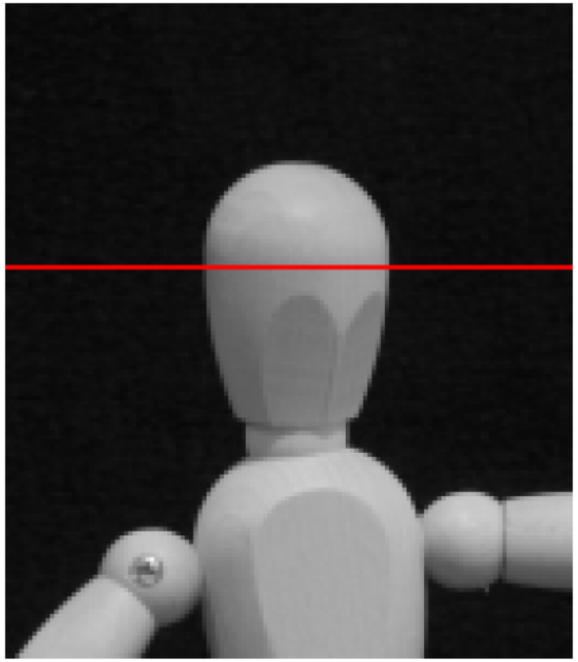
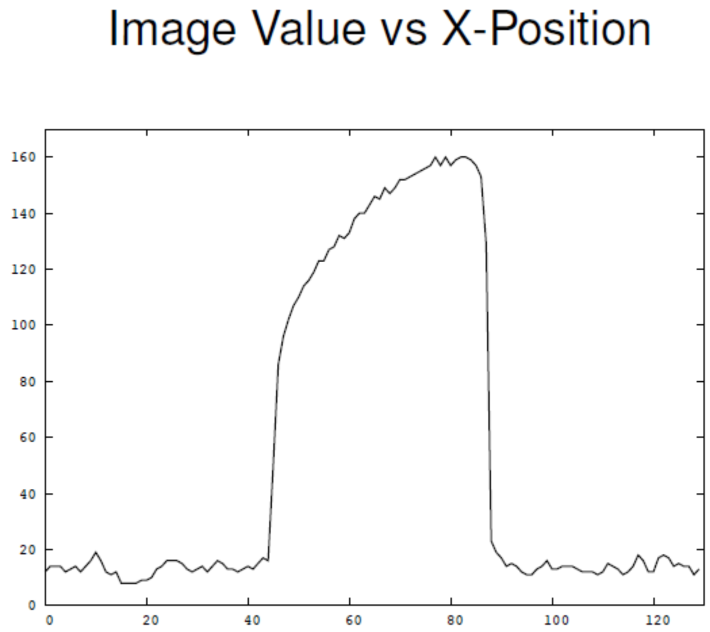
- Edges are significant local changes in the image and are important features for analyzing images.
- Edge detection is frequently the first step in recovering information from images.
- Edge detection : An image processing task that aims to find edges and contours in images.

