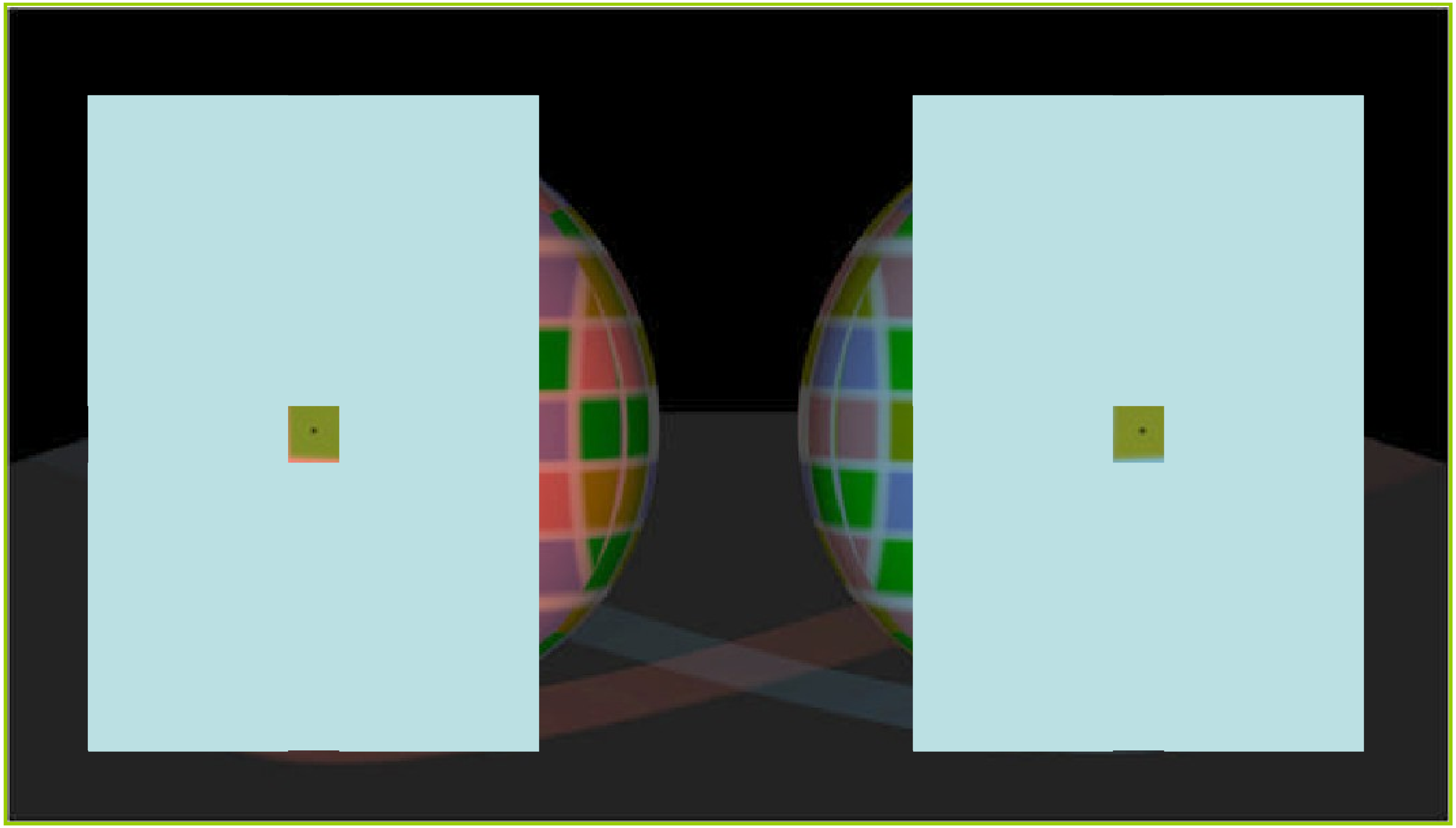
Features
Edges

Andrea Torsello
DAIS
Università Ca' Foscari
via Torino 155,
30172 Mestre (VE)

What is a Feature?

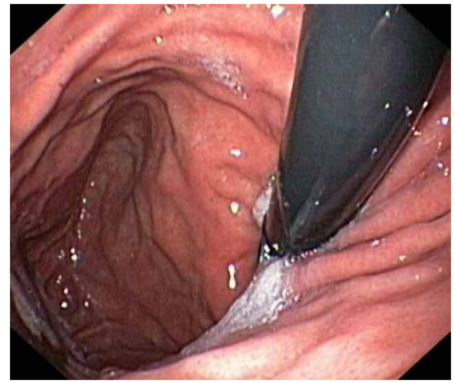


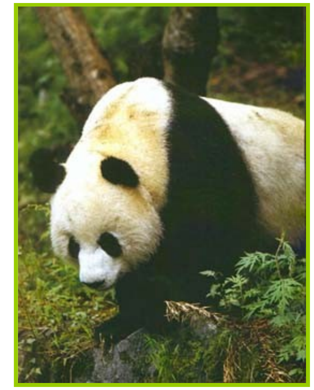
- Local, meaningful, detectable parts of the image

Features in Computer Vision

- What is a feature?
  - Location of sudden change
- Why use features?
  - Information content high
  - Invariant to change of view point, illumination
  - Reduces computational burden

What Makes For *Good* Features?

