DD2423: Feature Detection I

Receptive fields, Scale space, Edge detection

Tony Lindeberg

Division of Computational Science and Technology
KTH Royal Institute of Technology
Stockholm, Sweden

November 8, 2019

Feature detection

Basic question: How to define features from image data?

Feature-based paradigm: Compute a reduced subset of reliable image features so that a major part of subsequent processing can be restricted to a comparably smaller number of features than the number of image pixels.

Feature detection

Desirable purposes in computer vision:

- Matching details of objects between different views for correspondence, tracking, 3-D reconstruction, computing 3-D motion.
- Delimiting the outline of objects in natural scenes
- Detecting or delimiting subparts of objects
- Computing symmetry axes of elongated objects
- Computing image primitives for performing object recognition

Feature detection

How to define features from image data if the data in reality look like this?

172	143	184	200	180	153	101	101	116	87	63	105	77	71	67
176	141	170	193	176	153	103	102	123	99	67	113	75	82	83
177	153	188	185	173	161	114	83	99	94	71	123	80	85	87
178	175	198	205	200	206	157	109	97	131	119	116	70	73	79
164	164	160	153	134	119	111	95	99	110	115	121	79	48	48
163	133	123	105	80	66	66	81	104	74	55	84	62	55	54
173	118	91	66	62	60	52	76	103	77	57	93	69	76	78
169	123	112	79	113	112	63	63	78	69	55	78	75	81	76
169	135	121	106	110	113	65	68	71	67	72	129	124	113	77
188	160	184	113	80	55	48	61	73	65	61	85	68	50	51
189	211	222	183	151	128	126	134	146	149	139	90	72	57	71
162	170	175	174	173	176	180	177	177	170	164	150	145	146	143

(This is the result of showing the pixel values from the previous image at a lower resolution, after reducing the image size by a factor of eight.)

Feature detection

What approach should one take?

- Is it up to us as engineers to try to *design* or somehow figure out what are good image features?

  Would *any* type of feature detector do?

  Design $\Rightarrow$ Experiment $\Rightarrow$ New design $\Rightarrow$ New experiment $\Rightarrow$ ...
- Try to learn suitable features from collections of image data?
- Could we try to learn from biology?

  Many higher animals use vision as a main source of information to the surrounding world and solve the vision problem highly effectively.
- Is it possible to develop a mathematically based theory for how to perform feature detection from image data?

Interestingly, it is possible to address this problem systematically, by a well-founded theory, referred to as *scale-space theory*.

Results from this theory are also in good agreement with existing knowledge about early receptive fields in biological vision.

Feature detection

If we would like to formulate a *theory for early visual operations*:

- What should it be based on?
- What are the *constraints*?

Inherent variabilities in image data

Visual stimuli vary substantially on the retina due to geometric transformations and lighting variations in the environment.

Nevertheless the brain maintains a stable perception of the environment.

Figures from Lindeberg (2013) “A computational theory of visual receptive fields”, Biological Cybernetics, 107(6): 589-635.

Sources to variabilities in real world image data

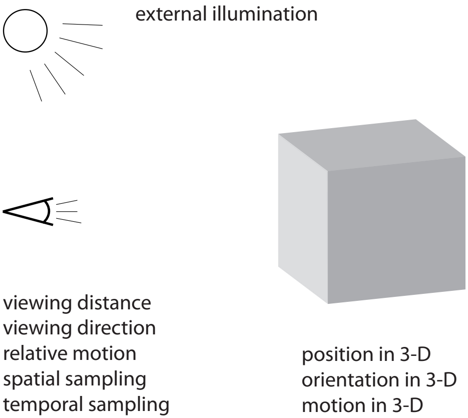
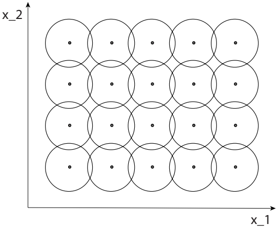


Figure from Lindeberg (2013) “Generalized axiomatic scale-space theory”, Advances in Imaging and Electron Physics, 178: 1-96.

Receptive fields

Receptive field: Region in the visual field for which a visual sensor/neuron/operator responds to visual stimuli.



Distribution of overlapping receptive fields over space and time.

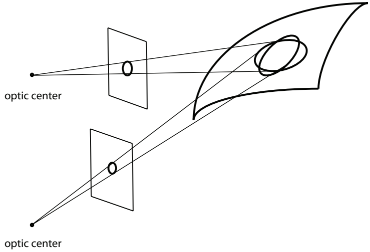
How should such receptive fields be designed in a principled way? (in biology / computer vision)

How to achieve invariant responses despite the variabilities in image data?

Figure from Lindeberg (2013) "A computational theory of visual receptive fields", Biological Cybernetics, 107(6): 589-635.

Covariant receptive fields

Basic requirement: Covariance under image transformations



If the family of receptive fields is not covariant under basic image transformations, then there will be a systematic error caused by the mismatch between the backprojected receptive fields.

Figure from Lindeberg (2013) “A computational theory of visual receptive fields”, Biological Cybernetics, 107(6): 589-635.

Basic image transformations

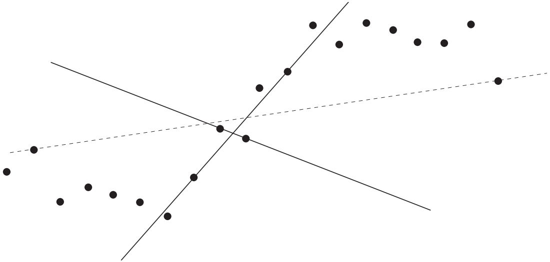
Local linearizations of non-linear image transformations:

- *scaling transformations* caused by objects of *different size* and at *different distances* to the observer
- *affine transformations* modelling *image deformations* caused by variations in the viewing direction

Such geometric image transformations are inherent to the image formation process and must therefore be taken into account when designing vision system for an unconstrained natural environment.