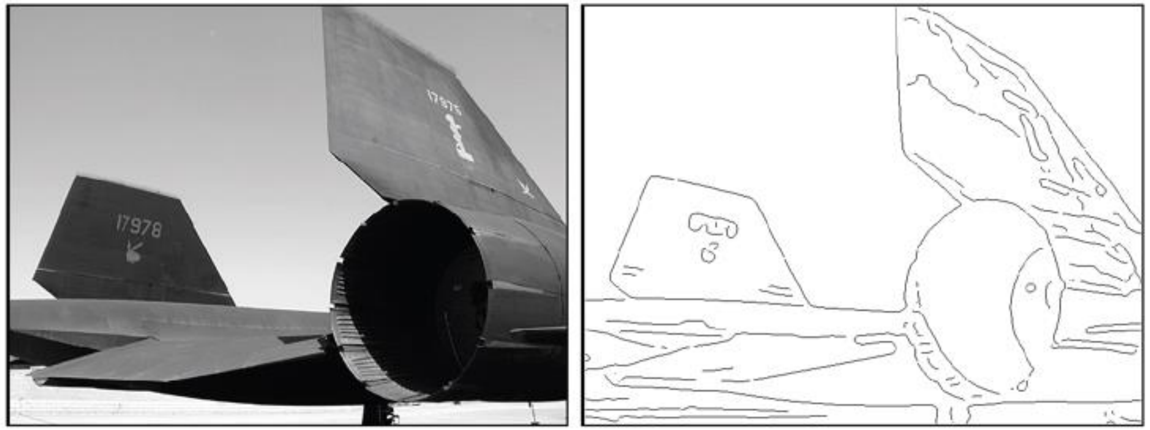




- An edge point is a point in an image with coordinates [i,j] at the location of a significant local intensity change in the image.

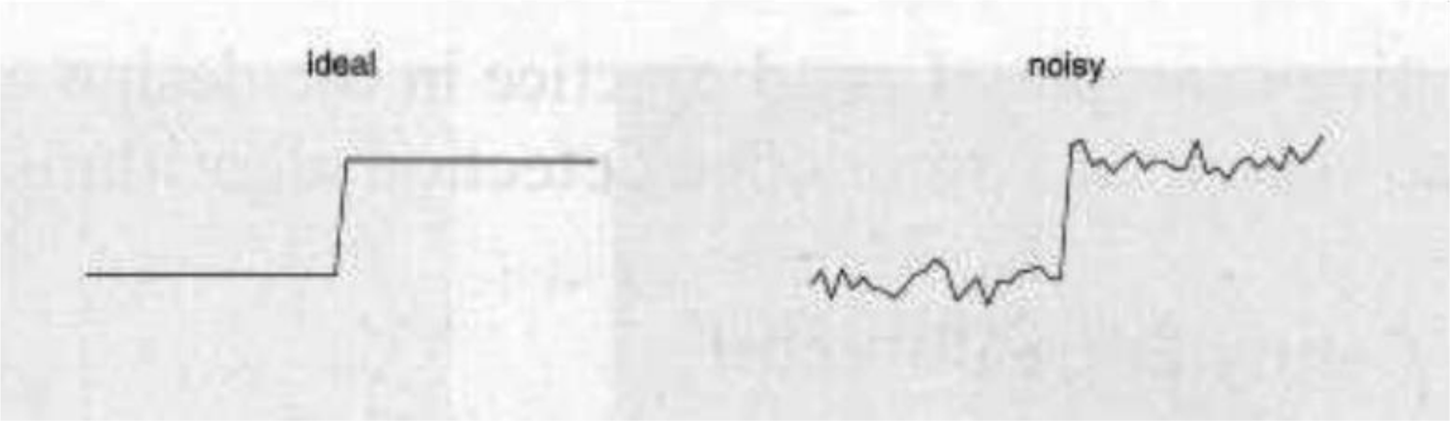
- An edge detector is an algorithm that produces a set of edges {edgepoints or edge fragments} from an image.

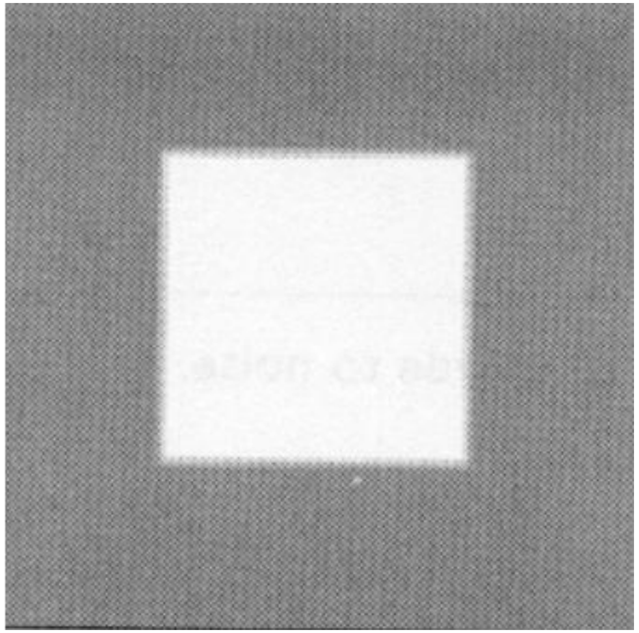
- A contour is a list of edges or the mathematical curve that models the list of edges.



