

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Digital Image Processing



SVM and Pattern Recognition System

GV: PGS.TS. Nguyễn Thị Hoàng Lan

Sinh viên thực hiện:

Nguyễn Quang Huy

Nguyễn Bá Hùng

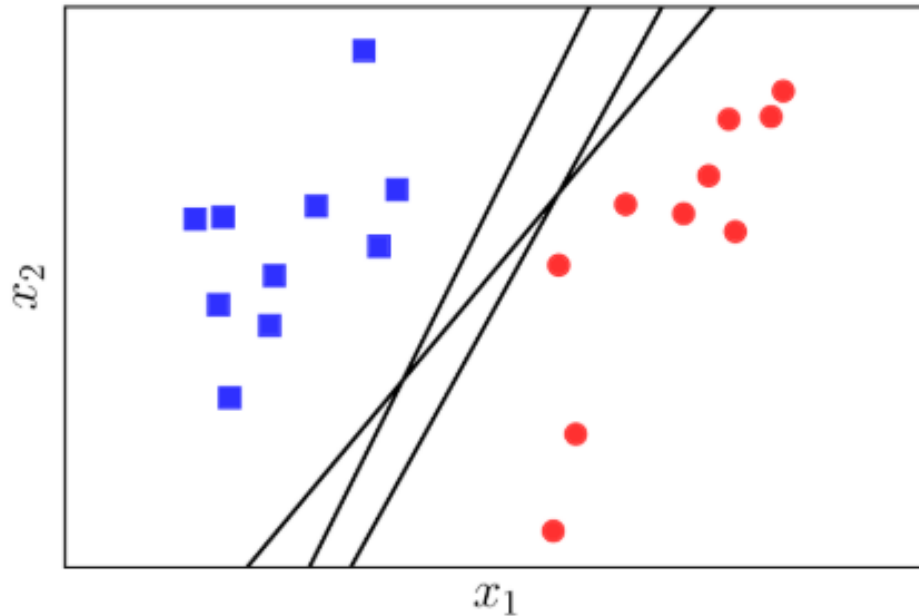
Lớp: KSTN-CNTT-K60



Nội dung

1. SVM và SVM phân lớp tuyến tính
2. Sơ đồ khối hệ thống nhận dạng
3. Vai trò SVM trong hệ thống nhận dạng
4. Hệ thống nhận dạng khuôn mặt sử dụng SVM
5. Cài đặt thử nghiệm
6. Đánh giá ảnh hưởng SVM đến hệ thống nhận dạng khuôn mặt

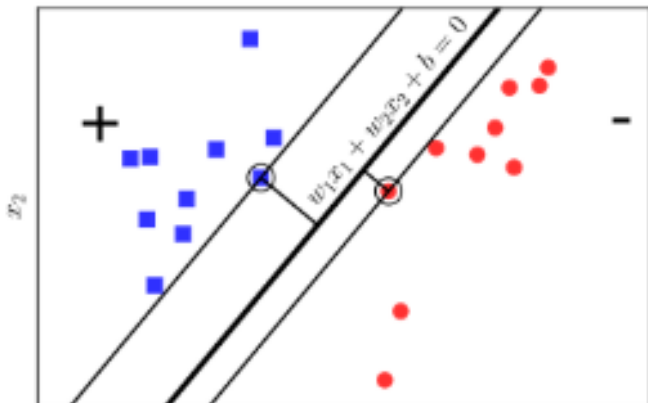
1. SVM và SVM phân lớp tuyến tính



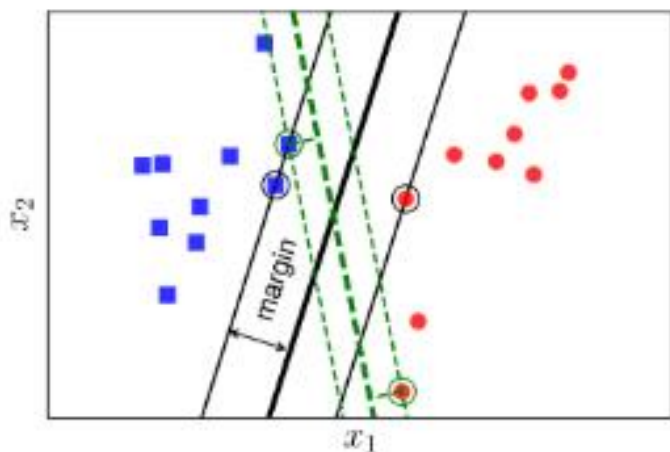
Đường nào
phân chia 2 lớp
tốt nhất?