Dependency on scale of observation

Even an as “simple” problem as detecting the edges of an object by derivative approximations may be strongly dependent on the scale of the image operators:



Notably, qualitatively very different results can be obtained depending on the spatial extent of the difference operator, where only spatial extents within a certain scale range give meaningful results for the task.

Figure from Lindeberg (2013) “Generalized axiomatic scale-space theory”, Advances in Imaging and Electron Physics, 178: 1-96.

Multi-scale structure of real-world images

More generally, real-world object are typically composed of different types of structures at different scales:

crowd

$\updownarrow$

person

$\updownarrow$

face

$\updownarrow$

eye

$\updownarrow$

iris

$\updownarrow$

$\vdots$

Figure from Lindeberg (2013) “A computational theory of visual receptive fields”, Biological Cybernetics, 107(6): 589-635.

Multi-scale representation

For a computer vision system that observes an a priori unknown scene, there is usually no way to know in advance what scales are appropriate for describing the interesting image structures in the scene.

- ⇒ Represent the image data at *all* scales simultaneously.
- ⇒ Expand the data set over additional dimension(s) using the scale of the receptive fields as the parameter.

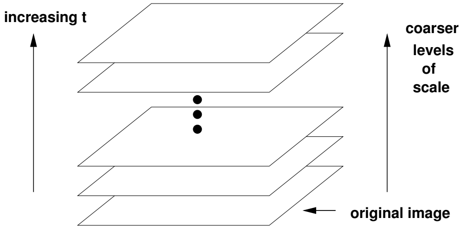


Figure adapted from Lindeberg (1994) Scale-Space Theory in Computer Vision, Springer.

Multi-scale descriptions

- **Physics:**
  Different types of descriptions depending on the scale of analysis

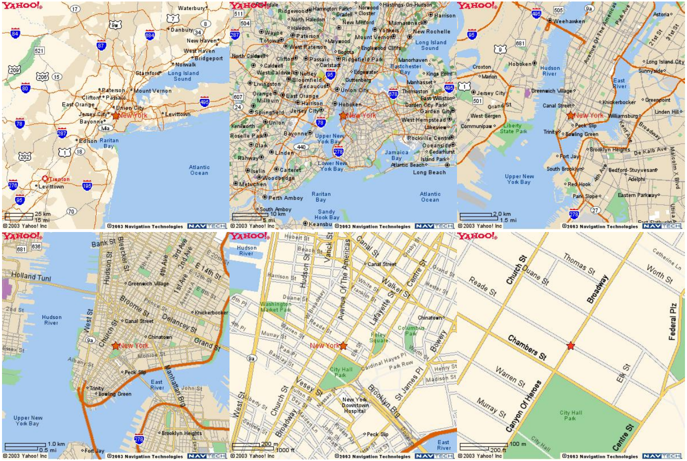
  quantum mechanics $\Rightarrow$ *particle physics* $\Rightarrow$ *thermodynamics* $\Rightarrow$ *solid mechanics* $\Rightarrow$ *astronomy* $\Rightarrow$ *relativity theory*

- **Cartography**
  Maps with different degrees of abstraction depending on scale

  building $\Rightarrow$ *city* $\Rightarrow$ *county* $\Rightarrow$ *country* $\Rightarrow$ *world*

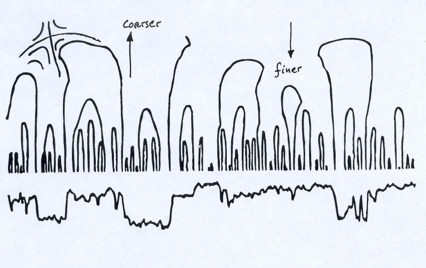
- **Computer vision**
  Dynamically varying scale levels must be handled *automatically*

Multi-scale maps



Scale-space representation

- Continuum of scale levels: $f(x) \mapsto L(x;\ s)$ with $L(x;\ 0) = f(x)$
- The transformation from a fine scale to a coarser scale *must not introduce new structures* not present in the original data.



- Formulate a well-founded theory for multi-scale image structures: *Scale-Space Theory*

Figure adapted from Witkin (1983) “Scale-space filtering”, Proc. Int. Joint Conf. on Art. Intell., 1019–1022.

Symmetry properties over spatial image domain

Given any image data f, define family of internal representations

$$L(\cdot;\ s) = \mathcal{T}_s\, f$$

over parameter s for family of image operators $\mathcal{T}_s$ that satisfies:

- *Linearity*

  $$\mathcal{T}_s(a_1 f_1 + a_2 f_2) = a_1 \mathcal{T}_s f_1 + b_1 \mathcal{T}_s f_2$$

  (as few irreversible decisions as possible, specifically scale-space properties defined over L transfer to any spatial derivative of L)

- *Shift invariance*

  $$\mathcal{T}_s\,(S_{\Delta x} f) = S_{\Delta x}\,(\mathcal{T}_s f)$$

  with $S_{\Delta x}$ denoting shift operator $(S_{\Delta x} f)(x) = f(x - \Delta x)$
  (visual interpretation of an object should be the same irrespective of its position in the image plane)

