Unit-3

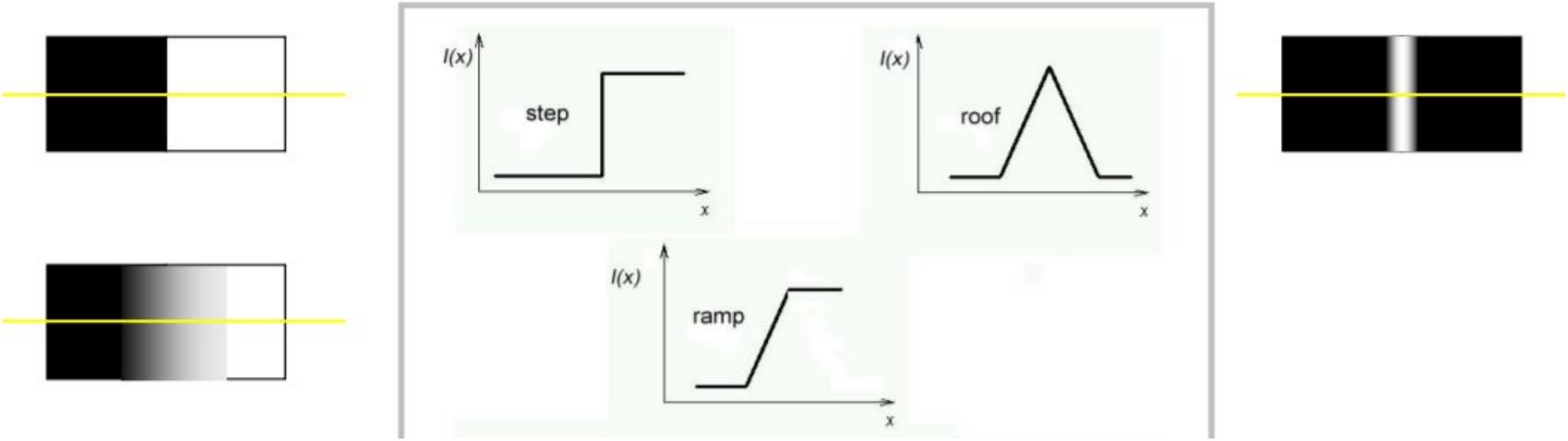
Image Segmentation: Classification of Image Segmentation Techniques, Region approach to Image Segmentation, Edge Based Segmentation, Edge detection - Gradient operator, Using First- Order Derivatives, Roberts Kernel, Prewitt Kernel, Sobel Kernel, Second derivative method of detecting edges in an image, Canny Edge Detector.

Sharpening Spatial Filters- High Pass Filter

- The principal objective of **sharpening is to highlight transitions in intensity.**
- Averaging is **analogous to integration**, it is logical to conclude that sharpening, can be accomplished by spatial differentiation.
- Fundamentally, 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 intensities.

Edge Definition

- Edge is a boundary between two regions with relatively distinct gray level properties.
- Edges are pixels where the brightness function changes abruptly.



000000030 60 90 120 150 180 210 240 255 255 255 255 255 255

Sharpening Spatial Filters - Foundation

- In the two sections that follow, we will discuss sharpening filters that are based on first- and second-order derivatives, respectively.
- We will focus attention initially on one-dimensional derivatives.
- In particular, we are interested in the behavior of these derivatives in areas of-
  - constant intensity,
  - at the onset and end of discontinuities (step and ramp discontinuities),
  - and along intensity ramps.

Sharpening Spatial Filters - Foundation

- The derivatives of a digital function are defined in terms of differences.
- There are various ways to define these differences. However, we require that any definition we use for a *first derivative*-
  1. must be zero in areas of constant intensity;
  2. must be nonzero at the onset of an intensity step or ramp; and
  3. must be nonzero along ramps.
- Similarly, any definition of a *second derivative*
  1. must be zero in constant areas;
  2. must be nonzero at the onset *and* end of an intensity step or ramp; and
  3. must be zero along ramps of constant slope.

