Tên: Nguyễn Đình Trí

MSSV: 20120218

Tên học phần: Xử lý ảnh số và video số

Báo cáo đồ án

Mục lục

I. Đánh giá.

Chức năng	Giải thuật	Tiến độ
Biến đổi màu	Tuyến tính	100%
	Phi tuyến	100%
	Cân bằng lược đồ xám	100%
	Đặc tả lược đồ xám	100%
Biến đổi hình học	Phép biến đổi affine với phép nội suy giá trị màu dựa vào người láng giềng gần nhất	100%
Làm trơn ảnh	Toán tử trung bình	100%
	Toán tử trung vị	100%
	Toán tử Gaussian	100%
Phát hiện biên cạnh (edge detection)	Toán tử Gradient	100%
	Toán tử Laplace	100%

II. Các chức năng.

1. Một số hàm chức năng phụ trong code.
   1.1. def OP_Convolution(matrix1, matrix2): Hàm thực hiện toán tử Convolution giữa 2 ma trận 3x3. Kết quả sau khi thực hiện Convolution sẽ được trả về.
      - matrix1, matrix2: Ma trận 3x3
   1.2. def Matrix_split3x3(matrix,x,y): Hàm tách lân cận 8 và điểm ảnh thành ma trận 3x3. Trả về ma trận 3x3
      - matrix: Ma trận giá trị điểm ảnh gốc
      - x,y: Vị trí pixel cần tách
   1.3. def OP_Median(matrix,x,y): Hàm toán tử trung vị. Trả về trung vị của một ma trận 3x3.
      - matrix: Ma trận giá trị điểm ảnh gốc
      - x,y: Vị trí pixel cần xét.
2. Giải thuật biến đổi màu.
   2.1. Phép biến đổi tuyến tính.
      2.1.1. Phương pháp.
         - Thay đổi độ sáng của từng điểm ảnh f (x, y) dựa vào hàm tuyến tính
         - Brightness: g (x, y) = f (x, y) + b)
         - Contrast: g (x, y) = a * f (x, y)

- Brightness + Contrast: g (x, y) = a * f (x, y) + b

2.1.2. Giải thuật.

- B1: Lặp từng pixel trong ảnh gốc.
- B2: Biến đổi điểm ảnh theo a, b theo công thức g=a*f+b.
  - Brightness: (a= 1, b=b).
  - Contrast: (a=a, b=0).
  - Brightness + Contrast: (a=a,b=b).
- B3: Lưu từng điểm ảnh vào g.

2.1.3. Kết quả

Brightness với a=1 và b=50

Contrast với a=1.5 và b=0

Brightness+Contrast với a=1.5 và b=50

2.2. Phép biến đổi phi tuyến.

2.2.1. Phương pháp

- Biến đổi giá trị độ xám của điểm ảnh theo hàm phi tuyến (log(x) và e^x)