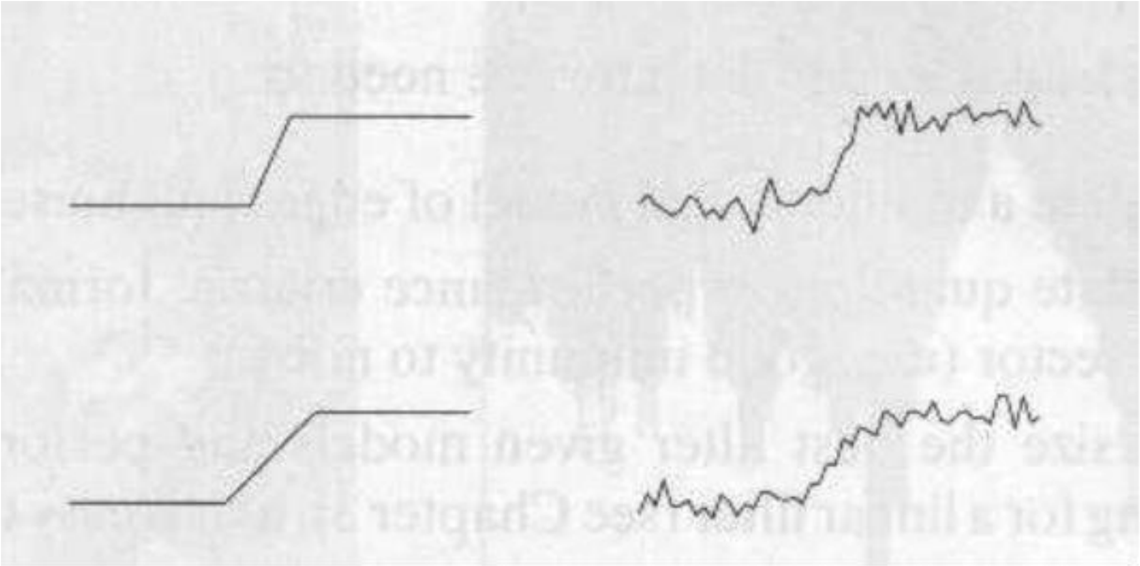
(a) (b)

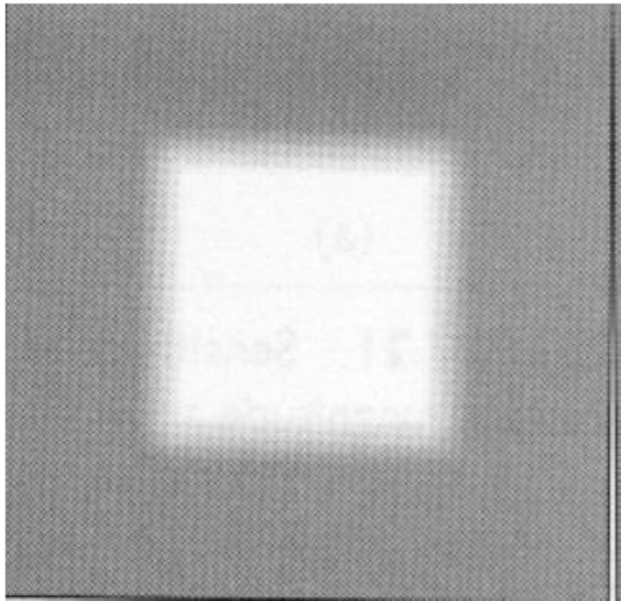
Step edge: the image intensity abruptly changes from one value on one side of the discontinuity to a different value on the opposite side.





