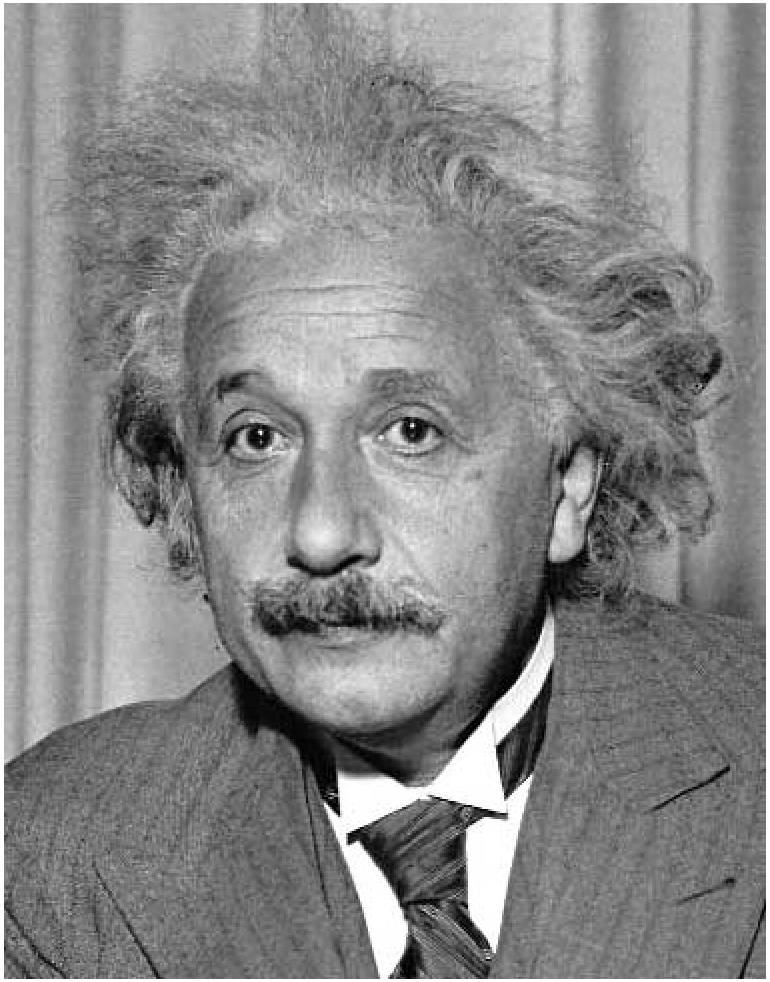
Templates, Image Pyramids, and Filter Banks

