

Lab 03

Đặc trưng cục bộ

Thông tin cá nhân

Họ và tên	Vũ Lê Thê Anh	Nguyễn Lê Hồng Hạnh
MSSV	1612838	1612849
Email	{1612838, 1612849}@student.hcmus.edu.vn	
SĐT	0961565087	0902719551

Yêu cầu bài tập

STT	Yêu cầu	Hoàn thành
1	Phát hiện điểm đặc trưng sử dụng thuật toán Harris và hiển thị điểm đặc trưng trên ảnh gốc	100%
2	Phát hiện điểm đặc trưng sử dụng thuật toán blob và hiển thị điểm đặc trưng trên ảnh gốc	100%
3	Phát hiện điểm đặc trưng sử dụng thuật toán DOG và hiển thị điểm đặc trưng trên ảnh gốc	100%
4	Đối sánh 2 ảnh sử dụng đặc trưng SIFT với thuật toán KNN	100%
6	Thực nghiệm đối sánh các phương pháp trên tập dữ liệu ảnh bìa CD/DVD. Đánh giá kết quả và nêu nhận xét về các thuật toán trên.	50%

Mục lục

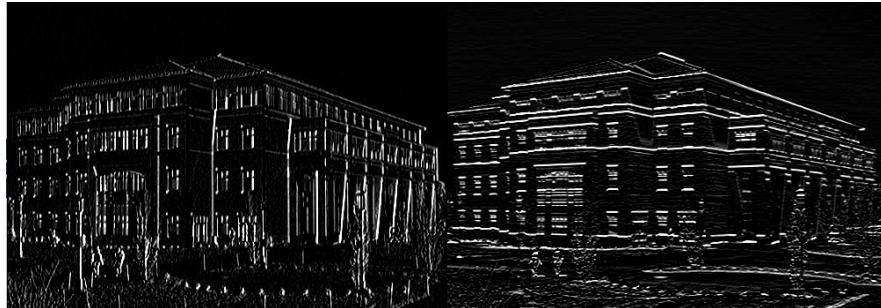
1	Phát hiện đặc trưng	3
1.1	Thuật toán Harris	3
1.2	Thuật toán LoG	4
1.3	Thuật toán DoG	4
2	Biểu diễn đặc trưng	6
3	Đôi sánh đặc trưng	7
4	k-Nearest Neighbor với đặc trưng SIFT	8
5	So sánh các phương pháp	9

1 Phát hiện đặc trưng

1.1 Thuật toán Harris

Harris là thuật toán dùng để xác định góc, gồm các bước sau:

- Tính giá trị gradient G_x, G_y của hình ảnh, ở đây chúng tôi dùng Sobel. (xem Hình 1)

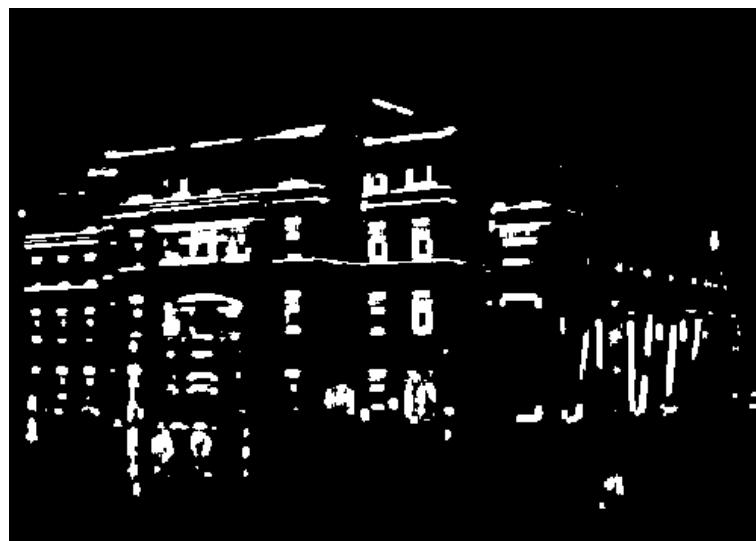


Hình 1: Giá trị gradient G_x (trái) và G_y (phải) của ảnh.