Symmetry properties over spatial image domain

- *Semi-group* structure over scale s

$$\mathcal{T}_{s_1} \mathcal{T}_{s_2} = \mathcal{T}_{s_1+s_2}$$

- *Scale covariance* under scaling transformations $x' = Sx$

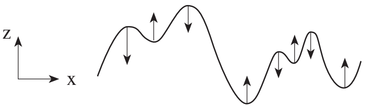
$$L'(x';\; s') = L(x;\; s) \quad \text{corresponding to} \quad \mathcal{T}_{\mathcal{S}(s)}\, \mathcal{S}\, f = \mathcal{S}\, \mathcal{T}_s\, f$$

(closedness and uniform treatment under scaling transformations)

Non-creation of structure with increasing scale: Non-enhancement of local extrema

Require: If at some scale s_0 a point $x_0 \in \mathbb{R}^N$ is a local maximum (minimum) for the mapping $x \mapsto L(x;\ s_0)$, then

- $(\partial_s L)(x;\ s) \leq 0$ at any spatial maximum
- $(\partial_s L)(x;\ s) \geq 0$ at any spatial minimum



Implies strong condition on the set of possible smoothing kernels $T(\cdot;\ s)$

Necessity result

Require

(i) linearity

(ii) shift invariance over space

(iii) semi-group property over scale

(iv) sufficient regularity properties over space and scale

(v) non-enhancement of local extrema

$\Rightarrow$ the scale-space representation over a 2-D spatial domain must satisfy

$$\partial_s L = \tfrac{1}{2} \nabla_x^T (\Sigma_0 \nabla_x L) - \delta_0^T \nabla_x L$$

for some 2×2 covariance matrix Σ_0 and some 2-D vector δ_0 with $\nabla_x = (\partial_{x_1}, \partial_{x_2})^T$

Proof in Lindeberg (2011) "Generalized Gaussian scale-space axiomatics comprising linear scale-space, affine scale-space and spatio-temporal scale-space", Journal of Mathematical Imaging and Vision, 40(1): 36–81.

Gaussian receptive fields

Require the convolution kernels to be *rotationally symmetric*

$$T(x;\ s) = g(x;\ s) = \frac{1}{2\pi s}\, e^{-x^T x/2s} = \frac{1}{2\pi s}\, e^{-(x_1^2+x_2^2)/2s}$$

with corresponding *Gaussian derivative operators*

$$(\partial_{x^\alpha} g)(x;\ s) = (\partial_{x_1^{\alpha_1} x_2^{\alpha_2}} g)(x_1, x_2;\ s) = (\partial_{x_1^{\alpha_1}} \bar{g})(x_1;\ s)\,(\partial_{x_2^{\alpha_2}} \bar{g})(x_2;\ s)$$

where

$$\tilde{g}(x_1;\ s) = \frac{1}{\sqrt{2\pi s}} e^{-x_1^2/2s}$$

$$\bar{g}_{x_1}(x_1;\ s) = -\frac{x_1}{s}\bar{g}(x_1;\ s) = -\frac{x_1}{\sqrt{2\pi} s^{3/2}} e^{-x_1^2/2s}$$

$$\bar{g}_{x_1 x_1}(x_1;\ s) = \frac{(x_1^2 - s)}{s^2}\bar{g}(x_1;\ s) = \frac{(x_1^2 - s)}{\sqrt{2\pi} s^{5/2}} e^{-x_1^2/2s}$$

Gaussian scale-space representation

original image

$s = 1$

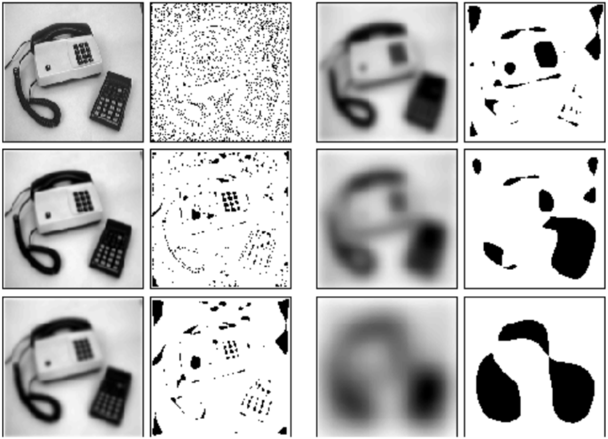
$s = 8$

$s = 64$

Figures from Lindeberg (2009) “Scale-space”, Encyclopedia of Computer Science and Engineering, IV: 2495–2504.

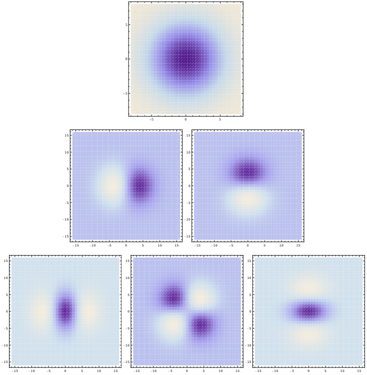
Different image structures at different scales

Dark grey-level blobs (local minima with spatial extent) at multiple scales:



Figures from Lindeberg (1994) Scale-Space Theory in Computer Vision, Springer.

Gaussian derivative kernels



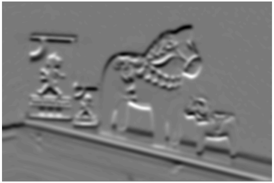
can be used as a *general basis* for expressing image operations such as feature detection, feature classification, surface shape, image matching and image-based recognition

Results of Gaussian derivative operators

$f(x, y)$

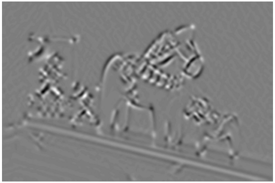
L_x

L_y

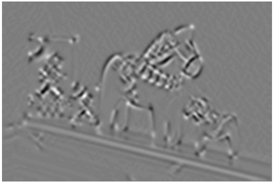


L_{xx}

L_{xy}



L_{yy}



Figures from Lindeberg (2013) “Generalized axiomatic scale-space theory”, Advances in Imaging and Electron Physics 178: 1-96.

