Computer Vision

Computer Science Tripos Part II

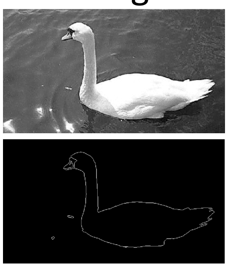
Dr Christopher Town

4. Edge detection operators; the Laplacian and its zero-crossings.

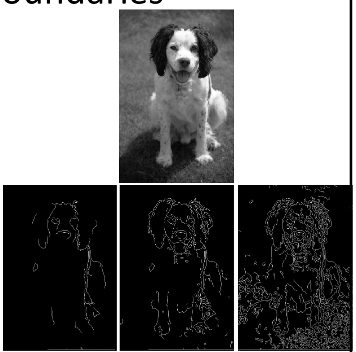
UNIVERSITY OF CAMBRIDGE

- Edges demarcate the boundaries of objects, or of material properties.
- Objects have parts, and these are typically joined with edges.
- The three-dimensional distribution of objects in a scene usually generates occlusions of some objects by other objects, and these form occlusion edges which reveal the geometry of the scene.
- Edges can be generated in more abstract domains than luminance. For example, if some image property such as colour, or a textural signature, or stereoscopic depth, suddenly changes, it forms a highly informative "edge" in that domain.
- Velocity fields, containing information about the trajectories of objects, can be organised and understood by the movements of edges. (The motions of objects in space generates velocity discontinuities at their edges.)
- The central problem of stereoscopic 3D depth vision is the "correspondence problem:" matching up corresponding regions of two images from spatially displaced cameras. Aligning edges is a very effective way to solve the correspondence problem. The same principle applies to measuring velocities (for image frames displaced in time, rather than displaced in space) by tracking edges to align corresponding regions and infer velocity (ratio of object displacement to temporal interval).

Edges vs. Boundaries



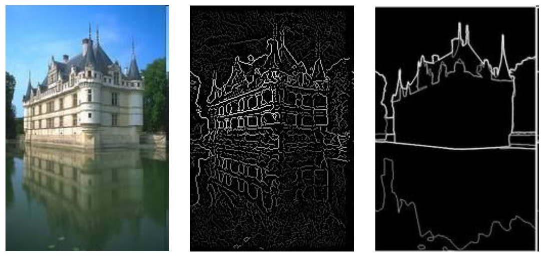
Edges useful signal to indicate occluding boundaries, shape.



…but quite often boundaries of interest are fragmented, and we have extra "clutter" edge points.

Slide credit: K. Grauman

What edges are important?



What causes an edge?

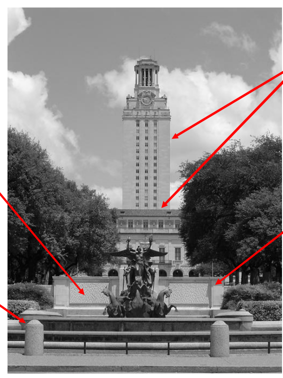
- Depth discontinuity
- Surface orientation discontinuity
- Reflectance discontinuity (i.e., change in surface material properties)
- Illumination discontinuity (e.g., shadow)

Slide credit: Christopher Rasmussen

What can cause an edge?

