Azmi Tewfiq. Hussain Al-Rawi,
Faculty of C. S.& I.T. University of Anbar

edge

Edges contain some of the most useful information in an image. We may use edges to measure the size of objects in an image; to isolate particular objects from their background; to recognize or classify objects. There are a large number of edge-finding algorithms in this field.

An edge may be loosely defined as a line of pixels showing an observable difference. For example, consider the two blocks of pixels shown in figure 1.

51	52	53	59
54	52	53	62
50	52	53	68
55	52	53	55

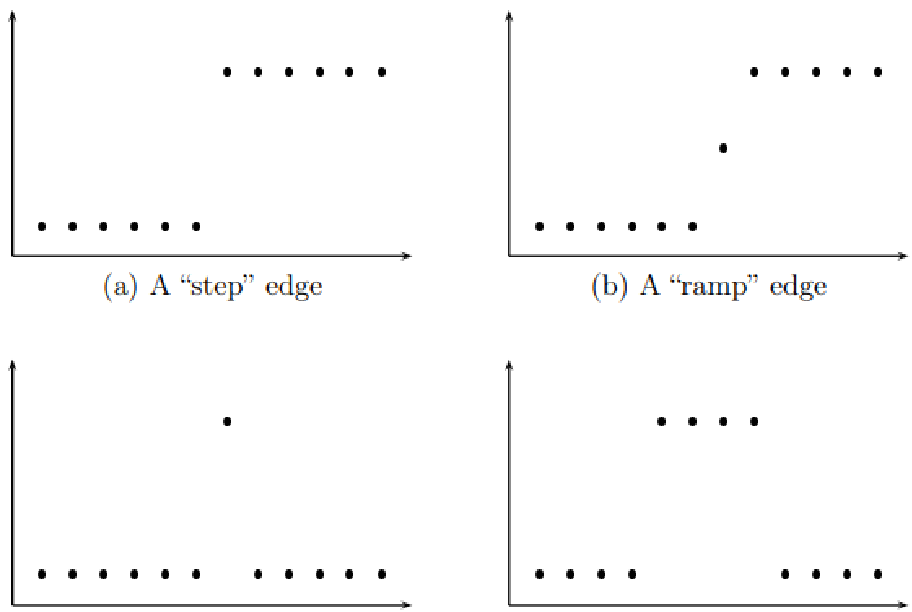
50	53	150	160
51	53	150	170
52	53	151	190
51	53	152	155

Figure 1: Blocks of pixels.

In the right hand block, there is a clear difference between the grey values in the second and third columns.

Differences and edges:

If we consider the grey values along this line, and plot their values, we will have something similar to that shown in figure 2 (a) or (b). If we now plot the differences between each grey value and its predecessor from the ramp edge, we would obtain a graph similar to that shown in figure 3



(a) A "step" edge (b) A "ramp" edge

(c) A line (d) A "roof"

Figure 2: Gray values across edges

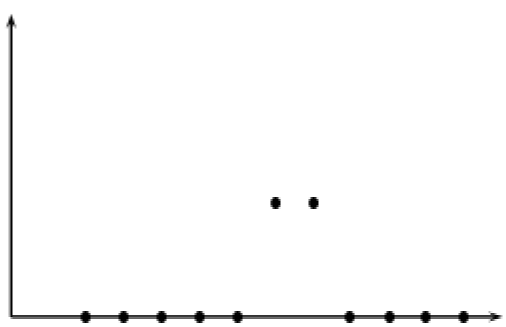


Figure 3: Differences of the edge function

To see where this graph comes from, suppose that the values of the "ramp" edge in figure 2(b) are, from left to right:

$$20,\ 20,\ 20,\ 20,\ 20,\ 20,\ 100,\ 180,\ 180,\ 180,\ 180,\ 180.$$

If we form the differences, by subtracting each value from its successor, we obtain:

$$0,\ 0,\ 0,\ 0,\ 0,\ 80,\ 80,\ 0,\ 0,\ 0,\ 0$$

and it is these values which are plotted in figure 3. It appears that the difference tends to enhance edges, and reduce other components. So if we could "difference" the image, we would obtain the effect we want. We can define the difference in three separate ways:

- the *forward difference*: $\Delta f(x) = f(x+1) - f(x)$,
- the *backward difference*: $\nabla f(x) = f(x) - f(x-1)$,
- the *central difference*: $\delta f(x) = f(x+1) - f(x-1)$.

