EDGE DETECTION

- Edges are important: human visual system can recognise objects from crude outlines

- Object boundaries show up as intensity discontinuities in an image

- A local edge is characterised by high -frequency components: local grey levels are changing rapidly in a small area of the image

Edge detection in a mammalian visual system

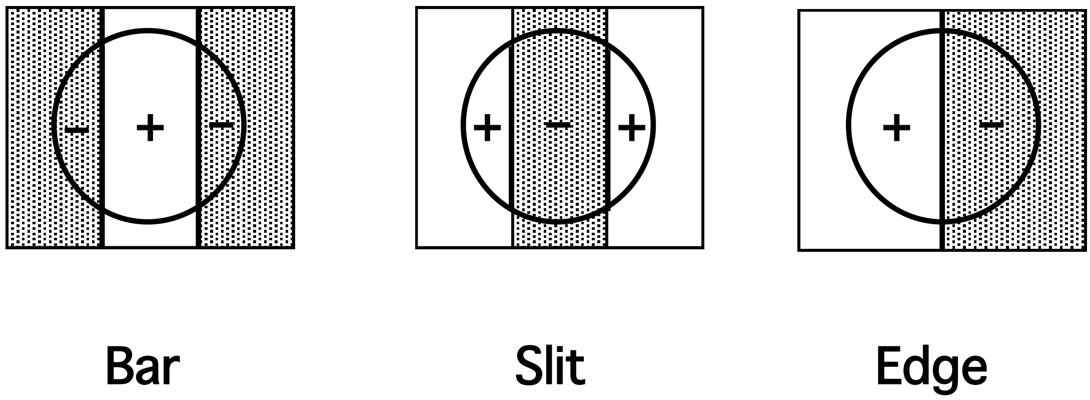
Experiments by Hubel & Weisel (59-68)

- cells in the visual cortex have definite receptive fields, i.e. each cell will respond only to light falling within a certain area of the retina

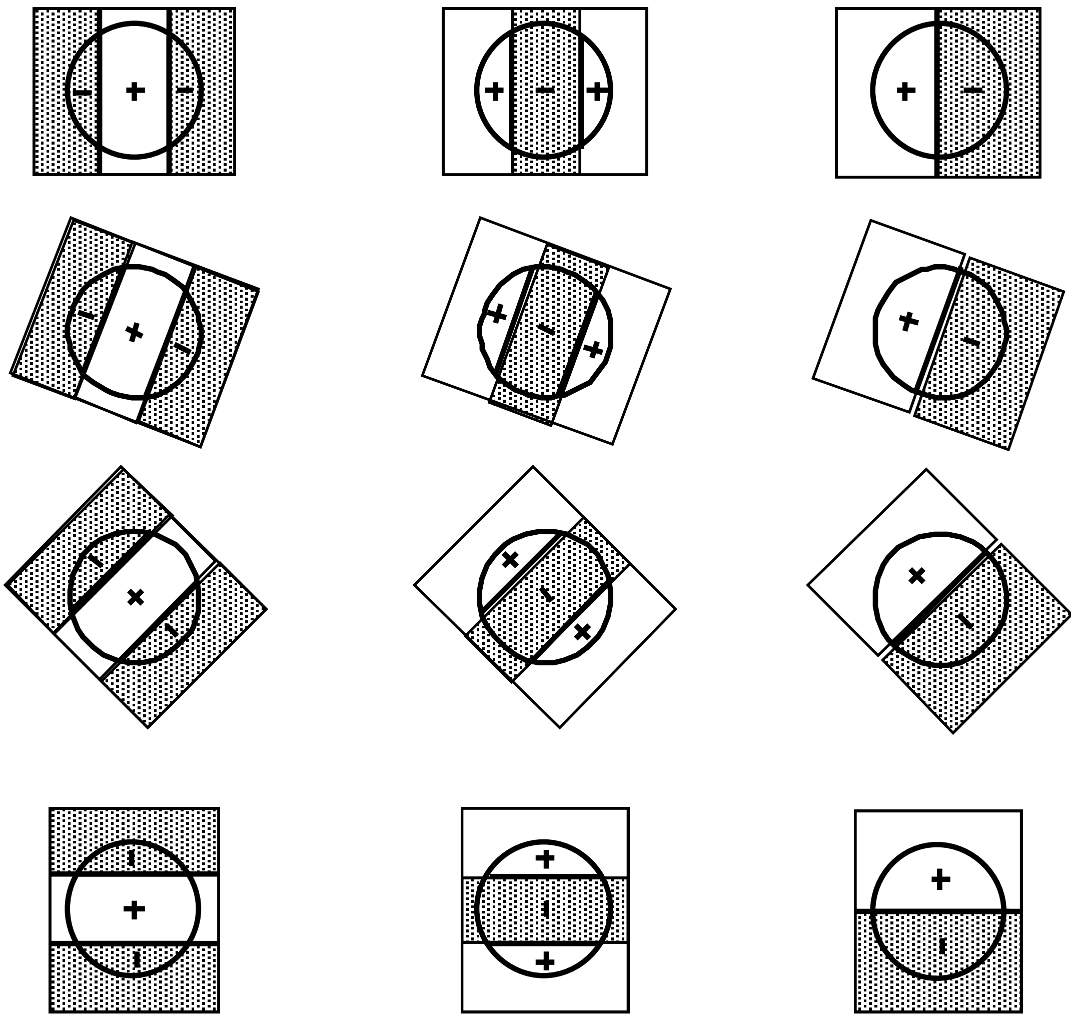
- the activation of a cell is dependant on the type and orientation of the stimuli falling within its receptive field

