

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA CÔNG NGHỆ THÔNG TIN 1

XỬ LÝ ẢNH



**Chủ đề 1: Bộ lọc ảnh nâng cao
chất lượng ảnh số**

Giảng viên hướng dẫn	: Phạm Hoàng Việt
Họ và tên sinh viên	: Bùi Tiến Thịnh
Mã sinh viên	: B22DCCN829
Mã lớp	: D22CNPM01
Nhóm lớp	: 02
Nhóm bài tập	: 24

Hà Nội – 2025

LỜI CẢM ƠN

Trước tiên, em xin được gửi lời cảm ơn chân thành và sâu sắc đến Học viện Công nghệ Bưu chính Viễn thông đã đưa môn học **Xử lý ảnh** vào chương trình đào tạo, giúp em có cơ hội tiếp cận những kiến thức thiết thực và hiện đại. Đặc biệt, chúng em xin bày tỏ lòng biết ơn sâu sắc tới thầy **Phạm Hoàng Việt**, giảng viên bộ môn, người đã tận tình giảng dạy, hướng dẫn và truyền đạt cho chúng em những kiến thức quý báu trong suốt quá trình học tập.

Trong suốt thời gian tham gia lớp học, em không chỉ lĩnh hội được những kiến thức chuyên môn bổ ích, mà còn học hỏi được tinh thần làm việc nghiêm túc, phương pháp học tập hiệu quả – những hành trang vô cùng cần thiết cho chặng đường học tập và công tác sau này. Nhờ sự tận tâm dẫn dắt của thầy, em đã có thể hoàn thành một đề tài bài tập lớn hoàn chỉnh cho môn học này – một trải nghiệm quý giá mà em rất trân trọng.

Dưới sự giúp đỡ và hướng dẫn tận tình của thầy **Phạm Hoàng Việt** em đã tìm hiểu và từng bước hoàn thành đề tài của mình. Tuy nhiên bài làm củ em có lẽ vẫn còn nhiều thiếu sót, chúng em rất mong nhận được sự góp ý của thầy để đề tài của chúng em được hoàn thiện hơn.

Em xin chân thành cảm ơn thầy ạ!

Mục lục

I.LÝ THUYẾT VỀ BỘ LỌC LÀM MỊN	7
1. Giới thiệu chung	7
2. Khái niệm và bản chất của bộ lọc làm mịn	7
3. Mục đích và ý nghĩa của làm mịn ảnh	8
4. Cơ chế hoạt động của bộ lọc làm mịn	8
5. Phân loại bộ lọc làm mịn	9
6. Bộ lọc tuyến tính (Linear Filters)	9
7. Bộ lọc phi tuyến (Non-linear Filters)	10
8. Mối quan hệ giữa làm mịn và miền tần số	10
9. Ảnh hưởng và hạn chế	11
10. Một số tham số ảnh hưởng	11
11. Ứng dụng của bộ lọc làm mịn	11
12. Kết luận	12
II.LÝ THUYẾT VỀ BỘ LỌC TRUNG BÌNH (MEAN FILTER)	12
1. Khái niệm chung	12
2. Mục đích và ý nghĩa	12
3. Nguyên lý hoạt động	13
4. Đặc điểm toán học và vật lý	14
5. Biểu diễn trong miền tần số	14
6. Ảnh hưởng của kích thước mặt nạ	14
7. Ảnh hưởng đến cấu trúc ảnh	15
8. Các biến thể của Mean Filter	15
9. Ưu điểm và hạn chế (về lý thuyết)	16
10. Vai trò trong chuỗi xử lý ảnh	16
11. Kết luận	16
12.Code minh họa bộ lọc Mean Filter	17
13.Ảnh minh họa bộ lọc Mean Filter	18
III.LÝ THUYẾT VỀ BỘ LỌC TRUNG VỊ (MEDIAN FILTER)	18
1. Khái niệm chung	18
2. Mục đích và ý nghĩa	18
3. Nguyên lý hoạt động	19
4. Đặc điểm toán học và tính chất	20
5. Cấu trúc mặt nạ (Mask / Kernel)	20

6. Ảnh hưởng trong miền tần số	20
7. Ảnh hưởng của kích thước mặt nạ	21
8. Các biến thể của bộ lọc trung vị	21
9. Ưu điểm và hạn chế (theoretical aspects).....	21
10. Ảnh hưởng đến cấu trúc và biên ảnh.....	22
11. Ứng dụng thực tế	22
12. Vị trí trong pipeline xử lý ảnh	22
13. Kết luận	22
14.Code minh họa bộ lọc Median Filter.....	23
15.Ảnh minh họa bộ lọc Median Filter	23
IV.LÝ THUYẾT VỀ BỘ LỌC GAUSSIAN (GAUSSIAN FILTER)	23
1. Khái niệm chung	23
2. Mục đích và vai trò.....	24
3. Nguyên lý hoạt động	24
4. Cấu trúc mặt nạ (kernel).....	25
5. Biểu diễn trong miền tần số.....	25
6. Ảnh hưởng của tham số σ (độ lệch chuẩn)	26
7. Đặc điểm và tính chất.....	26
8. Ảnh hưởng đối với cấu trúc ảnh.....	26
9. Ứng dụng trong xử lý ảnh	27
10. Mối quan hệ với Mean Filter.....	27
11. Tính chất đặc biệt trong miền tần số	27
12. Kết luận	27
13.Code minh họa bộ lọc Gaussian Filter	28
14.Ảnh minh họa thực tế bộ lọc Gaussian Filter.....	29
V.LÝ THUYẾT CHUNG VỀ BỘ LỌC PHÁT HIỆN BIÊN (EDGE DETECTION FILTER)	29
1.Khái niệm chung	29
2. Mục đích và vai trò của phát hiện biên	29
3. Nguyên lý hoạt động chung	30
4. Cơ sở toán học của phát hiện biên.....	31
5. Cấu trúc mặt nạ (Mask / Kernel).....	31
6. Nguyên lý hoạt động tổng quát của quá trình phát hiện biên.....	32
7. Ảnh hưởng và đặc điểm của phát hiện biên	32
8. Phát hiện biên trong miền tần số	32

9. Đặc điểm và vai trò trong chuỗi xử lý ảnh.....	33
10. Ứng dụng thực tế của phát hiện biên.....	33
11. Ảnh hưởng và các yếu tố quyết định chất lượng phát hiện biên.....	34
12. Kết luận	34
VI.LÝ THUYẾT VỀ BỘ LỌC LAPLACIAN	34
1. Khái niệm chung	34
2. Mục đích và vai trò.....	35
3. Cơ sở toán học.....	35
4. Các mặt nạ Laplacian phổ biến	35
6. Nguyên lý phát hiện biên bằng đạo hàm bậc hai	36
7. Đặc điểm và tính chất.....	37
8. Biểu diễn trong miền tần số.....	37
9. Ảnh hưởng đến ảnh	37
10. Laplacian of Gaussian (LoG)	38
11. Ưu điểm và hạn chế (theoretical)	38
12. Ứng dụng thực tế.....	39
13. Kết luận	39
14.Code minh họa bộ lọc Laplacian.....	40
15.Ảnh minh họa thực tế bộ lọc Laplacian	41
VII.LÝ THUYẾT VỀ BỘ LỌC PREWITT (PREWITT FILTER / PREWITT OPERATOR.....	41
1. Khái niệm chung	41
2. Mục đích và vai trò.....	41
3. Nguyên lý hoạt động	42
4. Cấu trúc mặt nạ Prewitt.....	42
5. Công thức tính toán	43
6. Ý nghĩa hướng biên.....	43
7. Đặc điểm và tính chất.....	44
8. Ảnh hưởng đối với ảnh.....	44
9. Biểu diễn trong miền tần số.....	44
10. Quy trình phát hiện biên bằng Prewitt	45
11. Ưu điểm và hạn chế (về lý thuyết)	45
12. Ứng dụng thực tế.....	45
13. Kết luận	46
14.Code minh họa bộ lọc Prewitt.....	46

15.Ảnh minh họa thực tế bộ lọc Prewitt.....	47
---------------------------------------------	----

VI.LÝ THUYẾT VỀ BỘ LỌC SOBEL (SOBEL FILTER / SOBEL OPERATOR)

.....	47
1. Khái niệm chung	47
2. Mục đích và vai trò.....	47
3. Nguyên lý hoạt động	48
4. Cấu trúc mặt nạ Sobel	48
5. Công thức tính toán	49
6. Ý nghĩa và hướng của biên.....	49
7. Đặc điểm nổi bật.....	50
8. Đặc tính toán học.....	50
9. Biểu diễn trong miền tần số.....	50
10. Ảnh hưởng đến ảnh sau lọc	51
11. Quá trình xử lý tổng quát.....	51
12. Ưu điểm và hạn chế (về lý thuyết)	51
13. Ứng dụng thực tế.....	51
14. Kết luận	52
15.Code minh họa bộ lọc Sobel	52
16.Ảnh minh họa thực tế bộ lọc Sobel	53

IX.SO SÁNH CÁC BỘ LỌC TRONG XỬ LÝ ẢNH SỐ

1. Phân nhóm chức năng	53
2. Bảng so sánh tổng quát 6 bộ lọc.....	53
3. Nhận xét chi tiết theo từng tiêu chí	55
4. Môi quan hệ giữa các bộ lọc	56
5. Môi quan hệ theo miền tần số	57
6. Kết luận tổng quát	57

I. LÝ THUYẾT VỀ BỘ LỌC LÀM MỊN

1. Giới thiệu chung

Trong quá trình xử lý ảnh, một trong nhng vấn đề cơ bản và quan trọng nhất là làm giảm nhiễu (noise reduction) và cải thiện chất lượng hiển thị của ảnh.

Ảnh thu được từ máy ảnh, cảm biến, thiết bị quét hoặc truyền qua mạng thường chứa nhiễu ngẫu nhiên do nhiều nguyên nhân khác nhau như:

- Sự dao động của ánh sáng,
- Nhiễu điện tử từ cảm biến,
- Lỗi lượng tử hóa,
- Sai lệch khi nén ảnh hoặc truyền dữ liệu.

Những loại nhiễu này làm ảnh bị hạt, đốm, mờ, mất chi tiết nhỏ và ảnh hưởng nghiêm trọng đến các bước xử lý tiếp theo như phát hiện biên (edge detection), phân đoạn (segmentation) hay nhận dạng (recognition).

Để khắc phục, người ta sử dụng bộ lọc làm mịn (smoothing filters) — đây là nhóm các phép toán không gian (spatial filters) được thiết kế nhằm làm giảm các biến thiên cường độ sáng nhỏ giữa các điểm ảnh lân cận, giúp ảnh trở nên “mượt” hơn, ít nhiễu hơn, nhưng vẫn giữ được cấu trúc tổng thể.

2. Khái niệm và bản chất của bộ lọc làm mịn

2.1. Định nghĩa

Bộ lọc làm mịn là một phép xử lý ảnh nhằm thay thế giá trị của mỗi điểm ảnh bằng một giá trị được tính toán từ một tập hợp các điểm lân cận xung quanh nó (gọi là *cửa sổ lọc* hoặc *mask/kernel*).

Mục đích là giảm sự biến động cục bộ của mức xám hoặc cường độ sáng, từ đó loại bỏ các chi tiết nhiễu, đường viền yếu, hoặc điểm sáng bất thường.

2.2. Bản chất toán học

Nếu ký hiệu:

- $f(x, y)$: ảnh gốc (giá trị cường độ sáng tại vị trí (x, y))
- $g(x, y)$: ảnh sau khi lọc
- $w(s, t)$: mặt nạ (kernel, filter mask) có kích thước $(2a + 1) \times (2b + 1)$

Thì phép lọc không gian tuyến tính được mô tả bằng:

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

Trong đó:

- $w(s, t)$ là trọng số cho mỗi điểm ảnh trong vùng lân cận;
- Tổng các trọng số thường được chuẩn hóa:

$$\sum_{s,t} w(s, t) = 1$$

nhằm bảo toàn độ sáng trung bình của ảnh.

3. Mục đích và ý nghĩa của làm mịn ảnh

3.1. Giảm nhiễu (Noise Reduction)

Mỗi điểm nhiễu trong ảnh thường là giá trị bất thường so với vùng lân cận. Khi tính trung bình (hoặc trung vị), những giá trị bất thường này bị “triệt tiêu” nhờ ảnh hưởng nhỏ trong tổng thể → ảnh trở nên mượt hơn.

3.2. Làm trơn vùng đồng nhất

Trong vùng mà cường độ sáng không thay đổi nhiều, làm mịn giúp tạo ra sự đồng nhất về sáng tối, giúp ảnh có vẻ “phẳng” hơn và giảm sần sùi.

3.3. Cải thiện khả năng phát hiện biên thật

Khi ảnh chứa quá nhiều nhiễu, các thuật toán phát hiện biên (Sobel, Canny, Laplacian, v.v.) có thể nhầm lẫn nhiễu với biên vật thể.

Làm mịn trước khi phát hiện biên giúp loại bỏ những dao động nhỏ không mong muốn → chỉ còn lại biên thực sự rõ ràng.

3.4. Chuẩn bị cho các bước xử lý nâng cao

Làm mịn ảnh là bước tiền xử lý (pre-processing) bắt buộc trong nhiều ứng dụng:

- Phân vùng ảnh (image segmentation)
- Nhận dạng đối tượng (object recognition)
- Trích đặc trưng (feature extraction)
- Xử lý ảnh y tế, vệ tinh, công nghiệp,...

4. Cơ chế hoạt động của bộ lọc làm mịn

Bộ lọc làm mịn hoạt động trên nguyên tắc tính toán lại giá trị của điểm ảnh dựa vào các điểm xung quanh.

Kích thước của vùng lân cận thường là 3×3 , 5×5 , hoặc 7×7 pixel, tùy yêu cầu.

Khi mặt nạ di chuyển trên toàn ảnh:

- Với mỗi vị trí, mặt nạ lấy mẫu các giá trị xung quanh,
- Áp dụng trọng số tương ứng,
- Và tính giá trị mới thay cho điểm ảnh trung tâm.

Nếu trọng số đều nhau → Mean Filter.

Nếu trọng số phân bố theo hàm Gauss → Gaussian Filter.

Kết quả là ảnh đầu ra có ít chi tiết nhiều hơn, nhưng các đường biên có thể bị làm mờ.

