

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



Đề tài 3a: Image filters

Kỹ thuật lọc ảnh và ứng dụng trong lọc nhiễu làm trơn.

BÀI TẬP MÔN HỌC

IT4090 – Xử lý ảnh – 2016-2017

Giảng viên hướng dẫn : PGS.TS Nguyễn Thị Hoàng Lan

Sinh viên thực hiện : Trần Thế Anh – 20130229
: Nguyễn Văn Thiệu – 20133750

Lớp : KSTN-CNTT-K58

HÀ NỘI 2016

Mục lục

IT4090 – Xử lý ảnh – 2016-2017	1
1. Lý thuyết	3
2. Các loại bộ lọc	6
2.1 Lọc tuyến tính.	6
2.2. Lọc phi tuyến	8
3. Vấn đề ứng dụng: Lọc nhiễu làm trơn ảnh bằng 2 bộ lọc khác nhau.....	9
3.1 Bộ lọc 1: Áp dụng bộ lọc trung bình trong lọc nhiễu làm trơn ảnh.	9
3.2 Bộ lọc 2: Áp dụng bộ lọc trung vị trong lọc nhiễu làm trơn ảnh.	10
3.3 Kết quả thực nghiệm	12
3.4. Phân tích, so sánh 2 phương pháp.....	17
• Phân tích:	17
• So sánh 2 bộ lọc:.....	17
4. Tài liệu tham khảo	18

1. Lý thuyết

- Một bộ lọc ảnh không gian bao gồm

(1) 1 mặt nạ (thường là 1 hình chữ nhật nhỏ)

(2) 1 phép toán được định nghĩa trước thực hiện trên các điểm ảnh được bao phủ bởi mặt nạ.

- Quá trình lọc ảnh gồm các bước sau:

(1) Xác định điểm trung tâm của mặt nạ;

(2) Tại điểm (x, y) đang xét (trùng với tâm mặt nạ), thực hiện phép toán lọc trên các điểm lân cận (bị mặt nạ bao phủ);

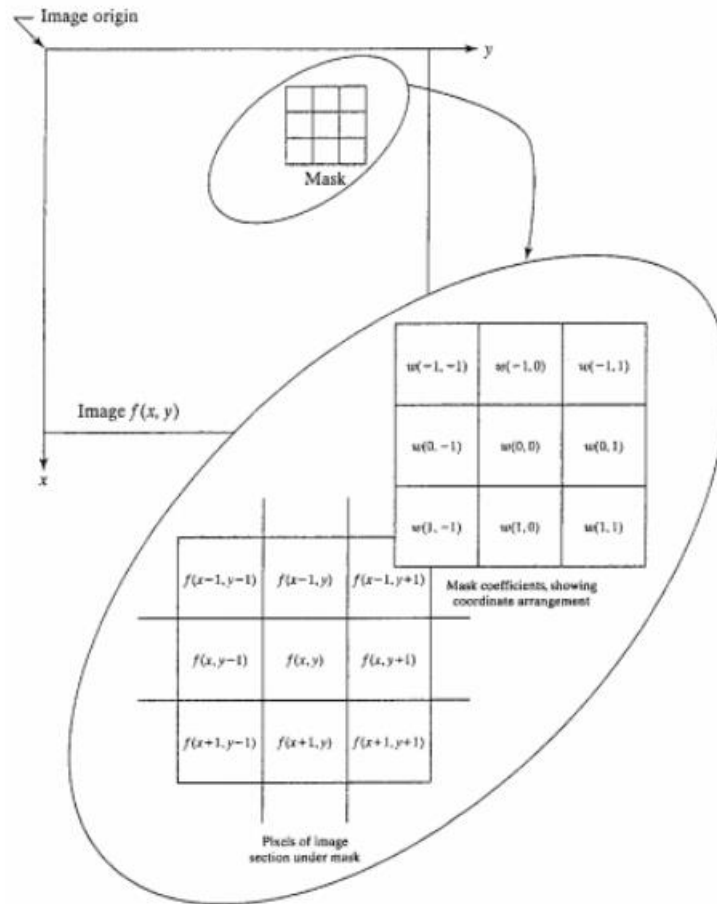
(3) Ghi nhận kết quả phép lọc là giá trị mức xám của điểm ảnh (x, y) trong ảnh đầu ra.

(4) Lần lượt trượt mặt nạ tới những điểm chưa xét. Lặp lại bước (2). Quá trình lọc kết thúc khi điểm trung tâm của mặt nạ lần lượt thăm hết từng điểm ảnh của ảnh đầu vào, ta thu được kết quả ảnh đã được lọc ở đầu ra.

- Dựa vào phép toán áp dụng trong bộ lọc mà ta có 2 loại bộ lọc:

+ Bộ lọc tuyến tính: khi phép toán thực hiện trên các điểm ảnh là tuyến tính

+ Bộ lọc phi tuyến: khi phép toán thực hiện trên các điểm ảnh là phi tuyến



Hình vẽ biểu diễn hoạt động của một bộ lọc không gian tuyến tính sử dụng mặt nạ kích thước 3*3.

- Tại bất kì điểm (x,y) nào của ảnh, đáp ứng $g(x,y)$ của bộ lọc là tổng các tích của hệ số lọc và giá trị các điểm ảnh bao phủ bởi mặt nạ:

$$g(x, y) = w(-1, -1)f(x - 1, y - 1) + w(-1, 0)f(x - 1, y) + \dots + w(0, 0)f(x, y) + \dots + w(1, 1)f(x + 1, y + 1)$$

- Hệ số trung tâm của mặt nạ $w(0,0)$ tương ứng với điểm ảnh ở vị trí (x,y) đang xét.

Mặt nạ kích thước $m*n$, ta thường chọn m,n là số lẻ.

Giả thiết rằng $m=2a+1$, $n=2b+1$, với $a, b \geq 1$.

Kích thước mặt nạ nhỏ nhất có ý nghĩa là 3*3.

