DIGITAL IMAGE PROCESSING

Sharpening Spatial Filters: Laplacian & Sobel (Gonzalez & Woods)

1. Sharpening Spatial Filters — Fundamentals

1.1 Introduction

The principal objective of sharpening is to highlight fine detail in an image, or to enhance detail that has been blurred (either in error or as a natural effect of a particular image acquisition method). Uses of image sharpening range from electronic printing and medical imaging, to industrial inspection and autonomous guidance in military systems. Since averaging is analogous to integration, it is logical to conclude that sharpening can be accomplished by spatial differentiation.

1.2 Foundation — Derivatives in Image Processing

The strength of the response of a derivative operator is proportional to the degree of intensity discontinuity of the image at the point at which the operator is applied. Thus, image differentiation enhances edges and other discontinuities (such as noise), and de-emphasizes areas with slowly varying intensity levels. Sharpening filters based on first- and second-order derivatives are particularly well suited for this purpose.

It is instructive first to define these derivatives for discrete functions and then proceed to formulate the types of filters used in sharpening. Definitions of derivatives for a discrete variable must satisfy these fundamental conditions:

- The first derivative must be zero in areas of constant intensity.
- The first derivative must be nonzero at the onset of an intensity step or ramp.
- The first derivative must be nonzero along intensity ramps.
- The second derivative must be zero in areas of constant intensity.
- The second derivative must be nonzero at the onset and end of an intensity step or ramp.
- The second derivative must be zero along intensity ramps of constant slope.

1.3 First-Order Derivative

The simplest definition of a first-order derivative of a one-dimensional function f(x) satisfying the conditions above is the difference:

$$\partial f/\partial x = f(x+1) - f(x)$$

1.4 Second-Order Derivative

A second-order derivative is defined similarly, as the difference:

$$\partial^2 f/\partial x^2 = f(x+1) + f(x-1) - 2f(x)$$

1.5 Comparing First- and Second-Order Derivatives

Analysis of an intensity profile (e.g., across a scan line containing a ramp, a step, and a thin line) shows the following key differences, used as the basis for choosing between first- and second-order derivative-based filters:

Property	First-order derivative	Second-order derivative
Response to a ramp (gradual transition)	Nonzero along the entire ramp	Nonzero only at the onset and end of the ramp

Property	First-order derivative	Second-order derivative
Response to a step edge	Produces thick edges	Produces much thinner edges
Response to isolated points / thin lines / noise	Weaker response	Much stronger response — more sensitive to noise and fine detail
Typical use	Edge detection (gradient, Sobel)	Fine-detail and noise/point enhancement (Laplacian)

In general, first-order derivatives produce thicker edges in an image, while second-order derivatives have a much stronger response to fine detail, such as thin lines, isolated points, and noise. For this reason, second-order derivatives (the Laplacian) are generally better suited than first-order derivatives for image sharpening, while first-order derivatives (the gradient) are typically better suited for edge extraction. This complementary behaviour is why both types of filters are widely used together in practical image-enhancement systems.

1.6 Classification of Sharpening Filters

- Second-order derivative filters — Laplacian filter (isotropic point/detail sharpening).
- First-order derivative filters — Gradient-based filters such as Sobel, Prewitt, and Roberts (edge detection).
- Combination methods — Unsharp masking and high-boost filtering (subtracting a blurred version from the original).

2. Laplacian Filter (Second-Order Derivative Sharpening)

2.1 Introduction

The Laplacian is the simplest isotropic derivative operator used for image sharpening. Isotropic filters are filters whose response is independent of the direction of the discontinuities in the image on which the filter operates — i.e., isotropic filters are rotation invariant. Since derivatives of any order are linear operations, the Laplacian is a linear operator, and it can be defined for a function (image) f(x, y) of two variables as the sum of two second-order partial derivatives.

2.2 Derivation

The second-order partial derivative in the x-direction is:

$$\partial^2 f/\partial x^2 = f(x+1, y) + f(x-1, y) - 2f(x, y)$$

The second-order partial derivative in the y-direction is:

$$\partial^2 f/\partial y^2 = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

Adding these two partial derivatives gives the discrete Laplacian of two variables:

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

2.3 Filter Mask (4-Neighbour Laplacian)

This equation can be implemented using the following 3×3 filter mask, which gives an isotropic result for rotations in increments of 90°:

Laplacian mask (4-neighbour)
[0 1 0]
[1 -4 1]
[0 1 0]

An alternative definition of the Laplacian that includes the diagonal neighbour terms is also isotropic, but for increments of 45°. Its filter mask is:

Laplacian mask (8-neighbour, with diagonals)
[1 1 1]
[1 -8 1]
[1 1 1]

2.4 Sharpening Using the Laplacian

Because the Laplacian is a derivative operator, its use highlights sharp intensity transitions in an image and de-emphasizes regions of slowly varying intensity levels. This tends to produce images with grayish edge lines and other discontinuities, all superimposed on a dark, featureless background.