- “simple cells” specialise in detecting the following basic **patterns** of light and dark:

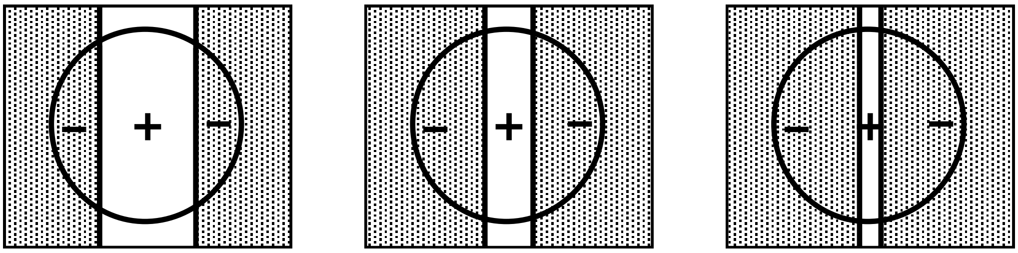
  - edge from light to dark
  - edge from dark to light
  - central excitory/inhibitory region flanked by two parallel regions of the opposite type



- orientational selectivity (10 - 20 degrees)

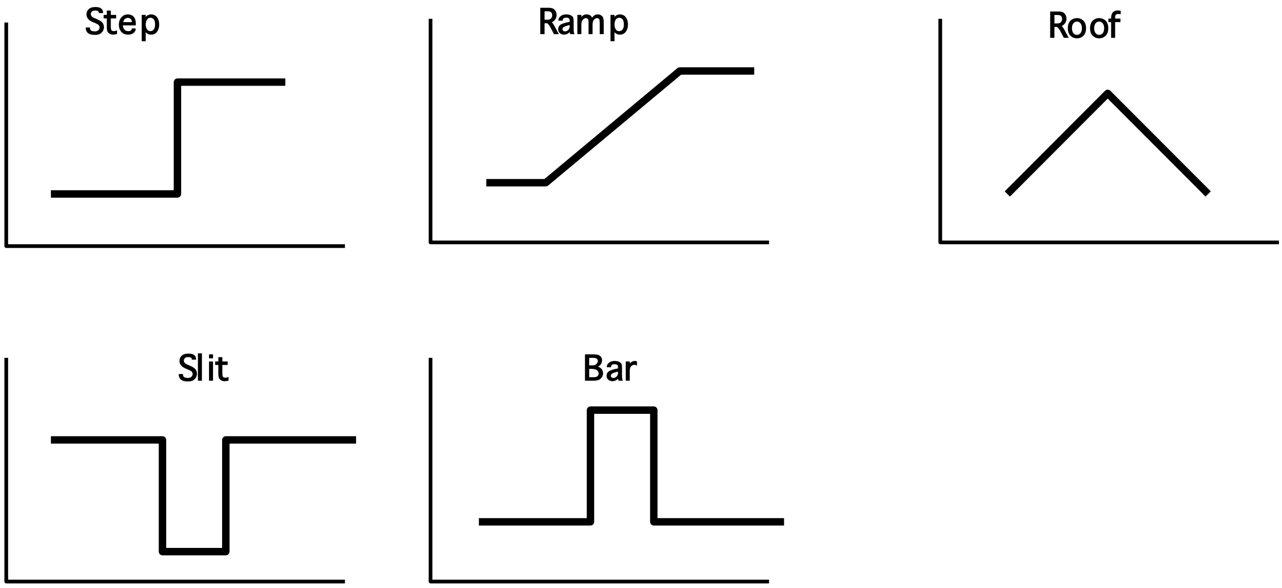


• spatial frequency tuning



Local edge detection by spatial filtering

Edge models



- Computational mechanism for edge detection: convolution

- Different types of edges may require different convolution kernels (edge operators)

- Most edge operators compute:

  direction - aligned with the direction of maximum change in grey level

  magnitude - measure of that change (in terms of grey level values)

Difference operator

Example

$$O(x,y) = I(x+1,y) - I(x,y)$$

- Extracts vertical, horizontal and diagonal edges
- Implemented as convolution of the image with kernels of the form

0	0	0
-1	1	0
0	0	0

Vertical

0	-1	0
0	1	0
0	0	0

Horizontal

-1	0	0
0	1	0
0	0	0

Vertical & horizontal

... e.t.c.

Difference operators:

- sensitive to noise
- highlight only dark-to-light edges in a single direction
- light-to-dark edges yield negative values and are normally set to 0 (black)

Gradient operator

directional

- Up to 8 gradient images can be generated
- Each enhances edges in one particular direction, e.g.

$$\frac{dI}{dx} = \frac{I(x + \Delta x, y) - I(x,y)}{\Delta x}$$

$$\frac{dI}{dy} = \frac{I(x, y + \Delta y) - I(x,y)}{\Delta y}$$