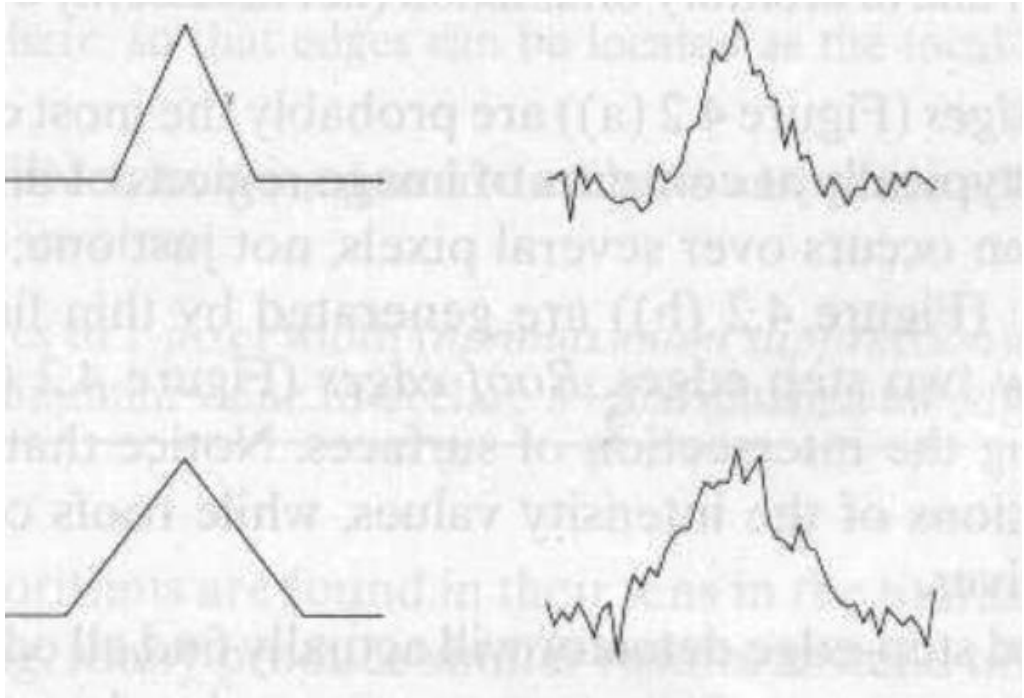
Ramp edge: a step edge where the intensity change is not instantaneous but occur over a finite distance.





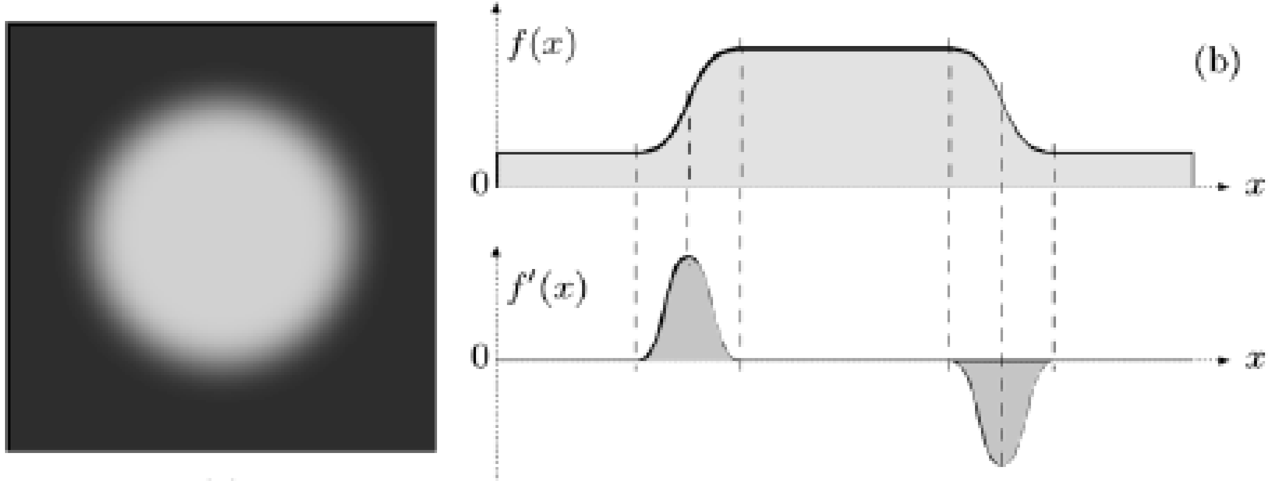
Ridge edge: the image intensity abruptly changes value but then returns to the starting value within some short distance (i.e., usually generated by lines).

