Today

Early vision on a single image

- Features

Reminder: Abstract of Final Project

Sunday, Feb 23 at 11:59pm, EST. Late submission penalty applies.

Include

- Title and name(s) of the team member(s)
- Topic: a research project or a survey
- A brief introduction on the background
- A brief discussion of scope of work
- Timeline and project management for a teamwork

At most one page

Each team only needs one abstract

Final Project Topics

Option 1: A complete research project

- Introduction (problem formulation/definition)
- literature review
- the proposed method and analysis
- experiment
- conclusion
- reference

Option 2: A survey research

- A well-defined problem or topic
- a complete list of previous (typical) work on this problem
- clearly and briefly describe it
- analyze each methods/groups and compare them
- give the conclusion and list of references (15+)

Final Project Requirement

Special requirements

- **Decide topic and write a one-page abstract**
- Progress report (discuss with the instructor)
- Research work and report writing
- Oral presentation (class, vision seminar, etc.)

Teamwork (2-person team) is acceptable ONLY for Option 1

- talk to the instructor first
- under a single topic, each member must have his/her own specific subtopic
- a combined report, but each member needs to clearly show his/her own contributions
- combined presentation if necessary

Project Requirement

Notes:

- you are encouraged to incorporate your own expertise in, but the project topic must be related to the content of this course
- discuss with the instructor on topic selection, progress, writing, and presentation
- use the library and online resource

Major research journals and conferences on computer vision

- International Conference on Computer Vision (ICCV)
- IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)
- IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)
- International Journal on Computer Vision (IJCV)
- You may find useful literature in them for your project

Image Features

Establishing correspondence

- 3D reconstruction
- Object tracking

Object recognition & classification

What are good features?

Wikipedia

www.shutterstock.com

World Wildlife Fund

iGuides.org

- Robust to variations, e.g., view point and illumination
- Reliable

Image Features

- **Gradient and edges**
- **Corners**
- **Line and curves**
- **Textures**
- **Advanced features**
  - Human crafted features
    - Haar, Gabor, SIFT, HOG, etc.
  - Learned features
    - Sparse coding
    - Deep features – features learned by deep learning

First-order Derivative based Edge Detection

Gradient $$\nabla f(x,y) = grad(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \partial f / \partial x \\ \partial f / \partial y \end{bmatrix}$$

Edge strength $$M(x,y) = \sqrt{g_x^2 + g_y^2}$$

$$\approx |g_x| + |g_y|$$

Gradient direction $$\alpha(x,y) = \tan^{-1}\left[\frac{g_y}{g_x}\right]$$

Edge direction $\alpha - 90^\circ$

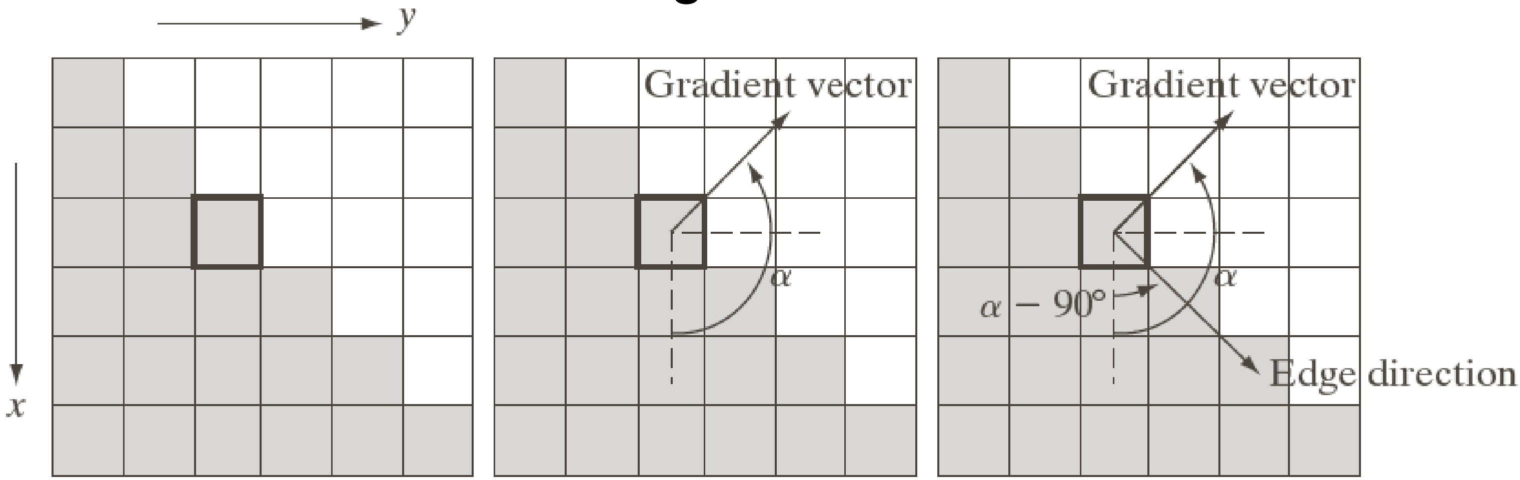


Figure from “Digital Image Processing”, Gonzalez and Woods

First-order Derivative

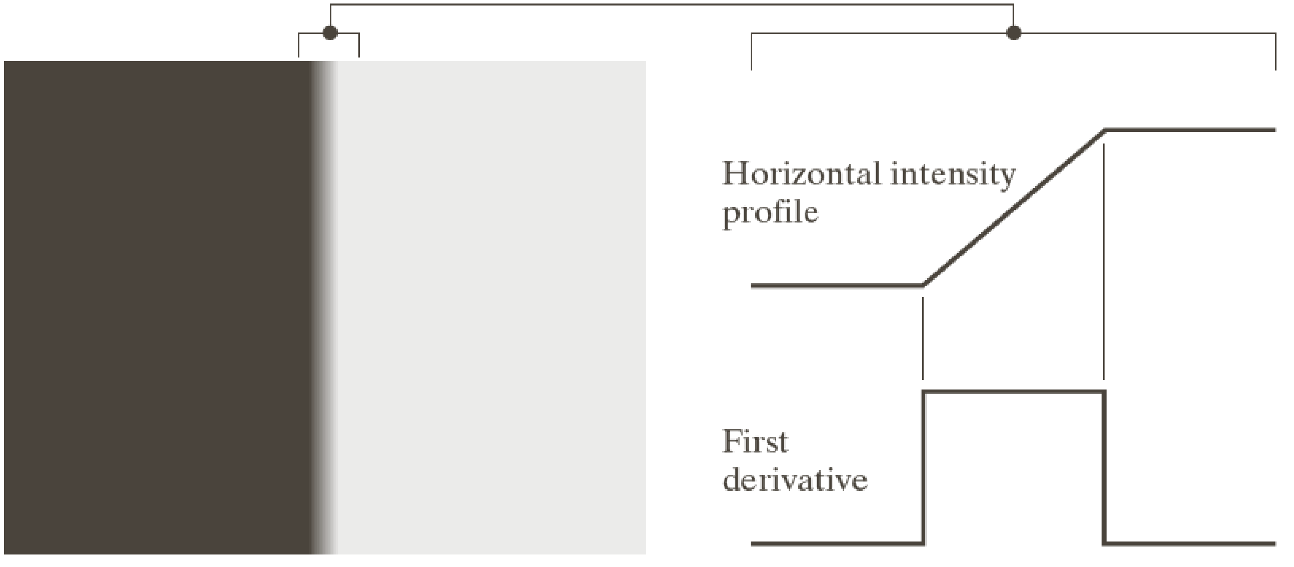


Figure from “Digital Image Processing”, Gonzalez and Woods

General strategy

- determine image gradient
- mark points where gradient magnitude is particularly large w.r.t neighbors