- Tính các giá trị $\sum G_x^2, \sum G_x G_y, \sum G_y^2$ và xây dựng ma trận

$$M = \begin{bmatrix} \sum G_x^2 & \sum G_x G_y \\ \sum G_x G_y & \sum G_y^2 \end{bmatrix}$$

- Tính giá trị $R = \det M - k(\text{trace } M)^2$ với $k = 0.04 - 0.06$.
- Loại bỏ các điểm không đạt ngưỡng. (xem Hình 2)



Hình 2: Giá trị thresh.

- Các điểm còn lại là các điểm đặc trưng cần tìm. (xem Hình 3)



Hình 3: Kết quả tìm corners.

1.2 Thuật toán LoG

Ta có thể tìm blob bằng cách tính giá trị Laplacian của Gaussian (LoG), gồm các bước:

- Tạo bộ lọc LoG với độ lệch chuẩn σ và kích thước $n \times n$ theo công thức:

$$K_\sigma(x, y) = -\frac{1}{\pi\sigma^4} \left(1 - \frac{x^2 + y^2}{2\sigma^2}\right) \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

- Với các giá trị σ thay đổi, xây dựng mô hình kích thước - không gian (scale-space) bằng cách áp bộ lọc LoG có σ tương ứng lên ảnh (xem Hình rồi xếp chồng các ảnh lên nhau).
- Thực hiện tìm các cực trị cục bộ, tức là với mỗi điểm trong mô hình scale-space, ta kiểm tra giá trị tại đó có là cực đại hay cực tiểu của 26 điểm lân cận hay không (9 điểm ở lớp trên, 9 điểm ở lớp dưới, và 8 điểm ở xung quanh trong cùng lớp).
- Sử dụng thuật toán Harris ở trên để kiểm tra xem các điểm cực trị tìm thấy có nằm ở góc hay không, loại bỏ các điểm ở trên biên hoặc vùng phẳng.
- Các điểm còn lại sau quá trình lọc sẽ là các điểm đặc trưng cần tìm.

1.3 Thuật toán DoG

Ta có thể xấp xỉ giá trị LoG bằng cách tính Difference of Gaussian, gồm các bước:

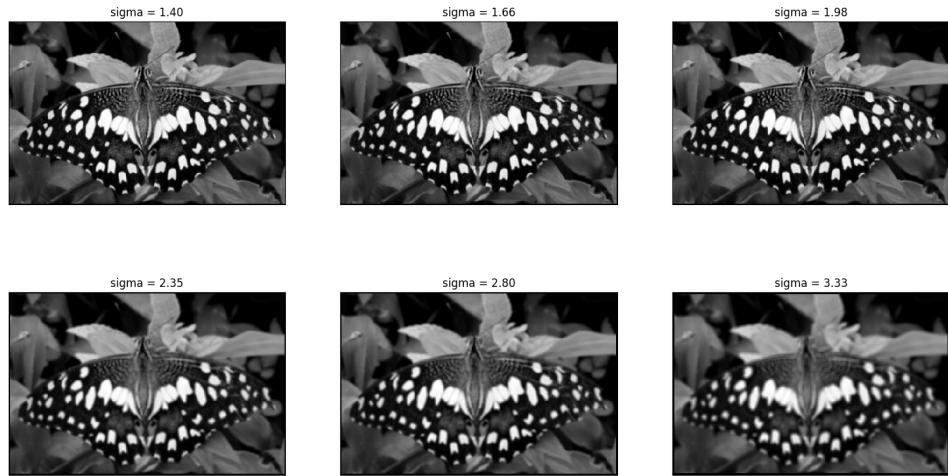
- Tạo bộ lọc Gaussian với độ lệch chuẩn σ và kích thước $n \times n$ theo công thức:

$$G_\sigma(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

- Với các giá trị σ thay đổi, xây dựng mô hình kích thước - không gian (scale-space) bằng