Roof edge: a ridge edge where the intensity change is not instantaneous but occur over a finite distance (i.e., usually generated by the intersection of two surfaces).



- 1-D edges
- Realistically, edges is a smooth (blurred) step function
- Edges can be characterized by high value first derivative

$$f'(x) = \frac{df}{dx}(x)$$



(a)

(c)

Edge Detection Using First Derivative

1D functions

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h} \approx f(x+1) - f(x) \quad (h=1)$$

Forward Difference

$$\Delta_+ f(x) = f(x+1) - f(x)$$

Backward Difference

$$\Delta_- f(x) = f(x) - f(x-1)$$

Central Difference

$$\Delta f(x) = \frac{1}{2}(f(x+1) - f(x-1))$$

Finite Differences as Convolutions

Forward Difference

$$\Delta_+ f(x) = f(x+1) - f(x)$$

Take a convolution kernel: $H = [1 \ \ -1 \ \ 0]$

$$\Delta_+ f = f * H$$

(Remember that the kernel H is flipped in convolution)

Finite Differences as Convolutions

Central Difference

$$\Delta f(x) = \frac{1}{2}\left(f(x+1) - f(x-1)\right)$$

Convolution kernel here is: $H = \left[\ \tfrac{1}{2}\ \ 0\ \ -\tfrac{1}{2}\ \right]$

$$\Delta f(x) = f * H$$

Notice: Derivative kernels sum to zero!

The gradient is the two-dimensional equivalent of the first derivative and is defined as the *vector*

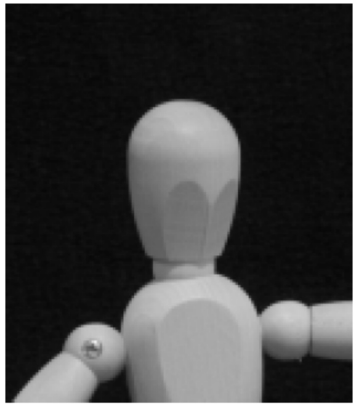
$$\mathbf{G}[f(x,y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}. \tag{5.1}$$

- Images have two parameters: $I(x, y)$
- We can take derivatives with respect to x or y
- Central differences:

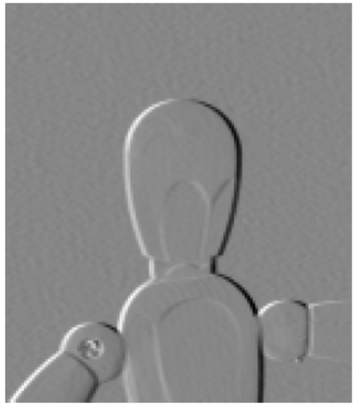
$$\Delta_x I = I * H_x, \quad \text{and} \quad \Delta_y I = I * H_y,$$

$$\text{where } H_x = \begin{bmatrix} 0.5 & 0 & -0.5 \end{bmatrix} \text{ and } H_y = \begin{bmatrix} -0.5 \\ 0 \\ 0.5 \end{bmatrix}$$