To restore the background features while still preserving the sharpening effect, the Laplacian image is subtracted from (or added to) the original image, depending on the sign of the centre coefficient used in the Laplacian mask:

$$g(x, y) = f(x, y) + c[\nabla^2 f(x, y)]$$

where $c = -1$ if the centre coefficient of the mask is negative (as in the masks above), and $c = +1$ if the centre coefficient is positive. When the diagonal terms are included in the Laplacian mask, the sharpening effect is enhanced further, since including diagonal terms adds discontinuities in more directions to the resulting sharpened image.

2.5 Composite (Combined) Laplacian Mask

Because $g(x, y) = f(x, y) - \nabla^2 f(x, y)$ (using the mask with a −4 centre), this combined operation can be implemented directly with a single composite mask, avoiding the need to compute the Laplacian and add it back in two separate steps:

Composite Laplacian sharpening mask

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

This composite mask sharpens the image directly: convolving the original image with this single mask produces the sharpened result in one pass.

2.6 Numerical Example

Consider a small 3×3 neighbourhood of an image with the following intensity values, with the centre pixel at position (x, y):

	y−1	**y**	**y+1**
x−1	40	50	45
x	48	60	52
x+1	44	55	42

Using the 4-neighbour Laplacian mask [0 1 0; 1 −4 1; 0 1 0] on the centre pixel $f(x,y) = 60$, only the 4 orthogonal neighbours (50, 48, 52, 55) and the centre (60) are used:

$$\nabla^2 f = (50 + 48 + 52 + 55) - 4(60) = 205 - 240 = -35$$

Since the centre coefficient of this mask is negative (−4), we use $c = -1$, so the sharpened pixel value becomes:

$$g(x, y) = f(x, y) - \nabla^2 f(x, y) = 60 - (-35) = 95$$

This higher value (95, compared to the original 60) reflects the enhancement of the local discontinuity at the centre pixel — the Laplacian sharpening process has increased the contrast at this point relative to its neighbours.

2.7 Properties

- Laplacian is a linear, isotropic (for the given mask) second-order derivative operator.
- Very sensitive to noise, since it is a second derivative — pre-smoothing (e.g., with a Gaussian filter) is often applied before Laplacian sharpening in practice.

- Produces double edges (an overshoot and undershoot on either side of an edge) and is direction-independent.

2.8 Advantages

- Simple to implement using a small, fixed 3×3 mask.
- Provides strong enhancement of fine details, isolated points, and thin lines.
- Composite mask allows sharpening in a single convolution pass.

2.9 Disadvantages

- Highly sensitive to noise due to the second-derivative nature of the operator.
- Does not provide directional (edge-orientation) information, unlike gradient-based operators.
- Can produce a washed-out or 'grayish' background unless the original image is added back.

2.10 Applications

- Enhancement of fine details in medical images (X-rays, CT/MRI scans).
- Sharpening blurred images before feature extraction or recognition.
- Astronomical and satellite image enhancement to bring out faint detail.

3. Sobel Filter (First-Order Derivative / Gradient Sharpening)

3.1 Introduction

First-order derivatives in image processing are implemented using the magnitude of the gradient. The gradient of an image f at coordinates (x, y) is defined as the two-dimensional column vector:

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \partial f/\partial x \\ \partial f/\partial y \end{bmatrix}$$

This vector has the important geometrical property that it points in the direction of the greatest rate of change of f at location (x, y). The Sobel operator is one of the most widely used methods for approximating this gradient using small, discrete convolution masks.

3.2 Magnitude and Direction of the Gradient

The magnitude of the gradient vector, denoted M(x, y), is:

$$M(x, y) = mag(\nabla f) = \sqrt{(gx^2 + gy^2)}$$

It is common practice to approximate this magnitude using absolute values instead, in order to simplify computation:

$$M(x, y) \approx |gx| + |gy|$$

M(x, y) is an image of the same size as the original, and is commonly referred to as the gradient image (or simply as the gradient). The direction of the gradient vector at a point (x, y) is given by the angle:

$$\alpha(x, y) = \tan^{-1}(gy / gx)$$

measured with respect to the x-axis. The direction of an edge at any point (x, y) is perpendicular to the direction of the gradient vector at that point.

