Filters (cont.)

CS 554 – Computer Vision

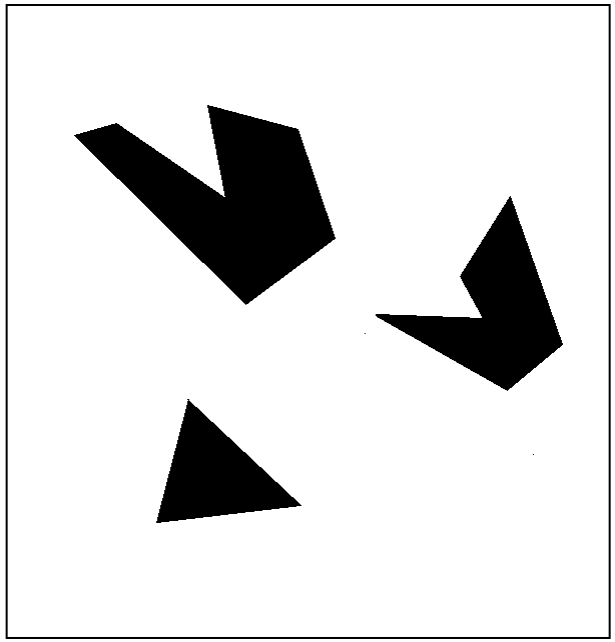
Pinar Duygulu

Bilkent University

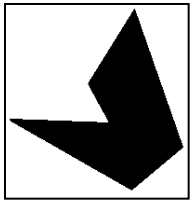
Today's topics

- Image Formation
- Image filters in spatial domain
  - Filter is a mathematical operation of a grid of numbers
  - Smoothing, sharpening, measuring texture
- Image filters in the frequency domain
  - Filtering is a way to modify the frequencies of images
  - Denoising, sampling, image compression
- Templates and Image Pyramids
  - Filtering is a way to match a template to the image
  - Detection, coarse-to-fine registration

Template matching



Scene

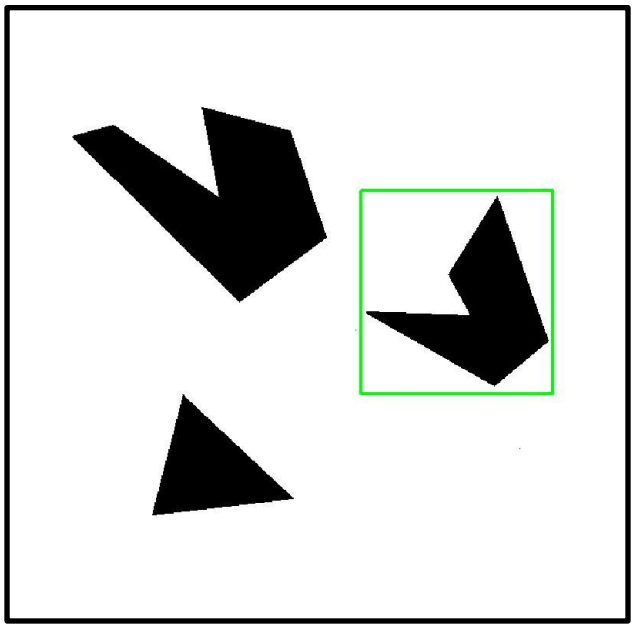


Template (mask)

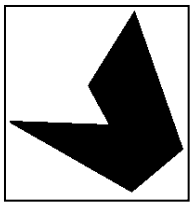
A toy example

Source: Darrell, Berkeley

Template matching



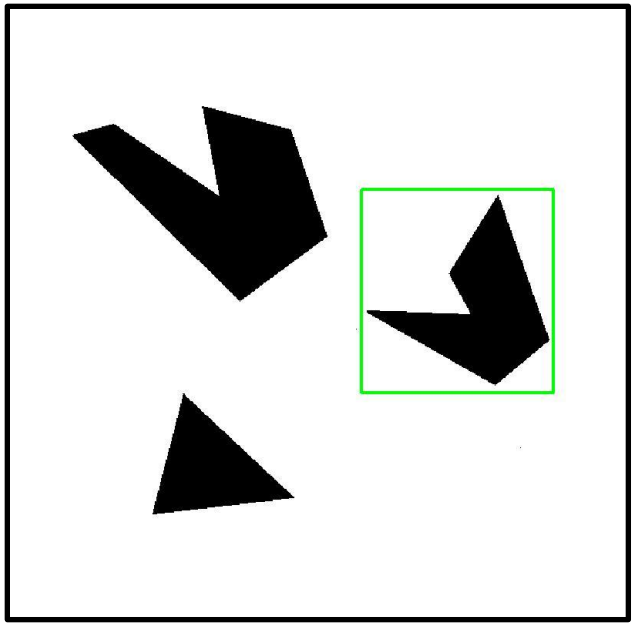
Detected template



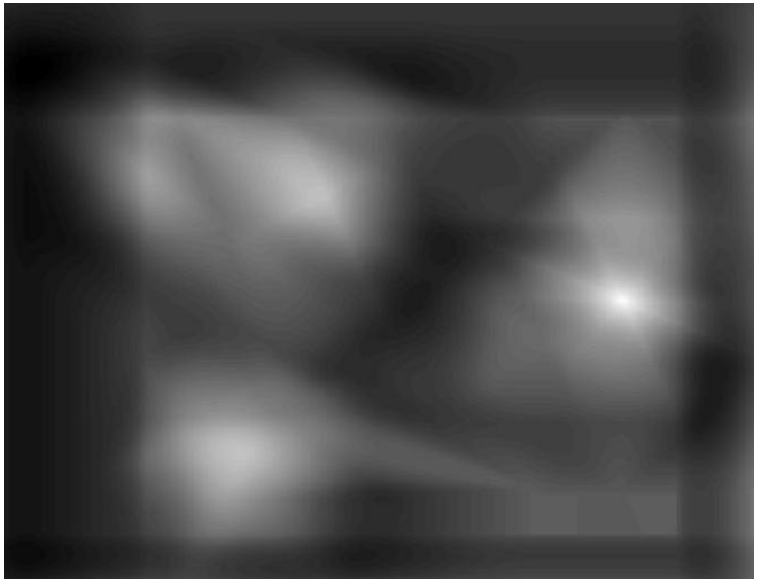
Template

Source: Darrell, Berkeley

Template matching



Detected template

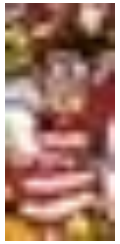


Correlation map

Source: Darrell, Berkeley

Where's Waldo?