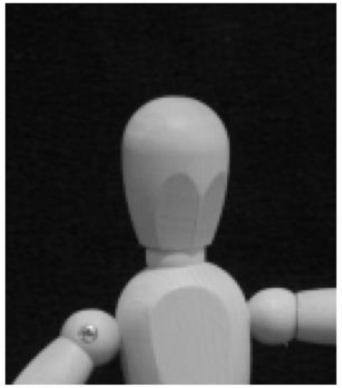
x-derivative using central difference:



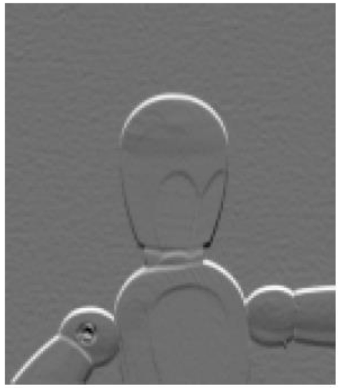
$$* \begin{bmatrix} \frac{1}{2} & 0 & -\frac{1}{2} \end{bmatrix} =$$



y-derivative using central difference:



$$* \begin{bmatrix} 0.5 \\ 0 \\ -0.5 \end{bmatrix} =$$



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

$$G[f(x,y)] \approx |G_x| + |G_y|$$

$$G[f(x,y)] \approx \max(|G_x|, |G_y|).$$

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

The Roberts cross operator provides a simple approximation to the gradient magnitude:

$$G\left[f[i,j]\right] = |f[i,j] - f[i+1,j+1]| + |f[i+1,j] - f[i,j+1]|. \qquad (5.10)$$

Using convolution masks, this becomes

$$G\left[f[i,j]\right] = |G_x| + |G_y| \qquad (5.11)$$

where G_x and G_y are calculated using the following masks:

$G_x =$

1	0
0	−1

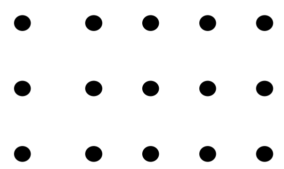
$G_y =$

0	−1
1	0

(5.12)

good approximation

(x+1/2,y+1/2)



Another Approximation

- Consider the arrangement of pixels about the pixel (*i*, *j*):

$$\begin{array}{ccc} a_0 & a_1 & a_2 \\ a_7 & [i,j] & a_3 \\ a_6 & a_5 & a_4 \end{array}$$

- The partial derivatives $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial y}$ can be computed by:

$$M_x = (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6)$$
$$M_y = (a_6 + ca_5 + a_4) - (a_0 + ca_1 + a_2)$$

- The constant *c* implies the emphasis given to pixels closer to the center of the mask.

Prewitt Operator

- Setting *c* = 1, we get the Prewitt operator:

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

M_x and M_y are approximations at (i, j)

Sobel Operator

- Setting c = 2, we get the Sobel operator:

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

M_x and M_y are approximations at (i, j)

Edge Detection Steps Using Gradient

(1) Smooth the input image ($\hat{f}(x, y) = f(x, y) * G(x, y)$)

