Institute of Informatics
Eötvös Loránd University
Budapest, Hungary

Basic Algorithms for Digital Image Analysis: a course

Dmitrij Csetverikov

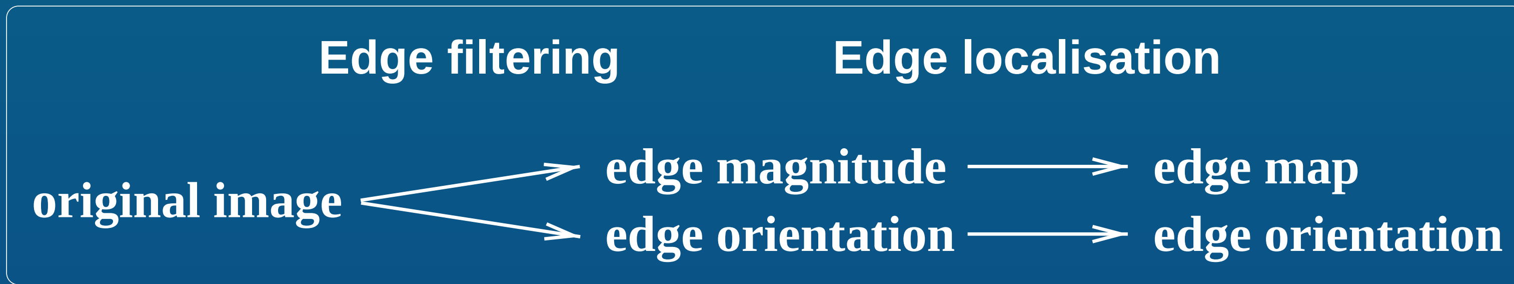
with help of Attila Lerch, Judit Verestóy, Zoltán Megyesi, Zsolt Jankó

http://visual.ipan.sztaki.hu

Lecture 7: Edge detection

- Principles of edge detection
- Criteria for good edge filters
- Gradient edge filters
- Canny edge detector
- Edge localisation:
  - Non-maxima suppression
  - Hysteresis thresholding
- Zero-crossing edge detector

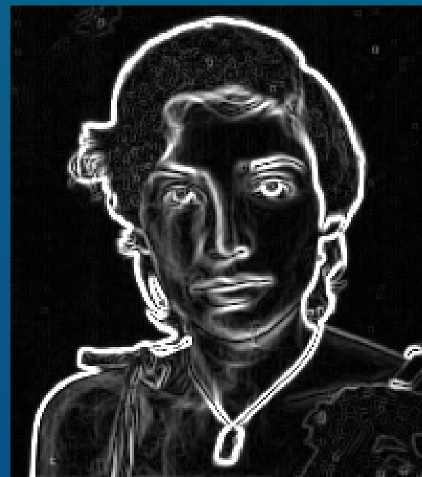
Principles of edge detection



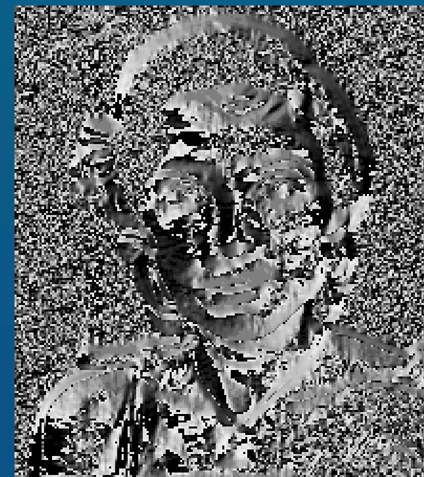
Steps of edge detection.

- Edge filter responds to edges and yields
  - Edge magnitude: strength of edge, a measure of local contrast
  - Edge orientation
- Tasks of edge localisation (post-processing):
  - Remove noisy edges
  - Remove 'phantom' edges, obtain thin contours
  - Obtain edge map: a binary edge image.
- Note: Noise smoothing may be applied before edge filtering.

