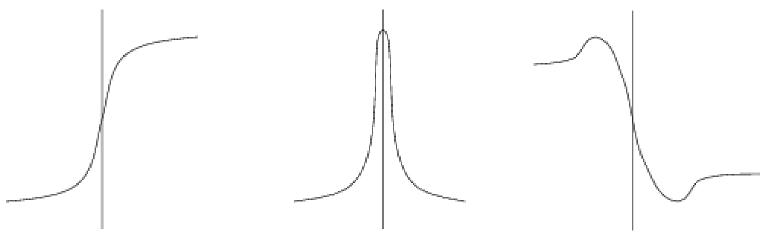
Image features

Image features, such as edges and interest points, provide rich information on the image content. They correspond to local regions in the image and are fundamental in many applications in image analysis: recognition, matching, reconstruction, etc. Image features yield two different types of problem: the **detection** of area of interest in the image, typically contours, and the **description** of local regions in the image, typically for matching in different images. In any case, they relate to the differential properties of the intensity function, for instance the gradient or the laplacian that are used to detect intensity discontinuities that occur at contours.



Example: the intensity function around a step edge and its first and second derivatives.

1 Definitions

Linear Filtering

The linear filtering of an image consists in convolving its intensity function $I(x, y)$ with a function $h(x, y)$ called impulse response of the filter.

$$I'(x, y) = h(x, y) * I(x, y),$$
$$I'(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} h(u, v)I(x - u, y - v)du\, dv,$$
$$I'(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} h(x - u, x - v)I(u, v)du\, dv,$$

In the discrete case:

$$I'(x, y) = \sum_{u=-H/2}^{+H/2} \sum_{v=-H/2}^{+H/2} h(u, v)I(x - u, y - v).$$

where H corresponds to the filter mask dimension.

The Image Gradient

The (intensity) gradient of an image is the vector $\nabla I(x, y)$ defined by:

$$\nabla I(x, y) = (\frac{\partial I(x, y)}{\partial x}, \frac{\partial I(x, y)}{\partial y})^t.$$

It is characterized by a magnitude m and a direction ϕ in the image :

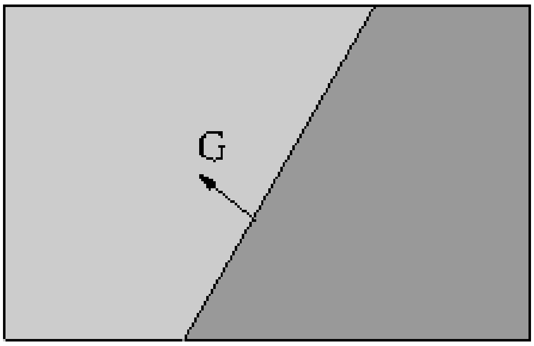
$$m = \sqrt{(\frac{\partial I(x, y)}{\partial x}^2 + \frac{\partial I(x, y)}{\partial y}^2)},$$
$$\phi = \arctan(\frac{\partial I(x, y)}{\partial y} / \frac{\partial I(x, y)}{\partial x}).$$

- ☞ The gradient direction maximizes the directional derivative.
- ☞ The directional derivative of $I(x, y)$ in the direction d is:

$$\nabla I(x, y) \cdot d.$$

- ☞ The gradient of a filtered image is:

$$\nabla I'(x, y) = \nabla(I(x, y) * h(x, y)) = \nabla I(x, y) * h(x, y) = I(x, y) * \nabla h(x, y).$$



The Image Laplacian

The laplacian of an image with intensity $I(x, y)$ is defined by:

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

☞ Invariant to image rotations.

☞ The laplacian is often used in image enhancement to increase contour effects:

$$I'(x, y) = I(x, y) - c\nabla^2 I(x, y).$$

☞ Higher sensitivity to noise than the gradient.

☞ The laplacian of a filtered image:

$$\Delta I'(x, y) = \Delta I(x, y) * h(x, y) = I(x, y) * \Delta h(x, y).$$

Separable Filters

A filter with impulse response $h(x, y)$ separable along the x and y axis is a filter for which:

$$h(x, y) = h_x(x)\ h_y(y),$$

hence for the filtering of an image:

$$I'(x, y) = h(x, y) * I(x, y),$$

$$I'(x, y) = h_y(y) * (h_x(x) * I(x, y)),$$

and the derivatives:

$$\frac{\partial I'(x,y)}{\partial x} = I(x,y) * (\frac{\partial h_x(x)}{\partial x}\, h_y(y)),$$
$$\frac{\partial I'(x,y)}{\partial y} = I(x,y) * (h_x(x)\, \frac{\partial h_y(y)}{\partial y}),$$
$$\Delta I'(x,y) = I(x,y) * (\Delta h_x(x) h_y(y) + h_x(x) \Delta h_y(y)),$$

The main interests of separable filters are to:

1. Transform bi-dimensional filtering of an image into two mono-dimensional filtering.
2. Reduce complexity: for a convolution with a filter of size H, complexity is $2H$ instead of H^2.
3. Allows recursive implementation of the filter.

Edge detection

Two main strategies:

1. Gradient strategy: detection of the local extrema in the gradient direction.
2. Laplacian strategy: detection of zero-crossing.

- ☞ These strategies rely on the fact that edges correspond to 0-order discontinuities of the intensity function.
- ☞ The derivative computation requires a pre-filtering of the images.v For instance: linear filtering for zero mean noises (e.g. white Gaussian noise and Gaussian filter) and non-linear filtering for impulse noise (median filter).