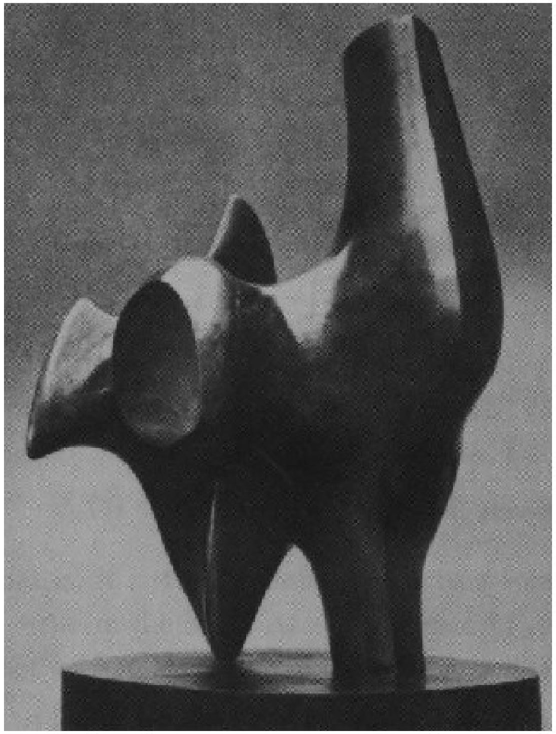
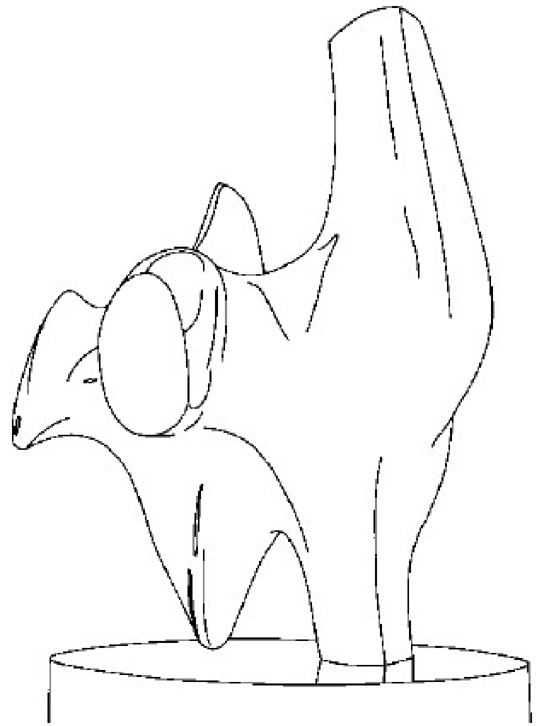




- Invariance
  - View point (scale, orientation, translation)
  - Lighting condition
  - Object deformations
  - Partial occlusion
- Other Characteristics
  - Uniqueness
  - Sufficiently many
  - Tuned to the task

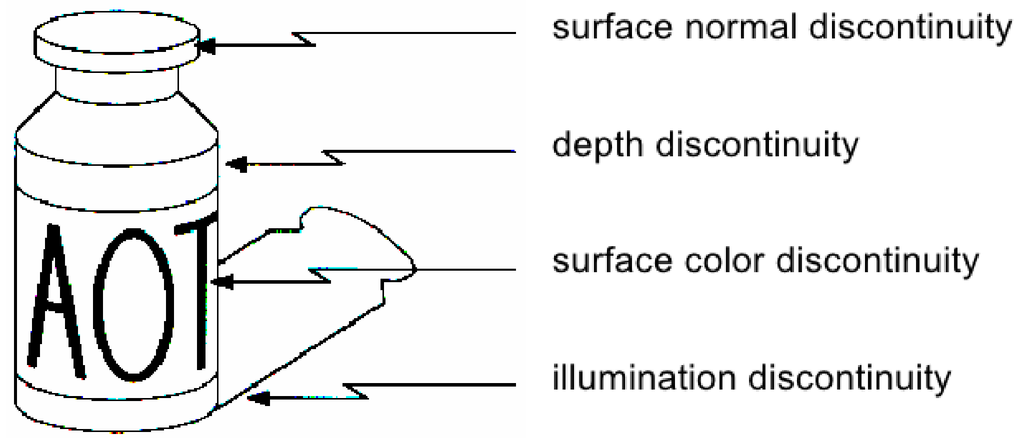
- Edges
- Corners and Junctions
- Texture
- Intensity gradients (distribution of ambient light)
- Correspondence of features across multiple images
  - Stereo
  - Flow
- First we will stay with features on a single image
- Goal: Define techniques that will “filter” out regions that have these features, and allow us to ignore most of the pixels.
- Use these features in higher level vision

Edge detection

- Convert a 2D image into a set of curves
  - Extracts salient features of the scene
  - More compact than pixels

Edges for inference



- Edges in an image have many causes
- An edge presents an opportunity to infer
- Looking at edges reduces information required
  - Look at a few pixels in a binary image as opposed to all pixels in a grayscale image
- Biological plausibility
  - Initial stages of mammalian vision systems involve detection of edges and local features

What Causes an Edge?

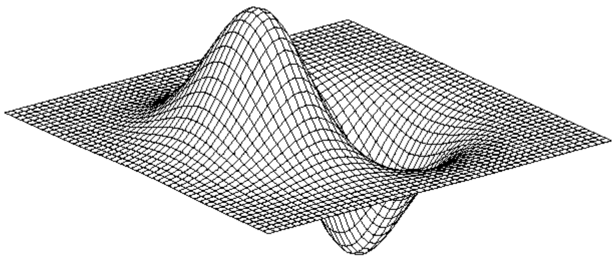
- Depth discontinuity
- Surface orientation discontinuity
- Reflectance discontinuity (i.e., change in surface material properties)
- Illumination discontinuity (e.g., shadow)

Edge detection

- How can you tell that a pixel is on an edge?
  1. Detection of short linear edge segments (edgels)
  2. Aggregation of edgels into extended edges
  3. Possibly combine the edges

- Edgel detection
  - Difference operators (linear filters)
  - Parametric-model matchers

