Edge Filters

Convolution vs correlation

- Convolution comes from filtering frequencies – multiplication in frequency domain is equivalent to convolution in space domain.
- Correlation is basically a template matching – the kernel is the template and the result of correlation measures how well the template correlates with certain part of the image.
- For kernels symmetrical about central element convolution and correlation give the same result.

What is an edge?

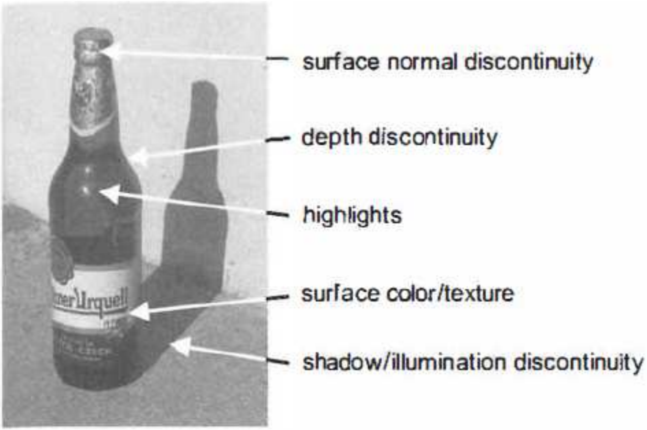

Figure 5.16: Origin of edges, i.e., physical phenomena in the image formation process which lead to edges in images.

Edge detection

- Changes in continuous functions are measured by derivatives
- Image is represented by 2-variable function, changes measured by partial derivatives.
- Change of image function can be described by gradient.

Edge detection

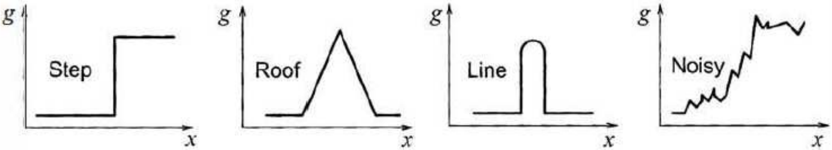

Figure 5.19: Typical edge profiles.

Edge filters

- Edges in an image are areas in the image with large intensity change.
- Edge is a property attached to an individual pixel. This property consists of two values – magnitude and direction
- Magnitude of an edge is the magnitude of the gradient.

$$|\mathrm{grad}\, g(x,y)| = \sqrt{\left(\frac{\partial g}{\partial x}\right)^2 + \left(\frac{\partial g}{\partial y}\right)^2}$$

- Direction of an edge is direction of the gradient minus 90 degrees.
- Edge filter detects the change of intensity in small neighborhood of a given pixel.

Edge filters

Edge filters or gradient operators can be divided into three categories

- Operators approximating derivatives of the image function using differences.
- Operators based on zero-crossings of the image function second derivative.
- Operators which attempts to match an image function to a parametric model of edges

Edge filters - Roberts operator

Kernels

$$h_1 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \qquad h_2 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

Edge filters - Prewitt operator

Kernels

$$h_1 = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix} \qquad h_2 = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}$$

Edge filters - Laplace operator

In continuous domain

$$\nabla^2 g(x,y) = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

Kernels in discrete domain for 4-neighborhood and 8-neighborhood respectively

$$h_4 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \qquad h_8 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 8 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Edge filters - Sobel operator

Kernels

$$h_1 = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \qquad h_2 = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

Kernel constructions

Approximation of spatial derivatives

- Derivative definition

$$\frac{df}{dx} = \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}$$

  suggests kernel $(-1, 1)$.

- Kernel with even number of elements - non-unique center point.

Kernel constructions - heuristic

Approximation of spatial derivatives

- Alternative definition of derivative

$$\frac{df}{dx} = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x}$$

  suggests kernel $(-1, 0, 1)$.

- Susceptible to noise. Averaging in vertical direction gives (allmost) Prewitt kernel.

$$\frac{1}{6}\begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}$$

Kernel constructions - derivation

Problem to solve:

- Given a set of discrete values of image brightness function find tangent plane, i.e. best linear fit

$$f(x, y) = ax + by + c = \mathbf{a}^T\mathbf{x},$$

$\mathbf{a}^T = (a\, b\, c)$, $\mathbf{x}^T = (x\, y\, 1)$

- $\dfrac{\partial f}{\partial x} = a$, $\dfrac{\partial f}{\partial y} = b$, $\mathrm{grad}(f) = (a\, b)$.