The existing approaches differ then with respect to the method used to estimate derivatives of the intensity function:

1. Finite differences.
2. Optimal filtering.
3. Prior intensity function model.

2 Estimating derivatives with finite differences

An image is discrete by nature, hence early approaches approximated derivatives using differences:

$$\nabla_u I(u,v) = I(u,v) - I(u-n,v),$$

where:

$$\nabla_u I(u,v) = I(u+n,v) - I(u-n,v),$$

with, in general, $n = 1$.

Such derivatives are computed by convolving the image with a mask of differences.

2.1 Roberts Operators (1962)

$$h1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad h2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

- ☞ $\phi = \arctan(I * h2 / I * h1) + \pi/4$.
- ☞ High sensitivity to noise due to the (small) mask size.

2.2 Prewitt Operators

$$h1 = 1/3 \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad h2 = 1/3 \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Gradient Masks in x et y.

- ☞ The convolution of the image by the above masks corresponds to the computation of the derivatives of the image filtered by a separable filter h:

  $$h(x,y) = h(x)\ h(y),$$

  with: $h = \frac{1}{3}[1\ 1\ 1]$ et $d = \nabla h = [-1\ 0\ 1]$. En effet :

  $$h1(x,y) = d(x)\ s(y),$$

  $$h2(x,y) = s(x)\ d(y),$$

☞ Directionnal Prewitt Masks:

$$h3 = 1/3 \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}, \qquad h2 = 1/3 \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix},$$

the gradient direction corresponds then to the mask giving the maximal response.

The estimation of the laplacian can proceed in a similar way by convolving the image with a mask of differences. For the second order derivative the $1D$ difference mask is: $\nabla^2 = [1 \; -2 \; 1]$. Thus in $2D$:

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

or:

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

Discrete Laplacian masks.

☞ Estimating the laplacian requires 1 convolution, the gradient 2.

☞ Invariance by rotation.

2.3 Sobel Operators (1972)

$$h1 = 1/4 \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad h2 = 1/4 \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

☞ Very popular (present in most standard image manipulation softwares).

☞ Corresponds to the convolution of the image with: $[1\ 2\ 1] * [-1\ 0\ 1]$.

☞ Directional masks exist but are computationally expensive.

2.4 Scharr Operators (1999)

Numerous local image descriptors consider gradient orientations using, for example, histograms. With the aim to improve the estimation of such gradient orientation, Scharr proposed the following operators obtained by optimizing the gradient estimation in the Fourier domain:

$$h1 = 1/16 \begin{bmatrix} -3 & 0 & 3 \\ -10 & 0 & 10 \\ -3 & 0 & 3 \end{bmatrix} \qquad h2 = 1/16 \begin{bmatrix} -3 & -10 & -3 \\ 0 & 0 & 0 \\ 3 & 10 & 3 \end{bmatrix}$$

☞ Scharr operators are recognized as more accurate than Sobel's one (e.g. in OpenCV for instance).

☞ Corresponds to the convolution of the image with:$[3\ 10\ 3] * [-1\ 0\ 1]$ where $[3\ 10\ 3]$ is a truncated discrete Gaussian filter, as for Sobel, but with a different standard deviation.

Figure 1: Sobel 3x3 on: the boat image; the boat image with impulse noise ; the noisy image filtered with a 3x3 Gaussian filter and a 3x3 median filter.

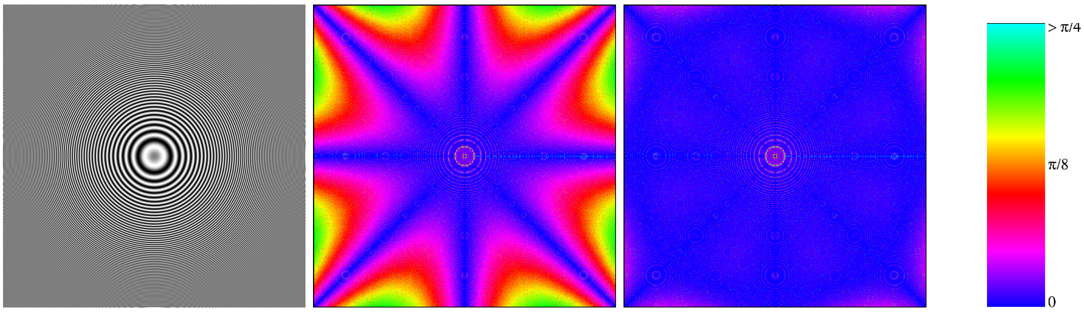


Figure 2: From left to right: test image with $I = sin(r^2)$; Sobel 3x3 angle errors; Scharr 3x3 angle errors; error colors.

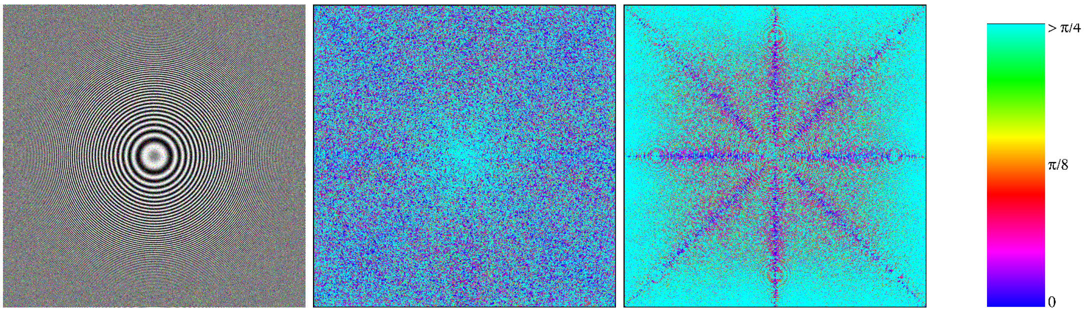


Figure 3: Test image with impulse noise; Scharr 3x3 ; Scharr 3x3 after 3x3 median filtering; error colors.

