. Hình dáng của các vùng con này được chọn để cho phép các đơn vị ẩn dò tìm các đặc trưng cho việc dò tìm khuôn mặt. Cụ thể, các sọc ngang cho phép các đơn vị ẩn dò tìm các đặc trưng như miệng, cặp mắt, trong khi các đơn vị ẩn với vùng tiếp thu hình vuông có thể dò tìm các đặc trưng như từng mắt, mũi, hai góc của miệng. Các thử nghiệm cho thấy rằng hình dạng chính xác của các vùng này không quan trọng, quan trọng là đầu vào được chia thành các vùng nhỏ thay vì dùng các kết nối hoàn toàn với toàn bộ đầu vào. Tương tự các mẫu liên kết đầu vào thường được dùng trong việc nhận dạng tiếng nói và ký tự [Waibel et al., 1989, Le Cun et al., 1989]. Mạng có một đầu ra giá trị thực, chỉ định window có chứa khuôn mặt hay không.

3.3.2.1 Ảnh huấn luyện khuôn mặt

Để dùng mạng neural phân loại cửa sổ là khuôn mặt hay không, ta cần các mẫu huấn luyện cho mỗi tập. Với các mẫu khuôn mặt ta dùng kỹ thuật đã trình bày trong phần 2.2 để canh biên các ảnh khuôn mặt trong đó một số điểm đặc trưng đã gán nhãn bằng tay. Sau khi canh biên, các khuôn mặt được co về về một kích thước, vị trí và hướng đồng nhất trong cửa sổ 20×20 pixel. Ảnh được co về với một lượng ngẫu nhiên từ $1/\sqrt{1.2}$ đến $\sqrt{1.2}$. Điều này cho phép bộ dò tìm được áp dụng ở mỗi vị trí pixel và ở mỗi tỷ lệ trong chóp ảnh, và vẫn dò tìm các khuôn mặt ở vị trí và tỷ lệ trung bình. Ngoài ra, để cho bộ dò tìm mạnh hơn với các biến đổi không đáng kể trong khuôn mặt, chúng được quay với một lượng ngẫu nhiên (tối đa 10°).

3.3.2.2 Ảnh huấn luyện không phải khuôn mặt

Ta cần nhiều ảnh không khuôn mặt để huấn luyện bộ dò tìm khuôn mặt, vì sự đa dạng của ảnh không khuôn mặt lớn hơn nhiều so với ảnh khuôn mặt. Một lớp ảnh không chứa khuôn mặt là các ảnh phong cảnh chẳng hạn cây, núi, và toà nhà.

Thu thập tập không khuôn mặt “đặc trưng” là việc khó. Hầu như bất kỳ ảnh nào cũng có thể được xem như là mẫu không khuôn mặt; không gian ảnh không khuôn mặt lớn hơn không gian ảnh khuôn mặt. Tiếp cận thông kê máy học cho rằng ta nên huấn luyện mạng neural trên cùng phân bố ảnh mà mạng thấy khi chạy. Với bộ dò tìm khuôn mặt, số mẫu khuôn mặt là 15,000, là một số thích hợp. Tuy nhiên, tập đại diện ảnh phong cảnh chứa gần 150,000,000 cửa sổ, và việc huấn luyện trên một cơ sở dữ liệu khuôn mặt có kích thước lớn như vậy là rất khó. Phần tiếp theo mô tả việc huấn luyện trên một cơ sở dữ liệu khuôn mặt này.

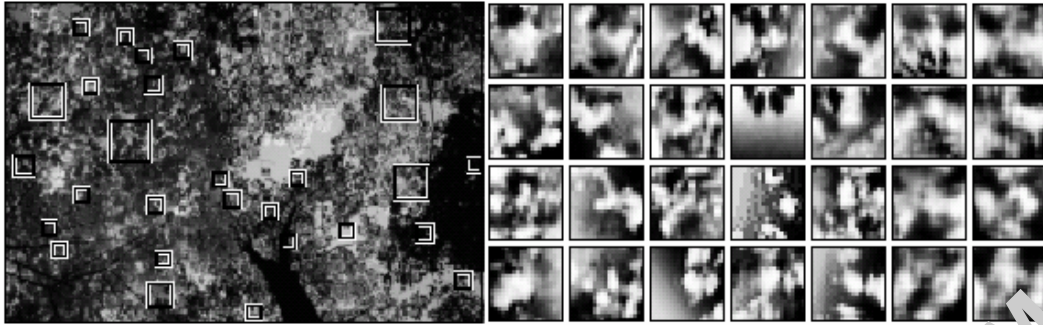
3.3.2.3 Phương pháp huấn luyện chủ động

Do khó khăn của việc huấn luyện với mọi mẫu âm có thể, ta dùng thuật toán [Sung, 1996]. Thay vì thu thập tập các ảnh trước khi việc huấn luyện bắt đầu, ảnh được thu thập trong quá trình huấn luyện, theo cách sau:

1. Tạo tập khởi tạo các ảnh không khuôn mặt bằng cách tạo 1000 ảnh ngẫu nhiên. Áp dụng các bước tiền xử lý cho mỗi ảnh này.
2. Huấn luyện mạng neural nhân tạo để cho ra 1 với các mẫu khuôn mặt, và -1 với các mẫu không khuôn mặt. Trong lần lặp đầu tiên của vòng lặp, các trọng số mạng được khởi tạo ngẫu nhiên. Sau lần lặp đầu tiên này, ta dùng các trọng số được tính qua việc huấn luyện trong lần lặp trước.
3. Chạy hệ thống trên ảnh phong cảnh *không chứa khuôn mặt*. Thu thập các ảnh con trong đó mạng nhận làm là khuôn mặt (hoạt hoá đầu ra >0).
4. Chọn ngẫu nhiên 250 ảnh con này, áp dụng các bước tiền xử lý, và sau đó thêm chúng vào tập mẫu âm. Sang Bước 2.

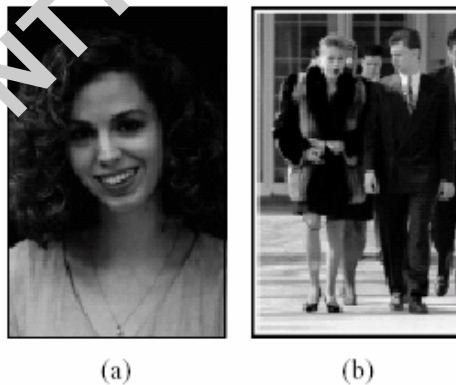
Thuật toán huấn luyện dùng trong Bước là thuật toán hồi quy lỗi chuẩn [Hertz et al., 1991]. Các nơon dùng hàm kích hoạt dạng tanh, cho đầu ra từ -1 đến 1, do đó ngưỡng 0 với dò tìm là khuôn mặt. Vì ta không huấn luyện với mọi mẫu âm, các đối số xác suất của phân trước không áp dụng cho việc thiết lập ngưỡng dò tìm.

Vì số mẫu âm lớn hơn nhiều so với số mẫu dương, các bó mẫu huấn luyện chỉ chứa các mẫu âm, sẽ không thích hợp cho việc huấn luyện mạng neural. Thay vì mỗi bó gồm 100 mẫu dương và âm lấy ngẫu nhiên từ toàn bộ tập huấn luyện, và truyền qua thuật toán hồi quy ngược. Ta chọn các bó huấn luyện có 50% mẫu âm và 50% mẫu dương. Điều này đảm bảo rằng ban đầu, khi tập mẫu dương nhiều hơn tập mẫu âm, mạng sẽ học từ cả hai tập.

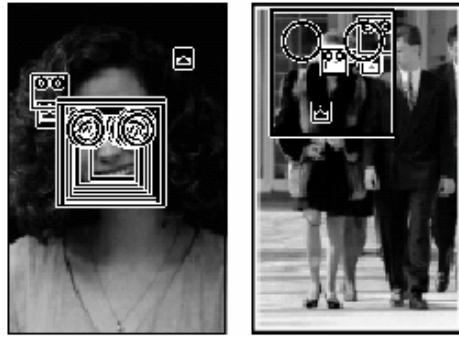


Hình 3-6 Trong khi huấn luyện, hệ thống đã huấn luyện một phần được áp dụng với các ảnh phong cảnh không chứa khuôn mặt (như bên trái). Bất kỳ vùng nào trong ảnh được dò là khuôn mặt là lỗi, và được thêm vào tập mẫu huấn luyện âm.

Một số mẫu không phải khuôn mặt được thu thập trong quá trình huấn luyện được cho trong Hình 3-6. Chú ý rằng một số mẫu tương tự khuôn mặt, dù chúng không gần các mẫu dương trong Hình 3-3. Sự xuất hiện của các mẫu này làm cho mạng neural học ranh giới chính xác giữa các ảnh khuôn mặt và không phải khuôn mặt. Dùng 120 ảnh phong cảnh để thu thập các mẫu âm theo cách bootstrap. Lần huấn luyện điển hình chọn khoảng 8000 ảnh không khuôn mặt từ 146,212,178 ảnh con có sẵn tại mọi vị trí và tỷ lệ trong huấn luyện ảnh phong cảnh.



Hình 3-7 Ảnh mẫu để thử nghiệm đầu ra của bộ dò tìm thẳng.



(a)

(b)

Hình 3-8 Đầu ra của mạng dò tìm

Khoa CNTT - ĐH KHTN TP.HCM

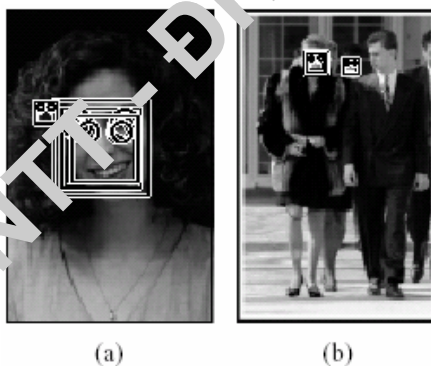
3.3.3 Phương pháp cải tiến chất lượng dò tìm khuôn mặt

(Theo Henry A. Royley, May 1999, CMU-CS-99-117)

Hình 3-8 cho thấy rằng đầu ra từ một mạng đơn vẫn còn nhiều dò tìm lỗi. Trong phần này, ta đưa ra hai chiến lược để cải thiện độ tin cậy: hợp nhất các dò tìm chồng lấp từ một mạng đơn và phân xử giữa đa mạng.

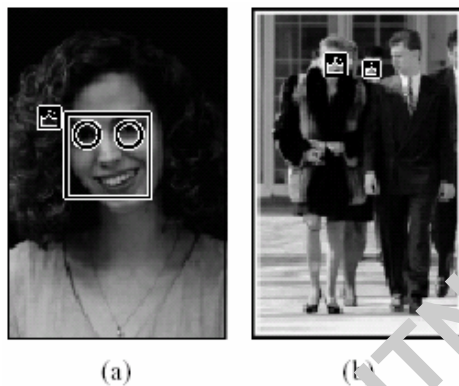
3.3.3.1 Các Heuristic loại bỏ thông tin thừa

Trong Hình 3-8, khuôn mặt được dò tìm với nhiều vị trí và tỷ lệ gần nhau, trong khi đó các dò tìm lỗi thường xuất hiện với ít vị trí và tỷ lệ hơn. Quan sát này dẫn tới heuristic loại trừ nhiều dò tìm lỗi. Với mỗi dò tìm, số các dò tìm khác trong lân cận của dò tìm đó có thể tính được. Nếu số đó lớn hơn một ngưỡng, vị trí đó được phân loại là một khuôn mặt. Trung tâm của các dò tìm gần nhau xác định vị trí dò tìm kết quả, do đó che lấp các dò tìm. Heuristic này được gọi là phân ngưỡng theo kích thước và cấp độ co ảnh trong đó *kích thước* là kích thước lân cận, tính theo số điểm ảnh và các bước biến đổi tỷ lệ theo dạng tứ phân, và *cấp độ chính* là tổng số dò tìm phải xuất hiện trong lân cận đó. Kết quả của việc áp dụng $\text{threshold}(4,2)$ với các ảnh trong Hình 3-8 được cho trong Hình 3-9.

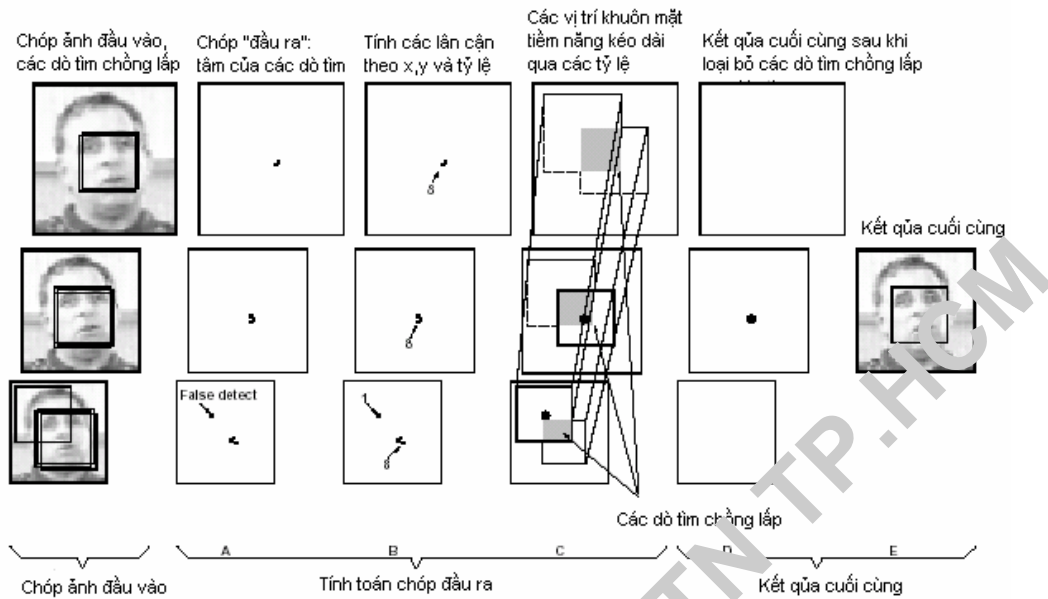


Hình 3-9 Kết quả áp dụng $\text{threshold}(4,2)$ với các ảnh trong Hình 3-8.

Nếu một vị trí được xác định chính xác là khuôn mặt, khi đó mọi vị trí khác chồng lấp nó có vẻ như là lỗi, và do đó có thể được loại trừ. Theo heuristic trên, ta giữ lại vị trí với nhiều dò tìm trong một lân cận nhỏ, và loại trừ các vị trí với số dò tìm ít hơn. Heuristic này được gọi là “chồng lấp”. Một số trường hợp heuristic này không đúng, như trong Hình 3-8b, trong đó một khuôn mặt che một phần khuôn mặt khác, và do đó bị mất khi áp dụng heuristic này.



Hình 3-10 Kết quả áp dụng trùng lấp với các ảnh của Hình 9.

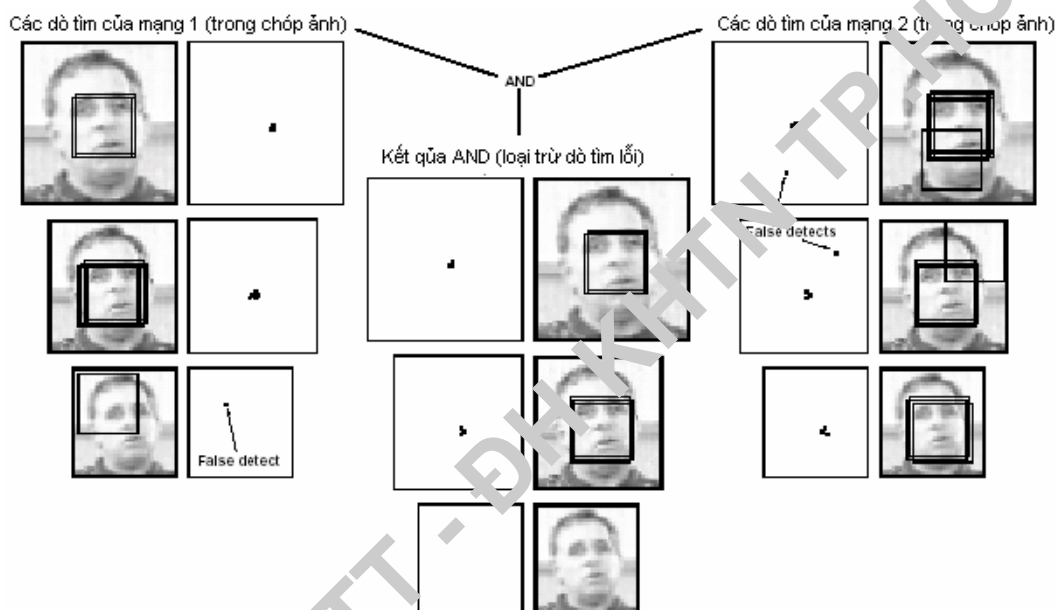


Hình 3-11 Cơ cấu trộn nhiều dò tìm từ một mạng đơn: A) Các dò tìm được ghi trong chụp “đầu ra”. B) tính số dò tìm trong lân cận của mỗi dò tìm. C) Bước cuối cùng là kiểm tra các vị trí khuôn mặt đã đưa ra về tính chồng lấp, và D) loại bỏ các dò tìm chồng lấp nếu tồn tại.

Việc thi hành hai heuristic này được minh họa trong Hình 3-11. Mỗi dò tìm ở một vị trí và tỷ lệ được đánh dấu trong pyramid (chớp) ảnh, gọi là chớp “đầu ra”. Sau đó mỗi dò tìm được thay bằng số dò tìm trong lân cận của vị trí đó. Áp dụng ngưỡng với các giá trị này, và tính trọng tâm (về cả vị trí và tỷ lệ) của các dò tìm trên (bước này được bỏ qua trong Hình 3-11). Khảo sát mỗi trọng tâm, bắt đầu từ trọng tâm có số dò tìm lớn nhất trong lân cận. Nếu bất kỳ vị trí trọng tâm khác biểu diễn khuôn mặt chồng lấp với trọng tâm hiện hành, được loại bỏ khỏi chớp đầu ra. Mọi vị trí trọng tâm còn lại tạo thành kết quả dò tìm cuối cùng.

3.3.3.2 Hệ thống Mạng Kết Hợp

Để giảm hơn nữa số lỗi, ta có thể áp dụng nhiều mạng, và phân xử đầu ra của chúng để tạo quyết định cuối cùng. Mỗi mạng được huấn luyện theo cách tương tự, nhưng với các trọng số ban đầu ngẫu nhiên, các ảnh không khuôn mặt ban đầu ngẫu nhiên, và hoán vị thứ tự trình bày của các ảnh phong cảnh. Việc dò tìm và tỷ lệ lỗi của các mạng riêng bị hạn chế. Tuy nhiên, vì các điều kiện huấn luyện khác nhau, và vì việc tự lựa chọn các mẫu huấn luyện âm, các mạng có các bias (khuyh hướng) khác nhau và tạo các lỗi khác nhau.



Hình 3-12 AND các đầu ra từ hai mạng trên các vị trí và tỷ lệ khác nhau có thể cải thiện độ chính xác dò tìm.

Thuật toán phân xử được minh họa trong Hình 3-12. Mỗi dò tìm ở một vị trí và tỷ lệ được ghi lại trong chóp ảnh, tương tự heuristic trước. Một cách để kết hợp hai chóp đó là AND chúng. Chiến lược này cho một dò tìm chỉ nếu cả hai mạng dò tìm một khuôn mặt ở chính xác cùng vị trí và tỷ lệ. Do khuyh hướng khác nhau của mỗi mạng chúng hiếm khi đồng ý với một dò tìm lỗi. Điều này cho phép AND để loại trừ hầu hết các dò tìm lỗi. Heuristic này có thể giảm tỷ lệ dò tìm vì một khuôn mặt được dò tìm bởi một mạng sẽ bị bỏ qua. Tuy nhiên, sau đây ta sẽ thấy các mạng riêng có thể dò tìm cùng tập khuôn mặt, để số khuôn mặt mất

do AND là nhỏ.

Các heuristic tương tự, chẳng hạn OR các đầu ra của hai mạng, hay bỏ phiếu giữa ba mạng, cũng đã được thử. Trong thực hành, các heuristic phân xử này có thể được dùng với các thuật toán lấy ngưỡng khác nhau đã mô tả trên. Chẳng hạn, AND có thể được thực hiện bằng việc kết hợp kết quả của hai mạng, và áp dụng $\text{threshold}(0,2)$, OR với $\text{threshold}(0,1)$, và bỏ phiếu bằng việc áp dụng $\text{threshold}(0,2)$ với các kết quả của ba mạng.

Các chiến lược phân xử như AND, OR, hay bỏ phiếu có vẻ như hợp lý, nhưng có thể một số heuristic kém hiển nhiên hơn có thể thực hiện tốt hơn. Để thử nghiệm thuyết này, ta dùng một mạng khác để phân xử đa mạng dò tìm. Với mọi vị trí, mạng phân xử khảo sát một lân cận nhỏ xung quanh vị trí đó trong chóp đầu ra của mỗi mạng đơn. Với mỗi chóp, ta tính các dò tìm trong vùng 3×3 pixel với ba tỷ lệ quanh vị trí đó, thu được ba số với mỗi bộ dò tìm, và đưa vào mạng phân xử. Mạng phân xử được huấn luyện để tạo ra một đầu ra dương cho tập các đầu vào cho trước chỉ nếu vị trí đó có chứa một khuôn mặt, và tạo ra một đầu ra âm cho các vị trí không chứa khuôn mặt. Như sẽ thấy trong phần kế, dùng mạng phân xử theo cách này tạo ra các kết quả có thể so sánh được với (và tốt hơn trong một số trường hợp) các kết quả tạo ra bởi các heuristic đã trình bày trước đây.

Chương 4 RÚT TRÍCH ĐẶC TRƯNG TỪ
KHUÔN MẶT

Khoa CNTT - ĐH KHTN TP.HCM

4.1 Tiếp cận theo phương pháp phân tích thành phần chính (Principal Component Analysis hay PCA)

4.1.1 Vector riêng, Trị riêng và sự chéo hoá của ma trận

Xét một toán tử tuyến tính f trong không gian R^n với các vector cơ sở :

$$e_i = [0 \dots 1 \dots 0]^T \text{ (với giá trị 1 nằm tại vị trí thứ } i) \quad (4.1.1)$$

Toán tử tuyến tính này sẽ được biểu diễn bởi một ma trận vuông T kích thước $n \times n$.

Một đại lượng vô hướng λ được gọi là trị riêng của toán tử f , hay của ma trận T , nếu tìm được một vector x , $x \neq 0$, sao cho

$$f(x) = \lambda x \quad (4.1.2)$$

$$\text{hay } T \cdot x = \lambda x. \quad (4.1.3)$$

Vector x khi đó được gọi là vector riêng của f , hay T , ứng với trị riêng λ . Ma trận T với kích thước $n \times n$ trên đây sẽ có tối đa n trị riêng và n vector riêng tương ứng. Một ma trận T khả nghịch đảo sẽ có đủ n trị riêng (kể cả trị riêng bội) và n vector riêng tương ứng.

4.1.2 Kì vọng và phương sai trong thống kê đa chiều

Ma trận \mathbf{T} (biểu diễn trong không gian \mathbf{R}^n với các vector cơ sở \mathbf{e}_i nêu trên) được gọi là chéo hóa được nếu tồn tại một cơ sở trong không gian \mathbf{R}^n sao cho ma trận \mathbf{T} biểu diễn trong cơ sở đó có dạng chéo (các phần tử ngoài đường chéo bằng $\mathbf{0}$).

Ví dụ: Khảo sát trên không gian \mathbf{R}^5 với ma trận chéo 5×5

$$\mathbf{T} = \begin{pmatrix} \lambda_1 & 0 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 & 0 \\ 0 & 0 & 0 & \lambda_4 & 0 \\ 0 & 0 & 0 & 0 & \lambda_5 \end{pmatrix}$$

Giả sử \mathbf{C} là ma trận các vector cơ sở mới được biểu diễn trong cơ sở $\{\mathbf{e}_i\}$. Ở đây, ma trận \mathbf{T} được chuyển từ cơ sở $\{\mathbf{e}_i\}$ sang cơ sở mới nên ma trận chuyển đổi cơ sở từ $\{\mathbf{e}_i\}$ sang \mathbf{C} cũng là \mathbf{C} . Nếu \mathbf{T} chéo hóa được tức là tồn tại ma trận \mathbf{C} khả nghịch (tức là \mathbf{C} tạo được một cơ sở trong \mathbf{R}^n) sao cho :

$$\mathbf{T}_c = \mathbf{C}^{-1} \mathbf{T} \mathbf{C} \quad (4.1.4)$$

có dạng chéo.

Nếu ta có \mathbf{C} là một ma trận có các cột là các vector cơ sở đã được chuẩn hóa của không gian \mathbf{R}^n thì $\mathbf{C}^T = \mathbf{C}^{-1}$, khi đó ta có thể viết :

$$\mathbf{T}_c = \mathbf{C}^T \mathbf{T} \mathbf{C}. \quad (4.1.5)$$

Ta có thể tìm được ma trận \mathbf{C} để chéo hóa một ma trận \mathbf{T} bằng cách tìm các vector riêng của ma trận \mathbf{T} . Ma trận \mathbf{C} là ma trận có các cột là các vector riêng của \mathbf{T} .

+ Kỳ vọng

Đối với thống kê nhiều chiều, mỗi một mẫu thống kê là một vector nhiều chiều. Giả sử ta có một biến ngẫu nhiên \mathbf{X} trong không gian tuyến tính n chiều.

$$\mathbf{X} = [x_1 \ x_2 \ \dots \ x_n]^T \quad (4.1.6)$$

Khi đó kỳ vọng của biến ngẫu nhiên \mathbf{X} cũng là một vector n chiều, trong thống kê, kỳ vọng $E[\mathbf{X}]$ của một biến ngẫu nhiên \mathbf{X} có thể được ước lượng bằng trung bình mẫu \bar{X} :

$$\bar{X} = \frac{1}{M} \sum_{i=1}^M X_i. \quad (4.1.7)$$

Trong đó M là tổng số mẫu có trong thống kê.

+ Ma trận hiệp phương sai

Giá trị phương sai trong thống kê một chiều là độ đo mức độ phân tán của biến ngẫu nhiên xung quanh kỳ vọng. Trong thống kê nhiều chiều, khái niệm này được mở rộng thành ma trận hiệp phương sai:

$$C = E[(X - E[X])(X - E[X])^T] \quad (4.1.8)$$

Ma trận hiệp phương sai là một ma trận đối xứng. Mỗi phần tử c_{ij} của ma trận là hiệp phương sai giữa hai thành phần x_i và x_j trong vector \mathbf{X} .

Nếu $c_{ij} = 0$ ta nói hai thành phần x_i và x_j là độc lập hay không phụ thuộc lẫn nhau. Nếu $c_{ij} \neq 0$, ta nói x_i và x_j không độc lập hay giữa chúng có mối tương quan với nhau.

Trong thống kê, ma trận hiệp phương sai được tính như sau:

$$C = \frac{1}{M-1} \sum_{i=1}^M (X_i - \bar{X})(X_i - \bar{X})^T \quad (4.1.9)$$

4.1.3 Kỹ thuật rút trích đặc trưng bằng phương pháp phân tích thành phần chính

Ý tưởng chính phân tích thành phần chính: Mục tiêu của phương pháp thành phần chính là giảm số chiều của một tập các vector sao cho vẫn đảm bảo được tối đa thông tin quan trọng nhất của tập học. Có thể nói **phân tích thành phần chính** tìm cách giữ lại những thành phần thống kê quan trọng nhất của tập mẫu.

Giả sử ta cần giảm số chiều của mẫu học từ n chiều xuống còn m chiều nghĩa là ta cần tìm một ánh xạ từ không gian n chiều xuống không gian nhỏ hơn

chỉ có m chiều ($m < n$). Gọi \mathbf{x} là một vector trong không gian n chiều, \mathbf{y} là một vector trong không gian m chiều. Ta có trung bình bình phương lỗi **MSE** (*mean square error*) khi loại bỏ một số thành phần trong \mathbf{x} để thu được \mathbf{Y} bằng tổng phương sai của những thành phần bị loại bỏ. Phương pháp phân tích thành phần chính sẽ tìm một phép biến đổi tuyến tính \mathbf{T} :

$$\mathbf{y} = \mathbf{T}^* \mathbf{x} \quad \mathbf{T} \text{ là ma trận } \mathbf{m} \times \mathbf{n} \quad (4.1.10)$$

sao cho trung bình bình phương lỗi là bé nhất.

Gọi \mathbf{M} là vector trung bình của các vector \mathbf{x} trong tập học \mathbf{X} .

$$\mathbf{M} = \frac{1}{M} \sum_{k=1}^M x_k, \quad M \text{ là số phần tử trong tập học.} \quad (4.1.11)$$

Gọi \mathbf{C} là ma trận hiệp phương sai của các các phần tử trong tập \mathbf{X} .

$$\mathbf{C} = \frac{1}{M-1} \sum_{k=1}^M (x_k - M)(x_k - M)^T, \quad \mathbf{C} \text{ là ma trận đối xứng } \mathbf{n} \times \mathbf{n} \quad (4.1.12)$$

Người ta chứng minh được rằng nếu \mathbf{T} là một ma trận mà mỗi hàng là một vector riêng của \mathbf{C} và m vector riêng này ứng với m trị riêng lớn nhất thì \mathbf{T} chính là phép biến đổi tuyến tính thỏa mãn điều kiện **MSE** nhỏ nhất.

Gọi Φ là ma trận vuông $n \times n$ mà mỗi cột là một vector riêng của \mathbf{C} đã được chuẩn hóa với phép biến đổi :

$$\mathbf{y} = \Phi^T \mathbf{x} \quad \mathbf{y} = (y^1, y^2, \dots, y^n) \quad (4.1.13)$$

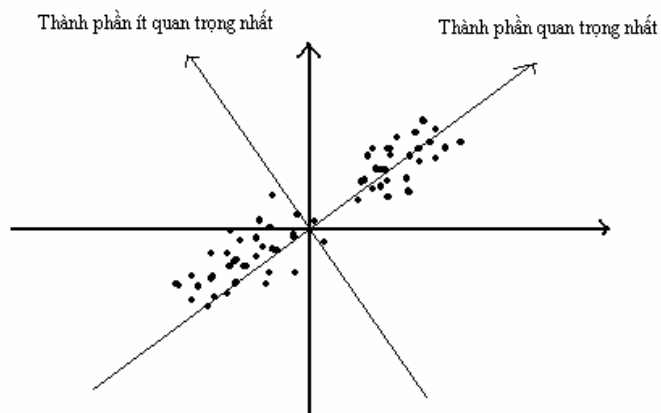
được gọi là phép biến đổi **Hotelling**.

Xét theo quan điểm của nhận dạng thì mỗi thành phần y^i của vector \mathbf{y} được xem như là một đặc trưng của vector mẫu \mathbf{x} . Các đặc trưng này là các đặc trưng độc lập với nhau vì ma trận hiệp phương sai của \mathbf{y} là

$$\mathbf{C}_y = \Phi^T \mathbf{C} \Phi \quad (4.1.14)$$

là một ma trận chéo (đã đề cập tới trong phần : Vector riêng, trị riêng và sự chéo hóa ma trận).

Tóm lại, phương pháp phân tích thành phần chính ánh xạ một vector từ không gian n chiều xuống không gian m chiều sẽ đi tìm các trị riêng và vector riêng của ma trận hiệp phương sai \mathbf{C} của tập \mathbf{X} và giữ lại m vector riêng ứng với m trị riêng lớn nhất làm cơ sở cho không gian m chiều này.



Hình 4-1 Hai trục tương ứng với hai thành phần quan trọng nhất và ít quan trọng nhất đối với tập mẫu có hai cluster như trên.

Khoa CNTT - ĐH KHTN TP.HCM

❖ **Lưu ý :**

Khi số lượng mẫu M trong tập X nhỏ hơn số chiều n , thay vì tính trực tiếp các vector riêng từ ma trận hiệp phương sai C , ta có thể tính các vector riêng theo phương pháp sau :

✓ **Bước 1 :** Tính ma trận kích thước $M \times M$, C' như sau :

$$C' = Y^T \cdot Y$$

với $Y_{n \times M} = [x_1, x_2, \dots, x_M]$ mỗi cột của ma trận là một phần tử x_i , $i=1..m$

✓ **Bước 2 :**

Tính M vector riêng EM_i và các trị riêng tương ứng của ma trận C' .
Chọn m vector riêng ứng với m trị riêng lớn nhất để tiếp tục bước 3.

✓ **Bước 3 :**

Chiếu các vector riêng M chiều này về lại không gian n chiều của các mẫu x_i bằng cách như sau :

$$En_i = \sum_{k=1}^M EM_k x_k \quad (4.1.15)$$

Các vector En_i thu được chính là các vector riêng cần tìm của ma trận C .

❖ **Cách xác định số thành phần chính hiệu quả nhất**

- ✓ Có hai phương pháp hữu ích giúp cho chúng ta lấy số lượng thành phần chính sao cho hiệu quả . Tất cả hai phương pháp này đều dựa trên mối quan hệ giữa các giá trị đặc trưng.
- ✓ Sắp xếp lại các giá trị đặc trưng tìm được theo thứ tự giảm dần về mặt giá trị $(1, \text{eigenvalue}[1]), (2, \text{eigenvalue}[2]), \dots, (p, \text{eigenvalue}[p])$ và Thứ tự này vẫn đảm bảo được thứ tự của các vector đặc trưng tương ứng.