Kernel constructions - derivation

For brightness image funcion $g(x,y)$ (i.e. gray-scale image) at some discrete neighborhood $\chi \subset \mathbb{Z} \times \mathbb{Z}$ minimize squared error funcion

$$E = \sum_{\chi}(f(x,y) - g(x,y))^2 = \sum_{\chi}(\mathbf{a}^T\mathbf{x} - g(x,y))^2 =$$

$$\mathbf{a}^T\left(\sum_{\chi}\mathbf{x}\mathbf{x}^T\right)\mathbf{a} + 2\mathbf{a}^T\sum_{\chi}\mathbf{x}g(x,y) + \sum_{\chi}g(x,y)^2.$$

Goal: Find $\mathbf{a}$ such that E is minimal.

Kernel constructions - derivation

Deriving E and setting the derivatives to zero leads to a system

$$2\mathbf{a}^T S - 2 \sum_{\chi} \mathbf{x} g(x, y) = \mathbf{0}.$$

with $S = \sum_{\chi} \mathbf{x}\mathbf{x}^T$. S is called scatter matrix.

Kernel constructions - derivation

Consider 3 by 3 neighborhood around origin so that coordinates x and y take only values 0, 1 and -1. Then

$$S = \begin{pmatrix} 6 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 9 \end{pmatrix}$$

Substituting for S in previous formula and solving for a, b gives the linear approximation of derivatives, e.g. for a

$$a = \sum_{\chi} g(x, y)x$$

Kernel constructions - derivation

$$a = \sum_{\mathcal{X}} g(x, y)x$$

is correlation of g with x coordinates of elements of the neighborhood. These coordinates are given by the matrix

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

Kernel constructions - derivation

Smoothing before edge detection:
$g(x, y)$ – brightness image function,
$h(x, y)$ – 2D gaussian kernel

- Smoothing = filtering by gaussian (symmetric) kernel

$$g * h = g \otimes h$$

- For linear operator D (in our case derivative)

$$D(g \otimes h) = D(h) \otimes g.$$

Kernel constructions - derivation

d-dimensional multivariate Gaussian

$$\frac{1}{(2\pi)^{d/2}|K|^{1/2}} \exp\left(-\frac{[(\mathbf{x}-\mu)]K^{-1}[\mathbf{x}-\mu]}{2}\right)$$

K – covariance matrix,
μ – mean vector.
For $d = 2$ and $K = \mathrm{diag}(\sigma^2, \sigma^2) = \sigma^2 I$

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

Edge filters

Other filters

- Robinson operator
- Kirsch operator

For Sobel operators the gradient magnitude and direction can be computed from filtered images. If x is image filtered by h_1 and y is image filtered by h_2, the gradient magnitude is $\sqrt{x^2 + y^2}$ and direction $\arctan(y/x)$.

Zero-crossing of second derivative

- Step edge corresponds to an abrupt change in the image function.

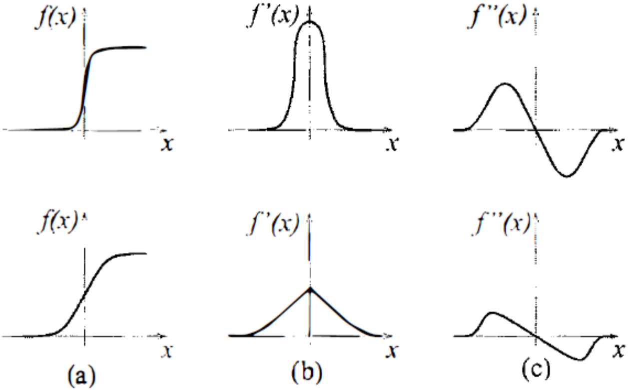

Figure: Sonka, Hlavac, Boyle: Image Processing, Analysis and Machine Vision

Zero-crossing of second derivative

- The first derivative should have an extremum at the position corresponding to the edge.

