CO70058 Computer Vision

Tutorial 1 - Edge Detection and Pre-Processing

1. A vertical edge has the following intensity profile in part of a raster image:

$$\begin{matrix} 3 & 4 & 8 & 15 & 25 & 44 & 50 & 52 \\ 3 & 4 & 8 & 15 & 25 & 44 & 50 & 52 \\ 3 & 4 & 8 & 15 & 25 & 44 & 50 & 52 \\ 3 & 4 & 8 & 15 & 25 & 44 & 50 & 52 \\ 3 & 4 & 8 & 15 & 25 & 44 & 50 & 52 \\ 3 & 4 & 8 & 15 & 25 & 44 & 50 & 52 \\ 3 & 4 & 8 & 15 & 25 & 44 & 50 & 52 \\ 3 & 4 & 8 & 15 & 25 & 44 & 50 & 52 \end{matrix}$$

What is the gradient profile through the edge when computed by:

A. The 3×3 Prewitt operator

$$h_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, h_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

B. The 3×3 Sobel operator

$$h_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, h_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

C. The 3×3 Laplacian operator

$$h = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Comment briefly on the results, identify the position of zero crossing signified by the Laplacian operator, and correlate this with the peak edge magnitude detected by the Prewitt and Sobel operators. Note that when calculating edge magnitude $g(x) = \sqrt{g_x^2 + g_y^2}$.

2. For a 3×3 block belonging to an image:

$$\begin{matrix} 7 & 12 & 9 \\ 6 & 7 & 8 \\ 3 & 4 & 5 \end{matrix}$$

Calculate the gradient direction $\tan^{-1}(g_y/g_x)$ of the pixel in the centre using

A. Prewitt horizontal and vertical operators
B. Sobel operators

Comment briefly on the result.

3. The standard mask for smoothing an image using convolution is

$$\begin{matrix} 1/36 & 1/9 & 1/36 \\ 1/9 & 4/9 & 1/9 \\ 1/36 & 1/9 & 1/36 \end{matrix}$$

which is an approximation to the Gaussian we described in the lectures. What would be the result for convolving this mask to the image pattern shown in Question 1 above? What is the computational cost? Is it possible to make this 2D convolution into multiple 1D convolutions? How? What would be the computational cost?