5. Phân loại bộ lọc làm mịn

Tùy theo tính chất toán học, bộ lọc làm mịn được chia thành hai nhóm lớn:

Nhóm	Nguyên lý	Đặc điểm
Tuyến tính (Linear filters)	Giá trị điểm ảnh mới = tổ hợp tuyến tính của các điểm lân cận	Dễ tính, phù hợp cho nhiễu ngẫu nhiên
Phi tuyến (Non-linear filters)	Sử dụng phép toán phi tuyến như lấy trung vị, min, max...	Hiệu quả với nhiễu “muối tiêu”, giữ biên tốt hơn

6. Bộ lọc tuyến tính (Linear Filters)

6.1. Nguyên lý

Lọc tuyến tính thực hiện phép tích chập (convolution) giữa ảnh gốc và mặt nạ:

$$g(x, y) = (f * w)(x, y) = \sum_{s, t} f(x + s, y + t) \cdot w(s, t)$$

6.2. Một số đặc điểm

- Dễ thực hiện bằng phép nhân và cộng.
- Giảm được nhiễu ngẫu nhiên (Gaussian noise).
- Tuy nhiên, có xu hướng làm mờ biên và giảm độ tương phản ở vùng ranh giới vật thể.

6.3. Ví dụ các bộ lọc tuyến tính phổ biến

- **Mean Filter (trung bình):** mỗi điểm ảnh mới bằng trung bình của vùng lân cận.

Ví dụ mặt nạ 3×3:

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

- **Gaussian Filter:** trọng số phân bố theo hàm Gauss \rightarrow điểm gần tâm có ảnh hưởng mạnh hơn.

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Gaussian Filter cho kết quả mịn tự nhiên, ít làm mờ biên hơn Mean Filter.

7. Bộ lọc phi tuyến (Non-linear Filters)

7.1. Nguyên lý

Không dựa vào phép cộng – nhân tuyến tính, mà dùng các phép thống kê hoặc so sánh giá trị.

7.2. Bộ lọc trung vị (Median Filter)

Đây là bộ lọc phi tuyến phổ biến nhất.

Thay vì lấy trung bình, bộ lọc trung vị sắp xếp các giá trị pixel trong vùng lân cận và lấy giá trị ở giữa (median) làm giá trị mới.

Ví dụ:

Vùng 3×3 có giá trị [11, 12, 11, 200, 13, 12, 11, 12, 10]

\rightarrow Trung vị = 12 \rightarrow pixel mới = 12

\rightarrow loại bỏ hoàn toàn giá trị ngoại lai 200 (nhiều).

7.3. Ưu điểm

- Loại bỏ hiệu quả nhiễu muối tiêu (salt-and-pepper noise).
- Giữ nguyên được đường biên sắc nét.

7.4. Nhược điểm

- Tính toán phức tạp hơn.
- Không áp dụng được trong miền tần số.
- Với ảnh lớn, thời gian xử lý lâu.

8. Mối quan hệ giữa làm mịn và miền tần số

Ảnh có thể được xem là tín hiệu hai chiều.

- Thành phần tần số thấp \rightarrow biểu diễn các vùng sáng/tối lớn (vùng đồng nhất).
- Thành phần tần số cao \rightarrow biểu diễn các chi tiết nhỏ, biên, hoặc nhiễu.

Bộ lọc làm mịn tương đương với “bộ lọc thông thấp (Low-pass filter)”, nghĩa là nó giữ lại thành phần tần số thấp và loại bỏ tần số cao.

Công thức trong miền tần số Fourier:

$$G(u, v) = H(u, v) \cdot F(u, v)$$

Trong đó:

- $F(u, v)$: biến đổi Fourier của ảnh gốc
- $H(u, v)$: hàm truyền (transfer function) của bộ lọc
- $G(u, v)$: ảnh sau khi lọc

Ý nghĩa: Làm mịn ảnh trong miền không gian \Leftrightarrow triệt tiêu tần số cao trong miền tần số.

9. Ảnh hưởng và hạn chế

9.1. Ưu điểm

- Giảm nhiễu rõ rệt.
- Làm mượt vùng đồng nhất.
- Làm nổi bật cấu trúc lớn của ảnh.
- Giúp chuẩn hóa dữ liệu cho xử lý tiếp theo.

9.2. Hạn chế

- Làm mờ biên vật thể.
- Mất chi tiết nhỏ, hoa văn tinh tế.
- Cần chọn kích thước mặt nạ phù hợp (kernel quá lớn sẽ làm ảnh mờ toàn phần).

10. Một số tham số ảnh hưởng

Tham số	Ảnh hưởng
Kích thước mặt nạ (kernel size)	Cửa sổ càng lớn \rightarrow làm mịn mạnh hơn nhưng mất chi tiết nhiều hơn
Phân bố trọng số (weights)	Nếu đều nhau \rightarrow mờ nhiều; nếu tập trung ở tâm \rightarrow giữ biên tốt hơn
Phương pháp xử lý biên (padding)	Ảnh hưởng tới viền ảnh (zero-padding, replicate, mirror)
Độ lệch chuẩn σ trong Gaussian Filter	σ lớn \rightarrow mờ nhiều; σ nhỏ \rightarrow mờ nhẹ

11. Ứng dụng của bộ lọc làm mịn

- Giảm nhiễu trước khi phát hiện biên, phân đoạn ảnh.
- Làm đẹp ảnh chụp (mặt, da, cảnh quan).
- Xử lý ảnh y tế (MRI, X-quang) để tách mô rõ hơn.
- Tăng chất lượng ảnh trong truyền hình, vệ tinh, hoặc ảnh công nghiệp.
- Chuẩn bị dữ liệu cho mạng nơ-ron học sâu (Deep Learning).

12. Kết luận

Bộ lọc làm mịn (Smoothing Filter) là kỹ thuật cơ bản nhưng vô cùng quan trọng trong xử lý ảnh số.

Nó giúp loại bỏ nhiễu, làm phẳng vùng đồng nhất, đồng thời tạo tiền đề cho các kỹ thuật phức tạp hơn như phát hiện biên, trích đặc trưng, phân đoạn và nhận dạng đối tượng.

Tuy nhiên, cần lựa chọn loại bộ lọc và tham số phù hợp để đạt được cân bằng giữa:

“Giảm nhiễu tốt” và “Giữ chi tiết ảnh”.

Các bộ lọc như Mean, Gaussian, Median, Bilateral là nền tảng quan trọng, mỗi loại có ưu – nhược điểm riêng tùy theo loại nhiễu và mục đích xử lý.

II. LÝ THUYẾT VỀ BỘ LỌC TRUNG BÌNH (MEAN FILTER)

1. Khái niệm chung

Bộ lọc trung bình (Mean Filter) là một trong những bộ lọc cơ bản nhất thuộc nhóm bộ lọc làm mịn tuyến tính (Linear Smoothing Filters) trong xử lý ảnh số.

Bộ lọc này hoạt động dựa trên nguyên lý tính trung bình cộng giá trị cường độ sáng (intensity) của các điểm ảnh lân cận trong một vùng xác định (neighborhood), sau đó gán kết quả trung bình đó cho điểm ảnh trung tâm.

Mục đích của Mean Filter là làm giảm nhiễu (noise) và làm trơn ảnh (smoothing) bằng cách loại bỏ các dao động cường độ nhỏ, giữ lại xu hướng tổng thể của vùng ảnh.

Khi áp dụng bộ lọc trung bình, các chi tiết nhỏ hoặc nhiễu lẻ tẻ sẽ bị triệt tiêu, trong khi các vùng sáng – tối lớn và đồng nhất trở nên phẳng hơn, mượt hơn.

Vì tính chất đơn giản, bộ lọc trung bình được xem là nền tảng của nhiều kỹ thuật lọc nâng cao như Gaussian Filter, Bilateral Filter hoặc LoG (Laplacian of Gaussian).

2. Mục đích và ý nghĩa

Việc áp dụng bộ lọc trung bình có ý nghĩa quan trọng trong các bước tiền xử lý (preprocessing) của xử lý ảnh.

Cụ thể, Mean Filter được dùng để:

1. Giảm nhiễu ngẫu nhiên (Random Noise):

Những điểm ảnh có giá trị cường độ lệch đáng kể so với vùng xung quanh sẽ được thay thế bằng trung bình của vùng lân cận, giúp ảnh “ổn định” và ít dao động hơn.

2. Làm mịn vùng đồng nhất:

Trong những vùng ảnh có giá trị cường độ tương đối giống nhau, Mean Filter giúp làm phẳng mức xám, tạo nên các vùng mượt, ít chên lệch.

3. Chuẩn bị cho các bước xử lý sau:

Làm mịn ảnh trước khi phát hiện biên, phân vùng, hay trích đặc trưng giúp kết quả chính xác và ổn định hơn.

4. Tăng chất lượng hiển thị ảnh:

Trong nhiều trường hợp, Mean Filter giúp giảm “hạt” (grain) hoặc hiện tượng “nhiều muối tiêu nhẹ”, làm ảnh trông mượt và dễ quan sát hơn.

3. Nguyên lý hoạt động

3.1. Cơ chế tổng quát

Bộ lọc trung bình sử dụng một mặt nạ (mask/kernel) hình vuông hoặc hình chữ nhật có kích thước xác định (thường là 3×3 , 5×5 hoặc 7×7).

Mặt nạ này được trượt qua từng điểm ảnh trong ảnh gốc.

Tại mỗi vị trí, giá trị trung bình của tất cả các điểm ảnh nằm trong mặt nạ được tính toán và gán cho điểm ảnh trung tâm.

Giả sử ảnh gốc là $f(x, y)$, ảnh sau khi lọc là $g(x, y)$.

Khi đó, công thức tổng quát của bộ lọc trung bình là:

$$g(x, y) = \frac{1}{N} \sum_{(s,t) \in W} f(x + s, y + t)$$

Trong đó:

- W : vùng lân cận (window) quanh điểm (x, y) , kích thước $m \times n$.
- $N = m \times n$: tổng số điểm ảnh trong vùng mặt nạ.
- $f(x + s, y + t)$: giá trị cường độ sáng tại vị trí lân cận.
- $g(x, y)$: giá trị cường độ mới tại điểm ảnh trung tâm.

Mỗi điểm trong vùng lân cận đóng góp như nhau (trọng số bằng nhau) vào giá trị trung bình.

3.2. Mặt nạ (mask/kernel) chuẩn

a. Mặt nạ 3×3 :

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

b. Mặt nạ 5×5 :

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

→ Tất cả phần tử trong mặt nạ đều bằng nhau và có tổng bằng 1.

4. Đặc điểm toán học và vật lý

4.1. Tính chất tuyến tính

Bộ lọc trung bình là một bộ lọc tuyến tính (Linear Filter), vì quá trình lọc tuân theo tính chất tuyến tính:

$$af_1(x, y) + bf_2(x, y) \Rightarrow ag_1(x, y) + bg_2(x, y)$$

Điều này có nghĩa là tổng của các ảnh sau khi lọc bằng tổng các ảnh được lọc riêng rẽ.

4.2. Tính chất chuẩn hóa

Tổng các trọng số trong mặt nạ luôn bằng 1:

$$\sum_{s,t} w(s, t) = 1$$

Điều này giúp bảo toàn tổng độ sáng trung bình của ảnh, nghĩa là ảnh sau lọc không bị sáng hơn hoặc tối đi so với ảnh gốc.

4.3. Ảnh hưởng vật lý

Bộ lọc trung bình làm giảm sự biến động cường độ sáng giữa các điểm ảnh lân cận. Nó hoạt động như một bộ lọc thông thấp (Low-pass Filter), loại bỏ các thành phần tần số cao (chi tiết nhỏ, biên, nhiễu) và chỉ giữ lại các thành phần tần số thấp (vùng mịn, phẳng).

5. Biểu diễn trong miền tần số

Trong miền tần số, quá trình làm mịn bằng bộ lọc trung bình tương đương với việc nhân phổ tần của ảnh với hàm truyền (transfer function) của bộ lọc thông thấp.

$$G(u, v) = H(u, v) \cdot F(u, v)$$

Trong đó:

- $F(u, v)$: biến đổi Fourier của ảnh gốc.
- $H(u, v)$: hàm truyền của bộ lọc trung bình.
- $G(u, v)$: phổ tần của ảnh sau khi lọc.

Bộ lọc trung bình làm giảm cường độ các tần số cao (đại diện cho chi tiết và biên) và giữ nguyên tần số thấp (đại diện cho vùng đồng nhất).

6. Ảnh hưởng của kích thước mặt nạ

- Khi kích thước mặt nạ nhỏ (3×3): ảnh được làm mịn nhẹ, giữ lại nhiều chi tiết.

- Khi kích thước mặt nạ lớn (5×5 , 7×7 , 9×9): ảnh mịn hơn, nhưng các biên và chi tiết nhỏ sẽ bị mờ đi đáng kể.
- Nếu mặt nạ quá lớn, ảnh có thể bị “mất nét toàn cục”.

Do đó, việc lựa chọn kích thước mặt nạ phụ thuộc vào mức độ nhiễu của ảnh và yêu cầu giữ chi tiết.

7. Ảnh hưởng đến cấu trúc ảnh

1. Giảm nhiễu hiệu quả:

Bộ lọc trung bình rất tốt trong việc loại bỏ nhiễu ngẫu nhiên (random noise), đặc biệt là nhiễu Gaussian nhẹ.

2. Làm mờ biên (Edge blurring):

Vì biên cũng là vùng có sự thay đổi cường độ mạnh, bộ lọc trung bình có xu hướng làm mờ biên giống như làm mờ nhiễu.

3. Giảm độ tương phản cục bộ:

Các vùng chuyển sáng-tối rõ ràng bị trung hòa dần \rightarrow ảnh trông “mềm” và “mượt” hơn.

4. Làm giảm chi tiết nhỏ:

Các đường mảnh, hoa văn nhỏ, chữ nhỏ trong ảnh có thể biến mất sau khi áp dụng bộ lọc.

8. Các biến thể của Mean Filter

Bộ lọc trung bình có nhiều biến thể dựa trên cách tính trung bình:

8.1. Arithmetic Mean Filter

Đây là dạng chuẩn của Mean Filter, tính trung bình cộng thông thường. Phù hợp cho nhiễu Gaussian, nhưng không tốt với nhiễu muối tiêu.

8.2. Geometric Mean Filter

Tính trung bình nhân:

$$g(x, y) = \left[\prod_{(s, t) \in W} f(x + s, y + t) \right]^{1/N}$$

Giữ chi tiết biên tốt hơn một chút, dùng trong ảnh vệ tinh hoặc ảnh radar.

8.3. Harmonic Mean Filter

Dựa trên trung bình điều hòa:

$$g(x, y) = \frac{N}{\sum_{(s, t) \in W} \frac{1}{f(x + s, y + t)}}$$

Hiệu quả với nhiễu "pepper" (điểm đen).