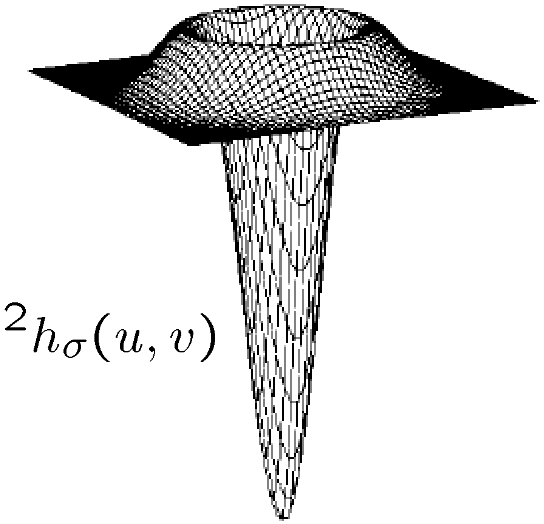
2D edge detection filters



derivative of Gaussian

$$\frac{\partial}{\partial x} h_\sigma(u, v)$$

Laplacian of Gaussian



$$\nabla^2 h_\sigma(u, v)$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0
0	1

0	−1
1	0

Roberts

−1	−1	−1
0	0	0
1	1	1

−1	0	1
−1	0	1
−1	0	1

Prewitt

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Sobel

Detection By Thresholding

UNIVERSITÀ CA' FOSCARI VENEZIA

Original image (Lena)

Norm of the gradient

Thresholding

- If the threshold is too high, important edges may be missed
- If the threshold is too low, noise may be mistaken for a genuine edge

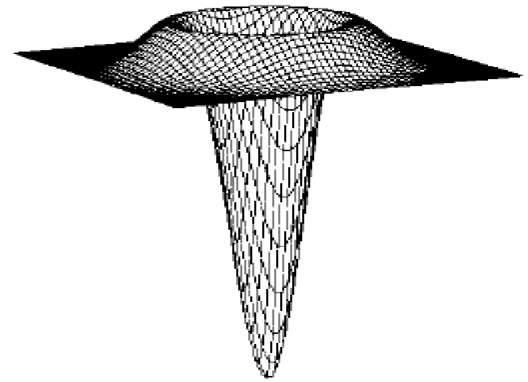
Marr-Hildreth Algorithm

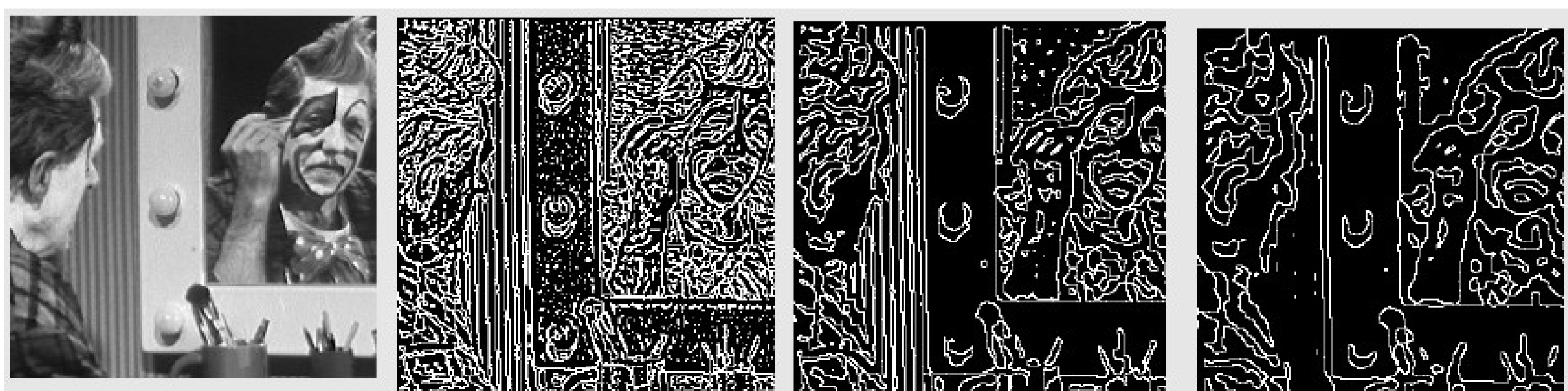
- Marr e Hildreth (1980) opted for a Laplacian of a Gaussian (LoG) filter
  - Edges correspond to maxima of the fisrst derivative and **zeroes of the second**
  - Isotropic → no need for several directional filters
  - Detect zero crossing of filtered image J

$$J = \nabla^2 G_\sigma * I$$

- Scale parameter σ related to detail

Laplacian of Gaussian





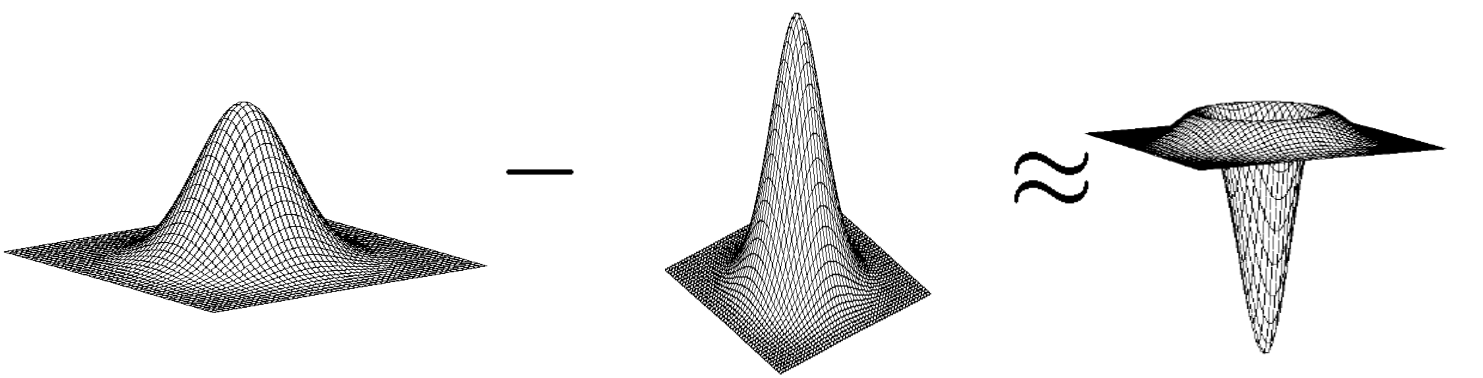
Difference of Gaussian

UNIVERSITÀ
CA' FOSCARI
VENEZIA

Original

Smoothed

Smoothed - Original



Gaussian — Gaussian ≈ Laplacian of Gaussian

Canny Edge Detector

- Canny tried to find the optimal edge detector, assuming a perfect step edge in gaussian noise
- Optimal means
  - Good Detection - it should mark all the edges and only all the edges
  - Good Localisation - the points marked should be as close to the real edge as possible
  - Minimal Response - each edge should be reported only once
- Canny used the Calculus of Variations
  - finds the function which best satisfies some functional