- ✓ Theo dõi sự biến thiên của chuỗi giá trị đặc trưng vừa được xếp lại, khi sự biến thiên này tiến đến một điểm ngưỡng (thông thường xấp xỉ bằng không). Thông thường đó chính là lúc mà ta chọn được đủ số lượng thành phần chính cần thiết .
- ✓ Với phương châm làm sao số lượng thành phần chính là thấp nhất đủ để giải thích khả năng phân tán tập mẫu học thành các lớp mẫu riêng cần thiết nhất.
- ❖ **Cách để nhận được các thành phần chính**
 - ✓ Các thành phần chính có thể nhận được bằng cách chiếu các vector dữ liệu có nhiều biến động vào không gian mở rộng từ các vector đặc trưng.
- ❖ **Các đánh giá quan trọng về rút trích đặc trưng bằng phương pháp PCA**
 - ✓ Khi lấy số đặc trưng càng về sau thì khả năng biến động cần thấp , có nghĩa nối quan hệ giữa các phân tử càng cao , thì sự giao nhau giữa các lớp mẫu trong tập mẫu càng lớn
 - ✓ Nhưng nếu lấy không đủ số lượng thành phần chính , thì khả năng phân tán của tập mẫu càng cao (Có thể tăng vượt ngoài số lớp mẫu cần thiết trong tập mẫu)

4.2 Tiếp cận theo phương pháp Biến đổi Cosine rời rạc

4.2.1 Ý nghĩa phép biến đổi DCT

Phép biến đổi Cosine rời rạc là một kỹ thuật biến đổi nhanh, và là một trong những phép biến đổi hữu ích nhất trong lĩnh vực xử lý tín hiệu số nói chung và lĩnh vực xử lý ảnh, video nói riêng. Mục đích của mã hóa Cosine rời rạc xử lý tín hiệu trên miền không gian pixel sang một tín hiệu mới trên miền tần số, nhằm giảm khối lượng dữ liệu của các tín hiệu, đồng thời vẫn bảo toàn tốt chất lượng của tín hiệu.

4.2.2 Các khái niệm quan trọng

➤ Định nghĩa 1

Phép biến đổi Cosine rời rạc hai chiều trên một ma trận $C = \{c(k, n)\}$ kích thước $N \times N$, cũng gọi là một phép biến đổi cosine rời rạc, được định nghĩa như sau

$$c(k, n) = \begin{cases} \frac{1}{\sqrt{N}} & k = 0, 0 \leq n \leq N-1 \\ \sqrt{\frac{2}{N}} \cos \frac{\pi(2n+1)k}{2N} & 1 \leq k \leq N-1, 0 \leq n \leq N-1 \end{cases} \quad (4.2.1)$$

➤ Định nghĩa 2

Phép biến đổi Cosine rời rạc một chiều trên một dãy số $\{u(n), 0 \leq n \leq N-1\}$ được định nghĩa như sau

$$v(k) = \alpha(k) \sum_{n=0}^{N-1} u(n) \cos \left[\frac{\pi(2n+1)k}{2N} \right] \quad 0 \leq k \leq N-1 \quad (4.2.2)$$

Trong đó:

$$\alpha(0) = \sqrt{\frac{1}{N}}, \quad \alpha(k) = \sqrt{\frac{1}{N}} \quad \text{với } 1 \leq k \leq N-1 \quad (4.2.3)$$

➤ **Định nghĩa 3**

Phép biến đổi nghịch Cosine rời rạc được định nghĩa như sau

$$u(n) = \sum_{k=0}^{N-1} \alpha(k)v(k) \cos\left[\frac{\pi(2n+1)k}{2N}\right] \quad 0 \leq n \leq N-1 \quad (4.2.4)$$

➤ **Định nghĩa 4**

Các vector cơ sở của phép biến đổi Cosine rời rạc trên khối 8×8 . Có thể có nhiều trường hợp các hệ số của phép biến đổi này rất nhỏ, giải thích nguyên nhân này là hầu hết năng lượng của dữ liệu được dồn về một vài hệ số đặc biệt nào đó và ở một vài trị trí đặc biệt nào đó trên miền tần số.

➤ **Định nghĩa 5**

Tín hiệu thực: một tín hiệu là một tín hiệu thực thì giá trị phần thực cũng chính là giá trị của tín hiệu gốc, còn phần ảo thì bằng không.

Khoa CNTT - ĐH KHTN TP HCM

4.2.3 Kỹ thuật mã hoá hệ số DCT

Sơ đồ phép biến đổi Cosine rời rạc xử lý giá trị các khối dữ liệu pixel thành các khối hệ số trong miền tần số. Nhằm tăng tốc độ xử lý của thuật toán ta thường chọn khối dữ liệu $C_{8 \times 8}$ hay $C_{16 \times 16}$, nhưng tiêu chuẩn chọn khối phổ biến nhất vẫn là 8×8 , có thể giải thích vấn đề này trên phương diện khả năng xử lý của phần cứng: Bởi vì, khối 8×8 trùng khớp với kích thước dữ liệu cực đại mà công nghệ vi mạch điện tử hiện thời có thể xử lý tại một thời điểm.

Khi sử dụng phép biến đổi Cosine rời rạc trên mỗi khối dữ liệu thô $C_{8 \times 8}$ và kết quả của phép biến đổi này cũng là một ma trận $c_{8 \times 8}$ phổ năng lượng trên miền tần số. Ta có thể nói rằng 64 giá trị số thay đổi thành 64 giá trị số khác. Tất cả các giá trị này đều thuộc về miền số thực $c_{u \times v} \in R : 0 \leq u, v \leq 7$. Theo như tính chất của Cosine rời rạc đã được trình bày ở phần trên ta đã loại bỏ được sự phức tạp của toán học trong xử lý của nó và cũng chính vì vậy Cosine rời rạc sẽ đơn giản hơn nhiều so với phép biến đổi nhanh Fourier(FFT). Nhưng phép phân tích Cosine rời rạc lại tương tự như phép biến đổi nhanh Fourier ở chỗ giá trị của mỗi hệ số trên ma trận DCT trong miền năng lượng quang phổ chính là biên độ của hàm cơ sở tương ứng với hệ số đó. Hình 4-2 sau đây sẽ mô phỏng 6 trong 64 hàm cơ sở đã sử dụng trong khối 8×8 DCT. Nó cũng phụ thuộc vào vị trí trên miền quang phổ mà biên độ đó được lưu trữ.

Các hàm cơ sở trên khối 8×8 DCT có dạng như sau :

$$c(u, v) = \frac{C(u)C(v)}{4} \sum_{x=0}^7 \sum_{y=0}^7 C(x, y) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \quad (4.2.14)$$

Trong đó :

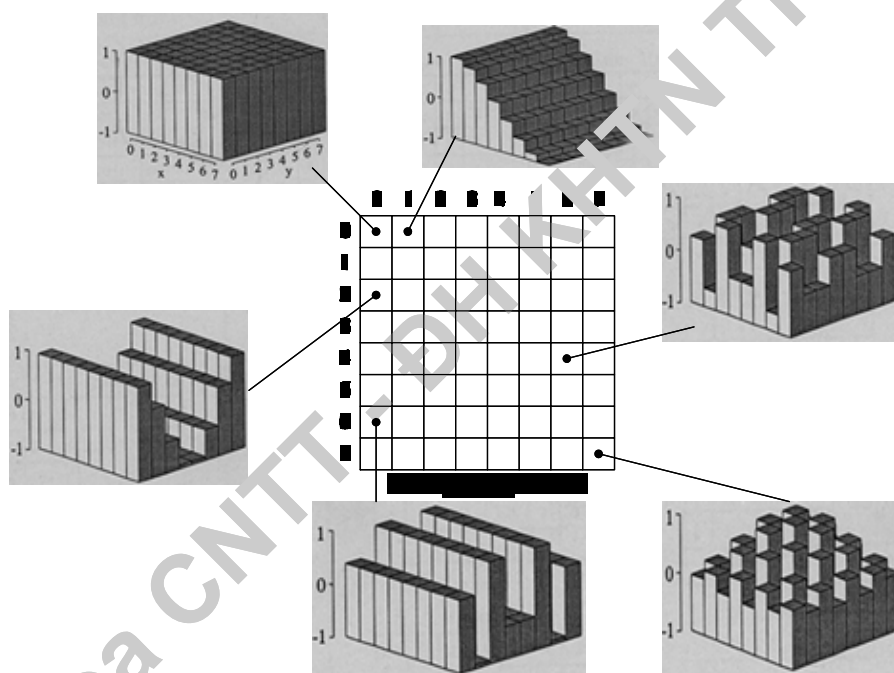
$C(x, y)$ - Các mẫu gốc trong khối ma trận 8×8 DCT

$c(u, v)$ - Các hệ số khối DCT 8×8

u - Tần số ngang chuẩn hóa ($0 \leq u \leq 7$)

v - Tần số đứng (mặt) chuẩn hóa ($0 \leq v \leq 7$)

$$C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & u, v = 0 \\ 1 & u, v = 1, \dots, 7 \end{cases} \quad (4.2.15)$$



Hình 4-2 Các hàm cơ sở của phép biến đổi Cosine rời rạc, Miền quang phổ của phép biến đổi Cosine rời rạc bao gồm một mảng hai chiều 8×8 , mỗi phần tử trong mảng là giá trị biên độ của một trong 64 hàm cơ sở.

Kết quả phép biến đổi Cosine rời rạc trên ma trận 8×8 , các thành phần có tần số thấp đã dồn về góc trên bên trái của ma trận quang phổ, trong khi đó các thành phần có tần số cao đã dồn về góc dưới bên phải của ma trận quang phổ.

Đối với hệ số tại vị trí $u, v = 0$, $C(u), C(v) = \frac{1}{\sqrt{2}}$, được gọi là thành phần

$$\text{DC của ma trận } 8 \times 8 \text{ DCT } c(0,0) = \frac{1}{8} \sum_{x=0}^7 \sum_{y=0}^7 C(x,y) \text{Cos}0 \cdot \text{Cos}0$$

(4.2.16)

Phương trình này cộng tất cả các giá trị trong khối 8×8 và chia kết quả cho 8. Theo thống kê thì kết quả này bằng 8 lần giá trị trung bình trong khối 8×8 .

Các hệ số còn lại tại các vị trí khác $u, v \neq 0$, $C(u), C(v) = 1$, được gọi là thành phần AC của ma trận 8×8 . Trong đó $c(0,1)$ bằng nửa chu kỳ của dạng sóng cosine khảo sát trên một chiều và $c(1,0)$ cũng bằng nửa chu kỳ của dạng sóng cosine khảo sát trên một chiều nhưng đã bị quay 90° .

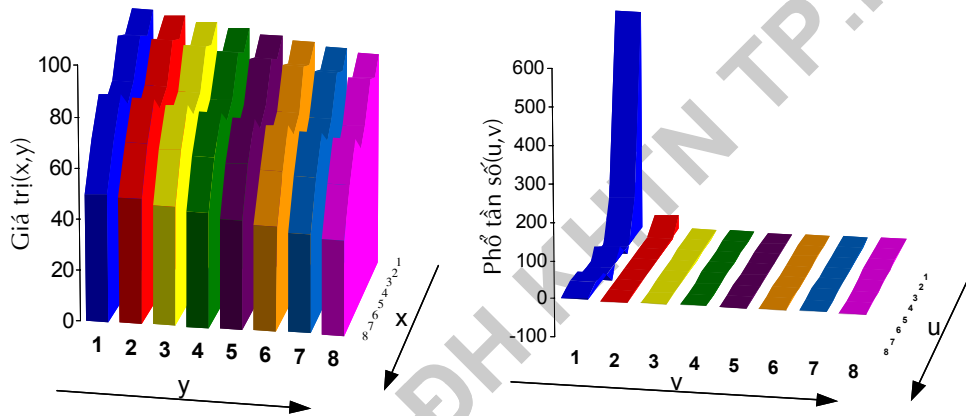
Trong khối 8×8 các giá trị của hệ số DCT đưa ra một giá trị DC lớn, biểu diễn giá trị trung bình từ khối 8×8 ban đầu và các giá trị nhỏ hơn rất nhiều so với DC đó chính là các thành phần có tần số cao theo chiều ngang và đứng. Tuy nhiên, các hệ số AC theo chiều ngang cao hơn các hệ số AC theo chiều đứng.

99	92	95	80	75	82	68	50
97	91	94	79	74	81	67	49
95	89	92	77	72	79	65	47
93	87	90	75	70	77	63	45
91	85	88	73	68	75	61	43
89	83	86	71	66	73	59	41
87	81	84	69	64	71	57	39

Bảng 4-1 Dữ liệu trên Ma trận hai chiều 8×8

		U							
	v	591	107	-17	28	-33	14	18	3
		35	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
		3	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0

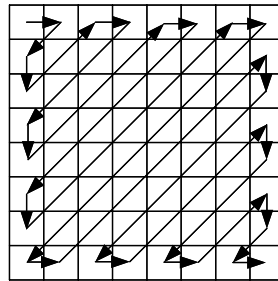
Bảng 4-2 Dữ liệu qua phép biến đổi 2D-DCT



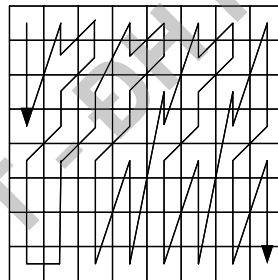
Hình 4-3 Quá trình mã hoá DCT trên một khối 8x8

4.2.4 Quét Zigzag

Từ khối các hệ số DCT, quét zigzag để có khả năng mã hóa và truyền dẫn theo kênh một chiều (1-D). Hình vẽ dưới đây sẽ đổi mảng hai chiều thành chuỗi liên tiếp các hệ số có tần số không gian tăng. Quét zigzag được chọn các hệ số có ý nghĩa nhất, nhóm các hệ số bằng 0 nhiều nhất có thể có. Sự phân bố các hệ số khác 0 phụ thuộc vào sự biến đổi giá trị của khối dữ liệu gốc và sự biến đổi mạnh về mặt giá trị trong khối dữ liệu gốc theo chiều đứng. Một cách quét khác.



Hình 4-4 Vẽ khối zigzag dạng 1



Hình 4-5 Vẽ khối zigzag dạng 2

Chương 5 SVM VÀ ỨNG DỤNG NHẬN DẠNG
KHUÔN MẶT

Khoa CNTT - ĐH KHTN TP.HCM

5.1 Cở sở lý thuyết của SVM

SVM là phương pháp học do Vladimir N. Vapnik đề xuất vào năm 1995, và ngày càng được sử dụng phổ biến trong nhiều lĩnh vực, đặc biệt là lĩnh vực phân loại mẫu và nhận dạng mẫu. Đồng thời có nhiều tính năng ưu việt so với các phương pháp cổ điển khác: dễ dàng xử lý, xử lý với tính ổn định cao trên dữ liệu phức tạp, có thể có số chiều lớn và quan trọng hơn cả là khả năng xử lý tổng quát.

5.1.1 Các khái niệm nền tảng

5.1.1.1 Đường bao tổng quát cho một hệ máy học

Khảo sát bao gồm l mẫu quan sát. Mỗi quan sát gồm một cặp: một vector $\mathbf{x}_i \in \mathbf{R}^n$, $i = 1, \dots, l$ với một giá trị xác định y_i mà giá trị của nó xuất phát từ việc gán chủ quan từ người tổ chức dữ liệu. Gọi $P(\mathbf{x}, \mathbf{y})$ là hàm phân phối xác suất giữa \mathbf{x} và \mathbf{y} và chưa được xác định tường minh. Cách tổ chức trên đây có tính tổng quát cao hơn so với việc kết hợp cứng giữa \mathbf{y} với mỗi \mathbf{x} , điều này cho phép tính được phân phối của \mathbf{y} dựa vào dữ liệu \mathbf{x} cho trước. Tuy nhiên, sau phần này, ta thừa nhận cố định \mathbf{y} với \mathbf{x} cho trước.

Hệ máy học có nhiệm vụ học ánh xạ $x_i \mapsto y_i$, được định nghĩa từ một tập hợp các ánh xạ $x \mapsto f(x, \alpha)$, trong đó hàm $f(x, \alpha)$ được gián nhãn bởi các tham số α (α có thể hiệu chỉnh được trong quá trình xử lý trên tập học). Hệ máy học có thể xem như là một hệ quyết định. Với dữ liệu đầu vào là \mathbf{x} cho trước, chọn ra một α thích hợp, và kết xuất sẽ là $f(x, \alpha)$. Việc chọn α có thể có nhiều cách khác nhau, ở đây chúng ta sẽ tiếp cận theo phương pháp máy học.

Lỗi thử nghiệm đối với một hệ máy học đã được huấn luyện:

$$R(\alpha) = \int \frac{1}{2} |y - f(x, \alpha)| dP(x, y) \quad (5.1)$$

Nếu tồn tại hàm mật độ $\mathbf{p}(\mathbf{x}, \mathbf{y})$ thì $d\mathbf{P}(\mathbf{x}, \mathbf{y})$ có thể được viết thành $d\mathbf{P}(\mathbf{x}, \mathbf{y}) = \mathbf{p}(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}$. Đây là cách viết khác của trung bình lỗi, nhưng trong trường hợp đã ước lượng được $\mathbf{P}(\mathbf{x}, \mathbf{y})$ thì cách viết này sẽ không còn ý nghĩa nữa.

$\mathbf{R}(\alpha)$ được gọi là lỗi kì vọng, hay lỗi. Ở đây ta gọi nó là lỗi thực. Lỗi huấn luyện (thực nghiệm) $\mathbf{R}_{\text{emp}}(\alpha)$ được định nghĩa là độ đo tỷ lệ lỗi trung bình xảy ra trên tập học (Ở đây chỉ chú trọng vào trường hợp dữ liệu là hữu hạn):

$$R_{emp}(\alpha) = \frac{1}{2l} \sum_{i=1}^l |y_i - f(x_i, \alpha)| \quad (5.2)$$

$R_{emp}(\alpha)$ là một giá trị tường minh tương ứng với một hệ số α riêng từ dữ liệu huấn luyện riêng $\{\mathbf{x}_i, \mathbf{y}_i\}$.

Đại lượng $\frac{1}{2}|y_i - f(x_i, \alpha)|$ được gọi là độ lệch. Trong trường hợp này, nó chỉ có thể lấy giá trị 0 và 1 . Chọn η sao cho $0 \leq \eta \leq 1$ và cho độ lệch nhận các giá trị này, với xác suất $1-\eta$, ta có:

$$R(\alpha) \leq R_{emp}(\alpha) + \sqrt{\left(\frac{h(\log(2l/h) + 1) - \log(\eta/4)}{l} \right)} \quad (5.3)$$

h là một số nguyên không âm và còn được gọi là chiều VC (Vapnik Chervonenkis) (**VC-dimension**). Gọi vế phải của (5.3) là đường bao lỗi hay biên lỗi. Các thuật ngữ trước đây của một số nhà nghiên cứu: (Guyon et al., 1992) gọi là lỗi được thừa nhận nhưng cách gọi này có thể gây ra một số nhầm lẫn, vì nó thực sự chỉ là đường bao trên miền lỗi chứ không phải là giá trị chính xác của lỗi, và nó chỉ đúng ở một xác suất nào đó nên thật sự là không đảm bảo được độ đo này là chính xác. Thuật ngữ thứ hai là độ tin cậy Vapnik Chervonenkis.

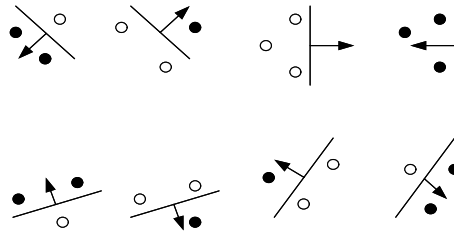
5.1.1.2 Chiều VC (VC-dimension)

Chiều VC là một thuộc tính của tập hàm $\{f(\alpha)\}$ (ta dùng α như là một tập các tham số có đặc điểm chung sau: Cứ chọn ra một hệ số α thì sẽ xác định được một hàm riêng tương ứng), và dùng chiều VC chúng ta có thể biểu diễn các dạng biến thể khác nhau của hàm f có thể có. Ở đây chỉ khảo sát các loại hàm cho trường hợp giải quyết bài toán nhận dạng mẫu hai lớp, như vậy $f(\alpha) \in \{-1, 1\} \forall \mathbf{x}, \alpha$. Cho trước tập quan sát gồm l (mẫu), có thể gán nhãn theo 2^l cách, và với mỗi cách gán nhãn, có thể tìm được một thành viên của tập $\{f(\alpha)\}$ gán chính xác các nhãn này, ta gọi tập các điểm như vậy bị shatter (phân cách) bởi tập hàm này. Chiều VC cho tập hàm $\{f(\alpha)\}$ được định nghĩa là số điểm huấn luyện lớn nhất có thể bị phân tách bởi $\{f(\alpha)\}$. Chú ý rằng, nếu chiều VC là h khi đó tồn tại ít nhất một tập h điểm có thể bị phân cách, nhưng không chắc mọi tập h điểm có thể bị phân cách.

5.1.1.3 Phân hoạch tập dữ liệu bằng các siêu mặt có hướng

Giả sử không gian dữ liệu là \mathbf{R}^2 , và tập hàm $\{f(\alpha)\}$ là các đường thẳng có hướng, như vậy với một đường cho trước, mọi điểm mẫu trên một mặt của đường sẽ được gán nhãn bằng 1 , và các điểm thuộc về mặt khác sẽ được gán nhãn -1 . Hướng mũi

tên trong Hình 5-1 xác định mặt của đường thẳng mà các điểm thuộc về mặt đó sẽ được gán nhãn 1. Có thể tìm được ba điểm phân cách bởi tập các hàm này, nhưng không thể tìm được bốn điểm. Do đó chiều VC cho các đường có định hướng trong không gian hai chiều (\mathbf{R}^2) là 3.



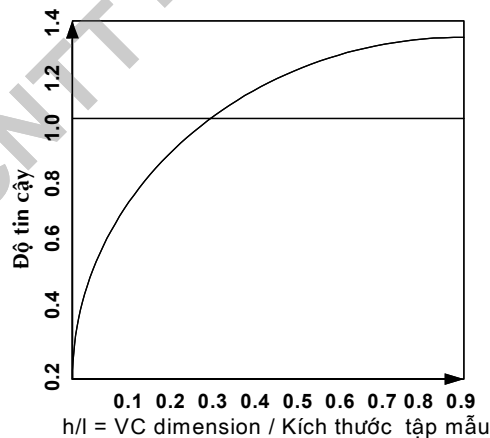
Hình 5-1 Ba điểm trong \mathbf{R}^2 .

Xét các siêu mặt trong không gian n chiều (\mathbf{R}^n).

Định lý 1: Khảo sát tập mẫu gồm m điểm trong không gian \mathbf{R}^n . Chọn một điểm bất kỳ làm điểm tọa độ gốc. Khi đó m điểm này có thể bị phân rã bằng các siêu mặt (đường thẳng có định hướng) nếu và chỉ nếu vị trí các vector của các điểm đang đề cập là độc lập tuyến tính (“Mangasarian, 1969”).

Hệ quả: VC-dimension cho các siêu mặt có hướng trong \mathbf{R}^n là $n+1$, vì ta luôn có thể chọn $n+1$ điểm dữ liệu, và sau đó chọn một điểm bất kì trong số đó làm điểm gốc, để vị trí các vector của các điểm đang đề cập là độc lập tuyến tính.

5.1.1.4 Cực tiểu đường bao lỗi trên cơ sở cực tiểu chiều VC

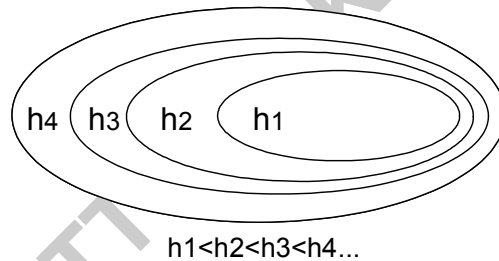


Hình 5-2 Độ tin cậy VC là hàm đơn điệu theo h

Hình 5-2: Cho thấy nhóm biểu thức thứ hai bên vế phải của phương trình (5.3) $(\sqrt{\frac{h(\log(2l/h)+1)-\log(\eta/4)}{l}})$ biến thiên theo h , bằng cách chọn độ tin cậy 95% ($\eta = 0.05$), tập mẫu huấn luyện $l=10,000$ (mẫu). Chiều VC là hàm tăng đều theo h . Điều này đúng với bất kỳ giá trị của l .

5.1.1.5 Cực tiểu hoá lỗi theo cấu trúc (SRM)

Thuật ngữ độ tin cậy VC trong (5.3) tùy thuộc vào việc chọn các họ hàm khác nhau, trong khi lỗi huấn luyện và lỗi thực phụ thuộc vào một hàm riêng được chọn bằng thủ tục huấn luyện. Ta muốn tìm một tập con của tập các hàm được chọn, để đường bao lỗi cho tập con đó là cực tiểu. Chiều VC h là số nguyên, nên không thể biến đổi liên tục. Thay vào đó, người ta đưa ra một “cấu trúc” bằng việc chia toàn bộ tập các hàm thành các tập con lồng nhau Hình 5-3. Với mỗi tập con, ta có thể tính h , hay đường bao của h . Cực tiểu hóa lỗi theo cấu trúc (SRM) khi đó bao gồm việc tìm tập con hàm cực tiểu đường bao lỗi thực. Việc này được thực hiện bằng cách huấn luyện chuỗi các máy, cho mỗi tập con, trong đó với một tập con cho trước mục đích của việc huấn luyện là cực tiểu lỗi huấn luyện.



Hình 5-3 Các tập hàm học lồng vào nhau được sắp thứ tự theo chiều VC.

5.1.2 SVM tuyến tính

5.1.2.1 Trường hợp dữ liệu có thể phân cách được

Bắt đầu với trường hợp đơn giản nhất: máy học được huấn luyện trên dữ liệu có thể phân loại tuyến tính. Gán nhãn dữ liệu huấn luyện $\{x_i, y_i\}$, $i = 1, \dots, l$, $y_i \in \{-1, 1\}$, $x_i \in \mathbf{R}^d$. Giả sử có các siêu mặt phẳng phân loại mẫu dương với mẫu âm (gọi là “siêu mặt phân cách”). Điểm x nằm trên siêu mặt thỏa phương trình $w \cdot x + b = 0$, trong đó w là pháp tuyến của siêu mặt, $|b| / \|w\|$ là khoảng cách từ siêu

mặt đến gốc toạ độ, và $\|\mathbf{w}\|$ độ lớn (Euclide) của \mathbf{w} . Đặt $\mathbf{d}_+(\mathbf{d}_-)$ là khoảng cách ngắn nhất từ siêu mặt phân cách đến mẫu dương (âm) gần nhất. Định nghĩa “bờ” (margin) của siêu mặt phân cách (kí hiệu \mathbf{r}), là $(\mathbf{d}_+)+(\mathbf{d}_-)$. Với trường hợp tập mẫu có thể phân loại tuyến tính, thuật toán SVM chỉ đơn giản là tìm siêu mặt có khoảng cách bờ là cực đại cực đại. Các mô tả trên đây được công thức hoá như sau: giả sử mọi điểm trong tập học thỏa các ràng buộc:

$$\mathbf{x}_i\mathbf{w} + \mathbf{b} \geq +1 \quad \text{với } y_i = +1 \quad (5.4)$$

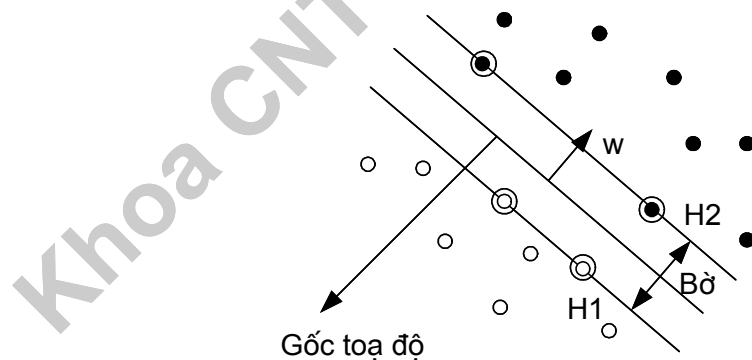
$$\mathbf{x}_i\mathbf{w} + \mathbf{b} \leq -1 \quad \text{với } y_i = -1 \quad (5.5)$$

Kết hợp thành một bất đẳng thức ràng buộc:

$$y_i(\mathbf{x}_i\mathbf{w} + \mathbf{b}) - 1 \geq 0 \quad \forall i \quad (5.6)$$

Các mẫu dữ liệu thỏa công thức (5.4) nằm trên siêu mặt \mathbf{H}_1 : $\mathbf{x}_i\mathbf{w} + \mathbf{b} = 1$ có pháp tuyến là vector \mathbf{w} , và khoảng cách đến gốc toạ độ là $|\mathbf{1}-\mathbf{b}|/\|\mathbf{w}\|$. Tương tự, các mẫu thỏa công thức (5.5) nằm trên siêu mặt \mathbf{H}_2 : $\mathbf{x}_i\mathbf{w} + \mathbf{b} = -1$, có pháp tuyến là \mathbf{w} và khoảng cách đến gốc toạ độ là $|\mathbf{-1}-\mathbf{b}|/\|\mathbf{w}\|$. Khi đó, $\mathbf{d}_+ = \mathbf{d}_- = 1 / \|\mathbf{w}\|$ và bờ $\mathbf{r} = 2/\|\mathbf{w}\|$. Lưu ý rằng \mathbf{H}_1 và \mathbf{H}_2 song song với nhau, và không có điểm dữ liệu nào nằm giữa chúng. Vì vậy, ta có thể tìm cặp siêu mặt có bờ (\mathbf{r}) cực đại, bằng việc cực tiểu $\|\mathbf{w}\|$ với ràng buộc (5.6).

Ta mong muốn lời giải cho trường hợp không gian hai chiều có dạng như trong Hình 5-4. Những điểm huấn luyện thỏa phương trình (5.6) (tức những điểm nằm trên một trong hai siêu mặt $\mathbf{H}_1, \mathbf{H}_2$), và việc loại bỏ chúng làm thay đổi lời giải, được gọi là các vector hỗ trợ; đó là các điểm được bao bằng các hình tròn trong Hình 5-4.



Hình 5-4 Siêu mặt phân cách tuyến tính cho trường hợp phân cách được và kí hiệu các support vector chính là các điểm được bao bằng viền tròn

Một cách để giải quyết bài toán là dùng hàm **Largange**. Có hai lý do cho

điều này. **Thứ nhất** là ràng buộc (5.6) sẽ được thế bằng ràng buộc trên hệ số nhân Lagrange, để dễ làm việc hơn. **Thứ hai** là dữ liệu huấn luyện sẽ chỉ xuất hiện dưới dạng phép nhân vô hướng giữa các vector. Điều này cho phép tổng quát hoá cho trường hợp phi tuyến.

Đưa ra các hệ số nhân Lagrange dương $\alpha_i, i = 1, \dots, l$ và $\alpha_i > 0$, cho các ràng buộc bất đẳng thức (5.6). (Nhớ rằng với ràng buộc dạng $c_i \geq 0$, phương trình ràng buộc được nhân với hệ số nhân Lagrange dương và bị trừ khỏi hàm mục tiêu. Với các ràng buộc đẳng thức, hệ số nhân Lagrange không bị ràng buộc). Khi đó hàm Lagrange có dạng như sau:

$$L_p = \frac{1}{2} \|w\|^2 - \sum_{i=1}^l \alpha_i y_i (x_i w + b) + \sum_{i=1}^l \alpha_i \quad (5.7)$$

Ta phải cực tiểu L_p theo $(w), (b)$, đồng thời đòi hỏi đạo hàm của L_p triệt tiêu với mọi α_i , với điều kiện $\alpha_i \geq 0$ (gọi tập ràng buộc này là C_1). Hay giải bài toán đối ngẫu đó là cực đại L_p với điều kiện đạo hàm của L_p triệt tiêu với w, b và cũng với ràng buộc $\alpha_i \geq 0$ (gọi tập ràng buộc này là C_2).

Đạo hàm L_p triệt tiêu với w và b ta có các điều kiện:

$$w = \sum_i \alpha_i y_i x_i \quad (5.8)$$

$$\sum_i \alpha_i y_i = 0 \quad (5.9)$$

Vì đây là các ràng buộc tuyến tính Thay vào (5.7) ta được:

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j \quad (5.10)$$

Việc huấn luyện SVM (trường hợp tuyến tính và có thể phân loại) là làm cực đại L_D theo α_i , với ràng buộc (5.9) và α_i dương, với lời giải được tính theo (5.8). Trong lời giải, các mẫu thỏa $\alpha_i > 0$ được gọi là “vector hỗ trợ”, và nằm trên một trong hai siêu mặt phẳng H_1 và H_2 . Các điểm dữ liệu còn lại có $\alpha_i = 0$ và nằm trên H_1 hoặc H_2 (thỏa đẳng thức (5.6)) hoặc nằm về một phía của H_1 hoặc H_2 (thỏa bất đẳng thức (5.6)). Với những máy này, các vector hỗ trợ là các thành phần tới hạn của tập huấn luyện. Chúng nằm gần đường biên quyết định; nếu mọi điểm huấn luyện khác bị loại bỏ và việc huấn luyện được lặp lại, siêu mặt phẳng phân cách được tìm thấy không đổi.