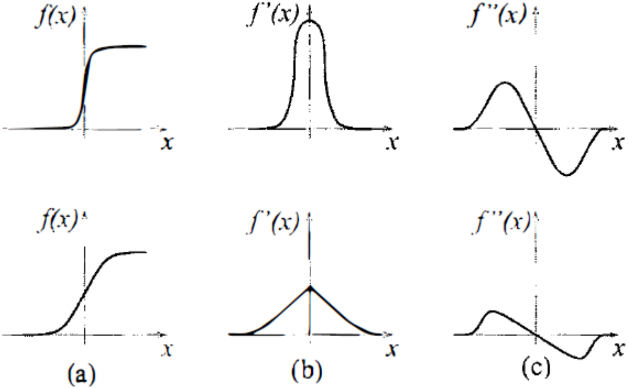

Figure: Sonka, Hlavac, Boyle: Image Processing, Analysis and Machine Vision

Zero-crossing of second derivative

- The second derivative is then zero at the position of the edge

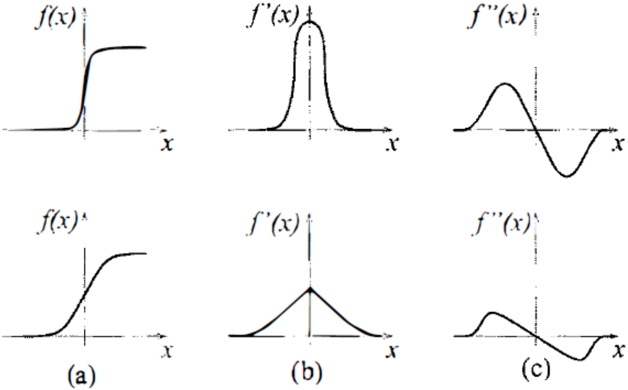

Figure: Sonka, Hlavac, Boyle: Image Processing, Analysis and Machine Vision

Zero-crossing of second derivative

Problem is noise – solution, filtering with Gaussian (there are other reasons for this type of filtering as well).

$$G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}},$$

Second derivative is modeled by Laplacian applied onto image function filtered by Gaussian

$$\nabla^2[G(x, y, \sigma) * f(x, y)]$$

Zero-crossing of second derivative

From convolution properties

$$\nabla^2[G(x,y,\sigma) * f(x,y)] = [\nabla^2 G(x,y,\sigma)] * f(x,y)$$

And expression in brackets on the right side can be computed analytically

$$\nabla^2 G(x,y,\sigma) = \left(\frac{x^2+y^2-\sigma^2}{\sigma^4}\right) e^{-\frac{x^2+y^2}{2\sigma^2}}$$

This is called Laplacian of Gaussian (LoG). This will be the convolution kernel.

Zero-crossing of second derivative

The convolution kernel is

$$\nabla^2 G(x,y,\sigma) = c\left(\frac{x^2+y^2-\sigma^2}{\sigma^4}\right)e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Example of 5×5 LoG discrete convolution kernel.

$$\begin{pmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{pmatrix}$$

Zero-crossing of second derivative

Problems in discrete domain

- There are no real zero crossings of the response function
- Solved by considering sign change in small 2×2 neighborhoods.
- Too many edges.
- Solution: Get rid of those edges for which the first derivative is small.

Edge detector problems

- Too thick edges,
- missing edges,
- extra/false edges.

Canny Edge Detector

- Calculate two partial derivatives f_x, f_y,
- compute magnitude of the gradient $\nabla f = (f_x, f_y)$

$$|\nabla f| = \sqrt{f_x^2 + f_y^2}$$

and direction of the gradient

$$\measuredangle \nabla f = \tan^{-1}\left(\frac{f_y}{f_x}\right)$$

- perform nonmaximal suppression, i.e. find only local maxima of gradient (e.g. thinnig,...). Output is $N(x, y)$.

Canny Edge Detector

- $N(x, y)$ contains one pixel wide, properly located edges,
- dual threshold approach (hysteresis),
- two threshold values $\tau_1 \gg \tau_2$.
- Detect all edges in N with magnitude greater than τ_2 (weak edges)
- If a weak edge is connected with a strong edge (edge in N with magnitude greater than τ_1) it is accepted as an edge or else it is rejected.