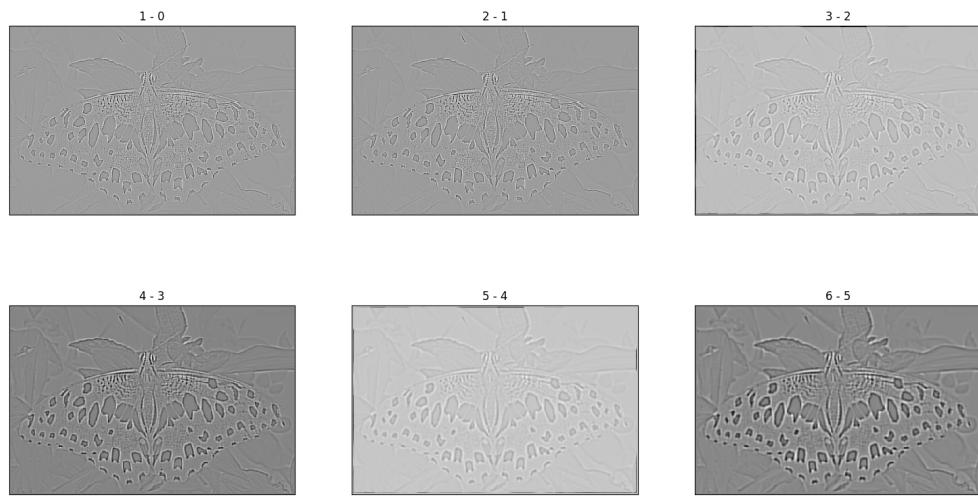
cách:

- Áp bộ lọc Gaussian có σ tương ứng lên ảnh (xem Hình 4) rồi xếp chồng các ảnh lên nhau.



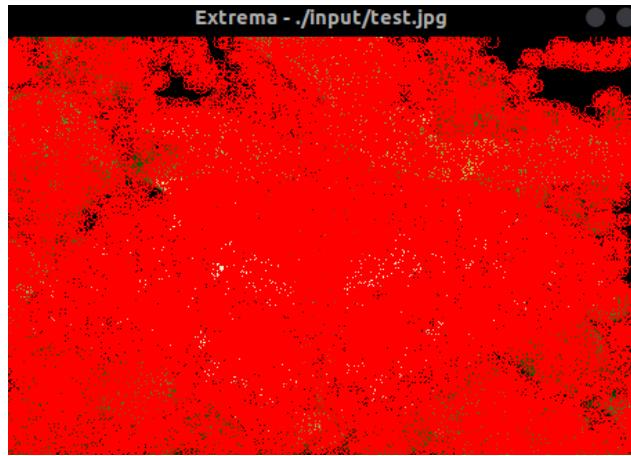
Hình 4: Ảnh qua các bộ lọc Gaussian với σ khác nhau.

- Với hai ảnh ở hai lớp kề cận, lấy hiệu giữa chúng. Các hiệu (xem Hình 5) được xếp chồng sẽ là scale-space cần tìm.

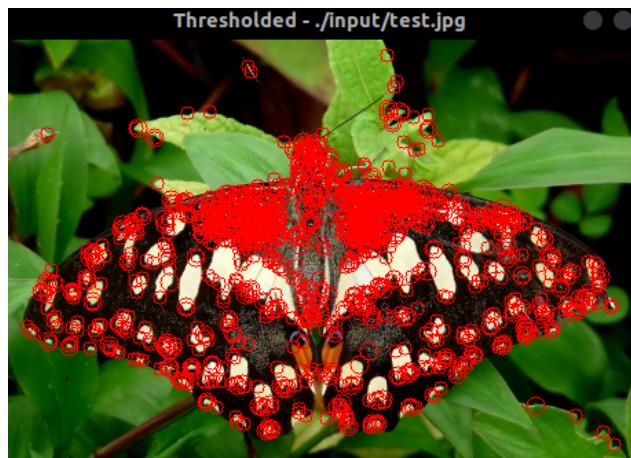


Hình 5: Các lớp DoG của ảnh.