5.1.2.2 Điều kiện tối ưu Karush-Kuhn-Tucker

Điều kiện Karush-Kuhn-Tucker (KKT) có vai trò quan trọng đối với bài toán tối ưu ràng buộc. Với bài toán trên, điều kiện KKT có thể phát biểu:

$$\frac{\partial}{\partial w_v} L_P = w_v - \sum_i \alpha_i y_i x_{iv} = 0 \quad v = 1, \dots, d \quad (5.11)$$

$$\frac{\partial}{\partial b} L_P = -\sum_i \alpha_i y_i = 0 \quad (5.12)$$

$$y_i(\mathbf{x}_i \mathbf{w} + \mathbf{b}) - 1 \geq 0 \quad i = 1, \dots, l \quad (5.13)$$

$$\alpha_i \geq 0 \quad \forall i \quad (5.14)$$

$$\alpha_i (y_i(\mathbf{w} \mathbf{x}_i + \mathbf{b}) - 1) = 0 \quad \forall i \quad (5.15)$$

Điều kiện KKT là điều kiện cần và đủ để \mathbf{w} , \mathbf{b} , α là một lời giải (Fletcher, 1987). Do đó giải bài toán SVM tương đương với việc tìm lời giải cho các điều kiện KKT.

Vector \mathbf{w} được tính bằng cách huấn luyện, nhưng \mathbf{b} thì không. Tuy nhiên \mathbf{b} dễ dàng tính được khi sử dụng các điều kiện KKT bổ sung, công thức (5.15), bằng việc chọn i có $\alpha_i \neq 0$ và tính \mathbf{b} (lấy giá trị trung bình của \mathbf{b} từ các phương trình trên).

5.1.2.3 Trường hợp dữ liệu không thể phân cách được

Thuật toán trên chỉ phù hợp cho trường hợp tập dữ liệu học có thể phân cách được, khi áp dụng cho dữ liệu không thể phân cách tuyến tính, sẽ không nhận được lời giải khả thi do hàm Lagrange lớn. Để mở rộng ý tưởng này cho trường hợp dữ liệu không thể phân cách ta đưa thêm các biến slack (chùng, lỏng) dương $\xi_i \geq 0$, $i = 1, \dots, l$ vào các ràng buộc như sau:

$$\mathbf{x}_i \mathbf{w} + \mathbf{b} \geq +1 - \xi_i, \quad \text{với } y_i = +1 \quad (5.34)$$

$$\mathbf{x}_i \mathbf{w} + \mathbf{b} \leq -1 + \xi_i, \quad \text{với } y_i = -1 \quad (5.35)$$

$$\xi_i \geq 0 \quad \forall i \quad (5.36)$$

Khi có lỗi xuất hiện, ξ_i tương ứng sẽ lớn hơn 1, như vậy $\sum_i \xi_i$ là biên trên của số lỗi huấn luyện. Do đó một cách để gán thêm lỗi huấn luyện, là thay đổi hàm mục tiêu từ việc cực tiểu $\|\mathbf{w}\|^2/2$ sang việc cực tiểu $\|\mathbf{w}\|^2 / 2 + C(\sum_i \xi_i)^k$, trong đó C là tham số do người dùng chọn, và C càng lớn thì tỉ lệ lỗi sẽ càng thấp.

Khi đó hàm **Lagrange** trở thành:

Cực đại:

$$L_D \equiv \sum_i \alpha_i - \frac{1}{2} \sum \alpha_i \alpha_j y_i y_j x_i x_j \quad (5.37)$$

với điều kiện:

$$0 \leq \alpha_i \leq C, \quad (5.38)$$

$$\sum_i \alpha_i y_i = 0 \quad (5.39)$$

lời giải được cho bởi

$$\mathbf{w} = \sum_{i=1}^{N_S} \alpha_i y_i x_i$$

(5.40)

trong đó N_S là số vector hỗ trợ. Lời giải được minh hoạ trong Hình 5-5.

Xét điều kiện KKT:

$$L_P = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i - \sum_i \alpha_i \{y_i (x_i \cdot \mathbf{w} + b) - 1 + \xi_i\} - \sum_i \mu_i \xi_i \quad (5.41)$$

với μ_i là các hệ số nhân **Lagrange** đảm bảo ξ_i dương. Khi đó điều kiện

KKT là

$$\frac{\partial L_P}{\partial w_v} = w_v - \sum_i \alpha_i y_i x_{iv} = 0 \quad (5.42)$$

$$\frac{\partial L_P}{\partial b} = -\sum_i \alpha_i y_i = 0 \quad (5.43)$$

$$\frac{\partial L_P}{\partial \xi_i} = C - \alpha_i - \mu_i = 0 \quad (5.44)$$

$$y_i (x_i \cdot \mathbf{w} + b) - 1 + \xi_i \geq 0 \quad (5.45)$$

$$\xi_i \geq 0 \quad (5.46)$$

$$\alpha_i \geq 0 \quad (5.47)$$

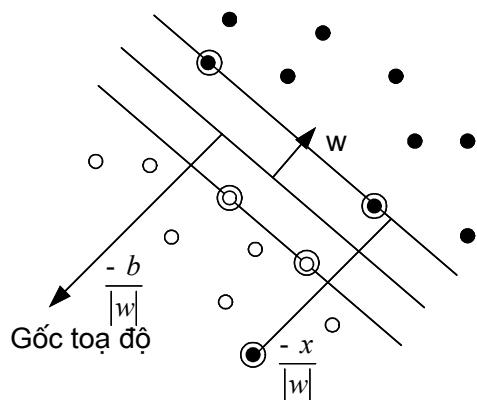
$$\mu_i \geq 0 \quad (5.48)$$

$$\alpha_i \{y_i (x_i \cdot \mathbf{w} + b) - 1 + \xi_i\} = 0 \quad (5.49)$$

$$\mu_i \xi_i = 0 \quad (5.50)$$

Có thể dùng các điều kiện bổ sung **KKT**, công thức (5.49) và (5.50), để xác định ngưỡng **b**. Công thức (5.44) và (5.50) cho thấy rằng $\xi_i = 0$ nếu $\alpha_i < C$. Do đó có thể lấy điểm huấn luyện bất kỳ thỏa $0 < \alpha_i < C$ để dùng công thức (5.44) (với $\xi_i =$

0) để tính b .



Hình 5-5 Siêu mặt phân cách tuyến tính cho trường hợp không phân cách được.

Khoa CNTT - ĐH KHTN TP.HCM

5.1.3 SVM phi tuyến

Làm thế nào các phương thức đã khảo sát trên có thể được tổng quát hoá cho trường hợp hàm quyết định không phải là hàm tuyến tính đối với dữ liệu? Ta thấy rằng dữ liệu trong bài toán huấn luyện, công thức (5.37) – (5.39), xuất hiện dưới dạng tích vô hướng $x_i \cdot x_j$. Giả sử ta đã ánh xạ dữ liệu sang không gian Euclide

H khác (số chiều có thể vô hạn) dùng hàm ánh xạ Φ :

$$\Phi: R^d \mapsto H \quad (5.51)$$

Khi đó thuật toán huấn luyện chỉ phụ thuộc vào dữ liệu qua tích vô hướng trong H, tức là hàm có dạng $\Phi(x_i) \cdot \Phi(x_j)$. Nếu có một hàm xử lý chính **K** (hàm Kernel) mà $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$, ta sẽ chỉ cần dùng hàm **K** trong thuật toán huấn luyện, mà không cần biết dạng tường minh của Φ là gì? Chẳng hạn:

$$K(x_i, x_j) = e^{-\|x_i - x_j\|^2 / 2\sigma^2} \quad (5.52)$$

Trong ví dụ này, H có số chiều vô hạn, vì thế không dễ làm việc với Φ một cách tường minh. Tuy nhiên, nếu thay $x_i \cdot x_j$ bằng $K(x_i, x_j)$ trong thuật toán huấn luyện, thuật toán sẽ tạo ra vector hỗ trợ trong không gian số chiều vô hạn, hơn nữa thời gian huấn luyện tương đương thời gian huấn luyện trên dữ liệu chưa được ánh xạ. Mọi xem xét trong các phần trước vẫn thỏa, vì ta vẫn làm việc với trường hợp phân cách tuyến tính, nhưng trong một không gian khác.

Sử dụng hệ thống này như thế nào? Ta cần tìm w , trong H (5.40). Nhưng khi thử nghiệm, máy được sử dụng bằng cách tính tích vô hướng của mẫu thử nghiệm x với w , hay cụ thể hơn tính dấu của

$$f(x) = \sum_{i=1}^{N_s} \alpha_i y_i \Phi(s_i) \cdot \Phi(x) + b = \sum_{i=1}^{N_s} \alpha_i y_i K(s_i, x) + b \quad (5.53)$$

trong đó s_i là support vector. Vậy ta có thể dùng $K(s_i, x) = \Phi(s_i) \cdot \Phi(x)$.

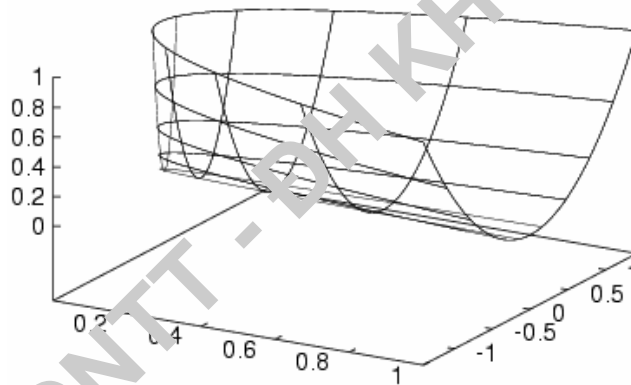
Gọi không gian dữ liệu là Λ . (Λ để chỉ không gian có số chiều nhỏ, và H để chỉ không gian có số chiều lớn: thường thì đầu ra của Φ có chiều cao hơn đầu vào). Nhớ rằng, ngoài việc w nằm trên H ra, nói chung không có vector nào trong Λ , qua ánh xạ Φ , thành w . Nếu có, $f(x)$ trong công thức (5.53) có thể được tính bằng một bước, bỏ qua việc tính tổng (và tạo ra vector hỗ trợ tương ứng nhanh

hơn N_S lần). Dễ dàng tìm được dạng hàm xử lý chính (Kernel) (chẳng hạn hàm phép nhân vô hướng \mathbf{x}_i trong Λ) để thuật toán huấn luyện và lời giải tìm được là độc lập với số chiều của cả Λ và H .

Giả sử dữ liệu là các vector trong không gian \mathbf{R}^2 , và chọn $K(x_i, x_j) = (x_i \cdot x_j)^2$. Khi đó dễ dàng tìm được không gian H , và ánh xạ Φ từ \mathbf{R}^2 vào H , để $(x \cdot y)^2 = \Phi(x) \cdot \Phi(y)$: chọn $H = \mathbf{R}^3$ và

$$\Phi(x) = \begin{pmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{pmatrix} \quad (5.54)$$

Với dữ trong Λ đã xác định trong hình vuông $[-1,1] \times [-1,1] \in \mathbf{R}^2$, ảnh của Φ được biểu diễn trong Hình 5-6. Hình này cũng cho thấy rằng ảnh của Φ có thể nằm trong không gian có số chiều lớn hơn, nhưng nó chỉ là một mặt (có thể méo mó) mà số chiều bên trong vẫn là Λ .



Hình 5-6 Ảnh, trong H , với hình vuông $[-1,1] \times [-1,1] \in \mathbf{R}^2$ dưới ánh xạ Φ

Chú ý rằng hàm ánh xạ Φ và không gian H là không duy nhất với một hàm xử lý chính (kernel) cho trước. Ta có thể chọn $H = \mathbf{R}^3$ và

$$\Phi(x) = \frac{1}{\sqrt{2}} \begin{pmatrix} (x_1^2 - x_2^2) \\ 2x_1x_2 \\ (x_1^2 + x_2^2) \end{pmatrix} \quad (5.55)$$

hay $H = \mathbf{R}^3$ và

$$\Phi(x) = \begin{pmatrix} x_1^2 \\ x_1^2 x_1^2 \\ x_1^2 x_1^2 \\ x_2^2 \end{pmatrix} \quad (5.56)$$

Điều kiện Mercer

Hàm xử lý chính nào tồn tại tại cặp $\{H, \Phi\}$, với các thuộc tính đã mô tả ở trên, hàm nào không? Câu trả lời được cho bởi điều kiện Mercer (Vapnik, 1995; Courant và Hilbert, 1953):

Khoa CNTT - ĐH KHTN TP.HCM

Tồn tại ánh xạ Φ và

$$K(x, y) = \sum_i \Phi(x)_i \cdot \Phi(y)_i \quad (5.57)$$

nếu và chỉ nếu, với bất kỳ $g(x)$ mà

$$\int g(x)^2 dx \quad \text{xác định} \quad (5.58)$$

thì

$$\int K(x, y)g(x)g(y)dxdy \geq 0 \quad (5.59)$$

Với các trường hợp cụ thể, không dễ kiểm tra điều kiện **Mercer** có thỏa hay không. Công thức (5.59) phải thỏa với mọi hàm g với L_2 xác định (nghĩa là thỏa (5.58)). Tuy nhiên, có thể chứng minh điều kiện thỏa với phân tử của phép nhân tích vô hướng là một số nguyên dương: $K(x, y) = (x \cdot y)^p$. Ta phải chứng tỏ rằng

$$\int (\sum_{i=1}^d x_i y_i)^p g(x)g(y)dxdy \geq 0 \quad (5.60)$$

Khai triển $(\sum_{i=1}^d x_i y_i)^p$

$$\frac{p!}{r_1!r_2!\dots(p-r_1-r_2\dots)!} \int x_1^{r_1} x_2^{r_2} \dots y_1^{r_1} y_2^{r_2} g(x)g(y)dxdy \quad (5.61)$$

kết hợp với vế trái (5.59):

$$= \frac{p!}{r_1!r_2!\dots(p-r_1-r_2\dots)!} (\int x_1^{r_1} x_2^{r_2} \dots g(x)dx)^2 \geq 0 \quad (5.62)$$

5.1.4 Chiều VC của SVM

Ta thấy rằng chiều VC của SVM có thể rất lớn (thậm chí vô hạn). Ta sẽ khảo sát lý do tại sao không phụ thuộc vào điều này mà SVM vẫn thực hiện tổng quát hoá tốt. Tuy nhiên cần nhấn mạnh rằng đây chỉ là các lý do cơ bản. Và hiện tại chưa có thuyết nào bảo đảm họ SVM cho trước sẽ có độ chính xác cao với bài toán cho trước.

Gọi hàm xử lý chính thỏa điều kiện Mercer là hàm xử lý chính dương, và không gian H tương ứng \mathbf{H} là không gian nhúng. Và gọi không gian nhúng bất kỳ có chiều cực tiểu với một hàm xử lý chính cho trước là một “không gian nhúng cực tiểu”. Ta có

Định lý: Đặt K là hàm xử lý chính dương tương ứng với không gian nhúng cực tiểu H . Khi đó chiều VC của SVM tương ứng (với C trong (5.38) được phép lấy mọi giá trị) bằng $\dim(H) + 1$.

5.1.5 Hạn chế của phương pháp SVM

Có lẽ hạn chế lớn nhất của tiếp cận theo phương pháp SVM nằm ở việc lựa chọn hàm xử lý chính. Khi đã có hàm xử lý chính, bộ phân loại SVM chỉ có một tham số cho người dùng chọn (ngưỡng lỗi), nhưng hàm xử lý chính là bài toán lớn với khả năng của các tham số. Một số nghiên cứu đã giới hạn hàm xử lý chính bằng cách dùng tri thức cho trước, nhưng việc chọn hàm xử lý chính tốt nhất cho một bài toán cho trước vẫn là một bài toán cần nghiên cứu.

Giới hạn thứ hai là tốc độ và kích thước, cả trong huấn luyện và thử nghiệm.

5.2 Nhận dạng khuôn mặt người với SVM

Trong đề tài này, SVM được kết hợp với cây nhị phân để giải quyết bài toán nhận dạng khuôn mặt đa lớp.

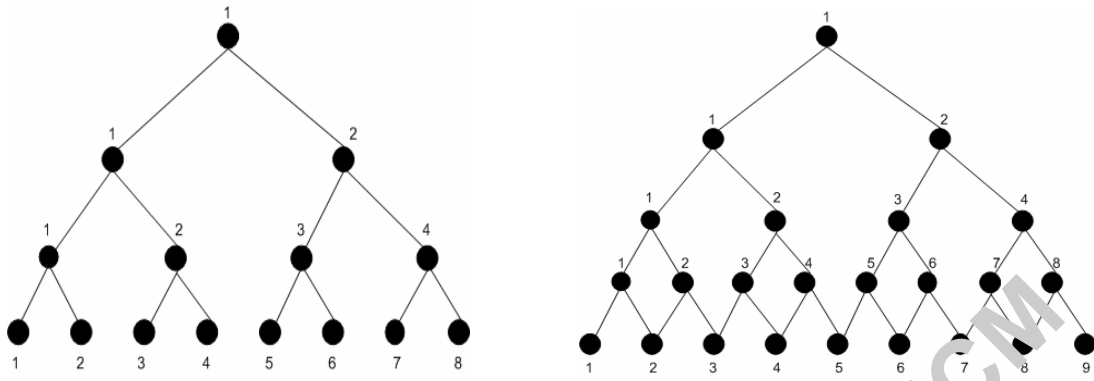
5.2.1 Nhận dạng đa lớp dùng SVM với cây nhị phân

Hệ thống nhận dạng mẫu đa lớp có thể có được bằng cách dùng SVM. Có hai chiến lược cho mục đích này: (1) chiến lược one-against-all (một đối tất cả) để phân loại mỗi lớp với mọi lớp còn lại; (2) chiến lược một đối một để phân loại giữa từng cặp. Chiến lược đầu tiên thường cho kết quả phân loại nhập nhằng. Ta theo chiến lược thứ hai cho bài toán nhận dạng đa lớp.

Giả sử có tám lớp trong tập dữ liệu, cây quyết định được biểu diễn như trong Hình 5-7, trong đó các số 1-8 mã hoá các lớp. Các số mã hoá các lớp là tùy ý không có nghĩa theo thứ tự. Bằng cách so sánh từng cặp, chọn ra một lớp biểu diễn “phần thắng” của hai lớp hiện hành. Các lớp được chọn (từ cấp thấp nhất của cây nhị phân) sẽ lên cấp trên với vòng thử nghiệm khác. Cuối cùng một lớp duy nhất sẽ xuất hiện ở đỉnh của cây.

Khi c không là bội số của 2, ta phân tích: $c = 2^{n_1} + 2^{n_2} + \dots + 2^{n_l}$, với $n_1 \geq n_2 \geq \dots \geq n_l$. Nếu c lẻ thì $n_1 = 0$ và nếu c chẵn thì $n_1 > 0$. Cách phân tích c không duy nhất. Sau khi phân tích, việc nhận dạng được thực hiện trong từng cây nhị phân, các lớp đầu ra của các cây nhị phân này được dùng lại để tạo ra cây nhị phân khác. Quá trình này được lặp lại cho đến khi chỉ còn một đầu ra.

SVM học $c(c-1)/2$ hàm phân biệt trong giai đoạn huấn luyện, và thực hiện $c-1$ phép so sánh dưới cấu trúc cây nhị phân đã tạo ra.



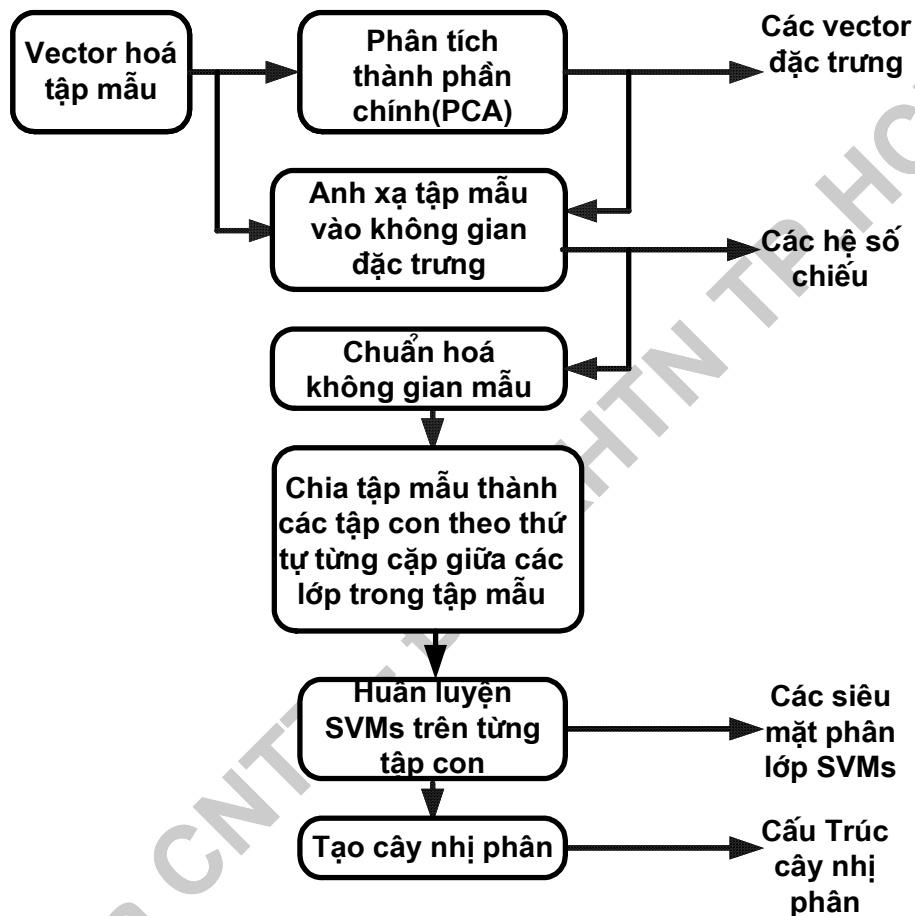
Hình 5-7 Trái: Cấu trúc cây nhị phân với số lớp bằng số mũ của 2. Phải: số lớp không bằng số mũ của 2.

Khoa CNTT - ĐH KHTN TP. HCM

5.2.2 Nhận dạng khuôn mặt dùng SVM

5.2.2.1 Giai đoạn huấn luyện hệ thống

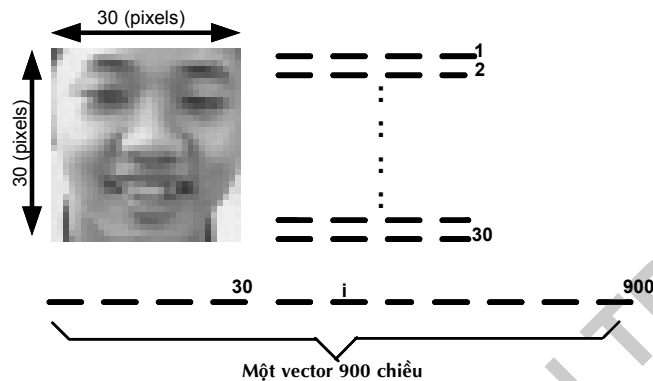
5.2.2.1.1 Huấn luyện SVM cho bài toán nhận dạng khuôn mặt



Hình 5-8 Các tác vụ huấn luyện hệ thống SVMs nhận dạng khuôn mặt

5.2.2.1.2 Vector hoá tập mẫu khuôn mặt thô

Đây chính là một bước biểu diễn ảnh khuôn mặt vào máy tính, hình thức biểu diễn này chúng tôi đã có đề cập trong phần mô tả dữ liệu nhận dạng khuôn mặt. Chi tiết vector hoá một mẫu khuôn mặt được trình bày trong Hình 5-9 dưới đây.



Hình 5-9 Vector hoá mẫu khuôn mặt

- Tách ảnh mẫu khuôn mặt 30×30 (pixels) thành 30 dòng ảnh theo thứ tự từ trên xuống dưới, mỗi dòng ảnh bao gồm 30 (điểm ảnh).
- Vector hoá mẫu khuôn mặt được thực hiện bằng cách nối liên tiếp các dòng ảnh theo thứ tự này liên tiếp nhau và kết quả từ một ma trận ảnh hai chiều 30×30 thành một vector có số chiều bằng 900.

5.2.2.1.3 Rút trích đặc trưng khuôn mặt

Giai đoạn rút trích đặc trưng khuôn mặt bao gồm ba bước chính đó là thực hiện phép phân tích thành phần chính, Ánh xạ tập mẫu vào không gian đặc trưng, và chuẩn hoá không gian mẫu.

Trong tất cả các hệ nhận dạng mà đặc biệt là các hệ nhận dạng tự động nói riêng, mức độ thành công không những phụ thuộc vào các thuật toán nhận dạng tốt mà còn phụ thuộc rất nhiều vào tập mẫu dữ liệu huấn luyện. Việc chọn tập mẫu huấn luyện sao cho phù hợp với mục đích ứng dụng yêu cầu và vừa đảm bảo được tính tổng quát cho một hệ nhận dạng là rất khó. Bởi vì ta không thể lường trước mọi biến thể có thể tác động đến đối tượng quan tâm lúc thu nhận ảnh. Các ảnh hưởng phổ biến mà có thể liệt kê ở đây: Thứ nhất, môi trường lấy mẫu biến động rất phức tạp như điều kiện thời tiết trong lúc lấy mẫu, điều kiện độ sáng, môi trường có nhiều đối tượng khác tương tự như đối tượng quan tâm. Thứ 2, các biến thể nội tại trong đối tượng ta quan tâm (các biến đổi không bình thường của mẫu), sự khác nhau về khoảng cách lấy mẫu, mặt phẳng quan sát đối tượng trong lúc lấy mẫu ... đó là một trong những vô số nhập nhằng mà một hệ thống nhận dạng phải đối mặt. Thứ 3, chính là chất lượng thiết bị thu ảnh cũng như giới hạn khả năng tính toán của thiết bị xử lý nhận dạng. Để khử bớt phần nào tính nhập nhằng từ dữ liệu không có cách nào khác là phải tiền xử lý dữ liệu. Ở đây chúng tôi chọn phương pháp phân tích thành phần chính (PCA) để tiền xử lý và rút trích đặc trưng từ dữ liệu. PCA là một phương pháp rút trích đặc trưng tự động, không giám sát. Tính ưu việt của phương pháp xử lý này không những khử nhập nhằng tốt mà còn giảm được đáng kể khối lượng dữ liệu lưu trữ và xử lý cho hệ thống nhận dạng sau này).

Trước hết sử dụng PCA để rút ra các eigenvector (vector đặc trưng) hay các eigenface (khuôn mặt đặc trưng) của các ảnh khuôn mặt huấn luyện. Các khuôn mặt đặc trưng này cũng được dùng trong giai đoạn thử nghiệm. Chi tiết về cơ sở lý thuyết của PCA đã được trình bày (tham khảo lại phần phân tích và rút trích đặc trưng khuôn mặt bằng PCA).

Tập mẫu học trong đề tài bày bao gồm 30 lớp và mỗi lớp bao gồm 100 mẫu huấn luyện (Kỹ thuật nhân 20 mẫu nguyên thủy của mỗi lớp thành 100 mẫu bằng 4 phép xử lý ảnh cơ bản: Phép Mirror, tăng cường nhiễu, bộ lọc trung bình, bộ lọc Gamma). Kích thước khuôn mặt huấn luyện 30×30 sẽ được vector hoá thành một vector 900 chiều, ta có thể xem mỗi thành phần của vector này tương

ứng với một giá trị điểm ảnh trên ảnh gốc. Như vậy toàn bộ tập mẫu bao gồm 3000 vector và mỗi vector 900 chiều.

1. Xây dựng vector trung bình mẫu M 900 chiều
2. Xây dựng ma trận hiệp phương sai D (900×900) chiều.
3. Tính các giá trị đặc trưng và các vector đặc trưng tương ứng
 - + Với ma trận D (900×900) thì tương ứng 900 giá trị đặc trưng (eigenvalues).
 - + Ta có thể tạo ra các vector đặc trưng (eigenvectors) dựa trên các giá trị đặc trưng tương ứng. Trong cách tiếp cận của đề tài này và bằng thực nghiệm chúng tôi đã chọn 100 vector đặc trưng tương ứng với 100 giá trị đặc trưng lớn nhất.
 - + Một ma trận K mà mỗi cột là một vector đặc trưng 900 chiều. Vậy không gian ma trận K (900×100) chiều

Khoa CNTT - ĐH KHTN TP.HCM

4. Ánh xạ tập mẫu ban đầu vào không gian đặc trưng, sẽ hình thành không gian mẫu mới hay còn gọi là không gian mẫu đặc trưng
- + Lần lượt chiếu từng mẫu trong tập huấn luyện cũ $x_i = (x_i^1, x_i^2, \dots, x_i^{900})$ vào không gian đặc trưng $K_{900 \times 100}$ và kết quả phép chiếu này hình thành một vector mới $y_{i(1 \times 100)} = x_{i(1 \times 900)} * K_{(900 \times 100)}$
 - + Sau khi thực hiện phép chiếu này cho 3000 mẫu trong tập mẫu và kết quả thu được một tập mẫu mới (Tập mẫu học đặc trưng của Tập mẫu học ban đầu) $Y = \{y_i; i=1 \rightarrow 3000\}$ và trong đó $y_i = (y_i^1, y_i^2, \dots, y_i^{100})$.

Như vậy qua phép phân tích PCA, ta đã rút gọn số chiều của mẫu học từ 900 giảm xuống còn 100 chiều (đây chính là những thành phần quan trọng nhất của mẫu học, và chính các thành phần này tạo ra sự khác biệt cho các vector mẫu trong tập mẫu ban đầu)

5.2.2.1.4 Tạo các bộ phân loại nhị phân

Một lần nữa chúng tôi nhấn mạnh rằng, SVMs là một bộ phân loại nhị phân và có khả năng phân loại tốt với tính tổng quát cao từ tập mẫu dữ liệu được tổ chức thành hai lớp. Vì lẽ đó, SVMs được sử dụng rất phổ biến trong việc xử lý liên quan đến khuôn mặt đó là dò tìm khuôn mặt trên ảnh mà trong đó chỉ có duy nhất hai lớp: Lớp thuộc về khuôn mặt và lớp được gọi là không phải khuôn mặt người. Một chủ đề chính liên quan đến việc xử lý khuôn mặt, đó là bài toán nhận dạng khuôn mặt thì SVMs có cách giải quyết như thế nào?

Để huấn luyện hệ thống SVMs cho ứng dụng nhận dạng khuôn mặt người trên ảnh, chúng tôi đã sử dụng một cơ chế kết hợp nhiều bộ phân loại nhị phân SVMs. Tập mẫu khuôn mặt gồm nhiều người cần cho mục đích nhận dạng sẽ được biểu diễn trong máy tính và chúng ta thực hiện tác vụ gán nhãn cho từng lớp khuôn mặt theo thứ tự đã chọn (Tham khảo đến phần **mô tả dữ liệu nhận dạng**). Có nhiều phương pháp để nhị phân hoá tập mẫu khuôn mặt của nhiều người, nhưng hai cơ chế thông thường nhất là Một-đối-Một hay Một-đối-Tất cả. Một-đối-Tất cả là cơ chế tạo bộ mẫu nhị phân mà trong đó một bên là lớp khuôn mặt cần nhận dạng so với bên còn lại là tất cả các khuôn mặt của các lớp còn lại, như vậy tập mẫu của chúng ta gồm 30 người thì sẽ có tất cả 30 bộ phân loại nhị phân, nhưng theo thực nghiệm thì cơ chế này có nhiều khiếm khuyết, bởi vì ngay trong chính bản thân của mỗi bộ phân loại nhị phân có chứa sự hỗn tạp của rất nhiều lớp với nhau, nên siêu mặt có được từ dạng này không có khả năng phân tách mạnh. Một dạng tiếp cận khác theo chiến lược Một-đối-Một, có nghĩa lần

lược mỗi lớp khuôn mặt sẽ được so sánh với tất cả các khuôn mặt còn lại, như vậy với tập mẫu 30 người thì chúng tôi cần phải xây dựng $30(29-1)/2 = 435$ (bộ phân loại nhị phân). Với cách tiếp cận theo cơ chế này thì Siêu mặt tạo ra từ mỗi bộ phân loại nhị phân sẽ có khả năng phân loại tương minh giữa hai lớp khuôn mặt với nhau và đó cũng chính là phương pháp mà chúng tôi đã sử dụng trong đề tài.

5.2.2.1.5 Huấn luyện cho mỗi bộ phân loại nhị phân từ các tập

mẫu nhị phân hoá hai lớp khuôn mặt với nhau

- Đối với cách tiếp cận của đề tài chúng tôi đã sử dụng tập học với gồm $l = 200$ (ảnh khuôn mặt) để huấn luyện một bộ phân loại nhị phân phân biệt hai loại khuôn mặt khác nhau.