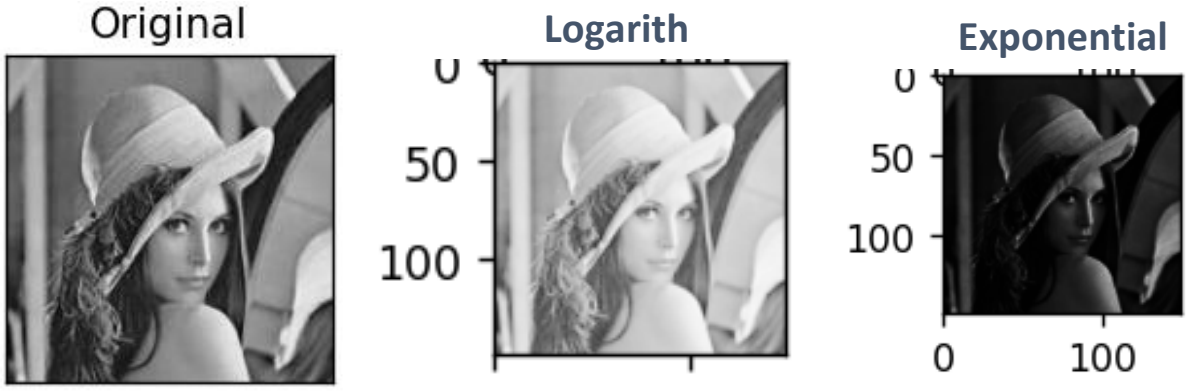
2.2.2. Giải thuật

- B1: Lặp từng pixel trong ảnh gốc.
- B2: Biến đổi điểm ảnh theo a, b theo công thức.

- G(x,y)=c*log(f(x,y))
- $G(x,y)=e^{f(x,y)}$

- B3: Lưu từng điểm ảnh vào G.
  - Chú ý vì hàm e mũ sẽ rất lớn nên ta sẽ giới hạn 0<=G(x,y)<=255 ➔ giảm f(x,y) xuống bằng cách chia cho 46

2.2.3. Kết quả



2.3. Cân bằng lược đồ xám

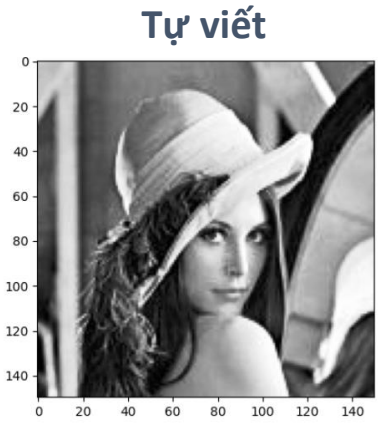
2.3.1. Phương pháp

- Biến đổi giá trị điểm ảnh dựa vào phân bố xác xuất.

2.3.2. Giải thuật

- B1: Tạo mảng h có chiều dài 256 (0->255).
- B2: Tính lược đồ xám của ảnh f lưu vào h.
  - h[f(x,y)] += 1
- B3: Tính lược đồ độ xám tích lũy của f, lưu vào t
  - t[0] = h[0]
  - t[p] = t[p-1] + h[p], với p=1,2,…..,255
- B4: Chuẩn hóa t về đoạn [0,255]
  - t[p] = round((255/(N*M))*t[p])
- B5: Tạo ảnh kết quả des(x,y) = t[f(x,y)]

2.3.3. Kết quả



2.4. Đặc tả lược đồ xám

2.4.1. Phương pháp

- Biến đổi giá trị điểm ảnh dựa vào lược đồ xám cho sẵn g.

2.4.2. Giải thuật

- B1: Tạo mảng h có chiều dài 256 (0->255).
- B2: Tính lược đồ xám của ảnh f lưu vào h.
  - h[f(x,y)] += 1
- B3: Tính lược đồ độ xám tích lũy của f, lưu vào t
  - t[0] = h[0]
  - t[p] = t[p-1] + h[p], với p=1,2,…..,255
- B4: Chuẩn hóa t về đoạn [0,255]
  - t[p] = round((255/(N*M))*t[p])
- B5: Tính lược đồ độ xám tích lũy của g, lưu vào G
  - G[0] = g[0]
  - G[p] = G[p-1] + g[p], với p=1,2,…..,255
- B6: Chuẩn hóa t về đoạn [0,255]
  - G[p] = round((255/(N*M))*g[p])
- B7: Tạo ảnh kết quả
  - des(x,y)= G[t[f(x,y)]]

2.4.3. Kết quả

- Vì chưa có lược đồ xám cho sẵn nên chưa có được ảnh kết quả

3. Phép biến đổi hình học

3.1. Phép biến đổi affine với phép nội suy giá trị màu dựa vào người láng giềng gần nhất.

3.1.1. Phương pháp.

- Biến đổi vị trí mới từng điểm ảnh trong f thành f' thông qua phép ánh xạ T.
- Nhưng nếu làm vậy sẽ xuất hiện lỗ trống trong ảnh kết quả vì vì các tọa độ của ảnh f và f' được xác định trên lưới tọa độ nguyên => Duyệt từ ảnh kết quả để tìm vị trí ở ảnh gốc và sử dụng phép nội suy giá trị màu để suy ra kết quả ử ảnh kết quả.

