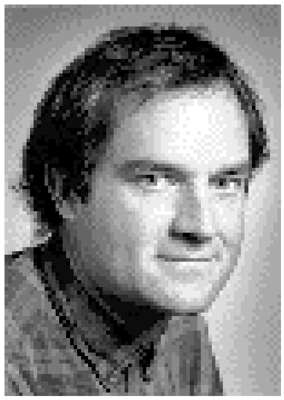
Canny Edge Detector



- Canny (1984) introduces several good ideas to help.
- **References:** Canny, J.F. A computational approach to edge detection. IEEE Trans Pattern Analysis and Machine Intelligence, 8(6): 679-698, Nov 1986.

Canny Edge Detection

- Basic idea is to detect at the zero-crossings of the second directional derivative of the smoothed image
- in the direction of the gradient where the gradient magnitude of the smoothed image being greater than some threshold depending on image statistics.
- It seeks out zero-crossings of

$$\partial^2 (G * I) / \partial n^2 = \partial([\partial G / \partial n] * I) / \partial n$$

n is the direction of the gradient of the smoothed image.

Canny's zero-crossings

- Canny zero-crossings correspond to the first-directional-derivative's maxima and minima in the direction of the gradient.
- Maxima in magnitude reasonable choice for locating edges.

Optimal Edge Detector Design

- Canny derives his filter by optimizing a certain performance index that favors true positive, true negative and accurate localization of detected edges
- Analysis is restricted to linear shift invariant filter that detect unblurred 1D continuous step
- Other justifiable performance criteria are possible and will lead to different filters.

What are Canny's Criteria?

- **Good detection**: low probability of not marking real edge points, and falsely marking non-edge points.

$$SNR = \frac{\left| \int_{-w}^{w} G(-x) f(x) dx \right|}{n_o \sqrt{\int_{-w}^{w} f^2(x) dx}}$$

- *f* is the filter, *G* is the edge signal, denominator is the root-mean-squared response to noise *n(x)* only.

Localization Criterion

- Good localization: close to center of the true edge

$$Localization = \frac{1}{\sqrt{E[x_0^2]}} = \frac{\left| \int_{-w}^{w} G'(-x) f'(x) dx \right|}{n_o \sqrt{\int_{-w}^{w} f'^2(x) dx}}$$

- a measure that increases as localization improves.
- Use reciprocal of the rms distance of the marked edge from the center of the true edge.

Localization Criterion

- The localization criteria equation is a bit hard to understand. The book's description doesn't help either, I think. It is a technical detail that you are not responsible for it. I will put Canny's derivation in the lecture notes for your information.
- The basic intuition: if we assume the filter's response is maximum at the edge when there is no noise, what is the expected distance of the local maximum in the response as we change the filter? The numerator is actually the second derivative of the filtered response, indicating how steep the slope of the zero-crossing of the filtered response is. The steeper is this slope, the sharper is the localization.

Eliminating Multiple Response

- Only one response to a single edge: implicit in first criterion, but make explicit to eliminate multiple response.
- The first two criteria can be trivially maximized by setting $f(x)=G(-x)$*!*
- What is this? This is a truncated step (difference of box operator).
- What is its problem?

Inter-maximum Spacing

- Ideally, want to make the distance between peaks in the noise response approximate the width of the response of the operator to a single step.
- The mean distance between two adjacent maxima in the filtered response (or zero-crossing of their derivatives) can be derived as:

$$x_{zc}(f) = p\left(\frac{\int_{-\infty}^{\infty} f'^2(x)dx}{\int_{-\infty}^{\infty} f''^2(x)dx}\right)^{1/2}$$

- Set this distance a fraction k of the operator width W, Seek f satisfies this constraint with a fixed k. $x_{zc}(f) = kW$

Inter-maximum Spacing

- Again, this is a technical detail that is hard to understand. If you want to understand it, you have to go back to another mathematical result derived for zero-crossing by Rice, “Mathematical anlaysis of random noise” Bell System Techn J. vol 24, pp 46-156, 1945.