Giới thiệu SVM



Khoảng cách từ điểm gần nhất của mỗi class (các điểm được khoanh tròn) tới đường phân chia là như nhau \rightarrow công bằng



Việc *margin* rộng hơn sẽ mang lại hiệu ứng phân lớp tốt hơn vì sự phân chia giữa hai classes là rạch ròi hơn



Khoảng cách 1 điểm tới một mặt phẳng

- Khoảng cách từ 1 điểm có tọa độ (x_0, y_0) tới đường thẳng có phương trình $w_1x + w_2y + b = 0$:

$$\frac{|w_1x_0 + w_2y_0 + b|}{\sqrt{w_1^2 + w_2^2}}$$

- Khoảng cách từ một điểm có tọa độ (x_0, y_0, z_0) tới một mặt phẳng $w_1x + w_2y + w_3z + b = 0$:

$$\frac{|w_1x_0 + w_2y_0 + w_3z_0 + b|}{\sqrt{w_1^2 + w_2^2 + w_3^2}}$$

- Không gian nhiều chiều: Khoảng cách từ một điểm (vector) có tọa độ x_0 tới siêu mặt phẳng $w^T x + b = 0$:

$$\frac{|w^T x_0 + b|}{\|w\|_2}$$

$$\|w\|_2 = \sqrt{\sum_{i=1}^d w_i^2}$$

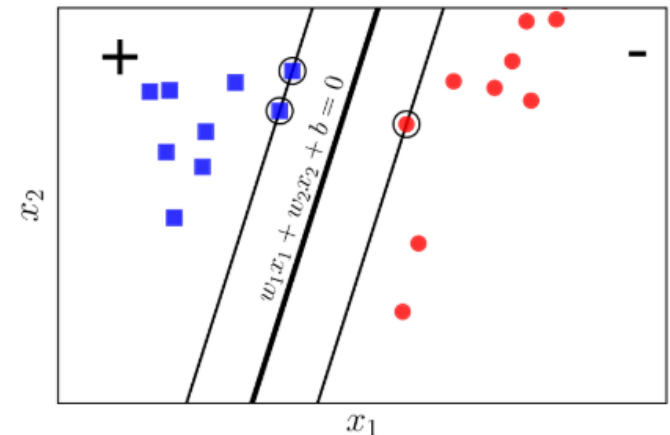
Phương pháp SVM thuần

- Giả sử các cặp dữ liệu $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$
 - $x_i \in \mathbb{R}^d$ thể hiện là đầu vào của một dữ liệu
 - y_i là nhãn của dữ liệu đó.
 - d là số chiều của dữ liệu
 - N là số điểm dữ liệu.
- Các điểm vuông xanh thuộc class 1 ($y_1 = 1$)
Các điểm tròn đỏ thuộc class -1 ($y_2 = -1$).
Mặt phẳng $\mathbf{w}^T \mathbf{x} + b = w_1 x_1 + w_2 x_2 + b = 0$
Khoảng cách từ (x_n, y_n) tới mặt phân chia:

$$\frac{y_n(\mathbf{w}^T \mathbf{x}_n + b)}{\|\mathbf{w}\|_2}$$

- Khoảng cách từ một điểm gần nhất tới mặt đó:

$$\text{margin} = \min_n \frac{y_n(\mathbf{w}^T \mathbf{x}_n + b)}{\|\mathbf{w}\|_2}$$

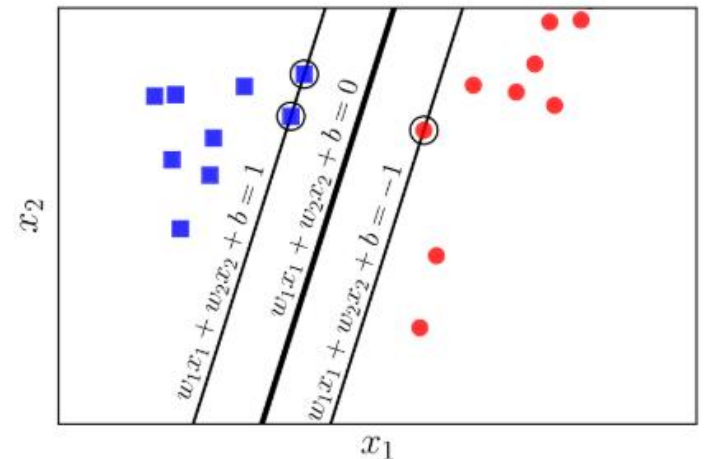


Phương pháp SVM thuần

- Bài toán tối ưu trong SVM là tìm w và b sao cho margin này đạt giá trị lớn nhất:

$$(\mathbf{w}, b) = \arg \max_{\mathbf{w}, b} \left\{ \min_n \frac{y_n(\mathbf{w}^T \mathbf{x}_n + b)}{\|\mathbf{w}\|_2} \right\} = \arg \max_{\mathbf{w}, b} \left\{ \frac{1}{\|\mathbf{w}\|_2} \min_n y_n(\mathbf{w}^T \mathbf{x}_n + b) \right\}$$

- Giả sử thay $w = kw$ và $b = kb \rightarrow$ mặt phân chia không thay đổi \rightarrow khoảng cách từng điểm đến mặt phân chia không đổi \rightarrow margin không đổi. Giả sử: $y_n(\mathbf{w}^T \mathbf{x}_n + b) = 1$ với những điểm nằm mặt phân chia nhất.