Affine Gaussian receptive fields

Relax rotational symmetry to *mirror symmetry* $T(-x;\ s) = T(x;\ s)$
$\Rightarrow$ *affine Gaussian kernels*

$$T(x;\ s) = g(x;\ \Sigma) = \frac{1}{2\pi\sqrt{\det \Sigma}} e^{-x^T \Sigma^{-1} x/2}$$

where Σ denotes any symmetric positive semi-definite 2×2 matrix

Affine scale-space is *closed* under affine transformations

$$f_L(\xi) = f_R(\eta) \quad \text{where} \quad \eta = A\xi + b$$

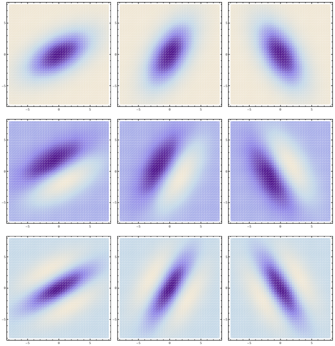
and

$$L(\cdot;\ \Sigma_L) = g(\cdot;\ \Sigma_L) * f_L(\cdot), \quad R(\cdot;\ \Sigma_R) = g(\cdot;\ \Sigma_R) * f_R(\cdot)$$

imply

$$L(x;\ \Sigma_L) = R(y;\ \Sigma_R) \quad \text{where} \quad \Sigma_R = A \Sigma_L A^T$$

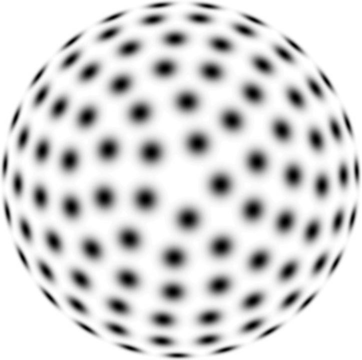
Affine Gaussians and directional derivatives



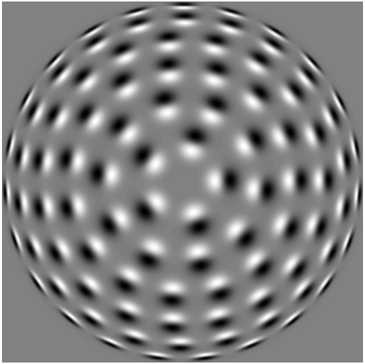
Affine scale-space used for computing *affine invariant image descriptors* for *e.g.* cues to surface shape, image-based matching and recognition

Affine Gaussian receptive fields

Zero-order kernels



First-order derivatives



Here with uniform distribution on the hemisphere to obtain covariance with respect to all possible viewing directions relative to objects in the world.

Figures from Lindeberg (2013) “A computational theory of visual receptive fields”, Biological Cybernetics, 107(6): 589-635.

Covariance properties

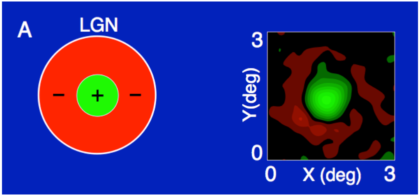
Covariance properties of spatial receptive fields:

- *rescalings* of image space dimensions
- *affine transformations* of the spatial domain

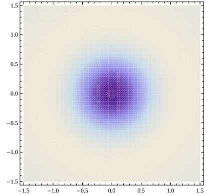
Allow the vision system to handle

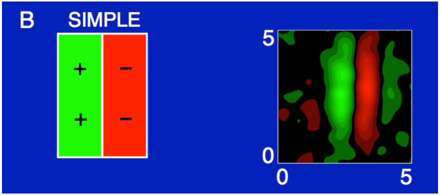
- image data acquired with different *resolution*
- image structures of different spatial *extent*
- objects at different *distances* from the camera
- the linear component of *perspective deformations*

Biological receptive fields in LGN and V1

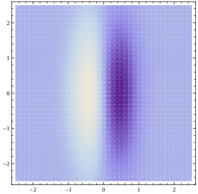


$\nabla^2 g(x, y;\ s)$



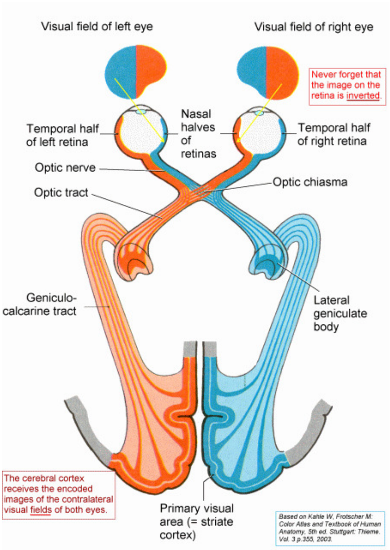


$\partial_x g(x, y;\ \Sigma)$



Figures from Lindeberg (2013) “Invariance of visual operations at the level of receptive fields”, PLOS ONE, 8(7): e66990:1–33.