Sharpening Spatial Filters - Foundation

- Basic definition of first-order derivative of a one-dimensional function given by the difference-

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

- Second-order derivative is defined as the difference-

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

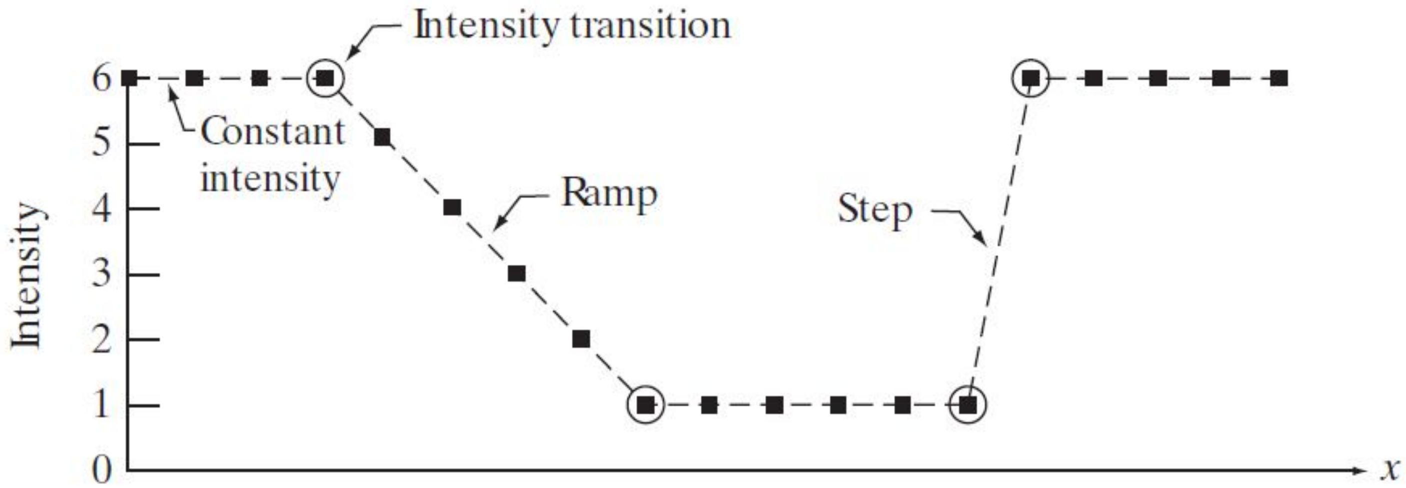
Explanation- $\partial^2 f/\partial x^2 = (\ f(x + 1) - f(x)\ ) - (\ f(x) - f(x-1)\ )$

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

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

Illustration of the first and second derivatives of a 1-D digital function

- As the figure shows, the scan line contains an **intensity ramp**, **three sections of constant intensity,** and an **intensity step**.
- The circles indicate the onset or end of intensity transitions.



Scan line	6	6	6	6	5	4	3	2	1	1	1	1	1	1	6	6	6	6	6 → x
1st derivative		0	0	−1	−1	−1	−1	−1	0	0	0	0	0	5	0	0	0	0	
2nd derivative		0	0	−1	0	0	0	0	1	0	0	0	0	5	−5	0	0	0	