- Tập huấn luyện có dạng sau:

$$\mathbf{X} = \{(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)\}$$

x_i^d là mẫu thứ i trong tập học. $d \in (1..40)$

y_i là nhãn được gán cho mẫu i ($y_i \in \{-1, +1\}$).

Gọi $\alpha_1, \alpha_2, \dots, \alpha_l$ là tham số tương ứng với x_1, x_2, \dots, x_l trong tập học.

Xây dựng phương trình Siêu mặt \mathbf{H} : $w^* \cdot x + b = 0$.

Trong đó $w = \sum_i \alpha_i y_i x_i$ và $\sum_i \alpha_i y_i = 0$.

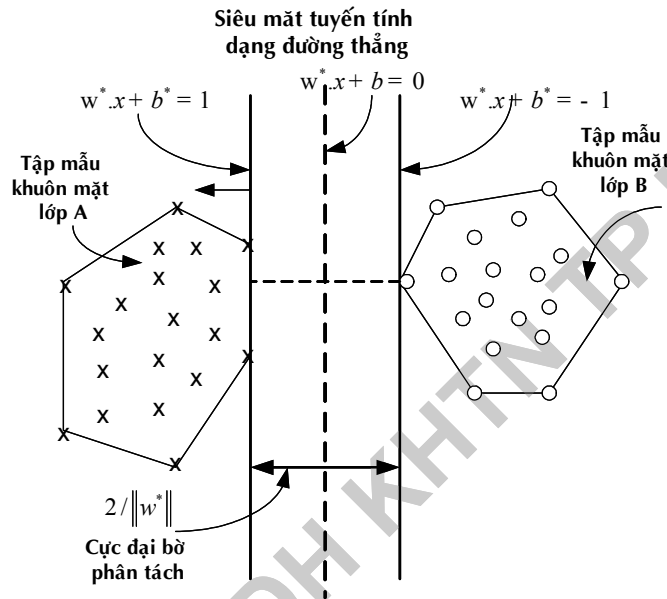
Sao cho thỏa bất đẳng thức sau: $y_i(x_i \cdot w + b) - 1 \leq 0, \forall i$.

- Nhưng để tính được w, b thì ta phải tính được a tham số đặc trưng cho mẫu khuôn mặt từ tập dữ liệu học ($a > 0$).
- Huấn luyện một bộ phân loại nhị phân với phương pháp SVM từ tập mẫu cho mục đích phân biệt các khuôn mặt của hai người khác nhau theo 3 bước sau đây:

- ✓ **Bước 1: Giai đoạn tìm các vector hỗ trợ (hay còn gọi các Vector Support) hay còn gọi đi xác định tham số a .**

Theo lý thuyết tổng quát về SVMs mà chúng tôi đã trình bày trên phần giới thiệu về cơ sở lý thuyết. Cấp độ và mức độ sử dụng SVMs để xử lý cũng khác nhau tùy thuộc vào mức độ phức tạp khác nhau của tập dữ liệu học.

Trong trường hợp nào chúng ta sử dụng bộ phân loại phi tuyến để xây dựng siêu phẳng phân lớp: Khi dữ liệu khuôn mặt của hai người cần phân loại được lưu trữ trên tập mẫu nhị phân hình thành hai tập Compact, có nghĩa rằng tập mẫu của hai khuôn mặt này có thể được phân loại bằng một hàm tuyến tính như trên Hình 5-10.



Hình 5-10 Mô phỏng phân lớp khuôn mặt giữa hai người bằng hàm tuyến tính

Trong trường hợp tập mẫu khuôn mặt của hai người cần phân loại có kiểu phân bố như dạng trên Hình 5-10 thì phương pháp tính các Vector hỗ trợ bằng cách sử dụng công thức như sau:

- Để thỏa công thức (5.6): $y_i(x_i w + b) - 1 \geq 0 \forall_i$
- Ràng buộc này được chuyển sang giải quyết bài toán phương trình Lagrangian theo công thức (5.7):

$$L_p = \frac{1}{2} \|w\|^2 - \sum_{i=1}^l \alpha_i y_i (x_i w + b) + \sum_{i=1}^l \alpha_i$$

- Để cực đại bờ phân tách $r = \frac{2}{\|w\|}$ cùng với các vector hỗ trợ

tim được thì phương trình L_p phải cực tiểu theo w, b và

L_p' đạo hàm bậc 1 là một phương trình theo a_i ($a_i > 0$).

$$\text{Ta có được: } \sum_i a_i y_i = 0 \quad (*)$$

- Theo tính chất hai mặt của phương trình Lagrangian thì tính cực tiểu L_p khó và theo công thức (5.10) thay vào đó ta sẽ tính cực đại theo dạng đối ngẫu L_D :

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j \text{ nên } L_D' \text{ là đạo hàm triệt tiêu}$$

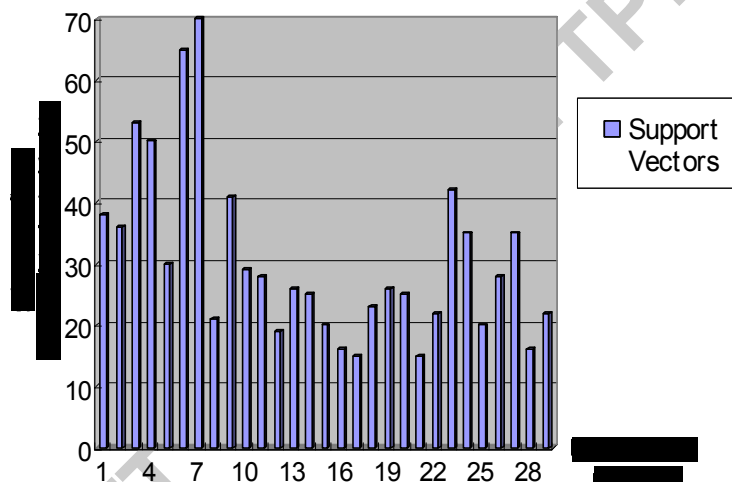
theo a_i .

- Vì L_D là một phương trình đa biến, để tính được L_D' bằng cách lấy đạo hàm riêng phần trên từng biến a_i . (**)
- Kết hợp các điều kiện (*) và (**) ta sẽ dễ dàng tính được tham số α_i .
- Các vector hỗ trợ được chọn tương ứng với mẫu nào thoả $\alpha_i > 0$.
- Kết quả của việc huấn luyện trên 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “Lớp 1” với các khuôn mặt của 29 người còn lại bằng SVM tuyến tính.

1/2→11	1/2	1/3	1/4	1/5	1/6	1/7	1/8	1/9	1/10	1/11
VS	38	36	53	50	30	65	70	21	41	29
1/12→21	1/12	1/13	1/14	1/15	1/16	1/17	1/18	1/19	1/20	1/21
VS	28	19	26	25	20	16	15	23	26	25
1/22→30	1/22	1/23	1/24	1/25	1/26	1/27	1/28	1/29	1/30	
VS	15	22	42	35	20	28	35	16	22	

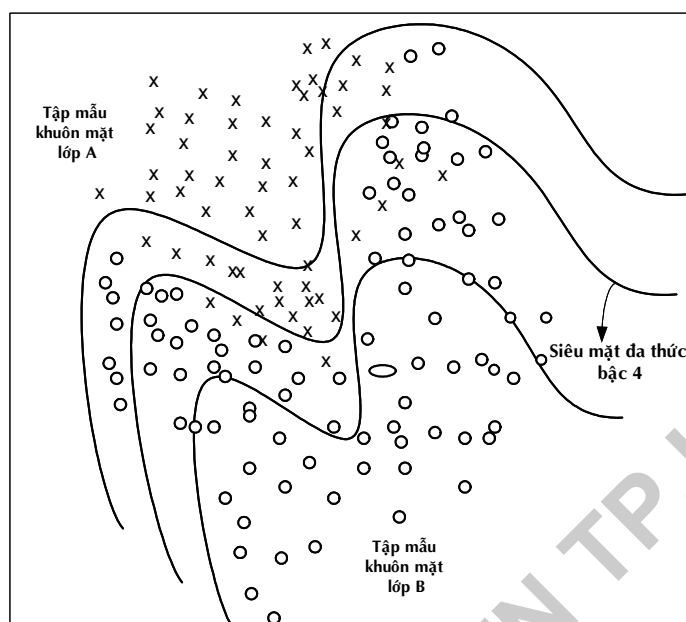
VS: Số vector support trên mỗi bộ phân loại nhị phân.

Bảng 5-1 Số Vector hỗ trợ tính được từ 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “lớp 1” với 29 lớp khuôn mặt khác.



Hình 5-11 Biểu diễn số liệu bảng 1 lên đồ thị

Nhưng trong trường hợp khác ta lại sử dụng bộ phân loại phi tuyến để xây dựng siêu mặt phân lớp (Tham khảo đến 4. Support Vector Machine phi tuyến) đó là khi dữ liệu khuôn mặt của hai người cần phân loại quá phức tạp: Do hai khuôn mặt quá giống nhau, hay bản thân mỗi khuôn mặt có quá nhiều biến đổi trong tập học, vì thế chúng ta mới sử dụng một số loại hàm phi tuyến (Phổ biến nhất vẫn là các loại hàm đa thức bậc 2,3 hay bậc 4) bởi vì chính các siêu mặt đa thức mới đủ khả năng len lỏi vào từng mẫu để tìm ra các đặc trưng cần thiết để phân lớp theo mong đợi như biểu diễn trên Hình 5-11.



Hình 5-12 Mô phỏng phân lớp khuôn mặt giữa hai người quá nhiều đặc trưng tương đương hay biến động.

Trong trường hợp tập mẫu khuôn mặt của hai người cần phân loại có kiểu phân bố như dạng trên Hình 5-12 thì phương pháp tính các Vector hỗ trợ bằng cách sử dụng công thức như sau:

- Đối với dữ liệu phân bố như trong trường hợp trên, chúng ta định nghĩa thêm một biến mới là x_i , $i = 1, \dots, l$ (l số mẫu trên tập nhị phân hoá)

- Theo công thức (5.34),(5.35),(5.36) thì ta sẽ có:

$$\mathbf{x}_i \mathbf{w} + \mathbf{b} \geq +1 - \xi_i, \quad \text{với } y_i = +1$$

$$\mathbf{x}_i \mathbf{w} + \mathbf{b} \leq -1 + \xi_i, \quad \text{với } y_i = -1$$

$$\xi_i \geq 0 \quad \forall i$$

Ràng buộc này được chuyển sang giải quyết bài toán phương trình Lagrangian theo công thức (5.41):

$$L_p = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i - \sum_i \alpha_i \{y_i(x_i \cdot \mathbf{w} + b) - 1 + \xi_i\} - \sum_i \mu_i \xi_i$$

- Như vậy, nếu một mẫu khuôn mặt bị phân loại lỗi thì $x_i >$

1, nên ta có thể xem $\sum_i \xi_i$ như là đường biên hay còn gọi

chặn trên của lỗi trên tập mẫu huấn luyện, như vậy chúng ta có thể xác định được chi phí tạo thêm lỗi nhằm đưa hàm

mục đích Lagrangian L_p tính cực tiểu từ dạng $\|\mathbf{w}\|^2/2$ sang

tính cực tiểu ở dạng $\|\mathbf{w}\|^2/2 + C(\sum_i \xi_i)^k$, và C do người

dùng chọn và C càng lớn thì tỉ lệ phân loại lỗi trên tập huấn luyện càng thấp.

- Dựa vào biến x_i mà chúng tôi có thể kiểm soát được mẫu

khuôn mặt nào bị phân loại lỗi xảy ra trong lúc huấn luyện bộ phân loại nhị phân bằng hàm tuyến tính.

+ Để khắc phục được các lỗi này, chúng ta không thể cố gắng khi xử lý SVMs với hàm xử lý tuyến tính. Nhất thiết phải xử lý SVMs với hàm xử lý phi tuyến.

+ Theo công thức (5.53) ta đặt Φ là hàm ánh xạ phi tuyến nhằm mục đích ánh xạ x_i vào không gian đặc trưng H:

$$\Phi: R^d \mapsto H$$

+ Ràng buộc trên phương trình Lagrangian

$$L_p = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i - \sum_i \alpha_i \{y_i(\Phi(x_i) \cdot \mathbf{w} + b) - 1 + \xi_i\} - \sum_i \mu_i \xi_i$$

- Để cực đại bờ phân tách $r = \frac{2}{\|\mathbf{w}\|}$ cùng với các vector hỗ trợ

tìm được thì phương trình L_p phải cực tiểu theo w, b và

L_p đạo hàm bậc 1 là một phương trình theo a_i ($a_i > 0$).

$$w = \sum_i a_i y_i F(x_i) \quad \text{và} \quad \sum_i a_i y_i F(x_i) F(x_j) = 0. \quad (***)$$

Việc tính toán cũng giống như trường hợp xử lý tuyến tính, theo tính chất hai mặt của phương trình Lagrangian thì tính cực tiểu L_p khó và thay vào đó ta sẽ tính cực đại theo dạng

đối ngẫu L_D :

$$\begin{aligned} L_D &= \sum_{i=1} a_i - 1/2 \sum_{i=1} \sum_{j=1} a_i a_j y_i y_j F(x_i) F(x_j) \\ &= \sum_{i=1} a_i - 1/2 \sum_{i=1} \sum_{j=1} a_i a_j y_i y_j K(x_i, x_j). \end{aligned}$$

nên L_D là đạo hàm triệt tiêu theo a_i . (****)

- Trong đó K là một trong những hàm xử lý chính được xác định trong công thức (5.66), công thức (5.68), công thức (5.69) hay công thức (5.70).

- Kết hợp (***) và (****) ta sẽ tính được a_i tương ứng cho mẫu thứ i .

1/2→11	1/2	1/3	1/4	1/5	1/6	1/7	1/8	1/9	1/10	1/11
VS	45	42	65	67	34	63	69	30	54	32
1/12→21	1/12	1/13	1/14	1/15	1/16	1/17	1/18	1/19	1/20	1/21
VS	33	29	39	32	22	25	22	37	37	36
1/22→30	1/22	1/23	1/24	1/25	1/26	1/27	1/28	1/29	1/30	
VS	27	25	55	52	41	40	51	25	40	

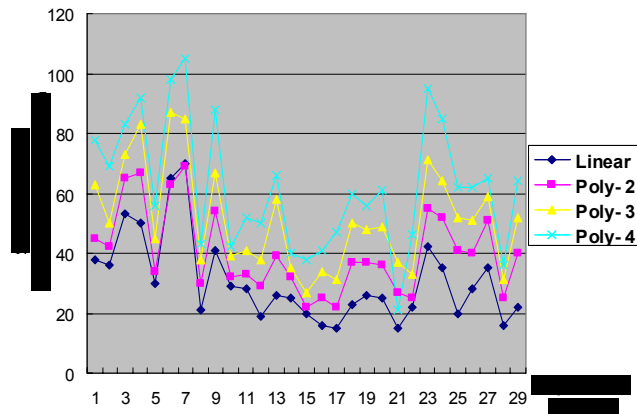
Bảng 5-2 Kết quả của việc huấn luyện từ 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “Lớp 1” với các khuôn mặt của 29 người còn lại bằng SVM phi tuyến có dạng đa thức bậc 2 (Poly-2).

1/2→11	1/2	1/3	1/4	1/5	1/6	1/7	1/8	1/9	1/10	1/11
VS	63	50	73	83	45	87	85	38	67	39
1/12→21	1/12	1/13	1/14	1/15	1/16	1/17	1/18	1/19	1/20	1/21
VS	41	38	58	35	27	34	31	50	48	49
1/22→30	1/22	1/23	1/24	1/25	1/26	1/27	1/28	1/29	1/30	
VS	37	33	71	64	52	51	59	31	52	

Bảng 5-3 Kết quả của việc huấn luyện từ 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “Lớp 1” với các khuôn mặt của 29 người còn lại bằng SVM phi tuyến có dạng đa thức bậc 2 (Poly-3).

1/2→11	1/2	1/3	1/4	1/5	1/6	1/7	1/8	1/9	1/10	1/11
VS	78	69	83	92	56	98	105	43	88	42
1/12→21	1/12	1/13	1/14	1/15	1/16	1/17	1/18	1/19	1/20	1/21
VS	52	50	66	40	38	41	47	60	56	61
1/22→30	1/22	1/23	1/24	1/25	1/26	1/27	1/28	1/29	1/30	
VS	21	46	95	85	62	62	65	36	64	

Bảng 5-4 Kết quả của việc huấn luyện từ 29 bộ phân loại nhị phân đầu tiên để phân biệt khuôn mặt “Lớp 1” với các khuôn mặt của 29 người còn lại bằng SVM phi tuyến có dạng đa thức bậc 2 (Poly-4).



Hình 5-13 Biểu diễn số liệu bảng 1(Linear), bảng 2(Poly-2), bảng 3(Poly-3), bảng 4 (Poly-4) trên cùng một đồ thị

Hình 5-13 cho thấy rằng ứng với mỗi loại phương trình xử lý SVMs khác nhau thì chúng sẽ có cách chọn ra các vector hỗ trợ phục vụ cho mục đích phân loại cũng khác nhau. Hầu hết các nguồn dữ liệu trong thế giới thực đều có dạng bản thân không thể phân loại được bằng tuyến tính và nhận dạng khuôn mặt cũng không phải là một ngoại lệ. Để chứng minh cho nhận định này chúng tôi sẽ khảo sát kỹ trong phần kết quả nhận dạng của phương pháp SVM.

✓ **Bước 2: Xây dựng siêu mặt phân loại căn cứ trên các mẫu gọi là các Vector hỗ trợ**

Sau khi đã tìm ra được số vector hỗ trợ cần thiết cho bộ phân loại nhị phân trong cả trường hợp hàm xử lý chính là dạng tuyến tính hay dạng đa thức bậc-2, bậc-3, bậc-4. Bây giờ chúng ta sẽ khảo sát cách xây dựng siêu mặt phân loại tập dữ liệu nhị phân từ các vector hỗ trợ mà hệ thống đã tìm được đó chính là các mẫu huấn luyện x_i trong tập học với tham số $\alpha_i > 0$.

- **Đối với trường hợp SVMs sử dụng hàm xử lý tuyến tính:**

$$w = \sum_i \alpha_i y_i x_i$$

$$H: w \cdot x + b = 0$$

$$H: \sum_{i=1}^{\text{sup port vectors}} \alpha_i y_i (x_i \cdot x) + b$$

Đây chính là phương trình siêu mặt mà chúng ta cần phải xây dựng từ SVMs sử dụng hàm xử lý chính dạng tuyến tính.

- **Đối với trường hợp SVMs sử dụng hàm xử lý phi tuyến**

Phải có cơ chế chuyển tập dữ liệu phân phối phi tuyến sang dạng phân phối tuyến tính trong không gian đặc trưng có số chiều cao H bằng một phép ánh xạ Φ . Khi đó w sẽ được tính có dạng sau đây:

$$w = \sum_i \alpha_i y_i F(x_i)$$

$$H: \sum_{i=1}^{\text{sup port vector}} \alpha_i y_i F(x_i) \cdot F(x) + b$$

$$H: \sum_{i=1}^{\text{sup port vector}} \alpha_i y_i K(x_i, x) + b$$

✓ **Bước 3: Xác định ngưỡng b**

Khi ta đã xác định được giá trị trọng số w tối ưu cho siêu mặt phân loại nhị phân dựa trên các mẫu khuôn mặt được gọi là các vector hỗ trợ thì việc tính ngưỡng b (hay còn gọi là hệ số bias) quá dễ dàng:

- **Đối với SVMs tuyến tính:**
 - Sau khi đã thỏa các ràng buộc tối ưu, theo công thức (5.15)

của điều kiện KKT ta có:

$$\alpha_i(y_i(\mathbf{w}x_i + \mathbf{b}) - 1) = 0 \quad \forall i$$

- Chọn một điểm mẫu bất kì x_i để thay vào công thức trên là có thể tính được b

$$b = \frac{1}{y_i} - \mathbf{w} \cdot x_i$$

- **Đối với SVMs phi tuyến**

- Theo công thức (5.49) của điều kiện KKT ta có:

$$\alpha_i \{y_i(x_i \cdot \mathbf{w} + b - 1 + \xi_i)\} = 0$$

- Chọn một điểm mẫu bất kì x_i để thay vào công thức trên là có thể tính được b

$$b = \frac{1 - x_i \cdot \mathbf{w}}{y_i}$$

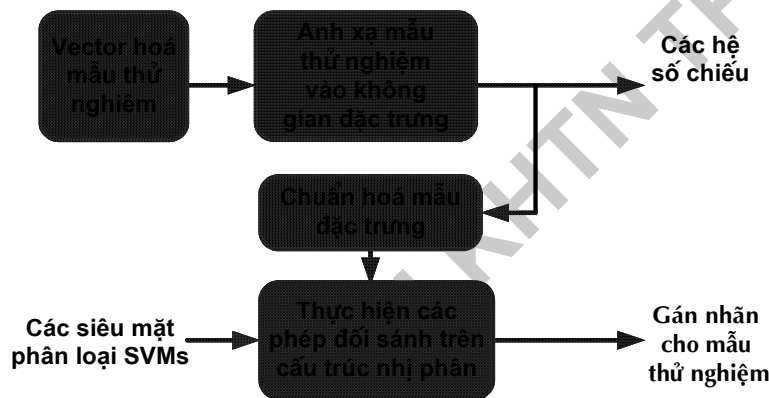
Khoa CNTT - ĐH KHTN TP.HCM

5.2.2.1.6 Khởi tạo kiến trúc cây nhị phân

- Căn cứ vào dữ liệu huấn luyện cho hệ thống nhận dạng khuôn mặt, Số người cần phân loại dựa trên khuôn mặt sẽ tương ứng với một vị trí Node lá trên cây nhị phân. Như vậy trong ứng dụng này ta sẽ có cây nhị phân với 30 (lá).
- Thứ tự của từng Node lá trên cây nhị phân cũng là thứ tự theo nhãn của mỗi lớp đã được gán trong phần mô tả dữ liệu cho hệ thống nhận dạng.

5.2.2.2 Giai đoạn nhận dạng khuôn mặt

5.2.2.2.1 Nhận dạng khuôn mặt dùng SVM



Hình 5-14 Các tác vụ nhận dạng khuôn mặt

5.2.2.2.2 Kỹ thuật nhận dạng khuôn mặt SVM

5.2.2.2.2.1 Vector hoá tập mẫu khuôn mặt thô

Thực hiện kỹ thuật Vector hoá mẫu thử nghiệm tương tự như trong giai đoạn huấn luyện 5.2.2.1.2

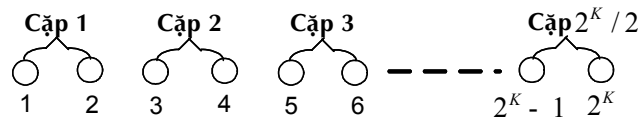
5.2.2.2.2.2 Rút trích đặc trưng khuôn mặt

Thực hiện kỹ thuật Rút trích đặc trưng từ mẫu thử nghiệm cũng thực hiện tương tự như trong giai đoạn huấn luyện 5.2.2.1.3

5.2.2.2.2.3 Đưa mẫu thử nghiệm khuôn mặt x vào cấu trúc nhị phân và thực hiện đối sánh trên từng mô hình nhị phân SVMs

- Cấu trúc cây nhị phân được mô phỏng tương tự như Hình 5-7.

- **Trường hợp 1:** Cơ chế so sánh với trường hợp số người cần nhận dạng là số mũ của 2. Mô phỏng trong Hình 5-15 bên trái.
 - ✓ Số lớp khuôn mặt cần phân loại có thể có $2, 4, 8, 16, 32, 64, \dots, 2^K$.
 - ✓ Ta thực hiện việc so sánh mẫu x theo từng cấp
 - ✓ Căn cứ vào số Node lá có được trên cấu trúc cây nhị phân, Chúng ta sẽ thực hiện nhóm hai lớp khuôn mặt liên tiếp nhau thành một cặp nhị phân và thứ tự nhóm từ trái qua phải và thông tin chứa trong mỗi cặp là nhãn của hai lớp tương ứng được nhóm.

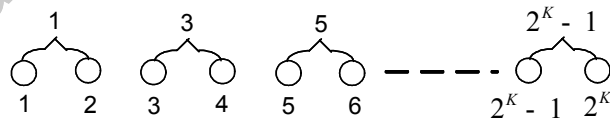


Hình 5-15 Mô phỏng cách ghép thành từng cặp nhị phân từ các Node lá của cây nhị phân

- ✓ Dựa vào thông tin gán nhãn cho mỗi lớp trong một cặp ta sẽ xác định được cặp này tương ứng với mô hình nhị phân SVMs nào đã xây dựng trong bước huấn luyện.
- ✓ Sau khi đã xác định mô hình SVMs nào tương ứng cho mỗi cặp được nhóm, kế tiếp chúng ta sẽ đưa mẫu thử nghiệm vào từng mô hình SVMs để phân loại. Trên mỗi bộ phân loại chúng ta xác định được mẫu thử nghiệm x gần với lớp nào nhất bằng hàm quyết định:

$$x \hat{I} \begin{cases} H_1 & \text{sign}(w \cdot x + b) \geq 1 \\ H_2 & \text{sign}(w \cdot x + b) < 1 \end{cases}$$

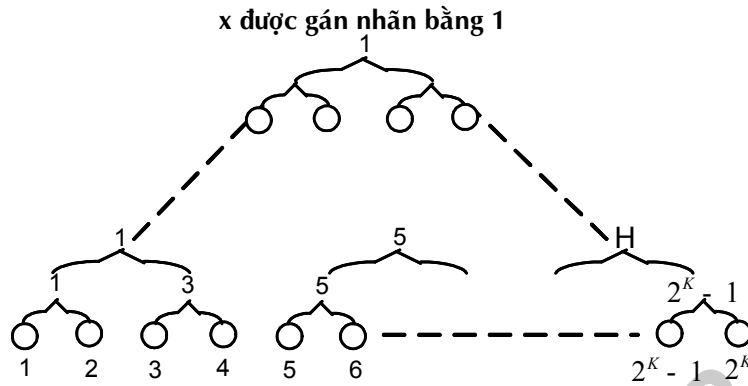
- ✓ Kết quả phân loại mẫu x trên $2^K / 2$ cặp đầu tiên giả sử như Hình 5-16 dưới đây:



Hình 5-16 Kết xuất phân loại mẫu x ở cấp 1.

- ✓ Số Lớp kết xuất từ các bộ phân loại nhị phân mà gần với mẫu khuôn mặt x nhất sẽ được tiếp tục sử dụng ở cấp thứ 2. Việc gom nhóm và tìm các lớp gần với x sẽ được thực hiện tương tự như cấp 1, Chúng ta sẽ thực hiện cho đến khi nào chỉ có một lớp duy nhất và lớp này nằm ngay

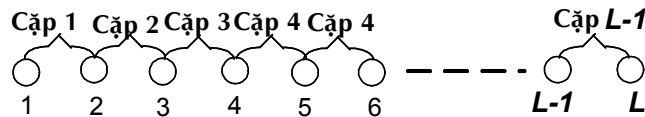
trên đỉnh của cấu trúc cây nhị phân và nhãn của nó sẽ được gán cho mẫu x , Kết quả trên Hình 5-17.



Hình 5-17 Kết quả mẫu x được nhận dạng với nhãn thuộc về khuôn mặt của người “Lớp1”.

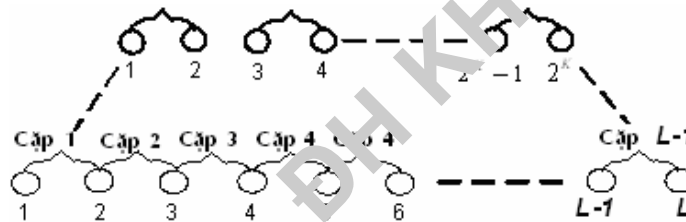
Khoa CNTT - ĐH KHTN TP.HCM

- ✓ **Nhận xét:** Để có được kết quả nhận dạng này hệ thống phải thực hiện $2^K - 1$ phép so sánh.
- **Trường hợp 2:** Số người cần nhận dạng trong tập học không bằng số mũ của 2. Mô phỏng trong Hình 5-17 bên phải.
 - ✓ Số lớp khuôn mặt cần phân loại không bằng số mũ của 2, ($l \neq 2^K$)
 - ✓ Trong trường hợp này chúng ta không chọn cách ghép nhóm như trên mà chọn cách ghép nhóm theo kiểu nối liên tục Hình 5-18.



Hình 5-18 Mô phỏng cách ghép thành từng cặp nhị phân từ các Node lá của cây nhị phân

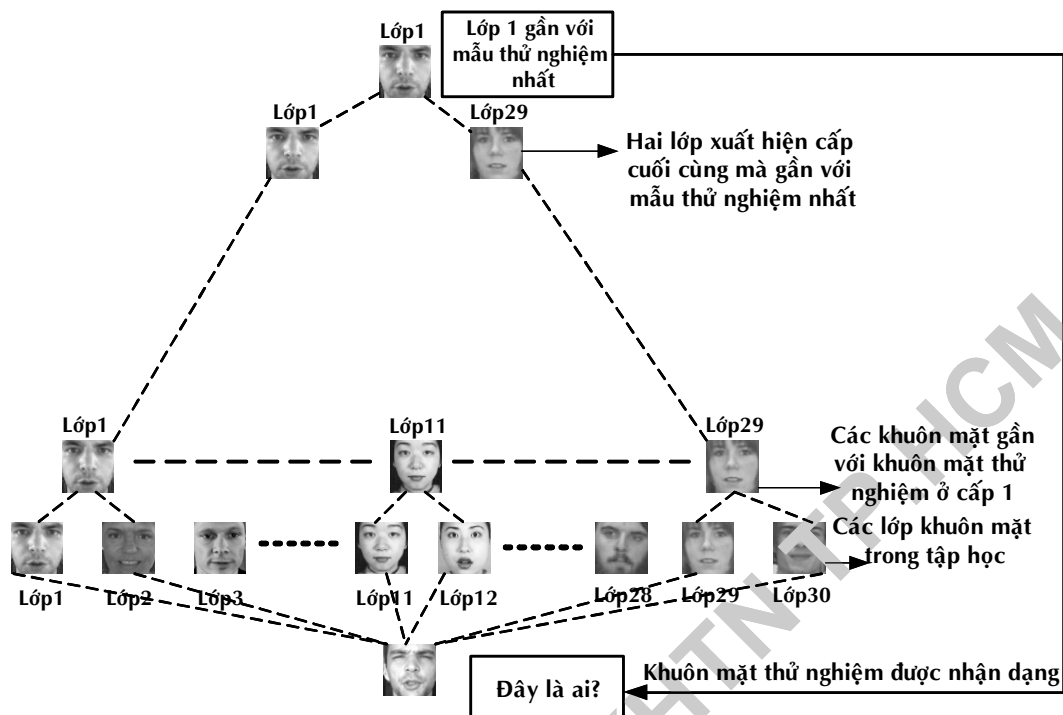
- ✓ Chúng ta sẽ thực hiện tương tự cho đến khi nào số nhãn xuất hiện trên cấu trúc của cây nhị phân bằng số mũ của 2, Hình 5-19.



Hình 5-19 Quá trình xây dựng cây nhị phân từ cấp có $L-1$ cặp đến cấp có $2^{L/2}$ cặp phân loại nhị phân.

- ✓ Từ bước này trở về cuối cách so sánh được thực hiện tương tự như trường hợp 1.
- ✓ **Nhận xét:** Để có được kết quả nhận dạng này hệ thống phải thực hiện $\frac{L(L+1)}{2} - \frac{2^K(2^K - 1)}{2} + 2^K - 1$ phép so sánh. Trong đó nhóm $\frac{L(L+1)}{2} - \frac{2^K(2^K - 1)}{2}$ là số phép so sánh từ cấp 1 đến cấp mà có số lớp xuất hiện bằng một số mũ của 2, và nhóm $2^K - 1$ là số phép so sánh còn lại để biết được nhãn của mẫu khuôn mặt x.

5.2.2.2.3 Mô phỏng quá trình nhận dạng khuôn mặt



Hình 5-20 Nhận dạng Mẫu thử nghiệm chưa được quan sát thuộc về Người 1 là đúng

Trong thí nghiệm này chúng tôi đã lấy một mẫu khuôn mặt chưa được quan sát trong tập học và kết quả chương trình nhận dạng đúng là khuôn mặt này thuộc về người có nhãn “Người-1” sau khi đi qua một chuỗi các phép so sánh trên các bộ phân loại nhị phân SVMs.