However, an image is a function of two variables, so we can generalize these definitions to include both the x and y values:

$$\begin{aligned} \Delta_x f(x,y) &= f(x+1,y) - f(x,y) & \qquad \Delta_y f(x,y) &= f(x,y+1) - f(x,y) \\ \nabla_x f(x,y) &= f(x,y) - f(x-1,y) & \qquad \nabla_y f(x,y) &= f(x,y) - f(x,y-1) \\ \delta_x f(x,y) &= f(x+1,y) - f(x-1,y) & \qquad \delta_y f(x,y) &= f(x,y+1) - f(x,y-1) \end{aligned}$$

Some difference filters: To see how we might use δx to determine edges in the x direction, consider the function values around a point p(x,y).

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

To find the filter which returns the value δ_x, we just compare the coefficients of the function's values in δ_x with their position in the array:

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

This filter thus will find vertical edges in an image and produce a reasonably bright result. However, the edges in the result can be a bit "jerky"; this can be overcome by smoothing the result in the opposite direction; by using the filter

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

Both filters can be applied at once, using the combined filter:

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

This filter, and its companion for finding horizontal edges:

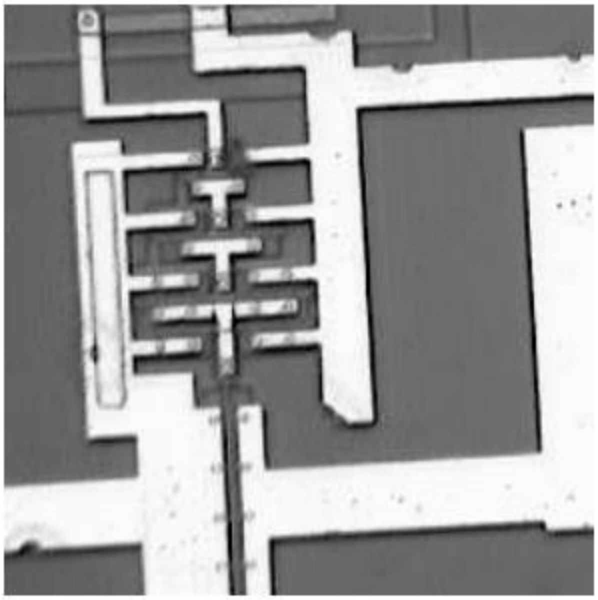
$$P_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

are the Prewitt filters for edge detection.

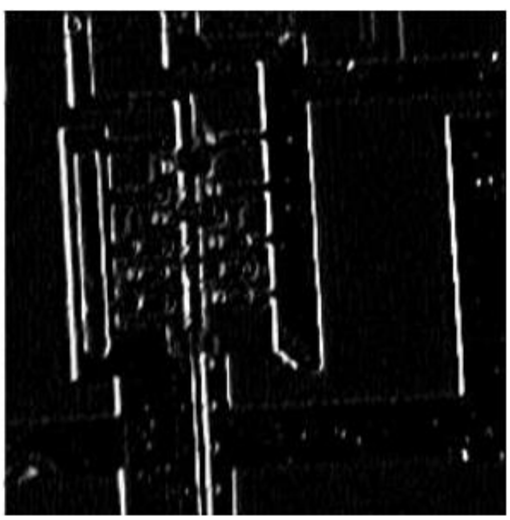
So if P_x and were the grey values produced by applying P_y and to an image, then the output grey value can be chosen by any of these methods

1. $v = \max\{|p_x|, |p_y|\}$,
2. $v = |p_x| + |p_y|$,
3. $v = \sqrt{p_x^2 + p_y^2}$.