8.4. Contra-harmonic Mean Filter

Công thức tổng quát hơn:

$$g(x, y) = \frac{\sum f^{q+1}(x + s, y + t)}{\sum f^q(x + s, y + t)}$$

- Nếu $q > 0$: khử nhiễu "pepper".
- Nếu $q < 0$: khử nhiễu "salt".

9. Ưu điểm và hạn chế (về lý thuyết)

Ưu điểm:

- Dễ hiểu, dễ cài đặt, nhanh chóng.
- Là bộ lọc tuyến tính đơn giản nhất và có độ ổn định cao.
- Giảm nhiễu ngẫu nhiên hiệu quả.
- Giữ nguyên tổng độ sáng trung bình của ảnh.

Hạn chế:

- Làm mờ biên, mất chi tiết nhỏ.
- Không phân biệt nhiễu và cấu trúc thật (nếu vùng nhỏ mà biến thiên mạnh, cũng bị coi là nhiễu).
- Không thích hợp cho nhiễu muối tiêu mạnh.

10. Vai trò trong chuỗi xử lý ảnh

Bộ lọc trung bình thường được sử dụng ở giai đoạn đầu tiên của pipeline xử lý ảnh - trước khi phát hiện biên, phân vùng, hoặc trích đặc trưng.

Ví dụ quy trình tiêu chuẩn:

Ảnh gốc → Làm mịn bằng Mean Filter → Phát hiện biên bằng Sobel hoặc Canny → Phân tích đặc trưng.

Nhờ khả năng giảm nhiễu, Mean Filter giúp các bộ lọc phát hiện biên hoạt động chính xác hơn và tránh phát hiện sai biên do nhiễu.

11. Kết luận

Bộ lọc trung bình (Mean Filter) là kỹ thuật cơ bản nhưng cực kỳ quan trọng trong xử lý ảnh số.

Nó giúp làm giảm nhiễu, làm phẳng vùng đồng nhất, đồng thời ổn định cường độ sáng trong ảnh.

Dù có hạn chế là làm mờ biên và mất chi tiết, Mean Filter vẫn là công cụ nền tảng cho nhiều thuật toán nâng cao hơn (như Gaussian, Bilateral hoặc LoG).

Với nguyên lý đơn giản, hiệu quả cao và tốc độ tính toán nhanh, bộ lọc trung bình luôn được dùng như bước tiền xử lý quan trọng nhất trong mọi pipeline xử lý ảnh.

12.Code minh họa bộ lọc Mean Filter

```
def mean_filter(gray: np.ndarray, ksize: int = 3) -> np.ndarray:
    if gray.ndim != 2:
        raise ValueError("mean_filter chỉ xử lý ảnh xám 2D (H, W).")

    if ksize % 2 == 0:
        ksize += 1
    if ksize < 3:
        ksize = 3

    g = gray.astype(np.float32)
    pad = ksize // 2

    # Padding ảnh ở biên bằng replicate (lấp biên)
    padded = np.pad(g, pad_width=pad, mode="edge")

    H, W = g.shape
    out = np.zeros_like(g, dtype=np.float32)

    kernel = np.ones((ksize, ksize), dtype=np.float32) / (ksize * ksize)

    for i in range(H):
        for j in range(W):
            region = padded[i:i+ksize, j:j+ksize]
            out[i, j] = np.sum(region * kernel)

    return out
```

13. Ảnh minh họa bộ lọc Mean Filter



III. LÝ THUYẾT VỀ BỘ LỌC TRUNG VỊ (MEDIAN FILTER)

1. Khái niệm chung

Bộ lọc trung vị (Median Filter) là một bộ lọc làm mịn phi tuyến (Non-linear Smoothing Filter) được sử dụng rộng rãi trong xử lý ảnh số để loại bỏ nhiễu đột biến (impulse noise) — đặc biệt là nhiễu muối tiêu (salt-and-pepper noise) — mà vẫn bảo toàn được biên (edge-preserving) của ảnh.

Thay vì tính trung bình cộng như Mean Filter, bộ lọc Median lấy giá trị trung vị (median) của các điểm ảnh trong vùng lân cận làm giá trị mới cho điểm ảnh trung tâm. Cách làm này giúp loại bỏ hiệu quả các giá trị nhiễu cực trị (outliers) mà không bị ảnh hưởng mạnh bởi chúng.

2. Mục đích và ý nghĩa

Bộ lọc trung vị được thiết kế nhằm:

1. Khử nhiễu muối tiêu (Salt-and-Pepper Noise):

Đây là loại nhiễu xuất hiện dưới dạng các điểm trắng hoặc đen rải rác trong ảnh.

Vì các điểm nhiễu có giá trị cực đại hoặc cực tiểu, nên khi lấy trung vị, chúng không ảnh hưởng mạnh đến kết quả.

2. Giảm nhiễu mà vẫn giữ được biên:

Các bộ lọc tuyến tính (như Mean, Gaussian) thường làm mờ biên, trong khi Median Filter giữ nguyên ranh giới vật thể.

3. Tiền xử lý trong thị giác máy tính:

Trước khi phát hiện biên hoặc trích đặc trưng, ảnh thường được lọc trung vị để loại bỏ nhiễu đơn lẻ mà không làm mất chi tiết.

4. Làm mịn mà không mất cấu trúc chính:

Median Filter chỉ tác động mạnh ở những vị trí nhiễu, giữ nguyên vùng phẳng, nên ảnh sau lọc trông tự nhiên hơn.

3. Nguyên lý hoạt động

3.1. Cơ chế tổng quát

Bộ lọc trung vị sử dụng một mặt nạ (mask hoặc kernel), thường có kích thước 3×3 , 5×5 hoặc 7×7 , trượt qua từng vị trí điểm ảnh trong ảnh gốc.

Tại mỗi vị trí, nó thu thập các giá trị cường độ (intensity) trong vùng mặt nạ, sắp xếp chúng theo thứ tự tăng dần, và chọn giá trị trung vị làm giá trị mới của điểm ảnh trung tâm.

Giả sử vùng lân cận W quanh điểm ảnh (x, y) có N phần tử $\{p_1, p_2, \dots, p_N\}$.

Ta có công thức:

$$g(x, y) = \text{median}\{f(x + s, y + t) \mid (s, t) \in W\}$$

Trong đó:

- $f(x + s, y + t)$: giá trị cường độ của các điểm trong vùng lân cận.
- $g(x, y)$: giá trị mới sau khi lọc.
- W : mặt nạ (neighborhood).
- Median: giá trị ở giữa dãy số khi các phần tử được sắp xếp tăng dần.

3.2. Ví dụ minh họa

Giả sử vùng 3×3 xung quanh một điểm ảnh có giá trị:

$$\begin{bmatrix} 10 & 11 & 12 \\ 255 & 13 & 11 \\ 10 & 12 & 11 \end{bmatrix}$$

Sau khi sắp xếp các giá trị:

$$[10, 10, 11, 11, 11, 12, 12, 13, 255]$$

→ Giá trị trung vị là 11.

Kết quả: điểm trung tâm (255 – bị nhiễu sáng mạnh) được thay bằng 11, ảnh được làm mịn và khử nhiễu hiệu quả.

4. Đặc điểm toán học và tính chất

1. Phi tuyến (Non-linear):

Khác với Mean và Gaussian, Median Filter không thể biểu diễn bằng phép tích chập tuyến tính.

Vì vậy, nó không thỏa mãn tính chất chồng chập (superposition):

$$\text{median}(af_1 + bf_2) \neq a \cdot \text{median}(f_1) + b \cdot \text{median}(f_2)$$

2. Bảo toàn biên (Edge-preserving):

Các giá trị biên thường nằm trong phạm vi cường độ của vùng xung quanh → không bị thay đổi nhiều sau khi lọc.

Do đó Median Filter không làm mờ biên mạnh như bộ lọc tuyến tính.

3. Không gây thay đổi độ sáng tổng thể:

Vì chỉ thay thế giá trị trung tâm bằng một giá trị có sẵn trong vùng → không tạo ra giá trị mới lạ.

4. Độ ổn định cao với nhiễu cực trị:

Các giá trị rất lớn hoặc rất nhỏ (salt & pepper) không ảnh hưởng đáng kể đến kết quả trung vị.

5. Cấu trúc mặt nạ (Mask / Kernel)

Các mặt nạ phổ biến:

5.1. Mặt nạ 3×3 (cơ bản nhất)

Gồm 9 điểm ảnh lân cận → median là phần tử thứ 5 khi sắp xếp tăng dần.

$$W = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix} \Rightarrow g(x, y) = \text{median}(a_1, a_2, \dots, a_9)$$

5.2. Mặt nạ 5×5 hoặc 7×7

Được dùng khi ảnh có nhiễu nhiều hoặc cần làm mịn mạnh hơn, nhưng sẽ tăng chi phí tính toán đáng kể.

6. Ảnh hưởng trong miền tần số

Do Median Filter là phi tuyến, nó không có hàm truyền rõ ràng trong miền tần số (frequency domain) như các bộ lọc tuyến tính.

Tuy nhiên, về mặt trực quan, nó vẫn hoạt động tương tự như một bộ lọc thông thấp (low-pass filter):

- Loại bỏ thành phần tần số cao (nhiều, biến thiên đột ngột).
- Giữ lại thành phần tần số thấp (vùng phẳng, cường độ ổn định).

Nhưng khác với các bộ lọc tuyến tính, Median Filter không gây hiện tượng làm mờ biên (edge blurring), vì biên không bị “pha trộn” bằng phép trung bình.

7. Ảnh hưởng của kích thước mặt nạ

Kích thước mặt nạ	Hiệu ứng lên ảnh
3×3	Làm mịn nhẹ, giữ biên tốt. Phù hợp ảnh nhiễu thấp.
5×5	Giảm nhiễu mạnh hơn nhưng có thể làm mờ chi tiết nhỏ.
7×7	Làm phẳng mạnh, mất cấu trúc nhỏ, dùng cho ảnh nhiễu nặng.

Kích thước mặt nạ càng lớn, mức độ khử nhiễu càng mạnh nhưng đồng thời chi tiết ảnh càng giảm.

8. Các biến thể của bộ lọc trung vị

Bộ lọc trung vị có nhiều cải tiến nhằm tăng hiệu quả trong các trường hợp đặc biệt:

8.1. Adaptive Median Filter (Bộ lọc trung vị thích nghi)

- Kích thước mặt nạ thay đổi tùy theo mức độ nhiễu trong vùng ảnh.
- Nếu vùng có ít nhiễu → dùng mặt nạ nhỏ.
- Nếu vùng nhiễu nặng → tự động mở rộng mặt nạ.
→ Giữ biên tốt hơn và tránh làm mờ không cần thiết.

8.2. Weighted Median Filter (Trung vị có trọng số)

- Các điểm gần tâm được lặp lại nhiều lần trong dãy sắp xếp → tạo hiệu ứng “trọng số”.
- Kết hợp ưu điểm của Gaussian và Median.

8.3. Rank-order Filter

- Median là trường hợp đặc biệt của Rank-order Filters, trong đó phần tử chọn không nhất thiết là trung vị, có thể là phần tử thứ k bất kỳ.
- Dùng trong các ứng dụng đặc biệt cần loại bỏ nhiễu chọn lọc.

9. Ưu điểm và hạn chế (theoretical aspects)

Ưu điểm:

- Loại bỏ hiệu quả nhiễu muối tiêu: vì giá trị nhiễu nằm ngoài dải trung tâm → không ảnh hưởng đến trung vị.
- Giữ biên tốt: các vùng chuyển sắc rõ ràng ít bị mờ hơn so với Mean hay Gaussian.
- Không làm thay đổi cường độ trung bình toàn ảnh.
- Không tạo giá trị mới: giá trị đầu ra luôn nằm trong tập giá trị gốc → tránh xuất hiện điểm giả.

Hạn chế:

- Tốn thời gian tính toán hơn bộ lọc tuyến tính, vì cần sắp xếp các giá trị trong vùng lân cận.
- Không hiệu quả với nhiễu Gaussian liên tục.
- Mất chi tiết nhỏ nếu mặt nạ quá lớn.
- Khó phân tích trong miền tần số do tính phi tuyến.

10. Ảnh hưởng đến cấu trúc và biên ảnh

- **Trong vùng phẳng:** trung vị gần bằng giá trị gốc → ảnh hầu như không thay đổi.
- **Tại biên:** trung vị vẫn phản ánh chính xác ranh giới → biên không bị “pha loãng” như Mean Filter.
- **Tại điểm nhiễu:** giá trị cực trị bị loại bỏ → ảnh được làm sạch hiệu quả.

Do đó, Median Filter vừa giữ cấu trúc chính, vừa loại bỏ các điểm sai lệch cục bộ.

11. Ứng dụng thực tế

- Khử nhiễu muối tiêu trong ảnh xám hoặc màu.
- Tiền xử lý trước phát hiện biên (Canny, Sobel).
- Xử lý ảnh y tế (CT, MRI, X-ray): loại bỏ đốm nhiễu mà không làm mờ mô.
- Xử lý ảnh vệ tinh, ảnh radar: giảm nhiễu đột biến do phản xạ mạnh.
- Phục hồi ảnh cũ: làm sạch các điểm đen trắng bị hỏng trên ảnh quét.

12. Vị trí trong pipeline xử lý ảnh

Median Filter thường được áp dụng ở bước làm mịn (smoothing stage) – đặc biệt trong các bài toán mà nhiễu có dạng “impulse noise”.

Pipeline tiêu biểu:

Ảnh gốc → Khử nhiễu bằng Median Filter → Phát hiện biên → Phân vùng / Nhận dạng.

Trong nhiều thuật toán phát hiện biên như Canny Edge Detector, bộ lọc Gaussian và Median có thể thay thế cho nhau tùy loại nhiễu.

13. Kết luận

Bộ lọc trung vị (Median Filter) là bộ lọc làm mịn phi tuyến hiệu quả nhất trong việc loại bỏ nhiễu muối tiêu và nhiễu đột biến, mà vẫn giữ nguyên các đường biên và chi tiết lớn của ảnh.

Nhờ nguyên lý lấy giá trị trung vị thay vì trung bình, Median Filter không bị ảnh hưởng bởi các điểm ngoại lai (outliers), nên ảnh sau lọc có chất lượng tự nhiên, mịn mà vẫn rõ ràng.