- Với mọi n :

$$y_n(\mathbf{w}^T \mathbf{x}_n + b) \geq 1$$

- Bài toán tối ưu:

$$(\mathbf{w}, b) = \arg \max_{\mathbf{w}, b} \frac{1}{\|\mathbf{w}\|_2}$$

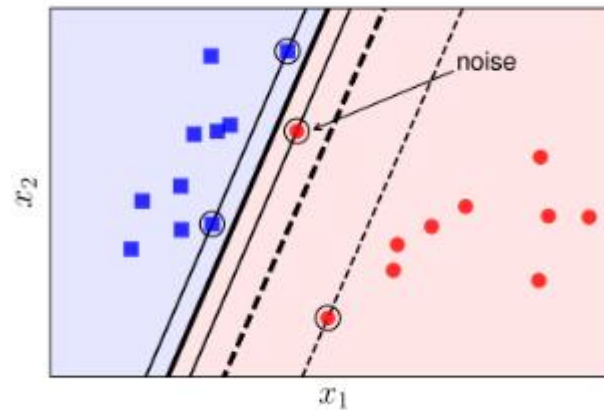
$$\text{subject to: } y_n(\mathbf{w}^T \mathbf{x}_n + b) \geq 1, \forall n = 1, 2, \dots, N$$

$$\rightarrow (\mathbf{w}, b) = \arg \min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|_2^2$$

$$\text{subject to: } 1 - y_n(\mathbf{w}^T \mathbf{x}_n + b) \leq 0, \forall n = 1, 2, \dots, N$$

Soft margin SVM

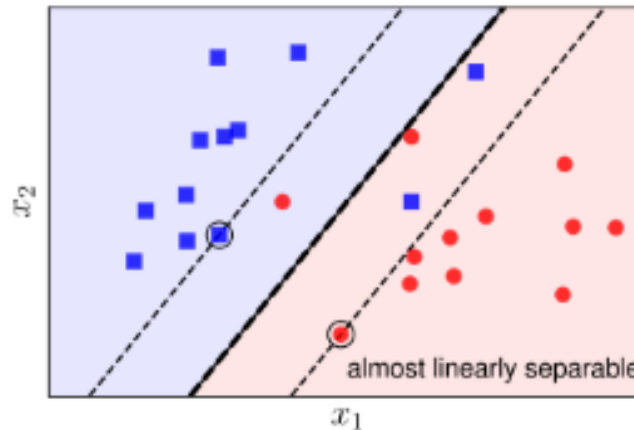
- Có 2 trường hợp: TH1



- Dữ liệu vẫn phân chia tuyến tính nhưng có 1 số điểm nhiễu của lớp tròn nhỏ ở quá gần so với lớp vuông xanh.
 - Nếu sử dụng SVM thuần sẽ tạo ra 1 margin rất nhỏ.
 - Nếu hi sinh điểm nhiễu này thì sẽ có margin tốt hơn nhiều.

Soft margin SVM

- Có 2 trường hợp: TH2



- Dữ liệu không phân chia tuyến tính nhưng gần giống với phân chia tuyến tính.
 - Nếu sử dụng SVM thuần thì vô nghiệm
 - Nếu hi sinh những điểm gần biên giữa hai classes, thì vẫn tạo được một đường phân chia khá tốt như đường nét đứt đậm.

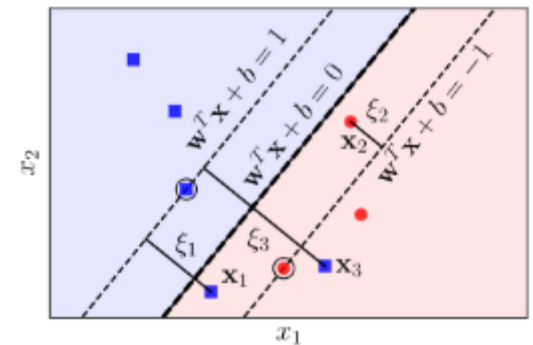
Soft margin SVM

■ Ý tưởng:

- Hy sinh một vài điểm dữ liệu bằng cách chấp nhận cho chúng rơi vào vùng không an toàn.
- Hạn chế sự hy sinh này
- Hàm mục tiêu là sự kết hợp tối đa margin và tối thiểu sự hi sinh.

■ Giới thiệu biến Slack:

- Với những điểm nằm trong vùng an toàn $\xi_n=0$
- Những điểm nằm trong vùng không an toàn nhưng vẫn đúng phía với đường phân chia tương ứng với các $0 < \xi_n < 1$ ví dụ x_2 .
- Những điểm nằm ngược phía class so với đường biên ứng $\xi_n > 1$, $\xi_i = |w^T x_i + b - y_i|$. ví dụ như x_1, x_3



■ Hàm mục tiêu:

$$\frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{n=1}^N \xi_n$$

trong đó C là một hằng số dương và $\xi = [\xi_1, \xi_2, \dots, \xi_N]$.