Locations of LGN and V1 in the visual pathway



LGN neurons over spatial domain

- Most cells approximately *circular center-surround*
- Corresponding *scale-space model*

$$h_{LGN}(x_1, x_2, t;\ s, \tau) = \pm(\partial_{x_1 x_1} + \partial_{x_2 x_2})\, g(x_1, x_2;\ s)$$

where

  - $\pm$ determines polarity
  - $(\partial_{x_1 x_1} + \partial_{x_2 x_2})$ spatial Laplacian
  - $g(x_1, x_2;\ s)$ rotationally symmetric spatial Gaussian
  - s spatial scale parameter

Simple cells over spatial domain

- Receptive fields *oriented* in the spatial domain
- Idealized receptive field model for the *spatial component*

$$h_{space}(x_1, x_2;\ s) = (\cos\varphi\,\partial_{x_1} + \sin\varphi\,\partial_{x_2})^m\, g(x_1, x_2;\ \Sigma)$$

where

- $(\cos\varphi\,\partial_{x_1} + \sin\varphi\,\partial_{x_2})$ directional derivative operator
- m order of spatial differentiation
- $g(x_1, x_2;\ \Sigma)$ affine Gaussian kernel

Gaussian derivatives

The above results state that *Gaussian derivatives* based on either rotationally symmetric Gaussian kernels or affine Gaussian kernels constitute a canonical basis for expressing visual operations.

Such Gaussian derivatives can be computed at any scale and up to any order N, leading to a so-called multi-scale N-jet representation.

$$\begin{array}{l} \{L_x, L_y, L_{xx}, L_{xy}, L_{yy}, \dots\} \; at \; s = s_1 \\ \{L_x, L_y, L_{xx}, L_{xy}, L_{yy}, \dots\} \; at \; s = s_2 \\ \{L_x, L_y, L_{xx}, L_{xy}, L_{yy}, \dots\} \; at \; s = s_3 \\ \qquad\qquad \vdots \end{array}$$

Feature detection from differential invariants

Our next issue concerns how to *combine* such Gaussian derivatives for designing feature detectors.

A basic paradigm consists of combining Gaussian derivatives into *differential invariants* that are invariant to e.g. rotations.

In the following we will show how this can be done regarding the topics of:

- edge detection
- interest point detection
  (more recent terminology for blob detection and corner detection)

and show applications of computing scale-invariant image descriptors for image-based matching and object recognition.

Edge detection: General

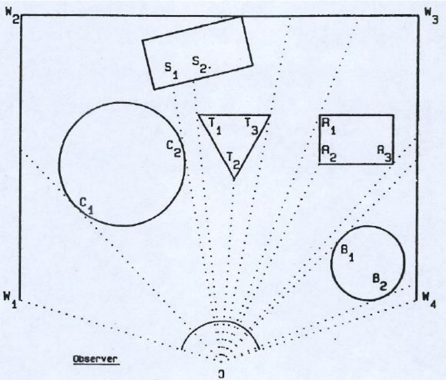
Under rather general assumptions about the image formation process:

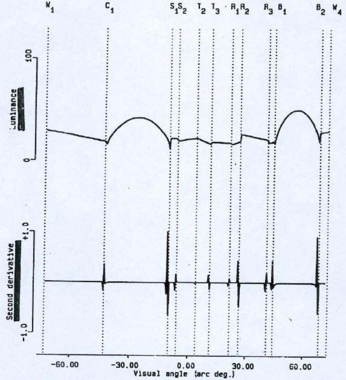
- the world consists of smooth regular surfaces with different reflectance properties where
- a discontinuity in image brightness corresponds to a discontinuity in:
  - depth
  - surface orientation
  - reflectance or
  - illumination

Edge-based approach to computer vision:

- detect discontinuities in image brightness (edges) and
- characterize these with respect to the physical phenomena that gave rise to them.

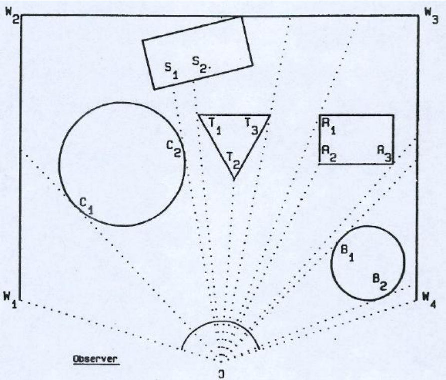
Edge detection in idealized noise free situation

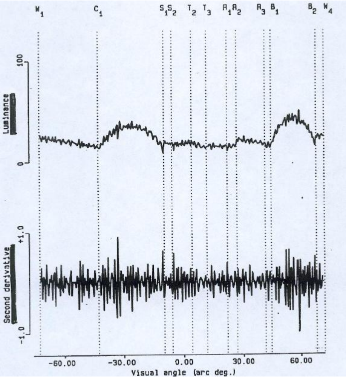




Luminance and second order derivative.

Figures from Watt (1988) Visual Processing: Computational, Psychophysical and Cognitive Research, Lawrence Erlbaum.



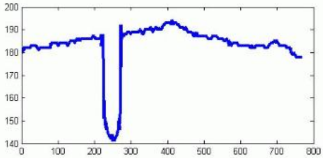


Noisy luminance and second order derivative.

Figures from Watt (1988) Visual Processing: Computational, Psychophysical and Cognitive Research, Lawrence Erlbaum.

Why edges?

- Edge features constitute important features to humans
- Independent of illumination
- Easy to detect computationally
- Used to form higher level features (lines, curves, corners, etc)
- Natural primitives in CAD-like models of man-made objects



How do edges look in practice?

Idealized models:

Step edge **Ramp edge** **Line** **Ridge**

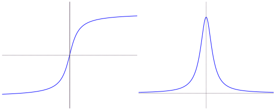
In practice, edges are blurred and noisy:

Problem: Notion of discontinuity does not exist for discrete data!