Masks for Calculating the Gradient (3x3)

Gradient in vertical/horizontal

Gradient in diagonal

Prewitt

Gradient in vertical/horizontal:

−1	−1	−1
0	0	0
1	1	1

−1	0	1
−1	0	1
−1	0	1

Gradient in diagonal:

0	1	1
−1	0	1
−1	−1	0

−1	−1	0
−1	0	1
0	1	1

Sobel

Gradient in vertical/horizontal:

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Gradient in diagonal:

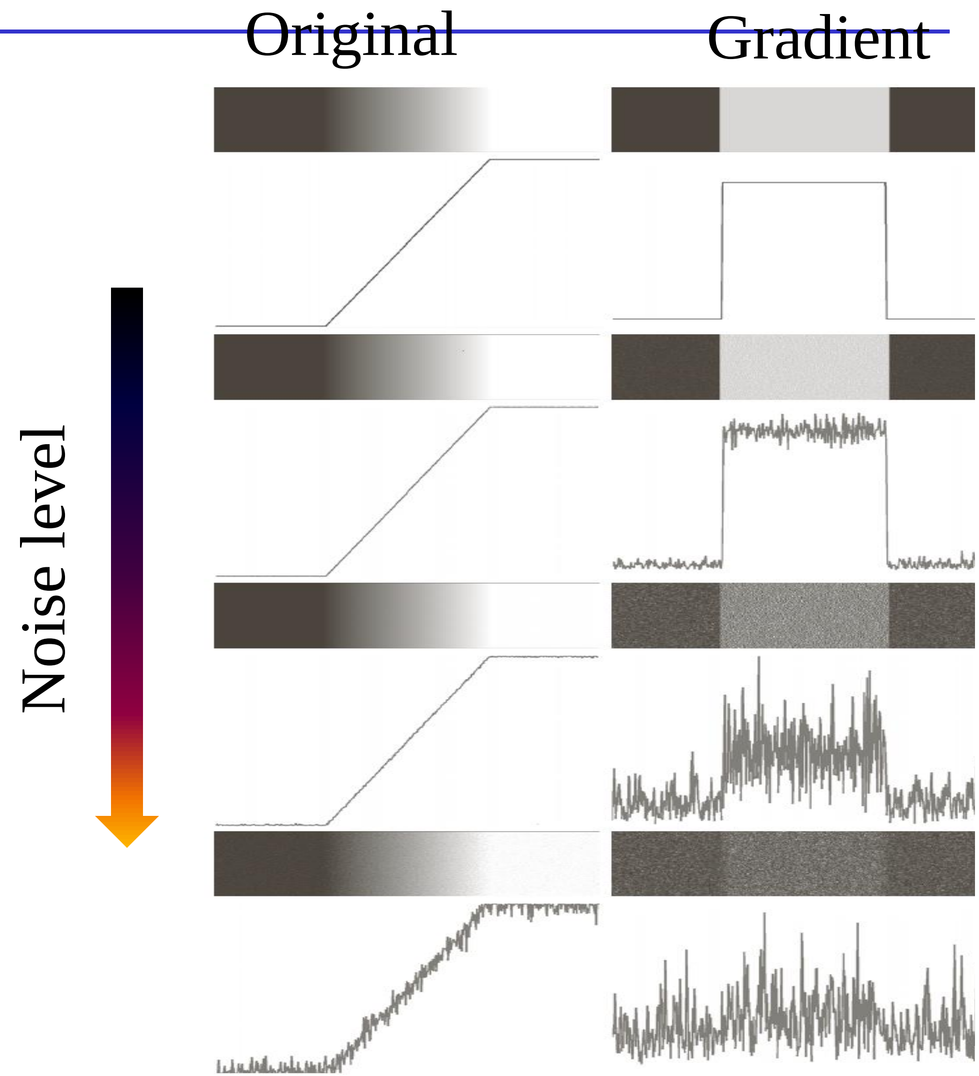
0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Sobel operator performs edge detection and smoothing simultaneously.

Smoothing and Differentiation

Issue: noise



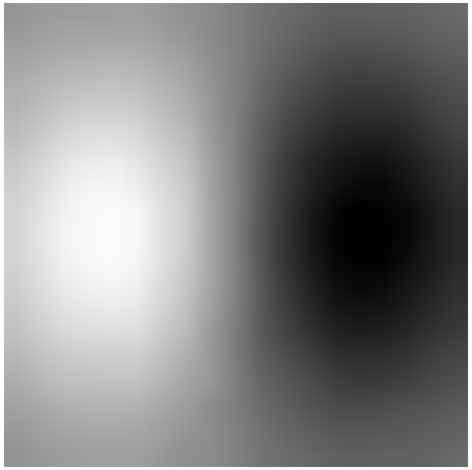
1. Image Smoothing
2. Detecting edge points
3. Edge localization

Figure from "Digital Image Processing", Gonzalez and Woods

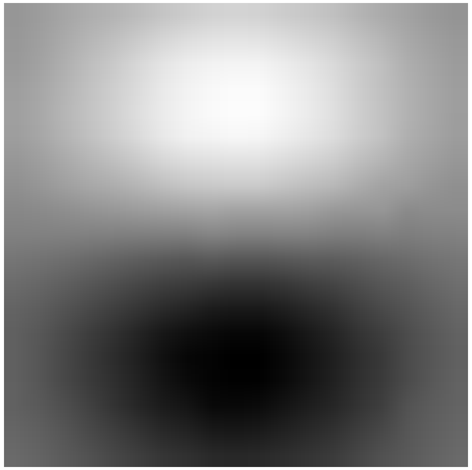
Smoothing and Differentiation

Solution

- Smooth before differentiation
- Two convolutions for smooth and differentiate?
- Actually, no - we can use a derivative of Gaussian filter
  - because differentiation is convolution, and convolution is associative