- Khi trượt mặt nạ trên ảnh đầu vào, một vấn đề ta cũng cần quan tâm là tiến hành lọc tại các điểm nằm gần biên ảnh. Với một bộ lọc vuông $n \times n$, tại vị trí cách biên một khoảng cách $\frac{n-1}{2}$ bộ lọc sẽ có biên trùng khít với biên ảnh, nhưng đối với các điểm ảnh nằm gần biên thì một hoặc một số hàng hoặc cột của ma trận lọc sẽ nằm bên ngoài ảnh.

Có nhiều giải pháp để giải quyết vấn đề này. Một giải pháp đơn giản là ta chỉ tiến hành xử lý tại các điểm có khoảng cách không nhỏ hơn $\frac{n-1}{2}$ so với biên ảnh. Kết quả là ảnh sau lọc có kích thước nhỏ hơn so với ảnh gốc nhưng toàn bộ điểm ảnh đều được xử lý.

Trong trường hợp cần ảnh sau xử lý có cùng kích thước với ảnh gốc, ta có thể mở rộng ảnh gốc bằng cách:

- + Thêm giá trị 0 vào các điểm ảnh bên ngoài,
- + Thêm các điểm ảnh bên ngoài lặp lại giá trị của điểm ảnh biên,
- + Thêm các mức xám bên ngoài đối xứng gương với các điểm ảnh bên trong qua biên.

- Có 2 khái niệm cần nắm rõ trước khi đi sâu vào lọc ảnh tuyến tính, đó là: Tương quan và Tích chập. Đối với tương quan, mặt nạ lọc trượt trên ảnh và tính tổng của các điểm bị mặt nạ che phủ ở từng vị trí. Đối với tích chập, cơ chế tương tự nhưng chỉ khác là mặt nạ lọc được quay 180° trước khi trượt. Ta sẽ giải thích rõ hơn dưới đây.

Đối với một bộ lọc kích thước $m \times n$, chúng ta mở rộng hình ảnh với tối thiểu là $m-1$ dãy số 0 ở phía trên và phía dưới; và $n-1$ cột của các giá trị 0 ở bên trái và bên phải. Trong trường hợp này, m và n là bằng 3, vì vậy chúng ta mở rộng f với hai hàng 0 trên và dưới và hai cột 0 bên trái và bên phải, như hình (b) cho thấy. Hình (c) cho thấy vị trí ban đầu của mặt nạ lọc để thực hiện tương quan, và hình (d) cho thấy kết quả tương quan đầy đủ. Hình (e) cho thấy kết quả cắt tương ứng. Lưu ý là kết quả thu được là các giá trị mặt nạ được quay bằng 180° . Đối với tích chập, chúng ta xoay trước mặt nạ 180° và tiến hành các bước như trên. Hình (f), (g), (h) cho ta thấy kết quả. Ta thấy tích chập của một hàm với một xung là bản sao của hàm tại địa điểm của xung. Nếu mặt nạ lọc là đối xứng, tương quan và tích chập mang lại kết quả tương tự.

(a) Origin $f(x,y)$ and $w(x,y)$

0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	0	0	0	0
0	0	0	0	0

$w(x,y)$

1	2	3
4	5	6
7	8	9

(b) Padded f

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	0	1	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	0

(c) Initial position for w

1	2	3	0	0	0	0	0	0
4	5	6	0	0	0	0	0	0
7	8	9	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	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	0

(d) Full correlation result

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	9	8	7	0	0	0
0	0	0	6	5	4	0	0	0
0	0	0	3	2	1	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

(e) Cropped correlation result

0	0	0	0	0
0	9	8	7	0
0	6	5	4	0
0	3	2	1	0
0	0	0	0	0

(f) Rotated w

9	8	7	0	0	0	0	0	0
6	5	4	0	0	0	0	0	0
3	2	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	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	0

(g) Full convolution result

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	1	2	3	0	0	0
0	0	0	4	5	6	0	0	0
0	0	0	7	8	9	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

(h) Cropped convolution result

0	0	0	0	0
0	1	2	3	0
0	4	5	6	0
0	7	8	9	0
0	0	0	0	0

- Bởi vì nhiều ngẫu nhiên thông thường bao gồm quá trình chuyển đổi mạnh mức cường độ, ứng dụng rõ ràng nhất của làm mịn là giảm nhiễu. Tuy nhiên, các cạnh (mà hầu như luôn luôn là các tính năng mong muốn của một hình ảnh) cũng được đặc trưng bởi quá trình chuyển đổi cường độ sắc nét, bộ lọc trung bình có tác dụng phụ không mong muốn rằng chúng làm giảm cạnh.

- Một ứng dụng khác của loại này của quá trình bao gồm việc làm mịn các đường nét sai do sử dụng một số lượng không đủ mức cường độ. Một sử dụng chính của các bộ lọc trung bình là trong việc giảm chi tiết "không thích hợp" trong một hình ảnh.

- Hình dưới đây là 2 mặt nạ lọc tuyến tính kích thước sử dụng trong làm mịn ảnh.

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \quad \frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

- Ở bộ lọc thứ nhất, giá trị mức xám đầu ra là giá trị trung bình chuẩn của các giá trị ở các điểm ảnh bị bao phủ bởi mặt nạ. Các hệ số của bộ lọc là 1 thay vì là 1/9 là để cho việc tính toán được hiệu quả hơn. Sau khi quá trình lọc kết thúc, tất cả các giá trị điểm ảnh sẽ được chia cho 9. Bộ lọc không gian mà các hệ số trong mặt nạ đều bằng nhau được gọi là bộ lọc trung bình (Mean Filter).

- Ở bộ lọc thứ hai, các giá trị điểm ảnh được nhân với các hệ số khác nhau, thể hiện sự quan trọng (trọng lượng) so với các điểm khác. Điểm ảnh ở giữa mặt nạ được nhân với giá trị lớn hơn, thể hiện sự quan trọng lớn hơn trong tính toán giá trị trung bình. Các hệ số khác được coi như là khoảng cách từ điểm đó tới trung tâm mặt nạ, các láng giềng trực giao xa hơn nên có trọng số nhỏ hơn so với các láng giềng trực tiếp. Chiến lược đặt hệ số như thế cho mặt nạ nhằm nỗ lực giảm mờ trong quá trình làm mịn ảnh.