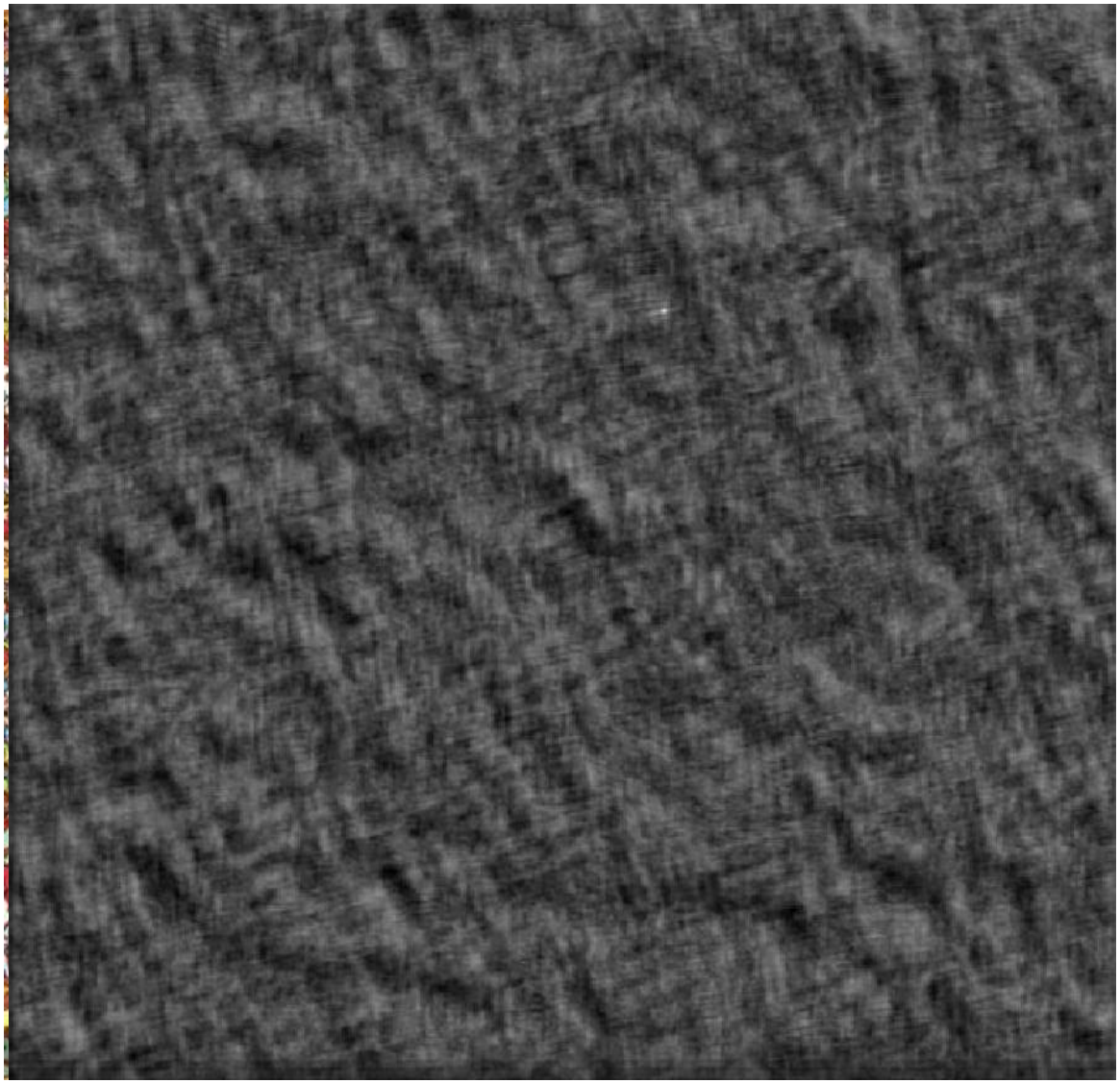
Scene



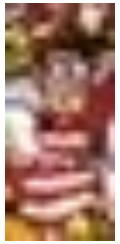
Template

Source: Darrell, Berkeley

Where's Waldo?



Scene

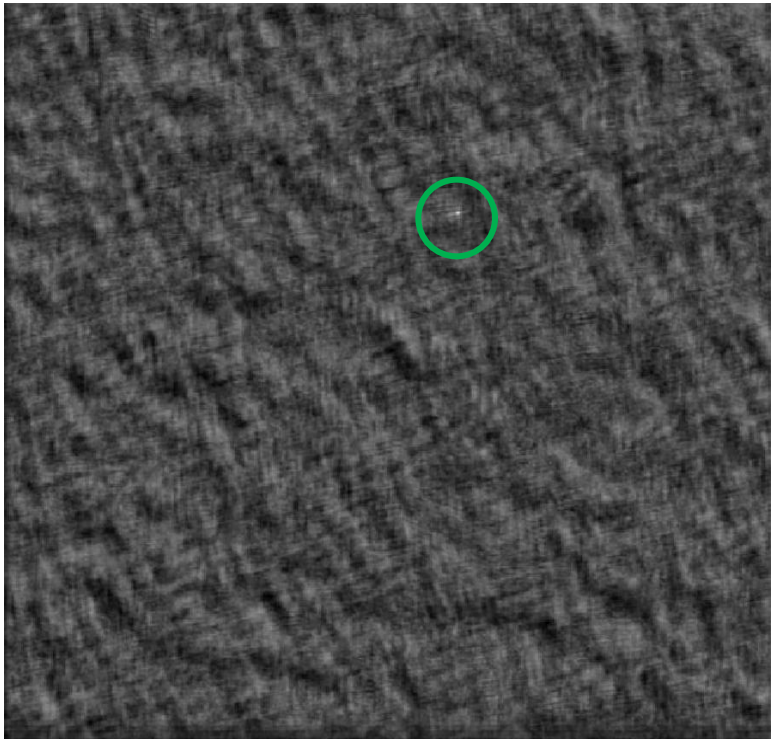


Template

Source: Darrell, Berkeley

Where's Waldo?

Detected template



Correlation map

Source: Darrell, Berkeley

Template matching

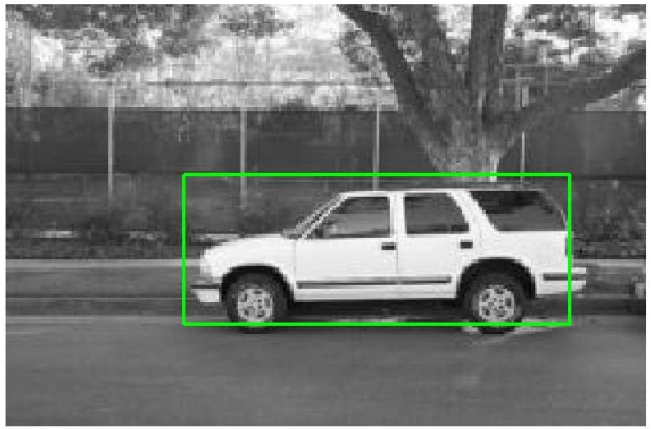
Scene

Template

What if the template is not identical to some subimage in the scene?

Source: Darrell, Berkeley

Template matching



Detected template

Template

Match can be meaningful, if scale, orientation, and general appearance is right.

Source: Darrell, Berkeley

Application

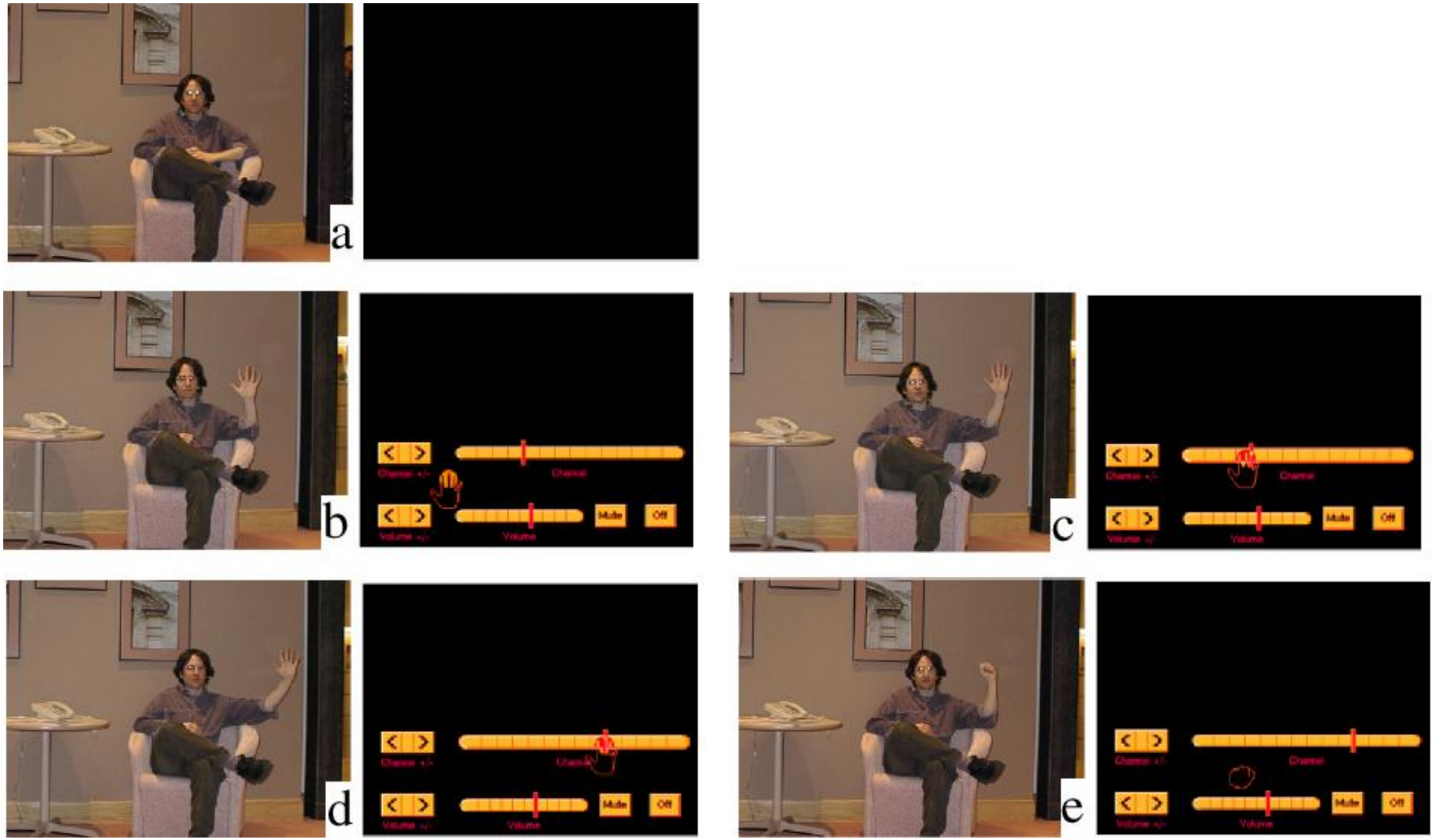
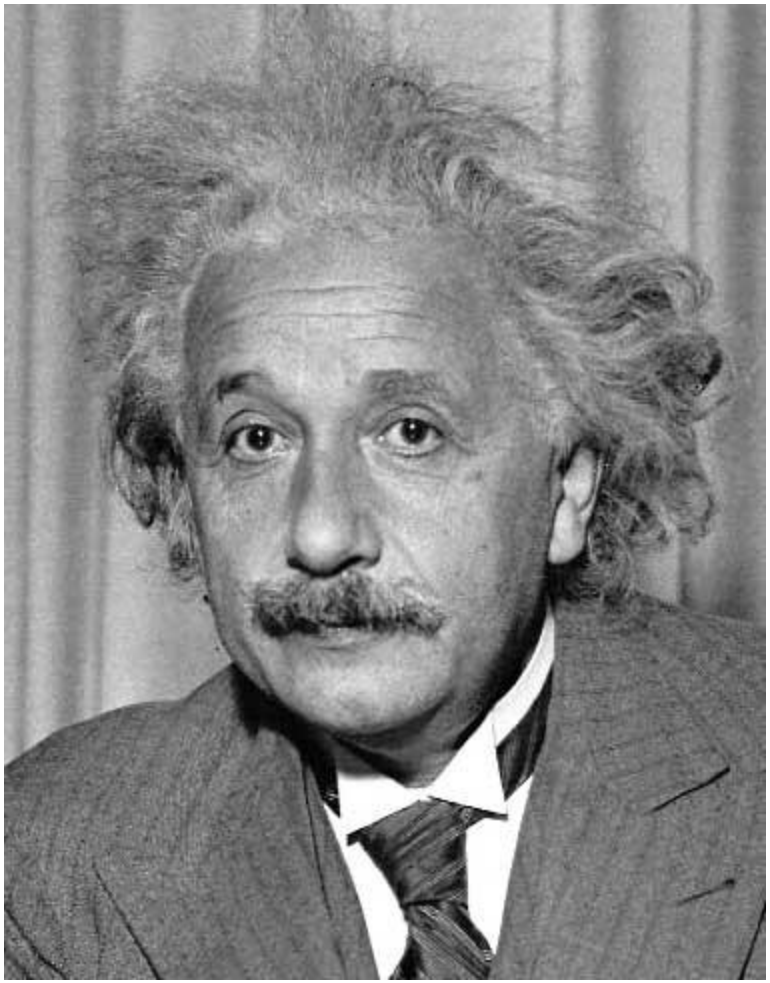


