THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Edge Detection

Edges in images are areas with strong intensity contrasts; a jump in intensity from one pixel to the next.

The process of edge detection significantly reduces the amount of data and filters out unneeded information, while preserving the important structural properties of an image.

There are many different edge detection methods, the majority of which can be grouped into two categories:

- Gradient,
- and Laplacian.

The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image.

The Laplacian method searches for zero crossings in the second derivative of the image .

We will look at two examples of the gradient method, Sobel and Prewitt.

40

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Edge Detection

Edge detection is a major application for convolution.

What is an edge:

- A location in the image where is a sudden change in the intensity/colour of pixels.
- A transition between objects or object and background.
- From a human visual perception perspective it attracts attention.

Problem: Images contain noise, which also generates sudden transitions of pixel values.

Usually there are three steps in the edge detection process:

1) **Noise reduction**
   Suppress as much noise as possible without removing edges.
2) **Edge enhancement**
   Highlight edges and weaken elsewhere (high pass filter).
3) **Edge localisation**
   Look at possible edges (maxima of output from previous filter) and eliminate spurious edges (often noise related).

41

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Edge Detection

Gradient Estimation

Estimation of the intensity gradient at a pixel in the x and y direction, for an image f, is given by:

$$\frac{\partial f}{\partial x} = f(x+1, y) - f(x-1, y)$$

$$\frac{\partial f}{\partial y} = f(x, y+1) - f(x, y-1)$$

We can introduce noise smoothing by convoluting with a low pass filter (e.g. mean, Gaussian, etc)

The gradient calculation (g_x,g_y) can be expressed as:

$$g_x = h_x * f(x, y)$$

$$g_y = h_y * f(x, y)$$

42

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Sobel Filter

The Sobel filter is used for edge detection.

It works by calculating the gradient of image intensity at each pixel within the image. It finds the direction of the largest increase from light to dark and the rate of change in that direction.

The result shows how abruptly or smoothly the image changes at each pixel, and therefore how likely it is that that pixel represents an edge.

It also shows how that edge is likely to be oriented.

The result of applying the filter to a pixel in a region of constant intensity is a zero vector.

The result of applying it to a pixel on an edge is a vector that points across the edge from darker to brighter values.

43

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Sobel Filter

The sobel filter uses two 3 x 3 kernels. One for changes in the horizontal direction, and one for changes in the vertical direction.

The two kernels are convolved with the original image to calculate the approximations of the derivatives.

If we define Gx and Gy as two images that contain the horizontal and vertical derivative approximations respectively, the computations are:

$$G_x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} * A \quad \text{and} \quad G_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} * A$$

Where A is the original source image.

The x coordinate is defined as increasing in the right-direction and the y coordinate is defined as increasing in the down-direction.

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Sobel Filter

To compute G_x and G_y we move the appropriate kernel (window) over the input image, computing the value for one pixel and then shifting one pixel to the right. Once the end of the row is reached, we move down to the beginning of the next row.

The example below shows the calculation of a value of G_x:

a11	a12	a13	…	
a21	a22	a23	…	
a31	a32	a33	…	
…	…	…	…	

Input image

$$\text{kernel} = \begin{array}{|c|c|c|} \hline 1 & 0 & -1 \\ \hline 2 & 0 & -2 \\ \hline 1 & 0 & -1 \\ \hline \end{array}$$

b11	b12	b13	…	
b21	b22	b23	…	
b31	b32	b33	…	
…	…	…	…	

Output image (G_x)

$b_{22} = a_{13} - a_{11} + 2a_{23} - 2a_{21} + a_{33} - a_{31}$

45

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Edge Detection

The kernels contain positive and negative coefficients.

This means the output image will contain positive and negative values.

For display purposes we map the gradient of zero onto a half-tone grey level.

This makes negative gradients appear darker, and positive gradients appear brighter.

The kernels are sensitive to horizontal and vertical transitions.

The measure of an edge is its amplitude and angle. These are readily calculated from G_x and G_y.

46

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Sobel Filter

At each pixel in the image, the gradient approximations given by G_x and G_y are combined to give the gradient magnitude, using:

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

The gradient's direction is calculated using:

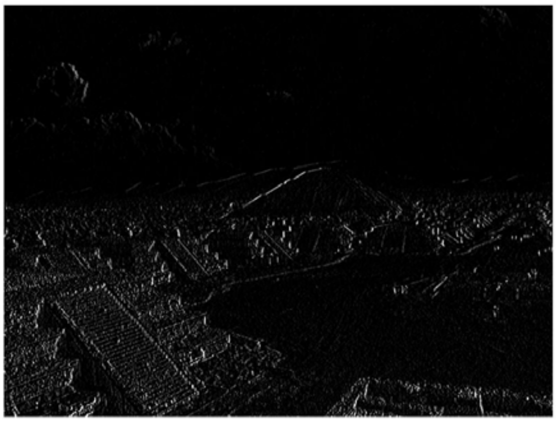
$$\Theta = \arctan\left(\frac{G_y}{G_x}\right)$$

A Θ value of 0 would indicate a vertical edge that is darker on the left side.

47

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Sobel Filter



The image to the right above is G_x, calculated as:

$$G_x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} * A$$

Where A is the original image to the left.

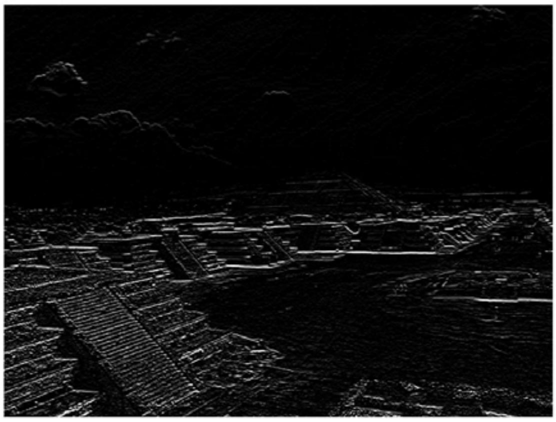
Notice the general orientation of the edges.

What would you expect to be different in G_y?

48

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Sobel Filter



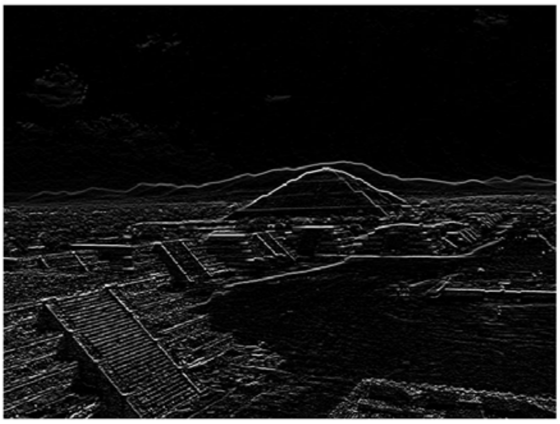
The image to the right above is Gy, calculated as: $G_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} * A$

Where A is the original image to the left.

What do we expect from the combined image?

49

Sobel Filter



The image to the right above is the result of combining the G_x and G_y derivative approximations calculated from image A on the left.

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Sobel Filter example

Convolve the Sobel kernels to the original image

10	50	10	50	10
10	55	10	55	10
10	65	10	65	10
10	50	10	50	10
10	55	10	55	10

Original image

1	0	-1
2	0	-2
1	0	-1

-1	-2	-1
0	0	0
1	2	1