Reflectance change: surface properties, texture

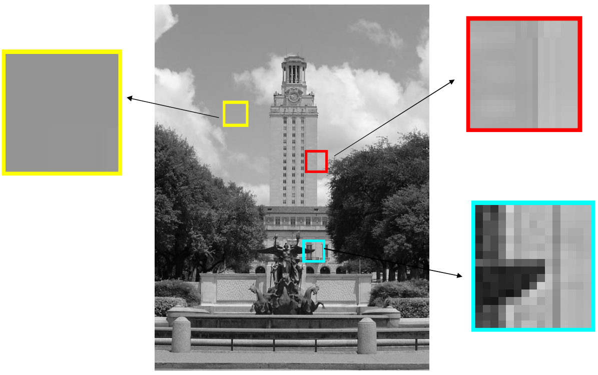
Change in surface orientation: shape



Depth discontinuity: object boundary

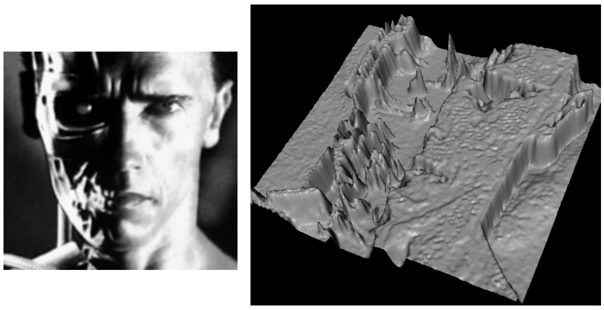
Cast shadows

Contrast and invariance



Dr Chris Town

Recall : Images as functions



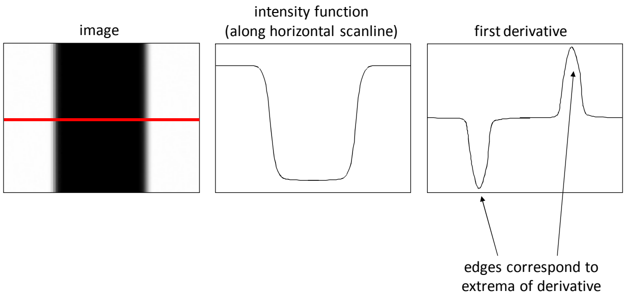
- Edges look like steep cliffs

Source: S. Seitz

Dr Chris Town

Derivatives and edges

An edge is a place of rapid change in the image intensity function.



Source: L. Lazebnik

Dr Chris Town

Differentiation and Convolution

- For the 2D function $f(x,y)$, the partial derivative is:

$$\frac{\partial f(x,y)}{\partial x} = \lim_{\varepsilon \to 0} \frac{f(x+\varepsilon, y) - f(x,y)}{\varepsilon}$$

- For discrete data, we can approximate this using finite differences:

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

- To implement the above as convolution, what would be the associated filter?

Slide credit: K. Grauman

Dr Chris Town

Partial Derivatives of an Image

$$\frac{\partial f(x,y)}{\partial x}$$

-1	1

or

1	-1

$$\frac{\partial f(x,y)}{\partial y}$$

-1
1

or

1
-1

Which shows changes with respect to x?

Slide credit: Kristen Grauman

Dr Chris Town

Differentiation and convolution

- Recall
$$\frac{\partial f}{\partial x} = \lim_{\varepsilon \to 0}\left(\frac{f(x+\varepsilon) - f(x)}{\varepsilon}\right)$$

- This is linear and shift invariant, so can be represented as a convolution.

- We could approximate this as
$$\frac{\partial f}{\partial x} \approx \frac{f(x_{n+1}) - f(x)}{\Delta x}$$

which is a convolution with Kernel

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

Benoit Macq

Dr Chris Town

Finite Difference in 2D

Definition

$$\frac{\partial f(x,y)}{\partial x} = \lim_{\varepsilon \to 0}\left(\frac{f(x+\varepsilon,y)-f(x,y)}{\varepsilon}\right)$$

$$\frac{\partial f(x,y)}{\partial y} = \lim_{\varepsilon \to 0}\left(\frac{f(x,y+\varepsilon)-f(x,y)}{\varepsilon}\right)$$

Convolution Kernels

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

Discrete Approximation

$$\frac{\partial f(x,y)}{\partial x} \approx \frac{f(x_{n+1},y_m)-f(x_n,y_m)}{\Delta x}$$

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

$$\frac{\partial f(x,y)}{\partial y} \approx \frac{f(x_n,y_{m+1})-f(x_n,y_m)}{\Delta x}$$

Benoit Macq

[-1 1]

g[m,n]

$\otimes$ [-1, 1] =

h[m,n]

f[m,n]

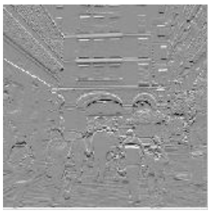
Benoit Macq

[-1 1]^T

g[m,n]

$\otimes$ [-1, 1]^T =

h[m,n]



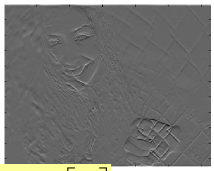
f[m,n]

Benoit Macq

Finite differences

I

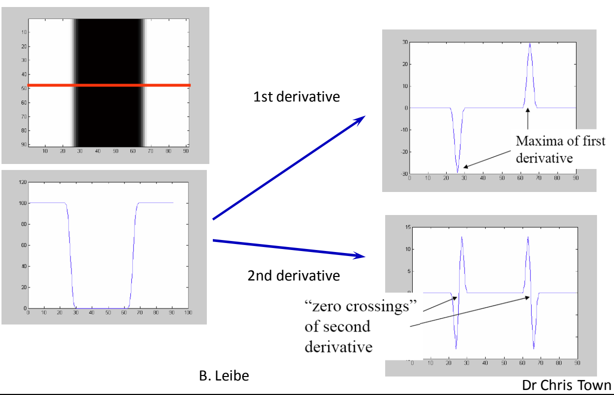
$I_x = I * \begin{bmatrix} 1 & -1 \end{bmatrix}$