Mặc dù tốn nhiều chi phí tính toán hơn các bộ lọc tuyến tính, Median Filter vẫn được xem là chuẩn mực cho khử nhiễu đột biến (impulse noise removal) trong các ứng dụng xử lý ảnh hiện đại.

14.Code minh họa bộ lọc Median Filter

```
def median_filter(gray: np.ndarray, ksize: int = 3) -> np.ndarray:
    if gray.ndim != 2:
        raise ValueError("median_filter chỉ xử lý ảnh xám 2D (H, W).")

    # đảm bảo kernel là số lẻ >=3
    if ksize % 2 == 0:
        ksize += 1
    if ksize < 3:
        ksize = 3

    g = gray.astype(np.float32)
    H, W = g.shape

    pad = ksize // 2

    # padding replicate để không bị viền đen
    padded = np.pad(g, pad_width=pad, mode="edge")

    out = np.zeros_like(g, dtype=np.float32)

    for i in range(H):
        for j in range(W):
            region = padded[i:i+ksize, j:j+ksize]
            out[i, j] = np.median(region)

    return out
```

15.Ảnh minh họa bộ lọc Median Filter

IV.LÝ THUYẾT VỀ BỘ LỌC GAUSSIAN (GAUSSIAN FILTER)

1. Khái niệm chung

Bộ lọc Gaussian (Gaussian Filter) là một loại bộ lọc làm mịn tuyến tính (linear smoothing filter) sử dụng hàm phân phối chuẩn (Gaussian function) để tính trọng số cho các điểm ảnh lân cận.

Nói cách khác, thay vì tính trung bình đều như Mean Filter, bộ lọc Gaussian ưu tiên trọng số lớn hơn cho các điểm ảnh gần trung tâm, và trọng số nhỏ dần cho các điểm ở xa.

Điều này giúp quá trình làm mịn trở nên tự nhiên, mượt và ít làm mờ biên hơn so với Mean Filter.

Bộ lọc Gaussian được coi là chuẩn mực (standard) trong hầu hết các bài toán khử nhiễu (denoising) và tiền xử lý (preprocessing) trong xử lý ảnh, thị giác máy tính, và học sâu.

2. Mục đích và vai trò

Mục tiêu chính của bộ lọc Gaussian là giảm nhiễu và làm trơn ảnh, đồng thời vẫn duy trì biên và cấu trúc hình học chính.

Các ứng dụng phổ biến gồm:

- 1. Khử nhiễu ngẫu nhiên (Random noise removal):**
Đặc biệt hiệu quả với nhiễu Gaussian — loại nhiễu phổ biến trong tín hiệu cảm biến.
- 2. Tiền xử lý cho phát hiện biên (Edge Detection):**
Làm mịn ảnh trước khi tính gradient (trong các thuật toán Sobel, Canny, LoG...).
- 3. Tiền xử lý trong học sâu (Deep Learning):**
Dùng để ổn định đặc trưng đầu vào, giảm nhiễu trong ảnh huấn luyện.
- 4. Làm trơn hình ảnh hiển thị (Smoothing):**
Tạo hiệu ứng ảnh mờ (blur effect) trong đồ họa, xử lý video, ảnh chân dung.

3. Nguyên lý hoạt động

3.1. Cơ sở toán học

Bộ lọc Gaussian được định nghĩa dựa trên hàm phân phối chuẩn (Gaussian Distribution) trong không gian 2 chiều:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Trong đó:

- (x, y) : tọa độ điểm trong mặt nạ (tính từ tâm).
- σ : độ lệch chuẩn (standard deviation), điều khiển độ mờ của bộ lọc.
- $G(x, y)$: giá trị trọng số của điểm ảnh tại vị trí (x, y) .

Trọng số lớn nhất ở tâm mặt nạ ($x = 0, y = 0$) và giảm dần theo khoảng cách ra biên, đúng theo quy luật chuông (bell curve) của phân bố Gaussian.

3.2. Nguyên lý tích chập

Ảnh đầu ra $g(x, y)$ thu được bằng phép tích chập (convolution) giữa ảnh gốc $f(x, y)$ và mặt nạ Gaussian $G(x, y)$:

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

Trong đó:

- *: ký hiệu phép tích chập.
- a, b : bán kính của mặt nạ (kích thước $= (2a+1) \times (2b+1)$).

3.3. Chuẩn hóa trọng số

Tổng tất cả giá trị trọng số trong mặt nạ được chuẩn hóa sao cho bằng 1:

$$\sum_{x,y} G(x,y) = 1$$

Điều này đảm bảo độ sáng trung bình của ảnh không thay đổi sau khi lọc.

4. Cấu trúc mặt nạ (kernel)

Mặt nạ Gaussian có dạng đối xứng tâm và giá trị giảm dần khi ra xa tâm.

Một ví dụ mặt nạ 5×5 khi $\sigma = 1.0$:

$$\frac{1}{273} \begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

- Trọng số trung tâm (41) lớn nhất.
- Các phần tử xa tâm có giá trị nhỏ hơn theo phân bố Gaussian.
- Tổng tất cả phần tử $= 273$ (được chuẩn hóa về 1 khi chia).

5. Biểu diễn trong miền tần số

Trong miền tần số, bộ lọc Gaussian tương ứng với bộ lọc thông thấp (Low-pass filter) có hàm truyền cũng là một hàm Gaussian.

Hàm truyền $H(u, v)$ được cho bởi:

$$H(u, v) = e^{-\frac{u^2+v^2}{2\sigma_f^2}}$$

Trong đó:

- (u, v) : tọa độ tần số trong miền Fourier.
- σ_f : tham số điều khiển độ rộng của bộ lọc trong miền tần số.

Vì Fourier của Gaussian cũng là Gaussian, nên bộ lọc Gaussian không tạo ra hiện tượng “ripple” (gợn sóng tần số) như các bộ lọc thông thấp lý tưởng khác.

Đây là lý do Gaussian Filter được ưa chuộng trong xử lý ảnh tần số — làm mịn mượt, không sinh nhiễu phụ.

6. Ảnh hưởng của tham số σ (độ lệch chuẩn)

Tham số σ quyết định mức độ làm mờ (blurring level):

Giá trị σ	Đặc điểm ảnh sau lọc
Nhỏ ($\sigma \leq 1$)	Làm mịn nhẹ, giữ được nhiều chi tiết, thích hợp ảnh ít nhiễu.
Trung bình ($\sigma \approx 1.5-2$)	Làm mịn vừa phải, giảm nhiễu Gaussian hiệu quả, vẫn giữ biên tương đối rõ.
Lớn ($\sigma \geq 3$)	Làm mịn mạnh, biên và chi tiết nhỏ bị xóa mờ, ảnh trông “phẳng” hơn.

Thông thường, kích thước hạt nhân (kernel size) được chọn theo quy tắc:

$$\text{Kernel size} \approx 6\sigma + 1$$

(đảm bảo vùng ảnh hưởng của hàm Gaussian đủ bao phủ 99% năng lượng tín hiệu).

7. Đặc điểm và tính chất

1. Tuyến tính và đối xứng:

Bộ lọc Gaussian là bộ lọc tuyến tính và đối xứng theo tâm, nghĩa là ảnh không bị méo hướng.

2. Không gây dịch chuyển pha (Phase shift):

Do tính đối xứng, bộ lọc Gaussian không làm lệch vị trí của các biên hoặc đối tượng trong ảnh.

3. Giữ biên tốt hơn Mean Filter:

Nhờ phân bố trọng số giảm dần, vùng gần trung tâm ảnh hưởng mạnh hơn vùng xa \rightarrow biên bị mờ ít hơn.

4. Tính chất phân rã liên tục:

Hàm Gaussian giảm dần theo khoảng cách, nên hiệu ứng làm mịn diễn ra “mượt” chứ không gắt.

5. Dễ tách thành tích chập 1D:

Bộ lọc Gaussian 2D có thể được tách thành tích của hai bộ lọc 1D:

$$G(x, y) = G(x) \cdot G(y)$$

\rightarrow Giúp giảm đáng kể chi phí tính toán (từ $O(n^2)$ xuống $O(2n)$).

8. Ảnh hưởng đối với cấu trúc ảnh

- Làm giảm nhiễu:**

Loại bỏ nhiễu Gaussian và nhiễu cảm biến nhẹ, đồng thời làm trơn cường độ vùng đồng nhất.

- Giảm độ tương phản biên:**

Biên được làm mềm, không còn gắt như trước, giúp chuẩn bị cho các thuật toán phát hiện biên hoạt động ổn định.

- **Làm mờ chi tiết nhỏ:**
Các hoa văn mảnh, chữ nhỏ, hoặc cấu trúc có kích thước $< \sigma$ sẽ bị mất đi.
- **Không gây méo hình:**
Vì Gaussian là hàm đối xứng, ảnh sau lọc không bị dịch chuyển hay méo hướng.

9. Ứng dụng trong xử lý ảnh

1. **Tiền xử lý phát hiện biên:**
Bộ lọc Gaussian được sử dụng trong các thuật toán phát hiện biên hiện đại như Canny Edge Detector hoặc Laplacian of Gaussian (LoG) để khử nhiễu trước khi tính gradient.
2. **Làm mịn ảnh trong học sâu:**
Dùng để tiền xử lý dữ liệu hình ảnh, giảm nhiễu và tăng khả năng tổng quát của mô hình CNN.
3. **Hiệu ứng hình ảnh (Image Blur):**
Tạo hiệu ứng “Gaussian Blur” trong các phần mềm đồ họa như Photoshop, GIMP, OpenCV.
4. **Phân tích ảnh y tế và ảnh vệ tinh:**
Giúp làm nổi bật cấu trúc lớn, giảm tác động của nhiễu và chi tiết nhỏ không cần thiết.

10. Mối quan hệ với Mean Filter

Về bản chất, Gaussian Filter là phiên bản cải tiến của Mean Filter.

Trong khi Mean Filter coi tất cả điểm ảnh trong mặt nạ đều đóng góp như nhau, Gaussian Filter lại ưu tiên những điểm gần trung tâm hơn.

Điều này giúp Gaussian Filter:

- Giữ biên và cấu trúc ảnh tốt hơn.
- Giảm nhiễu mà không gây hiệu ứng “mờ toàn cục” như Mean Filter.
- Tạo ra hiệu ứng làm mịn tự nhiên, mềm mại.

11. Tính chất đặc biệt trong miền tần số

- Hàm Gaussian trong miền không gian có biến đổi Fourier cũng là một hàm Gaussian trong miền tần số.
→ Điều này đảm bảo quá trình lọc không sinh ra nhiễu phụ (aliasing, ringing).
- Vì vậy, Gaussian Filter được xem là bộ lọc thông thấp tối ưu (Optimal Low-pass Filter) về cả tính toán và hiệu ứng hình ảnh.

12. Kết luận

Bộ lọc Gaussian (Gaussian Filter) là một công cụ làm mịn mạnh mẽ, hiệu quả, và tự nhiên nhất trong xử lý ảnh tuyến tính.

Với cơ sở toán học dựa trên phân bố chuẩn, bộ lọc này:

- Giảm nhiễu hiệu quả,
- Giữ biên mềm mại,
- Không gây méo pha,
- Và đảm bảo tính ổn định cao trong miền tần số.

Gaussian Filter được coi là bộ lọc chuẩn mực (standard smoothing operator) trong các hệ thống xử lý ảnh hiện đại, từ Canny Edge Detection, LoG, đến CNN preprocessing.

13.Code minh họa bộ lọc Gaussian Filter

```
def gaussian_filter(gray: np.ndarray, ksize: int = 3, sigma: float = 1.0) -> np.ndarray:
    if gray.ndim != 2:
        raise ValueError("gaussian_filter chỉ xử lý ảnh xám 2D (H, W).")
    # đảm bảo ksize là số lẻ >= 3
    if ksize % 2 == 0:
        ksize += 1
    if ksize < 3:
        ksize = 3

    g = gray.astype(np.float32)
    H, W = g.shape

    k = ksize // 2
    x = np.arange(-k, k + 1, 1, dtype=np.float32)
    y = np.arange(-k, k + 1, 1, dtype=np.float32)
    X, Y = np.meshgrid(x, y)

    kernel = np.exp(-(X**2 + Y**2) / (2 * sigma**2))
    kernel /= np.sum(kernel) # chuẩn hóa tổng = 1

    padded = np.pad(g, pad_width=k, mode="edge")
    out = np.zeros_like(g, dtype=np.float32)

    for i in range(H):
        for j in range(W):
            region = padded[i:i+ksize, j:j+ksize]
            out[i, j] = np.sum(region * kernel)

    return out
```

14. Ảnh minh họa thực tế bộ lọc Gaussian Filter



V.LÝ THUYẾT CHUNG VỀ BỘ LỌC PHÁT HIỆN BIÊN (EDGE DETECTION FILTER)

1. Khái niệm chung

Bộ lọc phát hiện biên (Edge Detection Filter) là kỹ thuật xử lý ảnh dùng để xác định và làm nổi bật ranh giới (boundary) giữa các vùng có sự khác biệt rõ ràng về cường độ sáng (intensity) hoặc màu sắc (color) trong ảnh.

Trong xử lý ảnh số, biên chính là khu vực mà giá trị cường độ thay đổi mạnh nhất, thể hiện sự chuyển tiếp từ vùng sáng sang vùng tối hoặc giữa hai vật thể khác nhau.

Nói cách khác, biên là đường ngăn cách giữa các đối tượng hoặc các đặc trưng hình học quan trọng trong ảnh, và việc phát hiện đúng các biên là bước cơ sở cho nhiều quy trình xử lý ảnh cao hơn như phân vùng (segmentation), nhận dạng vật thể (recognition), theo dõi chuyển động (tracking), và trích đặc trưng (feature extraction).

Bộ lọc phát hiện biên có thể được xem như “đôi mắt” của hệ thống xử lý ảnh, giúp máy tính hiểu được cấu trúc, hình dạng và ranh giới của thế giới thực được biểu diễn trong ảnh.

2. Mục đích và vai trò của phát hiện biên

Việc phát hiện biên có ý nghĩa cực kỳ quan trọng trong cả xử lý ảnh cơ bản lẫn thị giác máy tính hiện đại.

Các mục tiêu chính của quá trình này bao gồm:

1. **Xác định ranh giới vật thể:**

Giúp tách biệt các đối tượng trong ảnh, phân định vùng đối tượng và vùng nền. Biên là thông tin quan trọng nhất để xác định hình dạng (shape) và kích thước (size).

2. **Giảm khối lượng dữ liệu cần xử lý:**

Sau khi phát hiện biên, thay vì xử lý toàn bộ điểm ảnh, hệ thống chỉ cần tập trung vào các đường biên — tức là những khu vực mang thông tin quan trọng nhất.