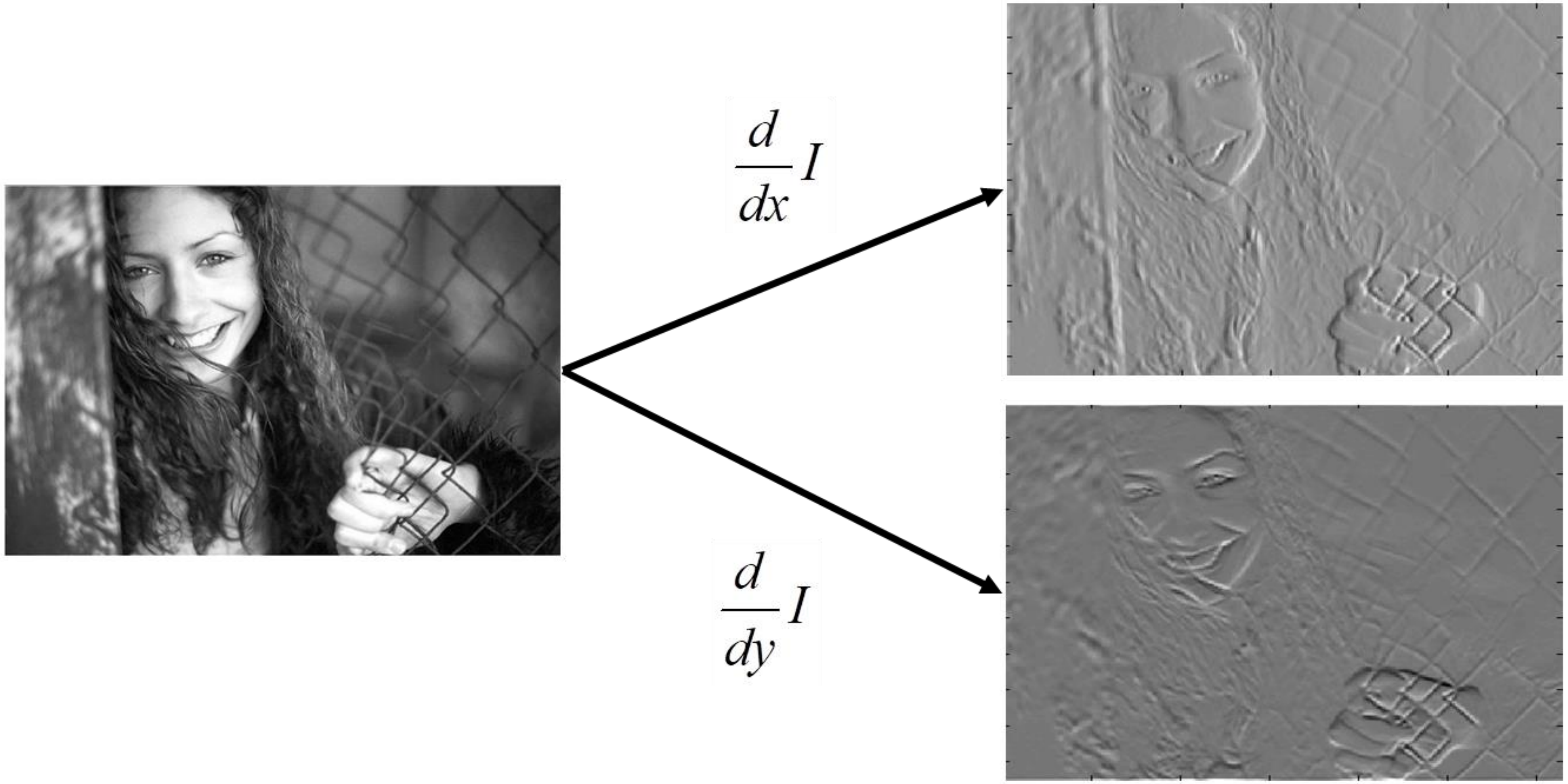
(2) $\hat{f}_x = \hat{f}(x, y) * M_x(x, y) \longrightarrow \frac{\partial f}{\partial x}$

(3) $\hat{f}_y = \hat{f}(x, y) * M_y(x, y) \longrightarrow \frac{\partial f}{\partial y}$

(4) $magn(x, y) = |\hat{f}_x| + |\hat{f}_y|$ (i.e., sqrt is costly!)

(5) $dir(x, y) = \tan^{-1}(\hat{f}_y / \hat{f}_x)$

(6) If $magn(x, y) > T$, then possible edge point


$\frac{d}{dx}I$
$\frac{d}{dy}I$

$$\nabla = \sqrt{\left(\frac{d}{dx} I\right)^2 + \left(\frac{d}{dy} I\right)^2}$$

$$\nabla \geq Threshold = 100$$