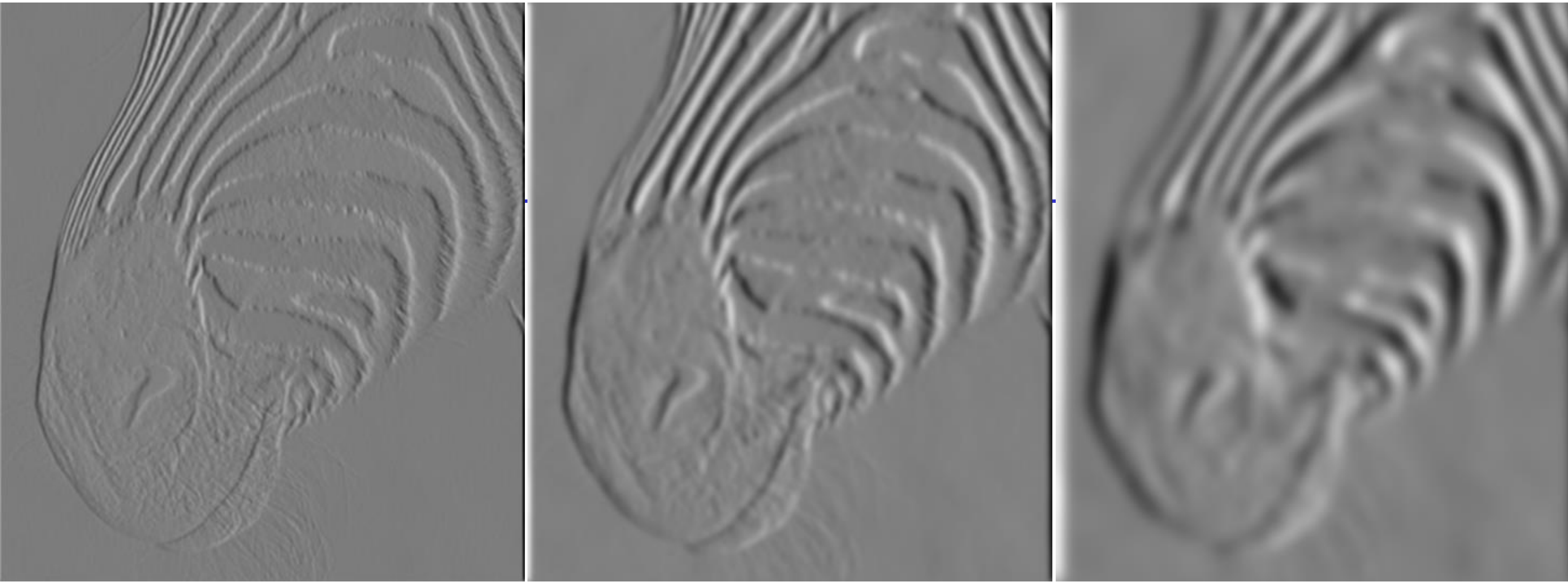


$$\frac{\partial G_\sigma}{\partial x}$$



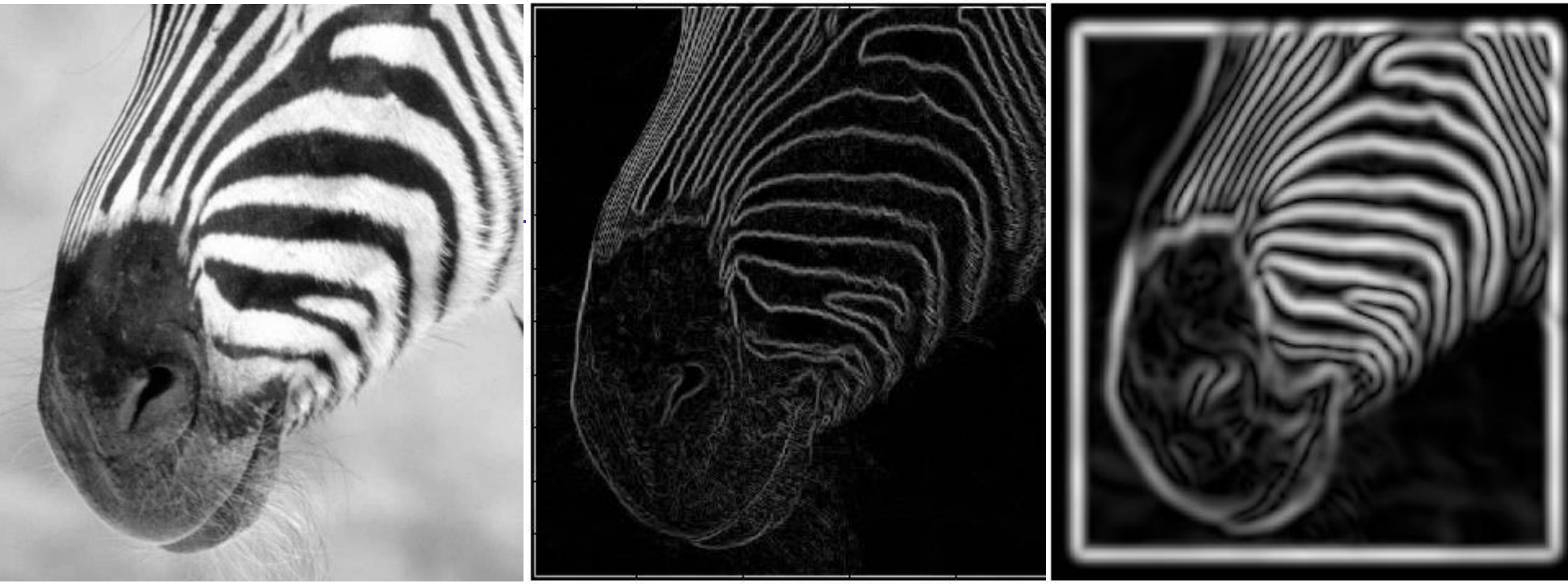
$$\frac{\partial G_\sigma}{\partial y}$$

Derivative of Gaussian



σ=1 pixel σ=3 pixels σ=7 pixels

The scale of the smoothing filter affects derivative estimates, and also the semantics of the edges recovered.