Điều này giúp giảm dữ liệu và tăng tốc độ xử lý mà vẫn giữ nguyên đặc trưng cần thiết.

3. **Trích đặc trưng hình dạng:**

Các thuật toán nhận dạng (recognition) hoặc phân loại (classification) thường dựa vào đường biên để xác định hình học và cấu trúc của vật thể.

4. **Tiền xử lý cho các bước nâng cao:**

- **Phân vùng ảnh (Segmentation):** dựa vào biên để chia ảnh thành các vùng riêng biệt.
- **Nhận dạng đối tượng (Object recognition):** xác định vị trí và loại vật thể.
- **Theo dõi chuyển động (Tracking):** phát hiện đường biên chuyển động qua thời gian.
- **Phục hồi 3D (3D reconstruction):** tái tạo mô hình không gian từ thông tin biên.

3. **Nguyên lý hoạt động chung**

3.1. **Bản chất vật lý và toán học**

Trong ảnh số, mỗi điểm ảnh (pixel) có giá trị cường độ sáng (intensity).

Khi ta di chuyển qua vùng ảnh, nếu giá trị này thay đổi chậm và liên tục, đó là vùng phẳng (flat area); ngược lại, nếu thay đổi nhanh và đột ngột, đó là vùng biên (edge).

Sự thay đổi này được đo bằng đạo hàm (derivative) của ảnh.

Vì ảnh là tín hiệu hai chiều $f(x, y)$, nên ta xét các đạo hàm theo hai hướng x (ngang) và y (dọc).

- Nếu đạo hàm của ảnh nhỏ \rightarrow không có biên (ảnh phẳng).
- Nếu đạo hàm lớn \rightarrow có biên (cường độ thay đổi mạnh).

3.2. **Công thức gradient 2D**

Gradient của ảnh tại điểm (x, y) được định nghĩa là:

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

Độ lớn (magnitude) và **hướng (direction)** của gradient là:

$$|\nabla f(x, y)| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

$$\theta(x, y) = \tan^{-1}\left(\frac{\partial f / \partial y}{\partial f / \partial x}\right)$$

- $|\nabla f|$: cho biết mức độ thay đổi cường độ tại vị trí (x, y).
- θ : chỉ hướng của biên (vuông góc với ranh giới vật thể).

Tại vùng phẳng, $|\nabla f| \approx 0$; tại biên, $|\nabla f|$ đạt cực đại.

4. Cơ sở toán học của phát hiện biên

4.1. Đạo hàm bậc nhất (First Derivative)

Dùng để xác định vùng có độ dốc lớn nhất trong ảnh.

Biên được coi là vị trí mà gradient đạt cực đại.

Đây là cơ sở của các bộ lọc như Sobel, Prewitt, Roberts, v.v.

4.2. Đạo hàm bậc hai (Second Derivative)

Dùng để xác định vị trí mà đạo hàm bậc nhất thay đổi dấu (zero-crossing).

Về thực quan, đạo hàm bậc hai đạt cực đại hoặc cực tiểu tại biên.

Đây là nền tảng của các bộ lọc như Laplacian hoặc Laplacian of Gaussian (LoG).

4.3. Biểu thức đạo hàm bậc hai (Laplacian)

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Điểm có $\nabla^2 f = 0$ hoặc chuyển dấu được xem là biên ảnh.

5. Cấu trúc mặt nạ (Mask / Kernel)

Giống với bộ lọc làm mịn, bộ lọc phát hiện biên cũng sử dụng mặt nạ tích chập (convolution mask), thường có kích thước 3×3 hoặc 5×5 .

Tuy nhiên, thay vì các hệ số đều dương như trong bộ lọc làm mịn, mặt nạ biên có giá trị dương, âm và 0 để đo độ chênh lệch cường độ giữa các hướng khác nhau.

Ví dụ:

- Mặt nạ theo hướng ngang: phát hiện biên dọc.

- Mặt nạ theo hướng dọc: phát hiện biên ngang.
- Mặt nạ kết hợp hai hướng: phát hiện biên chéo.

6. Nguyên lý hoạt động tổng quát của quá trình phát hiện biên

Quy trình phát hiện biên thường gồm **bốn bước chính**:

1. **Làm mịn ảnh (Smoothing):**
Ảnh đầu vào thường chứa nhiễu → nếu không làm mịn, nhiễu cũng được coi là biên. Do đó, cần lọc bằng Gaussian hoặc Median để giảm nhiễu.
2. **Tính đạo hàm (Gradient hoặc Laplacian):**
Sử dụng mặt nạ để tính xấp xỉ đạo hàm bậc 1 hoặc bậc 2.
3. **Xác định biên mạnh và yếu:**
Sử dụng ngưỡng (thresholding) để chọn những vùng có biến thiên cường độ lớn nhất.
4. **Làm mảnh và nối biên:**
Giảm độ dày của đường biên, loại bỏ các biên giả và nối liền các biên bị đứt.

Kết quả cuối cùng là một ảnh chứa các đường biên rõ nét, mảnh, và liên tục, thể hiện ranh giới chính xác của vật thể.

7. Ảnh hưởng và đặc điểm của phát hiện biên

7.1. Ảnh hưởng tích cực

- Tăng cường chi tiết quan trọng: nhấn mạnh vùng biên, giúp nhận dạng dễ hơn.
- Giảm dữ liệu: chỉ giữ lại phần cấu trúc cần thiết.
- Xác định hình dạng, vị trí, kích thước của vật thể.
- Tạo dữ liệu đặc trưng (feature maps) phục vụ cho học máy và deep learning.

7.2. Ảnh hưởng phụ và hạn chế

- Nhạy với nhiễu: nếu ảnh gốc nhiễu nhiều, biên giả dễ xuất hiện.
- Mất chi tiết nhỏ nếu làm mịn quá mạnh.
- Phụ thuộc vào ngưỡng: ngưỡng sai có thể bỏ sót hoặc phát hiện sai biên.
- Khó phát hiện biên trong vùng chiếu sáng yếu hoặc ảnh có độ tương phản thấp.

8. Phát hiện biên trong miền tần số

8.1. Quan hệ giữa miền không gian và miền tần số

- Biên tương ứng với thành phần tần số cao trong ảnh, vì nó biểu diễn sự thay đổi nhanh của cường độ sáng.
- Do đó, phát hiện biên tương đương với việc áp dụng bộ lọc thông cao (High-pass filter) trong miền tần số.

8.2. Biểu thức tổng quát trong miền tần số

$$G(u, v) = H(u, v) \cdot F(u, v)$$

Trong đó:

- $F(u, v)$: Biến đổi Fourier của ảnh gốc
- $H(u, v)$: Hàm truyền của bộ lọc phát hiện biên
- $G(u, v)$: Ảnh sau khi lọc

Các thành phần tần số cao được giữ lại, còn các thành phần tần số thấp bị giảm \rightarrow chỉ còn lại các đường biên thể hiện vùng chuyển tiếp mạnh.

9. Đặc điểm và vai trò trong chuỗi xử lý ảnh

Phát hiện biên thường không hoạt động độc lập mà nằm trong chuỗi xử lý ảnh nhiều giai đoạn.

Cụ thể:

Giai đoạn	Mục tiêu	Bộ lọc hoặc kỹ thuật
1. Làm mịn (Smoothing)	Giảm nhiễu, làm phẳng vùng đồng nhất	Mean, Gaussian, Median
2. Phát hiện biên (Edge Detection)	Xác định vùng biến thiên mạnh về cường độ	Sobel, Laplacian, Canny
3. Phân vùng (Segmentation)	Chia ảnh thành các vùng độc lập	Thresholding, Region Growing
4. Trích đặc trưng (Feature Extraction)	Mô tả biên, hình dạng, cấu trúc	Contour, HOG, SIFT, v.v.

Như vậy, làm mịn là bước tiền xử lý bắt buộc, còn phát hiện biên là bước trích thông tin cấu trúc quan trọng nhất.

10. Ứng dụng thực tế của phát hiện biên

- Phân tích hình dạng vật thể (shape analysis) trong công nghiệp, robot, hoặc CAD.
- Xử lý ảnh y tế: phát hiện ranh giới mô, tế bào, xương, khối u (MRI, CT, X-ray).
- Nhận dạng khuôn mặt và vật thể: xác định đường viền mắt, mũi, miệng, viền vật thể.
- Hệ thống giám sát an ninh: phát hiện chuyển động, vật thể trong khung hình.
- Thị giác máy tính (Computer Vision): tiền xử lý cho học sâu (CNN, YOLO, v.v.).
- Xe tự hành (Autonomous Driving): xác định làn đường, biển báo, vật cản.
- Ảnh vệ tinh và nông nghiệp số: phân tách khu vực đất, nước, cây trồng.

11. Ảnh hưởng và các yếu tố quyết định chất lượng phát hiện biên

Yếu tố	Ảnh hưởng
Nhiều (Noise)	Làm xuất hiện biên giả → cần làm mịn trước
Độ tương phản (Contrast)	Biên yếu trong ảnh mờ, ánh sáng kém
Kích thước mặt nạ (Kernel size)	Lớn → biên mượt nhưng mờ; nhỏ → chi tiết nhưng nhạy nhiễu
Ngưỡng (Threshold)	Chọn ngưỡng hợp lý giúp lọc biên yếu nhưng thật
Phương hướng (Orientation)	Một số bộ lọc chỉ nhạy theo hướng nhất định (x, y, chéo)

12. Kết luận

Bộ lọc phát hiện biên (Edge Detection Filter) là một công cụ nền tảng trong xử lý ảnh số, dùng để xác định ranh giới giữa các vùng có khác biệt về cường độ sáng. Bộ lọc này giúp làm nổi bật các chi tiết quan trọng, giảm dữ liệu cần xử lý, và tạo tiền đề cho các bước phân tích, nhận dạng, và học máy.

Phát hiện biên không chỉ là bước cơ bản trong pipeline xử lý ảnh, mà còn là cầu nối giữa xử lý ảnh mức thấp (low-level) như lọc nhiễu và xử lý mức cao (high-level) như nhận dạng hoặc trích xuất đặc trưng.

Tuy nhiên, để đạt hiệu quả cao, quá trình phát hiện biên cần:

- Làm mịn trước để khử nhiễu,
- Chọn đạo hàm phù hợp (bậc 1 hoặc bậc 2),
- Áp dụng ngưỡng chính xác,
- Và kết hợp bước làm mảnh biên (non-maximum suppression) để cho ra ảnh biên rõ ràng, sắc nét và đúng thực tế..

VI. LÝ THUYẾT VỀ BỘ LỌC LAPLACIAN

1. Khái niệm chung

Bộ lọc Laplacian (Laplacian Operator) là một bộ lọc phát hiện biên tuyến tính sử dụng đạo hàm bậc hai (second-order derivative) để xác định các vùng có sự thay đổi đột ngột về cường độ sáng (intensity) trong ảnh.

Khác với các bộ lọc đạo hàm bậc nhất (như Sobel, Prewitt), Laplacian không tính gradient theo hướng cụ thể, mà thay vào đó phát hiện biên theo tất cả các hướng cùng lúc nhờ vào việc sử dụng toán tử Laplace – một toán tử vô hướng mô tả độ cong (curvature) của hàm cường độ ảnh.

Toán tử Laplacian được sử dụng phổ biến trong:

- Phát hiện biên (Edge Detection)
- Tăng cường biên (Edge Enhancement)
- Tiền xử lý trong phân tích ảnh, segmentation, và thị giác máy tính

2. Mục đích và vai trò

Mục tiêu của bộ lọc Laplacian là xác định vị trí của các biên (edges) trong ảnh bằng cách phát hiện những điểm có độ biến thiên cường độ sáng cực đại hoặc cực tiểu.

Cụ thể, Laplacian giúp:

1. Xác định các biên theo mọi hướng (không định hướng như Sobel).
2. Làm nổi bật các vùng chuyển sắc mạnh, nơi đạo hàm bậc hai thay đổi dấu.
3. Phát hiện ranh giới vật thể chính xác, phục vụ cho segmentation và contour detection.
4. Kết hợp với Gaussian (Laplacian of Gaussian – LoG) để giảm nhiễu và cải thiện kết quả phát hiện biên.

3. Cơ sở toán học

3.1. Toán tử Laplace trong không gian liên tục

Trong toán học, toán tử Laplace của một hàm hai biến $f(x, y)$ được định nghĩa là:

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Nó biểu thị tổng của hai đạo hàm bậc hai theo trục x và y.

→ Nếu $f(x, y)$ là cường độ sáng của ảnh, thì Laplacian đo tốc độ thay đổi của gradient, tức là mức độ “cong” của mặt sáng.

Tại biên, gradient đạt cực đại hoặc cực tiểu → đạo hàm bậc hai thay đổi dấu → Laplacian có giá trị \pm lớn nhất.

3.2. Xấp xỉ rời rạc (Discrete Laplacian)

Trong ảnh số, đạo hàm được xấp xỉ bằng sai phân hữu hạn (finite difference).

Toán tử Laplace rời rạc có thể được viết dưới dạng:

$$\nabla^2 f(x, y) = f(x + 1, y) + f(x - 1, y) + f(x, y + 1) + f(x, y - 1) - 4f(x, y)$$

Từ đây, ta thu được mặt nạ tích chập (convolution mask) tương ứng.

4. Các mặt nạ Laplacian phổ biến

Có nhiều biến thể của Laplacian Operator, tùy cách xấp xỉ đạo hàm.

Các mặt nạ 3×3 thông dụng nhất:

4.1. Dạng cơ bản (4-neighbor)

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

4.2. Dạng mở rộng (8-neighbor)

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

4.3. Dạng âm đảo (negative version)

Một số hệ thống sử dụng giá trị âm:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

hoặc

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

Tùy vào quy ước hiển thị, các mặt nạ có thể đảo dấu để biểu diễn biên sáng trên nền tối hoặc ngược lại.

5. Công thức tính toán

Phép tích chập ảnh $f(x, y)$ với mặt nạ Laplacian L được định nghĩa:

$$g(x, y) = f(x, y) * L$$

Trong đó:

- $f(x, y)$: ảnh gốc (grayscale)
- L : mặt nạ Laplacian
- $*$: phép tích chập

Kết quả $g(x, y)$ cho ra các giá trị:

- Lớn dương hoặc âm tại vị trí biên (do thay đổi cường độ mạnh).
- Gần 0 tại vùng phẳng (không có biên thiên)).