Soft margin SVM

- Điều kiện ràng buộc miền:

$$y_n(\mathbf{w}^T \mathbf{x}_n + b) \geq 1 - \xi_n \Leftrightarrow 1 - \xi_n - y_n(\mathbf{w}^T \mathbf{x}_n + b) \leq 0, \quad \forall n = 1, 2, \dots, n$$

Và ràng buộc phụ $\xi_n \geq 0, \quad \forall n = 1, 2, \dots, N$.

- Bài toán tối ưu dạng chuẩn cho Soft-margin SVM:

$$(\mathbf{w}, b, \xi) = \arg \min_{\mathbf{w}, b, \xi} \frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{n=1}^N \xi_n$$

$$\text{subject to:} \quad \begin{aligned} 1 - \xi_n - y_n(\mathbf{w}^T \mathbf{x}_n + b) &\leq 0, \quad \forall n = 1, 2, \dots, N \\ -\xi_n &\leq 0, \quad \forall n = 1, 2, \dots, N \end{aligned}$$

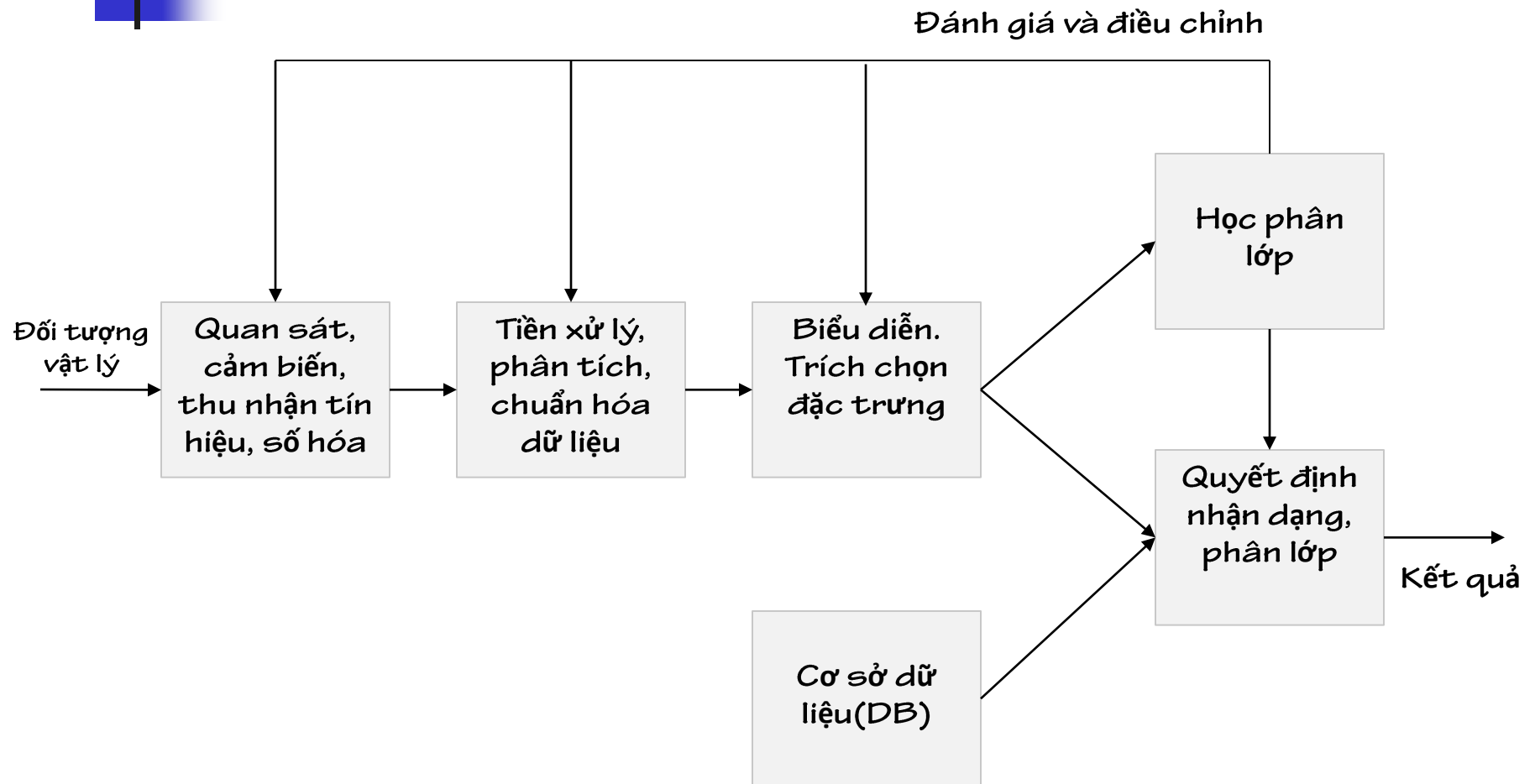
- Chúng ta có thể giải quyết bằng 2 cách:
 - Sử dụng đối ngẫu Lagrange
 - Hệ điều kiện KKT