- Biểu thức thực thi tổng quát để lọc tuyến tính một ảnh kích thước $M \times N$ bằng bộ lọc trung bình trọng số w kích thước $m \times n$ (m, n lẻ) :

$$g(x, y) = \frac{\sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t)}{\sum_{s=-a}^a \sum_{t=-b}^b w(s, t)}$$

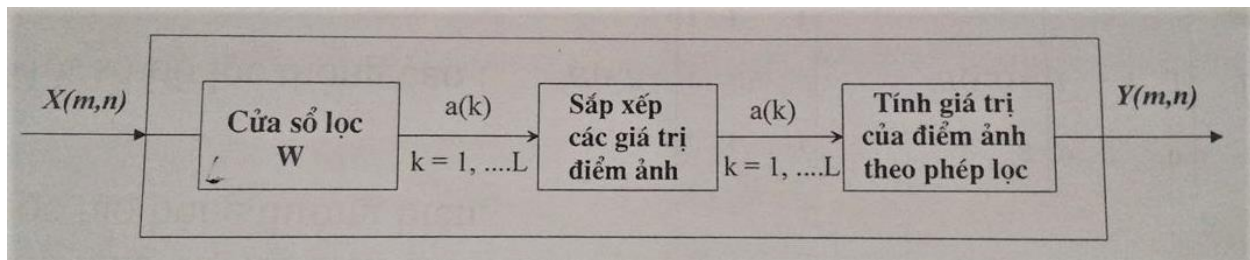
- Trong đó, $m = 2a+1$; $n = 2b+1$;

x, y thay đổi, biểu diễn mọi điểm ảnh của ảnh gốc: $x = 0, 1, 2, \dots, M - 1$

và $y = 0, 1, 2, \dots, N - 1$

2.2. Lọc phi tuyến

- Đối với bộ lọc phi tuyến, đầu ra là kết quả của phép toán phi tuyến tiến hành trên các lân cận bị mặt nạ bao phủ. Các bộ lọc hạng là bộ lọc tuyến tính phổ biến.



- Đối với bộ lọc hạng, các giá trị mức xám của các điểm ảnh lân cận bị mặt nạ bao phủ được sắp xếp thứ tự (xếp hạng), và đáp ứng tại điểm ảnh đó được xác định bởi kết quả xếp hạng.

- Các bộ lọc hạng cụ thể:

+ Lọc trung vị (Median Filter): đáp ứng tại từng điểm ảnh là giá trị chính giữa sau quá trình xếp hạng các giá trị của điểm ảnh lân cận. Tác dụng: hiệu quả trong giảm nhiễu xung (VD: nhiễu salt and pepper)

+ Lọc Max (lọc dẫn): đáp ứng tại từng điểm ảnh là giá trị lớn nhất sau quá trình xếp hạng các giá trị của điểm ảnh lân cận. Tác dụng: tìm các điểm sáng nhất của ảnh.

+ Lọc Min (lọc co): đáp ứng tại từng điểm ảnh là giá trị nhỏ nhất sau quá trình xếp hạng các giá trị của điểm ảnh lân cận. Tác dụng: tìm các điểm tối nhất của ảnh.

3. Vấn đề ứng dụng: Lọc nhiễu làm trơn ảnh bằng 2 bộ lọc khác nhau

- Trong phần này, chúng ta sẽ tìm hiểu một số phương pháp lọc nhiễu làm trơn ảnh.

Đầu vào: Ảnh bị nhiễu

Đầu ra: Ảnh được lọc nhiễu làm trơn.

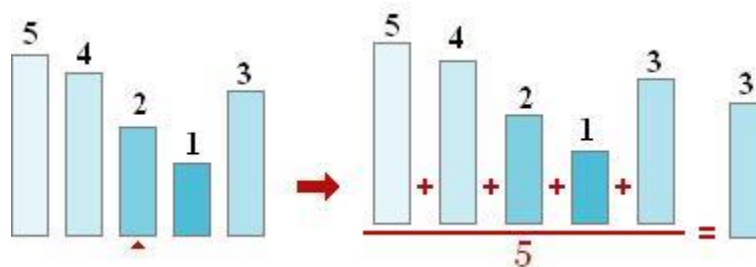
- Sử dụng 2 bộ lọc trung bình và trung vị có thể lọc nhiễu làm trơn ảnh nhưng tùy thuộc vào ảnh bị nhiễu theo kiểu nào mà kết quả đầu ra sẽ khác nhau.

3.1 Bộ lọc 1: Áp dụng bộ lọc trung bình trong lọc nhiễu làm trơn ảnh.

a. Khái quát:

- Lọc trung bình là kỹ thuật lọc tuyến tính, hoạt động như một bộ lọc thông thấp. Ý tưởng chính của thuật toán lọc Trung vị như sau: ta sử dụng một cửa sổ lọc (ma trận 3x3) quét qua lần lượt từng điểm ảnh của ảnh đầu vào input. Tại vị trí mỗi điểm ảnh lấy giá trị của các điểm ảnh tương ứng trong vùng 3x3 của ảnh gốc "lấp" vào ma trận lọc. Giá trị điểm ảnh của ảnh đầu ra là giá trị trung bình của tất cả các điểm ảnh trong cửa sổ lọc. Việc tính toán này khá đơn giản với hai bước gồm tính tổng các thành phần trong cửa sổ lọc và sau đó chia tổng này cho số các phần tử của cửa sổ lọc.