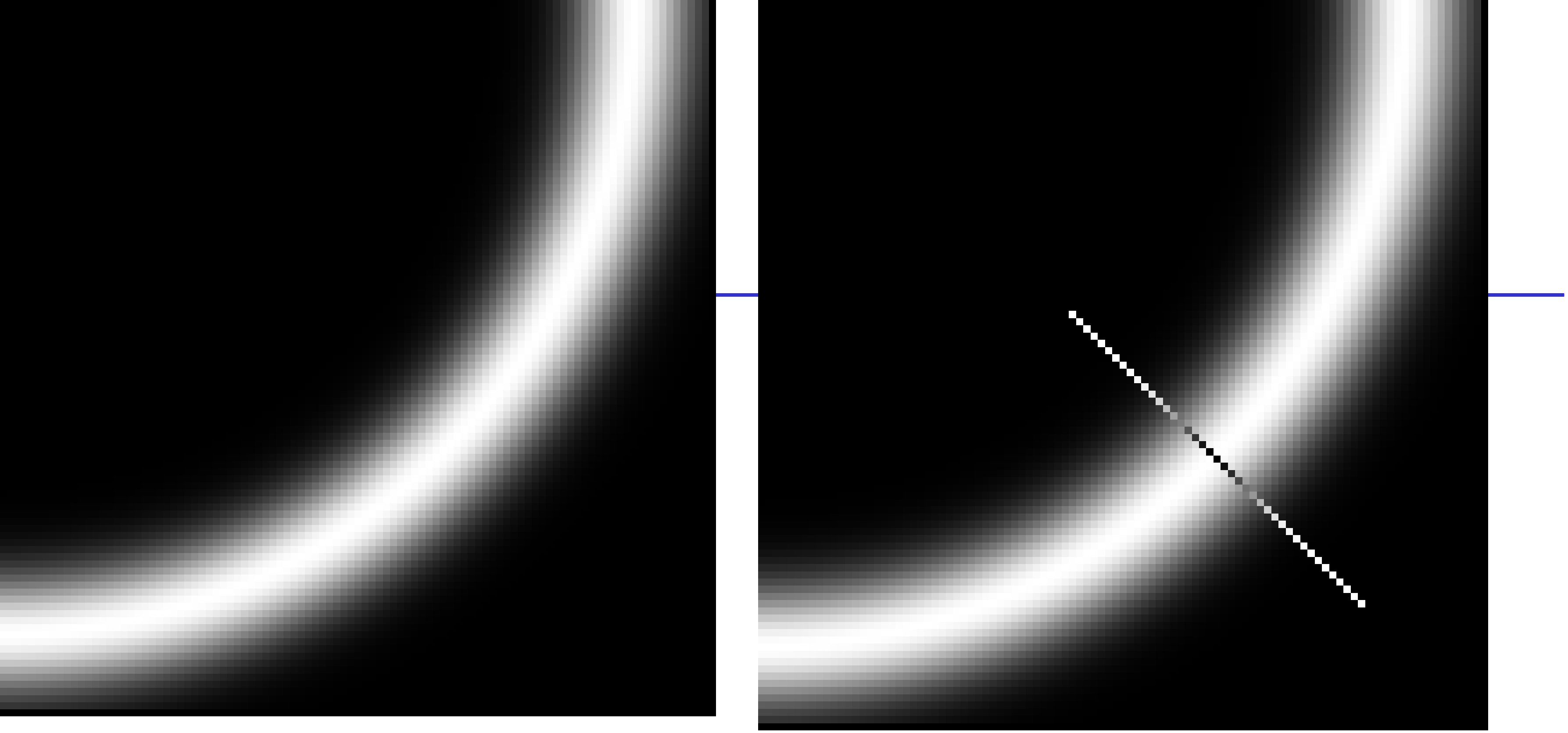


There are three major issues:

1) The gradient magnitude at different scales is different; which should we choose?

2) The gradient magnitude is large along thick trail; how do we identify the significant points?

3) How do we link the relevant points up into curves?



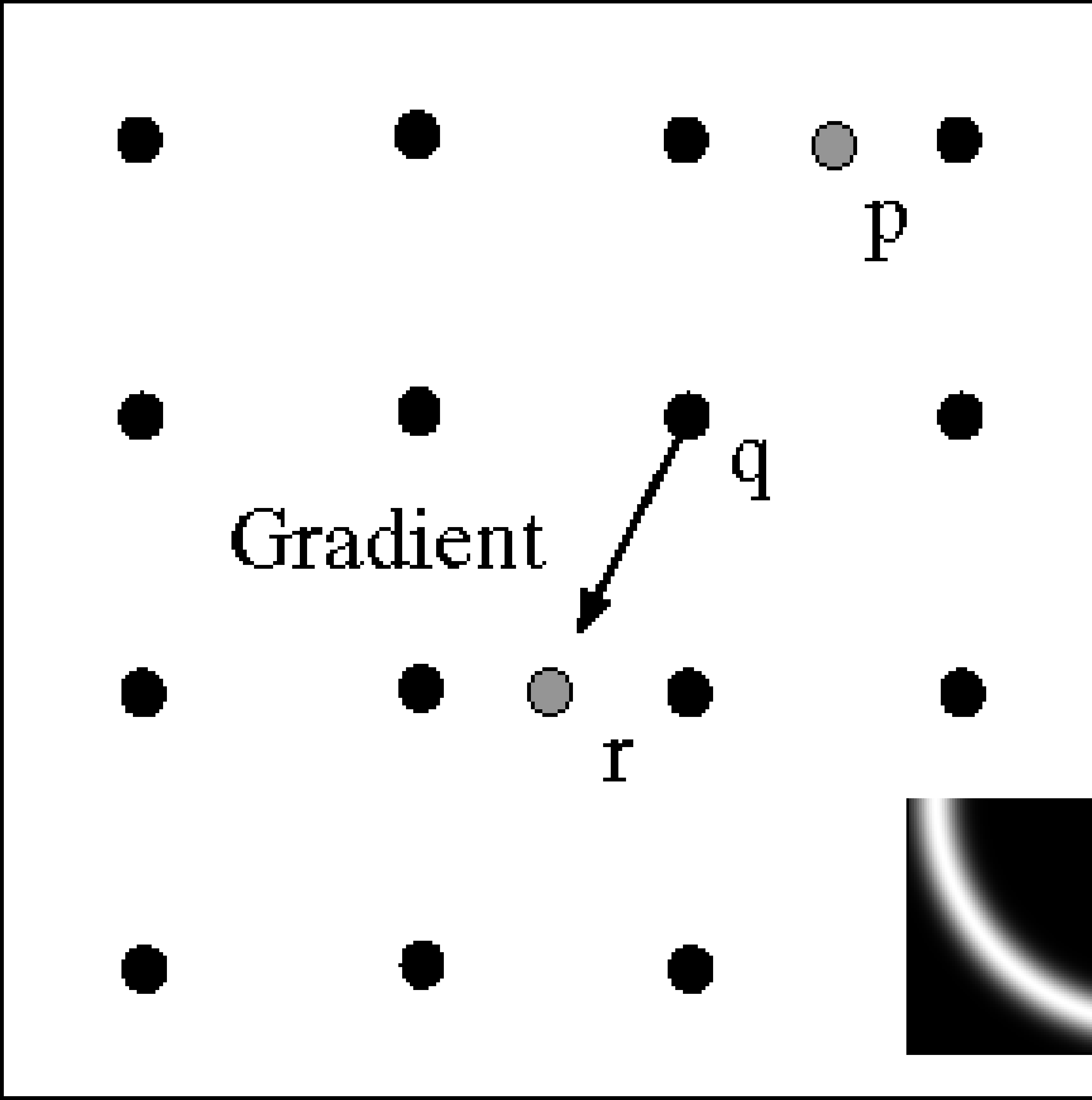
Edge

- Mark points along the curve where the magnitude is biggest.
-Look for a maximum along a slice normal to the curve (non-maximum suppression). These points should form a curve.