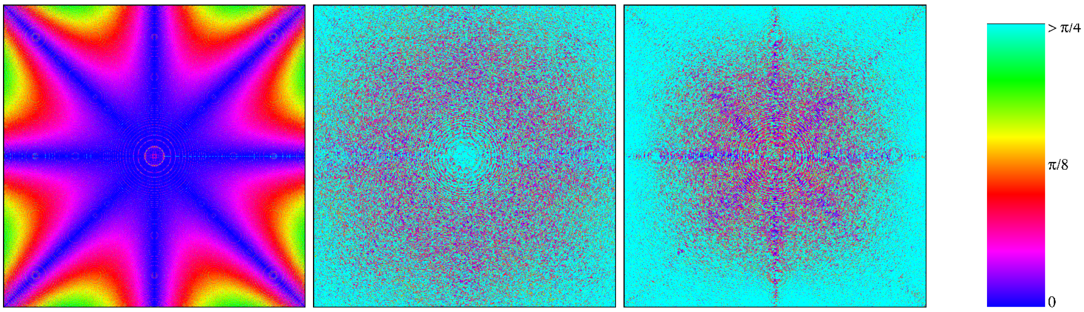


Figure 4: Sobel 5x5 on original test image ; Sobel 5x5 on test image with impulse noise ; Sobel 5x5 after 3x3 median filtering.

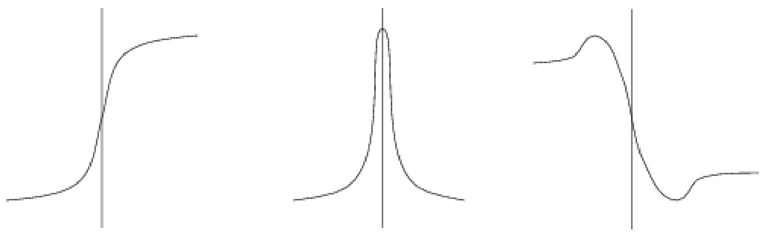
3 Differentiation with optimal filtering

The derivatives obtained with local convolution operators are local approximations. Consequently they present a high sensitivity to noise. Another strategy that was investigated consists in considering convolution with larger supports and with optimal filters with respect to feature localization and detection in the image. First the problem is reduced to a one dimensional estimation:

Let h be the (1D) smoothing filter then:

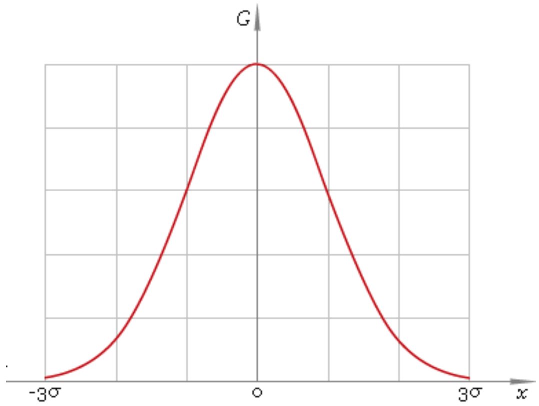
- $I(x, y) * h(x) * h(y)$ is the smooth image,
- $I(x, y) * h'(x) * h(y)$, $I(x, y) * h(x) * h'(y)$ are the images of the derivatives along the x and y directions,
- $I(x, y) * (h''(x) * h(y) + h(x) * h''(y))$ is the image of the Laplacian.

Note that the filter can often be implemented recursively (i.e. the output in one pixel can be determined with its neighbors values) with 1D convolutions (separable filter).



Example: convolution of a step edge intensity function with a Gaussian and its first and second derivatives.

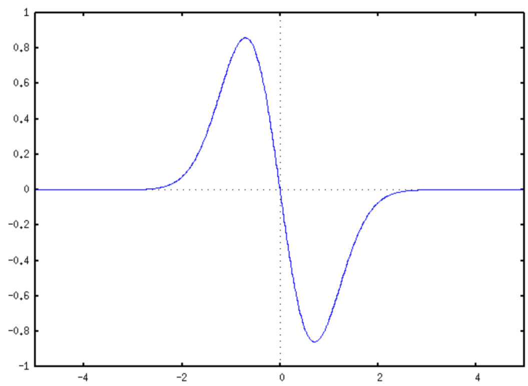
3.1 Gaussian filter



The Gaussian filter is one of the most popular smoothing filter. It writes:

$$h(x) = c\, e^{-x^2/2\tau^2},$$

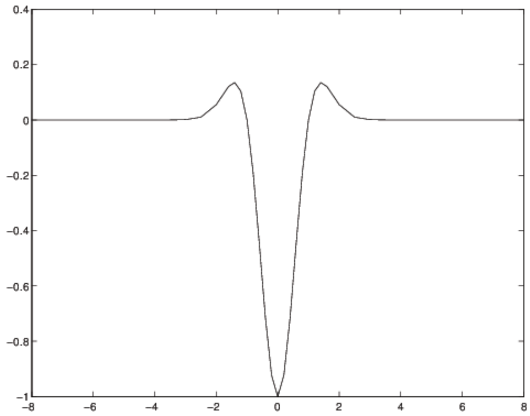
where c is a normalizing coefficient, e.g. $c = 1/\int h(x) = 1/\sqrt{(2\pi)}\sigma$.
The first derivative writes: $h'(x) = -c\,\frac{x}{\tau^2}\,e^{-x^2/2\tau^2}$.