Figure from “Computer Vision for Interactive Computer Graphics,” W.Freeman et al, IEEE Computer Graphics and Applications, 1998

Template matching

- Goal: find  in image

- Main challenge: What is a good similarity or distance measure between two patches?
  - Correlation
  - Zero-mean correlation
  - Sum Square Difference
  - Normalized Cross Correlation



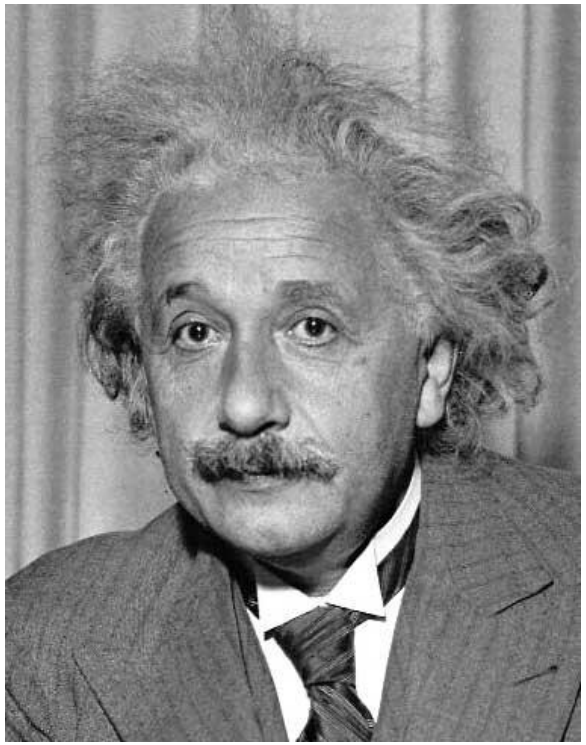
Source: Hays, Brown

Matching with filters

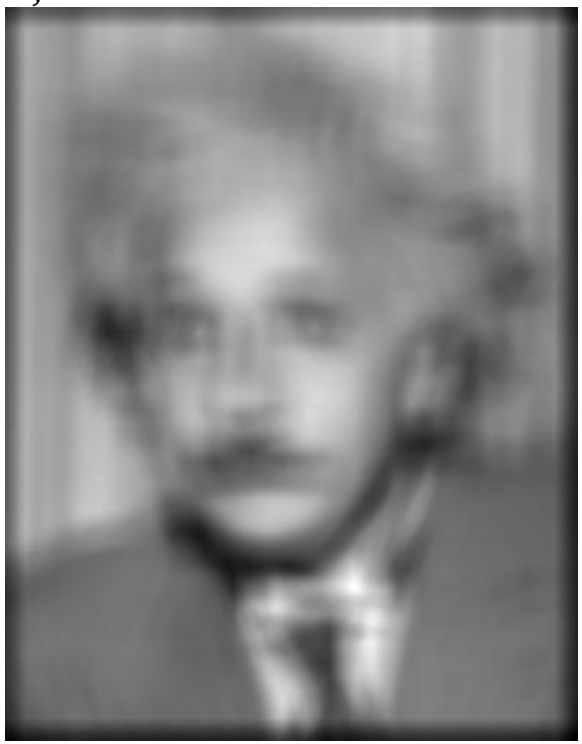
- Goal: find  in image
- Method 0: filter the image with eye patch

$$h[m,n] = \sum_{k,l} g[k,l]\ f[m+k,n+l]$$

f = image
g = filter



Input



Filtered Image

What went wrong?

response is stronger for higher intensity

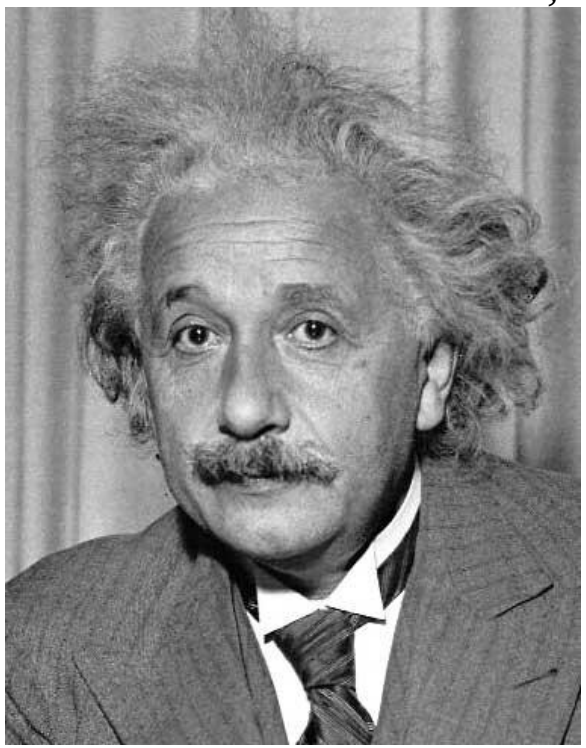
Source: Hays, Brown

Matching with filters

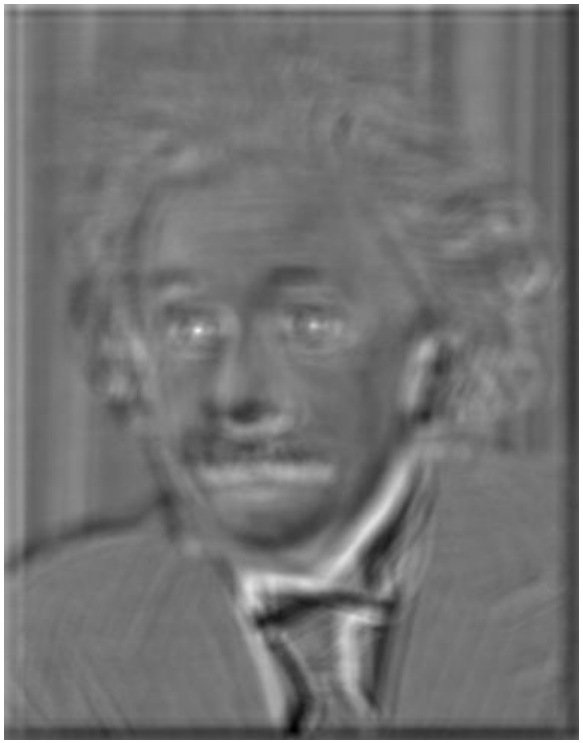
- Goal: find  in image
- Method 1: filter the image with zero-mean eye