5.2.3 Nhận xét và hướng phát triển tương lai

5.2.3.1 Ưu điểm

- Xét về độ tin cậy trong xử lý, SVMs là một thuật toán mà bản thân vốn có khả năng xử lý tổng quát hoá tốt.
- Thuật toán dễ hiểu, dễ cài đặt và thực nghiệm, có tính thực tiễn cao trong việc giải quyết các bài toán trí tuệ nhân tạo và việc ứng dụng vào thương mại là có thể
- SVMs rất thích hợp cho bài toán phân tích, xử lý và khai thác dữ liệu dữ liệu nhờ vào khả năng tính toán và phân tích rất mạnh ví dụ: SVMs là công cụ để phân tích sự khác biệt, phân tích hồi quy, và ước lượng hàm mật độ.
- Trong bài toán nhận dạng khuôn mặt, SVM là một trong các thuật toán nhận dạng tốt, tỉ lệ nhận dạng cao với tỉ lệ chính xác trung bình trên 92%. Và nhờ khả năng xử lý tổng quát hoá tốt nên rất thích hợp cho bài toán nhận dạng khuôn mặt mà trong đó tính biến động của thông tin trên ảnh cũng như thông tin trên từng khuôn mặt trong thế giới thực luôn thay đổi từng giây từng phút.
- Thời gian huấn luyện hệ thống rất nhanh với 30 lớp khuôn mặt khác nhau và tập mẫu học gồm 3000 (mẫu) nhưng thời gian huấn luyện chỉ mất từ 1 phút đến 2 phút. Nếu với khối lượng dữ liệu như thế này thì huấn luyện nhanh nhất trên ANN cũng mất đến 10 phút và HMM cũng mất 5 phút.
- Đối với mỗi mô hình SVM, số lượng tham số trong mỗi mô hình lớn nên chúng ta sẽ có nhiều cách khác nhau trong việc điều khiển khả năng xử lý tối ưu cho thuật toán.

5.2.3.2 Khuyết điểm và hạn chế

- Một trong những khiếm khuyết lớn nhất của thuật toán SVMs chỉ có khả năng phân biệt tập học nhị phân, vì lẽ đó nhiều ứng dụng trong thực tế SVMs không thể đáp ứng được.
- Để ứng dụng cho phân loại nhiều lớp ta phải có nhiều cách tổ chức rất phức tạp và dẫn đến chi phí tính toán cũng rất cao.
- Bài toán SVM và việc xử lý chủ yếu dựa trên nền tảng các các mẫu dữ liệu gọi là các Vector hỗ trợ tìm được trước khi thực sự tìm siêu mặt, nếu số vector hỗ trợ quá nhiều thì dẫn đến chi phí cho thời gian đi tìm siêu mặt cũng như chi phí do giai đoạn nhận dạng là rất lớn. Vì vậy SVMs thường ít thành công trong việc xử lý bài toán với khối lượng dữ liệu tập học lớn.
- Bài toán nhận dạng người dựa trên thông tin khuôn mặt là một bài toán phức tạp trong thế giới thực, Số người cần huấn luyện cho mỗi hệ thống có thể lên đến hàng ngàn hay hàng triệu người và dữ liệu thu thập từ mỗi người có thể lên tới hàng trăm, hay hàng ngàn kiểu khác nhau, và đây là một hệ thống đòi hỏi khả năng chính xác cũng như đảm bảo khả năng xử lý thời gian thực của hệ thống.
- Cũng không ít các tính chất và định lý từ thuật toán SVMs chủ yếu xuất phát từ các nhận định là chính, chưa được chứng minh chặt chẽ trên phương diện toán học.
- Số khuôn mặt cần thiết cho mục đích nhận dạng còn thấp (chỉ mới 30 người) và tập mẫu học cho từng người cũng rất hạn chế, những mẫu khuôn mặt xuất hiện trong ảnh với những tình huống đặc biệt là rất ít, nếu trong ứng dụng thực tế mà gặp phải những trường hợp này thì có thể thất bại.

5.2.3.3 Những đề xuất và cải tiến

5.2.3.3.1 Về mặt thuật toán học

- Với cách tiếp cận hiện nay việc ứng dụng SVMs mà nền tảng là bộ phân loại nhị phân, quá trình huấn luyện trên từng bộ phân loại nhị phân là độc lập với nhau, và bên cạnh đó việc so sánh trên từng bộ phân loại cũng xảy ra độc lập trong quá trình thử nghiệm, vì vậy một cơ chế xử lý song song có thể cải thiện được đáng kể thời gian huấn luyện cho hệ thống cũng tốc độ trong lúc nhận dạng.
- Cần tìm hiểu và nghiên cứu nhiều về các thuật toán tối ưu để giảm chi phí

tính toán cho SVM trong giai đoạn huấn luyện cũng như trong giai đoạn thử nghiệm chương trình.

- Cần tìm hiểu về một số phát triển mới từ cho thuật toán SVM mà khi giải quyết bài toán nhiều lớp thì không cần phải thông qua các bộ phân loại nhị phân. Hiện nay có một số đã tiếp cận kết hợp SVMs và Fuzzy để xây dựng bộ phân loại đa lớp chuẩn.

5.2.3.3.2 Về mặt chương trình ứng dụng

- Cần tăng cường số lượng người cho mục đích nhận dạng và thu thập càng nhiều mẫu bổ sung cho tập học càng nhiều càng tốt.
- Cần huấn luyện cho hệ thống với nhiều bộ tham số khác nhau, để rút ra được quy luật xử lý SVM cho ứng dụng nhận dạng khuôn mặt và có thể tìm ra bộ tham số tối ưu nhất cho mô hình SVMs.

Khoa CNTT - ĐH KHTN TP HCM

Chương 6 MÔ HÌNH MAKOV ẨN VÀ ỨNG
DỤNG NHẬN DẠNG KHUÔN MẶT

Khoa CNTT - ĐH KHTN TP.HCM

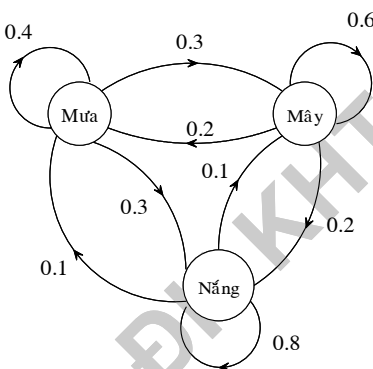
6.1 Giới thiệu mô hình Markov ẩn

Andrei A Markov (1895-1922) đã đưa ra một mô hình dùng để mô hình hoá chuỗi các sự kiện mà xác suất của sự kiện ở thời điểm hiện tại có liên quan đến sự xuất hiện của sự kiện ở thời điểm trước đó. Mô hình đó được gọi là chuỗi Markov.

6.1.1 Mô hình Markov

Chuỗi Markov có thể quan sát là chuỗi Markov mà kết xuất của tiến trình là một tập hợp các trạng thái ở mỗi thời điểm, trong đó mỗi trạng thái tương ứng với một biến cố có thể quan sát được.

Khảo sát mô hình Markov ba trạng thái dùng để biểu diễn tình trạng thay đổi thời tiết.



Hình 6-1 Mô hình Markov ba trạng thái biểu diễn thời tiết.

Giả sử mỗi ngày, thời tiết được quan sát sẽ rơi vào một trong ba trạng thái sau:

Trạng thái 1: Mưa

Trạng thái 2: Có mây

Trạng thái 3: Có nắng

Ma trận xác suất chuyển trạng thái:

$$A = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

Câu hỏi đặt ra: Xác suất để thời tiết tám ngày liên tiếp là {Nắng, Nắng, Nắng, Mưa, Mưa, Nắng, Mây, Nắng} là bao nhiêu?

Lời giải như sau:

Ta định nghĩa dãy quan sát O là:

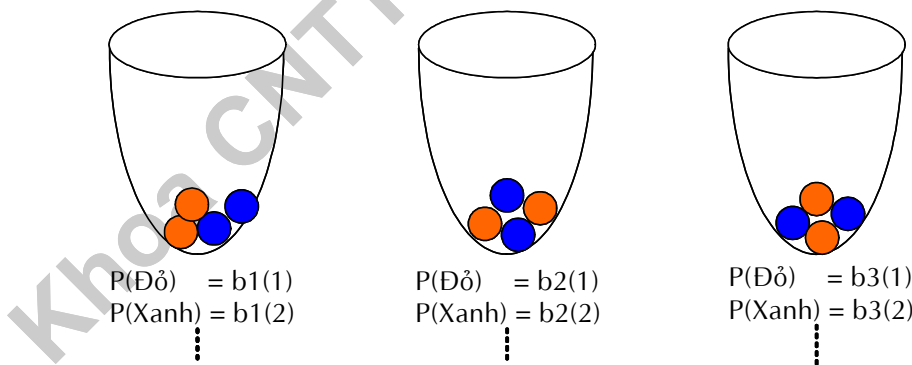
$$\begin{aligned}
O &= (\text{Nắng, Nắng, Nắng, Mưa, Mưa, Nắng, Mây, Nắng}) \\
&= (3, 3, 3, 1, 1, 3, 2, 3) \\
P(O) &= P[3, 3, 3, 1, 1, 3, 2, 3] \\
&= P[3]P[3|3]P[3|3]P[1|3]P[1|3]P[1|1]P[2|3]P[3|2] \\
&= \pi_3 \cdot (a_{33})^2 \cdot a_{31} \cdot a_{11} \cdot a_{13} \cdot a_{32} \cdot a_{23} \\
&= 1 \cdot 0.8 \cdot 0.8 \cdot 0.1 \cdot 0.4 \cdot 0.3 \cdot 0.1 \cdot 0.2 \\
&= 1.536 \cdot 10^{-4}
\end{aligned}$$

Với $\pi_i = P[q_1 = i]$, $1 \leq i \leq N$ là xác suất ban đầu của các trạng thái.

6.1.2 Mô hình Markov ẩn

Chuỗi Markov ẩn chính là một chuỗi Markov mà trong đó quan sát là một hàm xác suất theo trạng thái.

Giả sử trong một căn phòng có 3 bình (Hình 6-2), mỗi bình có vô số quả banh tô màu xanh, đỏ. Tiến trình để nhận được một dãy quan sát được thực hiện như sau: Một người trong phòng chọn ngẫu nhiên các banh trong các bình một cách ngẫu nhiên rồi đọc màu quả banh. Một người ở ngoài ghi lại chuỗi màu đó (ví dụ: Xanh, Đỏ, Xanh, Xanh, Đỏ...). Đó chính là chuỗi quan sát O. Coi mỗi trạng thái tương ứng với một bình. Vậy chuỗi biến đổi trạng thái đã bị ẩn trong chuỗi quan sát.



Hình 6-2 Mô phỏng mô hình Markov ẩn rời rạc bằng mô hình bình banh

Các thành phần của một mô hình Markov ẩn:

$\mathbf{q}: \mathbf{q} = \{q_1, \dots, q_t\} \in N^t$ là chuỗi biến đổi trạng thái giả định

N : Số trạng thái của mô hình mặc dù các trạng thái là ẩn.

M : Số đặc trưng phân biệt trên mỗi trạng thái

\mathbf{q}_t : là trạng thái tại thời điểm t .

$\mathbf{A} = \{\mathbf{a}_{i,j}\}$: không gian xác suất chuyển trạng thái của 2 thời điểm kế tiếp nhau

$\mathbf{a}_{i,j} \in \mathbf{A}$: là xác suất chuyển trạng thái \mathbf{q}_i tại thời điểm t sang \mathbf{q}_j tại thời điểm $t+1$

$\pi_i = P[q_1 = i]$: xác suất ban đầu của trạng thái i

$\pi = \{\pi_i\}$: tập các xác suất trạng thái đầu

$\mathbf{o}_t = v_t$: là quan sát được v_t tại thời điểm t

$\mathbf{O} = \{\mathbf{o}_1, \dots, \mathbf{o}_T\}$: Chuỗi quan sát theo thời gian t

\mathbf{B} : Ma trận phân bố xác suất của các đặc trưng khi khảo sát trên từng trạng thái

$\mathbf{b}_i(\mathbf{k})$: Xác suất nhận được đặc trưng v_k ở trạng thái i .

Ta dùng kí hiệu: $\lambda = (A, B, \pi)$ để chỉ tập tham số của mô hình.

6.1.2.1 Xác suất của chuỗi quan sát

Để đánh giá xem mô hình $\lambda = (A, B, \pi)$ được cho có đối sánh tốt với dãy quan sát $O = (o_1, o_2, \dots, o_T)$ hay không, ta cần tính $P(O | \lambda)$. Cách tính như sau:

Chọn ra một dãy trạng thái $q = (q_1, q_2, \dots, q_T)$ trong số N^T dãy trạng thái

$$\text{Khi đó } P(O | q, \lambda) = \prod_{t=1}^T P(o_t | q_t, \lambda) \quad (6.1)$$

Giả sử các quan sát là độc lập thống kê:

$$P(O | q, \lambda) = b_{q_1}(o_1) b_{q_2}(o_2) \dots b_{q_T}(o_T) \quad (6.2)$$

Xác suất dãy trạng thái q

$$P(q | \lambda) = \pi_{q_1} a_{q_1 q_2} \dots a_{q_{T-1} q_T} \quad (6.3)$$

$$\text{Mà ta có } P(O, q | \lambda) = P(O | q, \lambda) P(q | \lambda) \quad (6.4)$$

$$\text{Do đó } P(O | \lambda) = \sum_q P(O, q | \lambda) \quad q \in N^T \quad (6.5)$$

Có N^T dãy trạng thái và $O(T)$ phép tính, nên độ phức tạp của thuật toán bằng $O(N^T T)$. Để giảm độ phức tạp, thuật toán được cải tiến bằng 2 thủ tục: Thủ tục lùi và thủ tục tiến.

6.1.2.1.1 Thủ tục tiến

Gọi $\alpha_t(i)$ là xác suất chuỗi quan sát đến thời điểm t $o_1 o_2 \dots o_t$ và trạng thái i ở thời điểm t . $\alpha_t(i)$ còn được gọi là biến tiến

$$\alpha_t(i) = P[o_1 \dots o_t, q_t = i | \lambda] \quad (6.6)$$

Có thể tính $\alpha_t(i)$ bằng cách quy nạp:

- B1: Khởi tạo

$$\alpha_t(i) = \pi_i b_i(o_1) \quad 1 \leq i \leq N \quad (6.7)$$

- B2: Quy nạp

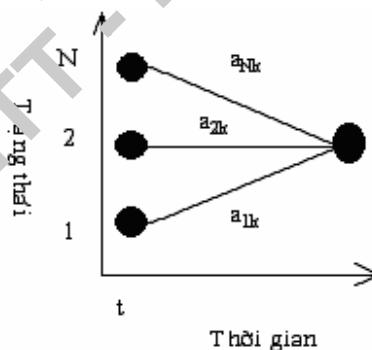
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(o_{t+1}) \quad , 1 \leq t \leq T-1, 1 \leq j \leq N \quad (6.8)$$

- B3: Kết thúc

$$P(O | \lambda) = \sum_{i=1}^N \alpha_T(i) \quad (6.9)$$

- B4: Đánh giá

Độ phức tạp: $O(N^2T)$



Hình 6-3 Tính toán theo thủ tục tiến ở một thời điểm

6.1.2.1.2 Thủ tục lùi

Gọi $\beta_t(i)$ biểu diễn xác suất của dãy quan sát riêng phần từ thời điểm $t+1$ cho tới kết thúc chuỗi quan sát, và trạng thái i trên chuỗi quan sát tại thời điểm t . $\beta_t(i)$ được gọi là biên lùi.

$$\beta_t(i) = P[o_t \dots o_T | q_t = i, \lambda] \quad (6.10)$$

Có thể tính $\beta_t(i)$ bằng cách quy nạp.

- B1: Khởi tạo

$$\beta_T(i) = 1, 1 \leq i \leq N \quad (6.11)$$

- B2: Quy nạp

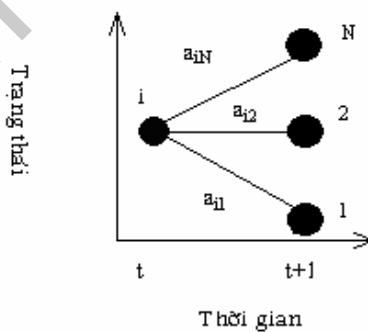
$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), t = T-1, T-2, \dots, 1, 1 \leq i \leq N \quad (6.12)$$

- B3: Kết thúc:

$$P(O | \lambda) = \sum_{i=1}^N \pi(i) b_i(o_1) \beta_1(i)$$

- B4: Đánh giá

Độ phức tạp: $O(N^2T)$



Hình 6-4 Tính toán theo thủ tục lùi ở một thời điểm

6.1.2.2 Dãy trạng thái tối ưu

Để tìm chuỗi trạng thái \mathbf{q} giải thích tốt nhất cho chuỗi quan sát \mathbf{O} , tức cần tìm $\mathbf{q} \in \mathbb{N}^t$ sao cho $P(\mathbf{O}, \mathbf{q} / \lambda_{A,B,\pi})$ lớn nhất, ta có thể dùng thuật toán Baum-Welch hay thuật toán Viterbi.

Thuật toán Baum-Welch

Gọi $\gamma_t(i) = P(q_t = i | \mathbf{O}, \lambda)$ là xác suất trạng thái i ở thời điểm t trên chuỗi quan sát \mathbf{O} và mô hình λ .

Ta có thể biểu diễn:

$$\begin{aligned}\gamma_t(i) &= P(q_t = i | \mathbf{O}, \lambda) \\ &= \frac{P(\mathbf{O}, q_t = i | \lambda)}{P(\mathbf{O} | \lambda)} \\ &= \frac{P(\mathbf{O}, q_t = i | \lambda)}{\sum_{j=1}^N P(\mathbf{O}, q_t = j | \lambda)}\end{aligned}\quad (6.13)$$

Vì $P(\mathbf{O}, q_t = j | \lambda) = \alpha_t(j)\beta_t(j)$ nên công thức trên được tính lại như sau:

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^N \alpha_t(j)\beta_t(j)}\quad (6.14)$$

Trong đó $\alpha_t(i)$ giải thích dãy riêng phần $o_1 o_2 \dots o_t$ và trạng thái i ở thời điểm t , còn $\beta_t(i)$ giải thích dãy quan sát còn lại $o_{t+1} o_{t+2} \dots o_T$ được cho bởi trạng thái i tại thời điểm t .

Dùng $\gamma_t(i)$, ta có thể tìm được trạng thái q_t^* phù hợp nhất ở từng thời điểm t :

$$q_t^* = \arg \max_{1 \leq i \leq N} [\gamma_t(i)], \quad 1 \leq t \leq T\quad (6.15)$$

Phương trình trên cực đại hoá kỳ vọng số các trạng thái đúng, tuy nhiên khi HMM có một số xác suất chuyển trạng thái bằng không ($a_{ij} = 0$) thì dãy trạng thái tối ưu

đó không còn hợp lệ nữa. Vì phương trình trên chỉ xác định trạng thái phù hợp nhất ở từng thời điểm mà không chú ý đến sự xuất hiện chung của dãy trạng thái.

Thuật toán Viterbi:

Gọi $\delta_t(i)$ là tỉ số tốt nhất theo đường đi đơn ở thời điểm t mà nó giải thích được t quan sát đầu tiên và kết thúc ở trạng thái i .

$$\delta_t(i) = \max_{q_1 q_2 \dots q_{t-1}} P(q_1 q_2 \dots q_{t-1}, q_t = i, o_1 o_2 \dots o_t | \lambda) \quad (6.16)$$

Có thể tính được $\delta_{t+1}(i)$ bằng cách quy nạp

$$\delta_{t+1}(j) = \max_i [\delta_t(i) a_{ij}] b_j(o_{t+1}) \quad (6.17)$$

B1: Khởi tạo

$$\delta_1(i) = \pi_i b_i(o_1) \quad 1 \leq i \leq N \quad (6.18)$$

$\Psi_1(i) = 0$ (Lưu vết chuỗi Markov)

B2: Tính đệ quy

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(o_t), \quad 2 \leq t \leq T; \quad 1 \leq j \leq N \quad (6.19)$$

$$\Psi_t(j) = \arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] \quad (6.20)$$

B3: Kết thúc

$$P^* = \max_{1 \leq i \leq N} [\delta_T(i)] \quad (6.21)$$

$$q_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)] \quad (6.22)$$

B4: Lần ngược lại đường đi (dãy trạng thái)

$$q_t^* = \Psi_{t+1}(q_{t+1}^*), \quad t = T-1, T-2, \dots, 1 \quad (6.23)$$

6.1.2.3 Hiệu chỉnh các tham số của mô hình

Để hiệu chỉnh các tham số của mô hình sao cho $\lambda_{A,B,\pi}(O)$ đạt cực đại ta có thể sử dụng phương pháp Segmental K-mean: điều chỉnh $P(O|\lambda)$ bằng cách làm cực đại giá trị $P(O,q|\lambda)$ với q là chuỗi trạng thái tối ưu được tìm bằng thuật toán Viterbi.

Khoa CNTT - ĐH KHTN TP.HCM

6.2 ỨNG DỤNG MÔ HÌNH MARKOV ẨN NHẬN DẠNG KHUÔN MẶT NGƯỜI

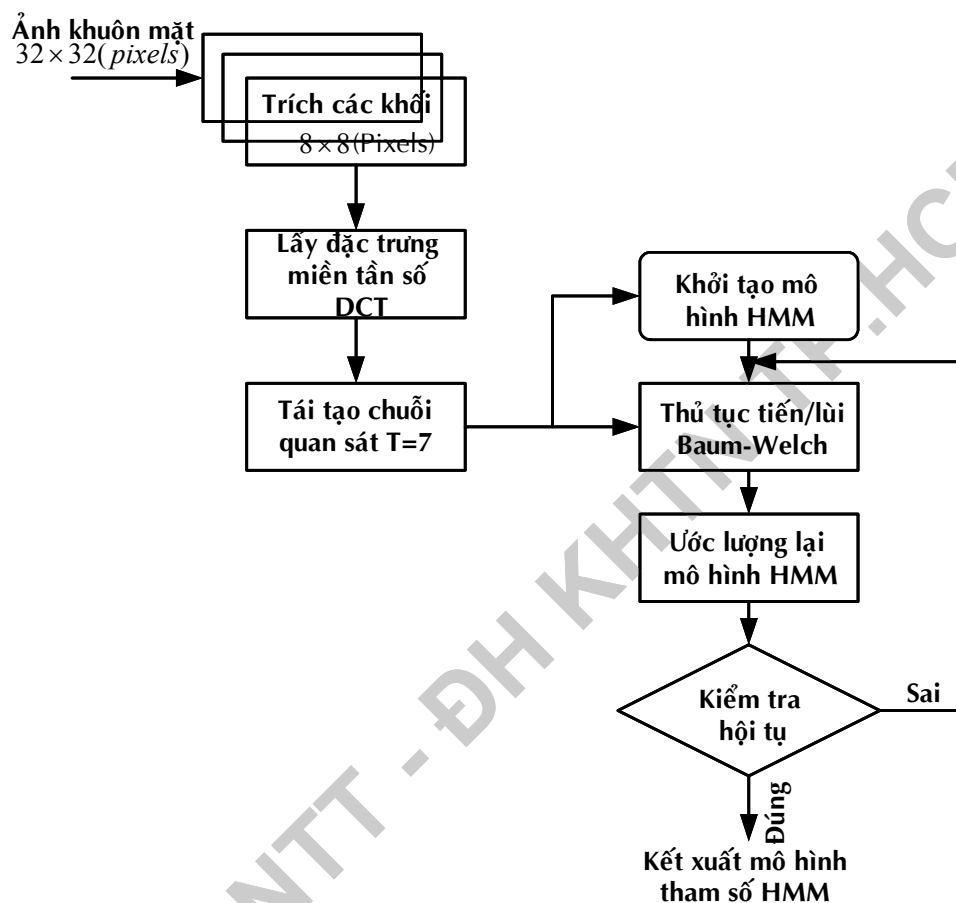
6.2.1 Ý tưởng

Với quan niệm về khuôn mặt thẳng như sau: Các đặc trưng thuộc về khuôn mặt được coi là quan trọng nhất như tóc, vầng trán, hai mắt, mũi, miệng, cằm, xuất hiện rất tự nhiên theo thứ tự ngữ nghĩa khi xét từ trên xuống dưới. Vì vậy, các miền đặc trưng thuộc về khuôn mặt có thể được mô hình hóa thành chuỗi quan sát theo thứ tự xuất hiện của các đặc trưng cũng như thứ tự quan sát về mặt thời gian. Từ đó chúng tôi quyết định tiếp cận theo phương pháp mô hình Markov ẩn cho bài toán nhận dạng khuôn mặt.

Khoa CNTT - ĐH KHTN TP HCM

6.2.2 Nhận dạng khuôn mặt bằng mô hình Markov ẩn

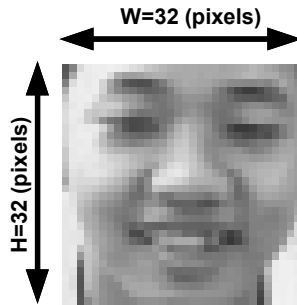
6.2.2.1 Giai đoạn huấn luyện hệ thống



Hình 6-5 Huấn luyện khuôn mặt bằng mô hình Markov ẩn rời rạc.

6.2.2.1.1 Ảnh khuôn mặt huấn luyện

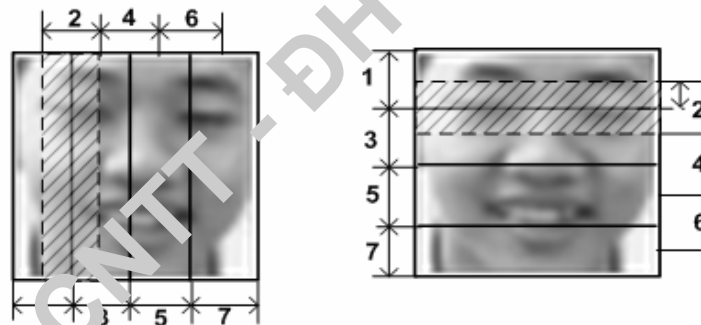
Tương tự như tập mẫu khuôn mặt đã sử dụng từ phương pháp học SVM, nhưng lại sử dụng kích thước ảnh chuẩn là 32×32 (pixels). Nguyên nhân của sự lựa chọn này được trình bày chi tiết trong phần rút trích đặc trưng từ ảnh khuôn mặt sử dụng trong mô hình Markov ẩn rời rạc. Chi tiết về mẫu khuôn mặt được biểu diễn trong Hình 6-6 như sau:



Hình 6-6 Mẫu khuôn mặt cho việc huấn luyện mô hình Markov ẩn rời rạc với kích thước chuẩn 32x32 (pixels)

6.2.2.1.2 Biểu diễn dữ liệu khuôn mặt theo mô hình Markov

Để giải quyết bài toán theo mô hình Markov ẩn rời rạc ta chia mỗi mẫu khuôn mặt trong tập học có chiều cao kí hiệu là H và chiều rộng kí hiệu là W ra thành nhiều khối nhỏ, mỗi khối có kích thước chiều cao P , chiều rộng vẫn bằng W . Các khối này có thể liên tiếp hay chồng lấp lên nhau. Hình 6-7 minh họa dưới đây:



P : Chiều cao của khối

L : Phần chồng lấp của hai khối liên tiếp

Hình 6-7 Tách mẫu huấn luyện $H \times W$ thành một chuỗi các khối con $P \times W$.

Mẫu khuôn mặt sẽ được lượng tử hoá thành các khối quan sát liên tiếp nhau. Hình bên trái sử dụng chiến lược lấy khối theo chiều ngang và mỗi khối được lấy bằng cách dịch chuyển theo thứ tự từ trái sang phải (Left to Right), hình bên phải sử dụng chiến lược lấy khối theo chiều chiều cao và mỗi khối được lấy bằng cách dịch chuyển theo thứ tự từ trên xuống dưới (Top to Bottom), trong cách tiếp cận này các khối chồng lấp lên nhau một khoảng L .

Gọi T là số khối lượng mã hoá từ mẫu khuôn mặt, và quy tắc để tính T

$$T = \frac{H-P}{P-L} + 1$$

Như vậy với kích thước mẫu $H = 32$, $P = 8$, $L = P/2 = 4$, thì $T = [(32-8)/(8-4)]+1 = 7$ (khối).

Hai chiến lược tách khối cho mô hình Markov ẩn rời rạc mang ý nghĩa khác nhau đối với cùng một khuôn mặt.

- ✓ Sử dụng chiến lược tách theo chiều ngang như trong Hình 6-8, có nghĩa lúc này ta quan niệm mỗi khối đều mang thông tin khá đầy đủ về khuôn mặt, thứ tự xuất hiện đồng thời các đặc trưng của một khuôn mặt trong mỗi khối theo thứ tự từ trên xuống có thể có như {Chân mày, mắt, má, miệng, cằm} hay cũng có thể {Chân mày, Mắt, Mũi, Miệng, Cằm}.
- ✓ Sử dụng chiến lược tách khối theo chiều đứng như trong Hình 6-9, có nghĩa lúc này ta quan niệm mỗi khối chỉ chứa đầy đủ về một đặc trưng cục bộ của khuôn mặt, và sự phân bố có tính đối xứng của các đặc trưng bên trong mỗi khối cũng có một ý nghĩa rất quan trọng, thứ tự xuất hiện các đặc trưng của khuôn mặt trong mỗi khối theo thứ tự từ trên xuống dưới lần lượt như sau {Hai chân mày } → {Hai mắt} → {Má, mũi, má} → {Miệng} → {Cằm}.

Gọi K là tập hợp các khối $K = \{k_1, k_2, k_3, k_4, k_5, k_6, k_7\}$ được tách ra từ một mẫu khuôn mặt, chỉ số đánh trên mỗi khối của cả hai trường hợp theo thứ tự từ $1 \rightarrow 7$ cũng là thứ tự xuất hiện của chuỗi quan sát trong mô hình Markov ẩn. Và mỗi khối được xem là một quan sát tại một thời điểm t ($t = 1, \dots, 7$). Nhiệm vụ của ta bây giờ là phải dạy cho máy tính biết được cách giải thích chuỗi quan sát trên bằng phương pháp ước lượng mô hình trên quan điểm xác suất thống kê.

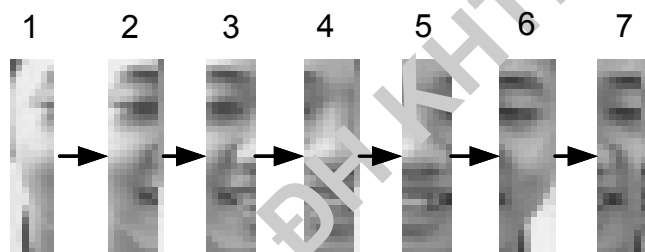
Gọi O là chuỗi các khối dữ liệu tách ra từ một khuôn mặt, ta có:

$$O = \{o_1, o_2, o_3, o_4, o_5, o_6, o_7\}$$

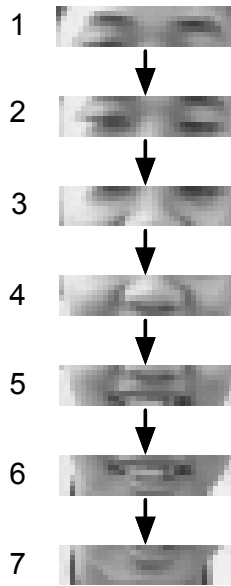
Trong đó mỗi quan sát o_i^d là một vector d chiều nhận được từ khối k_i có

kích thước $P \times W$ và $d = P \times W$ (pixels).

Kết quả tách khối theo hai chiều lược như sau:



Hình 6-8 Mẫu khuôn mặt sẽ được tách thành 7 khối theo thứ tự từ trái sang phải với mỗi khối là 32×8 (pixels)



Hình 6-9 Mẫu khuôn mặt được tách thành 7 khối theo thứ tự từ trên xuống dưới với mỗi khối là 32x8(pixels)

6.2.2.1.3 Kỹ thuật trích đặc trưng trên mẫu khuôn mặt

Nếu giá trị chuỗi quan sát là giá trị của điểm ảnh được trích ra trực tiếp từ các khối trên một ảnh mẫu khuôn mặt thì gặp phải hai bất lợi lớn: Thứ nhất, giá trị điểm ảnh từ ảnh chứa khuôn mặt không biểu diễn tốt được các đặc trưng chủ đạo của khuôn mặt, đặc trưng từ các giá trị này rất nhạy cảm với ảnh nhiễu, khuôn mặt bị quay, khuôn mặt bị che lấp, khuôn mặt bị dịch chuyển, biến đổi lớn từ nội tại của khuôn mặt do biểu hiện cảm xúc quá mạnh. Thứ hai, mỗi vector quan sát o_i^d có thể có số chiều rất lớn, $d = 32 \times 8 = 256$, điều này dẫn đến chi phí tính toán của hệ thống rất lớn ngay cả giai đoạn huấn luyện cũng như giai đoạn thử nghiệm.

Vì vậy rút trích đặc trưng là một tác vụ vô cùng quan trọng nhằm lấy ra những tri thức cô đọng, chính xác và ổn định nhất, đồng thời loại bỏ những thông tin dư thừa không cần thiết. Trong cách tiếp cận theo mô hình Markov ẩn để nhận dạng khuôn mặt ta chọn phương pháp rút trích đặc trưng hình học, không giám sát, đó là phương pháp biến đổi Cosine rời rạc hai chiều.

Để mô phỏng phương pháp rút trích đặc trưng và tái tạo thành vector quan sát, ta chọn chiến lược trích khối theo chiều đứng như Hình 6-9, và chọn khối đầu tiên k_1 để minh họa xử lý rút trích đặc trưng và sẽ tái tạo vector quan sát đầu tiên

o_1 trong chuỗi quan sát O. Bao gồm 5 bước chính sau đây:

Bước 1: Tách khối 8×32 (pixels)



Hình 6-10 Khối đầu tiên trong 7 khối cần được lượng hoá thành vector quan sát.