A section of a horizontal intensity profile from an image

Illustration of the first and second derivatives of a 1-D digital function

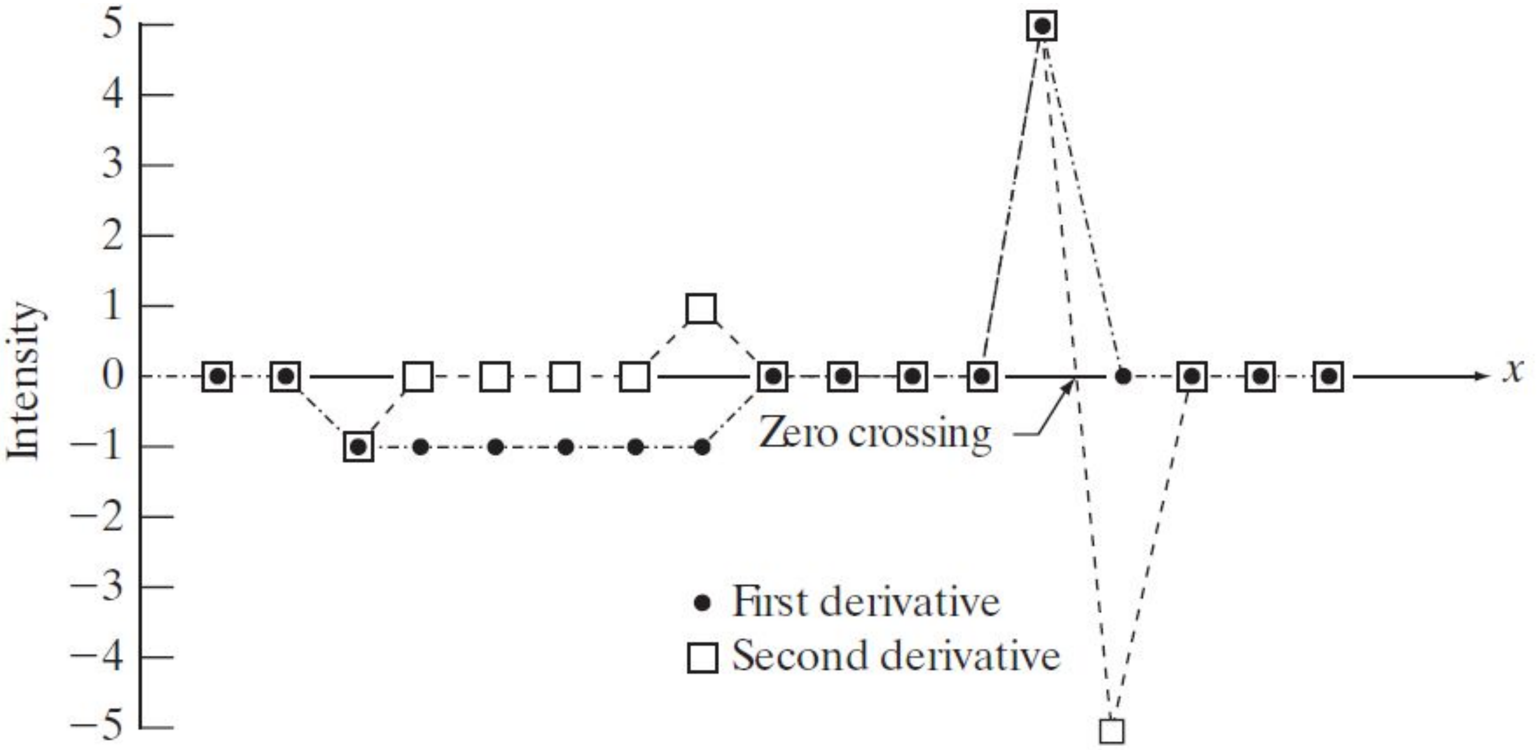
- When computing the first derivative at a location x, we subtract the value of the function at that location from the next point. So this is a "look-ahead" operation
  - In this method we take the 1 st derivative of the intensity value across the image and find points where the derivative is maximum then the edge could be located.
- Similarly, to compute the second derivative at x, we use the previous and the next points in the computation.
- To avoid a situation in which the previous or next points are outside the range of the scan line, we show derivative computations in Figure, from the second through the penultimate (second last) points in the sequence.

Illustration of the first and second derivatives of a 1-D digital function

- First, we encounter an area of constant intensity and, as Figures show, both derivatives are zero there, so condition (1) is satisfied for both.
- Next, we encounter an intensity ramp followed by a step, and we note that the first-order derivative is nonzero at the onset of the ramp and the step
- similarly, the second derivative is nonzero at the onset *and* end of both the ramp and the step; therefore, property (2) is satisfied for both derivatives
- we see that property (3) is satisfied also for both derivatives because the first derivative is nonzero and the second is zero along the ramp.

Illustration of the first and second derivatives of a 1-D digital function

Note that the sign of the second derivative changes at the onset and end of a step or ramp.
The 2nd derivative of an image where the image highlights regions of **rapid intensity change** and is therefore often used for edge detection- zero crossing edge detectors.



Scan line	6	6	6	6	5	4	3	2	1	1	1	1	1	1	6	6	6	6	6
1st derivative		0	0	−1	−1	−1	−1	−1	0	0	0	0	0	5	0	0	0	0	
2nd derivative		0	0	−1	0	0	0	0	1	0	0	0	0	5	−5	0	0	0	

Using the Second Derivative for Image Sharpening

- The approach basically consists of defining a discrete formulation of the second-order derivative and then constructing a filter mask based on that formulation.
- We are interested in ***isotropic*** filters, whose *response is independent of the direction of the discontinuities in the image to which the filter is applied.*
  - In other words, isotropic filters are *rotation invariant*, in the sense that rotating the image and then applying the filter gives the same result as applying the filter to the image first and then rotating the result.

Image Sharpening Filter -The Laplacian

- The simplest isotropic derivative operator is the Laplacian which, for a function (image) f(x,y) of two variables, is defined as-

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

  the Laplacian is a linear operator.