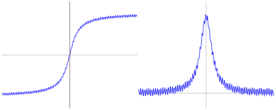
Fundamental problem

Differentiation is ill posed — an arbitrary small perturbation in the input can lead to arbitrarily large perturbation in the output:

Ex: $f(x) = \arctan(x) \qquad f'(x) = \frac{1}{1+x^2}$



$$f(x) = \arctan(x) + \epsilon \sin \omega x \qquad f'(x) = \frac{1}{1+x^2} + \epsilon\omega \cos \omega x \; .$$



The difference $\epsilon\omega \cos \omega x$ can be arbitrarily large if $\omega >> 1/\epsilon$.

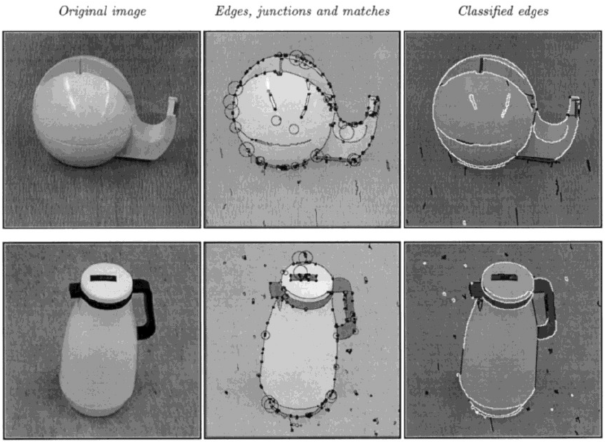
Noise reduction: Smoothing

- Basic idea: Precede differentiation by smoothing.
- Trade-off problem:
  - increasing amount of smoothing:
    - stronger suppression of noise,
    - higher distortions of “true” structures
  - decreasing amount of smoothing:
    - more accurate feature detection,
    - higher number of “false positives”.

Basic methods for edge detection

- Linear:
  - Differentiation (derivatives)
  - High-pass filtering
  - Matching with model patterns
- Non-linear:
  - Fitting of parameterized edge models
  - Non-linear diffusion
- Common approach:
  1. Detect edge points
  2. Link these to polygons
  3. Abstraction: Fit to model (straight lines, splines, ellipses)

Examples of abstracted edge descriptors



In right column: Black lines = straight lines, White curves = curved edges

Figures from Lindeberg and Li (1997) “Segmentation and classification of edges using minimum description length approximation and complementary junction cues”, Computer Vision and Image Understanding, 67(1): 88–98.

Edge attributes and problems

- Attributes:
  - Position
  - Orientation
  - Strength
  - Diffuseness (width)
- Problems:
  - Image noise
  - Interference (nearby structures at different scales)
  - Physical interpretation

Laplacian "edge detection"

Motivations

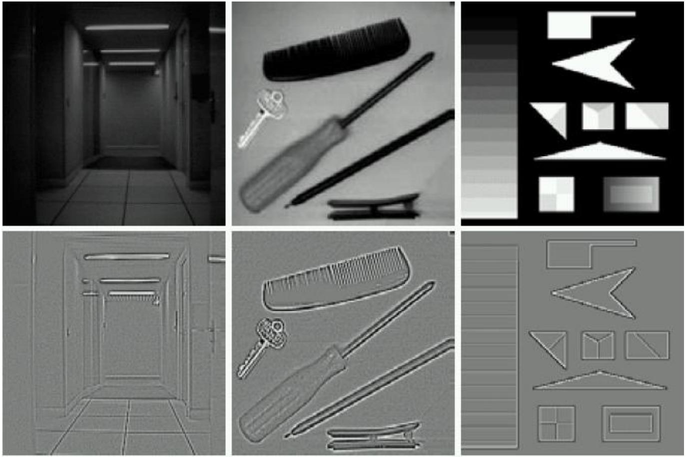
- For 1-D signals, edges correspond to peaks in the first-order derivative and to zero-crossings in the second-order derivative
- For 2-D signals, the Laplacian operator $\nabla^2 L = L_{xx} + L_{yy}$ is a rotationally symmetric operator that coincides with the second-order derivative along one-dimensional straight lines

$\Rightarrow$ Attempt to detect edges by zero-crossings of the Laplacian

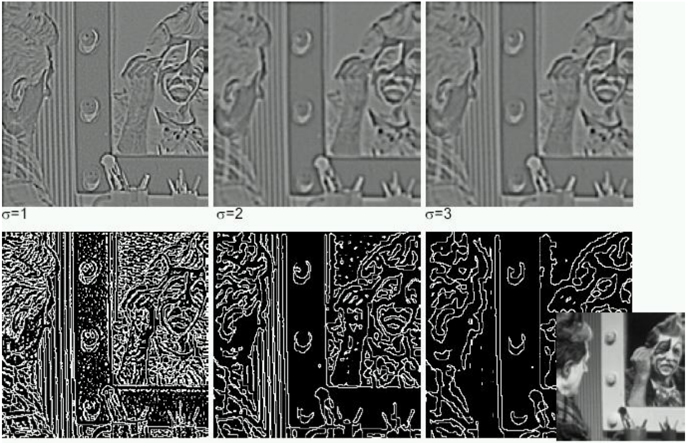
$$\nabla^2 (g(\cdot;\ t) * f) = 0$$

Proposed by Marr and Hildreth in 1980

Laplacian operator



Zero-crossings of the Laplacian



Laplacian “edge detection”

Major problems:

- Zero-crossings of the Laplacian also respond to “false edges” (corresponding to minima in the derivative response for 1-D structures)
- Poor localization for curved edges (a systematic offset that increases with the curvature of the edge)