Canny Edge Detector

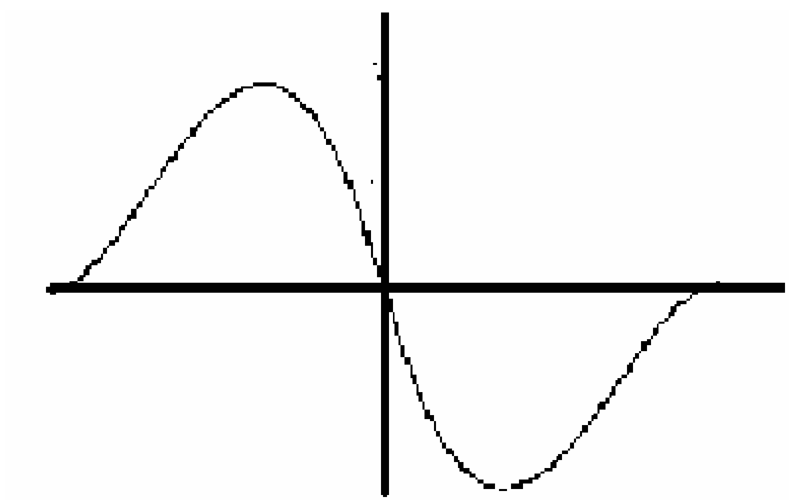
- Given a filter f, define the two objective functions:

  $\Lambda(f)$ large if f produces good localization

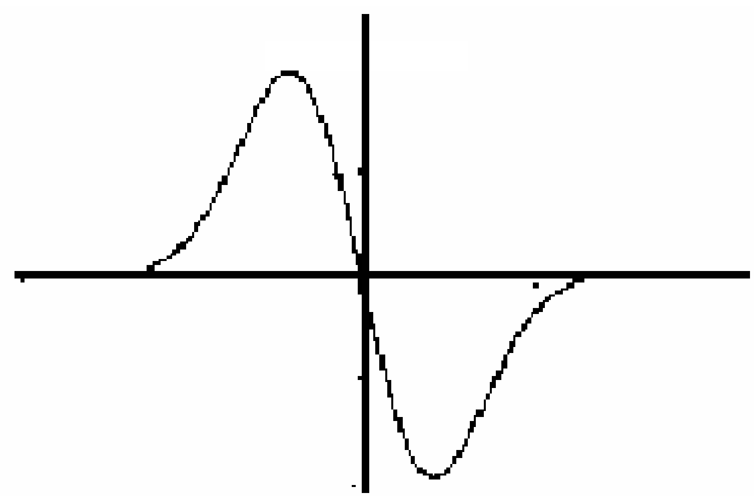
  $\Sigma(f)$ large if f produces good detection (high SNR)
- Problem: Find a family of filters f that maximizes the compromise criterion

$$\Lambda(f)\Sigma(f)$$

under the constraint that a single peak is generated by a step edge
- Solution: Unique solution, a close approximation is the Gaussian derivative filter!



Canny



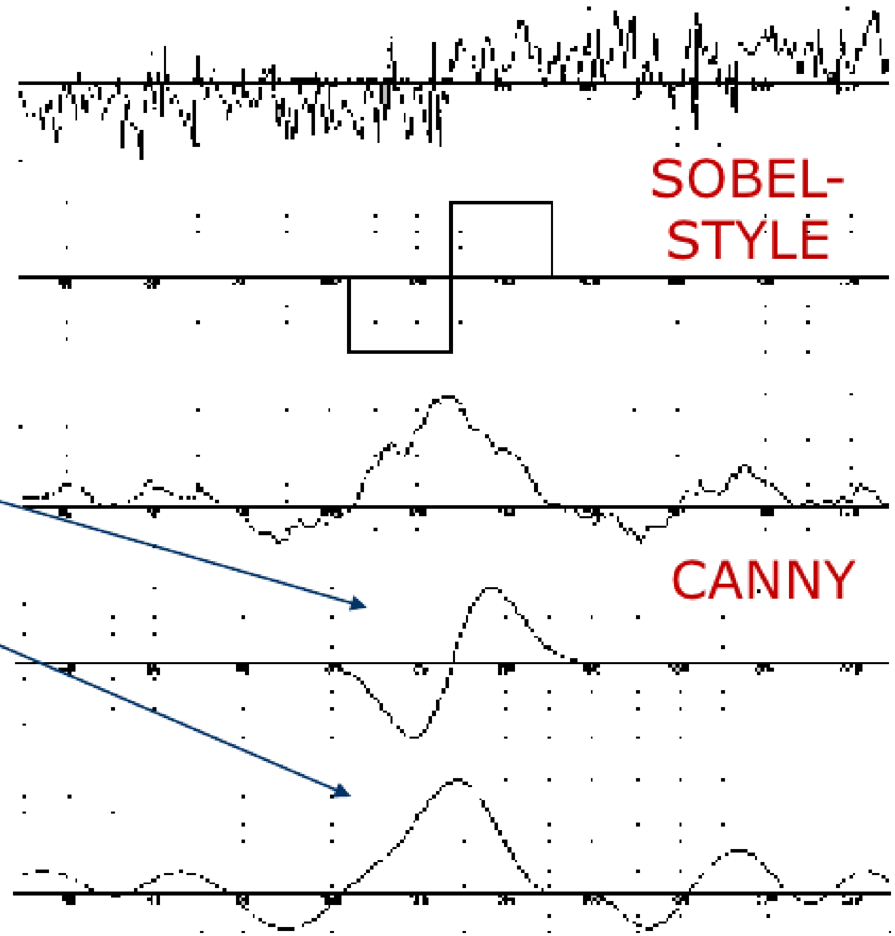
Derivative of Gaussian

Canny Edge Detector

- ❑ The optimal detector was described by sum of 4 exponential terms, but is very closely approximated by the 1st derivative of a Gaussian