Sẽ dễ hình dung hơn bằng mô tả trong hình dưới đây:



unfiltered values

5	3	6
2	1	9
8	4	7

$$5 + 3 + 6 + 2 + 1 + 9 + 8 + 4 + 7 = 45$$

$$45 / 9 = 5$$

mean filtered

*	*	*
*	5	*
*	*	*

b. Thuật toán:

Sơ lược một cách ngắn gọn các bước của giải thuật:

1. Quét cửa sổ lọc lần lượt lên các thành phần của ảnh đầu vào; điền các giá trị được quét vào cửa sổ lọc.
2. Xử lý bằng cách thao tác trên các thành phần của cửa sổ lọc.
3. Tính giá trị trung bình các thành phần trong cửa sổ lọc.
4. Gán giá trị trung bình này cho giá trị điểm ảnh ở đầu ra.

3.2 Bộ lọc 2: Áp dụng bộ lọc trung vị trong lọc nhiễu làm trơn ảnh.

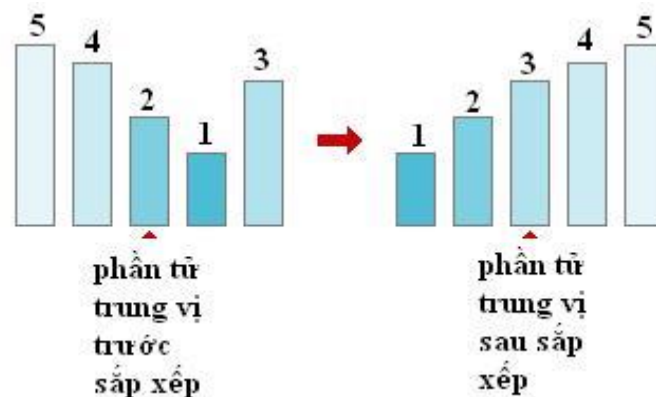
a. Khái quát:

- Lọc Trung vị là một kỹ thuật lọc phi tuyến (non-linear), nó khá hiệu quả đối với hai loại nhiễu: nhiễu đốm (speckle noise) và nhiễu muối tiêu (salt-pepper noise). Kỹ thuật này là một bước rất phổ biến trong xử lý ảnh.

- Ý tưởng chính của thuật toán lọc Trung vị như sau: ta sử dụng một cửa sổ lọc (ma trận 3x3) quét qua lần lượt từng điểm ảnh của ảnh đầu vào input. Tại vị trí mỗi điểm ảnh lấy giá trị của các điểm ảnh tương ứng trong vùng 3x3 của ảnh gốc "lấp" vào ma trận lọc. Sau đó sắp xếp các điểm ảnh trong cửa sổ này theo thứ tự (tăng dần hoặc giảm dần tùy ý). Cuối cùng, gán điểm ảnh nằm chính giữa (Trung vị) của

dãy giá trị điểm ảnh đã được sắp xếp ở trên cho giá trị điểm ảnh đang xét của ảnh đầu ra output.

Sẽ dễ hình dung hơn bằng mô tả trong hình dưới đây:



unfiltered values

6	2	0
3	97	4
19	3	10

in order:

0, 2, 3, 3, 4, 6, 10, 15, 97

median filtered

*	*	*
*	4	*
*	*	*

b. Thuật toán

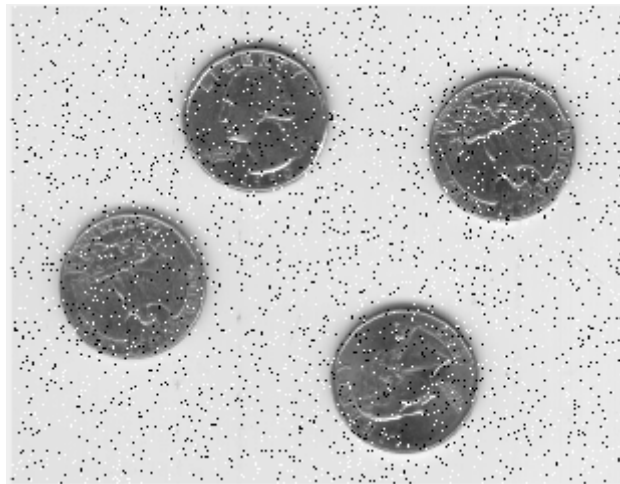
Sơ lược một cách ngắn gọn các bước của giải thuật 4 bước chính:

1. Quét cửa sổ lọc lên các thành phần của ảnh gốc; điền các giá trị được quét vào cửa sổ lọc.
2. Lấy các thành phần trong cửa sổ lọc để xử lý.
3. Sắp xếp theo thứ tự các thành phần trong cửa sổ lọc.
4. Lưu lại thành phần trung vị, gán cho giá trị điểm ảnh đầu ra.

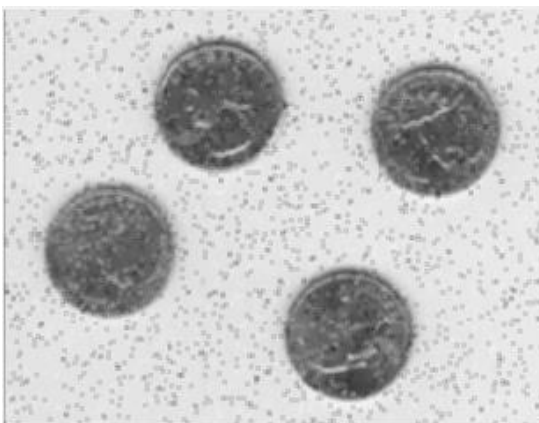
3.3 Kết quả thực nghiệm

- Môi trường thử nghiệm: 2 chương trình thực hiện 2 giải thuật lọc nhiễu làm trơn ảnh được viết và chạy trên matlab.
- Dữ liệu thử nghiệm: Mục đích chính để lọc nhiễu làm trơn ảnh nên dữ liệu ảnh vào là ảnh bị nhiễu. Do đó nhóm đã viết một chương trình khác dùng để tạo ảnh nhiễu, lưu lại. Rồi lấy ảnh nhiễu đó làm đầu vào của chương trình lọc làm trơn ảnh sử dụng 2 bộ lọc.
- Cách thức đánh giá: Do ở đây chỉ cài đặt thực hiện lọc dùng 2 bộ lọc trung bình và trung vị nên để đánh giá kết quả nhận được sẽ dựa vào trực quan nhìn thấy. Sẽ có ảnh đầu vào, và đầu ra là 2 ảnh sử dụng 2 bộ lọc. Dựa vào trực quan ta sẽ thấy được ảnh có chấp nhận được hay không (ảnh qua bộ lọc có tốt hơn hay không). Ví dụ:

Ảnh bị nhiễu được lấy làm đầu vào



Ảnh không chấp nhận được sau khi lọc

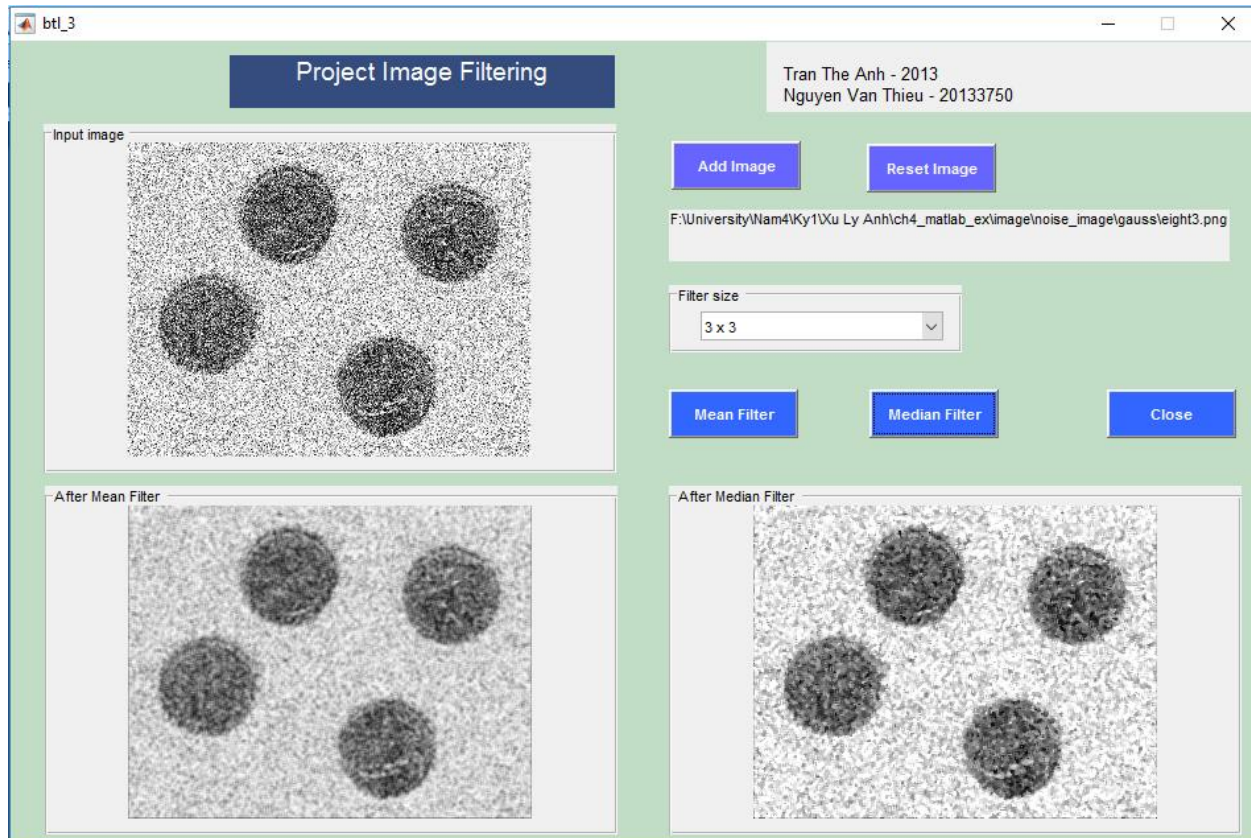


Ảnh chấp nhận (cải thiện tốt) sau khi lọc

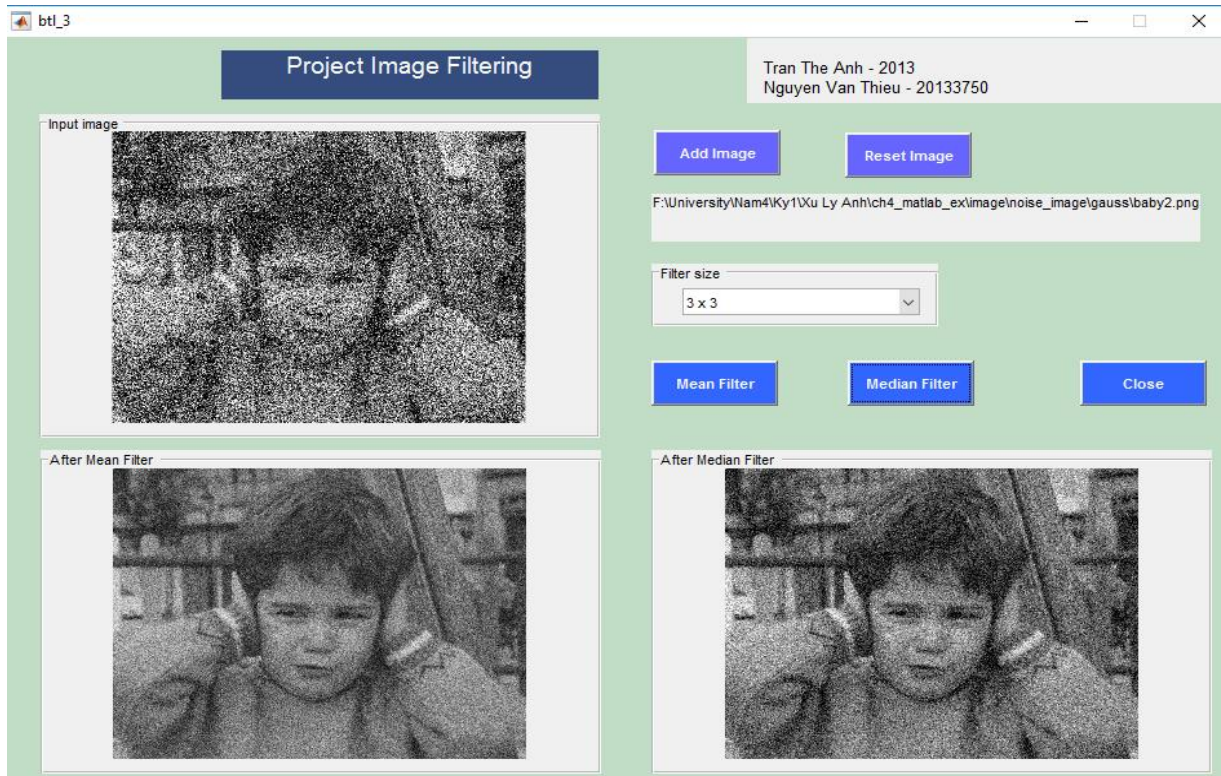


- Kết quả đầu ra:

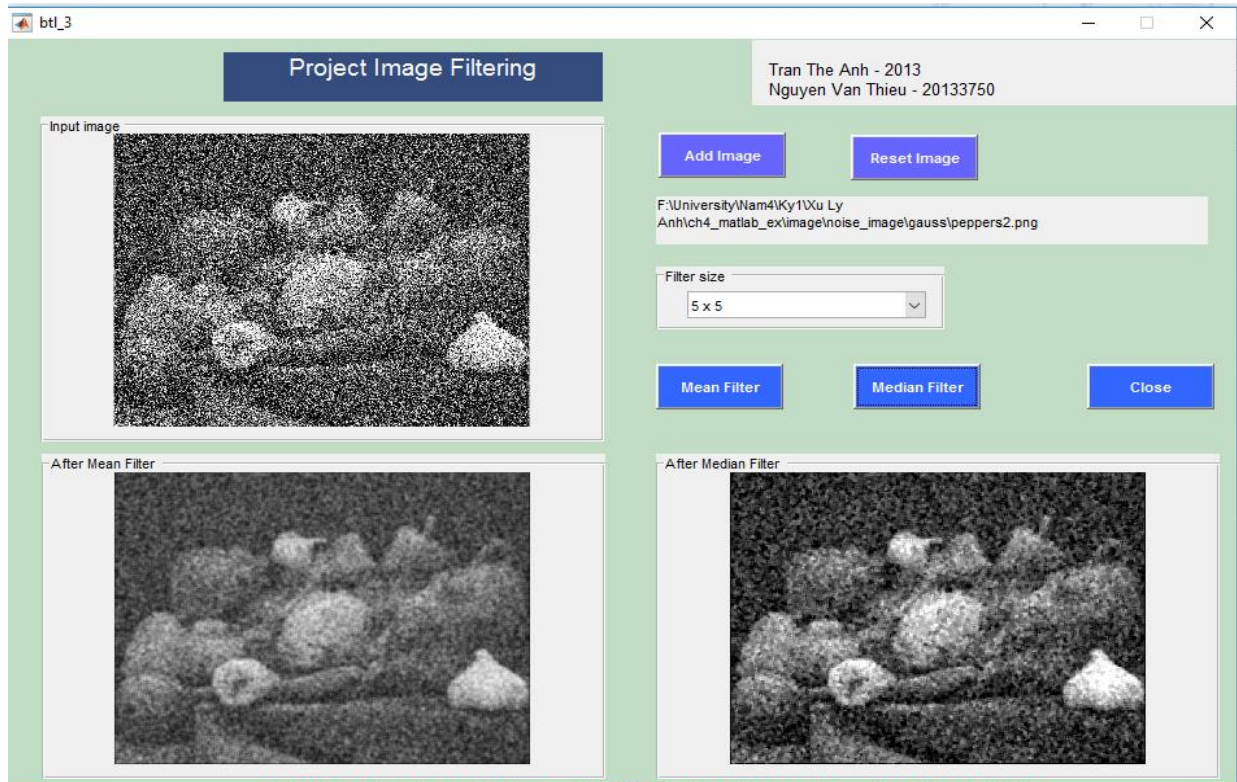
- Bên dưới là 1 số ảnh bị nhiễu Gauss được lấy làm đầu vào, và đầu ra là ảnh sau khi lọc bởi bộ lọc trung bình (Mean Filter) và trung bị (Median Filter)



Hình ảnh 4 đồng tiền xu bị nhiễu Gauss sau khi được lọc bởi 2 bộ lọc.