Bước 2: Tiếp tục tách khối 8×32 (pixels) thành 4 khối con 8×8 (pixels)



Hình 6-11 Tách khối 8×8 (pixels)

Bước 3: Chuẩn hoá từng điểm ảnh trên mỗi khối về các giá trị trong khoảng $[0,1]$ bằng công thức sau:

$$\text{Giá trị chuẩn hoá} = \frac{I(x,y)}{255}$$

Bước 4: Thực hiện tính các hệ số DCT bằng phép biến đổi Cosine rời rạc hai chiều trên mỗi khối 8×8 , và sử dụng dạng một của Zigzag (Hình 4-4) để lấy 20 hệ số DCT đầu tiên trên mỗi khối biến đổi đó.

Bước 5: Tái tạo vector quan sát từ các hệ số DCT trên mỗi khối con 8×8 (pixels)

Tái tạo vector quan sát đầu tiên o_1 từ khối quan sát đầu tiên k_1

✓ Gọi S là tập mẫu đặc trưng thu được từ k_1

$$S = \{s_1, s_2, s_3, s_4\}$$

Trong đó $s_i = (s_i^1, s_i^2, s_i^3, s_i^4, \dots, s_i^{19}, s_i^{20})$

✓ Từ kết quả biến đổi Cosine rời rạc hai chiều lần lượt trên 4 khối 8×8 (pixels) ở Bước 4 sử dụng đường quét Zigzag dạng một để chuyển 20 hệ số DCT đầu tiên trên mỗi bảng về vector một chiều, chú ý loại bỏ hệ số DC trên mỗi bảng.

$$s_1 = (0.206, 0.013, -0.18, -0.67, -0.666, -0.197, \dots, -0.085, -0.094, -0.052, 0.188, 0.082, 0.05)$$

$$s_2 = (-0.043, 0.617, 0.354, 0.044, 0.2602, 0.0063, \dots, -0.018, -0.042, -0.0366, -0.041, 0.0466, -0.049)$$

$$s_3 = (0.3947, 0.4642, 0.263, -0.1318, -0.0063, -0.0417, \dots, -0.0043, 0.0126, -0.0377, 0.027, 0.15, -0.062)$$

$$s_4 = (-0.078, 0.156, 0.563, 0.115, -0.007, -0.021, \dots, 0.0086, -0.036, -0.029, 0.054, -0.21, -0.059)$$

Vậy vector quan sát thứ nhất o_1 tương ứng với khối quan sát thứ nhất k_1 như sau:

$$o_1 = (\{s_1\}, \{s_2\}, \{s_3\}, \{s_4\})$$

$$= (s_1^1, s_1^2, \dots, s_1^{20}, s_2^1, s_2^2, \dots, s_2^{20}, s_3^1, s_3^2, \dots, s_3^{20}, s_4^1, s_4^2, \dots, s_4^{20})$$

$$= (o_1^1, o_1^2, \dots, o_1^i, \dots, o_1^{80})$$

Như vậy vector quan sát đầu tiên o_1 trong chuỗi quan sát O đó là vector chứa các hệ số DCT quan trọng nhất trong mỗi khối con 8×8 được trích ra từ khối quan sát đầu tiên k_1 . Đó là các hệ số chứa năng lượng của tín hiệu nhiều nhất và có ý nghĩa nhất.

6.2.2.1.4 Huấn luyện HMM

Sau khi thu thập 3000 mẫu cho tập học của 30 người khác nhau, mỗi người gồm 100 mẫu khuôn mặt có độ biến động tương đối phức tạp. Để giải quyết bài toán nhận dạng từ 30 người khác nhau này ta phải xây dựng 30 mô hình Markov ẩn rời rạc nhằm mục đích mô hình hóa cho các khuôn mặt của những người khác nhau.

Cấu trúc tham số của một mô hình Markov ẩn rời rạc (λ) cho tác vụ huấn luyện có dạng:

- ✓ Số trạng thái trên một mô hình Markov ẩn rời rạc được chọn $N = 10$
- ✓ Số thuộc tính cần quan sát trên mỗi trạng được chọn $M = 6$
- ✓ Chuỗi vector quan sát cho mỗi tiến trình huấn luyện $T = 7$

$$O = (o_1, o_2, o_3, o_4, o_5, o_6, o_7)$$

- ✓ Số chiều của mỗi quan sát o_i^d với $d = 80$.

Tác vụ học từ dữ liệu:

- ✓ **Bước 1:** Khởi tạo mô hình Markov ẩn rời rạc
 - + Mô hình $\lambda(A, B, \pi)$ được khởi tạo.
 - + Mẫu dữ liệu huấn luyện được lượng hoá thành chuỗi vector quan sát trên 7 thời điểm khác nhau và mỗi vector quan sát là 80 chiều.
 - + Biến cố xảy ra trong mỗi trạng thái là không thể xác định được, chúng ta chỉ có thể ước lượng bằng tính trung bình xác suất.
 - + Vector quan sát đầu tiên trong chuỗi quan sát sẽ được sử dụng để khởi tạo giá trị cho ma trận chuyển trạng thái A, và ma trận phân phối xác suất các ký hiệu quan sát trên mỗi trạng thái B.

+ Khởi tạo xác suất ban đầu cho các trạng thái có thể bất kỳ nhưng phải thoả điều kiện tổng xác suất ban đầu của các trạng thái phải luôn bảo toàn. Có nghĩa $\sum_{i=1}^N \pi_i = 1$.

✓ **Bước 2:** Ước lượng và cập nhật lại tham số cho mô hình

+Gọi $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$

+Cập nhật xác suất chuyển trạng thái i tại thời điểm t sang trạng thái j tại thời điểm $t+1$ \bar{a}_{ji} của ma trận \bar{A}

+Cập nhật xác suất để quan sát tại thời điểm t thuộc về trạng thái i $\bar{b}_i(t)$ của ma trận \bar{B}

+Cập nhật trạng thái xác suất ban đầu $\bar{\pi}_i$, xác suất này thường không thay đổi $\bar{\pi}_i = \pi_i$

✓ **Bước 3:** Kiểm tra độ hội tụ của thuật toán

Để tác vụ huấn luyện một mô hình Markov ẩn rời rạc là hội tụ thì cần phải thoả hai thuộc tính sau:

+ Xác suất ước lượng một chuỗi quan sát O thuộc về mô hình đang huấn luyện $\lambda: P(O/\lambda)$ là cực đại qua các lần lặp.

+ Sự khác nhau giữa xác suất ước lượng tại các vòng lặp liên tiếp nhau phải nhỏ hơn một ngưỡng C xác định trước.

Nếu quá trình huấn luyện một mô hình Markov ẩn rời rạc hội tụ (thoả hai điều kiện trên), có nghĩa một mô hình Markov ẩn rời rạc với các tham số đã được ước lượng phù hợp nhất cho tập các chuỗi quan sát. Và chuyển sang bước 5.

Nếu quá trình huấn luyện chưa hội tụ thì chuyển sang bước 4.

✓ **Bước 4:** Cơ chế huấn luyện được lặp lại tại bước 2.

✓ **Bước 5:** Kết thúc với một mô hình Markov ẩn đã được ước lượng từ các mẫu khuôn mặt khác nhau nhưng cùng thuộc về một người cần nhận dạng.

6.2.2.1.5 Đồ thị biểu diễn tác vụ học qua các vòng lặp và cực đại

xác suất ước lượng mô hình từ dữ liệu quan sát.

- ✓ Dưới đây chúng tôi sẽ mô phỏng lại các kết quả của quá trình huấn luyện một mô hình Markov ẩn rời rạc và tìm ra các quy luật điều khiển bộ tham số cho một mô hình mà có khả năng mô hình hóa tập các chuỗi quan sát tốt nhất, nhanh nhất, mà vẫn đảm bảo khả năng xử lý tổng quát hóa tốt trong bước nhận dạng khuôn mặt với chuỗi quan sát mới.
- ✓ Ở đây chúng tôi chỉ khảo sát việc đi tìm một mô hình Markov ẩn rời rạc thích hợp để mô hình hoá một tập quan sát cho trước, tập quan sát này là tập mẫu khuôn mặt của người đầu tiên trong kho dữ liệu khuôn mặt cần được huấn luyện cho ứng dụng nhận dạng khuôn mặt trên cách tiếp cận theo mô hình Markov ẩn rời rạc.
- ✓ Tập khuôn mặt được khảo sát trong thực nghiệm này có một số dạng trong Hình 6-12 sau:



Hình 6-12 Chuỗi quan sát từ người thứ nhất được gán nhãn “Người 1”.

- ✓ Trạng thái của một khuôn mặt biến đổi khá phức tạp.
 - + Các biến đổi ngay trong nội tại của khuôn mặt bao gồm: Mắt (có cả nhắm và mở), trên khuôn mặt xuất hiện râu cằm/có khuôn mặt thì không có, có khuôn mặt lại đeo kính.
 - + Các ảnh hưởng của môi trường ngoài: Có những ảnh thu được trong điều kiện ánh sáng rất kém, nhưng cũng có những ảnh xuất hiện với độ chói quá cao.
- ✓ Cơ chế thử nghiệm để chọn bộ tham số $< \text{Số trạng thái của mô hình (N), Số kí hiệu quan sát cho mỗi trạng thái (M)} >$ của một mô hình Markov ẩn được thực hiện như sau:
 - + Để kiểm tra mức độ ảnh hưởng tham số M (Số kí hiệu quan sát cho mỗi trạng thái) của một mô hình Markov ẩn rời rạc đến việc mô hình hóa tập mẫu khuôn mặt bằng một con số. Đó chính là xác suất $P(O|\lambda)$ để tập khuôn mặt khảo sát thuộc về mô hình sao cho càng cực đại bao nhiêu thì càng tốt bấy nhiêu.

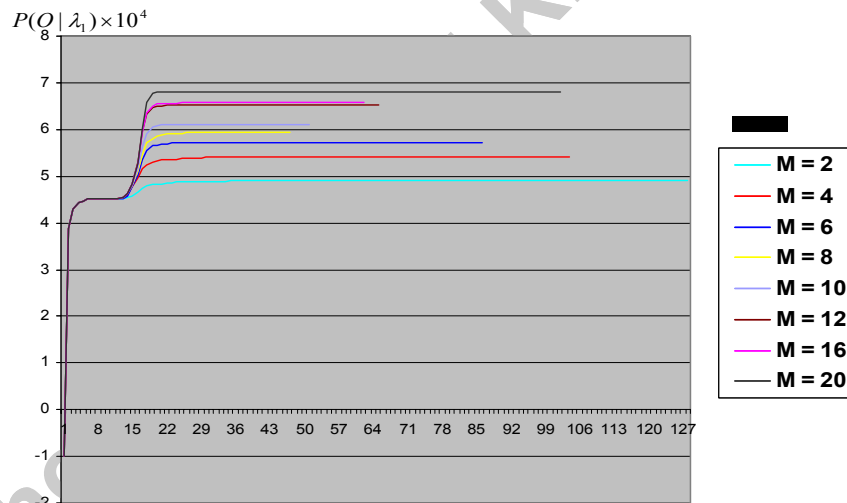
+ Giữ cố định giá trị số trạng thái của một mô hình có thể có $N = \{4,6,8,10\}$ và khảo sát $P(O|\lambda)$ lần lượt với các giá trị của tham số M khác nhau trên tập khuôn mặt của người thứ 1.

Khoa CNTT - ĐH KHTN TP.HCM

✓ **Thí nghiệm 1:** Kết quả thực hiện được trình bày trên Hình 6-13.

STT	N	M	Cực đại tổng xác suất
			$P(\text{Tập các khuôn mặt thuộc về "Người 1"} / \lambda_1)$
1	4	2	4.90473e+004
2	4	4	5.39684e+004
3	4	6	5.72160e+004
4	4	8	5.94079e+004
5	4	10	6.10240e+004
6	4	12	6.51353e+004
7	4	16	6.58552e+004
8	4	20	6.59536e+004

Bảng 6-1 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc với số trạng thái là 4 và hệ số Mixture thay đổi từ 2→20



Hình 6-13 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $N = 4$

- **Nhận xét kết quả từ Hình 6-13:**

- + Đường màu xanh da trời (đường biểu diễn dưới cùng) chính là kết quả của quá trình huấn luyện mô hình Markov ẩn rời rạc cho tập mẫu khuôn mặt “Người 1” tương ứng với $N = 4$, $M = 2$ và kết quả mô hình hoá bằng tổng xác suất thấp nhất $P(O|\lambda_1) = 4.90473e+004$.

- + Đường màu đen (đường biểu diễn cao nhất) chính là kết quả của quá trình huấn luyện mô hình Markov ẩn rời rạc cho tập mẫu khuôn mặt “Người 1” tương ứng với $N = 4$, $M = 20$ và kết quả mô hình hoá bằng tổng xác suất cao nhất $P(O|\lambda_1) = 6.59536e+004$.

- + Cùng một số trạng thái nhưng mô hình nào có số kí hiệu quan sát (M) nhiều hơn thì có thể mô hình hoá tập khuôn mặt tốt hơn.

- + Đề nghị: Cần tăng thêm giá trị của M để kiểm tra sự cực đại này đến một mức nào đó thì dừng

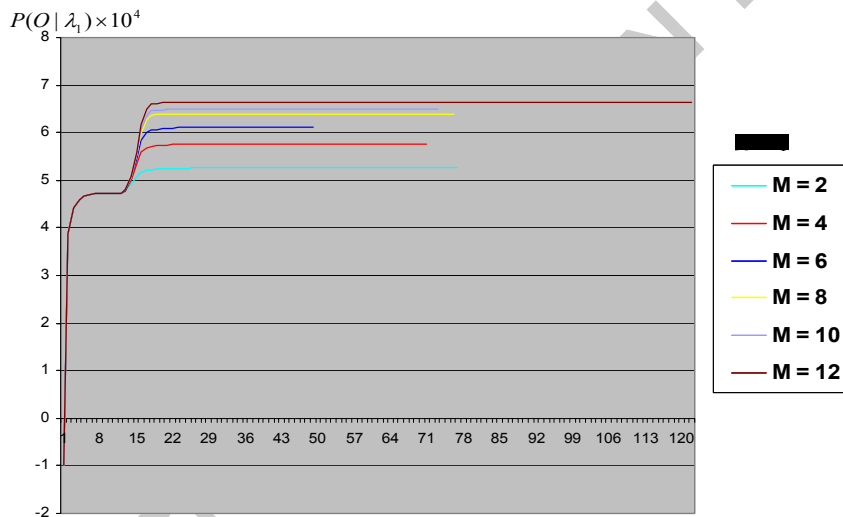
- + Khuyết điểm: Khi M càng tăng thì thời gian cho huấn luyện một mô hình Markov ẩn rời rạc cũng tăng lên đáng kể.

Khoa CNTT - ĐH KHNTT - HCM

✓ **Thí nghiệm 2:** Kết quả được trình bày trên (Hình 6-14)

STT	N	M	Cực đại tổng xác suất
			$P(\text{Tập khuôn mặt thuộc về "Người 1"} / \lambda_1)$
1	6	2	5.27965e+004
2	6	4	5.75586e+004
3	6	6	6.11844e+004
4	6	8	6.38800e+004
5	6	10	6.48651e+004
6	6	12	6.63666e+004

Bảng 6-2 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc với số trạng thái là 6 và hệ số Mixture thay đổi từ 2→12



Hình 6-14 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $N = 6$

• **Nhận xét kết quả từ Hình 6-14:**

+ Đường màu xanh da trời (đường biểu diễn dưới cùng) chính là kết quả của tiến trình huấn luyện một mô hình Markov ẩn rời rạc cho tập mẫu khuôn mặt “Người 1” tương ứng với $N = 6, M = 2$ và kết quả mô hình hoá bằng một con số đại diện, đó chính là tổng xác cực đại để ước lượng mô hình $\lambda_1: P(O|\lambda_1) = 4.90473e+004$. Và khả năng mô hình hoá này là thấp nhất so với 6 thử nghiệm cùng trên số trạng thái chung $N = 6$.

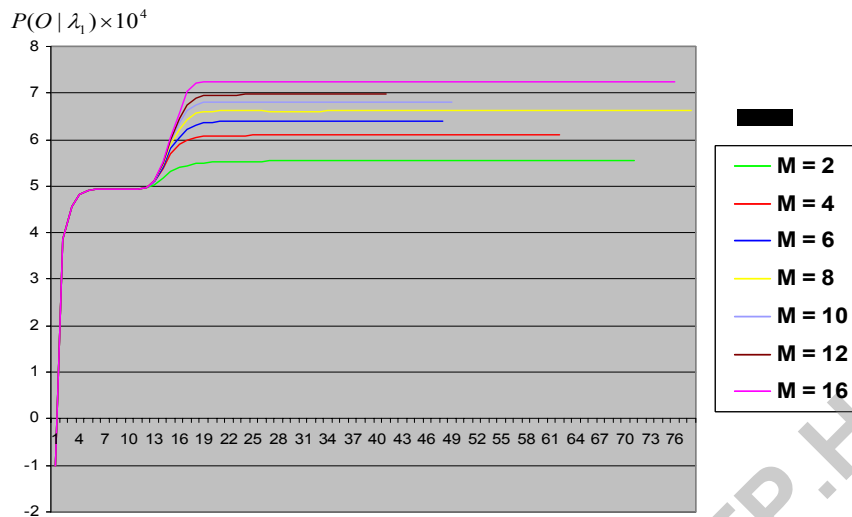
+ Đường màu mầu (đường biểu diễn cao nhất) chính là kết quả của quá trình huấn luyện một mô hình Markov ẩn rời rạc cho tập mẫu khuôn mặt “Người 1” tương ứng với $N = 6$, $M = 12$ và kết quả mô hình hoá bằng một con số đại diện, đó chính là tổng xác cực đại để ước lượng mô hình $\lambda_1: P(O|\lambda_1) = 6.63666e+004$. Và khả năng mô hình hoá này là cao nhất so với 6 thử nghiệm cùng trên số trạng thái chung $N = 6$.

+ Như vậy khi so kết quả với thí nghiệm 1, Thì quy luật số kí hiệu trong một thuộc tính càng cao thì làm cho khả năng mô hình hóa của mô hình Markov càng tốt là đúng.

✓ **Thí nghiệm 3:** Kết quả được trình bày trên (Hình 6-15)

STT	N	M	Cực đại tổng xác suất $P(\text{Tập khuôn mặt thuộc về "Người 1"} / \lambda_1)$
1	8	2	5.54989e+004
2	8	4	6.09219e+004
3	8	6	6.40407e+004
4	8	8	6.62552e+004
5	8	10	6.80923e+004
6	8	12	6.97235e+004
7	8	16	7.25107e+004

Bảng 6-3 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc với số trạng thái là 8 và hệ số Mixture thay đổi từ 2→16



Hình 6-15 Các tiến trình huấn luyện HMM cho tập khuôn mặt “Người 1” với $N = 8$

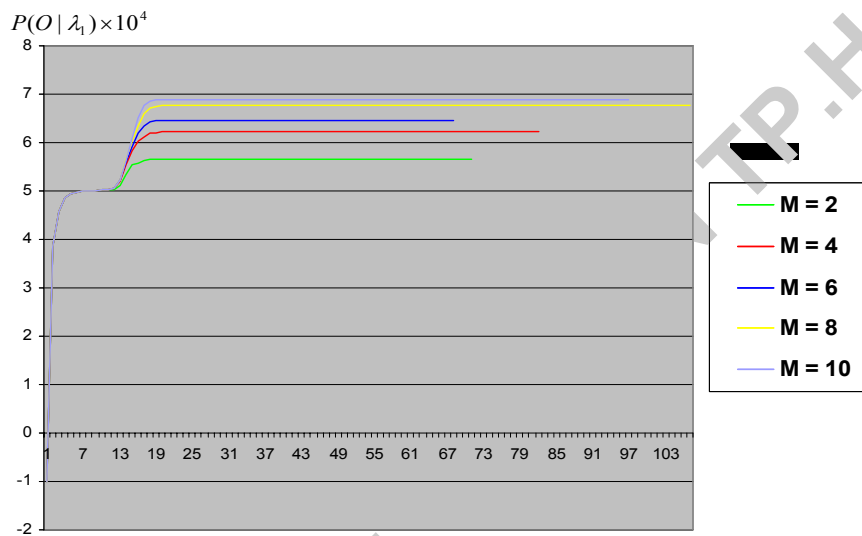
- Nhận xét kết quả từ Hình 6-15:**

 - + Đường màu xanh da trời (đường biểu diễn dưới cùng) chính là kết quả của tiến trình huấn luyện một mô hình Markov ẩn rời rạc cho tập mẫu khuôn mặt “Người 1” tương ứng với $N = 8$, $M = 2$ và kết quả mô hình hoá bằng một con số đại diện, đó chính là tổng xác cực đại để ước lượng mô hình $\lambda_1: P(O|\lambda_1) = 5.54989e+004$. Và khả năng mô hình hoá này là thấp nhất so với 6 thử nghiệm cùng trên số trạng thái chung $N = 8$.
 - + Đường màu hồng (đường biểu diễn cao nhất) chính là kết quả của quá trình huấn luyện mô hình Markov ẩn rời rạc cho tập mẫu khuôn mặt “Người 1” tương ứng với $N = 8$, $M = 16$ và kết quả mô hình hoá bằng một con số đại diện, đó chính là tổng xác cực đại để ước lượng mô hình $\lambda_1: P(O|\lambda_1) = 7.25107e+004$. Và khả năng mô hình hoá này là cao nhất so với 6 thử nghiệm cùng trên số trạng thái chung $N = 6$.
 - + Như vậy khi so kết quả với thí nghiệm 1 và 2, Thì quy luật số ký hiệu trong một thuộc tính càng cao thì làm cho khả năng mô hình hóa của mô hình Markov càng tốt vẫn đúng.
- ✓ **Thí nghiệm 4:** Kết quả được trình bày trên (Hình 6-16)

STT	N	M	Cực đại tổng xác suất
-----	---	---	-----------------------

			$P(\text{Tập khuôn mặt thuộc về "Người 1"} / \lambda_i)$
1	10	2	5.66818e+004
2	10	4	6.22766e+004
3	10	6	6.45893e+004
4	10	8	6.76202e+004
5	10	10	6.88324e+004

Bảng 6-4 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc với số trạng thái là 10 và hệ số Mixture thay đổi từ 2→10



Hình 6-16 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $N = 10$

- **Nhận xét kết quả từ Hình 6-16:**

+ Đường màu xanh da trời (đường biểu diễn dưới cùng) chính là kết quả của tiến trình huấn luyện một mô hình Markov ẩn rời rạc cho tập mẫu khuôn mặt “Người 1” tương ứng với $N = 10$, $M = 2$ và kết quả mô hình hoá bằng một con số đại diện, đó chính là tổng xác cực đại để ước lượng mô hình $\lambda_1: P(O|\lambda_1) = 5.66818e+004$. Và khả năng mô hình hoá này là thấp nhất so với 6 thử nghiệm cùng trên số trạng thái chung $N = 10$.

+ Đường màu hồng (đường biểu diễn cao nhất) chính là kết quả của quá trình huấn luyện mô hình Markov ẩn rời rạc cho tập mẫu khuôn mặt “Người 1” tương ứng với $N = 10$, $M = 10$ và kết quả mô hình hoá bằng một con số đại diện, đó chính là tổng xác cực đại để ước lượng mô hình $\lambda_1: P(O|\lambda_1) = 6.88324e+004$. Và khả năng mô hình hoá này là cao nhất so với 6 thử nghiệm cùng trên số trạng thái chung $N = 10$.

+ Như vậy khi so kết quả với thí nghiệm 1,2 và 3, Thì quy luật số kí hiệu trong một thuộc tính càng cao thì làm cho khả năng mô hình hóa của mô hình Markov càng tốt vẫn đúng.

+ Một nhận xét rất thú vị rút ra từ 4 thí nghiệm trên như sau: Chúng ta có thể thấy rằng việc khả năng mô hình hoá của một mô hình Markov ẩn là tuyến tính so với số ký hiệu quan sát trên mỗi thuộc tính của mô hình (hay còn gọi là số đặc trưng của trạng thái).

+ Nhưng một kết quả tăng khác lại rất đáng lo ngại về khả năng xử lý của mô hình Markov, đó là thời gian xử lý cũng tăng tuyến tính theo tham số M.

+ Để kiểm tra mức độ ảnh hưởng của tham số N (Số trạng thái) trong một mô hình mô hình Markov ẩn rời rạc đến việc mô hình hoá tập mẫu khuôn mặt bằng một con số. Đó chính là xác suất $P(O|\lambda)$ để tập khuôn mặt khảo sát thuộc về mô hình sao cho càng cực đại bao nhiêu thì càng tốt bấy nhiêu.

+ Bằng cách giữ cố định giá trị số kí hiệu quan sát trên một trạng thái cho một mô hình có thể có $M = \{2,4,6,8,10\}$ và khảo sát $P(O|\lambda)$ lần lượt với các giá trị của tham số N khác nhau trên tập khuôn mặt của người “Người 1”.

✓ **Thí nghiệm 5:** Kết quả thực hiện được trình bày trên Hình 6-17.

STT	Số quan sát/ Một trạng thái (M)	Số trạng thái (N)	Cực đại tổng xác suất $P(\text{Tập khuôn mặt thuộc về "Người 1"} / \lambda_1)$
1	2	4	4.90473e+004
2	2	6	5.27965e+004
3	2	8	5.54989e+004
4	2	10	5.66818e+004

Bảng 6-5 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 2 và hệ số trạng thái thay đổi từ 4→10

• **Nhận xét kết quả từ Hình 6-17:**

+ Kết quả từ Bảng 9 chính là kết quả của việc huấn luyện bốn mô hình tham số mô hình Markov ẩn rời rạc trên cùng một tập dữ liệu khuôn mặt được gán nhãn “Người 1”. Cái khác nhau chủ yếu từ bốn mô hình này chính là sử dụng các giá trị của tham số trạng thái N khác nhau và cùng sử dụng chung một giá trị của tham số là số kí hiệu quan sát M.

+ Với $N = 4$ và $M = 2$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 4.90473e+004$. Tiến trình ước lượng bộ tham số cho mô hình này được thể hiện trên đường màu xanh da trời từ hình 25 và con số thể hiện cho chất lượng mô hình hoá của mô hình với $\langle N=4 \rangle$ và $\langle M=2 \rangle$ là thấp nhất so với con số đánh giá việc mô

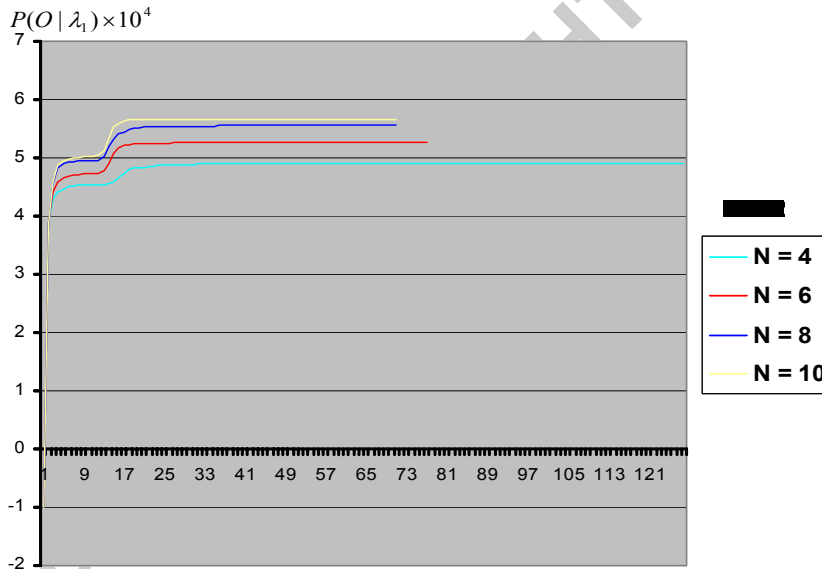
hình hoá của các mô hình khác, cũng sử dụng $M = 2$ nhưng N lớn hơn.

+ Với $N = 10$ và $M = 2$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 5.66818e+004$. Đó cũng chính là con số mô hình hoá lớn nhất của mô hình Markov mà có nhiều số trạng thái nhất, $N = 10$.

+ Trong thí nghiệm này với M không đổi, việc ước lượng tham số cho một mô hình Markov ẩn rời rạc dựa trên tập các chuỗi quan sát khuôn mặt sẽ tốt hơn khi sử dụng số trạng thái cho mô hình lớn hơn.

+ Chi phí tính toán cũng tăng lên đáng kể.



Hình 6-17 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $M = 2$

✓ **Thí nghiệm 6:** Kết quả thực hiện được trình bày trên Hình 6-18.

STT	M	N	Cực đại tổng xác suất $P(\text{Tập khuôn mặt thuộc về "Người 1"} / \lambda_1)$

1	4	4	5.39684e+004
2	4	6	5.75586e+004
3	4	8	6.09219e+004
4	4	10	6.22766e+004

Bảng 6-6 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 4 và hệ số trạng thái thay đổi từ 4→10

- **Nhận xét kết quả từ Hình 6-18:**

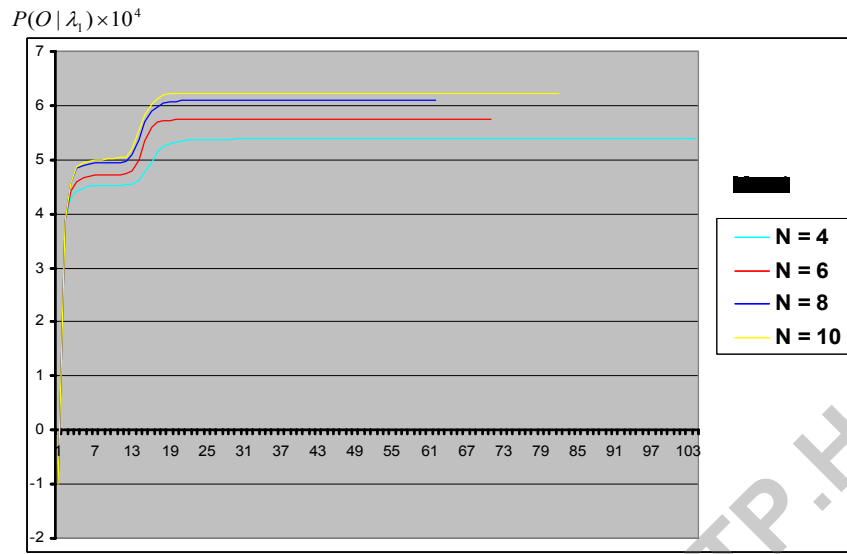
+ Với $N = 4$ và $M = 4$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 4.90473e+004$. Tiến trình ước lượng bộ tham số cho mô hình này được thể hiện trên đường màu xanh da trời từ hình 25 và con số thể hiện cho chất lượng mô hình hoá của mô hình với $\langle N=4 \rangle$ và $\langle M=4 \rangle$ là thấp nhất so với con số đánh giá việc mô hình hoá của các mô hình khác, cũng sử dụng $M = 4$ nhưng N lớn hơn.

+ Với $N = 10$ và $M = 4$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 6.22766e+004$. Đó cũng chính là con số mô hình hoá lớn nhất của mô hình Markov mà có nhiều số trạng thái nhất, $N = 10$.

+ Trong thí nghiệm này với M không đổi, việc ước lượng tham số cho một mô hình mô hình Markov ẩn rời rạc dựa trên tập các chuỗi quan sát khuôn mặt sẽ tốt hơn khi sử dụng số trạng thái cho mô hình lớn hơn.



Hình 6-18 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $M = 4$

Khoa CNTT - ĐH KHÍ QUẢN TP.HCM

✓ **Thí nghiệm 7:** Kết quả thực hiện được trình bày trên Hình 6-19.

STT	Số quan sát/ Một trạng thái (M)	Số trạng thái (N)	Cực đại tổng xác suất $P(\text{Tập khuôn mặt thuộc về "Người 1"} / \lambda_1)$
1	6	4	5.72160e+004
2	6	6	6.11844e+044
3	6	8	6.40407e+004
4	6	10	6.45893e+004

Bảng 6-7 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 6 và hệ số trạng thái thay đổi từ 4→10

• **Nhận xét kết quả từ Hình 6-19:**

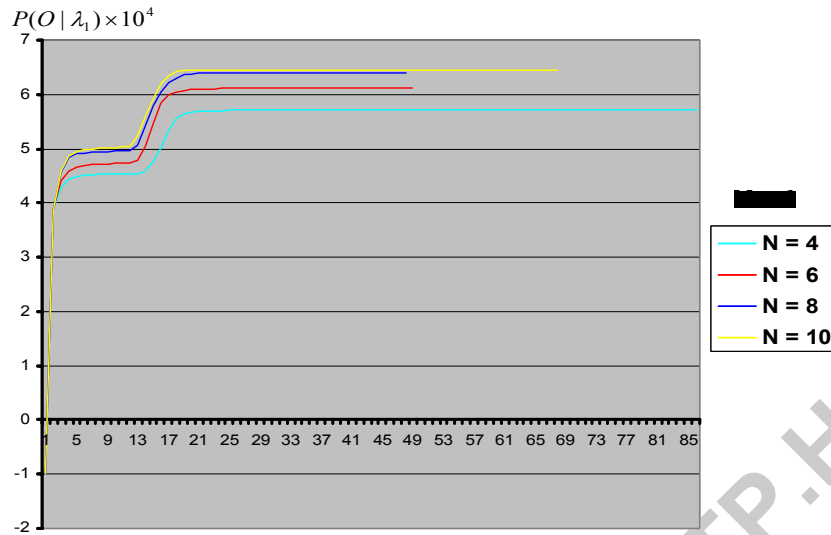
+ Với $N = 4$ và $M = 6$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 5.72160e+004$. Tiến trình ước lượng bộ tham số cho mô hình này được thể hiện trên đường màu xanh da trời từ hình 25 và con số thể hiện cho chất lượng mô hình hoá của mô hình với $\langle N=4 \rangle$ và $\langle M=6 \rangle$ là thấp nhất so với con số đánh giá việc mô hình hoá của các mô hình khác, cũng sử dụng $M = 6$ nhưng N lớn hơn.