$$\varphi = \max_{(x,y)} \left(\tan^{-1} \left(\frac{dI}{dx} \Big/ \frac{dI}{dy} \right) \right)$$

Gradient operator is a filter which:

- attenuates completely low frequency components
- passes “through” high frequency components
- constant brightness sequences turned into 0

Output value is a measure of edge strength, i.e. difference between neighbouring intensities in a particular direction

Examples of operators

Sobel

-1/4	0	1/4
-1/2	0	1/2
-1/4	0	1/4

-1/4	-1/2	-1/4
0	0	0
1/4	1/2	1/4

Roberts

0	0	0
-1	0	1
0	0	0

0	1	0
0	0	0
0	-1	0

divide by 2

0	0	1
0	0	0
-1	0	0

1	0	0
0	0	0
0	0	-1

divide by ¦2

Properties

- negative differences turned into 0 (or inverted)
- edges can be selectively analysed in a particular direction
- two outputs possible: edge gradient and edge direction
- the above gradient operators are non-linear filters

Laplacian operator

isotropic

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

- The discrete approximation of the Laplacian is the sum of the second spatial derivatives.

Laplacian

0	1	0
1	- 4	1
0	1	0

- Laplacian is an omnidirectional operator, i.e. enhances edges regardless of their orientation
- attenuates low frequency components
- passes through high frequency components
- constant brightness and linearly changing brightness sequences turned into 0

Properties

- highlights both positive and negative differences
- highlighting is sharper than for gradient operators
- does not provide information about orientation
- doubly enhances noise in the image