SVM cho vấn đề nhận dạng đa lớp

- Kết hợp nhiều SVM hai lớp lại với nhau từng đôi một
- Chiến lược:
 - Giả sử cần nhận dạng s lớp thì sử dụng s SVM.
 - SVM thứ i sẽ phân biệt lớp i với lớp còn lại (nếu i là lớp 1 thì lớp còn lại là -1).
 - Hàm nhận dạng của SVM thứ i nào cho ra kết quả bé nhất thì mẫu thử thứ x thuộc lớp thứ i đó.
- Ý tưởng:
 - Ban đầu sẽ phân biệt 1 lớp, có 2 khả năng:
 - Là lớp đó
 - Không phải là lớp đó (1)
 - Tìm mặt phẳng phân cách giữa lớp đó và không phải là lớp đó.
 - Xét tiếp lớp thứ 2 với dữ liệu (1) ở trên, có 2 khả năng:
 - Là lớp 2
 - Không phải là lớp 2
 - Tìm mặt phẳng phân cách giữa lớp 2 và không phải là lớp 2.
 - ...

2. Sơ đồ khối hệ thống nhận dạng





3. Vai trò của SVM trong hệ thống nhận dạng

- SVM là một phương pháp học phân lớp
- Được sử dụng để phân lớp các đối tượng từ tập dữ liệu học
- Tìm ra mặt siêu phẳng phân lớp các đối tượng, làm cơ sở cho việc quyết định nhận dạng
- Huấn luyện một lần, lưu trữ các tham số tìm được, áp dụng nhanh chóng để đưa ra quyết định nhận dạng



4. Hệ thống nhận dạng khuôn mặt sử dụng SVM

- Quan sát cảm biến, thu nhận tín hiệu, số hóa
- Tiền xử lý, phân tích, chuẩn hóa dữ liệu
 - Phát hiện và phân tách khuôn mặt, lấy vùng ảnh chứa khuôn mặt
- Biểu diễn. Trích chọn đặc trưng
 - Biểu diễn ảnh là một vector trong tọa độ không gian nhiều chiều, mỗi chiều biểu diễn một thuộc tính có giá trị là giá trị điểm ảnh của mỗi pixel
 - Xây dựng tập dữ liệu từ nhiều ảnh đã được gán nhãn các đối tượng
 - Trích chọn đặc trưng: Dùng phương pháp PCA để trích chọn và giữ lại k thành phần khác nhau nhất giữa các ảnh, lưu lại ma trận phép biến đổi