Two questions:

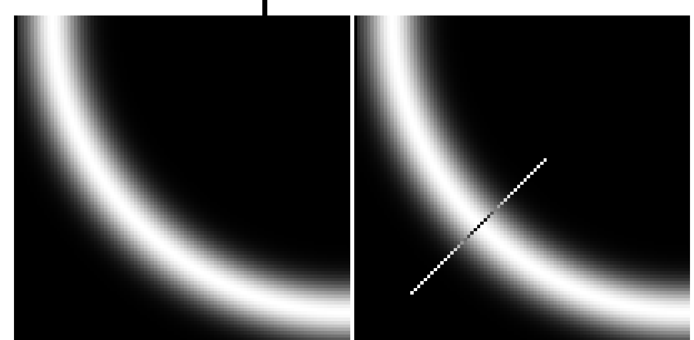
- How do we define the slice direction
- Where is the next one

Gradient direction

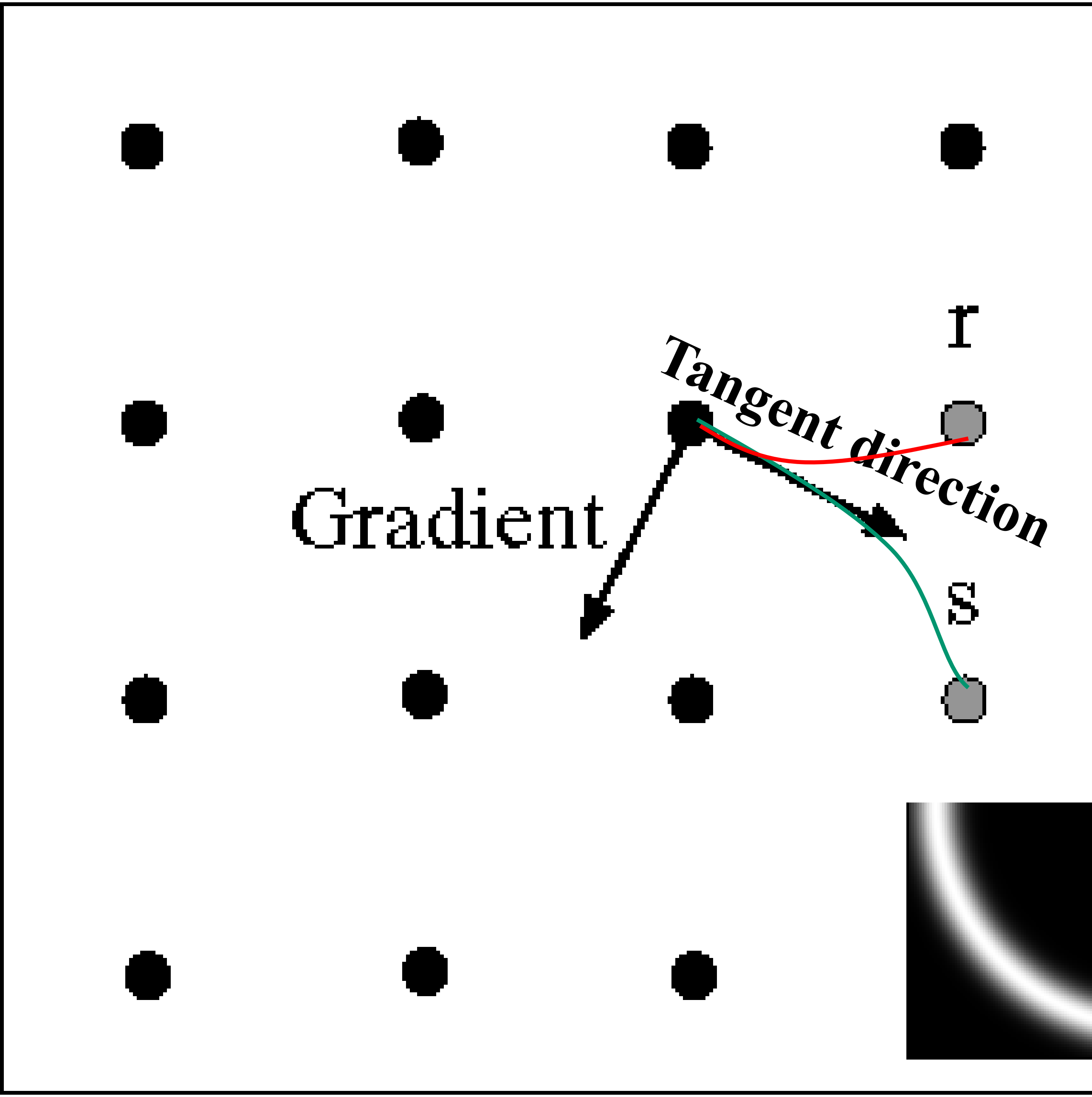


Non-maximum Suppression

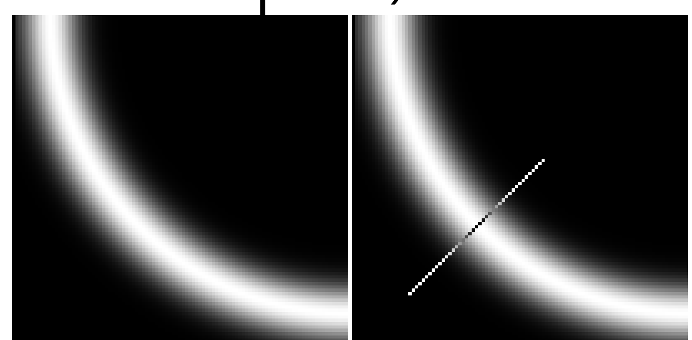
At *q*, we have a maximum if the value is larger than those at both *p* and at r. Interpolate to get these values.



Predicting the next edge point



Assume the marked point is an edge point. Then we construct the tangent to the edge curve (which is normal to the gradient at that point) and use this to predict the next points (here either *r* or *s*).