$$h[m,n] = \sum_{k,l} (f[k,l] - \bar{f})(g[m+k, n+l])$$

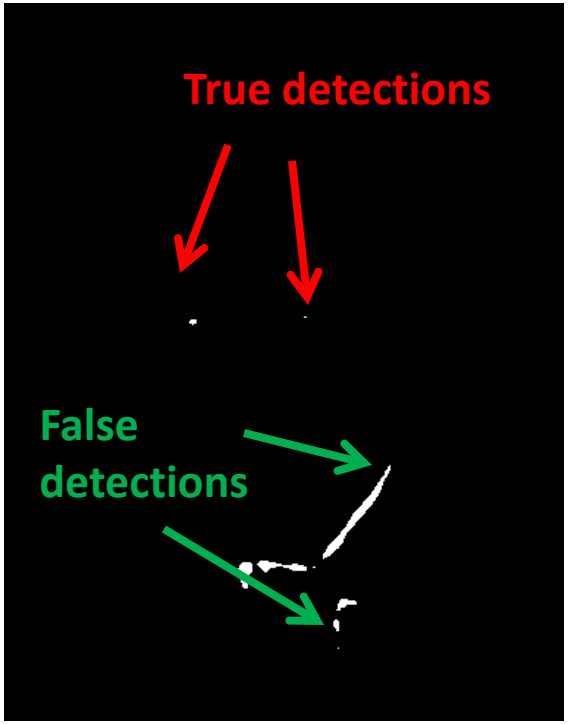
mean of f



Input



Filtered Image (scaled)



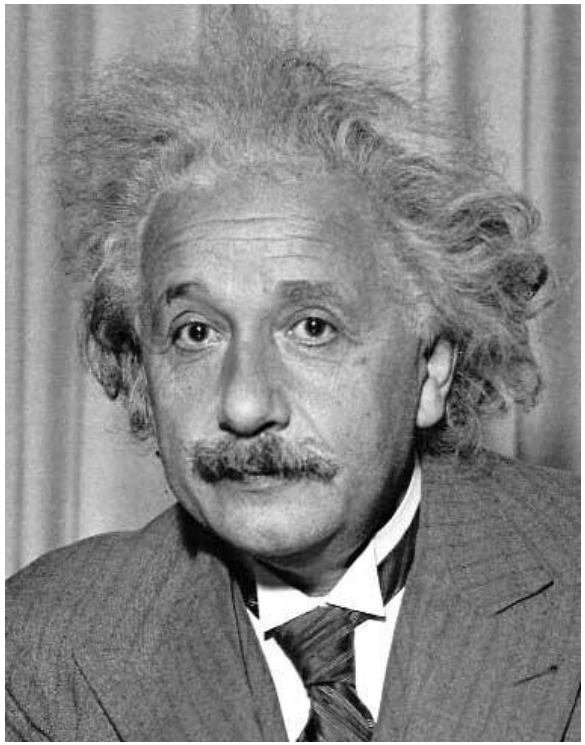
Thresholded Image

Source: Hays, Brown

Matching with filters

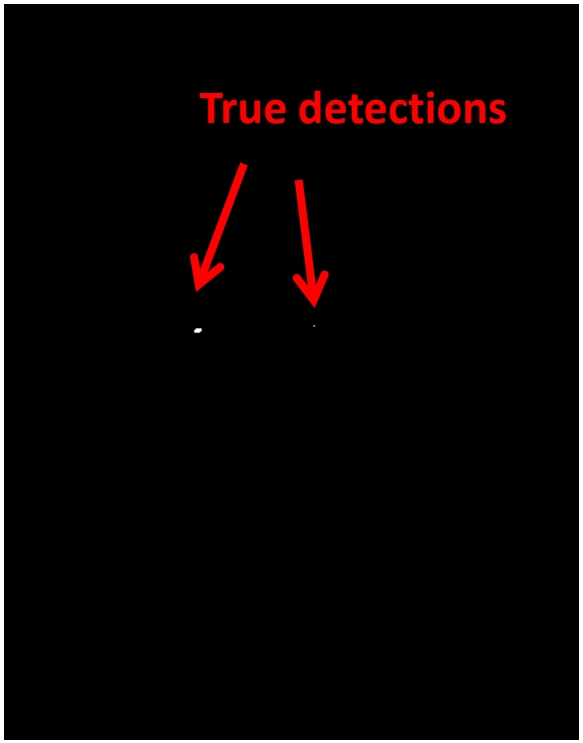
- Goal: find  in image
- Method 2: SSD

$$h[m,n] = \sum_{k,l} (g[k,l] - f[m+k,n+l])^2$$



Input

1- sqrt(SSD)



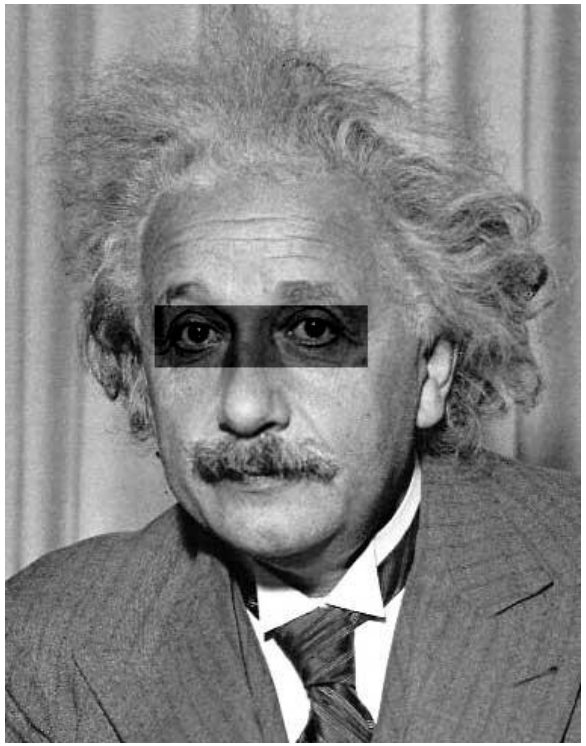
Thresholded Image

Source: Hays, Brown

Matching with filters

- Goal: find  in image
- Method 2: SSD

$$h[m,n] = \sum_{k,l} (g[k,l] - f[m+k,n+l])^2$$



Input

1- sqrt(SSD)

What's the potential downside of SSD?

SSD is sensitive to average intensity

Source: Hays, Brown

Matching with filters

- Goal: find  in image
- Method 3: Normalized cross-correlation

mean template

mean image patch

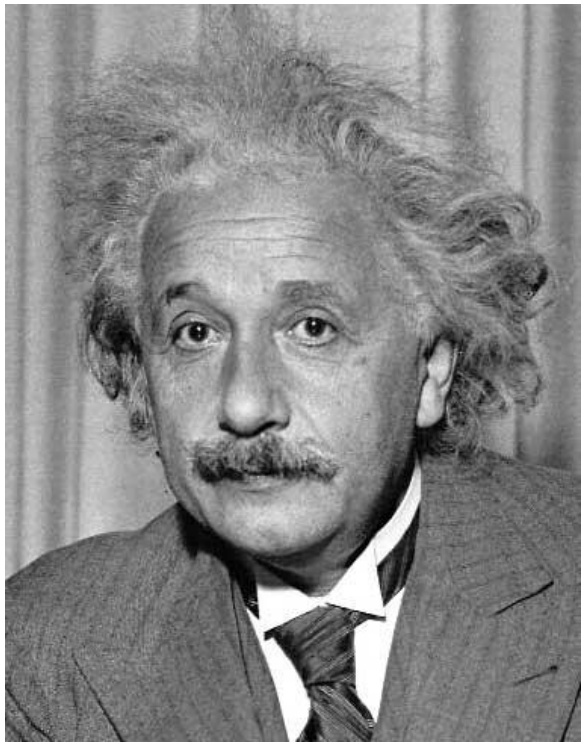
$$h[m,n] = \frac{\sum_{k,l}(g[k,l]-\bar{g})(f[m-k,n-l]-\bar{f}_{m,n})}{\left(\sum_{k,l}(g[k,l]-\bar{g})^2\sum_{k,l}(f[m-k,n-l]-\bar{f}_{m,n})^2\right)^{0.5}}$$