$I_y = I * \begin{bmatrix} 1 \\ -1 \end{bmatrix}$

Benoit Macq

Edges and Derivatives…



B. Leibe

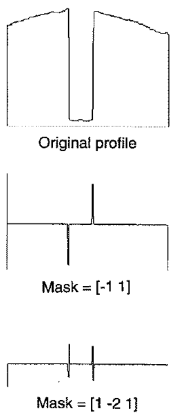
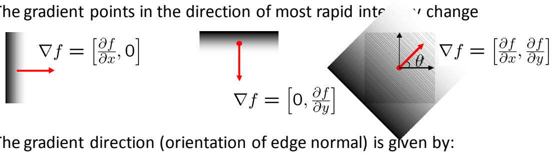


Image Gradient

- The gradient of an image:

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

- The gradient points in the direction of most rapid inte change



- The gradient direction (orientation of edge normal) is given by:

$$\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$$

- The edge strength is given by the gradient magnitude

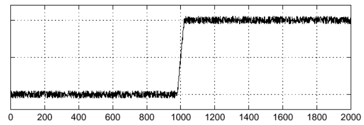
$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Slide credit: Steve Seitz

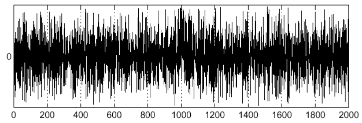
Effect of Noise

- Consider a single row or column of the image
  - Plotting intensity as a function of position gives a signal

$f(x)$



$\frac{d}{dx}f(x)$



Where is the edge?

Slide credit: Steve Seitz

Dr Chris Town

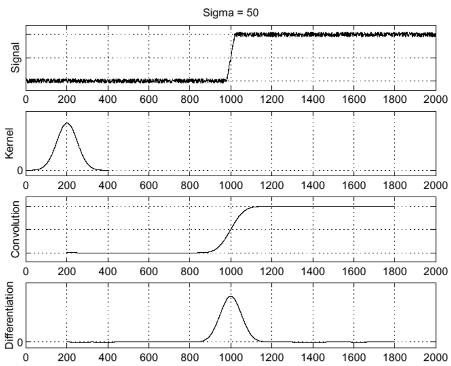
Solution: Smooth First

f

h

$h \star f$

$\frac{\partial}{\partial x}(h \star f)$



Where is the edge? Look for peaks in $\frac{\partial}{\partial x}(h \star f)$

Slide credit: Steve Seitz

Dr Chris Town

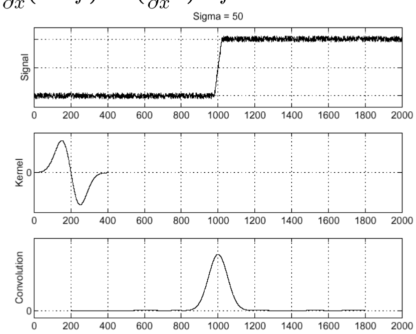
Derivative Theorem of Convolution

$$\frac{\partial}{\partial x}(h \star f) = \left(\frac{\partial}{\partial x}h\right) \star f$$

f

$\frac{\partial}{\partial x}h$

$\left(\frac{\partial}{\partial x}h\right) \star f$



Slide credit: Steve Seitz

Dr Chris Town

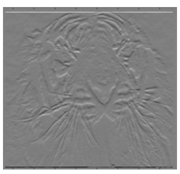
Assorted Finite Difference Filters

Prewitt: $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$; $M_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$

Sobel: $M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$; $\boxed{M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}}$

Roberts: $M_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$; $M_y = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

```
>> My = fspecial('sobel');
>> outim = imfilter(double(im), My);
>> imagesc(outim);
>> colormap gray;
```



Slide credit: K. Grauman

Dr Chris Town

Filter Properties

- Smoothing
  - Values positive
  - Sum to 1 → constant regions same as input
  - Amount of smoothing proportional to mask size
  - Remove "high-frequency" components; "low-pass" filter
- Derivatives
  - Opposite signs used to get high response in regions of high contrast
  - Sum to 0 → no response in constant regions
  - High absolute value at points of high contrast
- Filters act as templates
  - Highest response for regions that "look the most like the filter"

Dr Chris Town

Classical Operators

Prewitt's Operator

Smooth $\begin{bmatrix}1 & 1\\1 & 1\\1 & 1\end{bmatrix}$ → Differentiate $\begin{bmatrix}1 & -1\end{bmatrix}$

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

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

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

Dr Chris Town

Classical Operators

Sobel's Operator

Smooth $\begin{bmatrix}1 & 1\\2 & 2\\1 & 1\end{bmatrix}$ → Differentiate $\begin{bmatrix}1 & -1\end{bmatrix}$