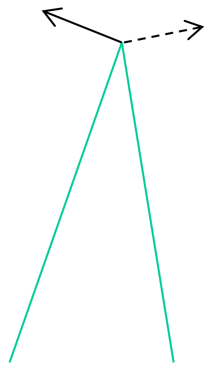
Remaining Issues

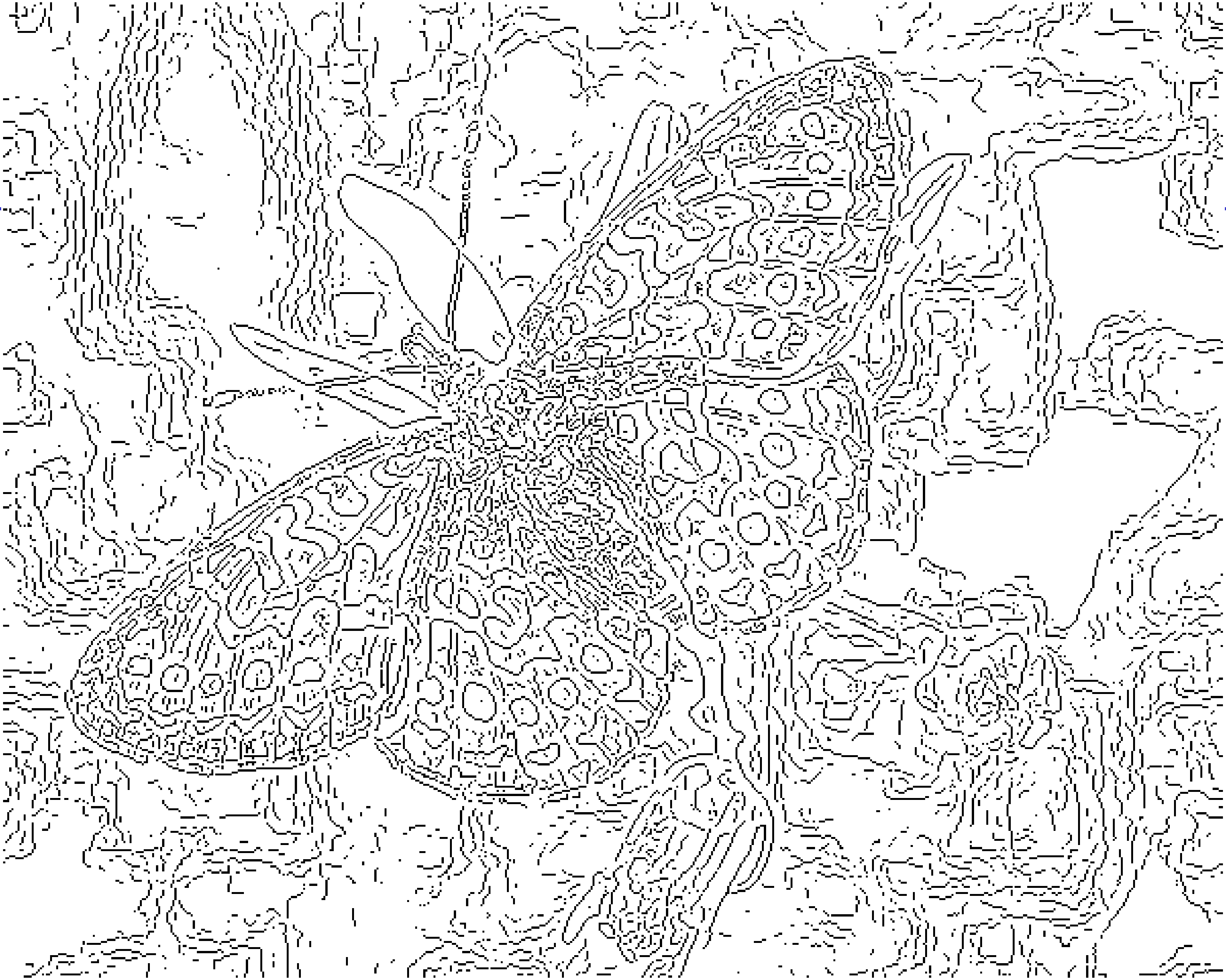
Check that maximum value of gradient value is sufficiently large

- use a high threshold to start edge curves and a low threshold to continue them.

Notice:

- Something nasty is happening at corners
- Scale affects contrast
- Edges aren’t bounding contours





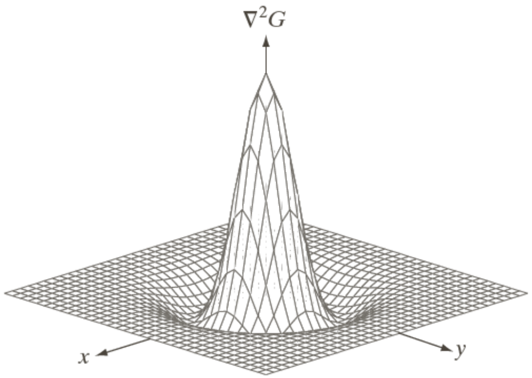
Fine scale, high threshold

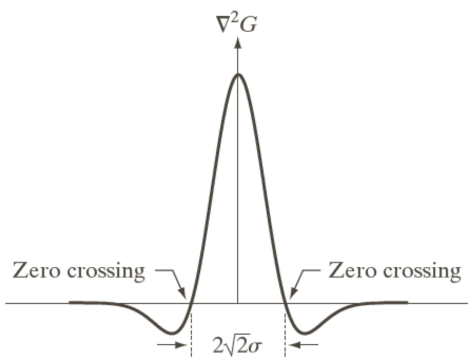
Coarse scale, high threshold

Coarse scale, low threshold

Second-order based Edge Detection: Laplacian of Gaussian (LOG)

Another way to detect an extrema in first derivative is to look for a zero-crossing in second derivative (Laplacian)





0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

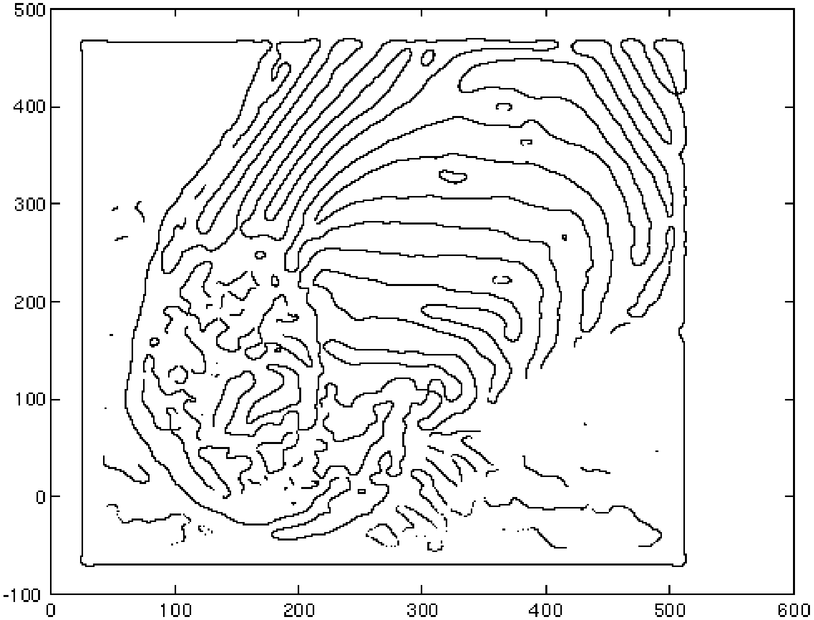
Bad idea to apply a Laplacian without smoothing

- smooth with Gaussian, and then apply Laplacian
- this is the same as filtering with a Laplacian of Gaussian filter