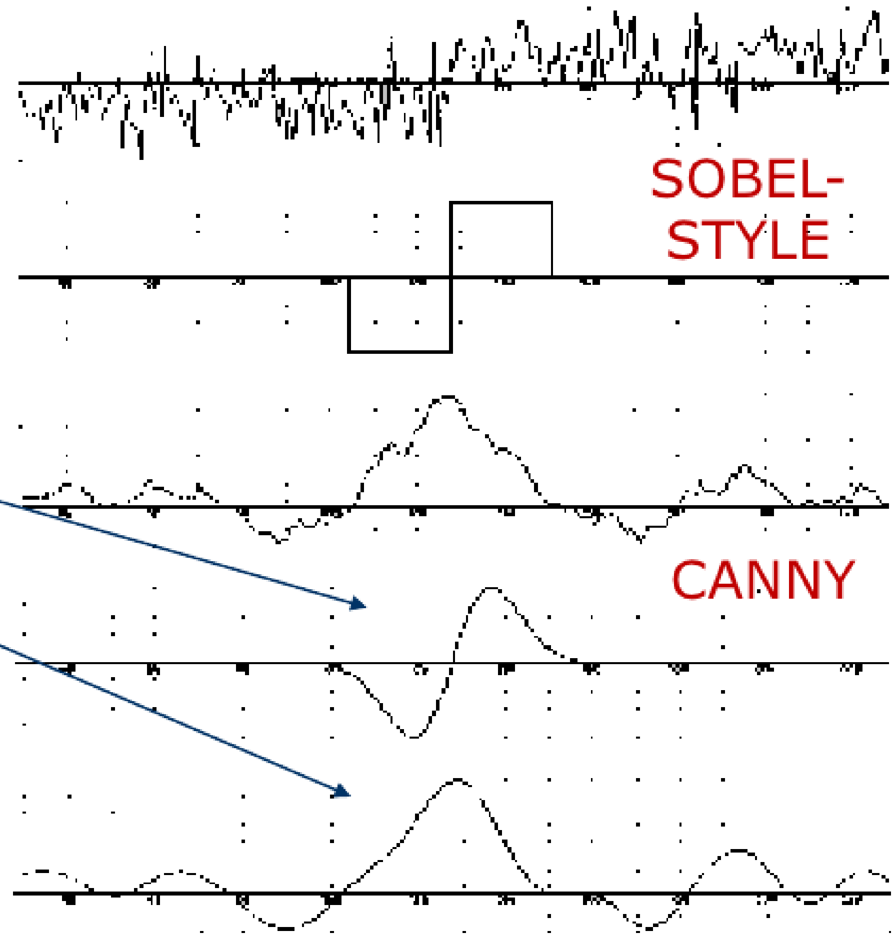
- ❑ Gives a cleaner response to a noisy edge than more square operators

- ❑ Canny also introduced thresholding with hysteresis



Canny Edge Detector

- ❑ The optimal detector was described by sum of 4 exponential terms, but is very closely approximated by the 1st derivative of a Gaussian
- ❑ Gives a cleaner response to a noisy edge than more square operators
- ❑Canny also introduced thresholding with hysteresis