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

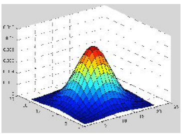
$\begin{bmatrix}1 & 2 & 1\\1 & 2 & 1\end{bmatrix}$ → $\begin{bmatrix}1\\-1\end{bmatrix}$

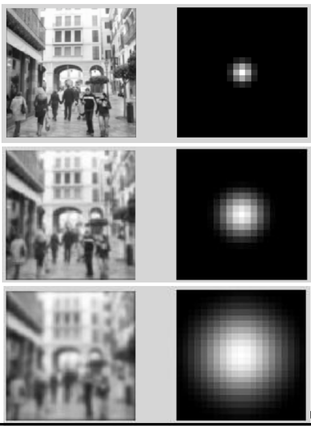
$$\begin{bmatrix}1 & 2 & 1\\0 & 0 & 0\\-1 & -2 & -1\end{bmatrix}$$

Dr Chris Town

Gaussian filter

$$G(x,y;\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$





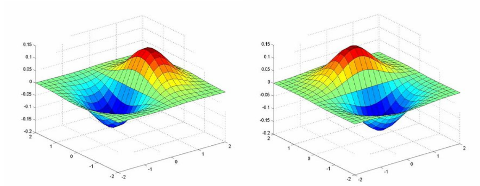
σ=1

σ=2

σ=4

ris Town

Derivative of Gaussian Filters



x-direction

y-direction

Source: Svetlana Lazebnik

Dr Chris Town

Laplacian

$$\nabla = \begin{bmatrix}\frac{\partial}{\partial x_1}\\ \frac{\partial}{\partial x_2}\\ \vdots\\ \frac{\partial}{\partial x_n}\end{bmatrix} \qquad \nabla f(\overline{x}) = \begin{bmatrix}\frac{\partial f(\overline{x})}{\partial x_1}\\ \frac{\partial f(\overline{x})}{\partial x_2}\\ \vdots\\ \frac{\partial f(\overline{x})}{\partial x_n}\end{bmatrix}$$

$$\begin{aligned}\nabla^2 &= \nabla\cdot\nabla\\ &= \textstyle\sum_{i=1}^n \frac{\partial}{\partial x_i}\frac{\partial}{\partial x_i}\\ &= \textstyle\sum_{i=1}^n \frac{\partial^2}{\partial x_i^2}\end{aligned} \qquad \nabla^2 f(\overline{x}) = \sum_{i=1}^{n}\frac{\partial^2 f(\overline{x})}{\partial x_i^2}$$

Dr Chris Town

Laplacian

∇^2 in just a 3 x 3 array is:

-1	-2	-1
-2	12	-2
-1	-2	-1

Other kernels are also sometimes used, e.g.:

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

Dr Chris Town

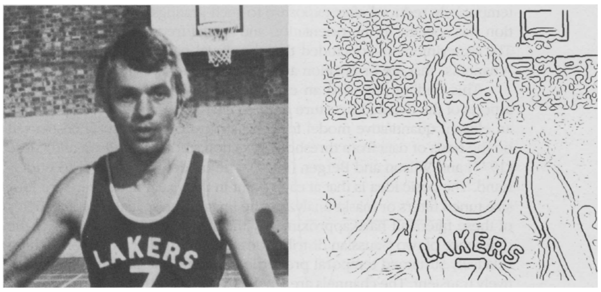


Figure 2 *Illustration of edge-detection by convolution with an isotropic Laplacian operator, and marking the zero-crossings of the result of the convolution.*

Dr Chris Town

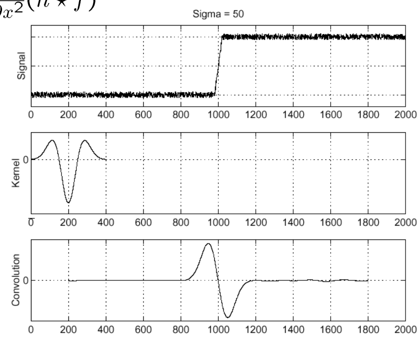
Laplacian of Gaussian (LoG)

- Consider $\frac{\partial^2}{\partial x^2}(h \star f)$

f

$\frac{\partial^2}{\partial x^2}h$

$(\frac{\partial^2}{\partial x^2}h) \star f$

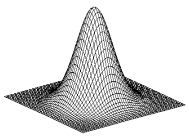


Where is the edge? Zero-crossings of bottom graph

Slide credit: Steve Seitz

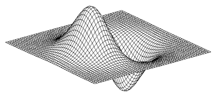
Dr Chris Town

2D Edge Detection Filters



Gaussian

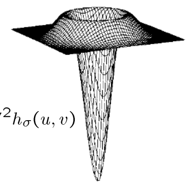
$$h_\sigma(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$