+ Với $N = 10$ và $M = 6$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 6.45893e+004$. Đó cũng chính là con số mô hình hoá lớn nhất của mô hình Markov mà có nhiều số trạng thái nhất, $N = 10$.

+ Trong thí nghiệm này với M không đổi, việc ước lượng tham số cho một mô hình Markov ẩn rời rạc dựa trên tập các chuỗi quan sát khuôn mặt sẽ tốt hơn khi sử dụng số trạng thái cho mô hình lớn hơn.



Hình 6-19 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $M = 6$

✓ **Thí nghiệm 8:** Kết quả thực hiện được trình bày trên Hình 6-20.

STT	M	N	Cực đại tổng xác suất $P(\text{Tập khuôn mặt thuộc về "Người 1"} / \lambda_1)$
1	8	4	5.94079e+004
2	8	6	6.38800e+004
3	8	8	6.62552e+004
4	8	10	6.76202e+004

Bảng 6-8 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 8 và hệ số trạng thái thay đổi từ 4→10

• **Nhận xét kết quả từ Hình 6-20:**

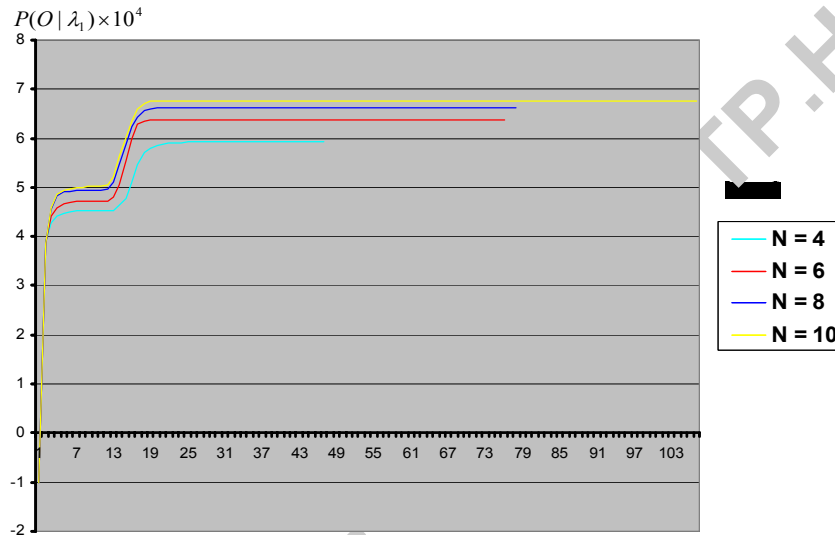
+ Với $N = 4$ và $M = 8$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 5.94079e+004$. Tiến trình ước lượng bộ tham số cho mô hình này được thể hiện trên đường màu xanh da trời từ hình 25 và con số thể hiện cho chất lượng mô hình hoá của mô hình với $\langle N=4 \rangle$ và $\langle M=8 \rangle$ là thấp nhất so với con số đánh giá việc mô hình hoá của các mô hình khác, cũng sử dụng $M = 8$ nhưng N lớn hơn.

+ Với $N = 10$ và $M = 8$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 6.76202e+004$. Đó cũng chính là con số mô hình hoá lớn nhất của mô hình Markov mà có nhiều số trạng thái nhất, $N = 10$.

+ Trong thí nghiệm này với M không đổi, việc ước lượng tham số cho một mô hình Markov ẩn rời rạc dựa trên tập các chuỗi quan sát khuôn mặt sẽ tốt hơn khi sử dụng số trạng thái cho mô hình lớn hơn.



Hình 6-20 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $M = 8$

✓ **Thí nghiệm 9:** Kết quả thực hiện được trình bày trên Hình 6-21.

STT	M	N	Cực đại tổng xác suất $P(\text{Tập khuôn mặt thuộc về "Người 1"} / \lambda_1)$
1	10	4	6.10240e+004
2	10	6	6.48651e+004
3	10	8	6.80923e+004
4	10	10	6.88324e+004

Bảng 6-9 Bảng số liệu khi thử nghiệm huấn luyện mô hình Markov ẩn rời rạc Hệ số Mixture bằng 10 và hệ số trạng thái thay đổi từ 4→10

- Nhận xét kết quả từ Hình 6-21:

+ Với $N = 4$ và $M = 10$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

$P(O|\lambda_1) = 6.10240e+004$. Tiến trình ước lượng bộ tham số cho mô hình này được thể hiện trên đường màu xanh da trời từ hình 29 và con số thể hiện cho chất lượng mô hình hoá của mô hình với $\langle N=4 \rangle$ và $\langle M=10 \rangle$ là thấp nhất so với con số đánh giá việc mô hình hoá của các mô hình khác, cũng sử dụng $M = 10$ nhưng N lớn hơn.

+ Với $N = 10$ và $M = 10$, Tổng xác suất được tính từ việc ước lượng cho mô hình λ_1 :

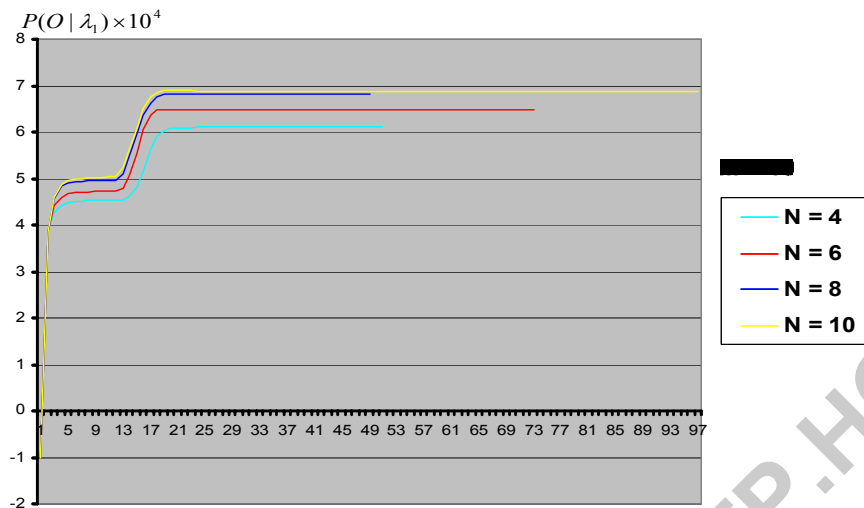
$P(O|\lambda_1) = 6.88324e+004$. Đó cũng chính là con số mô hình hoá lớn nhất của mô hình Markov mà có nhiều số trạng thái nhất, $N = 10$.

+ Trong thí nghiệm này với M không đổi, việc ước lượng tham số cho một mô hình Markov ẩn rời rạc dựa trên tập các chuỗi quan sát khuôn mặt sẽ tốt hơn khi sử dụng số trạng thái cho mô hình lớn hơn.

+ Ta có thể rút ra một quy luật quan trọng, Khi mô hình hoá cho tập quan sát phức tạp bằng một mô hình, việc sử dụng càng nhiều số trạng thái thì việc ước lượng càng tốt

+ Cũng giống như bốn thí nghiệm đầu tiên, và năm thí nghiệm được tiến hành liên tiếp sau này cũng đảm bảo chất lượng của mô hình hoá sẽ tăng tuyến tính đối với số trạng thái sử dụng để huấn luyện cho một mô hình.

+ Thời gian xử lý của hệ thống cũng tăng tuyến tính.



Hình 6-21 Các tiến trình huấn luyện mô hình Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $M = 10$ và Markov ẩn rời rạc cho tập khuôn mặt “Người 1” với $M = 10$

Tóm lại: sau khi thực hiện trên 9 thí nghiệm ta có thể rút ra kết luận sau:

- + Bộ tham số $\langle N \rangle \langle M \rangle$ đóng một vai trò lớn trong việc điều khiển khả năng mô hình hoá chuỗi quan sát của một mô hình Markov ẩn rời rạc.
- + Thời gian xử lý của hệ thống cũng phụ thuộc vào sự lựa chọn hai tham số này.

6.2.2.2 Giai đoạn nhận dạng khuôn mặt

Các bước tiền xử lý và rút trích đặc trưng với mẫu cần nhận dạng được thực hiện tương tự các bước đã làm trong quá trình huấn luyện để tạo ra chuỗi quan sát.

$$\text{Tính } P(O^{(t)} | \lambda_k) = \max_n P(O^{(t)} | \lambda_n)$$

Quyết định khuôn mặt thử nghiệm t được nhận dạng là người thứ k .

6.2.3 Nhận xét và hướng phát triển tương lai

6.2.3.1 Ưu điểm

Quá trình huấn luyện một mô hình Markov ẩn có thể tổng hợp tất cả các mẫu khác

nhau của cùng một lớp để đưa ra một mô hình chung cho tập mẫu học, và kết quả mô hình hoá này được đặc trưng bằng một con số. Vì thế hiện tượng quên mẫu đã được học là rất ít khả năng xảy ra.

6.2.3.2 Khuyết điểm

Chất lượng mô hình hoá của mô hình Markov phụ thuộc vào việc điều khiển giá trị của bộ tham số, đặc biệt là tham số về <**Số trạng thái của mô hình**> và <**Số kí hiệu quan sát trên một trạng thái**>. Khi tăng tuyến tính giá trị của bộ hai tham số này thì thời gian xử lý của hệ thống cũng tăng tuyến tính.

Khoa CNTT - ĐH KHTN TP.HCM

Chương 7 THIẾT KẾ CHƯƠNG TRÌNH VÀ
HƯỚNG DẪN SỬ DỤNG

Khoa CNTT - ĐH KHTN TP.HCM

7.1 Giới thiệu

- Chương trình được cài đặt trên môi trường Microsoft Visual C++ 6.0, chương trình nhằm mục đích xây dựng hệ thống nhận dạng khuôn mặt người dựa vào thông tin khuôn mặt trên ảnh với hai phương pháp tiếp cận chính nêu trên là phương pháp SVM và phương pháp mô hình Markov ẩn rời rạc.
- Minh họa kỹ thuật dò tìm khuôn mặt
- Minh họa các phương pháp rút trích tri thức đặc trưng của khuôn mặt trên ảnh.
- Minh họa việc nhận dạng khuôn mặt bằng phương pháp SVM.
- Minh họa việc nhận dạng khuôn mặt bằng phương pháp mô hình Markov ẩn rời rạc.

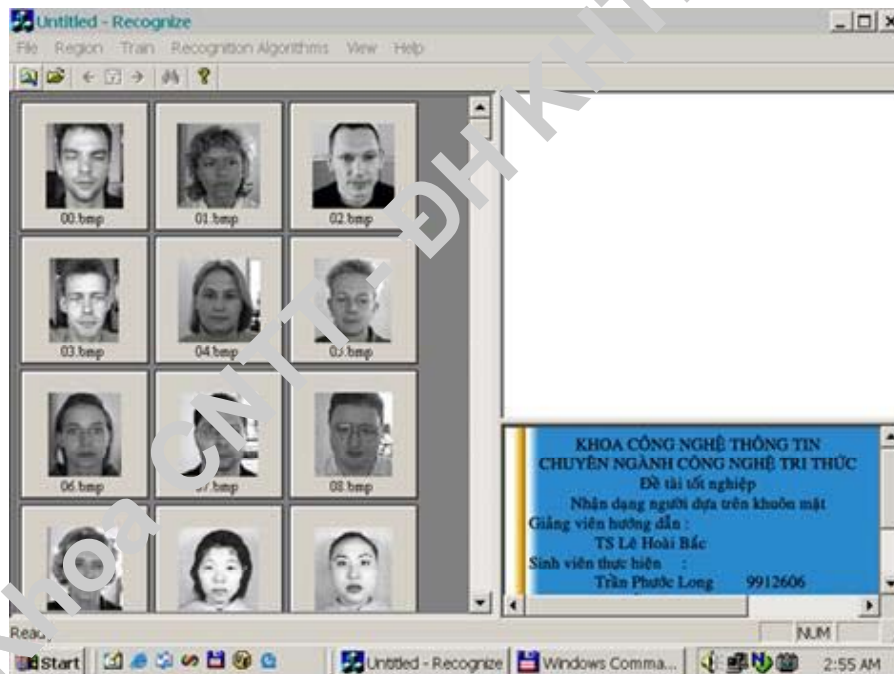
7.2 Thiết kế và cài đặt chương trình

- Chương trình được cài đặt hướng đối tượng. Sau đây là các lớp chính được thiết kế trong chương trình
 - ✓ Lớp thu nhận thông tin ảnh
CDib, LbitmapBase: Đây là những lớp đọc ảnh, lấy thông tin ảnh xử lý, hỗ trợ các chức năng xử lý ảnh khuôn mặt
 - ✓ Lớp nhận dạng theo phương pháp SVM.
CPCA, CRestorePCA: Đây là lớp quản lý rút trích đặc trưng khuôn mặt cô lập bằng phương pháp phương pháp phân tích thành phần chính.
SVMBase: Đây là lớp phục vụ cho mục đích huấn luyện và nhận dạng bằng phương pháp SVM .
 - ✓ Lớp nhận dạng theo phương pháp mô hình Markov ẩn rời rạc
CFloatDCT: Đây là lớp quản lý rút trích đặc trưng khuôn mặt trên miền tần số
CHMM: Đây là lớp phục vụ cho mục đích huấn luyện và nhận dạng bằng phương pháp mô hình Markov ẩn rời rạc hai chiều.

- ✓ Các lớp hỗ trợ khác
 - CMath:** Lớp hỗ trợ các hàm xử lý toán học
 - CDetection:** Lớp phục vụ cho việc dò tìm các khuôn mặt xuất hiện trên ảnh nếu có.
- ✓ Lớp tạo giao diện cho chương trình
 - LMyBrowse:** Dùng để xuất ảnh trên một cửa sổ màn hình dưới dạng Thumbnail.
 - CInputView:** Xuất ảnh cần nhận dạng ra màn hình
 - COutputView:** Lớp đặt logo của chương trình
- Ngoài ra đề tài còn sử dụng một số thư viện xử lý ảnh khác như: Thư viện xử lý ảnh LEADTOOLS của hãng LEAD Technology tại Website: <http://www.leadtools.com/>.

7.3 Giao diện màn hình và hướng dẫn sử dụng

- Màn hình chính chương trình



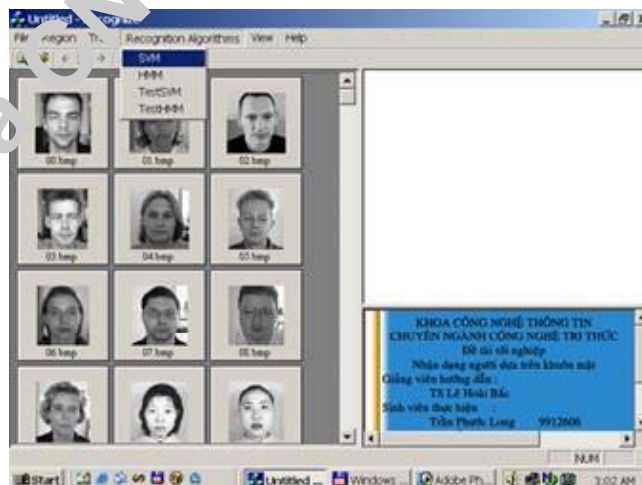
- ✓ Màn hình chính gồm 3 ba khung ảnh
- ✓ Khung ảnh bên trái: Hiển thị ảnh mẫu của những người có mặt trong hệ thống nhận dạng khuôn mặt, và hiển thị kết quả của người được nhận dạng
- ✓ Khung ảnh bên phải-phần dưới: Đó chính là logo chương trình,

Giới thiệu tên đề tài, Giáo viên hướng dẫn và sinh viên thực hiện chương trình.

- ✓ Khung ảnh bên phải-phần trên: Đó chính là vùng màn hình hiển thị ảnh được chọn cho mục đích nhận dạng, hiển thị kết quả dò tìm khuôn mặt
- Màn hình chọn ảnh thử nghiệm nhận dạng khuôn mặt
 - ✓ Để chọn ảnh thử nghiệm bạn sẽ trở vào biểu tượng (📁) trên màn hình chính
 - ✓ Chọn ảnh thử nghiệm đã được bố trí trong thư mục DBTest
- Màn hình hiển thị ảnh đã chọn để thử nghiệm nhận dạng



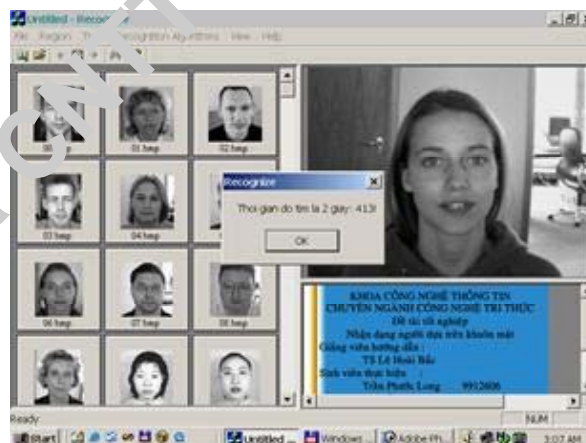
- Màn hình chọn nhận dạng theo phương pháp SVM




- ✓ Trên Menu mà hình chính chọn mục “**Recognition Algorithm**”
 - ✓ Tiếp tục chọn mục “**SVM**”
 - ✓ Xuất hiện thông điệp thông báo lấy dữ liệu thành công
- Màn hình chọn nhận dạng theo phương pháp mô hình Markov ẩn



- ✓ Trên Menu màn hình chính chọn mục “**Recognition Algorithm**”
 - ✓ Tiếp tục chọn mục “**HMM**”
 - ✓ Xuất hiện thông điệp thông báo lấy dữ liệu thành công
- Màn hình hiển thị thời gian tìm kiếm khuôn mặt trong ảnh nếu có




- ✓ Để dò khuôn mặt trên ảnh thử nghiệm bạn sẽ trở vào biểu tượng  trên màn hình chính.
- ✓ Thời gian dò tìm khuôn mặt sẽ được thông báo

- Màn hình cho biết vị trí khuôn mặt được tìm thấy bằng khung bao



- ✓ Kết quả dò tìm: đó chính là vị trí các khuôn mặt có thể có trên ảnh và vùng khuôn mặt được xác định bằng một khung bao.
- Màn hình hiển thị thời gian nhận dạng một khuôn mặt trên ảnh



- ✓ Để nhận dạng một khuôn mặt trên ảnh thử nghiệm bạn trở vào biểu tượng  trên màn hình chính.
- ✓ Thời gian nhận dạng một khuôn mặt sẽ được thông báo

- Màn hình cho biết kết quả nhận dạng khuôn mặt đã chọn



- + Kết quả nhận dạng người trong ảnh là ai trong số những người đã được lưu trữ trong hệ thống nhận dạng
- + Kết xuất tiến trình nhận dạng một khuôn mặt sẽ cho biết khuôn mặt của người nào trong cơ sở dữ liệu là tương đồng nhất với khuôn mặt cần nhận dạng, dấu hiệu nhận biết bằng đường bao viền xanh xuất hiện trên khuôn mặt của người đó.

Chương 8 THỰC NGHIỆM VÀ KẾT QUẢ

Khoa CNTT - ĐH KHTN TP.HCM

8.1 Dữ liệu và phương pháp thử nghiệm nhận dạng khuôn mặt

- Để đo khả năng nhận dạng khuôn mặt bằng hai phương pháp tiếp cận là SVM và mô hình Markov ẩn rời rạc, chúng tôi đã sử dụng tập thử nghiệm bao gồm **3341**(ảnh mẫu) khuôn mặt của 30 người khác nhau, và có thể chưa được quan sát trên tập học. Nhưng các ảnh khuôn mặt cho bước thử nghiệm này có thể rất tương đồng hay có một số biến đổi rất nhỏ so với ảnh đã thu thập trên tập học.
- Số lượng ảnh của mỗi người thu được trên tập thử nghiệm được trình bày trong Bảng 1 dưới đây:
- Đánh giá khả năng nhận dạng của 2 phương pháp tiếp cận:
{<Tỉ lệ % đúng từ người 1>,...,<Tỉ lệ % đúng từ người i >,...,<Tỉ lệ % đúng từ người 30>}
- **Tỉ lệ % đúng từ người i =**
 $(\text{Số mẫu xác minh đúng người thứ } i) * 100 / \text{Tổng số mẫu thử nghiệm từ người thứ } i$. **Ví dụ :**
 - + Tổng số mẫu thử nghiệm từ người thứ 1 = 285(mẫu)
 - + Số mẫu xác minh đúng người thứ 1 = 200(mẫu)
 - + Tỉ lệ % đúng từ người 1 = $200 * 100 / 285 = 70\%$
- Kết quả thử nghiệm từ tập mẫu của mỗi người được ghi nhận bằng hai giá trị: đó là chất lượng và thời gian nhận dạng.

Tỉ lệ % xác minh đúng người thứ i
Thời gian (s) xác minh tập mẫu người thứ i

Số người Nhận dạng	Ảnh gốc	Mirror	Noise	Gamma	Average	Tổng cộng
Người 1	57	57	57	57	57	285(ảnh)
Người 2	78	78	78	78	78	390(ảnh)
Người 3	30	30	30	30	30	151(ảnh)
Người 4	35	35	35	35	35	176(ảnh)
Người 5	50	50	50	50	50	251(ảnh)
Người 6	43	43	43	43	43	316(ảnh)
Người 7	67	67	67	67	67	331(ảnh)
Người 8	68	68	68	68	68	391(ảnh)
Người 9	49	49	49	49	49	246(ảnh)
Người 10	73	73	73	73	73	304(ảnh)
Người 11	5	5	5	5	5	25(ảnh)
Người 12	5	5	5	5	5	25(ảnh)
Người 13	5	5	5	5	5	25(ảnh)
Người 14	5	5	5	5	5	25(ảnh)
Người 15	5	5	5	5	5	25(ảnh)
Người 16	5	5	5	5	5	25(ảnh)
Người 17	5	5	5	5	5	25(ảnh)
Người 18	5	5	5	5	5	25(ảnh)
Người 19	5	5	5	5	5	25(ảnh)
Người 20	5	5	5	5	5	25(ảnh)
Người 21	5	5	5	5	5	25(ảnh)
Người 22	5	5	5	5	5	25(ảnh)
Người 23	5	5	5	5	5	25(ảnh)
Người 24	5	5	5	5	5	25(ảnh)
Người 25	5	5	5	5	5	25(ảnh)
Người 26	5	5	5	5	5	25(ảnh)
Người 27	5	5	5	5	5	25(ảnh)
Người 28	5	5	5	5	5	25(ảnh)
Người 29	5	5	5	5	5	25(ảnh)
Người 30	5	5	5	5	5	25(ảnh)

Bảng 8-1 Mô tả dữ liệu thử nghiệm thu thập từ mỗi người trong hệ thống nhận dạng

8.2 Kết quả Kết quả theo tiếp cận HMM

8.2.1 Thực nghiệm trên từng bộ tham số

- Để kiểm tra mức độ ảnh hưởng của hai tham số là số trạng thái(N) và số ký hiệu quan sát trên mỗi trạng thái (M) của một mô hình Markov ẩn đến kết quả nhận dạng khuôn mặt như thế nào và qua đó chọn ra một bộ tham số tốt nhất mà nó có khả năng giải thích tốt nhất các khuôn mặt của những người khác nhau.
- Tương ứng với mỗi trạng thái từ $N = \{4, 6, 8, 10\}$ chúng ta sẽ kết hợp với một tham số quan sát $M = \{2, 4, 6, 8, 10\}$.
- Mỗi mẫu khuôn mặt 32×32 (pixels) sẽ được vector hóa thành mẫu vector 560 (chiều) bằng phương pháp Biến đổi Cosine rời rạc hai chiều (2-DCT).
- Kết quả thí nghiệm ứng với mỗi bộ tham số bao gồm $\{N, M\}$ được trình bày dưới đây:
 - Vì số ảnh khuôn mặt đưa vào hệ thống nhận dạng cho những người từ Người 11 đến Người 30 rất ít chỉ 25 ảnh thử nghiệm cho mỗi người, và hệ thống nhận dạng tốt cho các trường hợp này với độ chính xác 99% đến 100%.
 - Dữ liệu khuôn mặt của những người có mã số từ Người 1 đến Người 10 được thu thập rất phong phú và tập dữ liệu khuôn mặt của những người này sẽ đánh giá tốt hơn về mô hình Markov ẩn rời rạc đã huấn luyện cho nhận dạng khuôn mặt, và dưới đây chúng tôi chỉ trình bày kết quả nhận dạng từ Người 1 đến Người 10 tương ứng với mỗi bộ tham số đã chọn cho mô hình thuật toán nhận dạng mô hình Markov ẩn rời rạc.
- Để so sánh kết quả nhận dạng giữa mô hình với bộ tham số (N-M) với mô hình với bộ tham số (N'-M') chúng tôi đã sử dụng một công thức tính như sau:

$$a = \sum_{i=1}^{30} [(Ti \text{ lệ \% người } i \text{ ứng với } N-M) - (Ti \text{ lệ \% người } i \text{ ứng với } N'-M')] (*)$$

Nếu $a > 0$ thì mô hình với bộ N-M cho chất lượng nhận dạng tốt hơn.

Nếu $a < 0$ thì mô hình với bộ tham số N'-M' cho chất lượng nhận dạng tốt hơn.

➤ Thực nghiệm 1 với $N = 4$, $M = \{2, 4, 6, 8, 10\}$.

Số người	M=2	M=4	M=6	M=8	M=10
Người 1	15(s)	18(s)	21(s)	24(s)	29(s)
	74%	78%	84%	82%	77%
Người 2	21(s)	25(s)	29(s)	31(s)	36(s)
	87%	91%	93%	92%	92%
Người 3	10(s)	12(s)	14(s)	16(s)	19(s)
	99%	100%	100%	99%	100%
Người 4	12(s)	13(s)	16(s)	18(s)	21(s)
	94%	96%	93%	93%	96%
Người 5	14(s)	16(s)	20(s)	25(s)	27(s)
	89%	92%	94%	93%	93%
Người 6	13(s)	15(s)	17(s)	20(s)	24(s)
	98%	98%	98%	98%	97%
Người 7	19(s)	20(s)	23(s)	27(s)	32(s)
	60%	73%	81%	82%	83%
Người 8	20(s)	23(s)	26(s)	31(s)	37(s)
	84%	89%	94%	93%	96%
Người 9	14(s)	16(s)	19(s)	22(s)	27(s)
	88%	89%	89%	87%	91%
Người 10	18(s)	19(s)	22(s)	26(s)	31(s)
	94%	96%	96%	96%	96%

Bảng 8-2 Kết quả hệ thống nhận dạng theo mô hình Markov với số trạng thái $N = 4$

➤ Áp dụng công thức (*) chúng ta thấy rằng với bộ tham số $N = 2$, $M = 6$ cho kết quả nhận dạng tốt nhất với thực nghiệm 1

➤ Thực nghiệm 2 với $N = 6$ và $M = \{2, 4, 6, 8, 10\}$

Số người	M=2	M=4	M=6	M=8	M=10
Người 1	17(s)	21(s)	23(s)	36(s)	48(s)
	70%	77%	76%	76%	85%
Người 2	21(s)	26(s)	33(s)	45(s)	1.10(s)
	90%	92%	94%	91%	95%
Người 3	11(s)	14(s)	17(s)	23(s)	25(s)
	99%	100%	99%	99%	99%
Người 4	12(s)	18(s)	19(s)	26(s)	34(s)
	90%	96%	94%	92%	92%
Người 5	16(s)	19(s)	24(s)	34(s)	43(s)
	89%	92%	91%	90%	95%
Người 6	14(s)	17(s)	22(s)	29(s)	40(s)
	97%	97%	98%	98%	98%
Người 7	19(s)	23(s)	29(s)	39(s)	53(s)
	86%	91%	95%	96%	96%
Người 8	21(s)	26(s)	33(s)	45(s)	1.01(s)
	86%	94%	95%	95%	95%
Người 9	15(s)	19(s)	24(s)	33(s)	44(s)
	93%	97%	97%	98%	99%
Người 10	19(s)	22(s)	29(s)	39(s)	51(s)
	97%	96%	94%	92%	95%

Bảng 8-3 Kết quả hệ thống nhận dạng theo mô hình Markov với số trạng thái $N = 6$

➤ Áp dụng công thức (*) chúng ta thấy rằng với bộ tham số $N = 6, M = 10$ cho kết quả nhận dạng tốt nhất với thực nghiệm 2.

➤ Thực nghiệm 3 với $N = 8$, $M = \{2, 4, 6, 8, 10\}$

Số người	M=2	M=4	M=6	M=8	M=10
Người 1	18(s)	35(s)	36(s)	53(s)	1.10(s)
	74%	81%	73%	87%	85%
Người 2	23(s)	45(s)	46(s)	1.07(s)	1.29(s)
	91%	95%	95%	95%	93%
Người 3	12(s)	24(s)	30(s)	35(s)	46(s)
	99%	98%	98%	98%	86%
Người 4	13(s)	16(s)	26(s)	38(s)	51(s)
	93%	92%	92%	88%	89%
Người 5	17(s)	33(s)	35(s)	49(s)	1.04(s)
	92%	94%	94%	92%	88%
Người 6	16(s)	30(s)	33(s)	44(s)	58(s)
	97%	98%	98%	96%	97%
Người 7	20(s)	29(s)	40(s)	59(s)	1.18(s)
	76%	83%	84%	81%	81%
Người 8	23(s)	50(s)	55(s)	1.09(s)	1.23(s)
	94%	93%	94%	94%	94%
Người 9	17(s)	33(s)	33(s)	49(s)	1.05(s)
	86%	84%	86%	84%	84%
Người 10	19(s)	38(s)	39(s)	1.00(s)	1.15(s)
	96%	93%	94%	92%	90%

Bảng 8-4 Kết quả hệ thống nhận dạng theo mô hình Markov với số trạng thái $N = 8$

➤ Áp dụng công thức (*) chúng ta thấy rằng với bộ tham số $N = 8$, $M = 4$ cho kết quả nhận dạng tốt nhất với thực nghiệm 3

➤ Thực nghiệm 4 với $N = 10$, $M = \{2, 4, 6, 8, 10\}$

Số người	M=2	M=4	M=6	M=8	M=10
Người 1	19(s)	23(s)	48(s)	1.10(s)	1.33(s)
	75%	80%	76%	72%	70%
Người 2	30(s)	60(s)	1.01(s)	1.30(s)	1.50(s)
	91%	92%	94%	93%	94%
Người 3	13(s)	20(s)	32(s)	47(s)	1.01(s)
	99%	100%	99%	98%	96%
Người 4	14(s)	22(s)	35(s)	51(s)	1.07(s)
	91%	91%	92%	85%	84%
Người 5	18(s)	27(s)	44(s)	1.05(s)	1.25(s)
	92%	92%	91%	92%	87%
Người 6	16(s)	24(s)	40(s)	58(s)	1.17(s)
	97%	97%	95%	91%	93%
Người 7	22(s)	33(s)	53(s)	1.18(s)	1.43(s)
	90%	96%	94%	97%	97%
Người 8	30(s)	40(s)	1.03(s)	1.31(s)	2.05(s)
	92%	92%	93%	89%	87%
Người 9	18(s)	27(s)	45(s)	1.05(s)	1.25(s)
	97%	99%	98%	98%	98%
Người 10	21(s)	35(s)	52(s)	1.16(s)	1.40(s)
	95%	97%	94%	95%	85%

Bảng 8-5 Kết quả hệ thống nhận dạng theo mô hình Markov với số trạng thái $N = 10$

➤ Áp dụng công thức (*) chúng ta thấy rằng với bộ tham số $N = 10$, $M = 4$ cho kết quả nhận dạng tốt nhất với thực nghiệm 4

8.2.2 Nhận xét

Đặc biệt trong tất cả các thí nghiệm này chúng tôi đã chọn được bộ tham số cho mô hình Markov mà có tỉ lệ nhận dạng khuôn mặt cao nhất trong toàn bộ chuỗi thực nghiệm là

➤ Số trạng thái $N = 6$ và số ký hiệu quan sát $M = 10$ kết quả như sau

Lớp 1	Lớp 2	Lớp 3	Lớp 4	Lớp 5	Lớp 6	Lớp 7	Lớp 8	Lớp 9	Lớp 10
85%	95%	99%	92%	95%	98%	96%	95%	99%	96%
48(s)	1.01(s)	25(s)	34(s)	43(s)	40(s)	53(s)	1.01(s)	44(s)	51(s)

Bảng 8-6 Kết quả nhận dạng tốt nhất với phương pháp mô hình Markov tại $N = 6$ và $M = 10$

8.3 Kết quả theo tiếp cận SVM

8.3.1 Thực nghiệm trên từng bộ tham số

➤ SVMs là một mô hình nhiều tham số nhưng việc chọn ra bộ tham số nào là tốt nhất thì đến bây giờ chưa ai có thể chứng minh được, cách chọn tốt nhất là dựa trên cơ sở thực nghiệm là chính. Trong cách tiếp cận nhận dạng khuôn mặt theo phương pháp này chúng tôi chú trọng vào hai tham số chính: thứ nhất, C (là giới hạn đường bao chặn trên đối với tham số α_i tương ứng với hàm phân loại f), thứ hai, dạng hàm xử lý K để ánh xạ dữ liệu từ không gian mẫu sang không gian đặc trưng mà trên đó có thể tạo ra được các tập compact đối với các khuôn mặt khác nhau từ dữ liệu tương ứng và qua đó chọn ra một bộ tham số tốt nhất mà nó có khả năng phân loại tốt các khuôn mặt của những người khác nhau trong ảnh.