6. Nguyên lý phát hiện biên bằng đạo hàm bậc hai

- Tại biên, đạo hàm bậc nhất (gradient) đạt cực đại hoặc cực tiểu.
- Do đó, đạo hàm bậc hai (Laplacian) bằng 0 và đổi dấu tại đúng vị trí biên.

→ Biên được phát hiện tại các điểm có giá trị Laplacian = 0 (zero-crossing points).

Điều này giúp Laplacian xác định biên rõ và chính xác, nhưng cũng khiến nó rất nhạy với nhiễu, vì nhiễu cũng tạo biến thiên nhỏ trong cường độ.

7. Đặc điểm và tính chất

1. Bộ lọc tuyến tính:

Có thể biểu diễn bằng phép tích chập.

$$L(af_1 + bf_2) = aL(f_1) + bL(f_2)$$

2. Đạo hàm bậc hai:

Phát hiện biên tại vùng thay đổi mạnh của gradient (điểm uốn cong trong đồ thị cường độ).

3. Định hướng độc lập:

Không phụ thuộc vào hướng biên – phát hiện biên theo mọi hướng.

4. Độ nhạy cao với nhiễu:

Do tính chất vi phân bậc hai → phóng đại cả biến thiên nhỏ do nhiễu.

5. Tổng hệ số bằng 0:

$$\sum L(i, j) = 0$$

→ Giữ nguyên độ sáng trung bình, chỉ làm nổi bật thay đổi cục bộ.

8. Biểu diễn trong miền tần số

Trong miền Fourier, Laplacian có hàm truyền:

$$H(u, v) = -4\pi^2(u^2 + v^2)$$

→ Là một bộ lọc thông cao mạnh (high-pass filter) vì nó khuếch đại các thành phần tần số cao (biên, chi tiết), đồng thời triệt tiêu các thành phần tần số thấp (vùng phẳng).

Nhờ vậy, Laplacian thường được dùng để làm sắc nét ảnh (image sharpening) bằng cách cộng ngược ảnh lọc vào ảnh gốc:

$$g'(x, y) = f(x, y) - \lambda \cdot g(x, y)$$

với λ là hệ số tăng cường biên.

9. Ảnh hưởng đến ảnh

1. Vùng phẳng:

Các điểm có biến thiên cường độ nhỏ → giá trị Laplacian ≈ 0 → không thay đổi.

2. Vùng có biên:

Tại biên, cường độ thay đổi nhanh \rightarrow đạo hàm bậc hai lớn \rightarrow biên được làm nổi bật.

3. Vùng nhiễu:

Nhiều nhỏ gây biến thiên mạnh \rightarrow Laplacian phản ứng mạnh \rightarrow xuất hiện biên giả. Do đó, thường phải làm mịn trước bằng Gaussian Filter.

10. Laplacian of Gaussian (LoG)

Để khắc phục nhược điểm nhạy nhiễu, người ta kết hợp Gaussian và Laplacian \rightarrow Laplacian of Gaussian (LoG).

Công thức:

$$LoG(x, y) = \nabla^2 [G(x, y) * f(x, y)] = [\nabla^2 G(x, y)] * f(x, y)$$

Với:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

và

$$\nabla^2 G(x, y) = \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

\rightarrow Tức là:

1. Làm mịn ảnh bằng Gaussian để giảm nhiễu.
2. Sau đó áp dụng Laplacian để phát hiện biên.

LoG cho kết quả phát hiện biên mượt và ổn định hơn hẳn Laplacian thuần túy.

11. Ưu điểm và hạn chế (theoretical)

Ưu điểm:

- Phát hiện biên theo mọi hướng (hướng độc lập).
- Độ chính xác cao ở vị trí biên thực (zero-crossing).
- Đơn giản, tuyến tính, dễ cài đặt.
- Có thể dùng để làm sắc nét (sharpening) hoặc phát hiện biên (edge detection).
- Kết hợp tốt với Gaussian (LoG) \rightarrow tăng độ bền với nhiễu.

Hạn chế:

- Rất nhạy với nhiễu: đạo hàm bậc hai khuếch đại nhiễu mạnh.
- Không cung cấp hướng biên (chỉ xác định vị trí).
- Cần bước làm mịn trước.

- Dễ tạo biên giả nếu ảnh có nhiều chi tiết nhỏ.

12. Ứng dụng thực tế

1. Phát hiện biên (Edge Detection):

Xác định ranh giới đối tượng trong ảnh (các điểm zero-crossing).

2. Tăng cường biên (Edge Enhancement):

Dùng để làm sắc nét ảnh mờ, kết hợp Laplacian với ảnh gốc.

3. Tiềm xử lý segmentation:

Loại bỏ vùng phẳng, giữ lại đường biên quan trọng.

4. Thị giác máy tính và học sâu:

Tạo bản đồ biên phục vụ trích đặc trưng (feature map).

5. Ảnh y tế, ảnh vệ tinh:

Tăng cường ranh giới mô, khối u, địa hình.

6. Các thuật toán nâng cao:

Là thành phần trong các kỹ thuật như LoG, Difference of Gaussian (DoG), Marr–Hildreth edge detector.

13. Kết luận

Bộ lọc Laplacian (Laplacian Operator) là một bộ lọc phát hiện biên tuyến tính bậc hai, dựa trên toán tử Laplace để xác định các vùng có thay đổi đột ngột về cường độ sáng.

Khác với Sobel và Prewitt, Laplacian không phụ thuộc hướng, giúp phát hiện biên ở mọi chiều, nhưng rất nhạy với nhiễu.

Trong thực tế, Laplacian thường kết hợp với Gaussian (LoG) để đạt kết quả mượt, ổn định và chính xác hơn.

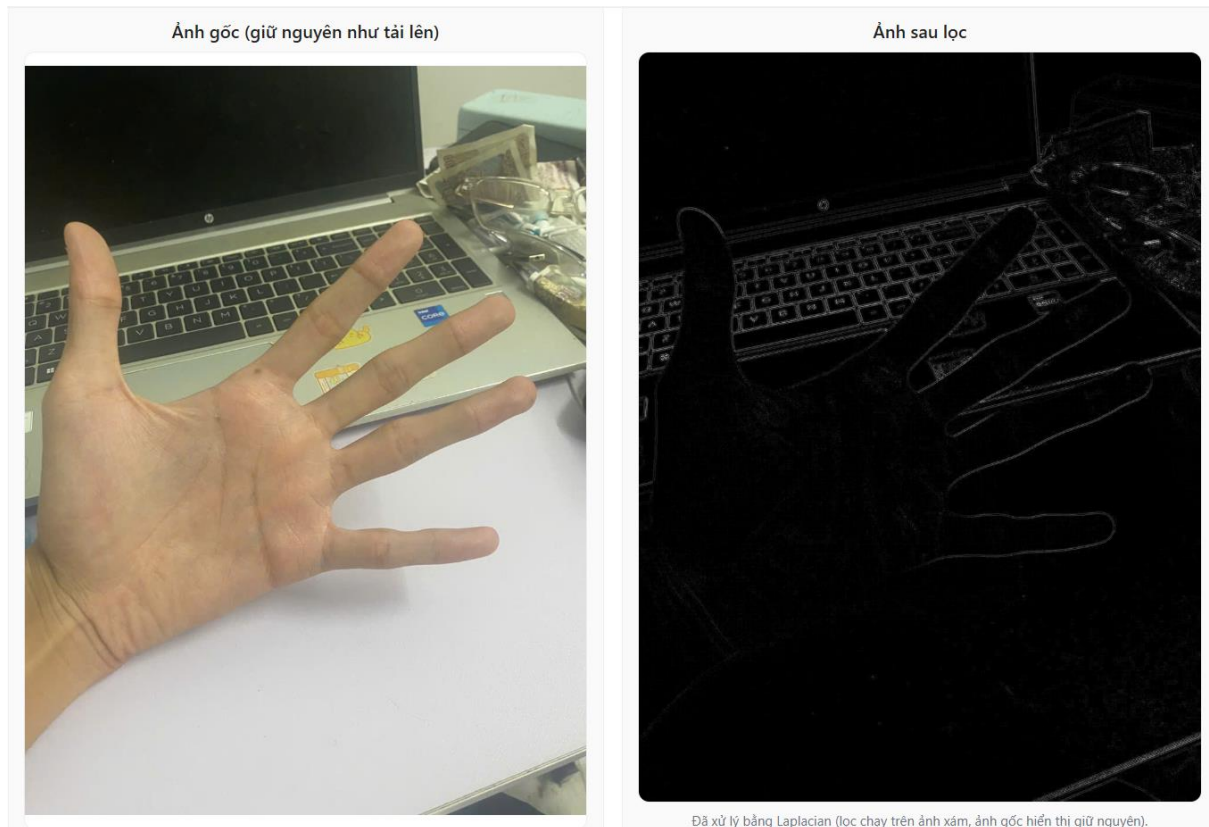
14.Code minh họa bộ lọc Laplacian

```
def laplacian_filter(gray: np.ndarray) -> np.ndarray:
    if gray.ndim != 2:
        raise ValueError("laplacian_filter chỉ xử lý ảnh xám 2D.")
    g = gray.astype(np.float32)
    H, W = g.shape
    gauss_kernel = np.array([
        [1, 2, 1],
        [2, 4, 2],
        [1, 2, 1]
    ], dtype=np.float32)
    gauss_kernel /= gauss_kernel.sum() # tổng = 1
    pad = 1 # cho kernel 3x3
    padded_g = np.pad(g, pad_width=pad, mode="edge")

    blurred = np.zeros_like(g, dtype=np.float32)
    for i in range(H):
        for j in range(W):
            region = padded_g[i:i+3, j:j+3]
            blurred[i, j] = np.sum(region * gauss_kernel)
    lap_kernel = np.array([
        [ 0, -1,  0],
        [-1,  4, -1],
        [ 0, -1,  0]
    ], dtype=np.float32)
    padded_blur = np.pad(blurred, pad_width=pad, mode="edge")
    lap = np.zeros_like(g, dtype=np.float32)
    for i in range(H):
        for j in range(W):
            region = padded_blur[i:i+3, j:j+3]
            lap[i, j] = np.sum(region * lap_kernel)
    lap = np.abs(lap)
    max_val = lap.max()
    if max_val > 0:
        lap = lap / max_val * 255.0

    return lap.astype(np.float32)
```


15. Ảnh minh họa thực tế bộ lọc Laplacian



VII. LÝ THUYẾT VỀ BỘ LỌC PREWITT (PREWITT FILTER / PREWITT OPERATOR)

1. Khái niệm chung

Bộ lọc Prewitt (Prewitt Operator) là một bộ lọc phát hiện biên tuyến tính (Linear Edge Detection Filter) thuộc nhóm bộ lọc đạo hàm bậc nhất (first derivative filters) trong xử lý ảnh số.

Nó được thiết kế để ước lượng gradient (đạo hàm riêng) của cường độ sáng theo hai hướng chính – ngang (x) và dọc (y) – nhằm xác định các vùng có sự thay đổi mạnh về mức xám, tức là các biên (edges) của vật thể trong ảnh.

Bộ lọc Prewitt là phiên bản đơn giản hóa của Sobel Operator:

- Dễ tính toán hơn vì trọng số đều nhau.
- Kết quả ổn định và trực quan trong các ảnh ít nhiễu.
- Thường được dùng trong giáo trình, thực hành và tiền xử lý ảnh cơ bản.

2. Mục đích và vai trò

Mục đích chính của bộ lọc Prewitt là phát hiện biên (edge detection), tức là tìm những điểm trong ảnh mà giá trị cường độ thay đổi nhanh và mạnh.

Cụ thể:

1. Xác định ranh giới vật thể: biên thể hiện đường bao của đối tượng.
2. Phát hiện hướng của biên: giúp xác định biên nằm ngang, dọc hay chéo.
3. Làm nổi bật các thay đổi cường độ sáng: tăng cường các chi tiết quan trọng, giảm vùng phẳng.
4. Chuẩn bị cho các bước phân vùng hoặc nhận dạng: ví dụ như trong object detection, contour extraction, hay feature matching.

3. Nguyên lý hoạt động

3.1. Cơ sở toán học

Biên được xác định bởi độ lớn của vector gradient tại mỗi điểm ảnh.

Gradient của ảnh 2D $f(x, y)$ được tính bằng:

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

Độ lớn của gradient:

$$|\nabla f(x, y)| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Hướng của biên:

$$\theta(x, y) = \tan^{-1}\left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}}\right)$$

Bộ lọc Prewitt xấp xỉ hai đạo hàm riêng phần $\frac{\partial f}{\partial x}$ và $\frac{\partial f}{\partial y}$ bằng hai mặt nạ tích chập (convolution masks) 3×3 .

4. Cấu trúc mặt nạ Prewitt

4.1. Mặt nạ theo hướng X (phát hiện biên dọc)

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

4.2. Mặt nạ theo hướng Y (phát hiện biên ngang)

$$G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Đặc điểm:

- Các hệ số ± 1 đều nhau \rightarrow cấu trúc đơn giản, đối xứng.
- Không có trọng số ưu tiên ở trung tâm như Sobel (trọng số ± 2).
- Mỗi mặt nạ chỉ lấy trung bình chênh lệch giữa hai vùng sáng và tối ở hai phía.

5. Công thức tính toán

Tại mỗi điểm ảnh (x, y) :

$$\begin{aligned} G_x &= (f * P_x)(x, y) \\ G_y &= (f * P_y)(x, y) \end{aligned}$$

Trong đó:

- $f(x, y)$: ảnh đầu vào (thường là ảnh xám).
- $*$: phép tích chập (convolution).
- P_x, P_y : hai mặt nạ Prewitt cho hướng x và y.

Độ lớn của gradient:

$$G = \sqrt{G_x^2 + G_y^2}$$

hoặc trong thực hành:

$$G \approx |G_x| + |G_y|$$

6. Ý nghĩa hướng biên

- G_x : đo sự thay đổi cường độ theo hướng ngang (x) \rightarrow phát hiện biên dọc (vertical edges).
- G_y : đo sự thay đổi cường độ theo hướng dọc (y) \rightarrow phát hiện biên ngang (horizontal edges).
- Kết hợp cả hai để tạo ảnh biên tổng hợp:

$$G(x, y) = \sqrt{G_x^2 + G_y^2}$$

hoặc dùng ảnh hướng:

$$\theta(x, y) = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

7. Đặc điểm và tính chất

1. Bộ lọc tuyến tính (Linear Filter):

Có thể biểu diễn bằng phép tích chập, thỏa mãn tính chất chồng chập.