- Thực hiện tìm các cực trị cục bộ, tức là với mỗi điểm trong mô hình scale-space, ta kiểm tra giá trị tại đó có là cực đại hay cực tiểu của 26 điểm lân cận hay không (9 điểm ở lớp trên, 9 điểm ở lớp dưới, và 8 điểm ở xung quanh trong cùng lớp). (xem Hình 6)
- Sử dụng thuật toán Harris ở trên để kiểm tra xem các điểm cực trị tìm thấy có nằm ở góc hay không, loại bỏ các điểm ở trên biên hoặc vùng phẳng.
- Các điểm còn lại sau quá trình lọc sẽ là các điểm đặc trưng cần tìm. (xem Hình 7)



Hình 6: Các điểm cực trị địa phương trên scale-space.



Hình 7: Các điểm đã qua loại bỏ biên cạnh và vùng phẳng.

2 Biểu diễn đặc trưng

Mỗi đặc trưng tìm thấy bằng 1 trong các phương pháp ở trên sẽ bao gồm tọa độ đặc trưng trên ảnh và kích thước đặc trưng. Dựa vào hai thông tin này, ta tìm góc chính của điểm đặc trưng bằng cách:

- Sử dụng giá trị gradient theo phương đứng và phương ngang tính được bằng Sobel, ta tính độ lớn và góc của vector gradient tại mỗi điểm.
- Chia 360 độ thành 36 phần, mỗi phần 10 độ. Ta thiết lập histogram góc trong vùng kích thước của đặc trưng dựa trên góc và độ lớn của các pixel trong vùng. Phần có giá trị lớn nhất chính là phần tương ứng với góc chính. (xem Hình 8)

Sau đó, xét lân cận 16×16 pixel xung quanh điểm đặc trưng với góc quay ứng với góc chính vừa tính được (xem Hình 9) Lân cận này chia làm 16 vùng 4×4 . Với mỗi vùng, ta lại tìm histogram dựa trên góc và độ lớn. Tuy nhiên lần này chúng ta chia 360 độ thành 8 phần, mỗi phần 45 độ. Histogram của 16 vùng, mỗi vùng 8 thành phần sẽ cho ta một vector 128 chiều biểu diễn điểm đặc trưng.



Hình 8: Các đặc trưng và góc chính.



Hình 9: Lân cận 16×16 của đặc trưng.