4. Hệ thống nhận dạng khuôn mặt sử dụng SVM

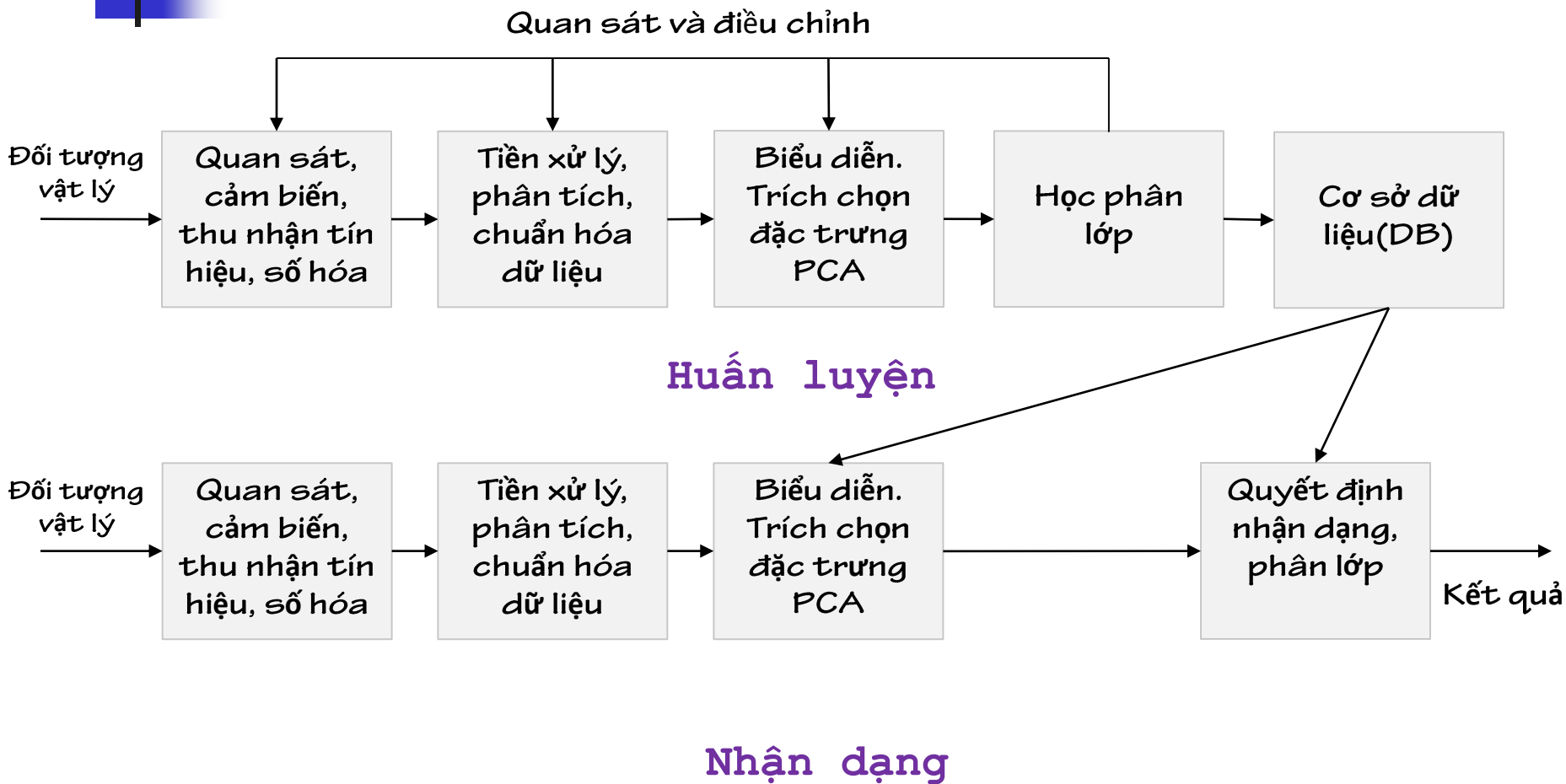
➤ Học phân lớp

- Thực hiện phân lớp các đối tượng bằng phương pháp SVM
- Lưu lại mặt siêu phẳng tìm được sau khi phân lớp các đối tượng

➤ Quyết định nhận dạng, phân lớp

- Với 1 ảnh, chuẩn hóa dữ liệu, kết hợp với tập dữ liệu học, sử dụng PCA để trích chọn ra k thành phần đặc trưng của ảnh
- Sử dụng mặt siêu phẳng tìm được để đánh giá và đưa ra kết quả

4. Hệ thống nhận dạng khuôn mặt sử dụng SVM





5. Cài đặt thử nghiệm

- Sử dụng một số thư viện python: *sklearn*, *matplotlib*, *numpy*, *joblib*,...
- Thu thập tín hiệu, số hóa
- Thực hiện phát hiện khuôn mặt, lấy vùng ảnh chứa khuôn mặt, kích thước 64x64 pixel
- Nhận diện 5 người khác nhau, thu thập 201 ảnh
- Chuyển từ ảnh màu sang ảnh đa mức xám, biểu diễn mỗi ảnh là 1 vector 4096(64x64) chiều, gán nhãn cho mỗi ảnh
- Xây dựng tập dữ liệu ma trận kích thước 201x4096, mỗi hàng là 1 vector biểu diễn 1 ảnh



5. Cài đặt thử nghiệm

- Thực hiện phương pháp PCA, chọn $k=120$, thu được tập dữ liệu kích thước 201×120 , lưu lại ma trận phép biến đổi PCA
- Thực hiện phân lớp bằng SVM từ tập dữ liệu thu được
- Lưu lại các tham số mặt siêu phẳng thu được trong file
- Thực hiện nhận dạng khuôn mặt
 - Với 1 ảnh đầu vào, thực hiện tiền xử lý, chuẩn hóa dữ liệu
 - Sử dụng ma trận phép biến đổi PCA thu được trong bước học phân lớp để trích chọn ra k thành phần giữ lại
 - Sử dụng mặt siêu phẳng đã tìm được đưa ra quyết định nhận dạng khuôn mặt