The function $-x\, e^{-x^2/2}$.

The second derivative: $h''(x) = c\,(\frac{x^2}{\tau^2} - 1)\,e^{-x^2/2\tau^2}$.

It was shown by Canny that the first derivative filter present good properties for edge detection. This filter was introduced by Marr and Hildreth for the estimation of the Laplacian of the intensity function in the image: the Laplacian of Gaussian



The function $(x^2 - 1)\, e^{-x^2/2}$.

or LoG operator.
Considering $r = x^2 + y^2$:

$$h(r) = c\, e^{-r^2/2\tau^2},$$

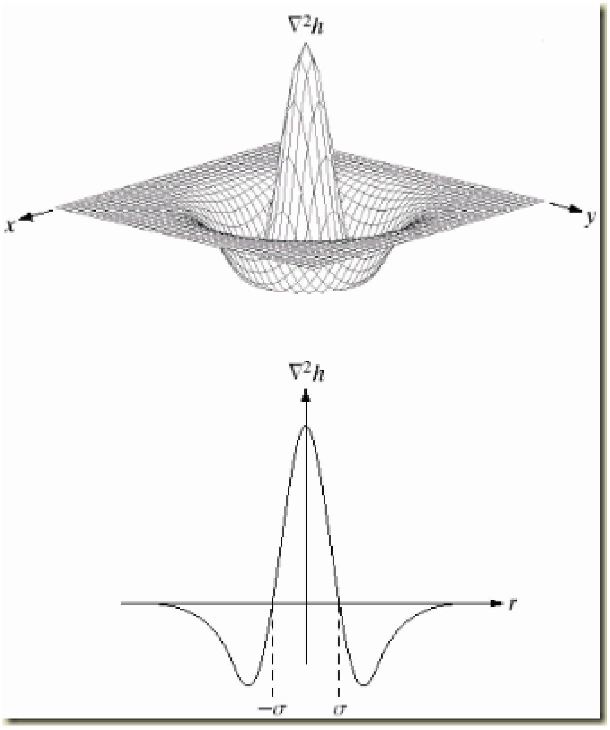
hence:

$$h''(r) = c\, (\frac{r^2}{\tau^2} - 1)\, e^{-r^2/2\tau^2},$$

$$h''(x,y) = c\, \frac{1}{\tau^2}\, (\frac{x^2+y^2}{\tau^2} - 1)\, e^{-(x^2+y^2)/2\tau^2},$$

where c normalizes to zero the sum of filter elements.

- ☞ The LoG operator is non-directional (or isotropic).
- ☞ Zero crossings are easier to determine than extrema.
- ☞ Noise sensitivity is increased.
- ☞ No information on the edge orientation.



The LoG operator.

3.2 Other filters

Other similar filters with with optimal properties with respect to edge detection have been proposed. For instance Deriche proposed following smoothing filter:

$$h(x) = k(\alpha\,|x| + 1)e^{-\alpha|x|},$$

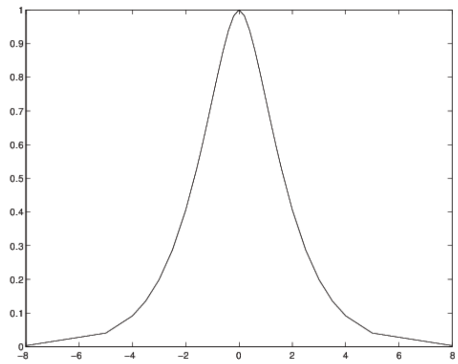
with:

$$k = \frac{(1 - e^{-\alpha})^2}{(1 + 2\alpha e^{-\alpha} - e^{-2\alpha})}.$$

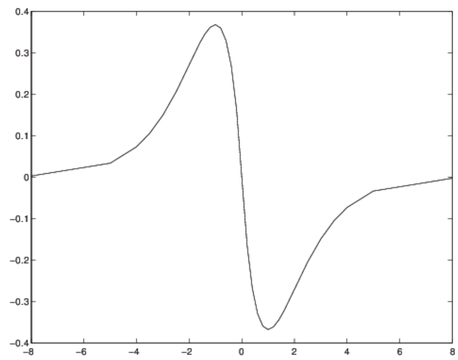
And:

$$h'(x) = -k' x e^{-\alpha|x|},$$
$$k' = \frac{(1 - e^{-\alpha})^2}{e^{-\alpha}}.$$

☞ Deriche filter is directional (anisotropic).



Impulse response.



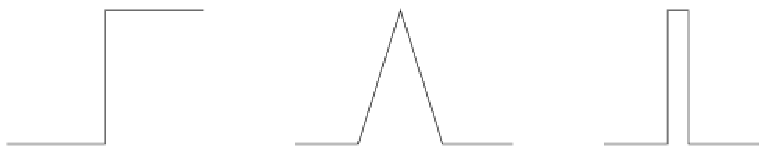
Impulse response of the derivative filter.