Derivative of Gaussian

$$\frac{\partial}{\partial x} h_\sigma(u,v)$$

Laplacian of Gaussian



$$\nabla^2 h_\sigma(u,v)$$

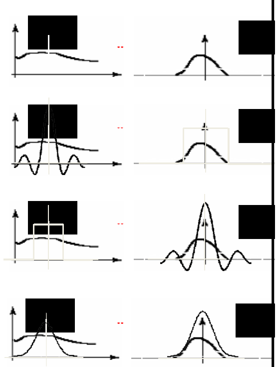
$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Slide credit: K. Grauman

Dr Chris Town

Effect of Filtering

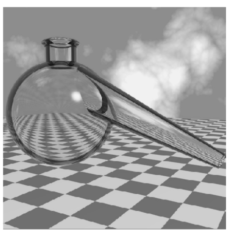
- Noise introduces high frequencies. To remove them, we want to apply a "low-pass" filter.
- The ideal filter shape in the frequency domain would be a box. But this transfers to a spatial sinc, which has infinite spatial support.
- A compact spatial box filter transfers to a frequency sinc, which creates artifacts.
- A Gaussian has compact support in both domains. This makes it a convenient choice for a low-pass filter.



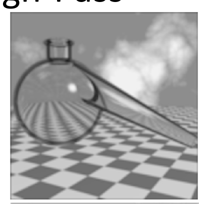
B. Leibe

Dr Chris Town

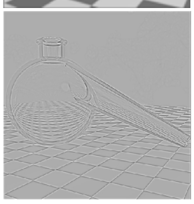
Low-Pass vs. High-Pass



Original image



Low-pass filtered



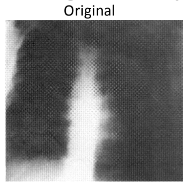
High-pass filtered

Image Source: S. Chenney

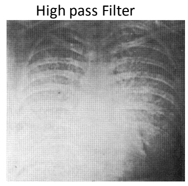
Dr Chris Town

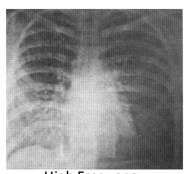
High Frequency Enhancement

Original

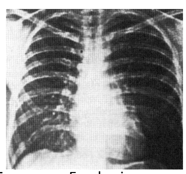


High pass Filter





High Frequency Emphasis



High Frequency Emphasis + Histogram Equalization

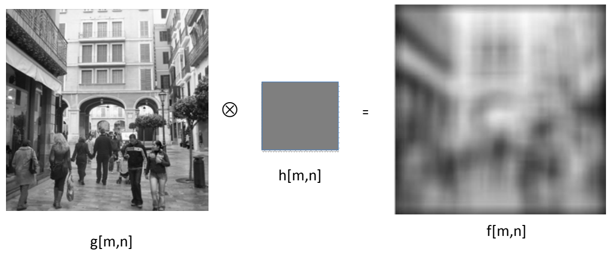
Slide credit: Michal Irani

Dr Chris Town

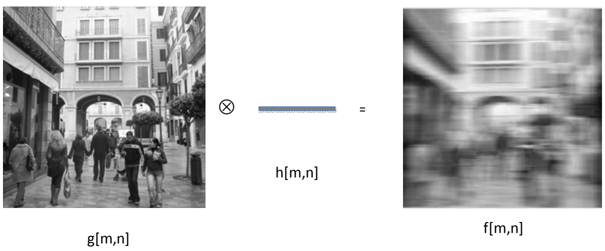
Properties of image filters

- isotropic (circularly symmetric), or anisotropic (directional)
- self-similar (dilates of each other), or not self-similar
- separable (expressible as product of two 1D functions), or not. Convolving with a filter kernel that is separable is the same as convolving with two 1D kernels, one in the x-direction and another in the y-direction.
- degree of conjoint uncertainty (i.e. minimal dispersion, or variance) in the information resolved
- size of support (dimensionality of the kernel)
- preferred non-linear outputs (zero-crossings; phasor moduli; energy)
- theoretical foundations (e.g. Logan's Theorem)