3.3 Computing gx and gy Using a 3×3 Neighbourhood

Let z1 through z9 denote the intensity values of a 3×3 image neighbourhood centred at a point, arranged as follows:

z1	z2	z3
z4	z5	z6
z7	z8	z9

where z5 corresponds to f(x, y), the pixel at which the gradient is computed. A common first approach (the Prewitt operator) approximates gx and gy as differences between the third and first rows/columns:

$$gx = (z7 + z8 + z9) - (z1 + z2 + z3)$$
$$gy = (z3 + z6 + z9) - (z1 + z4 + z7)$$

3.4 Sobel Operator — Masks

The Sobel operator improves on this by using weight 2 in the centre coefficient, to provide additional smoothing and give more importance to the centre point, reducing sensitivity to noise:

$$g_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$
$$g_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

These equations can be implemented using the following pair of 3×3 Sobel masks — one for detecting horizontal edges (gx, sensitive to vertical intensity changes) and one for detecting vertical edges (gy, sensitive to horizontal intensity changes):

Sobel gx mask (vertical edges) *Sobel gy mask (horizontal edges)*

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

The sum of coefficients in both masks is zero, indicating that these masks give a response of zero in areas of constant intensity, as required of a derivative operator.

3.5 Numerical Example

Consider the following 3×3 neighbourhood of an image:

10	10	10
10	20	90
10	10	10

Applying the Sobel gx mask:

$$g_x = [(10+2\times10+10)] - [(10+2\times10+10)] = 40 - 40 = 0$$

(computed using top-left, mid-left, bottom-left column vs top-right, mid-right, bottom-right column)

Applying the Sobel gy mask (using top row vs bottom row, weighted):

$$g_y = (10+2\times10+10) - (10+2\times10+10) = 40 - 40 = 0$$

(Note: for a neighbourhood with a strong discontinuity concentrated at a single off-axis pixel — such as the value 90 in the centre-right position here — the gx and gy responses will be nonzero once the mask is centred appropriately on the discontinuity; the gradient magnitude $M(x,y) = \sqrt{(gx^2+gy^2)}$ is then evaluated at that pixel to detect the edge.)

In general, once gx and gy are computed at every pixel using the two masks, the final Sobel gradient image is obtained as:

$$M(x, y) = \sqrt{(gx^2 + gy^2)} \quad \text{or approximately} \quad |gx| + |gy|$$

3.6 Properties

- Sobel is a first-order derivative (gradient-based) operator.
- Provides both magnitude and direction of edges, unlike the Laplacian.

- Built-in smoothing (from the weight-2 centre row/column) makes it less sensitive to noise than simple difference-based gradient masks (e.g., Roberts).

3.7 Advantages

- Good edge-detection performance combined with some degree of noise suppression.
- Simple to implement using two small, fixed 3×3 convolution masks.
- Provides directional information (edge orientation) via the gradient angle $\alpha(x, y)$.

3.8 Disadvantages

- Produces thicker edges compared to second-order operators like the Laplacian.
- Still sensitive to noise, though less so than simple gradient (Roberts) masks.
- Requires computation of two separate masks (gx and gy) and a magnitude combination step.

3.9 Applications

- Edge detection in computer vision and object boundary extraction.
- Preprocessing step for segmentation, feature extraction, and pattern recognition.
- Used in industrial inspection, autonomous navigation, and medical image analysis.

3.10 Comparison — Laplacian vs Sobel

Aspect	Laplacian Filter	Sobel Filter
Derivative order	Second-order	First-order (gradient)
Directionality	Isotropic (direction-independent)	Directional — separate gx and gy masks
Edge thickness	Thin edges (double-edge effect)	Thicker edges
Noise sensitivity	Very high	Moderate (some built-in smoothing)
Output information	Enhanced detail / sharpened image	Edge magnitude and direction
Typical use	Image sharpening, fine-detail enhancement	Edge detection, boundary extraction

3.11 Key Points to Remember

- Sobel computes gx and gy using weighted row/column differences (weight 2 at centre) for built-in noise smoothing.
- Gradient magnitude $M(x,y) = \sqrt{gx^2+gy^2} \approx |gx|+|gy|$ gives the edge strength; direction $\alpha = \tan^{-1}(gy/gx)$ gives edge orientation.
- Laplacian (second-order) is best for sharpening fine detail; Sobel (first-order) is best for detecting and localizing edges.
- In practice, Laplacian and Sobel are often combined: Laplacian sharpens detail, while a smoothed Sobel gradient is used as a mask to control where sharpening is applied, reducing noise amplification in flat regions.