Matlab: `normxcorr2(template, im)`

Source: Hays, Brown

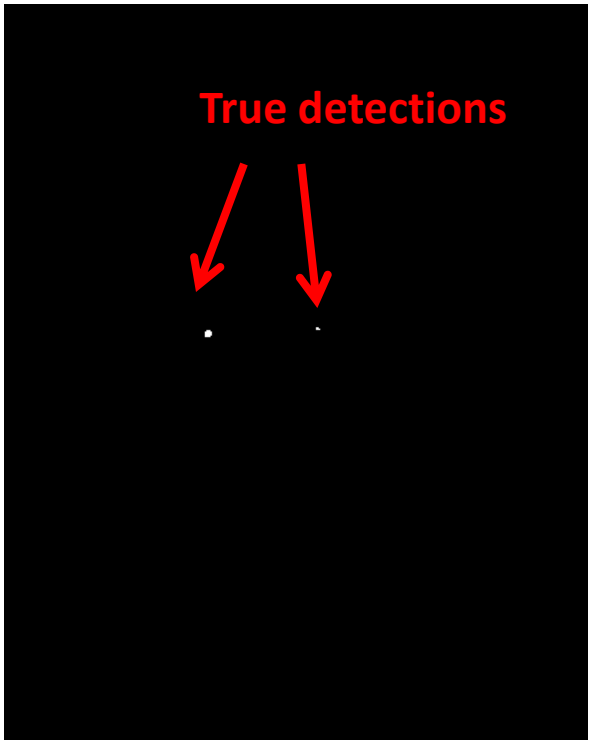
Matching with filters

- Goal: find  in image
- Method 3: Normalized cross-correlation



Input

Normalized X-Correlation

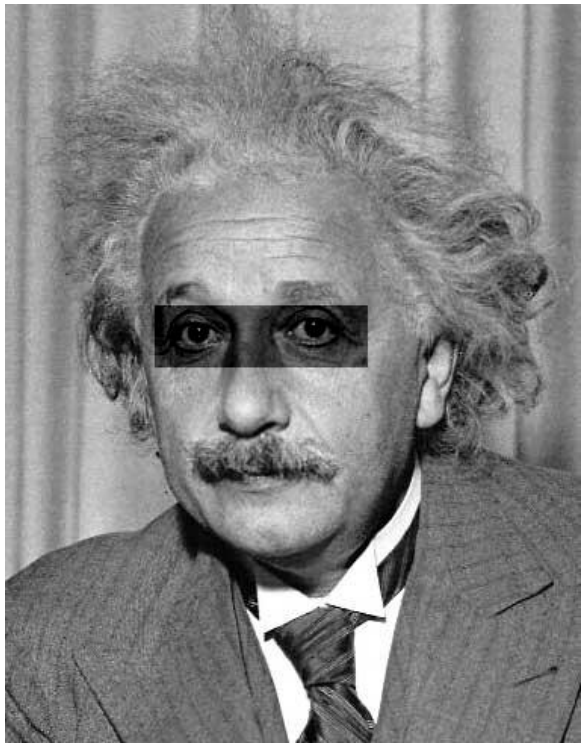


Thresholded Image

Source: Hays, Brown

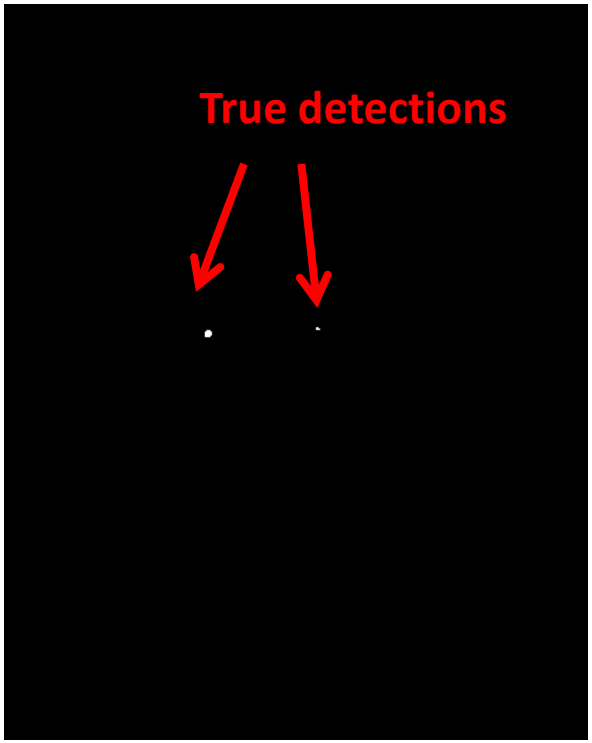
Matching with filters

- Goal: find  in image
- Method 3: Normalized cross-correlation



Input

Normalized X-Correlation



Thresholded Image

Source: Hays, Brown

Q: What is the best method to use?

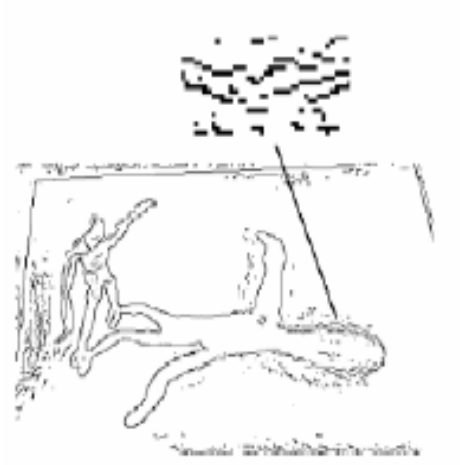
A: Depends

- SSD: faster, sensitive to overall intensity
- Normalized cross-correlation: slower, invariant to local average intensity and contrast

Source: Hays, Brown

Q: What if we want to find larger or smaller eyes?

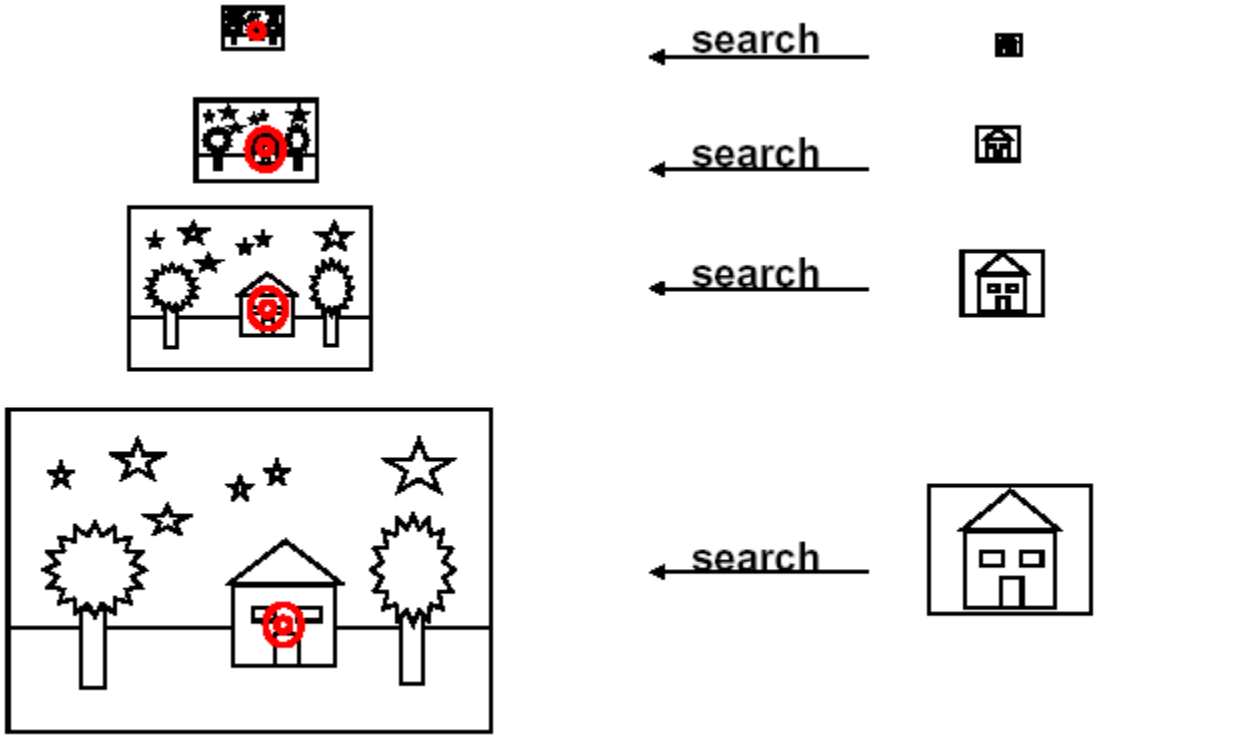
Motivation for studying scale.