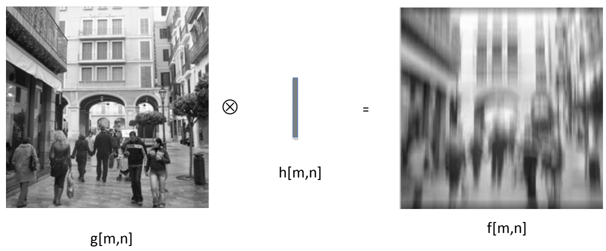
Rectangular filter



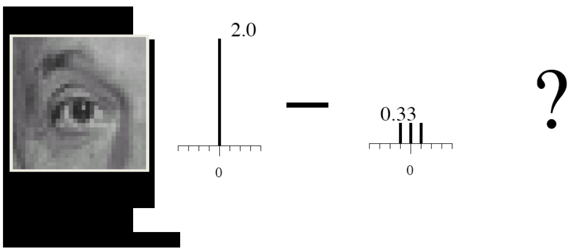
Rectangular filter



Rectangular filter

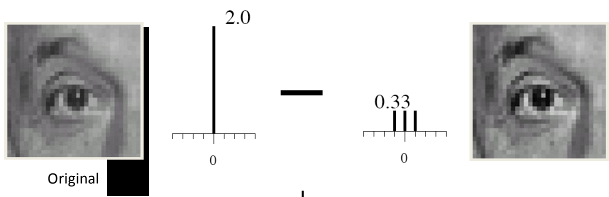


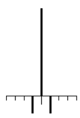
Quiz: What Effect Does This Filter Have?



Source: D. Lowe

Sharpening Filter

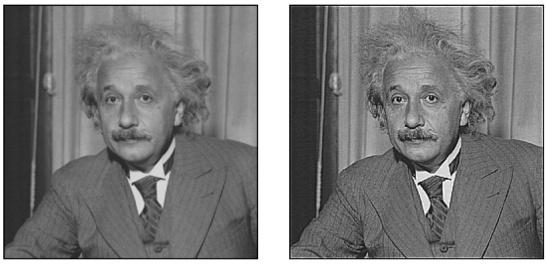




Sharpening filter
– Accentuates differences with local average

Source: D. Lowe

Sharpening Filter



before **after**

Source: D. Lowe

Dr Chris Town

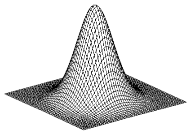
The Laplacian $\nabla^2 G_\sigma(x,y) * I(x,y)$ and its zero-crossings. Logan's Theorem.

$$\nabla^2 \equiv \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)$$

$$\nabla^2 f(x,y) \equiv \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) f(x,y) \overset{2DFT}{\Longrightarrow} -(\mu^2+\nu^2)F(\mu,\nu)$$

$$\nabla^2 \left[G_\sigma(x,y) * I(x,y)\right]$$

$$G_\sigma(x,y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-x^2/2\sigma^2} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-y^2/2\sigma^2} = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

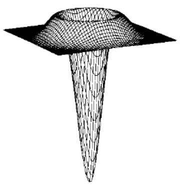


Dr Chris Town

The Laplacian $\nabla^2 G_\sigma(x,y) * I(x,y)$ and its zero-crossings. Logan's Theorem.

$$\nabla^2 \left[G_\sigma(x,y) * I(x,y)\right] = G_\sigma(x,y) * \nabla^2 I(x,y)$$

Laplacian of Gaussian



$$\nabla^2 \left[G_\sigma(x,y) * I(x,y)\right]$$

$$G_\sigma(x,y) * \nabla^2 I(x,y)$$

$$\left[\nabla^2 G_\sigma(x,y)\right] * I(x,y)$$

$$\nabla^2 G_\sigma(x,y) = \frac{\partial^2}{\partial x^2} G_\sigma(x,y) + \frac{\partial^2}{\partial y^2} G_\sigma(x,y) = \frac{x^2+y^2-2\sigma^2}{2\pi\sigma^6} e^{-(x^2+y^2)/2\sigma^2}$$

Dr Chris Town

<u>**Logan's Theorem:**</u> If a signal $f(x)$ is strictly bandlimited to one octave or less, so that the highest frequency component it contains is no greater than twice the lowest frequency component it contains

$$k_{max} \leq 2k_{min}$$

i.e. $F(k)$ the Fourier Transform of $f(x)$ obeys

$$F(|k| > k_{max} = 2k_{min}) = 0$$

and

$$F(|k| < k_{min}) = 0$$

and if it is also true that the signal $f(x)$ contains no complex zeroes in common with its Hilbert Transform (too complicated to explain here, but this constraint serves to exclude families of signals which are merely amplitude-modulated versions of each other), then the original signal $f(x)$ can be perfectly recovered (up to an amplitude scale constant) merely from knowledge of the set $\{x_i\}$ of <u>zero-crossings</u> of $f(x)$ alone:

Dr Chris Town

(1) This is a very complicated, surprising, and recent result (W F Logan, 1977).

(2) Only an existence theorem has been proven. There is so far no stable constructive algorithm for actually making this work – i.e. no known procedure that can actually recover $f(x)$ in all cases, within a scale factor, from the mere knowledge of its zero-crossings $f(x) = 0$; only the existence of such algorithms is proven.