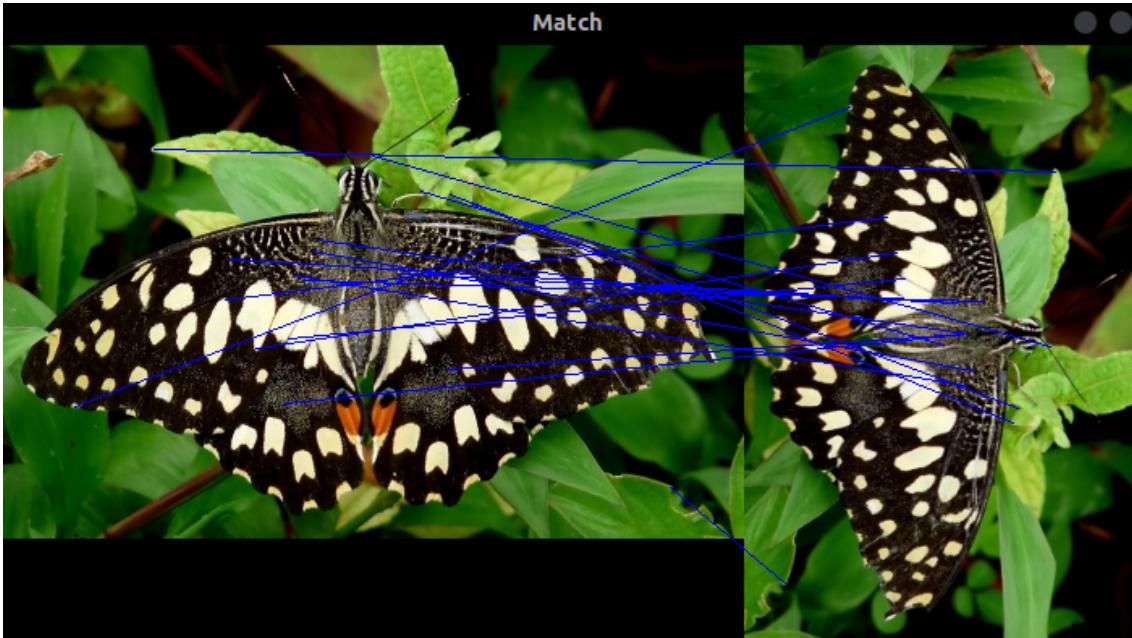
3 Đôi sánh đặc trưng

Mỗi đặc trưng được mô tả bằng một vector 128 chiều, do đó ta có thể so sánh hai đặc trưng bằng cách sử dụng một độ đo khoảng cách thích hợp. Ví dụ sử dụng độ đo khoảng cách Euclid.

$$u, v \in \mathbb{R}^d, D(u, v) = \|u - v\|_2 = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2 + \dots + (u_d - v_d)^2}$$

Như vậy, khi cho hai hình bất kì, ta có thể tìm vật trong hình này ở trong hình khác bằng cách nối các điểm đặc trưng lại với nhau. Hình 10 cho thấy một ví dụ của việc này. Ngoài ra, từ đó cũng có thể nhận thấy đặc trưng SIFT có tính chất bất biến với phép quay và phép phóng to thu nhỏ.

Sau khi tính khoảng cách, ta có thể đặt ngưỡng (ví dụ khoảng cách bé hơn 0.01) để quyết định xem hai điểm đặc trưng có cùng là một điểm hay không.



Hình 10: So khớp đặc trưng trong hai ảnh.

4 k-Nearest Neighbor với đặc trưng SIFT

Như đã trình bày ở trên, việc đối sánh đặc trưng có thể thực hiện bằng việc sử dụng một độ đo khoảng cách giữa hai vector 128 chiều. Vậy còn đối sánh hai hình cùng thuộc một lớp thì sao?

Giả sử ta có một tập huấn luyện gồm hai ảnh A (nhãn là x) và B (nhãn là y), trong ảnh A ta tìm được 2 vector đặc trưng là a_1 và a_2 , tương tự là các vector b_1, b_2 trong ảnh B. Như vậy, tập huấn luyện của ta có dạng $(a_1, x), (a_2, x), (b_1, y), (b_2, y)$.

Khi nhận một ảnh mới là ảnh C, ta muốn biết nhãn của ảnh mới này thì có thể thực hiện như sau:

- Tìm các vector đặc trưng trong ảnh C. Giả sử thu được 3 vector là c_1, c_2 và c_3 .
- Với mỗi vector đặc trưng, ta đối sánh với các vector đã có sẵn trong tập huấn luyện để tìm ra vector gần giống nhất. Giả sử ta tìm ghép được $c_1 = a_1, c_2 = b_1, c_3 = a_2$.
- Nhãn của ảnh mới sẽ chính là nhãn mà có nhiều đặc trưng được so khớp nhất. Tiếp theo giả sử trên, ta thấy có 2 vector đặc trưng được so khớp giữa a và c , có thể nhận định rằng c có cùng nhãn với a .

Một hướng tiếp cận khác là xác định độ giống nhau giữa hai ảnh bằng cách sử dụng một độ đo khoảng cách giữa hai tập hợp, ví dụ khoảng cách Hausdorff:

$$D(X, Y) = \max(\max_{x \in X} \min_{y \in Y} d(x, y), \max_{y \in Y} \min_{x \in X} d(x, y))$$

Trong đó X, Y chính là tập hợp các vector đặc trưng của ảnh 1 và ảnh 2, $d(x, y)$ là khoảng cách giữa hai vector đặc trưng x và y .