4 Edge Detection

Image edges come from:

- discontinuities of the reflectance function (textures, shadows),
- depth discontinuities (object edges),

and are characterized by discontinuities of the intensity function in images. Feature detection is therefore based on the observation of the derivatives of the intensity function and on the detection of local extrema of the gradient or zero crossing of the laplacian. A critical difficulty in this process results from the noise in the images. Such noise is present in each step of the acquisition process, e.g. sensor sensitivity and digitization.



Different types of edges: step, peak, roof.

The filters presented before allow to estimate the derivatives of an image, i.e. gradients and Laplacians. However **they do not identify edges** in the image and an additional step is required for that purpose.

4.1 Gradient approaches

Edges are characterized by local extrema of the gradient hence a first naive strategy is:

1. Estimate the gradient norm at all pixels in the image;
2. select pixels for which the gradient norm is above a user defined threshold.

☞ This does not efficiently discriminate edges from noise.

The computational approach that is traditionally used was introduced by Canny in 1986 and is still present in most standard image manipulation tools (OpenCV, Matlab, GIMP/Photoshop plugins, etc.). It is composed of the following steps:

1. Noise reduction: filter the image with a Gaussian filter (5x5 for instance).

2. Non-maximum suppression: extract local gradient extrema in the gradient direction. This means that for a pixel p values of the gradient along the line going through p and in the gradient direction are maximal in p. In practice, and due to pixel discretization, 4 directions are evaluated ($0deg$, $45deg$, $90deg$ and $135deg$).

3. Hysteresis thresholding: this step relies on a connexity assumption. The principle is to use 2 thresholds for the gradient norms: t_{low} and t_{high}. Pixels belonging to an edge are supposed to satisfy the 2 following conditions:

   (a) The pixel gradient norm is above t_{low},

   (b) The pixel is connected, through a pass composed of pixels with gradient norms above t_{low}, to at least one pixel with a gradient norm above t_{high}.