Numerical Optimization

- Maximize the first two criteria subject to the multiple response constraint (third criterion) numerically to find the `optimal edge' detector for different kinds of edges:

Roof and Ridge edge detectors close to 2nd derivative of a Gaussian.

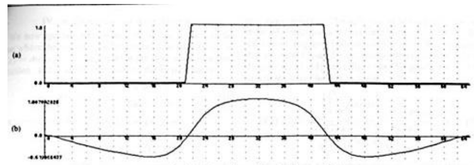


Fig. 2. A ridge profile and the optimal operator for it.

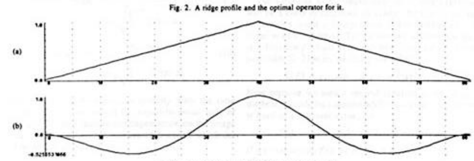


Fig. 3. A roof profile and an optimal operator for roofs.

Optimal Step Edge Detector

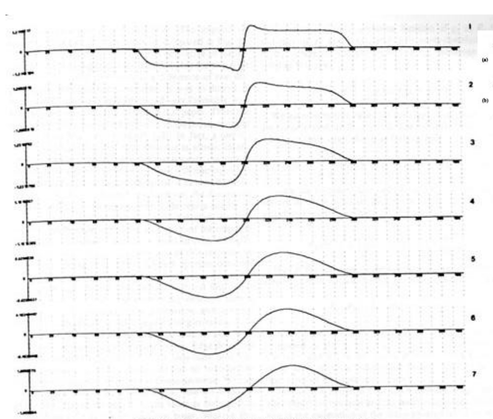


Fig. 5. Optimal step edge operators for various values of x_{max}. From top to bottom, they are x_{max} = 0.15, 0.3, 0.5, 0.8, 1.0, 1.2, 1.4.

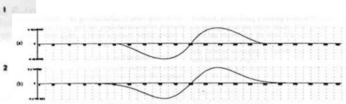


Fig. 6. (a) The optimal step edge operator. (b) The first derivative of a Gaussian.

Interestingly, it turns out to be the First derivative of the Gaussian.

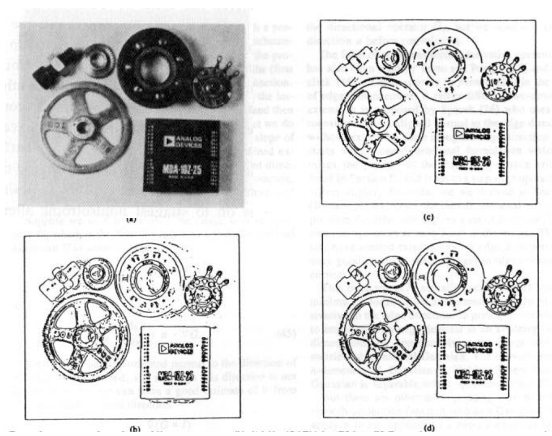
Threshold Determination

- **Adaptive Thresholds**: Use the statistics of the image itself to set the threshold.
- Used the histogram of $\|\nabla(I * G_s)\|^2$, and chose its value at some percentile, e.g. the median, as a reference value of edge strength.
- Set his thresholds as multiple of this value, in fact, not as a number, but as a slowly varying function on a coarse grid.

High and Low Thresholds

- **Hysteresis** method: Use two thresholds.
- The **high threshold** is used to find `seeds' for strong edges.
- Their strength should be large enough so that such an edge cannot be ignored.
- These seeds are grown into as long an edge in both directions as possible, so long as you can do this without the edge strength falling below the **low threshold.**

An Example



(a) original image
(b) threshold at T1
(c) thresholded at 2 T1
(d) image thresholded with hysteresis using both (b) and (c).