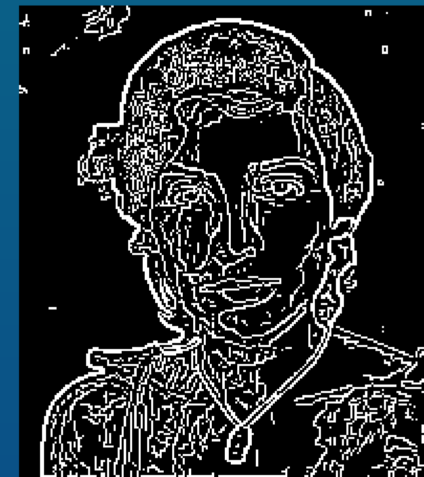
original image



edge magnitude



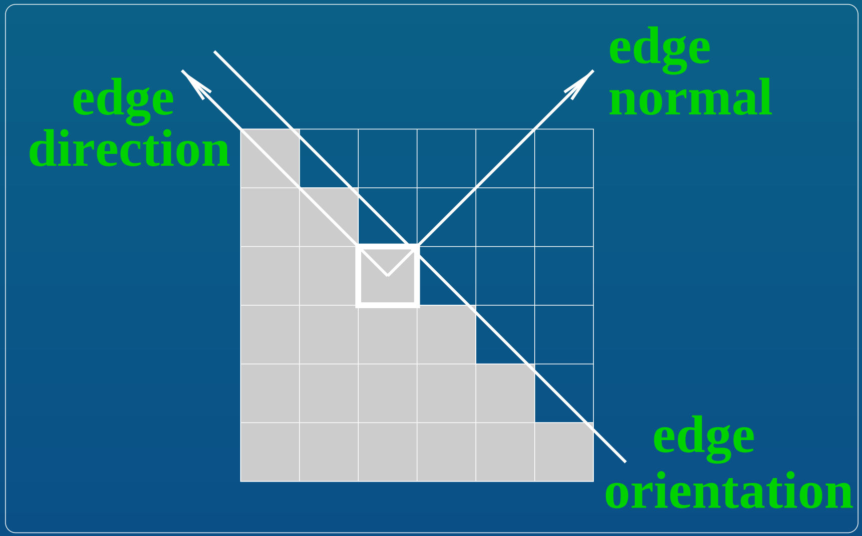
edge orientation



edge map

Example of edge detection by 3×3 Prewitt operator.
Edge orientation is circular data; shown intensity-coded.

Intensity profiles along lower (left) and upper (right) lines drawn in original image



Edge normal, edge direction and edge orientation.

- Edge normal: Direction of maximum intensity variation at edge point.
  - Unit vector perpendicular to the edge
- Edge direction: Direction tangent to the contour
  - Unit vector parallel to the edge
  - Convention needed for unumbiguous definition: e.g., 'dark on the left'
- Also used: Edge orientation, which is circular data interpreted modulo π.

Edge filters

Edge filters are high-pass filters using spatial derivatives of intensity function to

- enhance intensity variation across the edge
- suppress regions of constant intensity

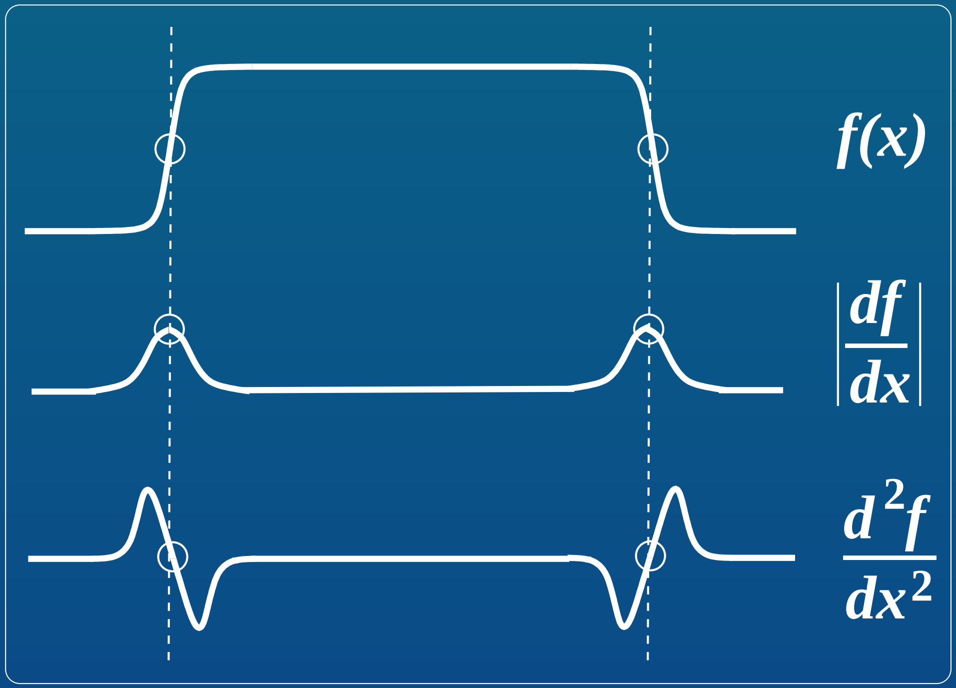
The following operators are applied in edge filtering:

- Intensity gradient is the vector composed of the first order partial derivatives:

$$\nabla f(x,y) \doteq \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right)$$

- Laplace operator is a scalar composed of the second order partial derivatives:

$$\Delta f(x,y) \doteq \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$



A signal and its first and second derivatives.

Edges are located at

- maxima of absolute value of first derivative
- zero-crossings of second derivative

Criteria for good edge filters

1. No response to flat regions $\Rightarrow$ Sum of mask values is zero: $\sum_{r,c} w(r,c) = 0$

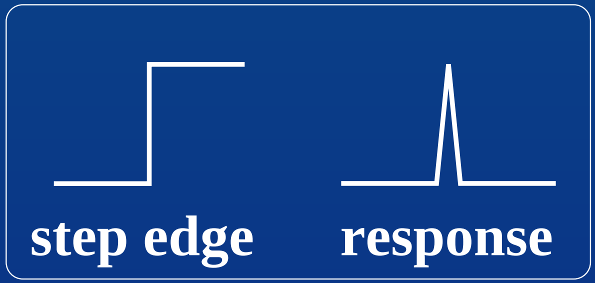
2. Isotropy: Response must be independent of edge orientation

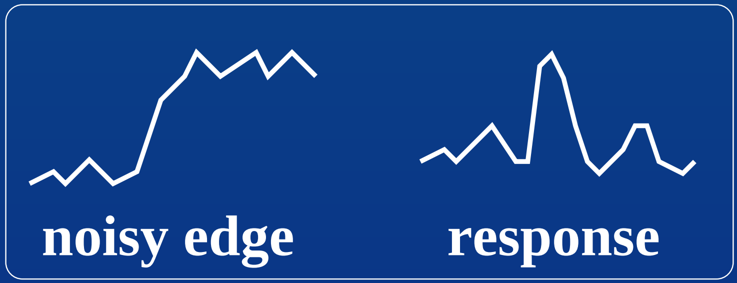
3. Good detection: Minimise the probabilities of

   - detecting spurious edges caused by noise (false positives)
   - missing real edges (false negatives)