Slides adapted from James Hays, Derek Hoiem, and Others

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

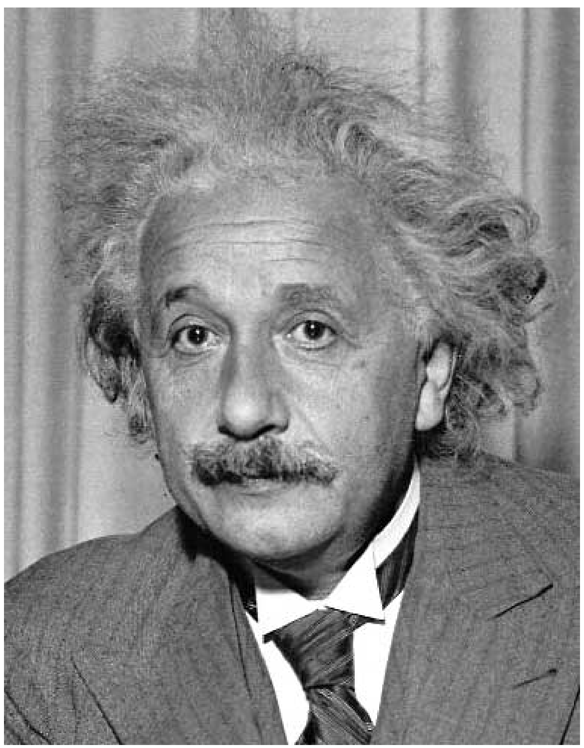


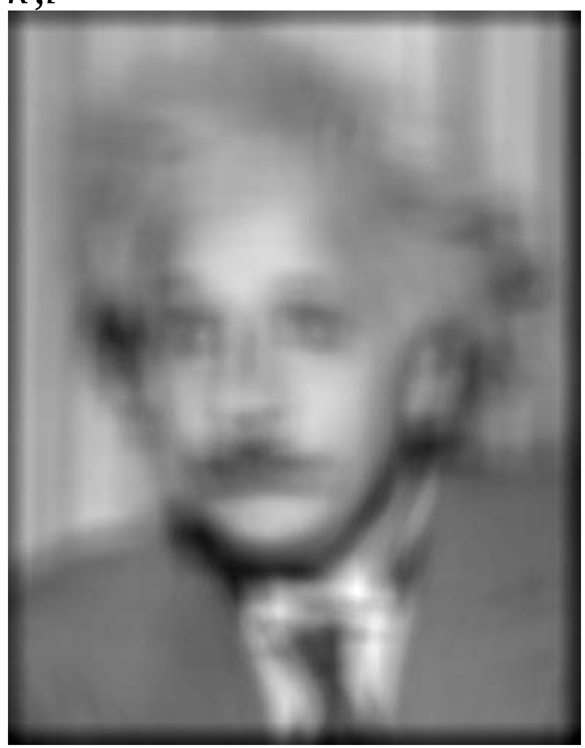
Matching with filters

- 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

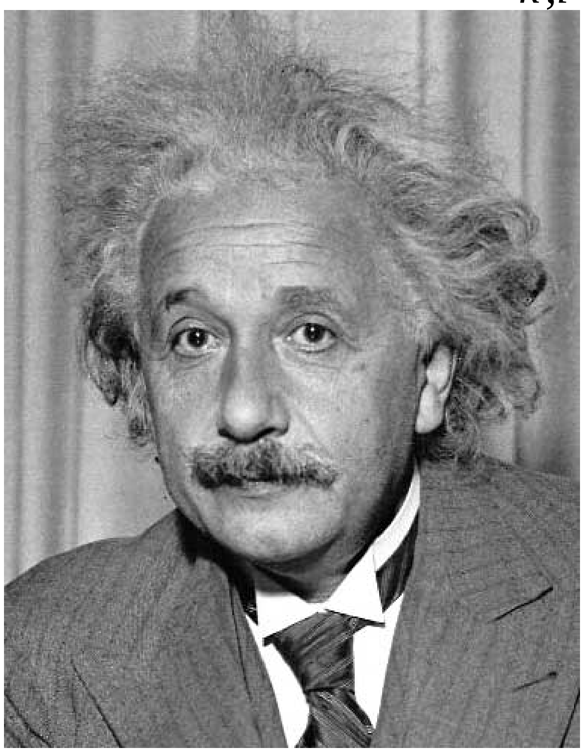
What went wrong?

Matching with filters

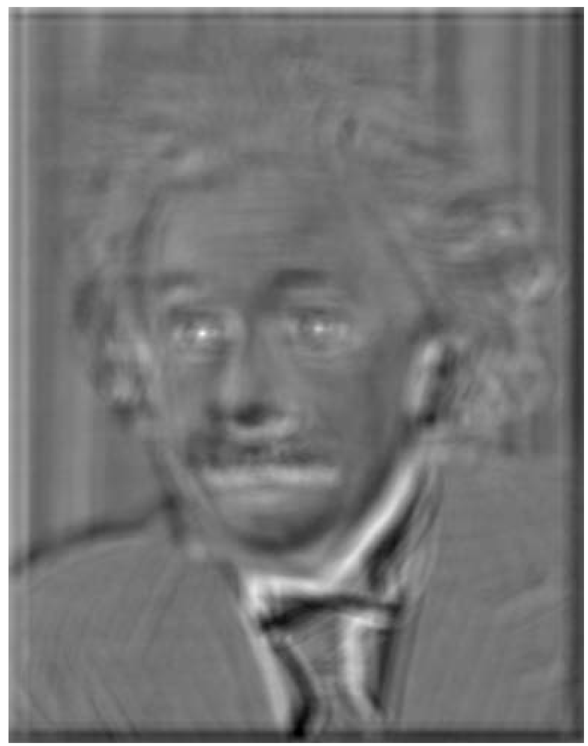
- Goal: find  in image
- Method 1: filter the image with zero-mean eye