2. Bộ lọc thông cao (High-pass Filter):

Giữ lại thành phần tần số cao (biên, chi tiết), loại bỏ thành phần tần số thấp (vùng phẳng).

3. Đối xứng và định hướng:

Hai mặt nạ G_x, G_y vuông góc nhau, cho phép phát hiện biên ở cả hai hướng chính.

4. Giá trị trung tâm bằng 0:

→ Chỉ phản ứng với các biến thiên sáng tối, không ảnh hưởng đến vùng phẳng.

5. Không có trọng số trung tâm:

→ Tính toán đơn giản, nhưng dễ nhạy nhiễu hơn Sobel.

8. Ảnh hưởng đối với ảnh

1. Vùng phẳng:

Cường độ gần như không thay đổi → gradient nhỏ → không phát hiện biên.

2. Vùng có thay đổi mạnh:

Gradient lớn → phát hiện là biên.

Càng chênh lệch sáng–tối rõ → biên càng đậm.

3. Vùng nhiễu nhẹ:

Có thể xuất hiện biên giả, vì Prewitt không làm mịn trước.

Thường kết hợp với Gaussian Filter để khử nhiễu.

4. Vùng biên chéo:

Cần kết hợp cả G_x và G_y để phát hiện chính xác.

9. Biểu diễn trong miền tần số

Trong miền Fourier, Prewitt tương đương với bộ lọc vi phân định hướng (directional differentiation filter), làm nổi bật các thành phần tần số cao theo hướng biên.

Công thức biểu diễn gần đúng:

$$H_x(u, v) = j2\pi u \cdot W(u, v)$$

$$H_y(u, v) = j2\pi v \cdot W(u, v)$$

Trong đó $W(u, v)$ là hàm truyền của bộ lọc trung bình (smoothing component).
→ Prewitt vừa vi phân, vừa có tác dụng làm mịn nhẹ (nhờ trung bình các hàng hoặc cột).

10. Quy trình phát hiện biên bằng Prewitt

Quy trình tổng quát khi áp dụng Prewitt Filter:

1. **Làm mịn ảnh (tùy chọn):**
Dùng Mean hoặc Gaussian để giảm nhiễu.
2. **Tích chập với mặt nạ G_x và G_y .**
3. **Tính độ lớn gradient:**

$$G = \sqrt{G_x^2 + G_y^2}$$

4. **Áp dụng ngưỡng (threshold):**
Chỉ giữ lại các điểm có gradient vượt ngưỡng (được coi là biên).
5. **Tạo ảnh biên:**
Các vùng có gradient lớn được tô đen hoặc trắng thể hiện đường biên vật thể.

11. Ưu điểm và hạn chế (về lý thuyết)

Ưu điểm:

- Đơn giản và tính toán nhanh: các hệ số ± 1 dễ thực hiện.
- Dễ triển khai trong phần cứng hoặc vi xử lý.
- Cho kết quả trực quan, rõ ràng trong ảnh ít nhiễu.
- Xác định được hướng biên (ngang, dọc).

Hạn chế:

- Nhạy với nhiễu: vì không có trọng số làm mịn như Sobel.
- Không tốt trong ảnh nhiễu cao hoặc ánh sáng kém.
- Biên yếu có thể bị bỏ qua.
- Không mượt bằng Sobel hoặc Canny.

12. Ứng dụng thực tế

- Phát hiện đường viền (Edge and Contour Detection): xác định ranh giới giữa các vùng trong ảnh.
- Tiền xử lý cho nhận dạng vật thể (Object Recognition).
- Phân tích hình dạng (Shape Analysis).
- Ảnh công nghiệp và robot: nhận dạng vật thể, kiểm tra lỗi sản phẩm.
- Xử lý ảnh y tế: tách mô, xác định đường biên tế bào, khối u.

- Phân vùng ảnh (Image Segmentation): làm nổi bật các vùng sáng–tối khác biệt.

13. Kết luận

Bộ lọc Prewitt (Prewitt Operator) là một trong những bộ lọc phát hiện biên tuyến tính cổ điển và cơ bản nhất trong xử lý ảnh số.

Nó dựa trên xấp xỉ đạo hàm bậc nhất theo hai hướng chính (ngang và dọc) để tìm các điểm có sự thay đổi mạnh về cường độ sáng.

Với cấu trúc đơn giản, dễ cài đặt, và tốc độ xử lý nhanh, Prewitt được sử dụng rộng rãi trong các ứng dụng phát hiện biên cơ bản, phân vùng ảnh, và thị giác máy tính thời gian thực.

Tuy nhiên, do thiếu cơ chế làm mịn nội tại, bộ lọc này nhạy với nhiễu hơn Sobel, và thường được sử dụng trong môi trường có nhiễu thấp hoặc sau bước khử nhiễu.

14. Code minh họa bộ lọc Prewitt

```
def prewitt_filter(gray: np.ndarray) -> np.ndarray:
    if gray.ndim != 2:
        raise ValueError("prewitt_filter chỉ xử lý ảnh xám 2D.")

    g = gray.astype(np.float32)
    H, W = g.shape

    # Kernel Prewitt
    kernel_x = np.array([[-1, 0, 1],
                        [-1, 0, 1],
                        [-1, 0, 1]],
                        dtype=np.float32)

    kernel_y = np.array([[ 1, 1, 1],
                        [ 0, 0, 0],
                        [-1, -1, -1]],
                        dtype=np.float32)

    pad = 1
    padded = np.pad(g, pad_width=pad, mode="edge")

    out = np.zeros_like(g, dtype=np.float32)

    # Tính gradient theo Prewitt
    for i in range(H):
        for j in range(W):
            region = padded[i:i+3, j:j+3]

            gx = np.sum(region * kernel_x)
            gy = np.sum(region * kernel_y)

            out[i, j] = np.sqrt(gx**2 + gy**2)

    return out.astype(np.float32)
```

15. Ảnh minh họa thực tế bộ lọc Prewitt



VI. LÝ THUYẾT VỀ BỘ LỌC SOBEL (SOBEL FILTER / SOBEL OPERATOR)

1. Khái niệm chung

Bộ lọc Sobel (Sobel Operator) là một bộ lọc phát hiện biên tuyến tính (linear edge detection filter) được sử dụng rộng rãi trong xử lý ảnh số.

Nó hoạt động dựa trên việc ước lượng đạo hàm bậc nhất (first derivative) của cường độ sáng (intensity) theo hai hướng — ngang (x) và dọc (y) — để phát hiện các vùng có sự thay đổi mạnh nhất về cường độ, tức là các biên (edges) của ảnh.

Khác với các bộ lọc đạo hàm đơn giản (như Roberts, Prewitt), bộ lọc Sobel kết hợp giữa phép tính gradient và làm mịn bằng trọng số, giúp giảm nhiễu và tạo kết quả ổn định hơn.

Vì vậy, Sobel là một trong những bộ lọc phát hiện biên kinh điển và được dùng nhiều nhất trong thực hành.

2. Mục đích và vai trò

Mục đích của bộ lọc Sobel là:

1. **Phát hiện các biên rõ ràng (edges)** trong ảnh bằng cách tìm ra những vị trí mà cường độ sáng thay đổi đột ngột.
2. **Xác định hướng và độ lớn của biên**, phục vụ cho các bước xử lý sau như nhận dạng, phân vùng, hoặc trích đặc trưng.

3. **Giảm ảnh hưởng của nhiễu**, nhờ việc tính gradient kết hợp với trọng số làm mịn.
4. **Tiền xử lý trong thị giác máy tính**, đặc biệt trong các hệ thống nhận dạng vật thể, khuôn mặt, hoặc phân tích cấu trúc ảnh.

3. Nguyên lý hoạt động

3.1. Bản chất toán học

Biên trong ảnh là vùng có gradient cường độ sáng lớn nhất.

Gradient của ảnh hai chiều $f(x, y)$ được định nghĩa là vector:

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

Độ lớn của gradient:

$$|\nabla f(x, y)| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Và hướng của biên:

$$\theta(x, y) = \tan^{-1}\left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}}\right)$$

Bộ lọc Sobel cung cấp xấp xỉ rời rạc cho hai đạo hàm riêng phần $\frac{\partial f}{\partial x}$ và $\frac{\partial f}{\partial y}$ thông qua hai mặt nạ tích chập (convolution kernels).

4. Cấu trúc mặt nạ Sobel

Bộ lọc Sobel sử dụng hai mặt nạ 3×3 , một cho hướng ngang (x) và một cho hướng dọc (y):

Mặt nạ theo hướng X (phát hiện biên dọc)

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

Mặt nạ theo hướng Y (phát hiện biên ngang)

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

Trong đó:

- Các phần tử ± 2 ở giữa hàng hoặc cột đóng vai trò trọng số, giúp làm mịn ảnh theo hướng vuông góc trước khi tính đạo hàm.
- Trọng số lớn hơn ở trung tâm \rightarrow giảm nhiễu tốt hơn so với các bộ lọc gradient đơn giản khác (như Prewitt).

5. Công thức tính toán

Tại mỗi điểm ảnh (x, y) :

$$G_x = (f * S_x)(x, y)$$

$$G_y = (f * S_y)(x, y)$$

Trong đó:

- $f(x, y)$: ảnh đầu vào.
- $*$: phép tích chập (convolution).
- S_x, S_y : hai mặt nạ Sobel theo hướng x và y.

Sau khi tính được hai ảnh gradient thành phần, độ lớn của gradient được tính bằng:

$$G = \sqrt{G_x^2 + G_y^2}$$

hoặc trong thực tế (để giảm chi phí tính toán):

$$G \approx |G_x| + |G_y|$$

6. Ý nghĩa và hướng của biên

- **Độ lớn gradient G** biểu thị **mức độ biến thiên cường độ sáng** — càng lớn, biên càng rõ.
- **Góc hướng θ** cho biết **hướng của biên** (vuông góc với hướng thay đổi mạnh nhất).
 - Nếu $\theta \approx 0^\circ \rightarrow$ biên dọc (thay đổi theo hướng x).
 - Nếu $\theta \approx 90^\circ \rightarrow$ biên ngang (thay đổi theo hướng y).
 - Nếu θ trung gian \rightarrow biên chéo.

7. Đặc điểm nổi bật

1. Kết hợp đạo hàm và làm mịn:

Bộ lọc Sobel không chỉ tính đạo hàm (gradient), mà còn có trọng số làm mịn theo hướng vuông góc \rightarrow giảm nhiễu, ổn định hơn Prewitt hoặc Roberts.

2. Đơn giản và hiệu quả:

Tính toán nhanh, dễ cài đặt, phù hợp với ảnh thời gian thực (real-time).

3. Giữ hướng biên tốt:

Có thể xác định cả độ lớn và hướng của biên rõ ràng.

4. Nhạy với ánh sáng mạnh:

Dễ phát hiện biên sáng–tối rõ ràng, nhưng có thể bỏ qua biên yếu hoặc mờ.

8. Đặc tính toán học

8.1. Tuyến tính

Sobel là bộ lọc tuyến tính, vì phép tích chập và cộng tuyến tính thỏa mãn:

$$\text{Sobel}(af_1 + bf_2) = a \cdot \text{Sobel}(f_1) + b \cdot \text{Sobel}(f_2)$$

8.2. Bộ lọc thông cao (High-pass Filter)

Vì Sobel đo lường mức thay đổi cường độ sáng (đạo hàm), nên nó loại bỏ các thành phần tần số thấp (vùng phẳng) và giữ lại tần số cao (vùng có biên).

Do đó, Sobel tương đương với bộ lọc thông cao trong miền tần số.

8.3. Tính đối xứng

Hai mặt nạ G_x và G_y là đối xứng nhau qua phép quay 90° , giúp phát hiện biên ở mọi hướng.

9. Biểu diễn trong miền tần số

Trong miền Fourier, Sobel tương đương với một bộ lọc vi phân định hướng (directional differentiation filter), nhấn mạnh các thành phần tần số cao theo hướng vuông góc với biên.

Công thức biểu diễn:

$$H_x(u, v) = j2\pi u \cdot W(u, v)$$

$$H_y(u, v) = j2\pi v \cdot W(u, v)$$

Trong đó $W(u, v)$ là hàm truyền của phép làm mịn tích hợp trong Sobel (trọng số ± 2). \rightarrow Điều này giúp Sobel phát hiện biên “sạch” hơn so với các bộ lọc vi phân đơn thuần (như Roberts).

10. Ảnh hưởng đến ảnh sau lọc

1. **Vùng phẳng:**
Gradient gần bằng 0 \rightarrow điểm không phải biên.
2. **Vùng chuyển sắc mạnh:**
Gradient lớn \rightarrow điểm nằm trên biên.
3. **Vùng nhiễu:**
Các dao động nhỏ tạo gradient giả, nhưng do Sobel có tính làm mịn nên hiện tượng này giảm đáng kể.
4. **Biên chéo:**
Biên chéo được phát hiện bằng cách kết hợp hai hướng G_x và G_y .

11. Quá trình xử lý tổng quát

Quy trình phát hiện biên bằng bộ lọc Sobel gồm các bước:

1. Làm mịn ảnh đầu vào (thường bằng Gaussian Filter) để loại bỏ nhiễu.
2. Tích chập ảnh với mặt nạ Sobel G_x và G_y để tính gradient theo hai hướng.
3. Tính độ lớn và hướng của gradient.
4. Áp dụng ngưỡng (thresholding) để xác định đâu là biên thực.
5. Tạo ảnh biên (edge map) hiển thị các vùng có gradient lớn.

12. Ưu điểm và hạn chế (về lý thuyết)

Ưu điểm:

- Dễ cài đặt, tính toán nhanh.
- Kết hợp tính đạo hàm và làm mịn \rightarrow ổn định, giảm nhiễu.
- Xác định được cả độ lớn và hướng biên.
- Phổ biến, được hỗ trợ sẵn trong hầu hết thư viện xử lý ảnh (OpenCV, MATLAB...).

Hạn chế:

- Nhạy với nhiễu mạnh: vẫn cần làm mịn trước (Gaussian).
- Không phát hiện tốt biên yếu hoặc vùng độ tương phản thấp.
- Độ chính xác kém hơn Canny, nhưng đơn giản hơn nhiều.
- Chỉ phù hợp cho ảnh grayscale (mức xám), không dùng trực tiếp cho ảnh màu (phải tách kênh trước).

13. Ứng dụng thực tế

- Phát hiện ranh giới vật thể (object contour detection).