Smoothing and Laplacian

The edge detection filters are sensitive to noise

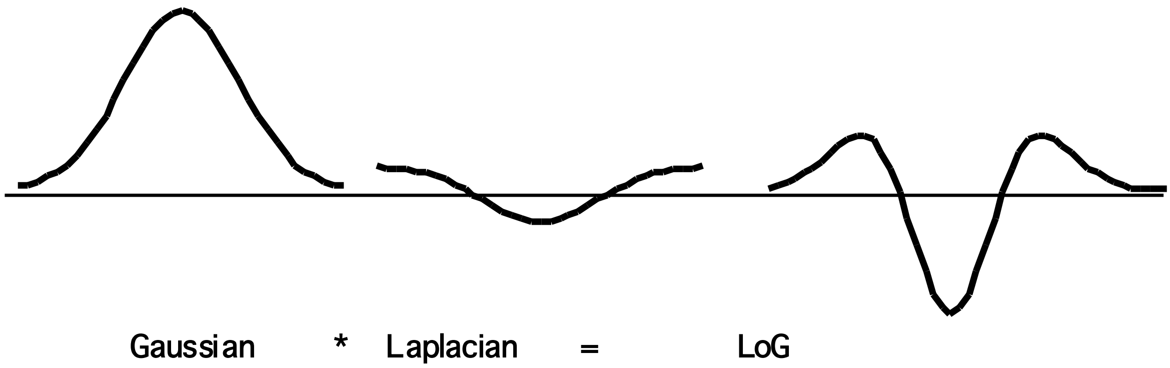
Laplacian-of-Gaussian (**LoG**) operator

- smooths image with Gaussian to select a desired frequency range
- applies Laplacian to detect edges

The size of Gaussian determines spatial scale of the edges:

- small Gaussian (little smoothing) will enable the detection of fine edges

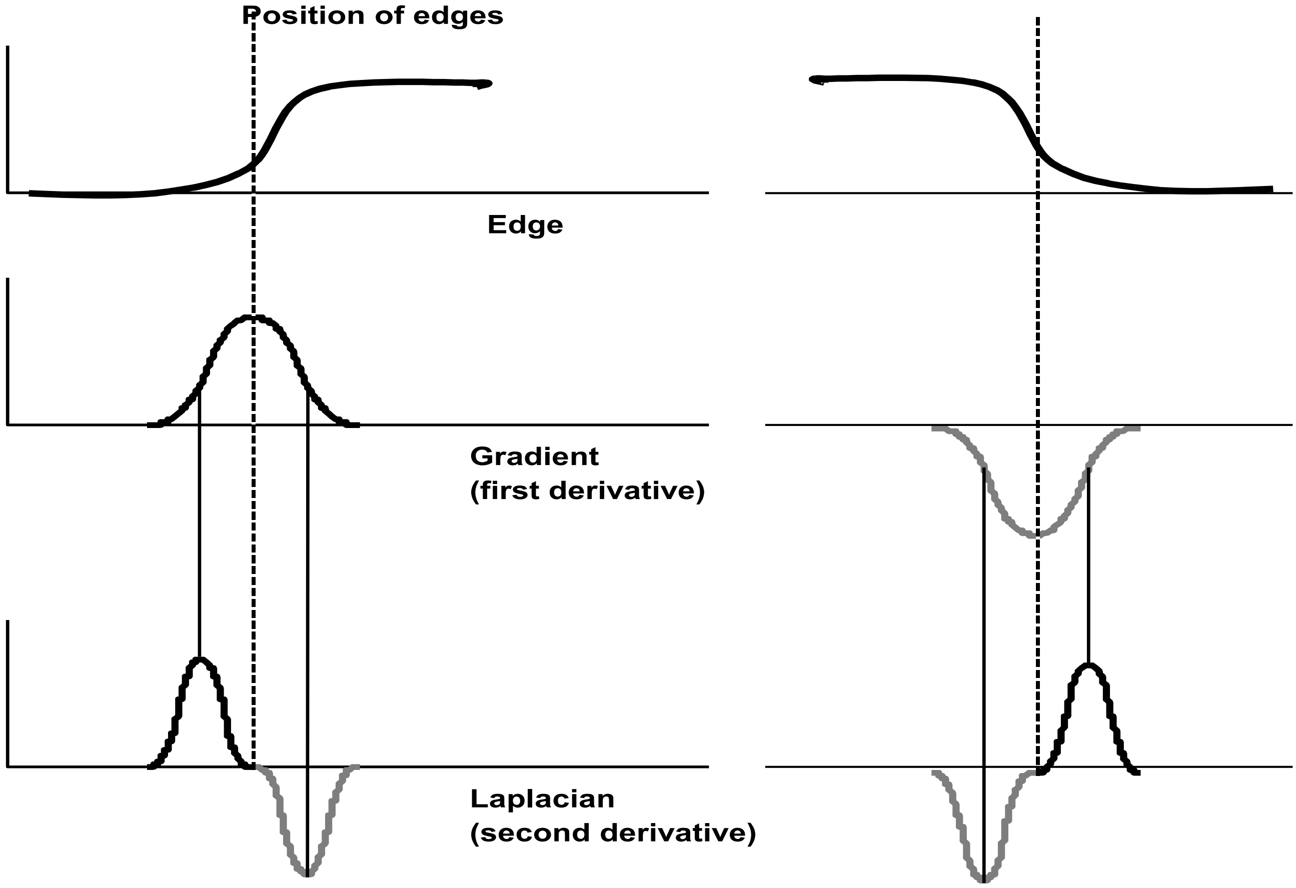
- large Gaussian (severe smoothing) will help to detect coarse edges