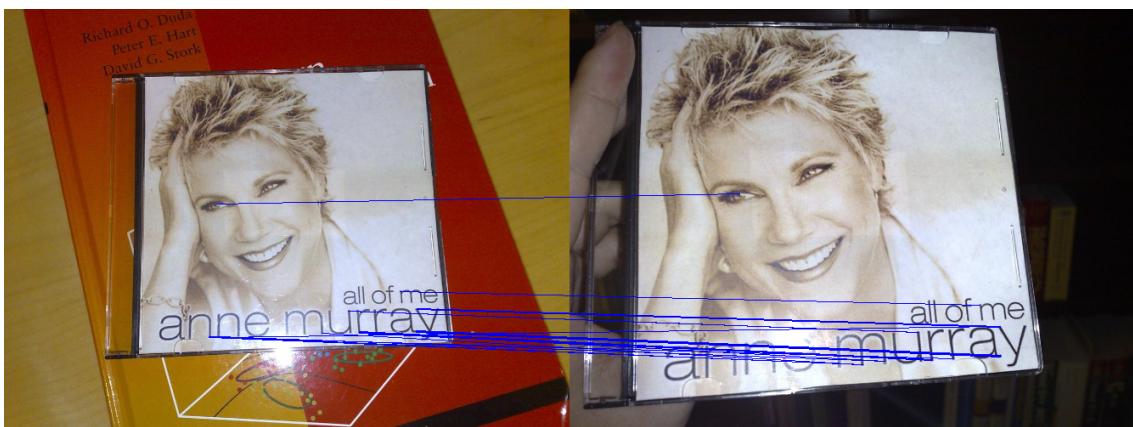
Với việc tính được khoảng cách giữa hai ảnh, ta có thể làm kNN thuần túy: khi có một ảnh mới đi vào

- Tính khoảng cách từ ảnh mới với mọi ảnh trong tập huấn luyện
- Chọn ra k ảnh gần nhất
- Nhãn của ảnh mới chính là nhãn xuất hiện nhiều nhất trong k ảnh được chọn ở trên.

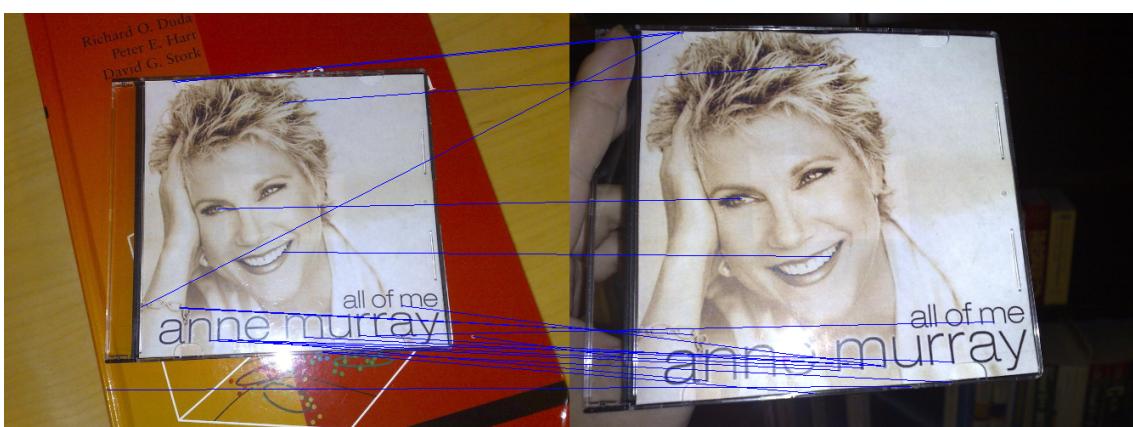
5 So sánh các phương pháp

Khi thử nghiêm, phương pháp Harris tìm ra điểm đặc trưng khá hiệu quả, tuy nhiên lại không có thông tin về kích cỡ của đặc trưng. Do đó, Harris hiệu quả khi cần so khớp hai ảnh mà vật thể có kích thước gần giống nhau.

Phương pháp LoG cho đặc trưng chính xác hơn xấp xỉ của nó là DoG, tuy nhiên thời gian chạy lại chậm hơn do phải lặp lại việc làm mờ Gaussian và áp bộ lọc Laplacian (thay vì chỉ cần làm mờ và lấy hiệu như DoG).



Hình 11: So khớp đặc trưng bằng Harris.



Hình 12: So khớp đặc trưng bằng DOG.

Tài liệu

- [1] SIFT: Theory and Practice (<http://aishack.in/>)
- [2] SIFT - Scale-Invariant Feature Transform (<http://weitz.de/>)