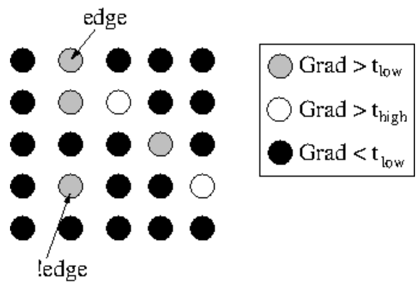


Figure 5: Hysteresis thresholding

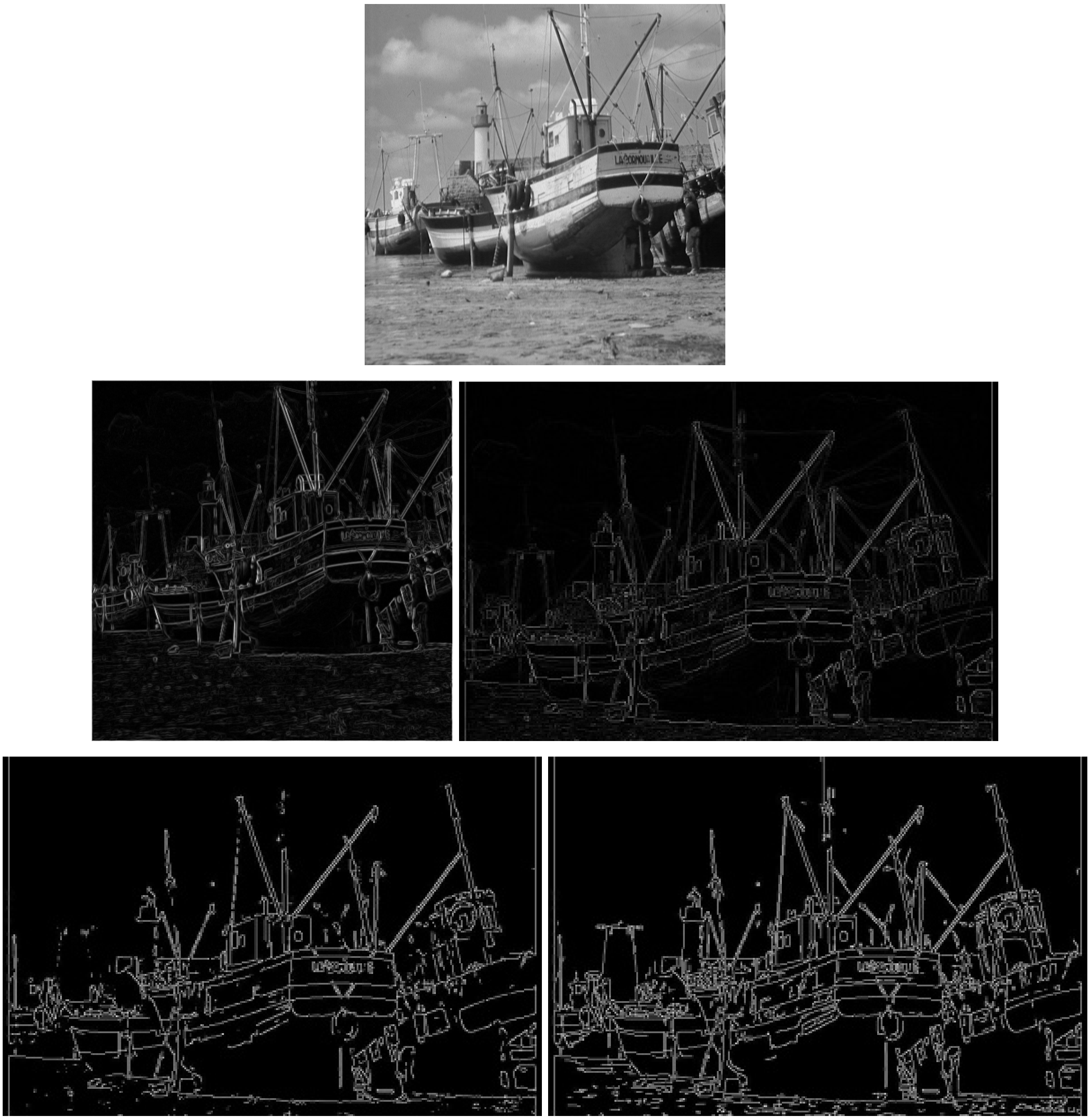


Figure 6: Boat image; Scharr 3X3; Deriche; Deriche with simple treshold; Canny-Deriche (hysteresis thresholding)

4.2 Laplacian approaches

Edges are characterized by zero crossings of the Laplacian. Edge detection in that case proceeds therefore in 3 steps:

1. Image smoothing.
2. Zero crossing detection. Pixels for which the Laplacian changes its sign are identified (positive-negative or negative-positive transitions).
3. Thresholding of zero crossings with high amplitudes (with hysteresis for instance).

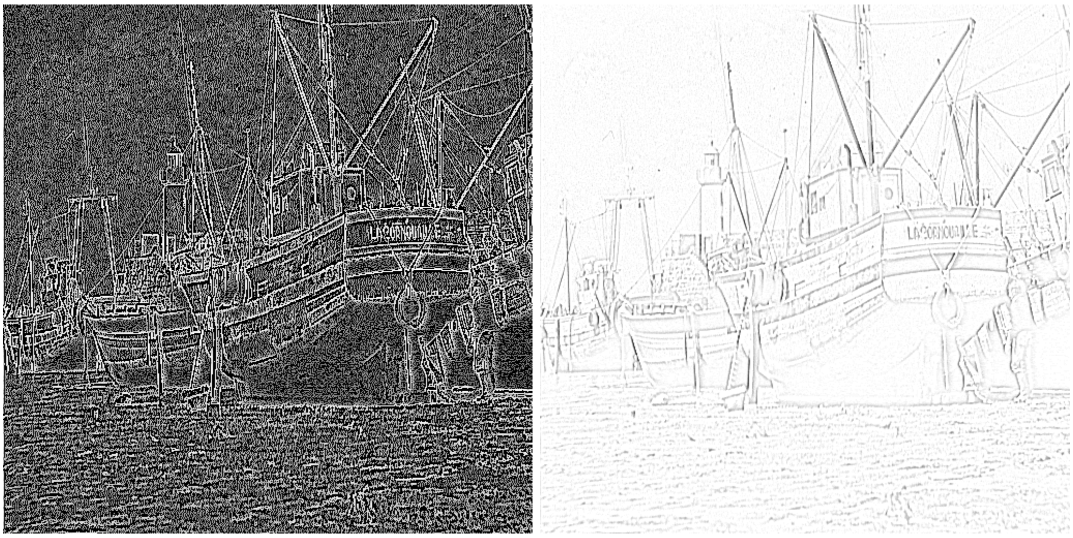
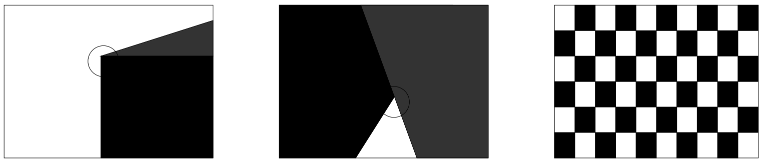


Figure 7: Boat image: Laplacian filter; DoG filter

5 Features points

Detecting features in images such as interest points is a preliminary step to numerous computer vision applications. Interest points usually correspond to double discontinuities of the intensity function. As for contours, these discontinuities may result from discontinuities of the reflectance function or from depth discontinuities. Interest points are for instance: corners, T-junctions or points with high texture variations.



Different types of interest points:
corners, T junctions and high texture variations

Some advantages of interest points with respect to contours:

1. More reliable source of information since the intensity function is better constraint.

2. Robust to occlusions (either visible or fully occluded).

3. No chaining required ($\neq$ contours !).

4. Present in a majority of images ($\neq$ contours !).

5.1 Different approaches

A number of approaches have been proposed to detect features (interest points) in images. They roughly fall into 3 categories:

1. Contour based approaches: the idea is to first detect contours. Interest points are then extracted along contours as points with maximal curvatures or intersections between contours.

2. Intensity based approaches: the idea is to directly consider the intensity function in images and to detect point where discontinuities occur.

3. Model based approaches: a model of the intensity function shape around an interest point is assumed and sought for in the image.

➔ Approaches from the second category were the most successful over the last decades. Reasons include: the independence with respect to contour detection (i.e. stability) and the independence with respect to the type of interest point (i.e. versatility).