Elongated Filters

- A model edge is not just a strong gradient: it is a prolonged contour with a strong perpendicular gradient all along it.
- Better filters for these structures are the anisotropic odd filters, i.e. the odd symmetric simple cells.
- Approximately, the first derivative of an elongated Gaussian:

$$K(s,t) = t \cdot e^{-\frac{s^2+4t^2}{2s^2}}$$

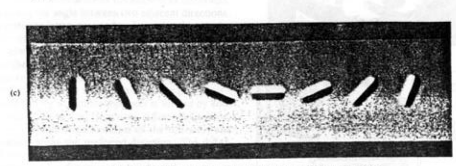


Fig. 10. Directional step edge mask. (a) Cross section parallel to the edge direction. (b) Cross section normal to edge direction. (c) Two-dimensional impulse responses of several masks.

An Example

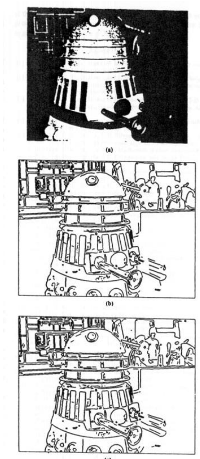


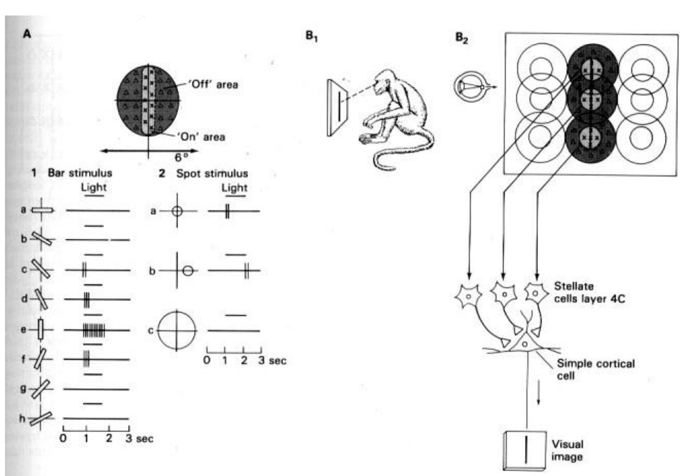
Fig. 11. (a) Dalek image 576 by 454 pixels. (b) Edges found using circular operator. (c) Directional edges (6 mask orientations).

(b) edges found by circular operator.

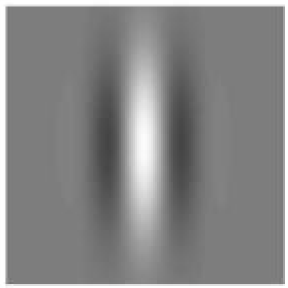
(c) edges found by 6 orientation directional masks.

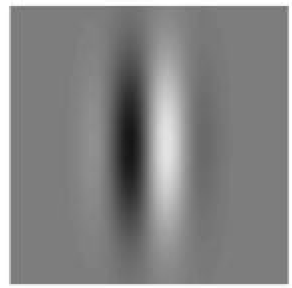
The basic idea is similar to anisotropic diffusion: Gaussian smoothing is modified so that smoothing along contours and do not smooth across contour.

Orientation-selective Simple Cells



Even and Odd Mother Gabor Wavelets

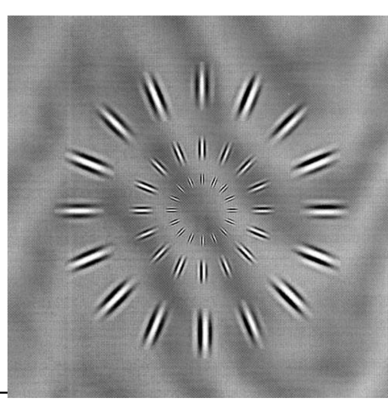


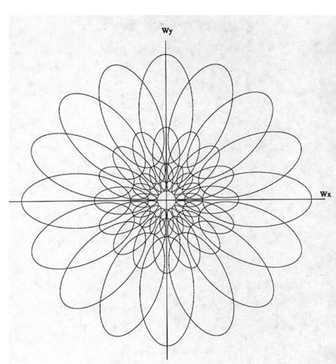


gabor(150,90,1) *gabor*(150,90,0)