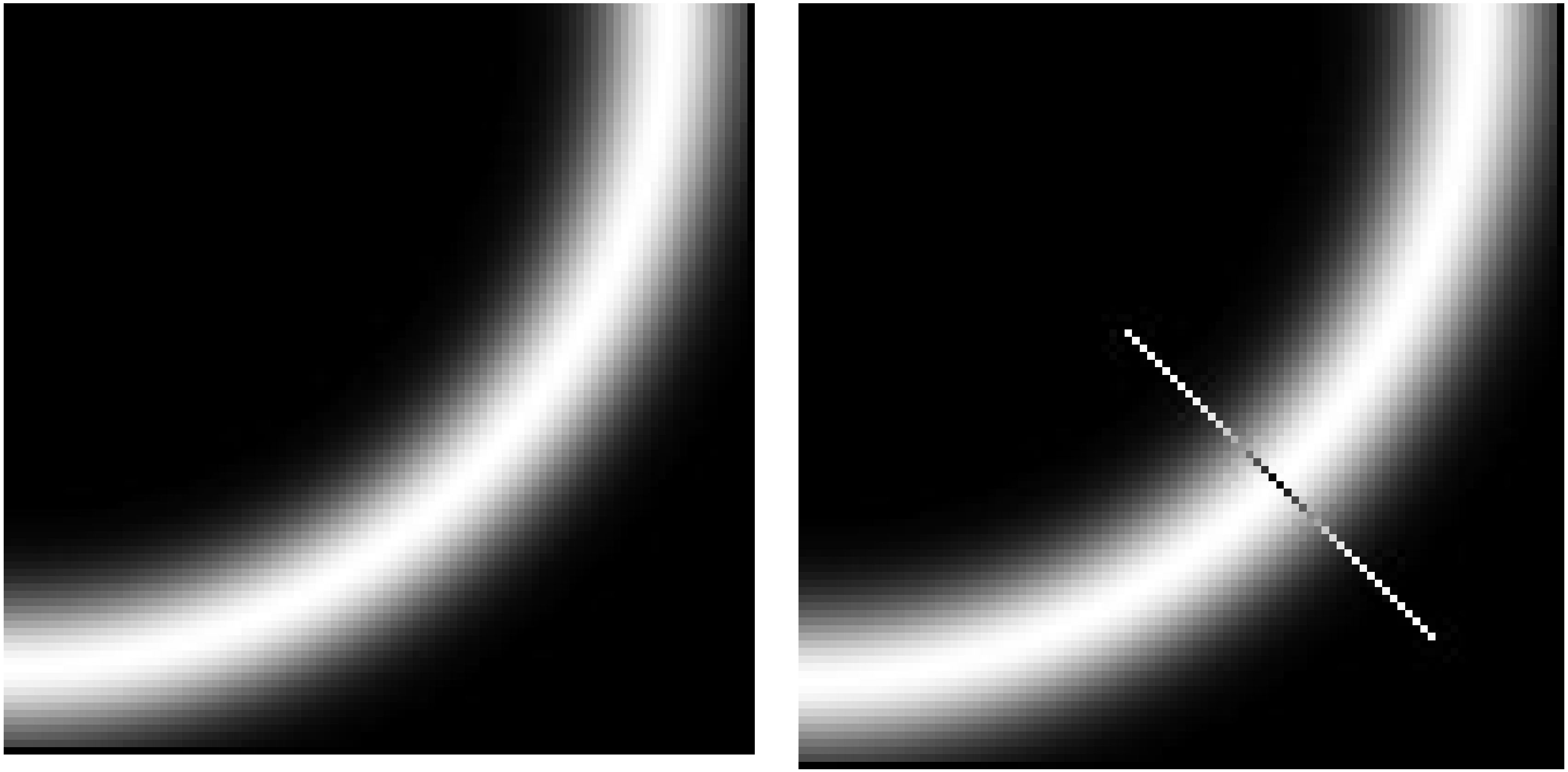
Canny Edge Detection

UNIVERSITÀ CA' FOSCARI VENEZIA

Steps:

1. Apply derivative of Gaussian
2. Non-maximum suppression
   - Thin multi-pixel wide “ridges” down to single pixel width
3. Linking and thresholding
   - Low, high edge-strength thresholds
   - Accept all edges over low threshold that are connected to edge over high threshold

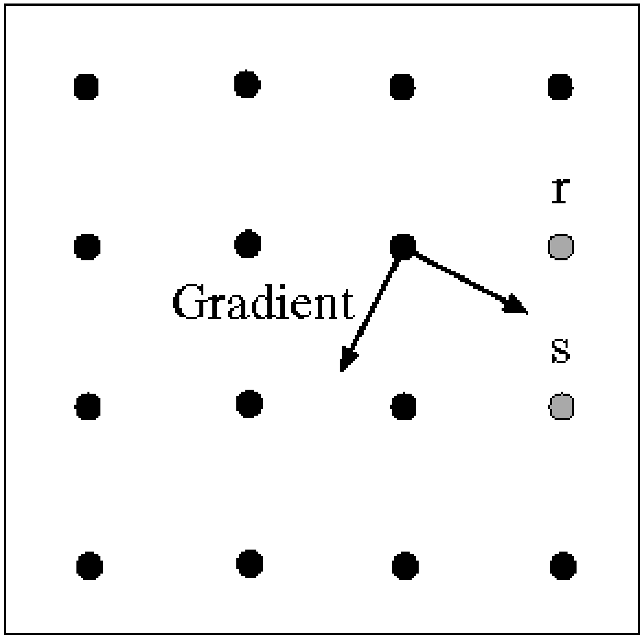
Non-Maximum Supression



Non-maximum suppression:

Select the single maximum point across the width of an edge.

Non-Maximum Supression

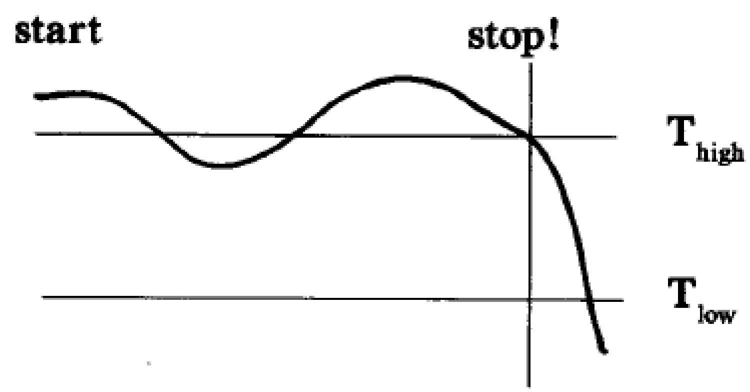


Assume the marked point is an edge point.

Take the normal to the gradient at that point and use this to predict continuation points (either r or s).

Hysteresis

- The industry standard thresholding method
  - apply an upper threshold T_{high}
  - follow edges until they fall below a lower threshold T_{low}
- Allows a band of variation, but assumes continuous edges
- User still selects parameters, but its easier, less precise



- Canny's main contribution is an understanding of why the method works - a Computational Theory of Edge Detection

Why is Canny so Dominant

- Still widely used after 20 years.
  - Theory is nice (but end result same).
  - Details good (magnitude of gradient).
  - Hysteresis an important heuristic.
  - Code was distributed.
  - Perhaps this is about all you can do with linear filtering.