- To express this equation in discrete form, we use the definition in equation $\partial^2 f/\partial x^2 = f(x+1) + f(x-1) - 2f(x)$, keeping in mind that we have to carry a second variable-
- In the x-direction, we have- $\partial^2 f/\partial x^2 = f(x+1, y) + f(x-1, y) - 2f(x,y) \quad .....1$
- In the y-direction we have - $\partial^2 f/\partial y^2 = f(x, y+1) + f(x, y-1) - 2f(x,y) \quad .....2$

Image Sharpening Filter -The Laplacian

- Therefore, it follows from the preceding three equations that the discrete Laplacian of two variables is given by taking the sum of partials -

$$\nabla^2 f(x, y) = f(x + 1, y) + f(x - 1, y) + f(x, y + 1) + f(x, y - 1) - 4f(x, y) \quad \ldots .3$$

- The mask is given by –

0	1	0
1	-4	1
0	1	0

$f(x-1, y-1)$	$f(x-1, y)$	$f(x-1, y+1)$
$f(x, y-1)$	$f(x, y)$	$f(x, y+1)$
$f(x+1, y-1)$	$f(x+1, y)$	$f(x+1, y+1)$

- The mask gives isotropic result in increments of 90°

Image Sharpening Filter -The Laplacian

- The diagonal directions can be incorporated in the definition of the digital Laplacian by adding two more terms to Equation 3, one for each of the two diagonal directions.
- The form of each new term is the same as either Eq. 1 or 2, but the coordinates are along the diagonals.
- Fig. (a) Filter mask used to implement Eq. 3
- Fig. (b) Mask used to implement an extension of this equation that includes the diagonal terms.
- Fig. (c) and (d) Two other implementations of the Laplacian found frequently in practice.

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

a b
c d

FIGURE 3.37

Image Sharpening Filter -The Laplacian

- Because the Laplacian is a derivative operator, its **highlights** intensity discontinuities in an image and **de-emphasizes** regions with slowly varying intensity levels.
- This will tend to produce images that have grayish edge lines and other discontinuities, all superimposed on a dark, featureless background.
- Background features can be “recovered” while still preserving the sharpening effect of the Laplacian simply by adding the Laplacian image to the original.
- *it is important to keep in mind which definition of the Laplacian is used.* ***If the definition used has a negative center coefficient, then we subtract,*** *rather than add, the Laplacian image to obtain a sharpened result.*

Image Sharpening Filter -The Laplacian

- Thus, the basic way in which we use the Laplacian for image sharpening is –

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

- Where f(x,y) and g(x,y) are the input and sharpened images, respectively
- The constant is c=-1 if the Laplacian filters in Fig. (a) or (b) are used, and c = 1 if either of the other two filters is used.