$\Rightarrow$ Not in any way a good edge detector

Gradient based edge detection

- Gradient vector $\nabla L = \begin{pmatrix} L_x \\ L_y \end{pmatrix} = \begin{pmatrix} \frac{\partial L}{\partial x} \\ \frac{\partial L}{\partial y} \end{pmatrix}$
- Measure edge strength by gradient magnitude $|\nabla L| = \sqrt{L_x^2 + L_y^2}$
- Convolve image by appropriate kernel prior to derivative computations

Gradient estimation

- Partial derivatives estimated by difference operators:

$$L_x(x) \approx \frac{L(x+h,y)-L(x-h,y)}{2h} \qquad \text{filter mask: } \begin{pmatrix} 0 & 0 & 0 \\ -1/2 & 0 & 1/2 \\ 0 & 0 & 0 \end{pmatrix} \text{ for } h = 1$$

$$L_y(x) \approx \frac{L(x,y+h)-L(x,y-h)}{2h} \qquad \text{filter mask: } \begin{pmatrix} 0 & 1/2 & 0 \\ 0 & 0 & 0 \\ 0 & -1/2 & 0 \end{pmatrix} \text{ for } h = 1$$

- Gradient direction:

$$\theta = \arctan \frac{L_y}{L_x} + n\pi = \mathrm{atan2}(L_x, L_y)$$

Accuracy of derivative approximations by local Taylor expansions

Taylor expansions:

$$f(x+h) = f(x) + hf'(x) + \tfrac{1}{2}h^2 f''(x) + \dots$$
$$f(x-h) = f(x) - hf'(x) + \tfrac{1}{2}h^2 f''(x) + \dots$$

By subtracting the second equation from the first we obtain

$$f'(x) = \tfrac{f(x+h)-f(x-h)}{2h} + \mathcal{O}(h^2)$$

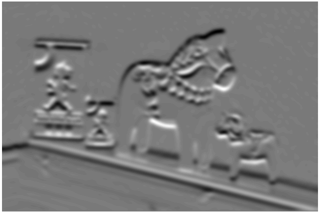
Higher order derivative approximations can constructed and analysed in a corresponding manner

Gradient estimation

original image

horizontal derivative

vertical derivative



Canny edge detection (1986)

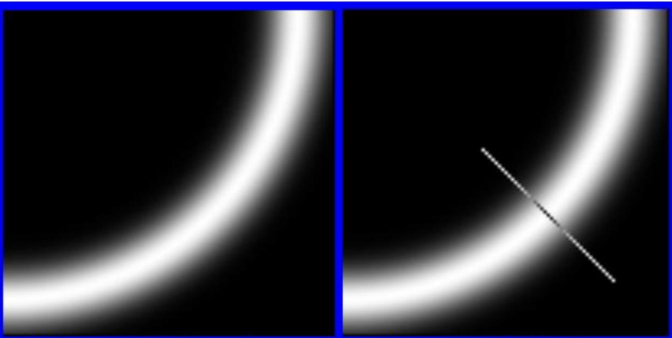
Typical problem: If you try to perform edge detection by thresholding on the edge strength, then the resulting edges may be several pixels wide

1. Convolve image by smoothing kernel
2. Estimate edge strength and edge normal direction
3. Threshold on edge strength and preserve only edge points that are local extrema of the edge strength in the edge normal direction

   (non-maximum suppression implemented by local search)

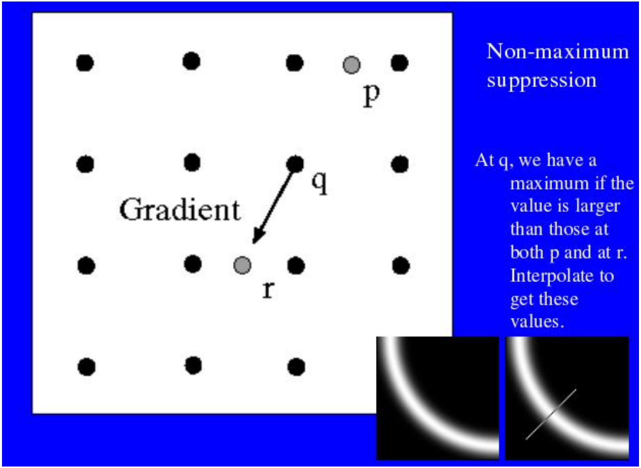
Canny derived an optimal smoothing kernel for handling trade-off issues in edge detection and then demonstrated that this kernel can be well approximated by a Gaussian kernel

Non-maximum suppression



We wish to mark points along the curve where the magnitude is biggest. We can do this by looking for a maximum along a slice normal to the curve (non-maximum suppression). These points should form a curve. There are then two algorithmic issues: at which point is the maximum, and where is the next one?

Non-maximum suppression



Edge linking

Link neighbouring edge pixels to connected contours:
Each point has strength $|\nabla L|$ and orientation θ

Algorithm: $\forall$ pixels

if ($|\nabla L| >$ threshold) [and is local maximum]

$\forall$ neighbour's

if ($|\nabla L|$ of neighbour $>$ threshold) [and is local maximum]

and ($| \Theta_{this} - \Theta_{neighbour} | <$ threshold)

link these pixels to be connected

Combine with efficient traversal procedure and mechanism for closing gaps.

Hysteresis thresholding

- Problem:
  Thresholding on gradient magnitude may lead to fragmented edges:
  - Too many maxima due to noise if threshold too low
  - Edges may disconnect at weak edge points if threshold too low
- Addressed by two thresholds T_{low} and T_{high}:
  1. In first phase, edge points only allows if edge strength $> T_{low}$
  2. In second phase, edge segments only preserved if at least some point on the segment has edge strength $> T_{high}$

Hysteresis thresholding

Figure 4.6 Output of HYSTERESIS_THRESH run on Figure 4.5, showing the effect of varying the filter's size. Left to right: $\sigma_f = 1, 2, 3$ pixel. The grey levels has been inverted (black on white) for clarity.