4. Good localisation: Detected edges must be as close as possible to true edges.

5. Single response: Minimise number of false local maxima around true edge.





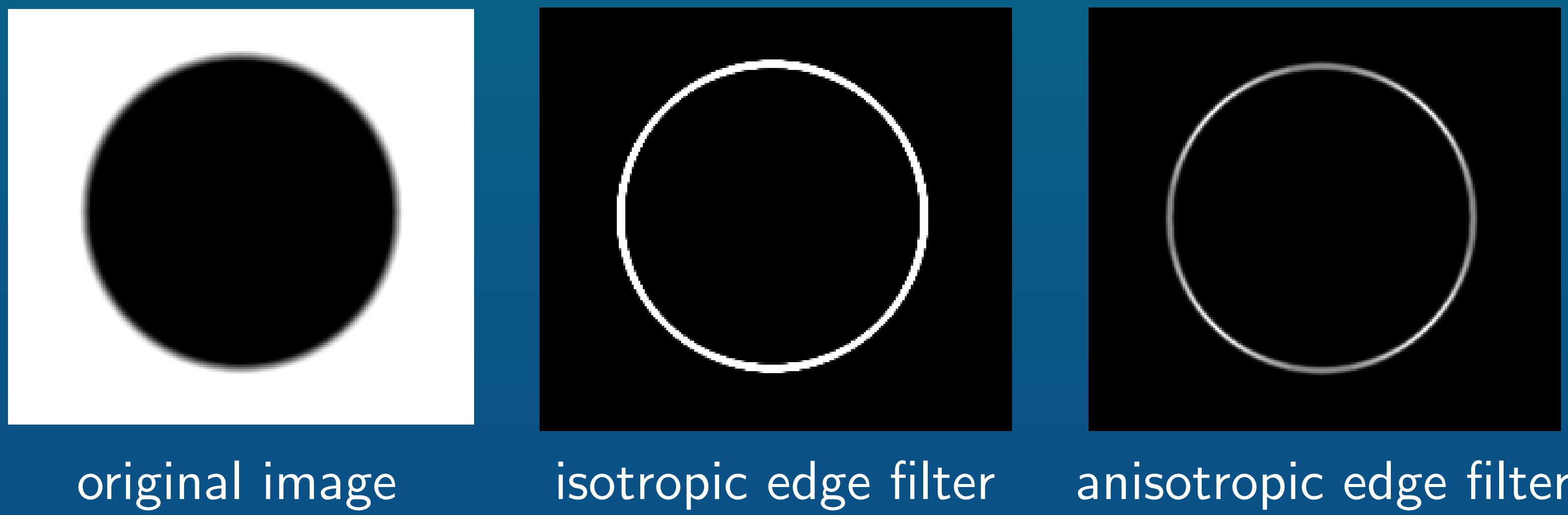


Illustration to the isotropy criterion.

- The isotropic edge filter yields uniform edge magnitude for all directions.

- The anisotropic edge filter yields non-uniform magnitude. In this illustration, the response depends on the edge orientation as follows:
  - Directions $45° \cdot k$ are slightly amplified
  - Directions $90° \cdot k$ are slightly suppressed

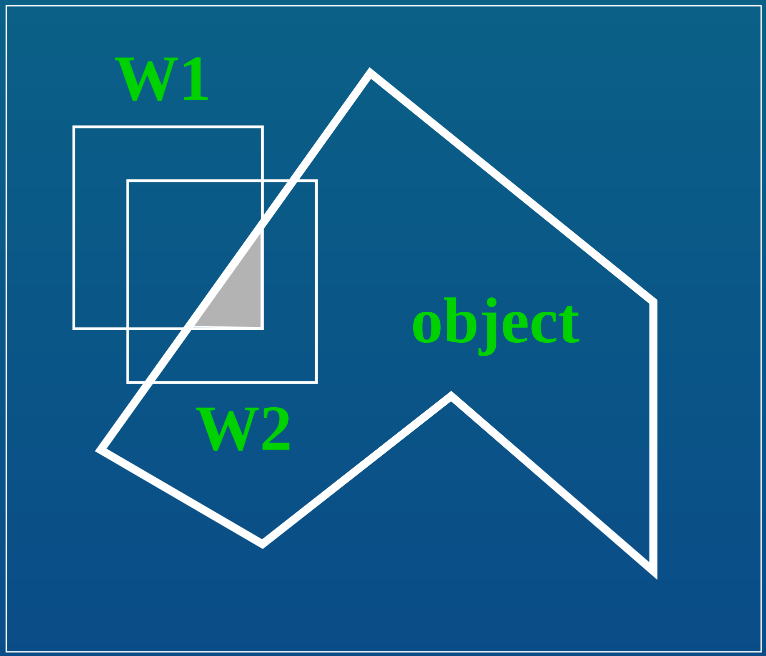


Illustration to the single response criterion.

- The same piece of contours is detected in window W1 and window W2.

$\Rightarrow$ 'Phantom' edges parallel to 'true' edges, thick contours

- The response depends on the overlap between the window and the contour.

- The multiple response is typical for all window-based detection tasks.

Gradient edge filters

Assume that the intensity function $f(x,y)$ is sufficiently smooth. The intensity gradient is the following vector:

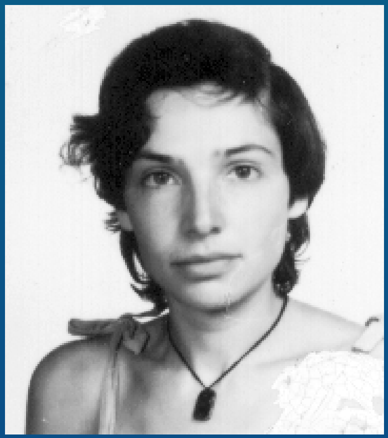
$$\nabla f(x,y) \doteq \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right) = (f_x, f_y)$$