6. Đánh giá ảnh hưởng SVM đến hệ thống nhận dạng khuôn mặt

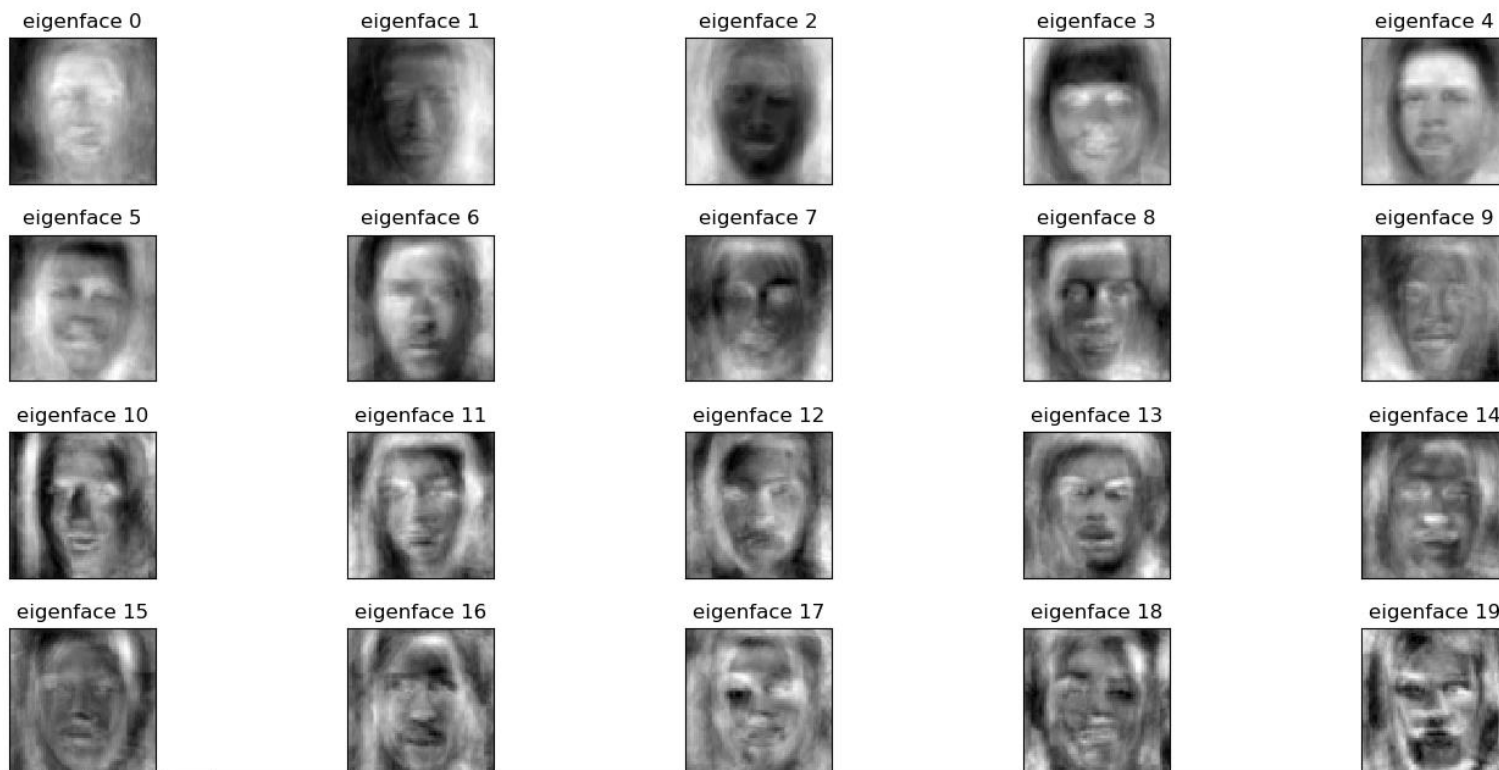
- Đánh giá
- Để thử nghiệm, đánh giá độ chính xác, chia tập dữ liệu thành tập học và tập thử

	precision	recall	f1-score	support
Bale	1.00	0.67	0.80	3
Messi	0.56	1.00	0.71	5
QuynhAnh	1.00	1.00	1.00	11
Ronaldo	1.00	0.80	0.89	10
Soda	1.00	0.50	0.67	2
avg / total	0.93	0.87	0.88	31
[[2 1 0 0 0]				
[0 5 0 0 0]				
[0 0 11 0 0]				
[0 2 0 8 0]				
[0 1 0 0 1]]				

Kết quả phân lớp SVM thu được

➤ Kết quả chi tiết (eigenfaces)