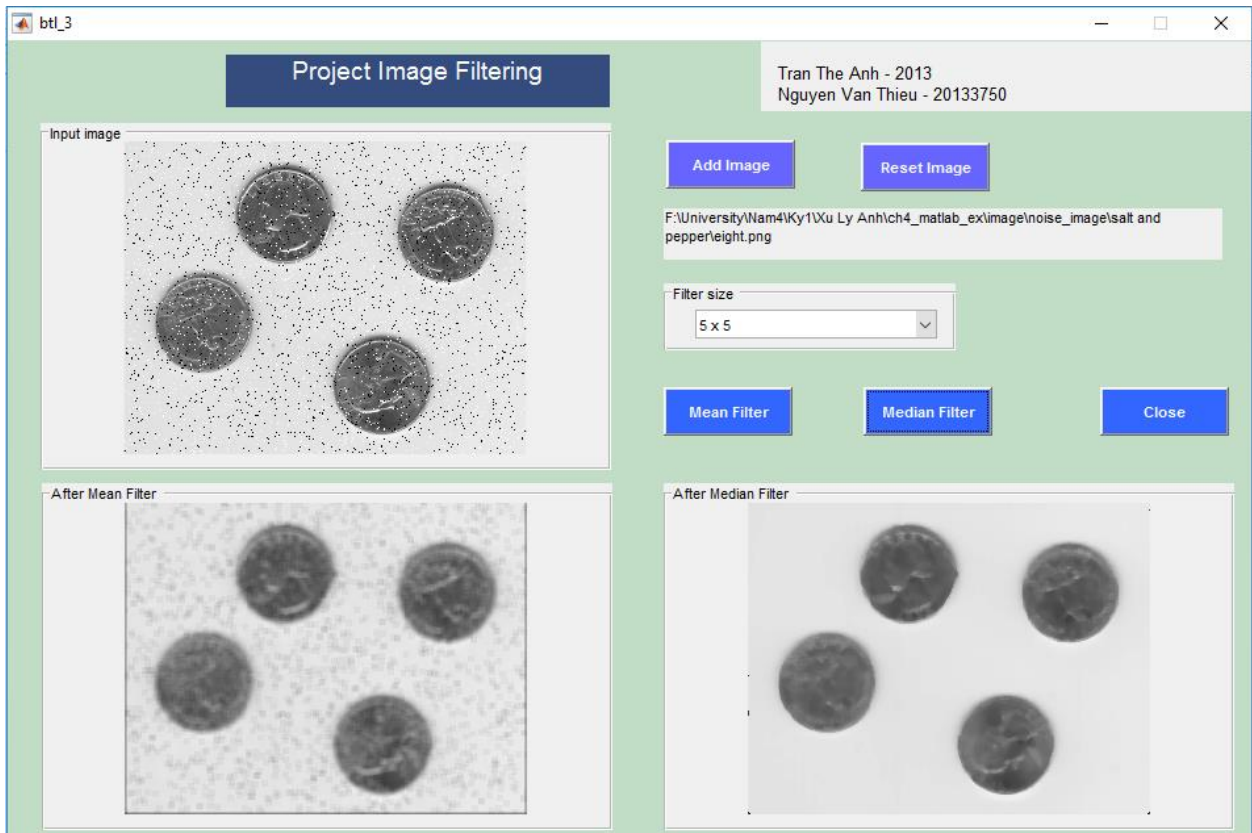


Hình ảnh bé trai bị nhiễu Gauss sau khi qua 2 bộ lọc.