The magnitude $M(x,y)$ and the orientation $\Theta(x,y)$ of the gradient vector are obtained as follows:

$$\begin{aligned} M(x,y) = \|\nabla f(x,y)\| &= \sqrt{f_x^2 + f_y^2} \\ \Theta(x,y) &= \arctan \frac{f_x}{f_y} \end{aligned}$$

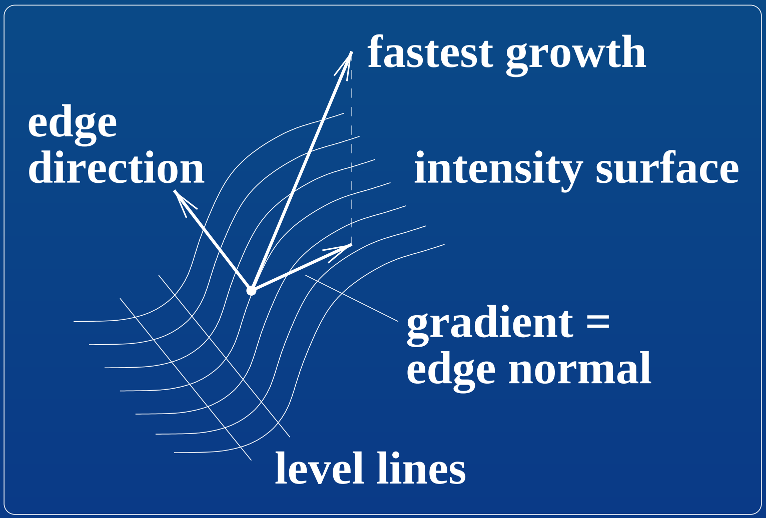
The gradient vector gives the direction and the magnitude of the fastest growth of intensity.

The meaning of the gradient vector


original image


intensity surface


thresholded image



Intensity surface of an edge and its gradient.

Simple 3×3 gradient masks

- In discrete images, partial derivatives are approximated by finite differences.
- The following family of gradient masks are used to compute the components of the gradient vector:

$$G_x = \frac{1}{p}\begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline 2-p & 0 & p-2 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \qquad G_y = \frac{1}{p}\begin{array}{|c|c|c|}\hline -1 & 2-p & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & p-2 & 1 \\ \hline \end{array}$$

- Different values of the parameter p result in different versions of the masks:

	Prewitt	Sobel	Isotropic
p	3	4	$2+\sqrt{2}$

  - When $p = 2 + \sqrt{2}$, the mask weights reflect the proximity to the mask origin.
  $\Rightarrow$ The operator becomes less sensitive to edge orientation.

Constraining the gradient masks

The above family of gradient masks obeys the following constraints:

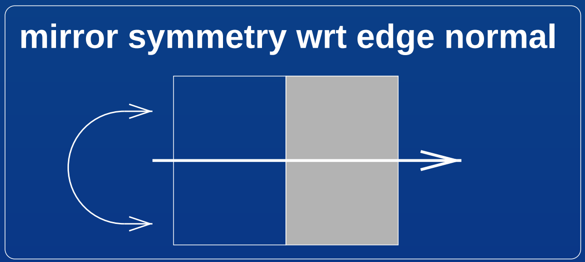
1. Mirror symetry with respect to (wrt) the edge normal:

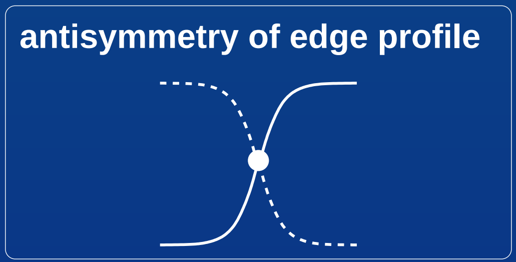
$$G_x(1,c) = G_x(3,c) \quad \text{and similar for } G_y$$

2. Antisymmetry wrt the edge orientation

$$G_x(r,1) = -G_x(r,3),\;\; G_x(r,2) = 0 \quad \text{and similar for } G_y$$

- requried for precise localisation of edges
- assumes antisymmetry of intensity profile of edge (sigmoid shape)





3. No response to flat regions: $\sum_{r,c} G_x(r,c) = \sum_{r,c} G_y(r,c) = 0$.
   - follows from the antisymmetry

4. Normalised response to ideal step edge of unit height: For such edge, the output value should be 1.

Using these constraints, the above family of gradient masks can be derived from a general unconstrained 3×3 mask: 9 free parameters reduce to 1.