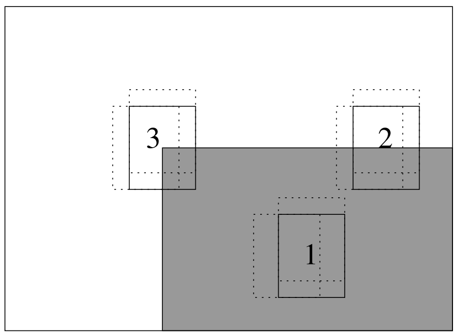
6 Interest Points: Moravec detector

An intuitive detector was proposed by Moravec in 1980 and has served as the basis for further and more evolved detectors. The idea is to consider the neighborhood of a pixel in the image (a window) and to determine mean changes of the intensity function when the neighborhood is moving in several directions. More precisely we consider the following function:

$$E(x,y) = \sum_{u,v} w(u,v) \;\; |I(x+u, y+u) - I(u,v)|^2,$$

that measures the mean of the intensity function variations when the neighborhood-window w is moved by (x, y), where:

- w specifies the neighborhood-window considered (value 1 inside the window and 0 outside);
- $I(u, v)$ is the intensity value at pixel (u, v).



The different situations considered by the Moravec detector.

Computing the values of the function E in the three following situations (see the above figure), we get:

1. The intensity is approximately constant in the area considered: E will take small values in any direction (x, y).
2. The area considered includes a contour: E will take small values for displacements along the contour direction and high values for displacements perpendicular to the contour.
3. The are considered includes a corner or an isolated point: E will take high values in any direction (x, y).

Consequently, the principle of the Moravec detector is to search for the local maxima (thresholding) of the minimal value of E over all pixels.

7 Interest Point: Harris detector

The Moravec detector works within a limited context and suffers from several limitations. Harris and Stephen identified some of these limitations and proposed in 1988 a popular detector that correct them: the *Harris detector*. The limitations of the Moravec detector taken into account by the Harris detector are:

1. The Moravec detector response is anisotropic due to the discretization of the moving directions that can be performed for intensity changes (45 degrees steps). To improve this aspect, one can consider the Taylor expansion of the intensity function around a pixel (u, v):

$$I(x+u, y+v) = I(u, v) + x\frac{\delta I}{\delta x} + y\frac{\delta I}{\delta y} + o(x^2, y^2).$$

   Hence:

$$E(x, y) = \sum_{u,v} w(u, v)[x\frac{\delta I}{\delta x} + y\frac{\delta I}{\delta y} + o(x^2, y^2)]^2,$$

   Neglecting the term $o(x^2, y^2)$ (which is valid for small displacements), we obtain the following analytical expression:

$$E(x, y) = Ax^2 + 2Cxy + By^2,$$

   with:

   - $A = \frac{\delta I}{\delta x}^2 \otimes w$
   - $B = \frac{\delta I}{\delta y}^2 \otimes w$
   - $C = (\frac{\delta I}{\delta x}\frac{\delta I}{\delta y}) \otimes w$

2. The Moravec detector response is noisy as a result of the neighborhood considered. The window function $w(u, v)$ is indeed a binary filter (values 0 or 1) applied over a rectangular neighborhood. To improve this aspect, Harris et Stephen proposed to used a Gaussian filter instead:

$$w(u, v) = \exp -(u^2 + v^2)/2\sigma^2.$$

3. Finally, the Moravec detector responds too strongly to contours due to the fact that only the minimal value of E in each pixel is considered. To take into account the general behavior of E locally, let us first write:

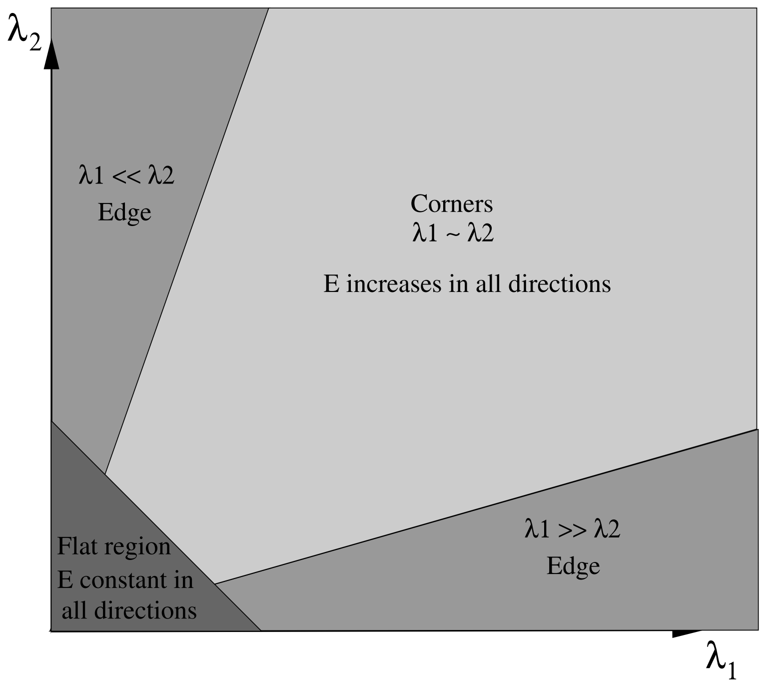
$$E(x, y) = (x, y) \cdot M \cdot (x, y)^t,$$

with:

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix}.$$

The matrix M describes the local behavior of the function E: the eigenvalues λ_1 and λ_2 of this matrix correspond to the principal curvatures associated to E locally and:

- both curvatures are low, the region under consideration presents an almost constatn intensity.
- One curvature is high while the other is low: the region contains a contour.
- Both curvatures are high: the intensity is varying in all directions which characterize a corner.



Classification of pixels with respect to curvatures/eigenvalues λ_1 and λ_2.