Applying each of P_x and P_y individually provides the results shown in below figures

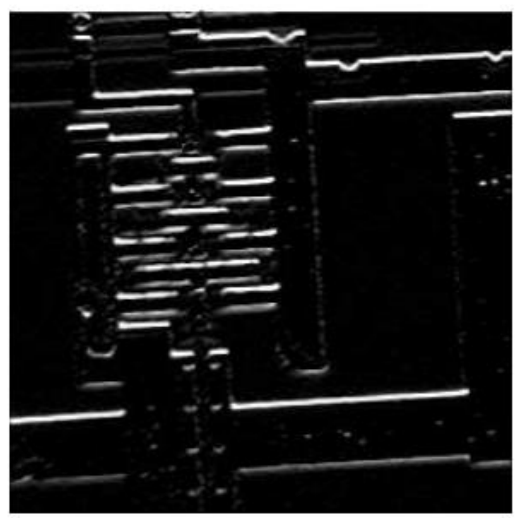


Original image integrated circuit



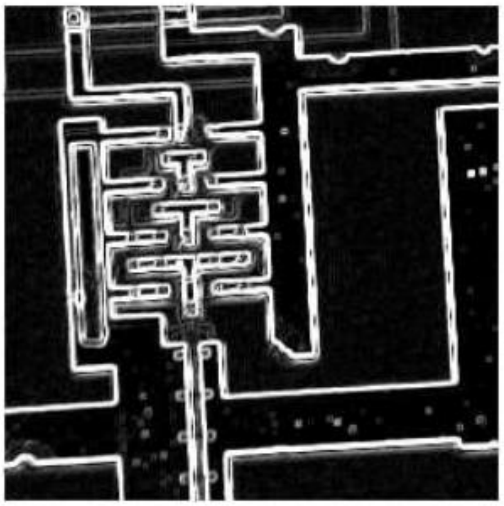
(a)

P_x

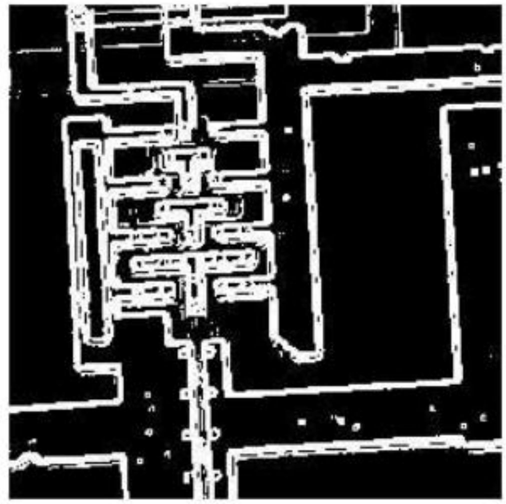


(b)

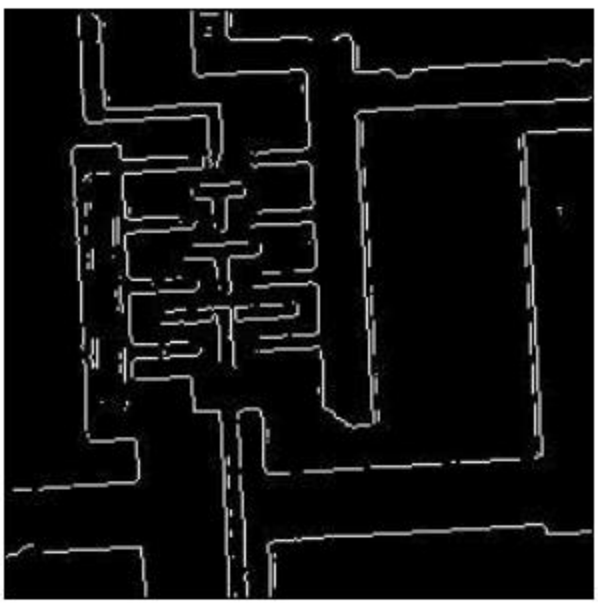
P_y



(a)



(b)



The prewitt option of edge after convert to binary image

Slightly different edge finding filters are the *Roberts cross-gradient filters*:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ and } \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

and the *Sobel filters*:

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} -1 & -2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}.$$

The *Sobel filters* are similar to the Prewitt filters, in that they apply a smoothing filter in the opposite direction to the central difference filter. In the Sobel filters, the smoothing takes the form

$$\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

takes the form which gives slightly more prominence to the central pixel