The most frequently used are the Prewitt and the Sobel operators whose X-masks are as follows:

Prewitt ($\frac{1}{3}$ ×)

-1	0	1
-1	0	1
-1	0	1

Sobel ($\frac{1}{4}$ ×)

-1	0	1
-2	0	2
-1	0	1

- The Y-masks are obtained by 90° rotation of the X-masks.

- The Prewitt operator is the simplest and the fastest.

The Canny edge detector

The Canny edge detector is optimal for noisy step edge under the following assumptions:

- The edge filter is linear.
- The image noise is additive, white (uncorrelated) and Gaussian.

The optimality criterion used by Canny combines

- good detection and
- good localisation

To satisfy the single response criterion, two post-processing (edge localisation) operations are used:

- Non-maxima suppression
- Hysteresis thresholding

A practical approximation of the Canny filter

The original optimal edge filter is quite complicated. A simple practical approximation is as follows:

1. Apply Gaussian filter obtaining smoothed image $g(x, y) = f(x, y) * w_G(x, y; \sigma)$
   - The Gaussian parameter σ determines the size of the edge filter
2. Apply gradient operator $\nabla g(x, y)$ and calculate edge magnitude and orientation.

The scale parameter σ is selected based on

- the desired level of detail: fine edges vs global edges;
- the noise level;
- the localisation-detection trade off: see template matching.

An efficient implementation of the Canny filter

An efficient implementation uses

- the commutativity and associativity of linear filters:

$$\nabla\big(f(x,y) * w_G(x,y)\big) = \nabla\big(w_G(x,y) * f(x,y)\big) = \nabla\big(w_G(x,y)\big) * f(x,y)$$

- and the separability of the Gaussian filter:

$$w_G(x,y) = w_G(x) \cdot w_G(y).$$

As a result, the filter is implemented as a sequence of convolutions with 1D masks.

Edge localisation

- Input: edge magnitude $M(x, y)$ and edge orientation $\Theta(x, y)$ in each pixel of the image.

- Output: Binary edge map, with 1's indicating edges, 0's indicating no edges.

Localisation selects those maxima of $M(x, y)$ that correspond to true edge pixels.

- Can be applied after any edge filter that computes a measure of edge strength and provides edge orientation.
  - Gradient: Canny, Prewitt
  - Non-gradient: For example, Mérő and Vassy

- Includes the following operations:
  - Non-maxima suppression to remove 'phantom' edges and, if possible, obtain 1-pixel wide contours
  - Hysteresis thresholding to remove noisy maxima without breaking the contours

Non-maxima suppression

- Due to the multiple response, edge magnitude $M(x, y)$ may contain wide ridges around the local maxima.
- Non-maxima suppression removes the non-maxima pixels preserving the connectivity of the contours.

Algorithm 1: Non-maxima suppression

1. From each position (x, y), step in the two directions perpendicular to edge orientation $\Theta(x, y)$.
2. Denote the inital pixel (x, y) by C, the two neighbouring pixels in the perpendicular directions by A and B.
3. If the $M(A) > M(C)$ or $M(B) > M(C)$, discard the pixel (x, y) by setting $M(x, y) = 0$.