ELDER AND ZUCKER: LOCAL SCALE CONTROL FOR EDGE DETECTION AND BLUR ESTIMATION

IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 20, NO. 7, JULY 1998

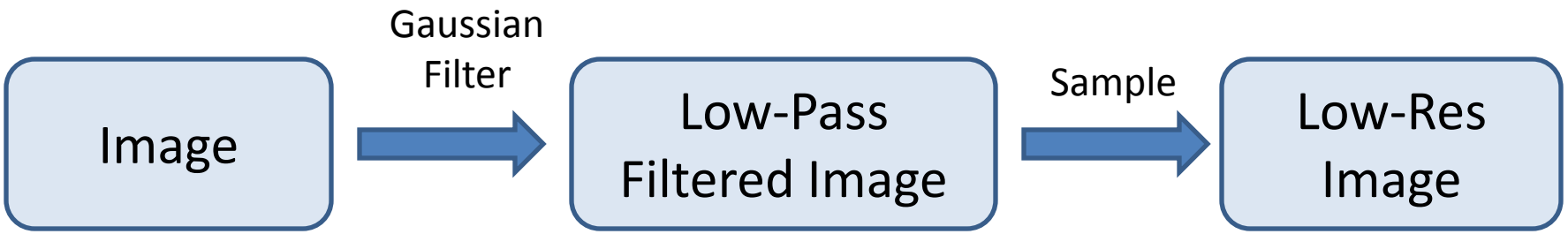
A: Image Pyramid



Irani & Basri

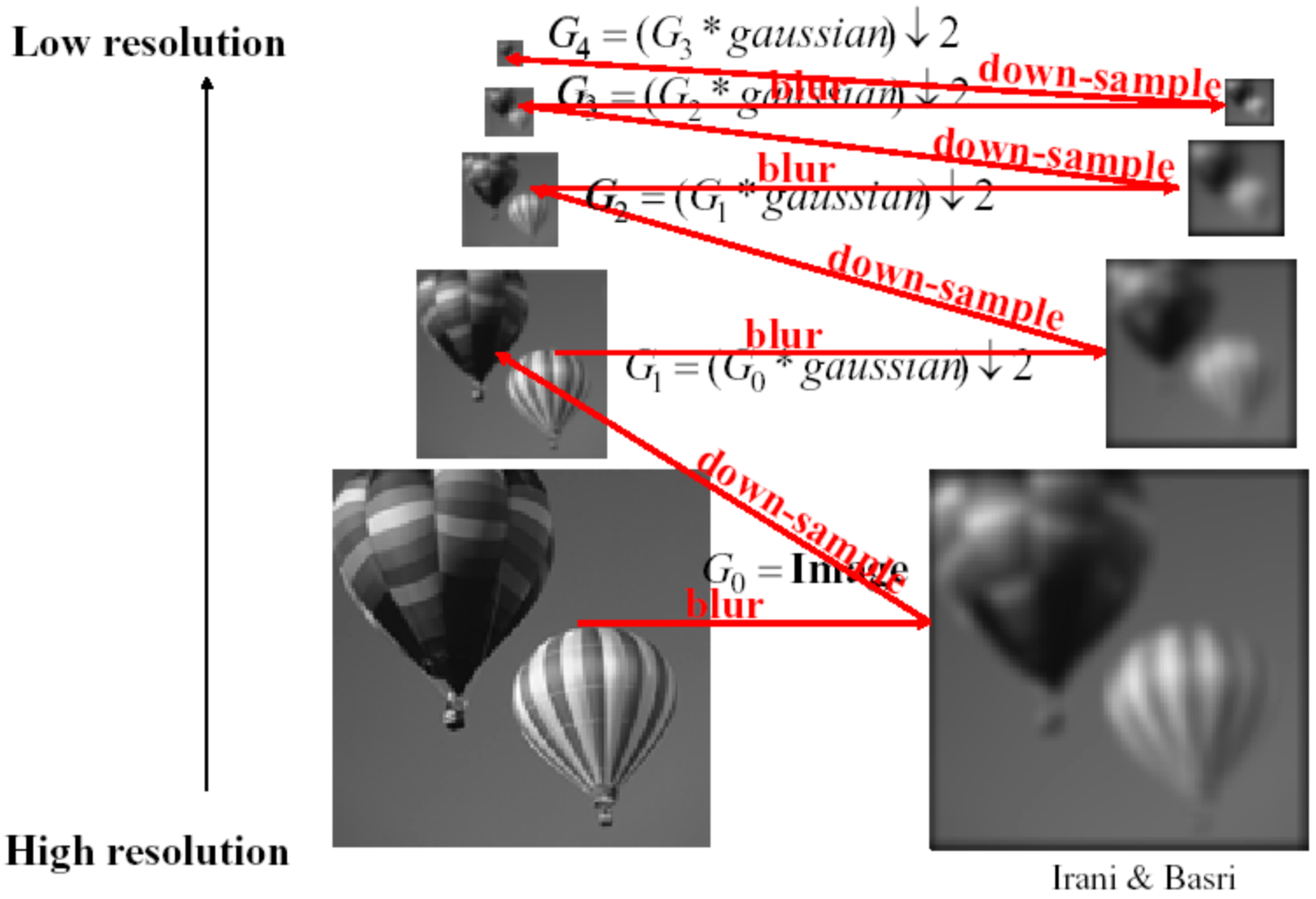
adapted from Michael Black, Brown University

Review of Sampling



Source: Hays, Brown

Gaussian Pyramid



adapted from Michael Black, Brown University

Gaussian pyramid

512 256 128 64 32 16 8

Source: Hays, Brown

Source: Forsyth

Template Matching with Image Pyramids

Input: Image, Template

1. Match template at current scale

2. Downsample image

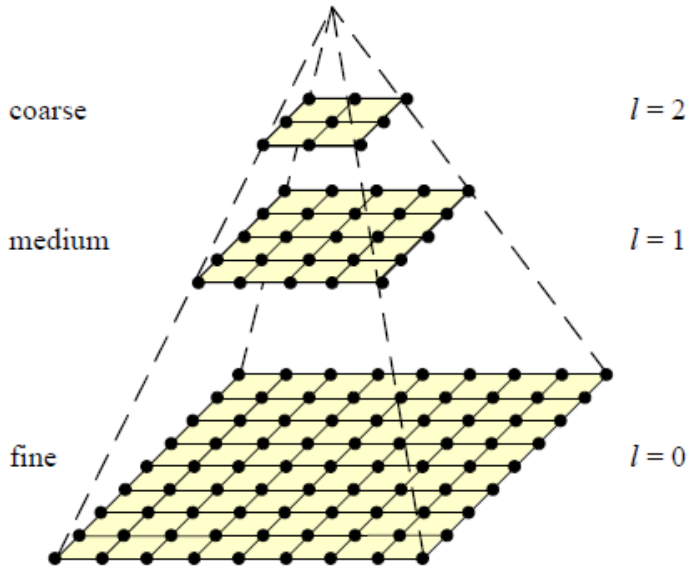
3. Repeat 1-2 until image is very small

4. Take responses above some threshold, perhaps with non-maxima suppression

Source: Hays, Brown

Coarse-to-fine Image Registration

1. Compute Gaussian pyramid
2. Align with coarse pyramid
3. Successively align with finer pyramids
   - Search smaller range