3.1.2. Giải thuật.

- B1: Xét tọa độ bất kỳ 3 điểm trong ảnh gốc lưu vào mảng souce và 3 điểm tương ứng bên ảnh đích lưu vào mảng des.
- B2: Sử dụng hàm cv2.getAffineTransform(des,souce) để có được ma trận biến đổi 2x3 và lưu vào M.
- B3: Tìm vị trí từng điểm ảnh trong f' trong f.
    - Vì phép biến đổi từ f sang f có dạng là:
        - G(i,j)=a*f(x,y)+b
    - Nên phép biến đổi ngược sẽ là
        - $f(x,y)= a^{-1}*g(i,j)-b$
          với a^{-1} là ma trận 2x2 đầu của M
          b là ma trận 2x1 cuối của M
          M=[a^{-1} b]
- B4: Sau khi tìm được vị trí thực hiện phép nội suy giá trị ảnh để lấy giá trị điểm ảnh cho ảnh gốc.
    - G(I,j)=f(round(x),round(y))
- Ảnh G sau khi được lặp hết là ảnh kết quả.

3.1.3. Kết quả

4. Giải thuật làm trơn ảnh

4.1. Toán tử trung bình

4.1.1. Phương pháp.

- Tính giá trị của điểm ảnh bằng giá trị trung bình của các điểm lân cận của điểm ảnh (tính cả chính điểm ảnh).

4.1.2. Giải thuật

- B1: Tạo ma trận biến đổi H.

  H= 1/9 * [1 1 1
  1 1 1
  1 1 1]

- B2: Tạo ma trận điểm ảnh des có giá trị tất cả là 0 với kích thước bằng với ma trận ảnh gốc.
- B3: Duyệt ma trận gốc từ vị trí 1 -> width-1 và 1->height (không duyệt biên của ảnh).
- B4: Tách lân cận 8 của điểm ảnh gốc thành ma trận 3x3 lưu vào I.
- B5: Thực hiện tích chập I và H lưu kết quả và des. des(i,j)=I.h
- Sau khi duyệt hết sẽ có ma trận điểm ảnh kết quả là des.

4.1.3. Kết quả.

4.2. Toán tử Gaussian

4.2.1. Phương pháp.

- Tính giá trị của điểm ảnh bằng phép tích chập các điểm lân cận của điểm ảnh (tính cả chính điểm ảnh) với ma trận biến đổi Gauss h[].