Differential edge detection (Lindeberg 1993)

Non-maximum suppression:

Edge point = point where the gradient magnitude assumes a maximum in the gradient direction.

This can be expressed in terms of :

- the gradient: $\nabla L = (L_x, L_y)^T$, the gradient magnitude: $\sqrt{L_x^2 + L_y^2}$.
- the normalized gradient direction: $e_v = \frac{\nabla L}{|\nabla L|} = \frac{(L_x, L_y)^T}{\sqrt{L_x^2 + L_y^2}}$
- Directional derivative in any direction α: $e_\alpha = (\cos\alpha, \sin\alpha)$:
  $\partial_\alpha = \cos\alpha\,\partial_x + \sin\alpha\,\partial_y$
- Directional derivative in gradient direction:
  $\partial_v = \left(\frac{L_x}{\sqrt{L_x^2+L_y^2}}\partial_x + \frac{L_y}{\sqrt{L_x^2+L_y^2}}\partial_y\right)$ with $L_v = \partial_v L = \sqrt{L_x^2 + L_y^2} = |\nabla L|$

Differential geometric edge definition

Requirements for gradient magnitude to be maximal in gradient direction:

$$\begin{cases} \partial_v(L_v) = 0 \\ \partial_{vv}(L_v) < 0 \end{cases} \quad \text{or} \quad \begin{cases} L_{vv} = 0 \\ L_{vvv} < 0 \end{cases}$$

In terms of coordinates:

$$\begin{aligned} L_{vv} &= (\cos\alpha\,\partial_x + \sin\alpha\,\partial_y)^2 L \\ &= \cos^2\alpha\, L_{xx} + 2\cos\alpha\sin\alpha\, L_{xy} + \sin^2\alpha\, L_{yy} \\ &= \tfrac{L_x^2}{L_x^2+L_y^2} L_{xx} + 2\tfrac{L_x L_y}{L_x^2+L_y^2} L_{xy} + \tfrac{L_y^2}{L_x^2+L_y^2} L_{yy} \\ &= \tfrac{(L_x^2 L_{xx} + 2 L_x L_y L_{xy} + L_y^2 L_{yy})}{L_x^2+L_y^2} = 0 \end{aligned}$$

Since the denominator is irrelevant, the edges are given by:

$$\begin{cases} \tilde{L_{vv}} = L_x^2 L_{xx} + 2 L_x L_y L_{xy} + L_y^2 L_{yy} = 0 \\ \tilde{L_{vvv}} = L_x^3 L_{xxx} + 3 L_x^2 L_y L_{xxy} + 3 L_x L_y^2 L_{xyy} + L_y^3 L_{yyy} < 0 \end{cases}$$

Differential edge detection

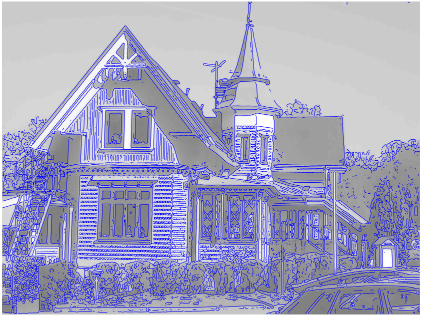
Differential edge detection in practice:

1. Convolve image by (discrete) Gaussian kernel
2. Compute partial derivatives up to order three and combine these into the differential invariants $L_{\tilde{v}\tilde{v}}$ and $L_{\tilde{v}\tilde{v}\tilde{v}}$ at every image point
3. Search for the zero-crossings of $L_{\tilde{v}\tilde{v}}$ that satisfy $L_{\tilde{v}\tilde{v}\tilde{v}} < 0$

   Gives subpixel accuracy and connected edge segments automatically

   Avoids issues of orientation estimation and handling as well as edge tracking in discrete non-maximum suppression
4. Can be combined with either a single low threshold on the gradient magnitude or hysteresis thresholding using two thresholds

Differential edge detection



Summary of good questions I

- What is meant by the concept of receptive field?
- Why is the notion of scale important in image analysis and computer vision?
- What is a scale-space representation? On what basis is it constructed?
- What structural requirements are natural to impose on early visual operations?
- What is meant by a Gaussian derivative? How and why are such operators important for vision?
- Show that convolution with the 2-D Gaussian kernel satisfies the 2-D diffusion equation.

Summary of good questions II

- Why is edge detection important for image understanding?
- Why is edge detection difficult in practice?
- What families of methods exist for edge detection?
- What information do image gradients provide?
- How does the Canny edge detector work?
- What is differential edge detection? How does this method compare to Canny edge detection?
- What is hysteresis thresholding?
- What should the image derivatives be equal to on edge points?
- Derive the differential invariants used for expressing differential edge detection.

Literature for further reading

- Lindeberg (2009) “Scale-space”, Encyclopedia of Computer Science and Engineering, John Wiley and Sons, Volume IV, pages 2495–2504, `http://dx.doi.org/10.1002/9780470050118.ecse609`

  Preprint available from `http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-40202`
- Lindeberg (1994) “Scale-space theory: A basic tool for analysing structures at different scales”, J. of Applied Statistics, 21(2): 224–270, `http://dx.doi.org/10.1080/757582976`

  Preprint available from `http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-40216`
- Lindeberg (2013) “A computational theory of visual receptive fields”, *Biological Cybernetics*, 107(6): 589-635. `http://dx.doi.org/10.1007/s00422-013-0569-z` (Open Access)