Example

Original image

Strong + connected weak edges

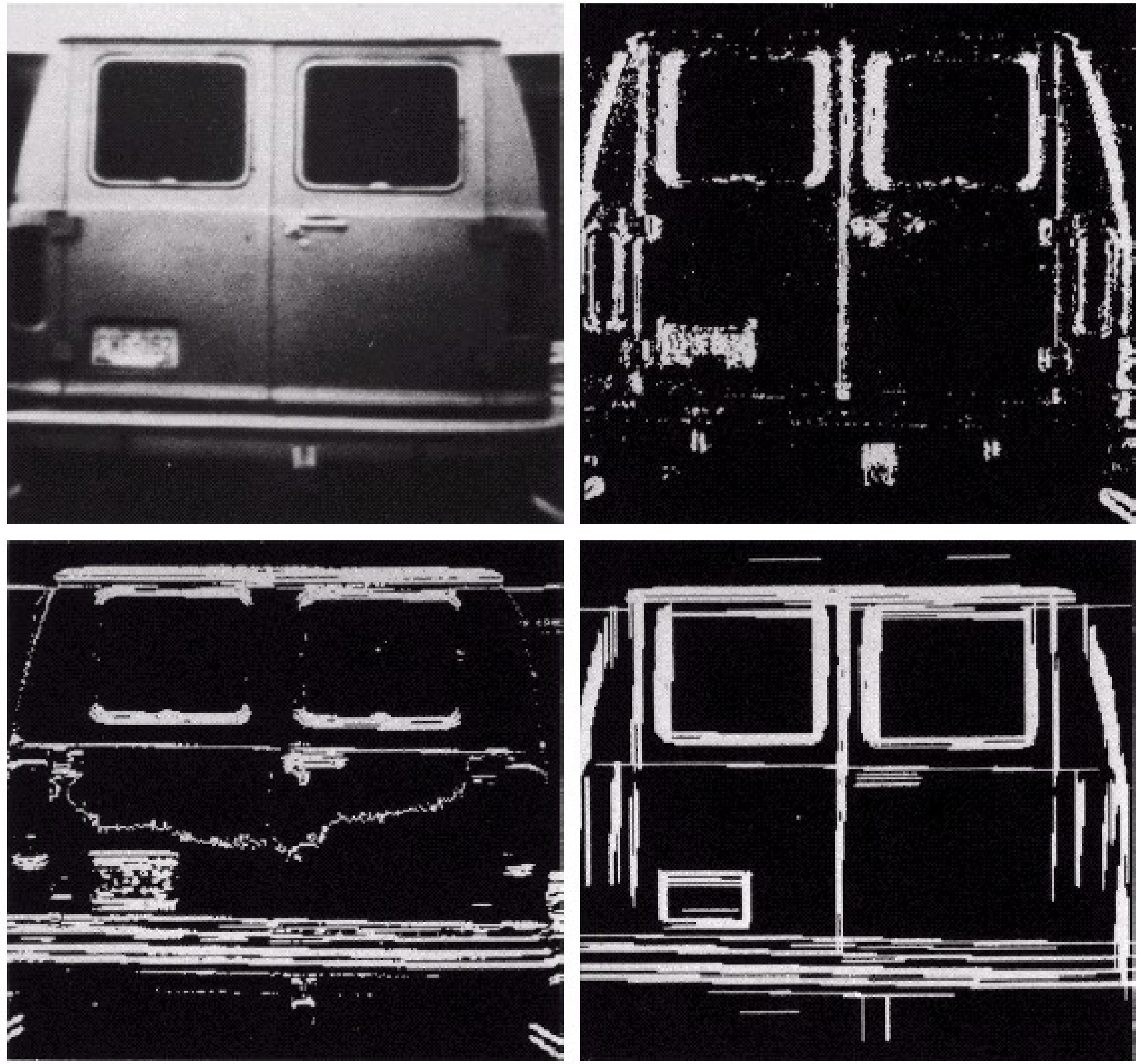
Strong edges only

Weak edges

Edge Linking

a b
c d

FIGURE 10.16
(a) Input image. (b) G_y component of the gradient. (c) G_x component of the gradient. (d) Result of edge linking. (Courtesy of Perceptics Corporation.)



Finding lines in an image

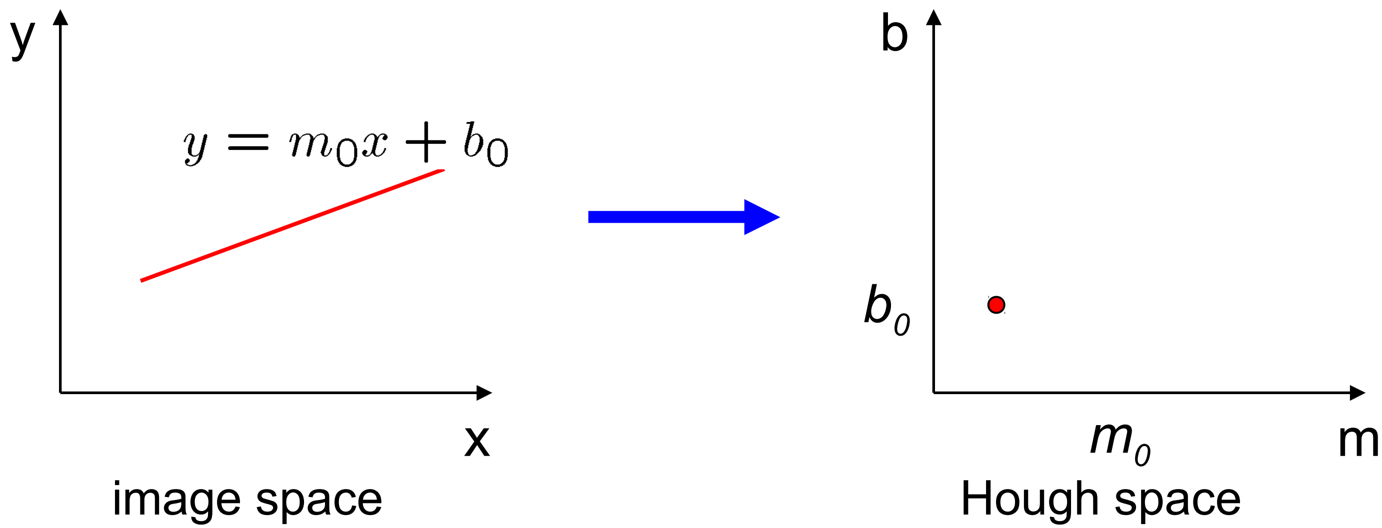
Option 1:

- Search for the line at every possible position/orientation
- What is the cost of this operation?