Figure 2

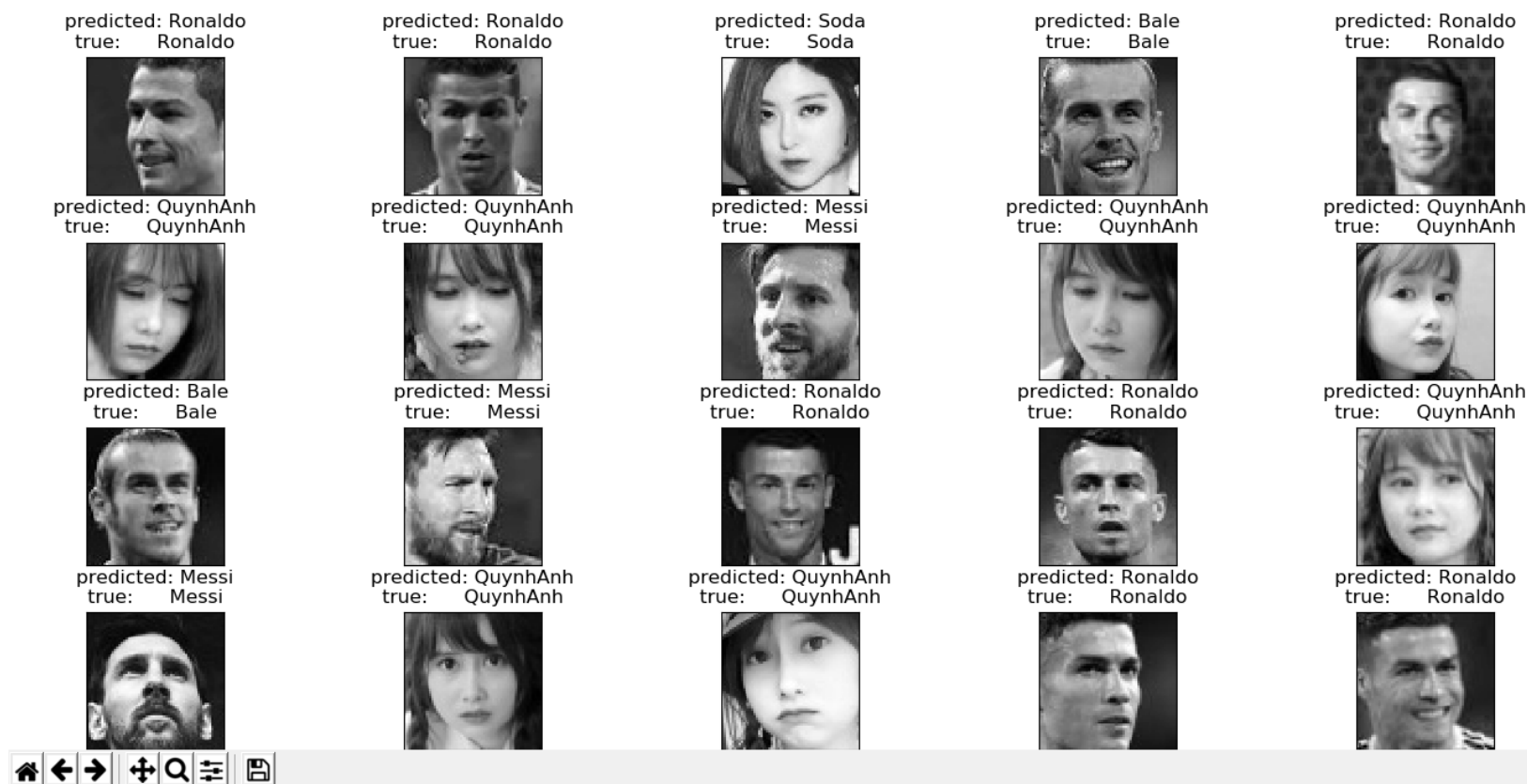


x=5.24183 y=43.3319 [-0.0251]

Kết quả phân lớp SVM thu được

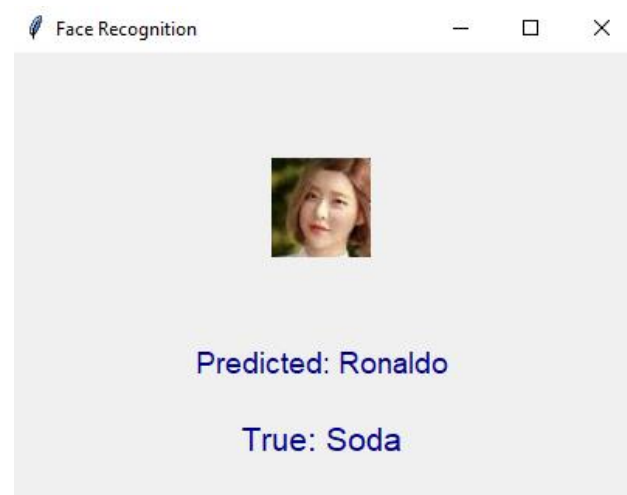
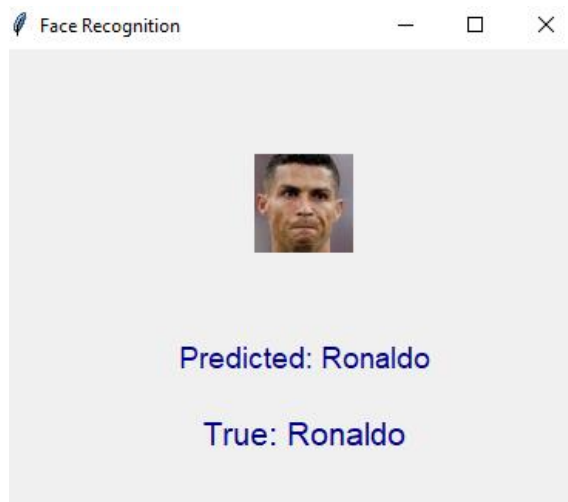
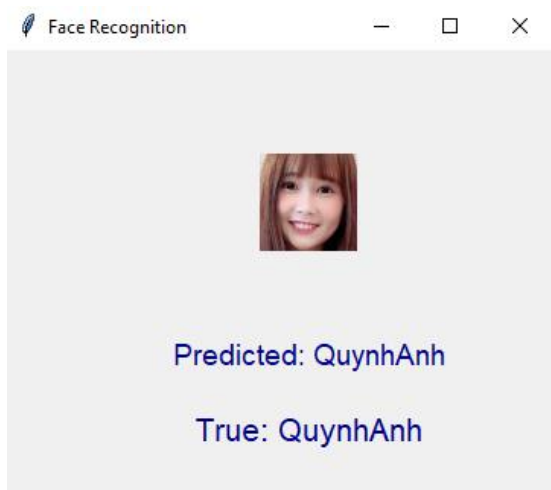
➤ Kết quả chi tiết

Figure 1



Kết quả phân lớp SVM thu được

- Một số kết quả nhận dạng





Thanks for Listening !!!