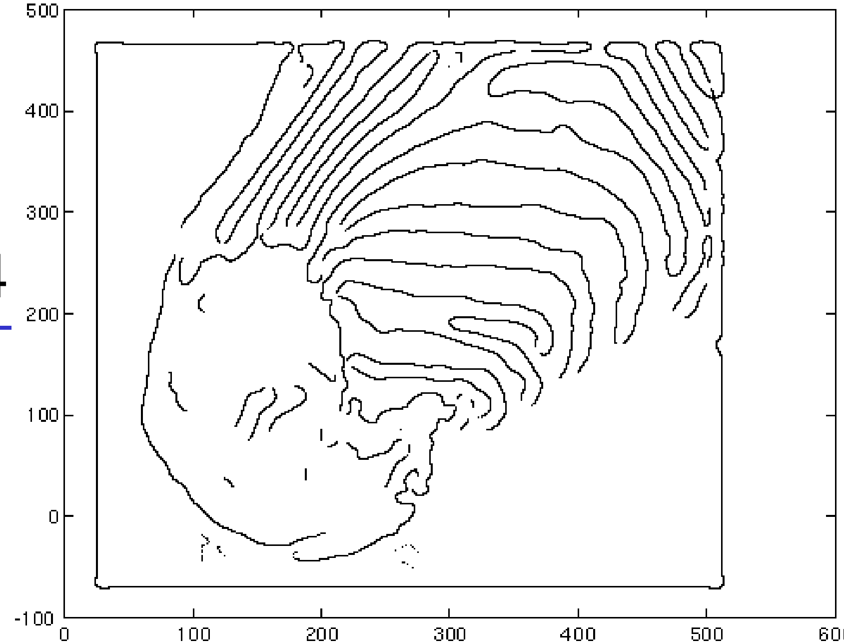
Now mark the zero-crossing points *where there is a sufficiently large derivative, and enough contrast*

$$LoG: \nabla^2 G(x,y) = \left[\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Figure from “Digital Image Processing”, Gonzalez and Woods