Gabor wavelet family

- Members of 1 Gabor wavelet family and their spatial frequency coverage:

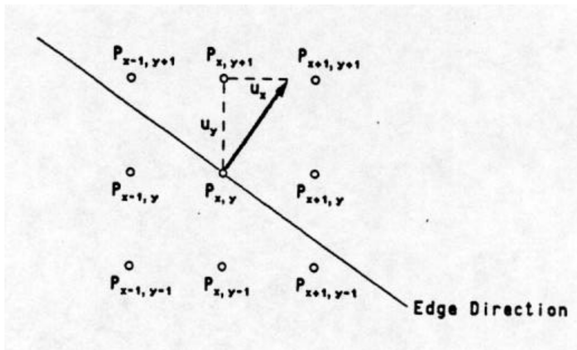




Tiling of spatial frequency plane by Gabor-wavelets

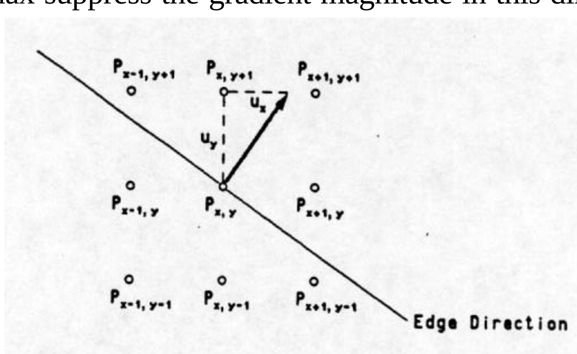
Non-maximum suppression

At each point, compute its edge gradient, compare with the gradients of its neighbors along the gradient direction. If smaller, turn 0; if largest, keep it.



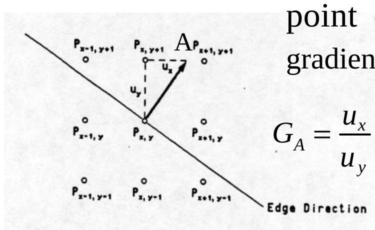
Non-maximum suppression

- The normal to the edge direction, given by arrow, has 2 components u_x and u_y . Use a 9 pixel neighborhood.
- Non-max suppress the gradient magnitude in this direction.



Estimation of Gradient

- Sampling is discrete, how to estimate gradient?
- Pick 2 pts in support closest to u.
- The gradient magnitudes at 3 pts define a plane, use this plane to locally approximate the gradient magnitude surface and to estimate the value at a point on the line. The interpolated gradient magnitude at A, for example, is

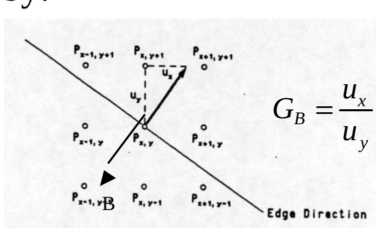


$$G_A = \frac{u_x}{u_y} G(x+1, y+1) + \frac{u_y - u_x}{u_y} G(x, y+1)$$

Note: $G(i, j), u_x, u_y$ are known.

Estimation of Gradient

- The interpolated gradient on the other side is given by:



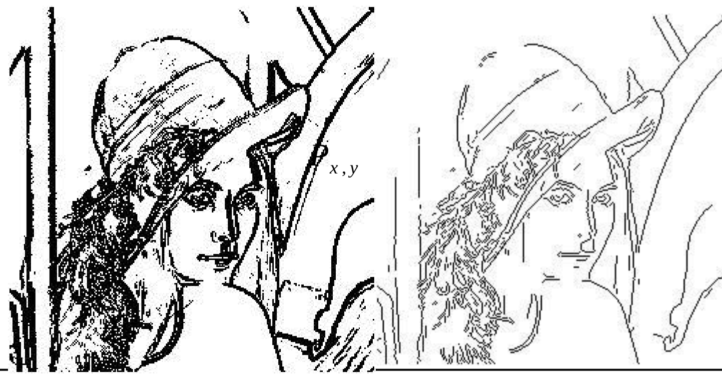
$$G_B = \frac{u_x}{u_y} G(x-1, y-1) + \frac{u_y - u_x}{u_y} G(x, y-1)$$