Option 2:

- Use a voting scheme: Hough transform

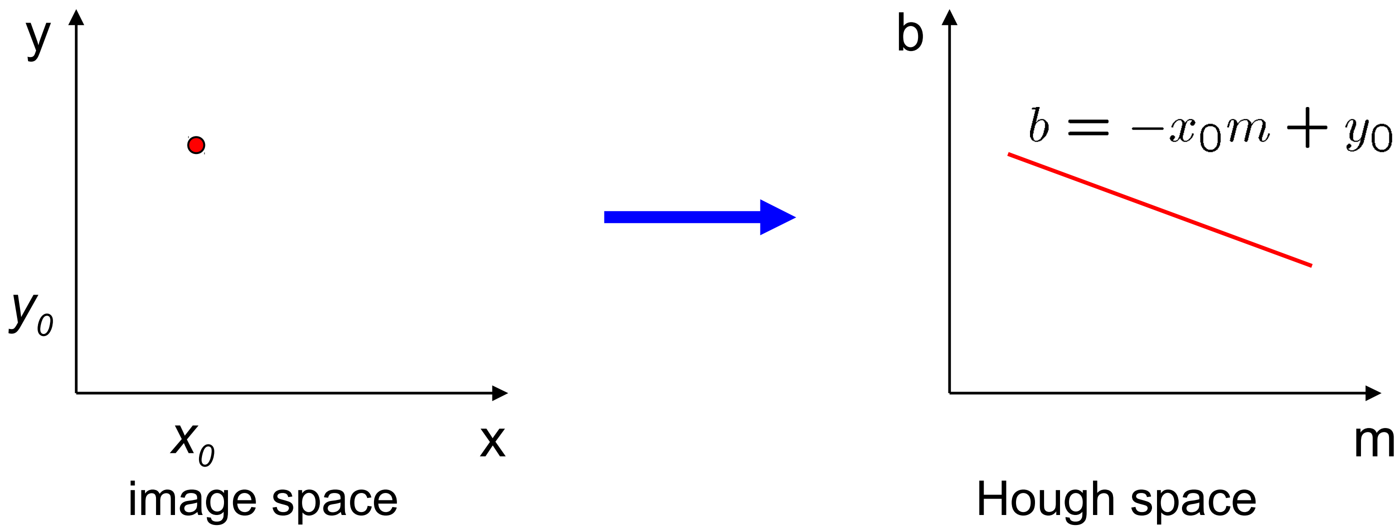
Hough Transform



Connection between image (x,y) and Hough (m,b) spaces

- A line in the image corresponds to a point in Hough space
- To go from image space to Hough space:
  - given a set of points (x,y), find all (m,b) such that y = mx + b

Hough Transform



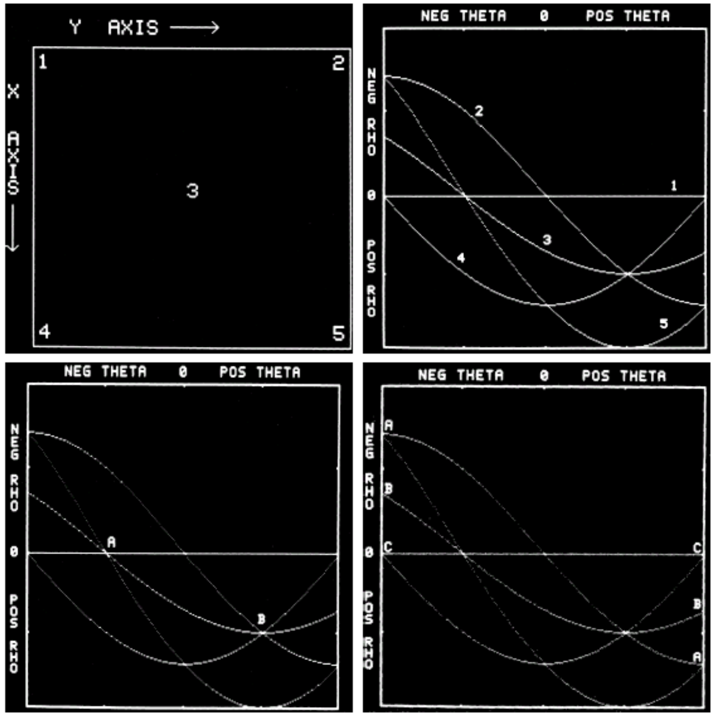
Connection between image (x,y) and Hough (m,b) spaces

- A line in the image corresponds to a point in Hough space
- To go from image space to Hough space:
  - given a set of points (x,y), find all (m,b) such that y = mx + b
- What does a point (x_0, y_0) in the image space map to?
  - A: the solutions of $b = -x_0 m + y_0$
  - this is a line in Hough space

Hough Transform

Typically use a different parameterization

- d is the perpendicular distance from the line to the origin θ is the angle this perpendicular makes with the x axis
- Why?



Hough Transform

Basic Hough transform algorithm

- Initialize H[d, θ]=0
- for each edge point I[x,y] in the image

  for θ = 0 to 180

  $$d = xcos\theta + ysin\theta$$

  H[d, θ] += 1
- Find the value(s) of (d, θ) where H[d, θ] is maximum
- The detected line in the image is given by

What's the running time (measured in # votes)?

Hough Transform

Extension 1: Use the image gradient

- same
- for each edge point I[x,y] in the image

  compute unique (d, θ) based on image gradient at (x,y)

  H[d, θ] += 1
- same
- same

What's the running time measured in votes?

Extension 2

- give more votes for stronger edges

Extension 3

- change the sampling of (d, θ) to give more/less resolution

Extension 4

- The same procedure can be used with circles, squares, or any other shape