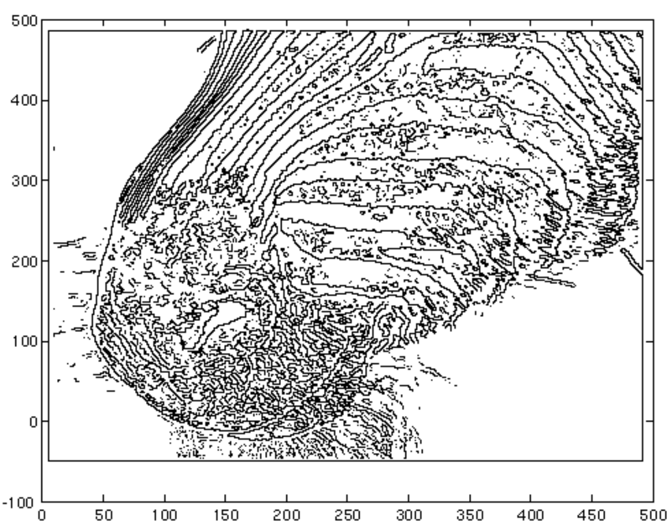
sigma=4



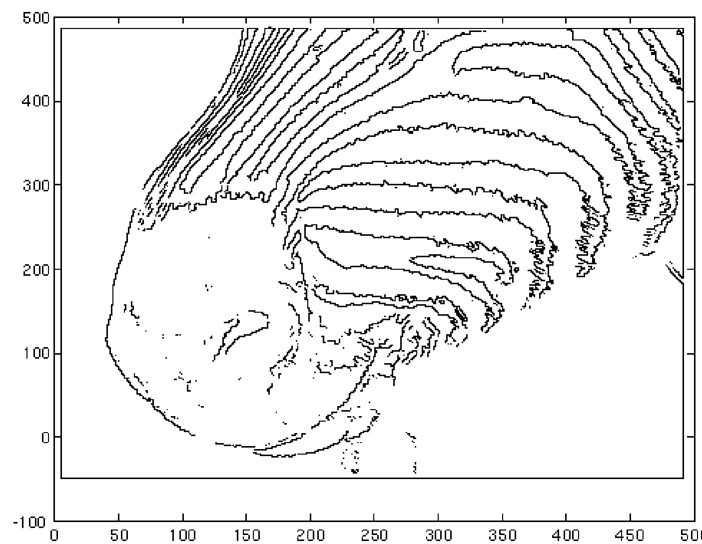
contrast=1

LOG zero crossings

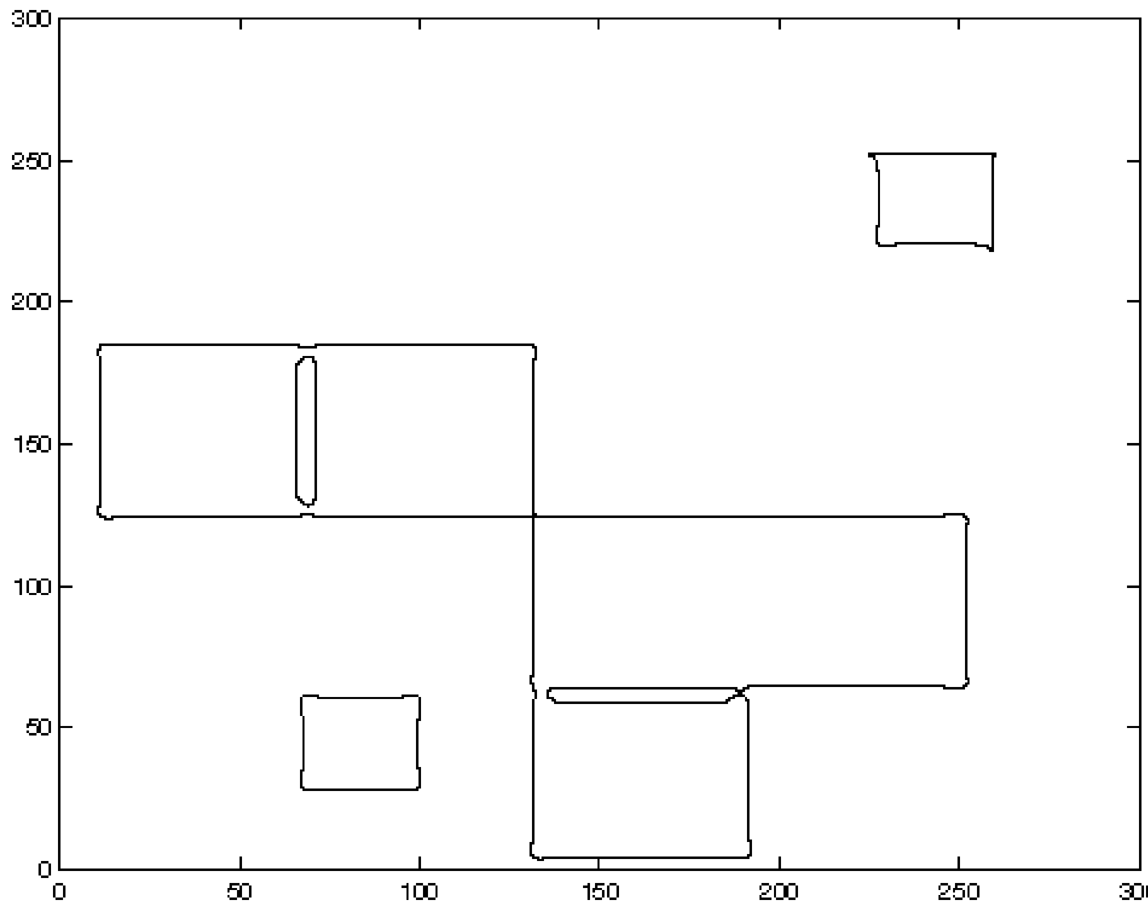
contrast=4



sigma=2



We still have unfortunate behavior at corners



Orientation Representations

The gradient magnitude is affected by illumination changes

- but its direction isn't

We can describe image patches by the swing of the gradient orientation

Important types:

- constant window
  - small gradient magnitudes
- edge window
  - few large gradient mags in one direction
- flow window
  - many large gradient mags in one direction
- corner window
  - large gradient magnitudes that swing

Representing Windows

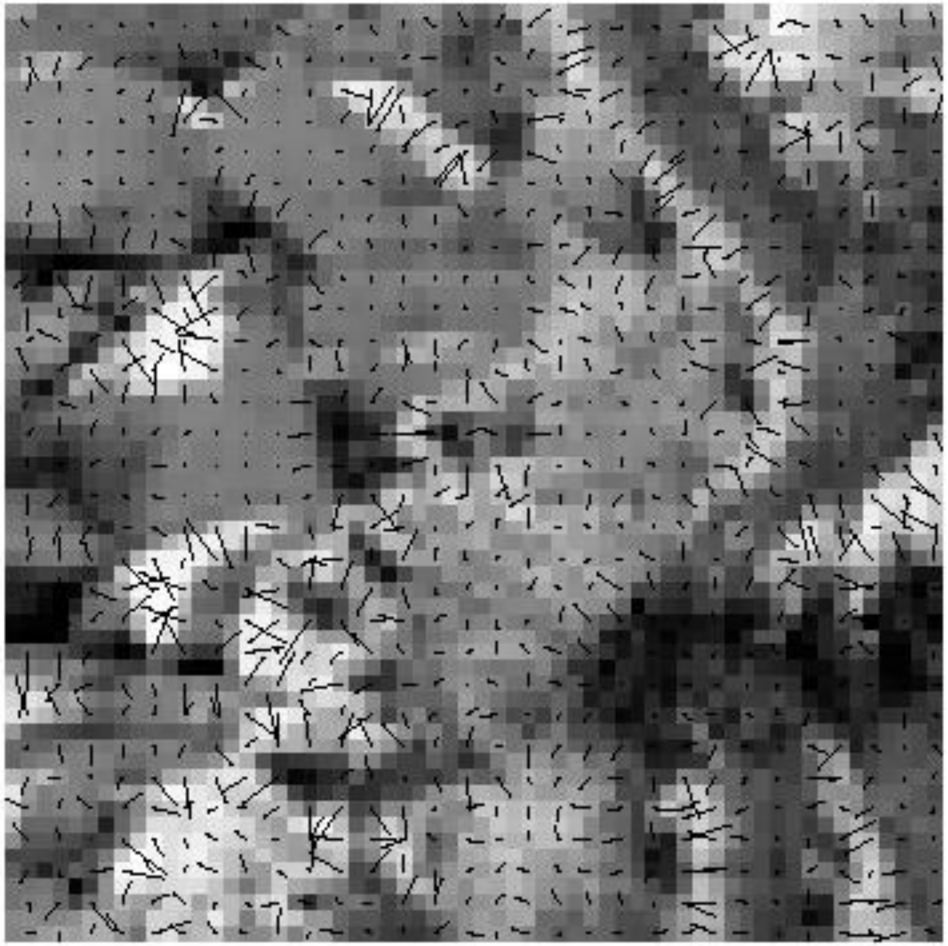
Types

- constant
  - small eigenvalues
- Edge
  - one medium, one small
- Flow
  - one large, one small
- corner
  - two large eigenvalues

$$H = \sum_{window} (\nabla I)(\nabla I)^T$$

$$= \sum_{window} \begin{Bmatrix} \left(\frac{\partial G_\sigma}{\partial x} \otimes I\right)\left(\frac{\partial G_\sigma}{\partial x} \otimes I\right) & \left(\frac{\partial G_\sigma}{\partial x} \otimes I\right)\left(\frac{\partial G_\sigma}{\partial y} \otimes I\right) \\ \left(\frac{\partial G_\sigma}{\partial x} \otimes I\right)\left(\frac{\partial G_\sigma}{\partial y} \otimes I\right) & \left(\frac{\partial G_\sigma}{\partial y} \otimes I\right)\left(\frac{\partial G_\sigma}{\partial y} \otimes I\right) \end{Bmatrix}$$

Details



From Edges to Boundary

Object boundary is a closed curve

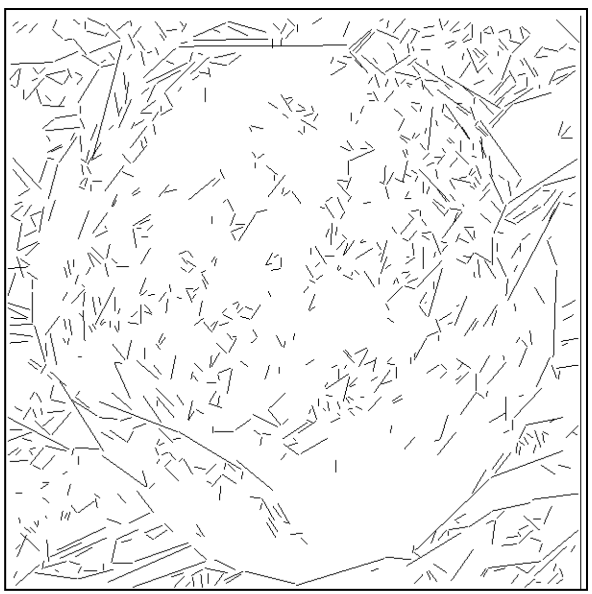
The detected edges are just fragments of boundaries. They are not connected

Edges may also come from noise

How to find the full object boundary from edge-detection output?

Edge-linking, edge-grouping, …

Brainstorm: How to Formulate and Solve this Problem?



Suggested Reading

S. Wang, T. Kubota, J. M. Siskind, J. Wang. Salient Closed Boundary Extraction with Ratio Contour, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(4):546-561, 2005

http://www.cse.sc.edu/~songwang/document/pami05.pdf