Consequently, Harris and Stephen proposed the following operator to detect corners in an image:

$$R = Det(M) - kTrace(M)^2$$

with : $Det(M) = AB - C^2$ et $Trace(M) = A + B$.

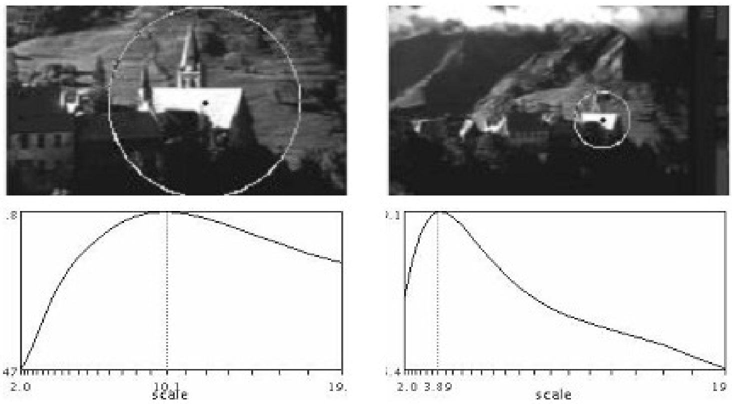
Values of R are positive around a corner, negative around a contour and low in a region of constant intensity.

8 Interest Point: SIFT

The SIFT algorithm (Scale Invariant Feature Transform) was proposed by David Lowe (university of British Columbia), in 1999 with the purpose of both detection and description of interesting area in the image (local features). It should be noticed that description is different from detection and consists in characterizing local image regions with the aim to recognize such regions (**to match**) in other images of the same scene. This algorithm has been very popular, not only in the computer vision community, and several modifications exist.

The general idea of SIFT is to find *features* that are invariant to several transformations: image rotation and scale, illumination, noise and minor changes in viewpoint.

8.1 Detection



Mikolajczyk (2002): The local LoG (Laplacian of Gaussians) extrema give the intrinsic scale.

The principle of the detection is therefore to find extrema in the scale-space representation of the image $I(x, y)$. This continuous representation is defined by the following function:

$$L(x, y, \sigma) = g_\sigma * I(x, y)$$

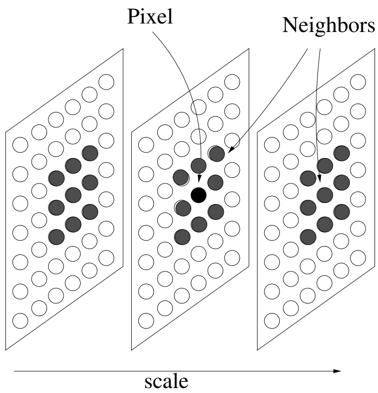
where g_σ is the Gaussian filter $g_\sigma = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}}$ and σ represents the scale parameter.

Consecutive Gaussian filtering of an image.

In order to find these extrema, and instead of considering the LoG function that is computationally expensive, the DoG (Difference of Gaussians) is used instead as an approximation:

$$DoG(x, y) = L(x, y, k\sigma) - L(x, y, \sigma)$$

The extrema are then pixels which are local minima/maxima of the DoG images across scales, i.e. with respect to their 8 spatial neighbors in the current scale image as well as their 9 neighbors in the next scale image and the 9 in the previous scale image.



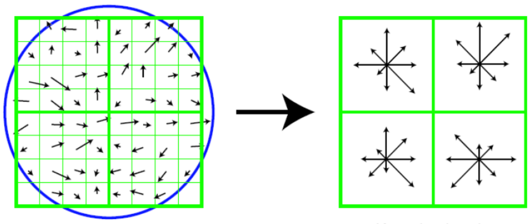
The extrema obtained this way are numerous. In order to filter them:

1. Candidates with low contrast are eliminated.

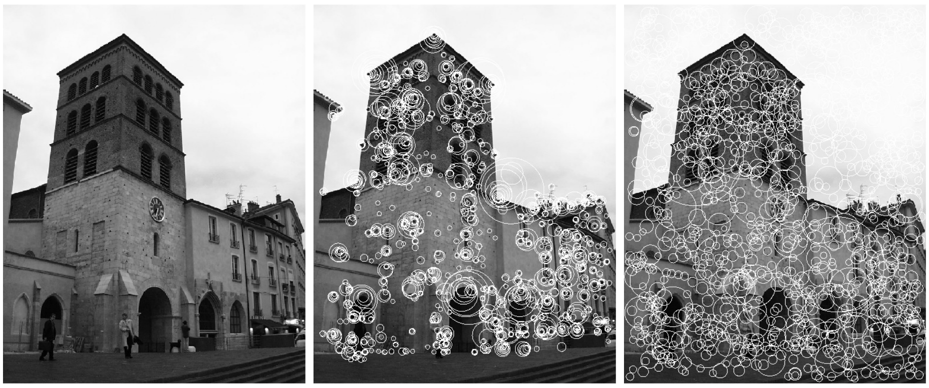
2. Responses corresponding to contours are eliminated by considering the Hessian of the DoG image and an operator close the Harris one.

8.2 Description

The description of a region of interest around a corner is represented by the histogram of gradient orientations in the region.



In the above case, the region is split into 4 subregions with 8 directions (the length in each direction represents the sum of the gradient modules having that direction in the subregion). Thus the description vector has 32 values. In typical applications, descriptors have 128 values: 4x4 subregions and 8 bins for directions.



Results with Harris and Sift.