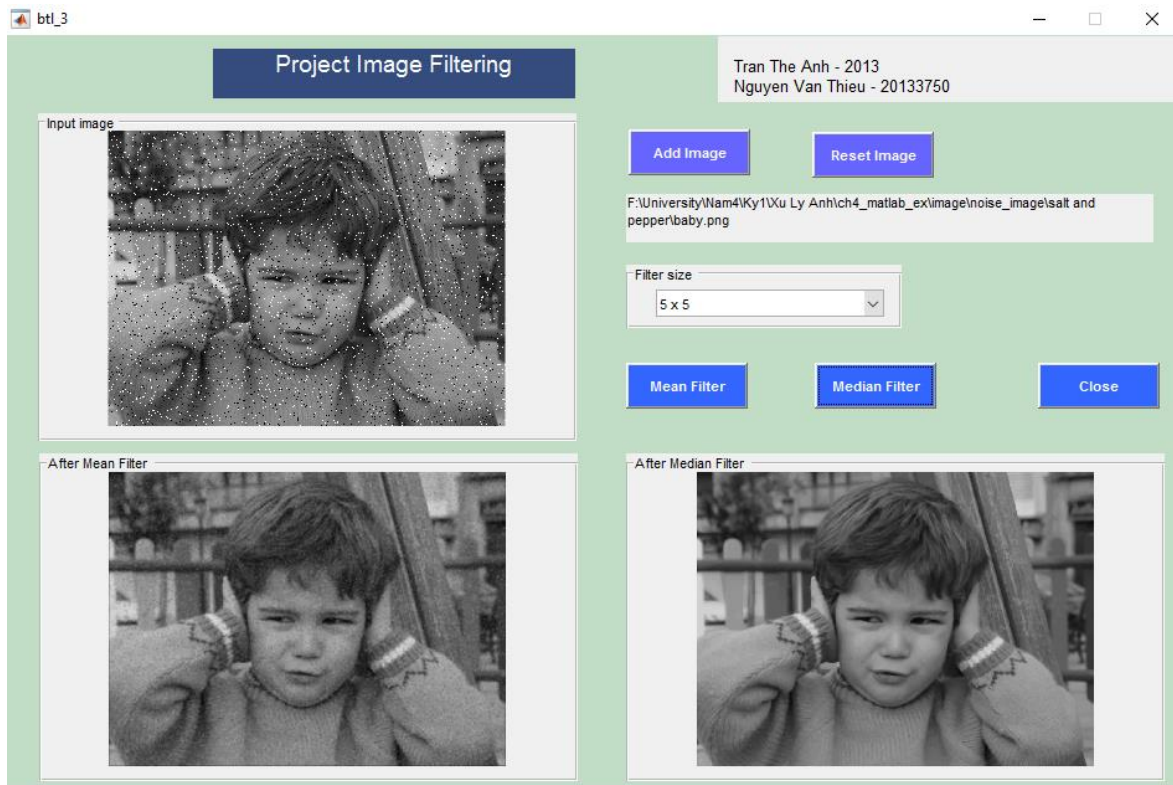


Hình ảnh 1 loạt các loại hoa quả bị nhiễu Gauss sau khi lọc

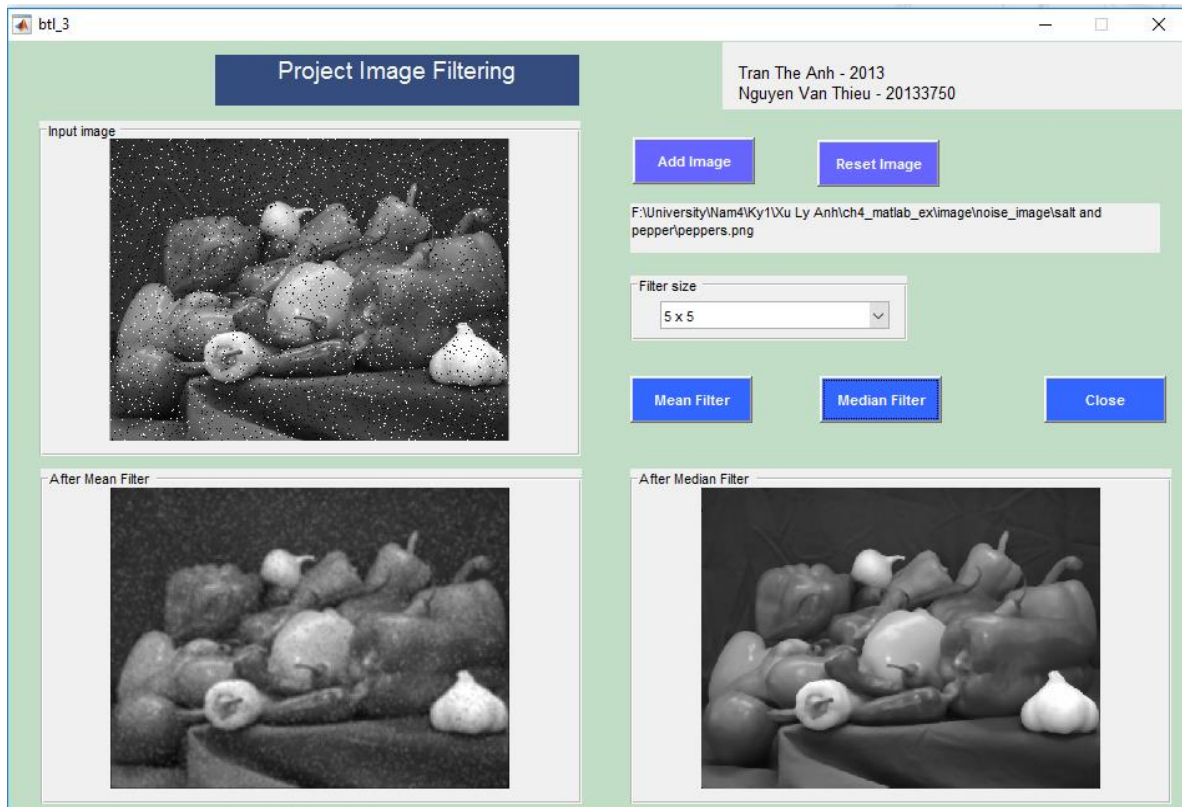
- Bên dưới là 1 số ảnh bị nhiễu Salt & Pepper được lấy làm đầu vào, và đầu ra là ảnh sau khi lọc bởi bộ lọc trung bình (Mean Filter) và trung vị (Median Filter)



Hình ảnh 4 đồng tiền xu bị nhiễu Salt & Peppers sau khi qua 2 bộ lọc Mean và Median



Hình ảnh cậu bé bị nhiễu Salt & Pepper sau khi lọc Mean và Median



Hình ảnh tập các loại hoa quả bị nhiễu Salt & Pepper sau khi lọc Mean và Median

3.4. Phân tích, so sánh 2 phương pháp

- Phân tích:

- Sau khi chạy chương trình và thử nghiệm trên nhiều loại ảnh khác nhau, nhóm em thấy rằng cả 2 bộ lọc đều cho ra kết quả tốt, nhưng tốt với tùy loại ảnh đầu vào. Nếu ảnh đầu vào bị nhiễu gauss thì bộ lọc trung bình cho ra kết quả tốt hơn. Còn nếu ảnh đầu vào bị nhiễu salt và pepper thì bộ lọc trung vị cho ra kết quả tốt hơn.
- Bộ lọc trung bình hiệu quả nhất trong việc khử nhiễu Gaussian nhưng đôi lại là sự mất các chi tiết hình ảnh cao tần. Kích thước kernel lớn hơn sẽ có hiệu quả lọc lớn hơn nhưng làm giảm chất lượng hình ảnh nên lọc này không hiệu quả trong việc khử nhiễu ‘salt and pepper’. Ta có thể cải tiến bộ lọc này bằng cách tạo một ngưỡng và chỉ thay thế các giá trị điểm ảnh hiện tại với giá trị trung bình của vùng lân cận nếu độ lớn của giá trị trung bình nằm nằm dưới ngưỡng này.
- Bộ lọc trung vị sử dụng hiệu quả trong những trường hợp có các giá trị pixel lớn hơn hoặc nhỏ hơn hẳn các giá trị lân cận, ví dụ như nhiễu “salt and pepper”, khi đó các thành phần nhiễu này do có mức xám khác biệt với các điểm lân cận sẽ được thay thế bằng mức xám gần bằng các điểm xung quanh.

- So sánh 2 bộ lọc:

	Bộ lọc trung bình (mean filter)	Bộ lọc trung vị (median filter)
Điểm chung	- Đầu vào ảnh bị nhiễu hoặc không thì ảnh đầu ra sau khi lọc đều được lọc và cải thiện. Tuy nhiên việc cải thiện ở đây chất lượng còn phụ thuộc vào từng bộ lọc cũng như từng loại ảnh bị nhiễu.	
Điểm mạnh	- Với ảnh bị nhiễu Gauss thì bộ lọc này sẽ cho đầu ra tốt (ảnh được lọc nhiễu và làm trơn tốt)	- Với ảnh bị nhiễu Salt & Pepper thì bộ lọc này sẽ cho đầu ra tốt (ảnh được lọc nhiễu và làm trơn tốt)
Điểm hạn chế	- Với ảnh không bị nhiễu Gauss, mà bị nhiễu khác thì ảnh đầu ra chất lượng không được tốt.	- Với ảnh không bị nhiễu Salt & Pepper, mà bị nhiễu khác thì ảnh đầu ra chất lượng không được tốt.

4. Tài liệu tham khảo

[1] Chương 3, **Digital Image Processing**, third edition, Rafael C.Gonzalez, Richard E.Woods

[2] **Giáo trình xử lý ảnh**, *PGS.TS Nguyễn Thị Hoàng Lan*, Đại học Bách Khoa Hà Nội.