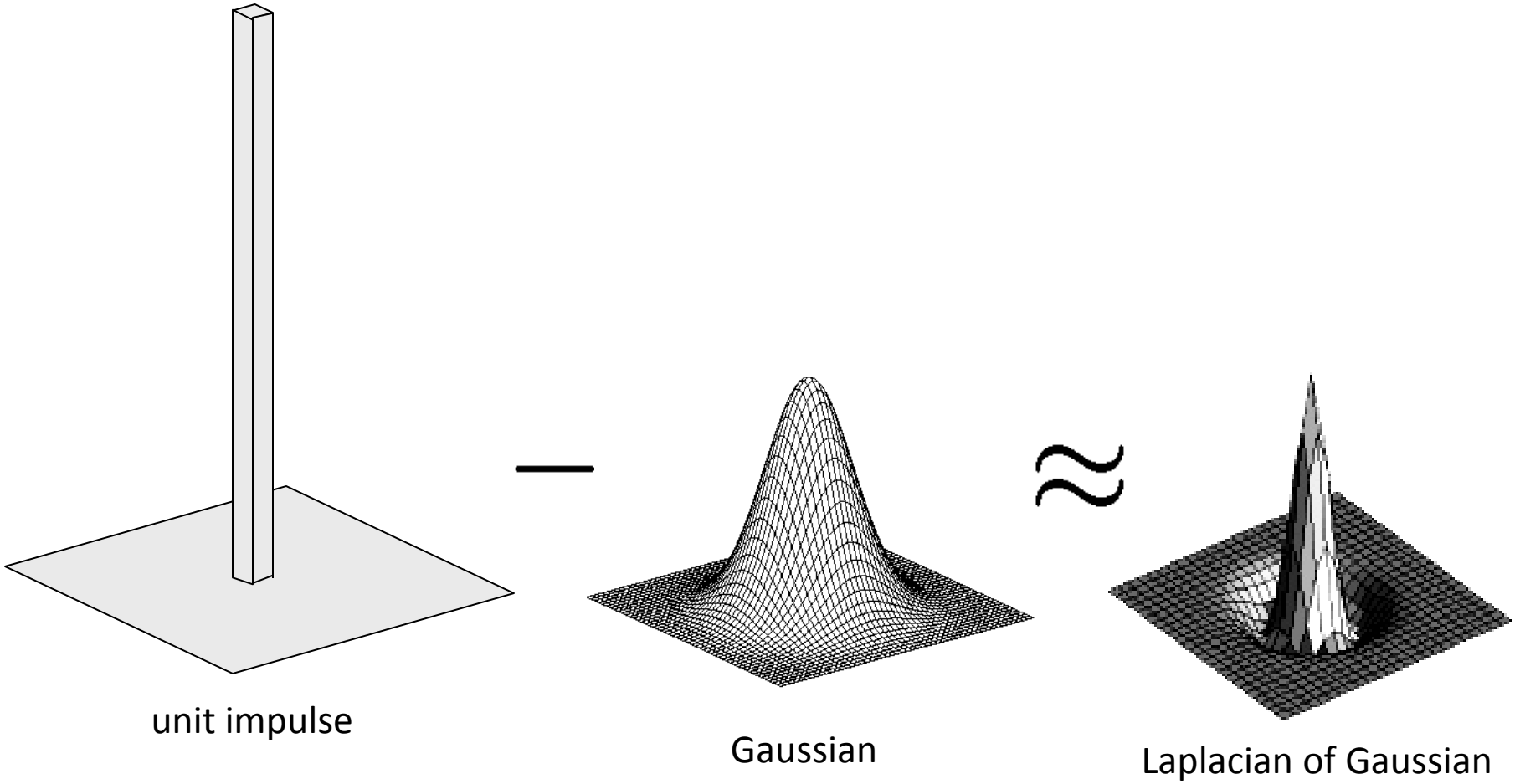
Why is this faster?

Are we guaranteed to get the same result?