Zero crossings of second spatial derivatives

- Problems with maxima of first and second derivatives
- Second derivative has zero crossing at the spatial location corresponding to the 'midpoint' of the edge
- This occurs in all directions except parallel to the edge; this provides means to determine edge orientation.

Zero crossings of second spatial derivatives



Multi-level (hierarchical) analysis

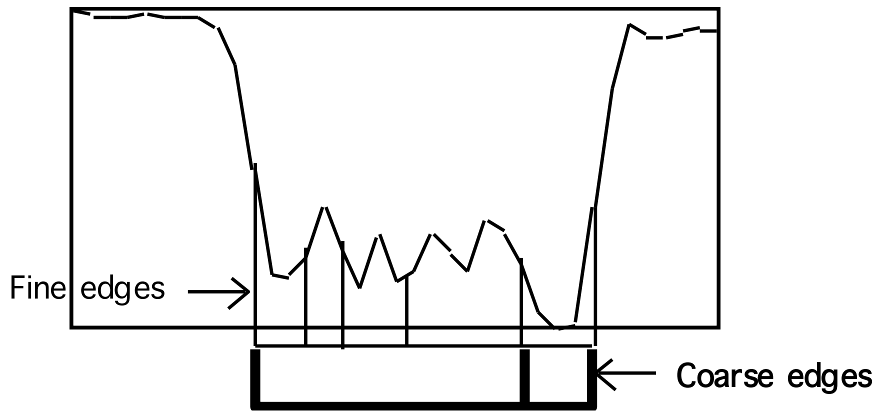
- Human visual system works on different levels of resolution.

- Normally processing proceeds from a coarse level to fine level. Such approach is computationally efficient.

- Processing involves the creation of a sequence of images at various spatial resolutions, spatially related to each other.

- A <u>scale-space</u> representation of an image may be formed by convolving the image with a sequence of filters containing a scale parameter.

- A consistent scale-space filter has a property that going from low to high resolution images existing features never disappear, but new ones may appear.



Spatially related sequence of images at increasing resolution can be thought of as a pyramid, with the original (high resolution) image at the bottom of the pyramid.

Canny edge detector

Design criteria

- shows only“important” edges
- edges are precisely localised
- a single response for a single edge

The steps of Canny edge detector algorithm:

- Convolve image $I(x,y)$ with a Gaussian function $G(x,y,\sigma)$, to obtain smoothed image

  $$S(x,y) = I(x,y) * G(x,y,\sigma)$$

  The degree of smoothing depends on parameter σ

- For each pixel in the image find the direction of the largest gradient:

$$G(x,y,\varphi) = \partial S(x, y) / \partial \varphi$$

$$\varphi (x,y) = \max(\tan^{-1}(G(x,y)))$$

and, for this direction, store the value of the gradient:

$$G(x,y) = \max(G(x,y,\varphi))$$

- Find accurate location of edges using the <u>maximum suppression</u>: -

  - for each pixel examine gradient values in direction of the gradient φ within a certain small neighbourhood; set to 0 all non-maximum values in that neighbourhood.

- Threshold the remaining edges so that only the strongest edges are retained.

Further use of edge images

- A parametric image shows <u>local</u> grey level discontinuities

- The image of edges can be used by high-level algorithms for feature extraction and/or object recognition.

- Edges can be interpreted by routines using higher level information (e.g. a model) to form object boundaries.