(3) The "Hilbert Transform" constraint (where the Hilbert Transform of a signal is obtained by convolving it with a hyperbola, $h(x) = 1/x$, or equivalently by shifting the phase of the positive frequency components of the signal $f(x)$ by $+\pi/2$ and shifting the phase of its negative frequency components by $-\pi/2$), serves to exclude ensembles of signals such as $a(x)\sin(\omega x)$ where $a(x)$ is a purely positive function $a(x) > 0$. Clearly $a(x)$ modulates the amplitudes of such signals, but it could not change any of their zero-crossings, which would always still occur at $x = 0, \frac{\pi}{\omega}, \frac{2\pi}{\omega}, \frac{3\pi}{\omega}, ...,$ and so such signals could not be uniquely represented by their zero-crossings.

Dr Chris Town

(4) It is very difficult to see how to generalize Logan's Theorem to two-dimensional signals (such as images). In part this is because the zero-crossings of two-dimensional functions are non-denumerable (uncountable): they form continuous "snakes," rather than a discrete and countable set of points. Also, it is not clear whether the one-octave bandlimiting constraint should be isotropic (the same in all directions), in which case the projection of the signal's spectrum onto either frequency axis is really low-pass rather than bandpass; or anisotropic, in which case the projection onto both frequency axes may be strictly bandpass but the different directions are treated differently.

(5) Logan's Theorem has been proposed as a significant part of a "brain theory" by David Marr and Tomaso Poggio, for how the brain's visual cortex processes and interprets retinal image information. The zero-crossings of bandpass-filtered retinal images constitute <u>edge information</u> within the image.

Dr Chris Town

The resulting bandwidth of a $\nabla^2 G_\sigma(x, y)$ filter is about 1.3 octaves, regardless of what value for scale parameter σ is used. Note that this doesn't *quite* satisfy the first constraint of Logan's Theorem.

As a practical matter, the $\nabla^2 G_\sigma(x, y) * I(x, y)$ approach to edge extraction tends to be very noise-sensitive. Many spurious edge contours appear that shouldn't be there. This defect inspired the development of more sophisticated non-linear edge detectors, such as Canny's, which estimates the local image signal-to-noise ratio (SNR) to adaptively optimise its local bandwidth. This, however, is very computationally expensive.

Finally, strong claims were originally made that $\nabla^2 G_\sigma(x, y) * I(x, y)$ edge-detecting filters describe how human vision works. In particular, the receptive field profiles of retinal ganglion cells were said to have this form.

Comparing Human and Machine Perception

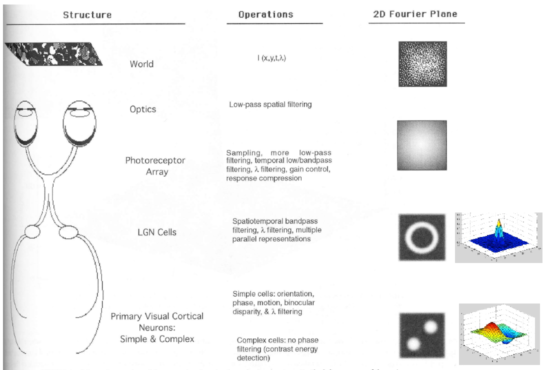
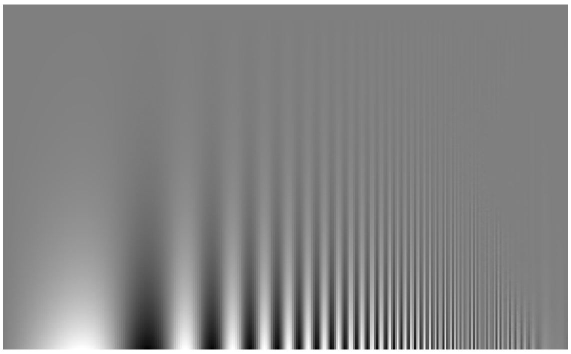


FIGURE 1 Schematic overview of the processing done by the early visual system. On the left, are some of the major structures to be discussed; in the middle, are some of the major operations done at the associated structure; in the right, are the 2-D Fourier representations of the world, retinal image, and sensitivities typical of a ganglion and cortical cell.

Contrast Sensitivity Function



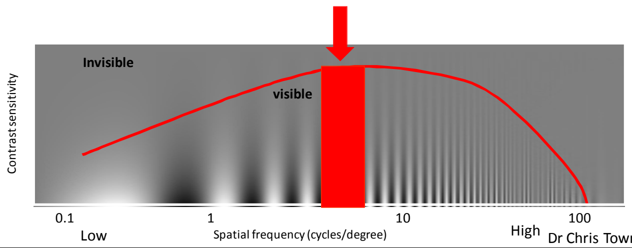
A demo of human contrast sensitivity as a function of spatial frequency. Frequency rises from left to right at a constant rate. Contrast drops from bottom to top at a constant rate. The bars are visible further up for middle frequencies, showing these are more salient to the human visual system.