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

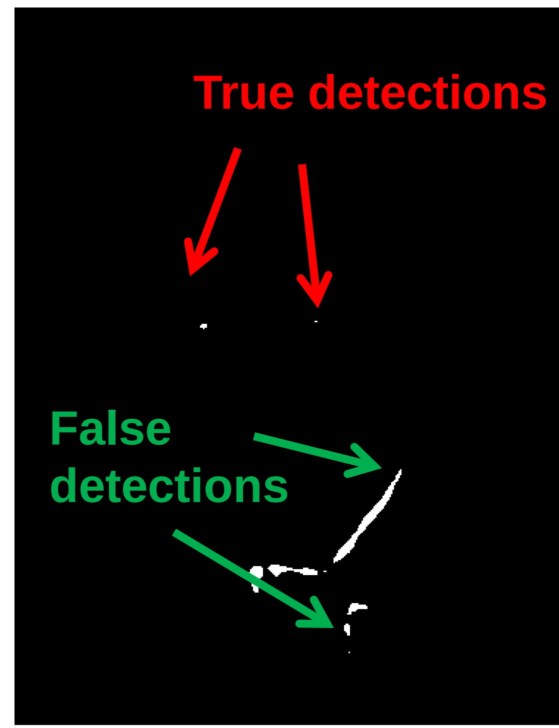
mean of f



Input



Filtered Image (scaled)

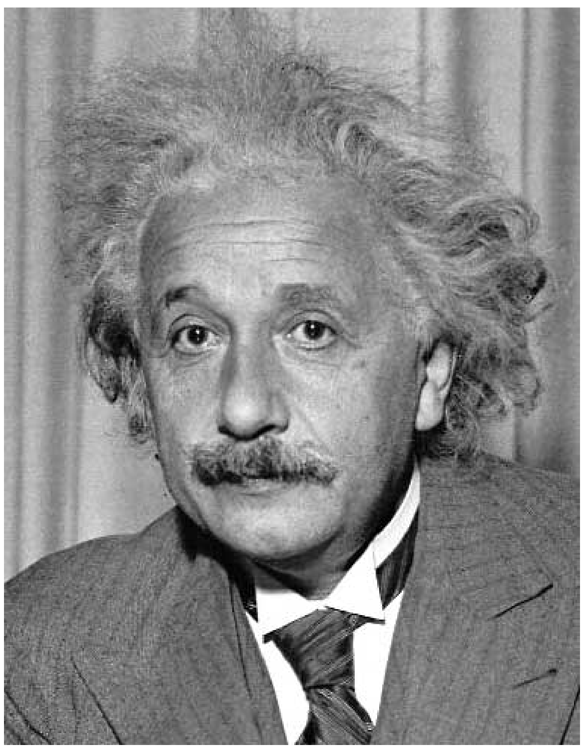


Thresholded Image

Matching with filters

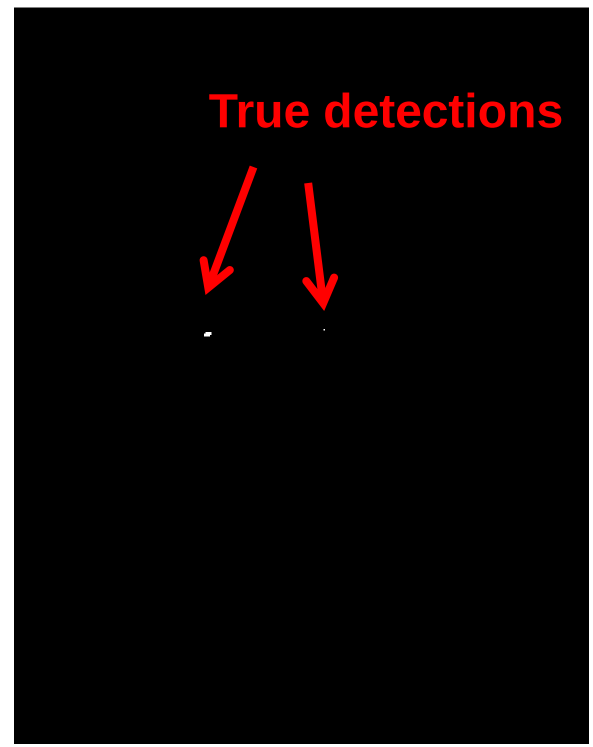
- Goal: find  in image
- Method 2: SSD

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



Input

1- sqrt(SSD)