- Tiền xử lý trong thị giác máy tính: nhận dạng hình dạng, phát hiện cạnh, trích đặc trưng.
- Hệ thống robot và xe tự hành: nhận dạng làn đường, vật cản.
- Ảnh y tế (CT, MRI, X-ray): làm nổi bật ranh giới mô, tế bào.
- Phân vùng ảnh (segmentation): dùng gradient làm tiêu chí chia vùng.

14. Kết luận

Bộ lọc Sobel (Sobel Operator) là một trong những phương pháp phát hiện biên kinh điển và quan trọng nhất trong xử lý ảnh.

Dựa trên xấp xỉ đạo hàm bậc nhất theo hai hướng x và y, Sobel cho phép xác định độ lớn và hướng của biên, đồng thời kết hợp trọng số làm mịn giúp giảm nhiễu hiệu quả.

Dù có độ chính xác không cao bằng các phương pháp hiện đại (như Canny), Sobel vẫn được sử dụng phổ biến nhờ độ đơn giản, tốc độ nhanh, và kết quả trực quan.

15. Code minh họa bộ lọc Sobel

```
def sobel_filter(gray: np.ndarray) -> np.ndarray:
    if gray.ndim != 2:
        raise ValueError("sobel_filter chỉ xử lý ảnh xám 2D (H, W).")

    g = gray.astype(np.float32)
    H, W = g.shape

    # Kernel Sobel
    kernel_x = np.array([[-1, 0, 1],
                        [-2, 0, 2],
                        [-1, 0, 1]],
                        dtype=np.float32)

    kernel_y = np.array([[-1, -2, -1],
                        [ 0,  0,  0],
                        [ 1,  2,  1]],
                        dtype=np.float32)

    pad = 1 # vì kernel 3x3
    padded = np.pad(g, pad_width=pad, mode="edge")

    out = np.zeros_like(g, dtype=np.float32)

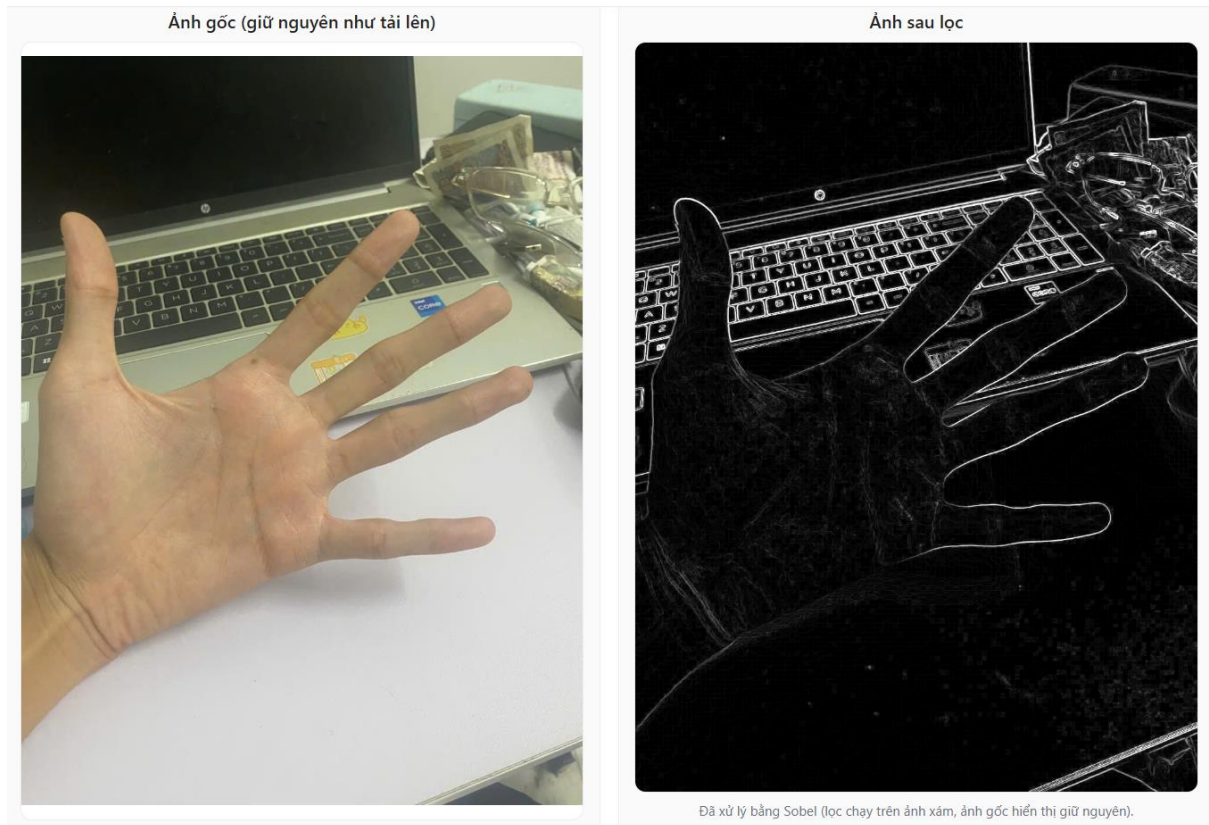
    # Tích chập thủ công Sobel
    for i in range(H):
        for j in range(W):
            region = padded[i:i+3, j:j+3]

            gx = np.sum(region * kernel_x)
            gy = np.sum(region * kernel_y)

            out[i, j] = np.sqrt(gx**2 + gy**2)

    return out
```

16. Ảnh minh họa thực tế bộ lọc Sobel



IX. SO SÁNH CÁC BỘ LỌC TRONG XỬ LÝ ẢNH SỐ

(Mean – Gaussian – Median – Sobel – Prewitt – Laplacian)

1. Phân nhóm chức năng

Nhóm	Mục đích chính	Các bộ lọc tiêu biểu
1. Bộ lọc làm mịn (Smoothing Filters)	Giảm nhiễu, làm trơn ảnh, chuẩn bị cho bước phát hiện biên.	Mean Filter, Gaussian Filter, Median Filter
2. Bộ lọc phát hiện biên (Edge Detection Filters)	Xác định vùng có sự thay đổi mạnh về cường độ sáng, tìm ranh giới vật thể.	Sobel Filter, Prewitt Filter, Laplacian Filter

2. Bảng so sánh tổng quát 6 bộ lọc

Tiêu chí	Mean Filter	Gaussian Filter	Median Filter	Sobel Filter	Prewitt Filter	Laplacian Filter
Loại bộ lọc	Làm mịn tuyến tính	Làm mịn tuyến tính có trọng số Gaussian	Làm mịn phi tuyến	Phát hiện biên tuyến tính	Phát hiện biên tuyến tính (bậc 1)	Phát hiện biên tuyến tính (bậc 2)

				tính (bậc 1)		
Mục đích chính	Giảm nhiễu và làm trơn ảnh	Giảm nhiễu mượt, giữ biên tốt hơn	Loại bỏ nhiễu muối tiêu, giữ biên rõ	Phát hiện biên theo hướng X, Y	Phát hiện biên theo hướng X, Y	Phát hiện biên theo mọi hướng
Nguyên lý	Trung bình cộng giá trị vùng lân cận	Trung bình có trọng số theo phân bố Gaussian	Lấy giá trị trung vị trong vùng lân cận	Tính đạo hàm bậc 1 (gradien t) theo X, Y	Tính đạo hàm bậc 1 theo X, Y (trọng số đều)	Tính đạo hàm bậc 2 tổng hợp theo X, Y
Tuyến tính / Phi tuyến	Tuyến tính	Tuyến tính	Phi tuyến	Tuyến tính	Tuyến tính	Tuyến tính
Giữ biên ảnh	Yếu (biên bị mờ)	Trung bình (biên mềm, ít mờ hơn Mean)	Tốt (ít làm mờ biên)	Tốt (biên rõ theo hướng)	Tốt (biên rõ theo hướng)	Cao (phát hiện mạnh theo mọi hướng)
Khả năng khử nhiễu	Trung bình, hiệu quả với nhiễu Gaussian nhẹ	Tốt, giảm nhiễu tự nhiên	Rất tốt với nhiễu muối tiêu	Trung bình, cần lọc trước	Trung bình, cần lọc trước	Kém, nhạy với nhiễu (phải làm mịn trước)
Mặt nạ đặc trưng (3×3)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	$\frac{1}{16} \begin{bmatrix} 1 & 1 & 2 & 1 \\ 2 & 4 & 2 & 2 \\ 1 & 2 & 1 & 1 \end{bmatrix}$	Không cố định, phụ thuộc dữ liệu	$\begin{bmatrix} -1 & 0 \\ -2 & 0 \\ -1 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$
Hướng phát hiện biên	Không có	Không có	Không có	Hai hướng (ngang & dọc)	Hai hướng (ngang & dọc)	Mọi hướng
Độ nhạy nhiễu	Thấp	Thấp	Rất thấp	Trung bình	Trung bình-cao	Cao
Độ phức tạp tính toán	Thấp (rất nhanh)	Trung bình	Cao hơn do sắp xếp trung vị	Trung bình	Thấp	Trung bình

Ứng dụng chính	Làm mịn ảnh, khử nhiễu nền	Khử nhiễu Gaussian, chuẩn bị phát hiện biên	Xử lý nhiễu muối tiêu, ảnh y tế	Phát hiện biên hướng, nhận dạng vật thể	Phát hiện biên đơn giản, giáo trình	Phát hiện biên tổng hợp, làm sắc nét (Sharpening)
Kết quả đầu ra	Ảnh mờ mịn, mất chi tiết	Ảnh mượt, ít mờ biên	Ảnh sạch nhiễu, biên rõ	Ảnh biên đậm, theo hướng gradient	Ảnh biên rõ, nhưng nhạy nhiễu	Ảnh có biên sáng/tối mạnh, độ tương phản cao
Độ chính xác phát hiện biên	Không áp dụng	Không áp dụng	Không áp dụng	Cao	Trung bình	Rất cao (nếu làm mịn trước)

3. Nhận xét chi tiết theo từng tiêu chí

3.1.Mục đích

- Ba bộ lọc đầu (Mean, Gaussian, Median) là bộ lọc làm mịn (low-pass) → loại bỏ nhiễu và chi tiết tần số cao.
- Ba bộ lọc sau (Sobel, Prewitt, Laplacian) là bộ lọc phát hiện biên (high-pass) → nhấn mạnh thay đổi cường độ sáng.

3.2. Mức độ tuyến tính

- Tuyến tính:** Mean, Gaussian, Sobel, Prewitt, Laplacian.
- Phi tuyến:** Median → không thể mô tả bằng tích chập, nhưng khử nhiễu rất mạnh.

3.3.Độ giữ biên

- Kém:** Mean (làm mờ toàn cục).
- Trung bình:** Gaussian (giảm nhiễu nhẹ, vẫn giữ biên mượt).
- Tốt:** Median (không làm mờ biên).
- Rất tốt:** Sobel, Prewitt (biên rõ ràng).
- Rất cao:** Laplacian (làm nổi biên mạnh theo mọi hướng).

3.4. Độ nhạy nhiễu

- Ít nhạy:** Mean, Gaussian, Median.
- Nhạy trung bình:** Sobel, Prewitt.

- **Rất nhạy:** Laplacian (phải dùng Gaussian trước \rightarrow LoG).

3.5. Tính định hướng

- **Không định hướng:** Mean, Gaussian, Median, Laplacian.
- **Định hướng (có hướng X–Y):** Sobel, Prewitt \rightarrow giúp xác định hướng biên.

3.6. Mức độ phức tạp

- **Đơn giản nhất:** Mean, Prewitt.
- **Vừa phải:** Gaussian, Sobel, Laplacian.
- **Cao hơn:** Median (do phải sắp xếp giá trị vùng lân cận).

3.7. Ứng dụng điển hình

Bộ lọc	Ứng dụng tiêu biểu
Mean Filter	Làm mịn ảnh cơ bản, tiền xử lý cho ảnh nền phẳng
Gaussian Filter	Làm mịn ảnh tự nhiên, khử nhiễu Gaussian, trước phát hiện biên
Median Filter	Khử nhiễu muối tiêu, ảnh y tế, ảnh vệ tinh
Sobel Filter	Phát hiện biên rõ ràng, tính hướng gradient
Prewitt Filter	Phát hiện biên cơ bản, môi trường nhiễu thấp
Laplacian Filter	Phát hiện biên tổng hợp, làm sắc nét (sharpening), LoG

4. Mối quan hệ giữa các bộ lọc

Nhóm Làm Mịn (Low-pass)

- **Mean Filter:** trung bình đều, làm mờ mạnh, dễ mất chi tiết.
- **Gaussian Filter:** trung bình có trọng số \rightarrow mượt tự nhiên, ít mờ biên.
- **Median Filter:** thay giá trị bằng trung vị \rightarrow loại bỏ nhiễu cực trị, giữ biên tốt.

\rightarrow Các bộ lọc này **làm giảm tần số cao**, giúp chuẩn bị cho giai đoạn phát hiện biên.

Nhóm Phát Hiện Biên (High-pass)

- **Sobel Filter:** đạo hàm bậc 1 + làm mịn nhẹ \rightarrow phát hiện biên định hướng X–Y.
- **Prewitt Filter:** đạo hàm bậc 1 đơn giản hơn Sobel, nhanh hơn nhưng nhạy nhiễu.
- **Laplacian Filter:** đạo hàm bậc 2, phát hiện biên mọi hướng, mạnh nhưng cần khử nhiễu trước.

→ Các bộ lọc này **làm nổi bật tần số cao**, thể hiện rõ các vùng biên và ranh giới vật thể.

5. Mối quan hệ theo miền tần số

Nhóm lọc	Loại lọc tần số	Ảnh hưởng
Mean, Gaussian, Median	Low-pass Filter	Loại bỏ nhiễu, làm mờ ảnh, giữ vùng phẳng
Sobel, Prewitt, Laplacian	High-pass Filter	Làm nổi biên, nhấn mạnh chi tiết, loại bỏ nền phẳng

6. Kết luận tổng quát

- **Mean Filter** → đơn giản, làm mịn mạnh, dễ mất chi tiết.
- **Gaussian Filter** → mượt, hiệu quả cao, tiêu chuẩn cho khử nhiễu nhẹ.
- **Median Filter** → mạnh mẽ với nhiễu muối tiêu, giữ biên tốt nhất trong nhóm làm mịn.
- **Sobel Filter** → phát hiện biên theo hướng X–Y, ổn định, giảm nhiễu nhẹ.
- **Prewitt Filter** → phát hiện biên đơn giản, phù hợp ảnh ít nhiễu.
- **Laplacian Filter** → phát hiện biên mạnh, mọi hướng, thường kết hợp với Gaussian (LoG).