- Mark $P_{x,y}$ as a maximum if $G(x,y) > G_A$ and $G(x,y) > G_B$
- Interpolation always involve 1 diagonal and 1 non-diagonal point. Avoid division by multiplying through by u_y .

CMU 15-385 Computer Vision   Spring 2002 Tai Sing Lee

Non-maximum suppression

- This scheme involves 4 multiplication per point, but it is not excessive.
- Works better than simpler scheme which compares the points $P_{x,y}$ with two of its neighbors.



Other Edge Operators

Origin: Approximating the intensity landscape with Planar surface, quadratic surface, or bicubic surface, and then take derivatives on this surface.

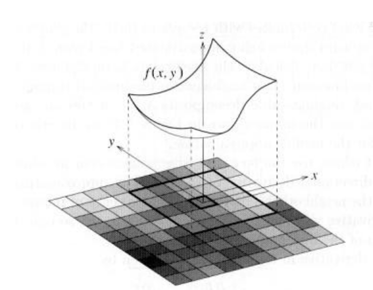
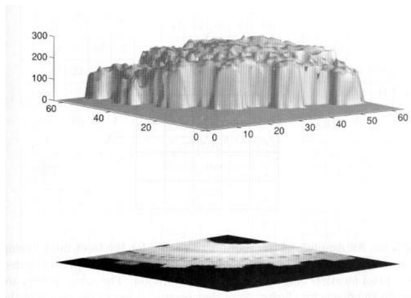
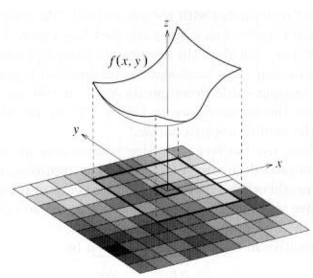


Image intensity surface





$$z = f(x,y) = k_1 + k_2 x + k_3 y + k_4 x^2 + k_5 xy + k_6 y^2 + k_7 x^3 + k_8 x^2 y + k_9 xy^2 + k_{10} y^3$$

Planar surface, quadratic surface, bicubic surface

Mean Square Error Fit

$$z = f(x,y) = k_1 + k_2 x + k_3 y + k_4 x^2 + k_5 xy + k_6 y^2 + k_7 x^3 + k_8 x^2 y + k_9 xy^2 + k_{10} y^3$$

- We can fit the intensity surface with these surfaces by adjusting the parameters k_1 to k_{10} to minimize the Euclidean norm or equivalently the mean square error:

$$E = \sum_u \sum_v (I(u,v) - f(u,v))^2$$

Robert Operator:

- The first simplest gradient operator: Robert's cross operator along diagonal:

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

- Or equivalently,

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

- They are derived to provide the gradient of the least square error planar surface fitted over a 2x2 window.

Sobel Operator

- Derived from

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

- The gradient of a surface smoothed by a mean filter.

Prewitt Operator

- 3x3 Prewitt (1970):

$$\nabla_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad \nabla_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

- 4x4 Prewitt (1970):

$$\nabla_x = \begin{bmatrix} -3 & -1 & 1 & 3 \\ -3 & -1 & 1 & 3 \\ -3 & -1 & 1 & 3 \\ -3 & -1 & 1 & 3 \end{bmatrix} \qquad \nabla_y = \begin{bmatrix} 3 & 3 & 3 & 3 \\ 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \\ -3 & -3 & -3 & -3 \end{bmatrix}$$

Derived by fitting a least square error quadratic surface over a 3x3 image window, then differentiating the fitted surface.

Trade off between SNR and Resolution