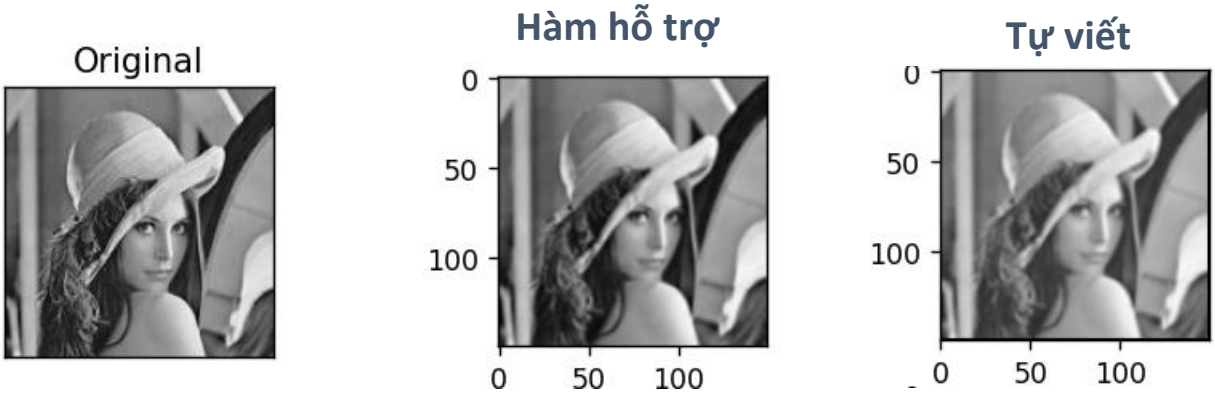
4.2.2. Giải thuật

- B1: Tạo ma trận biến đổi h.

$$h(i,j) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{i^2+j^2}{2\sigma^2}}$$

- B2: Tạo ma trận điểm ảnh des có giá trị tất cả là 0 với kích thước bằng với ma trận ảnh gốc.
- B3: Duyệt ma trận gốc từ vị trí 1 -> width-1 và 1->height (không duyệt biên của ảnh).
- B4: Tách lân cận 8 của điểm ảnh gốc thành ma trận 3x3 lưu vào I.
- B5: Thực hiện tích chập I và H lưu kết quả và des. des(i,j)=I.h
- Sau khi duyệt hết sẽ có ma trận điểm ảnh kết quả là des.

4.2.3. Kết quả



4.3. Toán tử trung vị

4.3.1. Phương pháp.

- Tính giá trị của điểm ảnh bằng phép lấy trung vị các điểm lân cận của điểm ảnh (tính cả chính điểm ảnh).

4.3.2. Giải thuật

- B1: Tạo ma trận điểm ảnh des có giá trị tất cả là 0 với kích thước bằng với ma trận ảnh gốc.
- B2: Duyệt ma trận gốc từ vị trí 1 -> width-1 và 1->height (không duyệt biên của ảnh).
- B3: Tách lân cận 8 của điểm ảnh gốc thành ma trận 3x3 lưu vào I.

- B4: Thực hiện lấy trung vị của ma trận I lưu vào des.

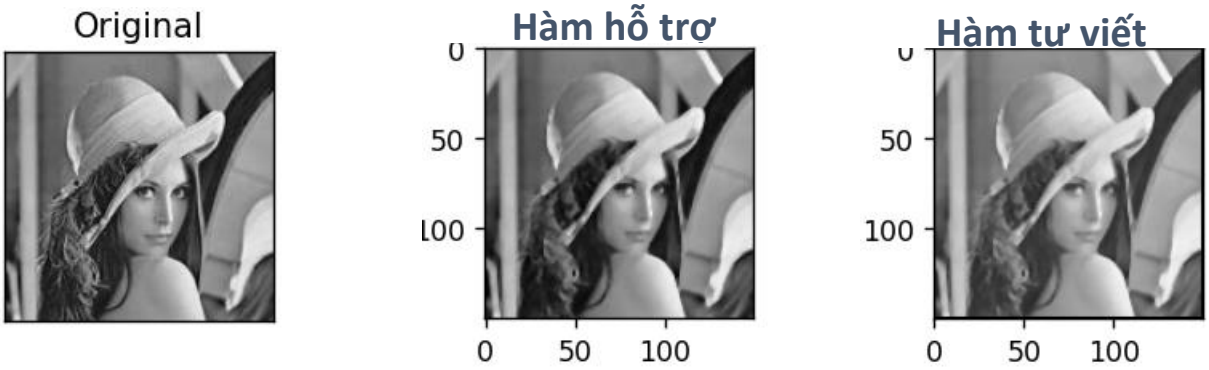
  Giả sử $\{f(x+i, y+j), (i,j) \in O\}$ được sắp thứ tư tăng dần và ký hiệu lại:

  $$I_1 < I_2 < ... < I_n, n = 2v+1$$

  $$med(I_i) = I_{v+1}$$

  des(i,j)=med(I)
- Sau khi duyệt hết sẽ có ma trận điểm ảnh kết quả là des.