Contrast Sensitivity Function

Blackmore & Campbell (1969)

Maximum sensitivity

~ **6** cycles / degree of visual angle



Gradients -> edges

Primary edge detection steps:

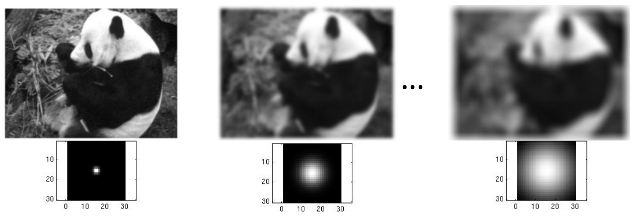
1. Smoothing: suppress noise
2. Edge enhancement: filter for contrast
3. Edge localization

   Determine which local maxima from filter output are actually edges vs. noise

   - Threshold, Thin

Smoothing with a Gaussian

Recall: parameter σ is the "scale" / "width" / "spread" of the Gaussian kernel, and controls the amount of smoothing.



Effect of σ on derivatives

σ = 1 pixel σ = 3 pixels

The apparent structures differ depending on Gaussian's scale parameter.

Larger values: larger scale edges detected
Smaller values: finer features detected

Dr Chris Town

So, what scale to choose?

It depends what we're looking for.

Too fine of a scale…can't see the forest for the trees.
Too coarse of a scale…can't tell the maple grain from the cherry.

Dr Chris Town

Original image

Town

Gradient magnitude image

Town

Thresholding gradient with a lower threshold

Town

Thresholding gradient with a higher threshold

Town

Canny edge detector

- Filter image with derivative of Gaussian
- Find magnitude and orientation of gradient
- **Non-maximum suppression**:
  - Thin multi-pixel wide “ridges” down to single pixel width
- Linking and thresholding (**hysteresis**):
  - Define two thresholds: low and high
  - Use the high threshold to start edge curves and the low threshold to continue them

- MATLAB: `edge(image, 'canny');`
- `>>help edge`

Source: D. Lowe, L. Fei-Fei

Dr Chris Town

The Canny edge detector

original image (Lena)

Dr Chris Town

The Canny edge detector

norm of the gradient

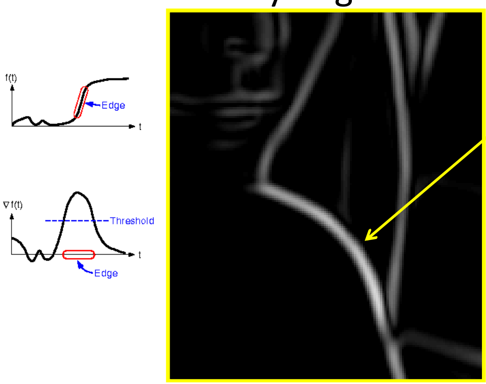
Dr Chris Town

The Canny edge detector

thresholding

Dr Chris Town

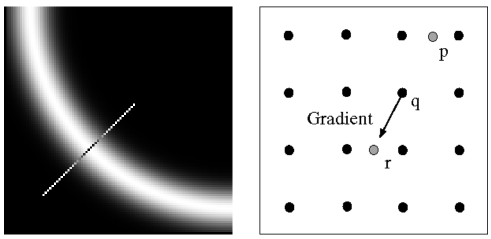
The Canny edge detector



How to turn these thick regions of the gradient into curves?

Dr Chris Town

Non-maximum suppression



Check if pixel is local maximum along gradient direction, select single max across width of the edge
- requires checking interpolated pixels p and r

Dr Chris Town

The Canny edge detector

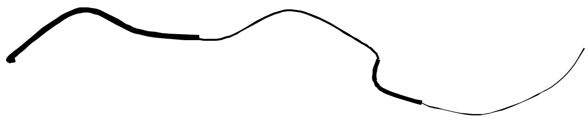
Problem: pixels along this edge didn't survive the thresholding

thinning
(non-maximum suppression)

Dr Chris Town

Hysteresis thresholding

- Check that maximum value of gradient value is sufficiently large
  - drop-outs? use **hysteresis**
    - use a high threshold to start edge curves and a low threshold to continue them.



Source: S. Seitz

Dr Chris Town

Hysteresis thresholding

original image

high threshold
(strong edges)

low threshold
(weak edges)

hysteresis threshold

Source: L. Fei-Fei

Dr Chris Town

Edge detection is just the beginning…

image | human segmentation | gradient magnitude

Berkeley segmentation database:
http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/

Source: L. Lazebnik

Dr Chris Town