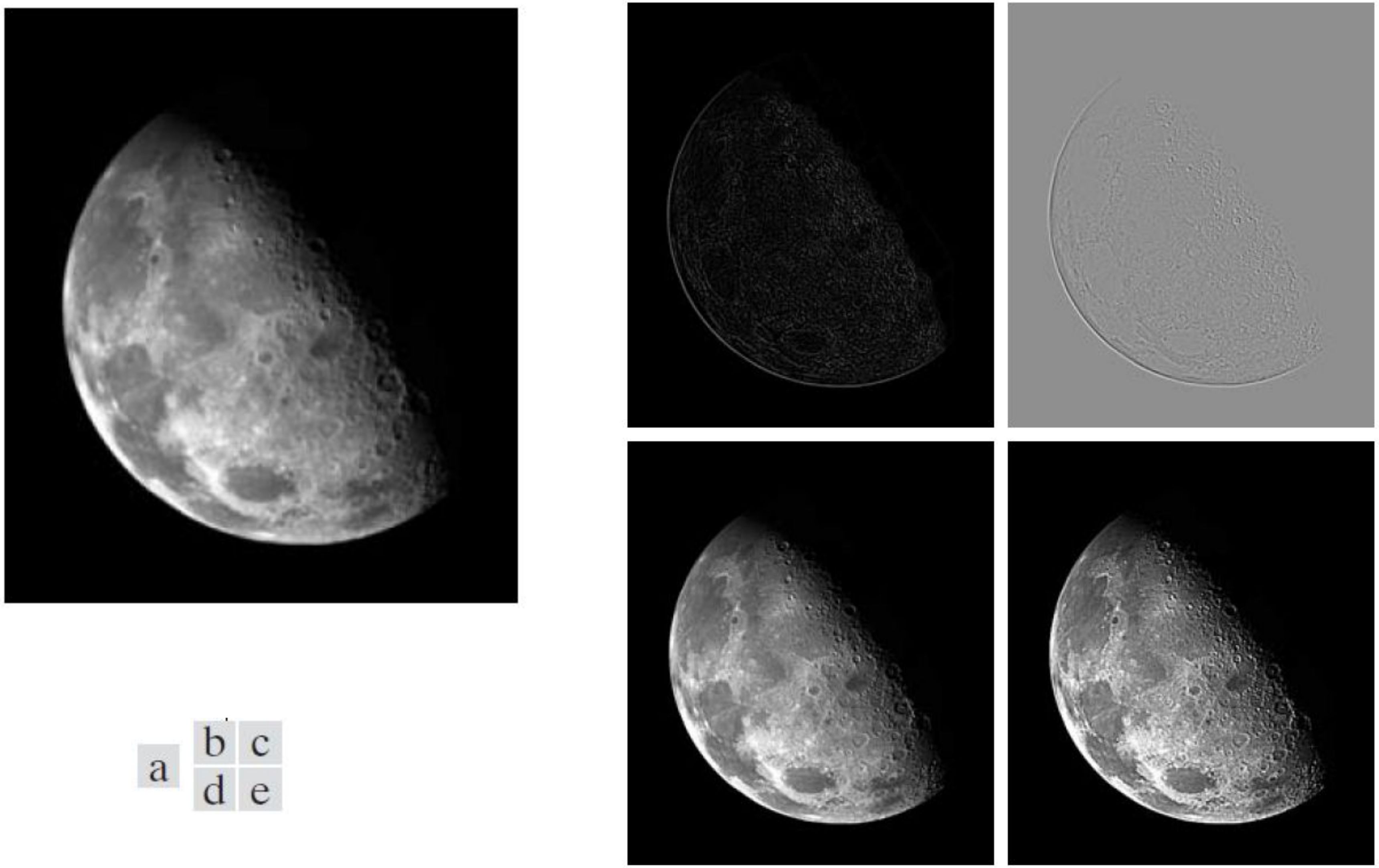


FIGURE 3.38
(a) Blurred image of the North Pole of the moon. (b) Laplacian without scaling. (c) Laplacian with scaling. (d) Image sharpened using the mask in Fig. 3.37(a). (e) Result of using the mask in Fig. 3.37(b). (Original image courtesy of NASA.)

```
%Input Image
clear all;
pkg load image;
A=imread('coins.png');
size(A);
figure,
subplot(2,2,1);imshow(A); title('original Image');
%Preallocate the matrices with zeros
I1=A;
I=zeros(size(A));
I2=zeros(size(A));
%Filter Masks


#F1=[0 1 0;1 -4 1; 0 1 0];
```

Unsharp Masking and High boost Filtering

A process that has been used for many years by the printing and publishing industry to sharpen images consists of subtracting an unsharp (smoothed) version of an image from the original image. This process, called *unsharp masking*, consists of the following steps:

1. Blur the original image.
2. Subtract the blurred image from the original (the resulting difference is called the *mask*.)
3. Add the mask to the original.

Unsharp Masking and High boost Filtering

Letting $\bar{f}(x, y)$ denote the blurred image, unsharp masking is expressed in equation form as follows. First we obtain the mask:

$$g_{\text{mask}}(x, y) = f(x, y) - \bar{f}(x, y) \qquad (3.6\text{-}8)$$

Then we add a weighted portion of the mask back to the original image:

$$g(x, y) = f(x, y) + k * g_{\text{mask}}(x, y) \qquad (3.6\text{-}9)$$

where we included a weight, k ($k \geq 0$), for generality. When k=1, we have unsharp masking. When k >1, the process is referred to as *highboost filtering*.

Using First-Order Derivatives for (Nonlinear) Image Sharpening—The Gradient

- First derivatives in image processing are implemented using the magnitude of the gradient. For a function f(x,y), the gradient of *f* at coordinates (*x*, *y*) is defined as the two-dimensional column *vector*-

$$\nabla f \equiv \operatorname{grad}(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$

- 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*).

Using First-Order Derivatives for (Nonlinear) Image Sharpening—The Gradient

- The *magnitude* (*length*) of vector ∇f denoted as *M*(*x*, *y*), where M(x,y) is the *value* at (*x*, *y*) of the rate of change in the direction of the gradient vector.

$$M(x, y) = \text{mag}(\nabla f) = \sqrt{g_x^2 + g_y^2}$$

- Note that *M*(*x*, *y*) is an image of the same size as the original. It is common practice to refer to this image as the *gradient image.*
- In some implementations, it is more suitable computationally to approximate the squares and square root operations by absolute values:

$$M(x, y) \approx |g_x| + |g_y|$$