4.3.3. Kết quả



5. Giải thuật phát hiện biên cạnh (Edge detection).

5.1. Toán tử Gradient.

5.1.1. Phương pháp

- Tính độ chênh lệch độ xám của từng điểm ảnh với các điểm lân cận bằng vector gradient.
- Những điểm nào có độ chênh lệch cao sẽ là biên.

5.1.2. Giải thuật

- B1: Tạo ra ma trận biến đổi Wx và Wy.
  - Pixel difference

    Wx= [0 0 0
    0 1 -1
    0 0 0]

    Wy= [0 -1 0
    0 1 0
    0 0 0]
  - Separated Pixel difference

    Wx= [0 0 0

1 0 -1
0 0 0]

Wy= [0 -1 0
0 0 0
0 1 0]

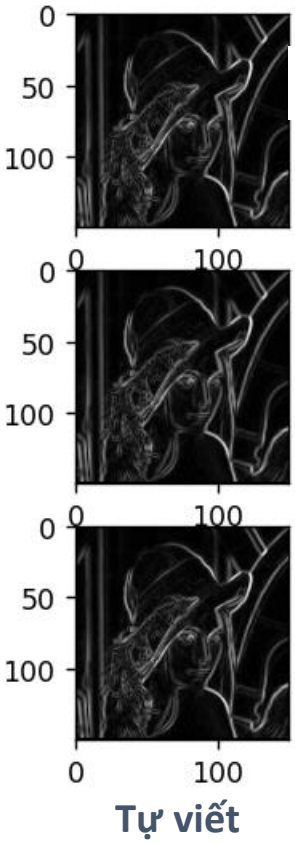
- Prewitt(k=1)
  Sobel(k=2)
  Frei-Chen(k=2sqrt(2))

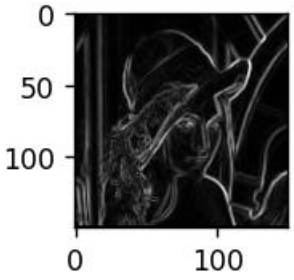
Wx=1/(k+2) [1 0 -1
k 0 -1
1 0 -1]

Wy=1/(k+2) [-1 -k -1
0 0 0
1 k 1]

- B2: Duyệt ma trận gốc từ vị trí 1 -> width-1 và 1->height (không duyệt biên của ảnh).
- B3: Tách lân cận 8 của điểm ảnh gốc thành ma trận 3x3 lưu vào I.
- B4: Thực hiện tích chập I và Wx, I và Wy lưu kết quả.
  Dx=I*Wx
  Dy=I*Wy
- B5: Tính độ lớn là lưu kết quả vào des.
  des[i][j]= sqrt(Dx^2+Dy^2)
- Sau khi duyệt hết sẽ có ma trận điểm ảnh kết quả là des.

5.1.3. Kết quả.





5.2. Toán tử Laplace.

5.2.1. Phương pháp

- Tính độ chênh lệch độ xám của từng điểm ảnh với các điểm lân cận bằng vector gradient.
- Những điểm nào có độ chênh lệch cao sẽ là biên.

5.2.2. Giải thuật

- B1: Tạo ra ma trận biến đổi W.
  Wx=1/(k+2)[1 1 1
  1 -8 1
  1 1 1]
- B2: Duyệt ma trận gốc từ vị trí 1 -> width-1 và 1->height (không duyệt biên của ảnh).
- B3: Tách lân cận 8 của điểm ảnh gốc thành ma trận 3x3 lưu vào I.
- B4: Thực hiện tích chập I và W lưu kết quả.
  des[i][j]= I*W
- Sau khi duyệt hết sẽ có ma trận điểm ảnh kết quả là des.

5.2.3. Kết quả