- Tương ứng với mỗi tham số $C = \{30, 50, 100, 200, 400\}$ chúng ta sẽ kết hợp lần lượt với một trong bốn loại hàm $K = \{\text{Hàm tuyến tính, Hàm đa thức bậc 2, Hàm đa thức bậc 3, Hàm đa thức bậc 4}\}$ để điều khiển mô hình học SVMs.
- Mỗi mẫu khuôn mặt 30×30 (pixels) sẽ được vector hóa thành mẫu vector mà chứa những thành phần quan trọng nhất của mẫu, và phương pháp PCA đã đáp ứng được yêu cầu đó. Số thành phần chính được giữ lại sau khi thực hiện phương pháp PCA trên tập mẫu ảnh khuôn mặt sẽ là thông tin thực sự được xử lý trong hệ thống nhận dạng khuôn mặt. Và thực nghiệm cho thấy rằng nếu chúng ta giữ lại 44 thành phần đầu tiên có thể cho chất lượng nhận dạng tốt nhất.
- Kết quả thí nghiệm tương ứng với mỗi bộ tham số bao gồm $\{C, K\}$ được trình bày dưới đây:
 - Vì số ảnh khuôn mặt đưa vào hệ thống nhận dạng cho những người từ Lớp 11 đến Lớp 30 rất ít chỉ 25 ảnh thử nghiệm cho mỗi người, và hệ thống nhận dạng tốt cho các trường hợp này với độ chính xác 99% đến 100%.
 - Dữ liệu khuôn mặt của những người có mã số từ Lớp 1 đến Lớp 10 được thu thập rất phong phú và tập dữ liệu khuôn mặt của những người này sẽ đánh giá tốt hơn về mô hình SVMs đã huấn luyện cho nhận dạng khuôn mặt, và kết quả dưới đây chúng tôi chỉ trình bày kết quả từ Lớp 1 đến Lớp 10.
- Để so sánh kết quả nhận dạng từ mô hình với bộ tham số (C, K) với mô hình với bộ tham số (C', K') chúng tôi đã sử dụng một công thức tính như sau:

$$a = \sum_{i=1}^{30} [(\text{Tỉ lệ \% người } i \text{ ứng với } C-K) - (\text{Tỉ lệ \% người } i \text{ ứng}$$

với $C'-K'$)] (**)

Nếu $a > 0$ thì mô hình với bộ $C-K$ cho chất lượng nhận dạng tốt hơn.

Nếu $a < 0$ thì mô hình với bộ tham số $C'-K'$ cho chất lượng nhận dạng tốt hơn.

- Hệ số $C = 30$ và $K = \text{Phương trình}\{\text{Tuyến tính, K-đa thức bậc 2, K-đa thức bậc 2, K-đa thức bậc 2}\}$

Số người	K-tuyến tính	K- đa thức bậc 2	K-đa thức bậc 3	K-đa thức bậc 4
Người 1	11(s)607	7(s)607	10(s)695	12(s)328
	92%	92%	94%	95%

Người 2	9(s)995	10(s)645	12(s)188	14(s)240
	92%	94%	95%	95%
Người 3	6(s)219	5(s)158	5(s)868	6(s)740
	96%	98%	98%	98%
Người 4	4(s)275	5(s)558	5(s)839	7(s)681
	85%	85%	95%	97%
Người 5	7(s)70	7(s)531	8(s)662	9(s)784
	78%	89%	90%	90%
Người 6	6(s)289	6(s)650	7(s)581	8(s)572
	94%	97%	98%	98%
Người 7	8(s)562	8(s)993	10(s)305	11(s)687
	76%	89%	91%	92%
Người 8	9(s)623	10(s)215	11(s)727	13(s)129
	77%	87%	89%	90%
Người 9	6(s)710	7(s)651	8(s)632	8(s)603
	98%	99%	99%	99%
Người 10	7(s)961	8(s)222	9(s)263	10(s)345
	92%	94%	93%	92%

Bảng 8-7 Kết quả nhận dạng với phương pháp SVMs với $C = 30$

- Áp dụng công thức (**)
chúng ta thấy rằng với bộ tham số $C = 30$, và phương trình xử lý chính K là dạng đa thức bậc 4 cho kết quả nhận dạng tốt nhất với thực nghiệm 1.
- Hệ số $C = 50$
và $K =$ Phương trình {Tuyến tính, K-đa thức bậc 2, K-đa thức bậc 2, K-đa thức bậc 2}

Số người	K-tuyến tính	K- đa thức bậc 2	K-đa thức bậc 3	K-đa thức bậc 4
Người 1	14(s)160	8(s)523	10(s)785	10(s)785
	95%	92%	94%	95%
Người 2	8(s)392	10(s)736	12(s)378	14(s)310
	92%	94%	95%	95%
Người 3	4(s)156	5(s)207	5(s)878	5(s)908
	95%	98%	98%	98%
Người 4	4(s)747	5(s)869	6(s)800	7(s)681
	85%	95%	96%	97%
Người 5	5(s)999	7(s)540	8(s)753	9(s)944
	78%	86%	90%	90%
Người 6	5(s)288	6(s)619	7(s)641	8(s)652
	94%	97%	98%	98%
Người 7	7(s)210	7(s)211	10(s)325	11(s)707
	79%	79%	91%	92%
Người 8	8(s)172	10(s)294	10(s)274	13(s)249
	77%	87%	87%	90%
Người 9	5(s)558	6(s)790	7(s)721	8(s)622
	98%	99%	99%	99%
Người 10	6(s)630	8(s)212	9(s)353	10(s)405
	92%	94%	93%	92%

Bảng 8-8 Kết quả nhận dạng với phương pháp SVMs với $C = 50$

➤ Áp dụng công thức (**) chúng ta thấy rằng với bộ tham số $C = 50$, Và phương trình xử lý chính K là dạng đa thức bậc 4 cho kết quả nhận dạng tốt nhất từ chuỗi các kết quả trên thử nghiệm 2.

➤ Hệ số $C = 100$

và $K =$ Phương trình {Tuyến tính, K-đa thức bậc 2, K-đa thức bậc 2, K-đa thức bậc 2}

Số người	K-tuyến tính	K- đa thức bậc 2	K-đa thức bậc 3	K-đa thức bậc 4
Người 1	14(s)371	14(s)371	10(s)865	15(s)512
	95%	92%	94%	95%
Người 2	8(s)312	10(s)906	12(s)397	14(s)401
	92%	94%	95%	95%
Người 3	6(s)769	4(s)126	5(s)257	5(s)258
	98%	95%	98%	98%
Người 4	4(s)686	5(s)908	6(s)860	7(s)761
	88%	95%	96%	97%
Người 5	5(s)889	7(s)601	8(s)732	9(s)954
	78%	86%	90%	90%
Người 6	5(s)238	6(s)690	7(s)691	8(s)692
	94%	97%	98%	98%
Người 7	7(s)120	9(s)153	10(s)475	11(s)787
	78%	89%	91%	92%
Người 8	8(s)202	10(s)305	11(s)938	13(s)339
	77%	87%	89%	90%
Người 9	5(s)538	6(s)820	7(s)731	8(s)672
	98%	99%	99%	99%
Người 10	6(s)569	8(s)202	9(s)343	10(s)495
	93%	94%	93%	93%

Bảng 8-9 Kết quả nhận dạng với phương pháp SVMs với $C = 100$

- Áp dụng công thức (***) chúng ta thấy rằng với bộ tham số $C = 100$, Và phương trình xử lý chính K là dạng đa thức bậc 4 cho kết quả nhận dạng tốt nhất từ thực nghiệm 3.
- Hệ số $C = 200$
và $K =$ Phương trình {Tuyến tính, K-đa thức bậc 2, K-đa thức bậc 2, K-đa thức bậc 2}

Số người	K-tuyến tính	K- đa thức bậc 2	K-đa thức bậc 3	K-đa thức bậc 4
Người 1	14(s)511	12(s)608	14(s)941	14(s)601
	95%	94%	95%	95%
Người 2	8(s)583	10(s)815	12(s)638	14(s)481
	92%	94%	95%	95%
Người 3	6(s)790	4(s)106	4(s)520	5(s)939
	98%	96%	96%	98%
Người 4	4(s)666	5(s)939	6(s)880	7(s)771
	86%	95%	96%	97%
Người 5	5(s)897	7(s)942	8(s)822	9(s)914
	78%	86%	90%	90%
Người 6	5(s)227	6(s)739	7(s)741	8(s)732
	94%	97%	98%	98%
Người 7	7(s)731	9(s)123	10(s)495	11(s)837
	76%	89%	91%	92%
Người 8	8:202	10:305	11:938	13:350
	77%	87%	89%	90%
Người 9	6(s)820	7(s)771	8(s)662	8(s)713
	99%	99%	99%	99%
Người 10	6(s)579	8(s)232	9(s)313	9(s)364
	92%	94%	93%	93%

Bảng 8-10 Kết quả nhận dạng với phương pháp SVMs với $C = 200$

- Áp dụng công thức (**) chúng ta thấy rằng với bộ tham số $C = 200$, và phương trình xử lý chính K là dạng đa thức bậc 4 cho kết quả nhận dạng tốt nhất với thực nghiệm 4.
- Hệ số $C = 400$,
và $K =$ Phương trình {Tuyến tính, K -đa thức bậc 2, K -đa thức bậc 2, K -đa thức bậc 2}

Số người	K-tuyến tính	K- đa thức bậc 2	K-đa thức bậc 3	K-đa thức bậc 4
Người 1	8(s)352	10(s)996	14(s)181	15(s)272
	92%	94%	95%	95%
Người 2	8(s)362	10(s)966	12(s)558	17(s)886
	92%	94%	95%	95%
Người 3	4(s)166	5(s)247	5(s)949	5(s)948
	96%	98%	98%	98%
Người 4	4(s)737	6(s)39	6(s)940	7(s)791
	87%	95%	96%	97%
Người 5	5(s)918	7(s)621	8(s)883	10(s)24
	77%	86%	90%	90%
Người 6	5(s)218	6(s)749	7(s)771	8(s)753
	93%	97%	98%	98%
Người 7	7(s)150	9(s)453	11(s)477	12(s)378
	75%	89%	91%	92%
Người 8	8(s)202	12(s)17	14(s)281	14(s)110
	98%	99%	99%	99%
Người 9	5(s)598	6(s)870	7(s)772	8(s)682
	98%	99%	99%	99%
Người 10	6(s)549	8(s)221	9(s)443	10(s)445
	92%	94%	93%	92%

Bảng 8-11 Kết quả nhận dạng với phương pháp SVMs với $C = 400$

- Áp dụng công thức (**) chúng ta thấy rằng với bộ tham số $C = 400$, và phương trình xử lý chính K là dạng đa thức bậc 3 cho kết quả nhận dạng tốt nhất với thực nghiệm 5.

8.3.2 Nhận xét

Từ chuỗi thực nghiệm này chúng tôi đã chọn ra 4 bộ tham số điều khiển mô hình SVMs mà cho ra các kết quả nhận dạng tốt nhất như sau

- $C = 400$ và phương trình xử lý chính K là dạng đa thức bậc 3

Lớp 1	Lớp 2	Lớp 3	Lớp 4	Lớp 5	Lớp 6	Lớp 7	Lớp 8	Lớp 9	Lớp 10
95%	95%	98%	96%	90%	98%	91%	99%	99%	93%
14(s)20	12(s)558	6(s)	7(s)	8(s)883	7(s)771	11(s)47	14(s)281	7(s)772	9(s)443

Bảng 8-12 Kết quả nhận dạng tốt nhất với phương pháp SVMs tại $C = 400$ và K là hàm xử lý chính dạng đa thức bậc 3

Từ chuỗi thí nghiệm này chúng ta có thể rút ra được vai trò quan trọng của từng tham số trong việc điều khiển tính tổng quát vốn có của phương pháp xử lý SVMs. Từ kết quả nhận dạng khuôn mặt trên mỗi bảng mà có cùng tham số C, và hàm xử lý K càng phức tạp thì tỉ lệ nhận dạng đúng trên mỗi người sẽ tăng lên đáng kể. Và quan sát trên các bảng với cùng hàm xử lý chính K, nhưng với các giá trị khác nhau của C thì kết quả nhận dạng trên mỗi người không thay đổi.

8.4 So sánh kết quả HMM và SVM

- Công thức đánh giá hai phương pháp dựa trên tiêu chuẩn chất lượng (Tỉ lệ %) và tiêu chuẩn về thời gian xử lý

$$a = \sum_{i=1}^{30} [(Tỉ\ lệ\ \% \text{ người } i \text{ ứng với SVM}) - (Tỉ\ lệ\ \% \text{ người } i \text{ ứng với HMM})]$$

(***)

Nếu $a > 0$ thì mô hình với phương pháp SVMs tốt hơn.

Nếu $a < 0$ thì mô hình với phương pháp HMM tốt hơn.

$$t = \sum_{i=1}^{30} [(Thời\ gian\ người\ i\ ứng\ với\ SVM) - (Thời\ gian\ người\ i\ ứng\ với\ HMM)]$$

(****)

Nếu $t > 0$ thì mô hình với SVMs kém hơn.

Nếu $t < 0$ thì mô hình với HMM kém hơn.

- Theo cách tiếp cận SVM chúng ta đã chọn được bộ tham số có tỉ lệ nhận dạng cao nhất đó là khi $C = 10$, K là dạng phương trình phi tuyến bậc 3 và với kết quả như sau:

Lớp 1	Lớp 2	Lớp 3	Lớp 4	Lớp 5	Lớp 6	Lớp 7	Lớp 8	Lớp 9	Lớp 10
95%	95%	98%	96%	90%	98%	91%	99%	99%	93%
14(s)20	12(s)558	6(s)	7(s)	8(s)883	7(s)771	11(s)47	14(s)281	7(s)772	9(s)443

- Theo cách tiếp cận HMM chúng ta đã chọn được bộ tham số có tỉ lệ nhận dạng cao nhất đó là khi $N = 6$, $M = 10$ và với kết quả như sau:

Lớp 1	Lớp 2	Lớp 3	Lớp 4	Lớp 5	Lớp 6	Lớp 7	Lớp 8	Lớp 9	Lớp 10
85%	95%	99%	92%	95%	98%	96%	95%	99%	96%
48(s)	1:01(s)	25(s)	34(s)	43(s)	40(s)	53(s)	1:01(s)	44(s)	51(s)

- Từ Lớp 11 (Người 11) đến Lớp 30 (Người 30) thì kết quả nhận dạng theo hai phương pháp SVMs và HMM là xấp xỉ nhau (tỉ lệ nhận dạng chênh lệch 1%).

- Nhận xét:

- Với cùng tập mẫu thử nghiệm kết quả tốt nhất theo hai phương pháp SVMs và HMM là tương đương nhau. Tỉ lệ nhận dạng chính xác trung bình theo SVM là 95% và tỉ lệ nhận dạng chính xác trung bình theo HMM là 95%. Áp dụng công thức (***) từ dữ liệu Bảng 8-6 và Bảng 8-12 thì $a \approx 0$.
- Nhưng khi xét về thời gian xử lý trên cùng tập mẫu nhận

dạng theo hai phương pháp và dùng công thức (****) từ dữ liệu Bảng 8-6 và Bảng 8-12 ta thấy rằng $t = -355(s)$ xấp xỉ -5 phút. Có nghĩa chi phí nhận dạng HMM là quá cao so với phương pháp SVMs.

- Trong hai cách tiếp cận cho hệ thống nhận dạng khuôn mặt chúng tôi đánh giá phương pháp SVMs là tốt hơn so với cách tiếp cận HMM.

Khoa CNTT - ĐH KHTN TP.HCM

Chương 9 NHẬN XÉT VÀ HƯỚNG PHÁT
TRIỂN

Khoa CNTT - ĐH KHTN TP.HCM

9.1 Thuận lợi

- Khi xây dựng hệ thống nhận dạng này chúng tôi đã tiếp cận với những phương pháp nhận dạng được coi là tốt nhất hiện nay trong lĩnh vực trí tuệ nhân tạo
- Chương trình nhận dạng với độ chính xác trung bình 95% so với phương pháp SVMs và 95% với phương pháp HMM.
- Cơ sở dữ liệu cho việc thử nghiệm hệ thống nhận dạng được thu thập từ nhiều nguồn khác nhau nhưng nhìn chung chất lượng dữ liệu khuôn mặt rất phù hợp với những yêu cầu đặt ra từ đề tài (ảnh hưởng của môi trường và sự biến đổi trên nội tại khuôn mặt không quá lớn lên tập mẫu học).
- Thiết bị sử dụng việc cho việc nghiên cứu và thực hiện đề tài rất tốt: Máy Pentium III 600, RAM 384M.
- Dù SVMs là một phương pháp học mới và mục đích của nó chuyên cho ứng dụng phân loại mẫu hai lớp bằng một siêu mặt phân tách, nhưng với kết quả đạt được như trên từ SVMs đã chứng tỏ được khả năng giải quyết vấn đề không kém so với một số phương pháp cổ điển như mạng neural nhân tạo hay mô hình Markov ẩn rời rạc mà có khả năng phân loại cùng một lúc nhiều lớp rất tốt.
- HMM là một phương pháp được coi là độc quyền cho nhận dạng tiếng nói trong việc mô hình hóa tín hiệu quan sát bằng một con số theo mô hình thống kê, nhưng với thí nghiệm trên đây thì HMM cũng phù hợp với lĩnh vực xử lý ảnh và thị giác máy tính.

- Khi nghiên cứu về hệ thống nhận dạng này chúng tôi không những nghiên cứu về thuật toán xử lý mà qua đó chúng tôi đã biết thêm nhiều về các công trình nghiên cứu về khuôn mặt người từ các lĩnh vực khác nhau và kết quả này thực sự có ý nghĩa trong việc nâng cao hiệu quả khả năng giao tiếp giữa con người và máy tính.
- Hệ thống này không những chú trọng vào kỹ thuật nhận dạng mà còn chú trọng vào kỹ thuật để nhận thức một mẫu là một khuôn mặt người nhờ đó mà chúng tôi đã xây dựng tương đối thành công một hệ thống nhận dạng khuôn mặt tự động.

9.2 Khó khăn

- Hệ thống nhận dạng với cách tiếp cận từ các phương pháp học rất mạnh, nhưng tính chặt chẽ về mặt toán học chưa cao, nhiều trường hợp tối ưu hóa đòi hỏi chúng ta phải thử nghiệm nhiều lần với nhiều bộ tham số khác nhau.
- Cơ sở dữ liệu còn rất hạn chế, chỉ mới thử nghiệm trên 30 người, đồng thời số lượng ảnh của mỗi người không có quá nhiều biến động lớn và đặc biệt là điều này có thể làm giảm chất lượng của hệ thống nhận dạng khi áp dụng trong thế giới thực mà trong đó khuôn mặt và môi trường thay đổi liên tục.
- Để đánh giá chất lượng của một hệ thống chúng tôi đã sử dụng một số độ đo nhưng thực sự vẫn chưa đảm bảo đánh giá được một cách khách quan và đôi lúc còn rất cảm tính.

9.3 Hướng phát triển tương lai

1. Định hướng tương lai về mặt kỹ thuật xử lý

Đây là một đề tài tương đối phức tạp, mặc dù chúng tôi đã cố gắng xây dựng một hệ thống có khả năng nhận dạng người dựa vào thông tin khuôn mặt xuất hiện trên ảnh tốt nhất. Nhưng với những hạn chế về mặt trình độ cũng như thời gian thực hiện, chúng tôi kết thúc đề tài với nghiên cứu về một mảng nhỏ trong lĩnh vực thị giác máy tính và những định hướng dưới đây sẽ hy vọng ngày càng hoàn thiện khả năng tương tác giữa người với máy tính qua hệ thống nhận dạng khuôn mặt.

- Đối với dữ liệu học: Cần phải tăng dần số mẫu học cũng như số người trong hệ thống nhận dạng, thỏa điều kiện này mới có thể nâng cao được khả năng nhận dạng tổng quát cho hệ thống, cũng như đáp ứng được nhu cầu thực tế từ thế giới thực mà số người sẽ phân loại có thể lên đến hàng ngàn, hàng triệu và hàng tỉ người.
- Đối với việc dò tìm khuôn mặt, đây điều kiện tiên quyết khi đi xây dựng một hệ thống nhận dạng khuôn mặt tự động: Cần phải nghiên cứu phương pháp dò tìm khuôn mặt có khả năng chống chịu tốt với những ảnh hưởng của môi trường và đặc biệt phải giải quyết bài toán dò tìm khuôn mặt xuất hiện ở nhiều góc độ khác nhau so với mặt phẳng ảnh để đảm bảo không bỏ sót những người cần nhận dạng.
- Đối với việc rút trích đặc trưng khuôn mặt: Cần phải tìm hiểu và nghiên cứu nhiều phương pháp rút trích đặc trưng khuôn mặt hiệu quả hơn để có thể bóc được thông tin nổi thuộc về khuôn mặt và có khả năng loại trừ tạp nhiễu nhiều nhất. Thông tin này phải đảm bảo thông tin tổng quát thuộc về một người duy nhất.
- Đối với việc nhận dạng:
 - ✓ Cần tìm hiểu kỹ hơn về bản chất của không gian nhận thức khuôn mặt người và cần trang bị những kiến thức về tâm lý học, sinh lý học, vật lý học cũng như khoa học về nhận thức khuôn mặt để có được một định hướng tốt hơn cho việc nhận dạng và tăng hiệu suất cho nhận dạng mà chỉ dựa vào tri thức từ khuôn mặt.
 - ✓ Cần xây dựng những hệ thống lai tạo bằng việc kết hợp nhiều phương pháp nhận dạng khác nhau để tăng ưu điểm cho mục đích nhận dạng và cũng giảm khuyết điểm của mỗi phương pháp nhận dạng riêng. Hiện này SVMs được kết hợp với nhiều phương pháp

khác như HMM, ANN, Fuzzy đã nâng cao khả năng phân loại đáng kể.

- ✓ Nhận dạng khuôn mặt thay đổi nhiều hướng nhìn khác nhau, có nhiều biến đổi từ môi trường ngoài cũng như những biến đổi bên trong nội tại của một khuôn mặt.
- ✓ Nghiên cứu cấu trúc giải phẫu của các cơ trên khuôn mặt và tìm hiểu các phương pháp nhận dạng khuôn mặt chuyển động trên video, ảnh động..vv.
- ✓ Một hệ thống nhận dạng hoàn chỉnh phải có khả năng phân biệt tính đơn nhất các các trung có thể có từ một người: giọng nói, vẫy tay, vẫy tay, dáng đi, dáng đứng..vv.
- Cần chú trọng xây dựng kỹ thuật xử lý tăng cường cho các thuật toán xử lý chính của chương trình, SVM, ANN và mô hình Markov ẩn. Bởi vì đặc trưng xử lý của hệ thống nhận dạng chúng tôi dựa trên cơ sở máy học là chính. Nhiệm vụ trong thời gian tới của chúng tôi sẽ cải tiến thuật toán sao cho khi bổ sung dữ liệu vào hệ thống thì chúng tôi không phải dạy lại cho máy mất nhiều giờ như hiện nay.

2. Định hướng tương lai về mặt ứng dụng

- Việc xây dựng hệ thống nhận dạng người trên ảnh tĩnh còn rất nhiều bất lợi, Mặc dù lượng thông tin trên ảnh rất phong phú nhưng khối lượng tri thức cần thiết cho mục đích xử lý của hệ thống còn hạn chế. Chúng tôi đang định hướng chuyên ứng dụng xử lý trên ảnh tĩnh sang ứng dụng xử lý trên video số.
- Xây dựng hệ thống bảo vệ quyền lợi truy cập vào/ra ở các cơ quan chính phủ ta, hay ở các cảng Hàng không sân bay..vv.
- Sở sở lý thuyết từ công trình nghiên cứu này tiếp tục làm cơ sở để xây dựng nhiều trình ứng dụng khác trong lĩnh vực máy tính. Kế hoạch mới trong tương lai chúng tôi có dự định nghiên cứu về ảnh Viễn thám một lĩnh vực hiện rất mới ở Việt Nam. Và đặc biệt theo chương trình từ bộ Công nghệ và Môi trường khoảng 2005 chúng ta sẽ phóng một vệ tinh viễn thám đầu tiên lên quỹ đạo, và chúng tôi cho rằng khoảng thời gian bây giờ đến 2005 là phù hợp cho việc chuẩn bị để chúng ta có thể khai thác tốt nhất nguồn tài nguyên này và chắc chắn đây là một nguồn lợi không nhỏ đối với sự phát triển của quốc gia trong tương lai.

Tổng kết

Khoa học phát triển không ngừng, những bước đi đầu tiên bao giờ cũng chập chững. Chúng tôi hy vọng trong một tương lai không xa khi mà những hệ thống nhận dạng của chúng ta đã đạt đến một độ tin cậy nhất định thì những ứng dụng về nó sẽ rộng rãi. Và những nghiên cứu tiếp theo từ lĩnh vực này sẽ được nâng lên ở một cấp độ cao hơn, hoàn thiện hơn.

Khoa CNTT - ĐH KHTN TP.HCM

TÀI LIỆU THAM KHẢO

Chương 1:

[1] Wenyi Zhao, Arvinth Krishnaswamy, Rama Chellappa, Danie L.Swets, John Weng, Discriminant Analysis of Principal Components for Face Recognition, Centre for Automation Research, University of Maryland (page 73) of Face Recognition from Theory and Applications.

[2] John Daugman, Phenotypic versus Genotypic Approches to Face Recognition, University of Cambridge, the Computer laboratory Cambridge CB3 3QG England (page 108) of Face Recognition from Theory and Applications.

[3] Emmanuel Viennet và Françoise Fogelman Soulie, Connectionists Methods for Human face Processing, University Paris 13, 93430 Villetaneuse, France (page 124) of Face Recognition from Theory and Applications.

[4] Antonio J.Colmenarez và Thomas S.Huang, Face Detection and Recognition, Department of Electrical and Computer Engineering, Coordinated Science Laboratory, and Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign, 405 N. Mathews Ave, USA (page 174) of Face Recognition from Theory and Applications.

[5] Kazunori Okada, Johannes Steffens, Thomas Maurer, Hai Hong, Egor Elagin, Hartmut Neven, and Christoph, Computer Science Department and Center for Neural Engineering, University of Southern California Los Angeles, USA (page 286) of Face Recognition from Theory and Applications.

- [6] Baback Moghaddam và Alex Pentland, Beyond Linear Eigenspaces: Bayesian Matching for Face Recognition, MIT Media Laboratory, 20 Ames St. Cambridge, MA 02139, USA (page 230) of Face Recognition from Theory and Applications.
- [7] Massimo Tistaelli và Enrico Grosso, Active Vision-base Face Recognition: Issues, Application and Techniques, University of Genoa, Department of Communication, Computer and System Science (DIST), Computer Vision Laboratory, Via Opera Pia 13-16145 Genoa, Italy (page 262) of Face Recognition from Theory and Applications.
- [8] Jeffrey Huang, Chengjun Liu, và Harry Wechsler, Eye Detection and Face Recognition Using Evolutionary Computation, Department of Computer Science, George Mason University, <http://www.chagall.gmu.edu/> (page 348) of Face Recognition from Theory and Applications.
- [9] Daniel Bgraham và Nigel M Allinson, Characterising Virtual Eigensignatures for General Purpose Face Recognition, Image Engineering and Neural Computing Group, Department of Electrical Engineering and Electronics, University of Manchester Institute of Science and Technology, Manchester M60 1QD, UK (page 446) of Face Recognition from Theory and Applications.
- [10] Jason M.Kinser, Pulse Images for Face Recognition, Institute for Biosciences, Bioinformatics, and Biotechnology, George Masson, University, Fairfax, VA 22030-4444 (page 503) of Face Recognition from Theory and Applications.
- [11] Oi Bin Sun, Chian Prong Lam và Jian Kang Wu, Using Differential Constraints to Generate a 3D Face Model from Stereo, Computer Graphics Laboratory (LIG), EPEL, CH-1015, Lausanne, SWITZER-LAND (page 556) of Face Recognition from Theory and Applications.
- [12] Fabien Cardinaux, Automatic Face Recognition in Weakly Constrained Environment, Phd Student – Vision Group, Dalle Molle Institute for Perceptual Artificial Intelligence, 21/05/2003.
- [13] Sanjiy K.Bhatia, Vasudevan Lakshininarayanan, Ashok Samal, Grant V.Welland, Parameters for Human Face Recognition, Department of Mathematics & Computer Science, School of Optometry and Department of Physics & Astronomy University of Missouri-St. Lonis. St. Lonis.MO 63121. Department of Computer Science & Engineering University of Nebraska – Lincoln Lincoln. NE 63588-0115, May 18, 1994.

Chương 3:

- [1] Henry A.Rowley, Neural Network-Based Face Detection, School of Computer

Science, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213. May 1999 CMU-CS 99-117.

[2] Henry A. Rowley, Shumeet Baluja, Takeo Kanade, Rotation Invariant Neural Network-Based Face Detection, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, Justsystem Pittsburgh Research Center, 4616 Henry Street, Pittsburgh, PA, 15213.

Khoa CNTT - ĐH KHTN TP.HCM

Chương 4:

- [1] GS.TSKH Nguyễn Kim Sách, Chương 4: Nén tín hiệu Video và Audio số, 4.3-Mã hoá DCT, Truyền hình số có nén và Multimedia, Nhà xuất bản khoa học và kỹ thuật
- [2] LK. JAIN, Chapter 5: IMAGE TRANSFORMS, 5.1-The Cosine Transform, Fundamentals Digital Image Processing, University of California, Davis. 1989 by Prentice-Hall, Inc, Englewood Cliffs, New Jersey 07632.
- [3] Steven W.Smith, Chapter 27: Data compression, The Scientist and Engineer's Guide to Digital Signal Processing, California Technical Publishing San Diego, California.

Chương 5:

- [1] Vladimir N.Vapnik, The Nature of Statistical Learning Theory, AT&Labs-Research, London University.
- [2] Vladimir N.Vapnik, Statistical Learning Theory, AT&Labs-Research, London University.
- [3] Emmanuel Viennet và Françoise Fogelman Soulie, Connectionist Methods for Human Face Processing, (viennet@lipn.univ-paris13.fr).
- [4] Jeffrey Huang, David Li, Xuhui Shao, và Harry Wechsler, Pose Discrimination và Eye Detection Using Support Vector Machines (SVM), Department of Computer Science George Mason University
- [5] Christopher J.C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, (<http://citeseer.nj.nec.com/burges98tutorial.html>)

[6] Võ Thị Mỹ Ngọc, SVM - ứng dụng lọc email, Luận văn cao học, Chuyên ngành: Công nghệ thông tin, Trường đại học khoa học tự nhiên. TP.Hồ Chí Minh 2002.

[7] Guodong Guo*, Stan Z. Li, Kap Luk Chan. Support Vector machines for face recognition. School of Electrical and Electronic, Nanyang Technology University, Singapore, Singapore 639798. Received 6 April 2000; accepted 17 January 2001.

Chương 6:

[1] Phạm Nam Trung, Lâm Chí Tín. Mô hình Markov ẩn cài đặt và thực nghiệm, Bộ môn công nghệ tri thức, Khoa công nghệ thông tin, Trường Đại Học Khoa Học Tự Nhiên-2001.

[2] Nguyễn Đức Hoàng Hạ, Cao Xuân Trường. Hỗ trợ trẻ khiếm thính luyện âm tập nói, Bộ môn công nghệ tri thức, Khoa công nghệ thông tin, Trường Đại Học Khoa Học Tự Nhiên-2002.

[3] Dr.Joseph Picone, Fundamentals of Speech recognition, Institute of Signal and Information Processing, Department of Electrical and Computer Engineering Mississippi State University.

[4] Ara V.Nefian and Monson H.Hayes III, Face detection and recognition using Hidden Markov Models, Center for Signal and Image Processing, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta.

[5] Ara V.Nefian and Monson H.Hayes III, Hidden Markov Models for face recognition, Center for Signal and Image Processing, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta.