

HƯỚNG DẪN THỰC HÀNH BÀI TẬP 3

Các phép toán lọc ảnh trên miền không gian

Nội dung

I. Phép toán tích chập (Convolution Operator).....	1
II. Lọc thông thấp - UĐ Làm trơn ảnh (Low-pass filtering - Smoothing).....	2
1. Bộ lọc trung bình (Average filtering)	2
2. Bộ lọc trung vị	3
⊙Phân tử trung vị	3
III. Lọc thông cao (High-pass filtering)	4
1. Bộ lọc trung bình	4
2. Bộ lọc dựa trên độ biến đổi giá trị mức xám (Derivative filters).....	5
i. Gradient filter/operator.....	5
ii. Roberts filter/operator.....	6
iii. Prewitt filter/operator	6
iv. Sobel filter/operator	7
IV. Lọc tăng cường (High-boost filtering)	7
V. Tài liệu tham khảo	8

I. Phép toán tích chập (Convolution Operator)

Cho ma trận w có dạng như sau:

$$w = \begin{bmatrix} w1 & w2 & w3 \\ w4 & w5 & w6 \\ w7 & w8 & w9 \end{bmatrix}$$

w thường là ma trận vuông, có kích thước 3×3 , 5×5 , 7×7 , ...

Cho ảnh f có kích thước $width \times height$. Phép tích chập giữa f và w tạo thành ảnh mới g có dạng như sau:

Với m chạy từ 1 đến $height$

Với n chạy từ 1 đến $width$

$$g(m,n) = [w_1f(m-1,n-1) + w_2f(m-1,n) + w_3f(m-1,n+1) \\ + w_4f(m,n-1) + w_5f(m,n) + w_6f(m,n+1) \\ + w_7f(m+1,n-1) + w_8f(m+1,n) + w_9f(m+1,n+1)]$$

Khi đó ta gọi:

- Phép tích chập giữa ảnh f và ma trận w gọi là lọc ảnh, với g là ảnh kết quả.
- h là mặt nạ hay bộ lọc dùng để lọc ảnh.

Đối với các điểm ở đường biên của ảnh thì có các cách giải quyết như sau:

- **Free boudary**: Bỏ đi những phần của w nằm ngoài kích thước ảnh.
- **Fixed boudary**: Thêm cột và dòng tạm vào xung quanh biên ảnh, giá trị của cột và dòng tạm có thể là hằng số hoặc bằng chính giá trị của dòng và cột ở biên ảnh.
- **Periodic boudary**: Tương tự kỹ thuật **Fixed boudary**, trong đó, giá trị của cột tạm ở bên trái bằng giá trị pixel ở biên bên phải ngược lại, giá trị của dòng tạm ở bên trên bằng giá trị pixel ở biên dưới và ngược lại.

II. Lọc thông thấp - ƯD Làm trơn ảnh (Low-pass filtering - Smoothing)

1. Bộ lọc trung bình (Average filtering)

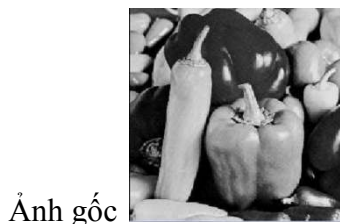
Làm trơn ảnh là làm cho mức độ chênh lệch giá trị màu của các pixel trong ảnh giảm bớt. Phép làm trơn ảnh có 2 ứng dụng:

- **Làm mờ các chi tiết trong ảnh:** vì lý do nghệ thuật hoặc để làm mất đi các chi tiết nhỏ nhất để chỉ giữ lại đối tượng chính trong ảnh.
- **Khử nhiễu:** do thiết bị thu nhận hình ảnh hoặc do điều kiện môi trường tác động làm cho vài pixel trong ảnh có giá trị sai lệch, trường hợp này gọi là nhiễu. Ta có thể điều chỉnh lại giá trị của chúng một cách tương đối.

Để thực hiện phép lọc làm trơn ảnh, ta sử dụng các bộ lọc có dạng như dưới đây, ta gọi chúng là bộ lọc trung bình. Ví dụ một vài bộ lọc trung bình kích thước 3x3:

$$\frac{1}{5} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \frac{1}{32} \begin{bmatrix} 1 & 3 & 1 \\ 3 & 16 & 3 \\ 1 & 3 & 1 \end{bmatrix} \quad \frac{1}{8} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Ví dụ làm mờ chi tiết trong ảnh:



Bộ lọc trung bình NxN

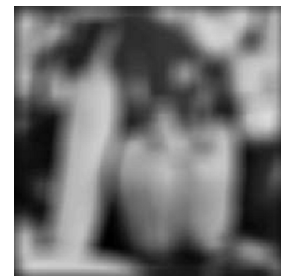
$$\frac{1}{N^2} \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}_{N \times N}$$



N = 3



N = 7



N = 21

Ví dụ khử nhiễu trong ảnh:

Ảnh gốc



Bộ lọc trung bình 5x5

$$\frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Ảnh nhiễu



Ảnh kết quả



Kết luận: Thích hợp với nhiễu có phân phối Gauss, nhưng làm nhòe đi đường biên của các đối tượng trong ảnh.

2. Bộ lọc trung vị

Đặc điểm: Khắc phục khuyết điểm của lọc trung bình là giúp giữ được đường biên của đối tượng. Thích hợp với nhiễu muối tiêu.

Giả sử ta chọn mặt nạ lọc trung vị có kích thước 3x3.

Input: ảnh f

Output: ảnh g

Thuật toán:

Cho i chạy từ 1 đến $height$

Cho j chạy từ 1 đến $width$

- Đặt tâm mặt nạ tại pixel ở vị trí $f(i, j)$ và chọn ra 9 điểm xung quanh (xem hình bên).
- Chọn ra **phần tử trung vị** và thay giá trị của pixel $g(i, j)$ bằng giá trị của phần tử trung vị.

$i-1, j-1$	$i-1, j$	$i-1, j+1$
$i, j-1$	i, j	$i, j+1$
$i+1, j-1$	$i+1, j$	$i+1, j+1$

⊙Phần tử trung vị

Giả sử ta có 9 giá trị: 15, 6, 6, 0, 1, 2, 10, 12, 4

- Sắp tăng dần: 0, 1, 2, 4, 6, 6, 10, 12, 15
- Chọn phần tử ở vị trí giữa, tức là ở vị trí $(9+1)/2 = 5$, tức là số 6. 6 chính là **phần tử trung vị** của dãy số này.

Tổng quát: có n giá trị được sắp tăng dần

- Nếu n là số chẵn, phần tử trung vị nằm ở vị trí $n/2$.
- Nếu n là số lẻ, phần tử trung vị nằm ở vị trí $(n+1)/2$.

Ví dụ:

Ảnh nhiễu muối tiêu



Ảnh được lọc trung bình



Ảnh được lọc trung vị



Ảnh nhiễu Gauss



Ảnh được lọc trung bình



Ảnh được lọc trung vị

III. Lọc thông cao (High-pass filtering)

Trái ngược với bộ lọc low-pass, có đặc điểm làm mờ chi tiết trong ảnh, lọc thông cao sẽ loại bỏ những vùng pixel nào có giá trị tương đối giống nhau và chỉ giữ lại những pixel mà giá trị màu xung quanh nó chênh lệch nhiều. Thường dùng để dò tìm đường biên của đối tượng trong ảnh.

1. Bộ lọc trung bình

Đặc điểm của các bộ lọc thuộc dạng này là tâm có giá trị dương, phần còn lại có giá trị âm hoặc bằng không. Ví dụ **hình a**:

$$\frac{1}{9} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

(a) High-pass filter

(b) Low-pass filter tương ứng

Bộ lọc high-pass ở **hình (a)** tương ứng với bộ lọc low-pass ở **hình (b)** như sau:

$$\frac{1}{9} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Tổng quát:

Bộ lọc high-pass trung bình = Ảnh f - Bộ lọc low-pass trung bình

Ví dụ:



Ảnh ban đầu



Ảnh sau khi lọc high-pass

2. Bộ lọc dựa trên độ biến đổi giá trị mức xám (Derivative filters)

Bộ lọc trung bình high-pass cũng mắc phải khuyết điểm giống như lọc trung bình low-pass là làm nhòe đi một phần chi tiết ảnh vì nó lấy trung bình cộng của các điểm xung quanh.

Một dạng bộ lọc high-pass khác giúp cho đường biên của đối tượng được xác định rõ ràng hơn, đó là các bộ lọc thuộc nhóm derivative filters.

i. Gradient filter/operator

Dựa theo khái niệm trong toán học, gradient của hàm 2 biến $f(x,y)$ có dạng như hình (a) và độ lớn của gradient được tính như hình (b)

$$\nabla f(x,y) = \begin{bmatrix} \frac{\partial f(x,y)}{\partial x} \\ \frac{\partial f(x,y)}{\partial y} \end{bmatrix} \quad |\nabla f(x,y)| = \left[\left(\frac{\partial f(x,y)}{\partial x} \right)^2 + \left(\frac{\partial f(x,y)}{\partial y} \right)^2 \right]^{1/2}$$

(a)
(b)

Có ý nghĩa là để đo mức độ biến đổi của hàm f theo 2 phương của trục x và y .

Tương tự, ta xem ảnh trong không gian 2 chiều như một hàm f theo 2 biến x, y . Với x, y tượng trưng cho chiều dọc và chiều ngang.

Dưới đây là một vài bộ lọc tính theo phương pháp gradient.

ii. Roberts filter/operator

Bộ lọc Roberts sử dụng 2 mặt nạ sau:

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$$

hoặc

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

Input: ảnh f

Output: ảnh g được lọc high-pass bằng bộ lọc Roberts

Thuật toán:

Cho x chạy từ 1 đến *height*

Cho y chạy từ 1 đến *width*

☞ Tại mỗi pixel ở vị trí (x, y) trên ảnh f , tính độ lớn gradient tại điểm đó:

$$|\nabla f(x, y)| = [(1 * f(x, y) - 1 * f(x + 1, y))^2 + (1 * f(x, y) - 1 * f(x, y + 1))^2]^{1/2}$$

hoặc

$$|\nabla f(x, y)| = [(1 * f(x, y) - 1 * f(x + 1, y + 1))^2 + (1 * f(x, y) - 1 * f(x + 1, y - 1))^2]^{1/2}$$

☞ Gán $g(x, y) = |\nabla f(x, y)|$

iii. Prewitt filter/operator

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

$$|\nabla f(x,y)| = [(f(x-1,y-1) + f(x-1,y) + f(x-1,y+1) - f(x+1,y-1) - f(x+1,y) - f(x+1,y+1))^2 + (f(x-1,y-1) + f(x,y-1) + f(x+1,y-1) - f(x-1,y+1) - f(x,y+1) - f(x+1,y+1))^2]^{1/2}$$

iv. Sobel filter/operator

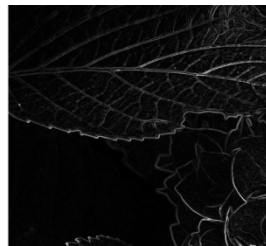
$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

$$|\nabla f(x,y)| = [(f(x-1,y-1) + 2 * f(x-1,y) + f(x-1,y+1) - f(x+1,y-1) - 2 * f(x+1,y) - f(x+1,y+1))^2 + (f(x-1,y-1) + 2 * f(x-1,y) + f(x-1,y+1) - f(x-1,y+1) - 2 * f(x,y+1) - f(x+1,y+1))^2]^{1/2}$$

Ví dụ:¹



Ảnh gốc



Roberts



Prewitt



Sobel

IV. Lọc tăng cường (High-boost filtering)

Tương tự lọc high-pass, chỉ có điều nhân thêm hệ số vào ảnh f như sau:

$$\text{Bộ lọc tăng cường} = \text{hệ số} * \text{Ảnh } f - \text{Bộ lọc low-pass trung bình}$$

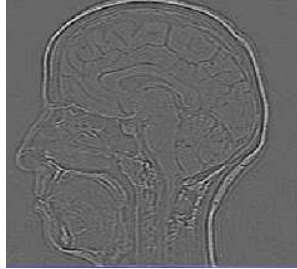
Ví dụ:

¹ Các ảnh kết quả được tăng cường bằng bộ lọc high-boost 3x3 để các đường biên được lộ rõ. Tham khảo trong [3].

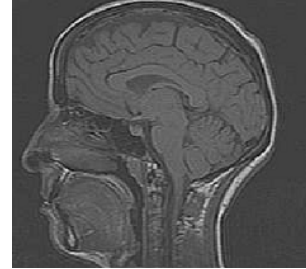
$$\frac{1}{9} \begin{bmatrix} -1 & -1 & -1 \\ -1 & 26 & -1 \\ -1 & -1 & -1 \end{bmatrix} = 3 * \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$



Ảnh gốc



Ảnh lọc high-pass



Ảnh lọc high-boost

V. Tài liệu tham khảo

- [1] Dương Anh Đức, Image Enhancement – Spatial Filtering, Khoa CNTT, ĐH Khoa Học Tự Nhiên, Tp HCM, 3/2007.
- [2] Lý Quốc Ngọc, *Bài giảng Xử lý ảnh số và video số*, Khoa CNTT, ĐH Khoa Học Tự Nhiên, Tp HCM, ?.
- [3] Nguyễn Quốc Bình, *Đồ án cao học môn Xử lý ảnh*, Khoa CNTT, ĐH Khoa Học Tự Nhiên, Tp HCM, 2010.