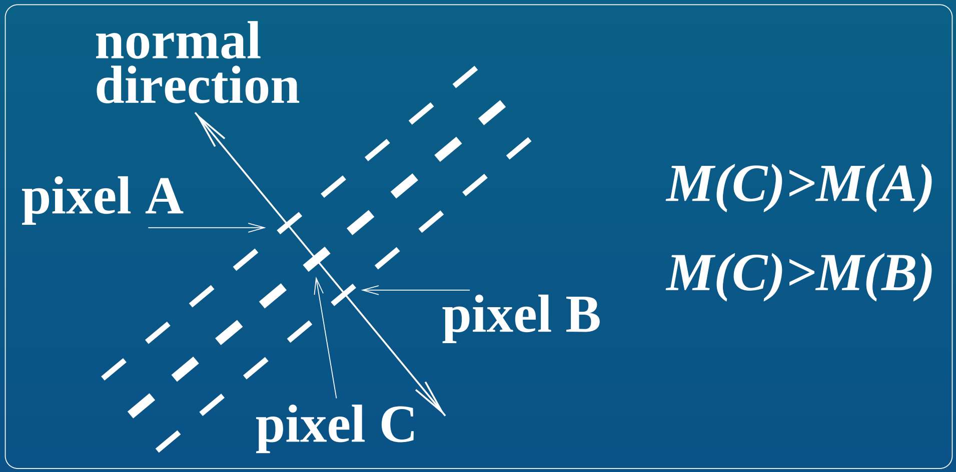
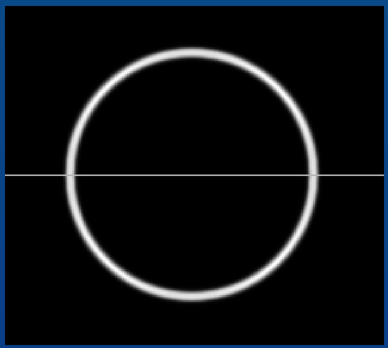
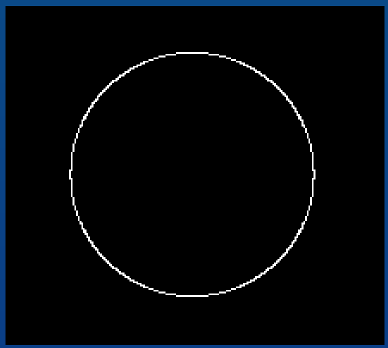


Illustration to the non-maxima suppression. Pixels A and B are deleted because $M(C) > M(A)$ and $M(C) > M(B)$. Pixel C is not deleted.



edge magnitude

intensity profile (resized)



result of NMS

Thinning wide contours in edge magnitude images by non-maxima suppression. The intensity profile along the indicated line is shown resized for better visibility.

Hysteresis thresholding

- The output of the non-maxima suppression still contains noisy local maxima.

- Contrast (edge strength) may be different in different points of the contour.

  $\Rightarrow$ Careful thresholding of $M(x, y)$ is needed to remove these weak edges while preserving the connectivity of the contours.

- Hysteresis thresholding receives the output of the non-maxima suppression, $M_{NMS}(x, y)$.

- The algorithm uses 2 thresholds, T_{high} and T_{low}.

  - A pixel (x, y) is called strong if $M_{NMS}(x, y) > T_{high}$.
  - A pixel (x, y) is called weak if $M_{NMS}(x, y) \leq T_{low}$.
  - All other pixels are called candidate pixels.

Algorithm 2: Hysteresis thresholding

1. In each position of (x, y), discard the pixel (x, y) if it is weak; output the pixel if it is strong.

2. If the pixel is a candidate, follow the chain of connected local maxima in both directions along the edge, as long as $M_{NMS} > T_{low}$.

3. If the starting candidate pixel (x, y) is connected to a strong pixel, output this candidate pixel; otherwise, do not output the candidate pixel.

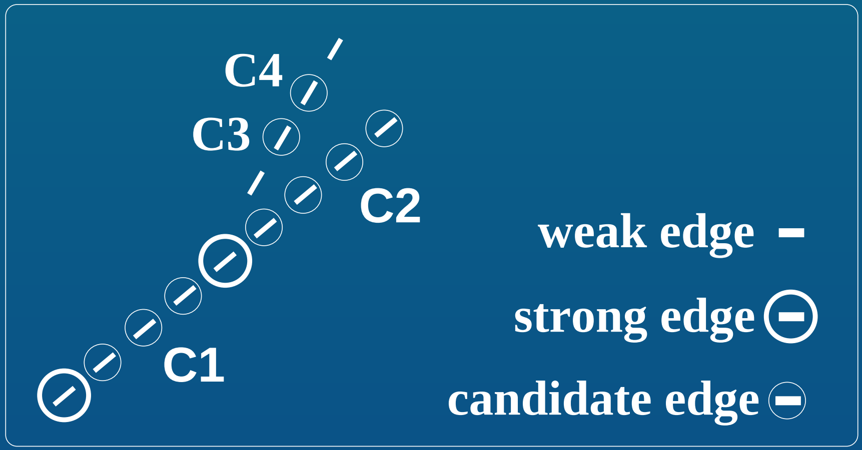


Illustration to the hysteresis thresholding. The candidate edges C1 and C2 are output, the candidate edges C3 and C4 are not.


original image


edge magnitude


$5, 20$


$20, 40$

Examples of edge localisation with different hysteresis thresholds.