Source: Hays, Brown

Slide: Hoiem

Laplacian filter



unit impulse

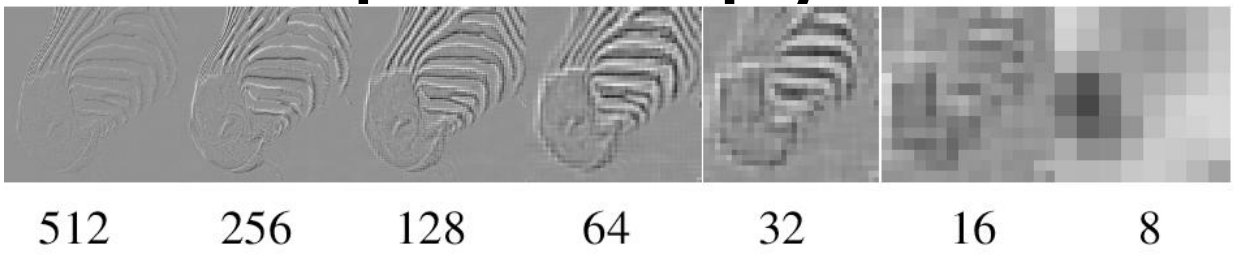
Gaussian

Laplacian of Gaussian

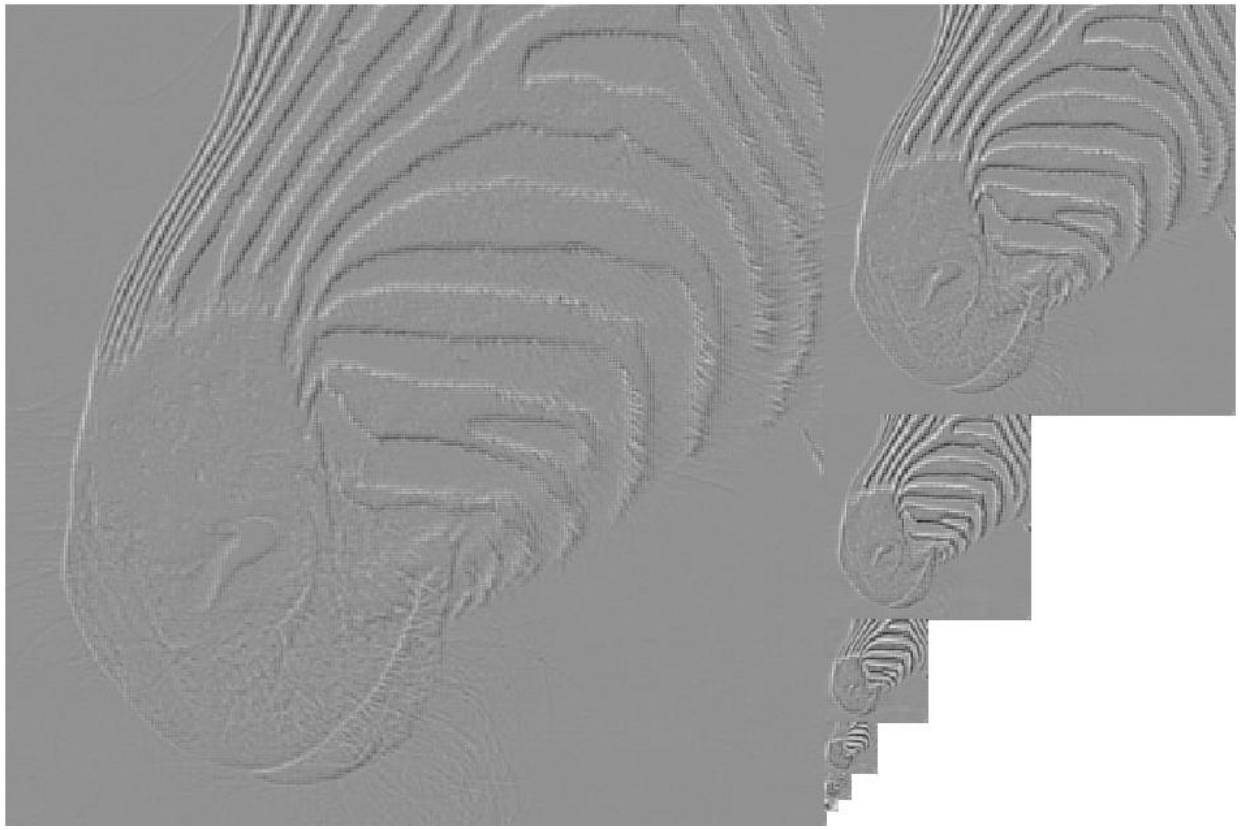
Source: Hays, Brown

Source: Lazebnik

Laplacian pyramid



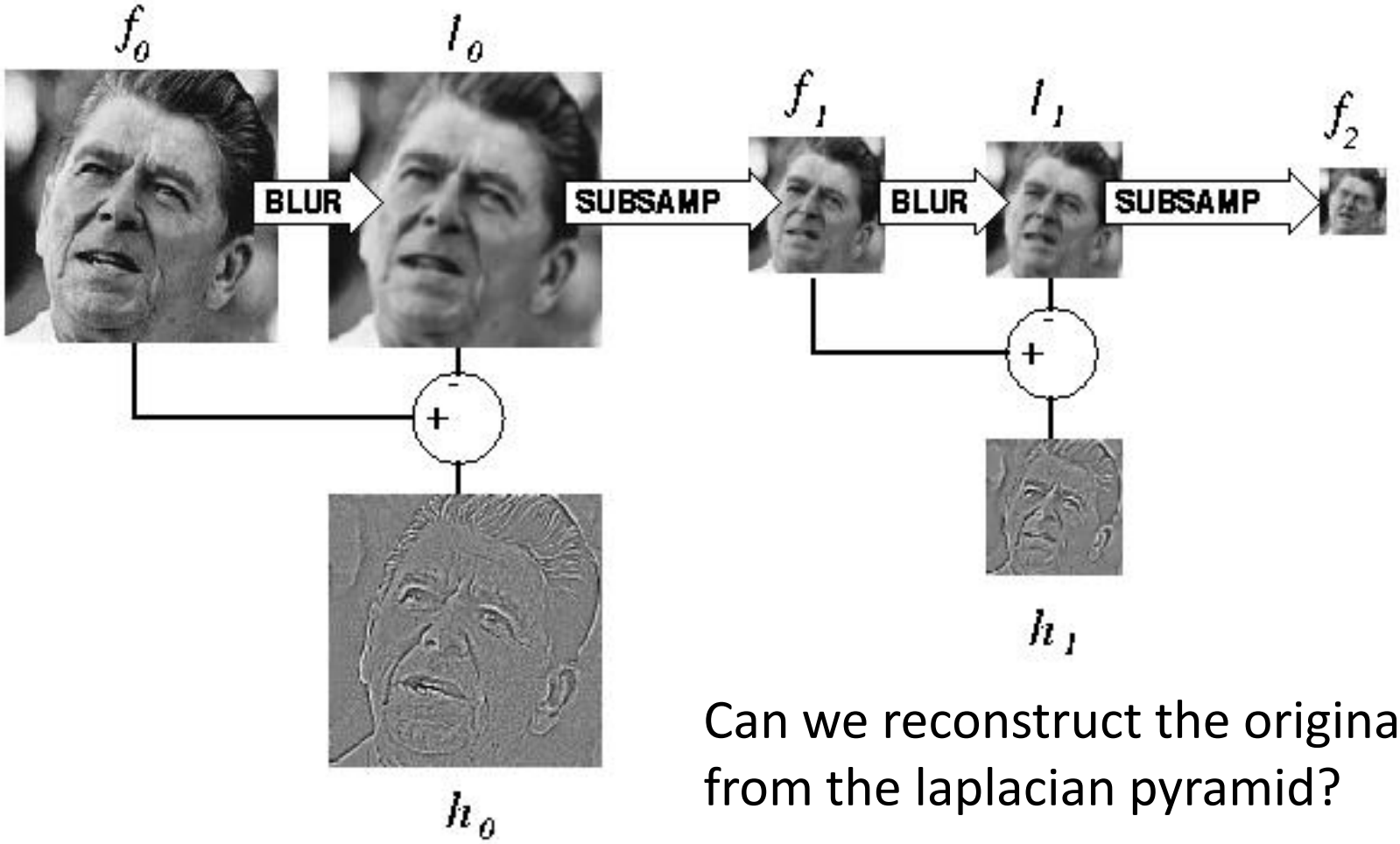
512 256 128 64 32 16 8



Source: Hays, Brown

Source: Forsyth

Computing Gaussian/Laplacian Pyramid

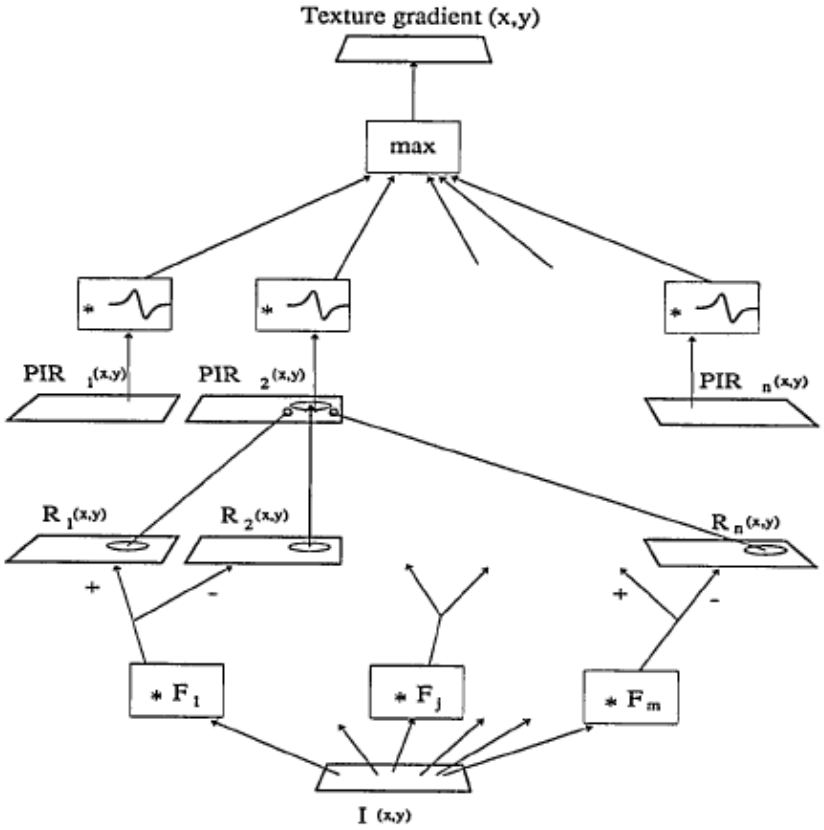


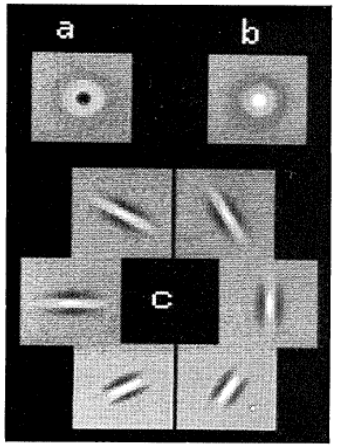
Can we reconstruct the original from the laplacian pyramid?

Source: Hays, Brown

http://sepwww.stanford.edu/~morgan/texturematch/paper_html/node3.html

Texture segmentation

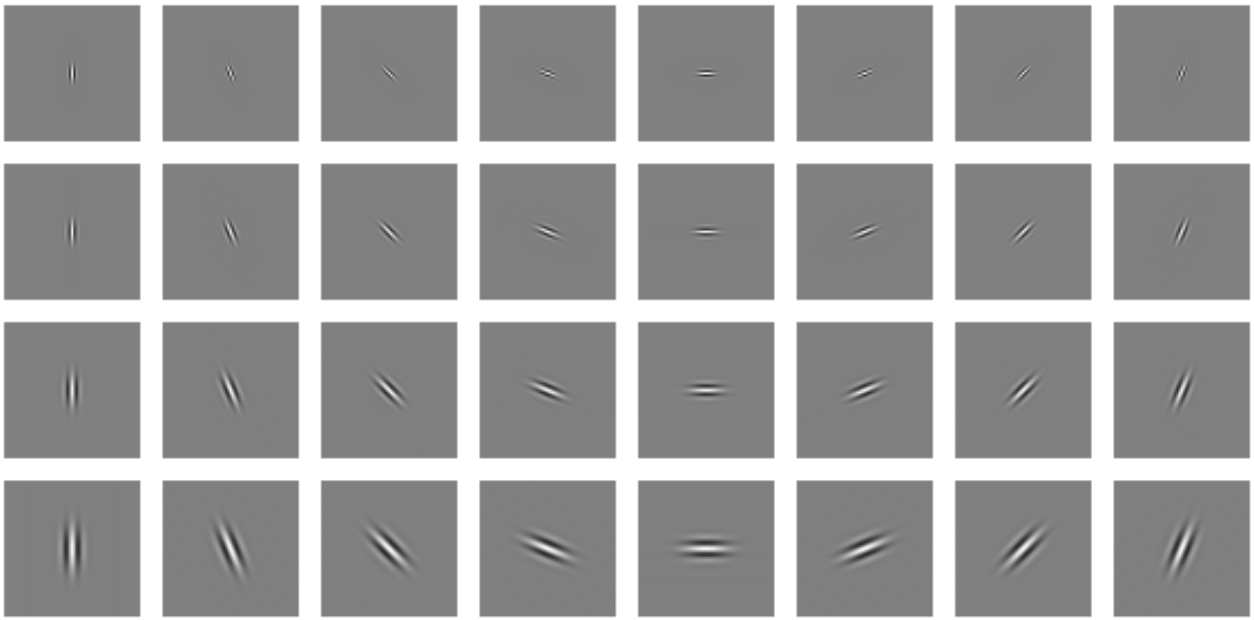




Malik & Perona, 1990. Preattentive texture discrimination with early vision mechanisms.

Clues from Human Perception

- Early processing in humans filters for various orientations and scales of frequency
- Perceptual cues in the mid-high frequencies dominate perception
- When we see an image from far away, we are effectively subsampling it



Early Visual Processing: Multi-scale edge and blob filters

Source: Hays, Brown