Thresholded Image

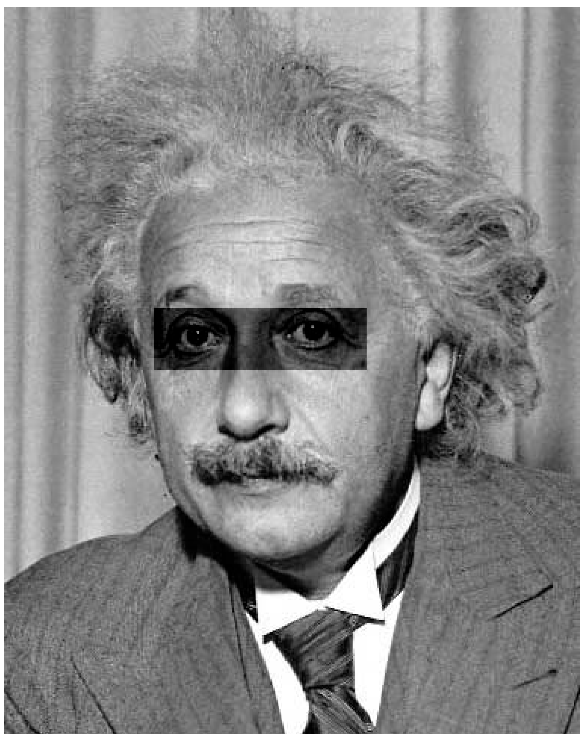
Slide: Hoiem

Matching with filters

- Goal: find  in image
- Method 2: SSD

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

What's the potential downside of SSD?



Input

1- sqrt(SSD)

Slide: Hoiem

Matching with filters

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

mean template

mean image patch

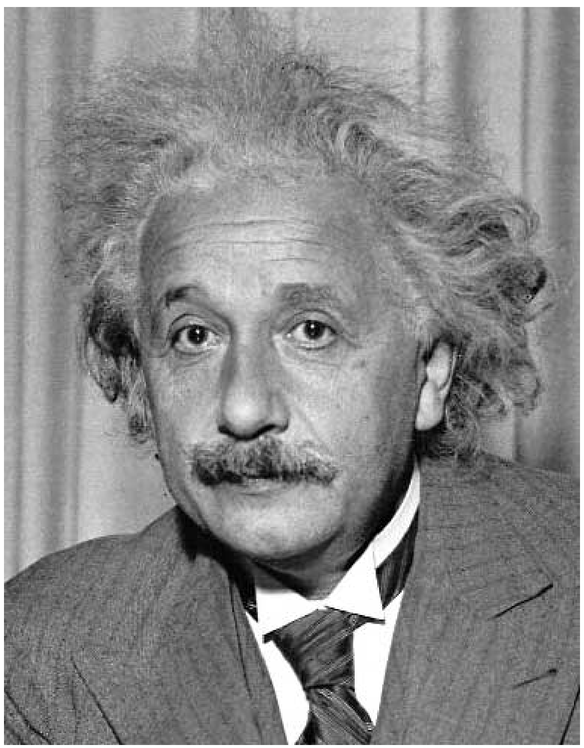
$$h[m,n]=\frac{\sum_{k,l}(g[k,l]-\bar{g})(f[m-k,n-l]-\bar{f}_{m,n})}{\left(\sum_{k,l}(g[k,l]-\bar{g})^2\sum_{k,l}(f[m-k,n-l]-\bar{f}_{m,n})^2\right)^{0.5}}$$

Matlab: `normxcorr2(template, im)`

Slide: Hoiem

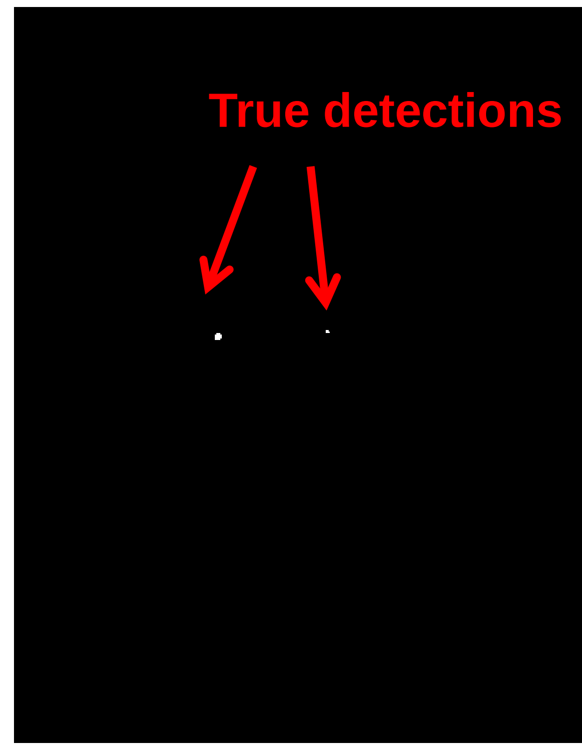
Matching with filters

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



Input

Normalized X-Correlation

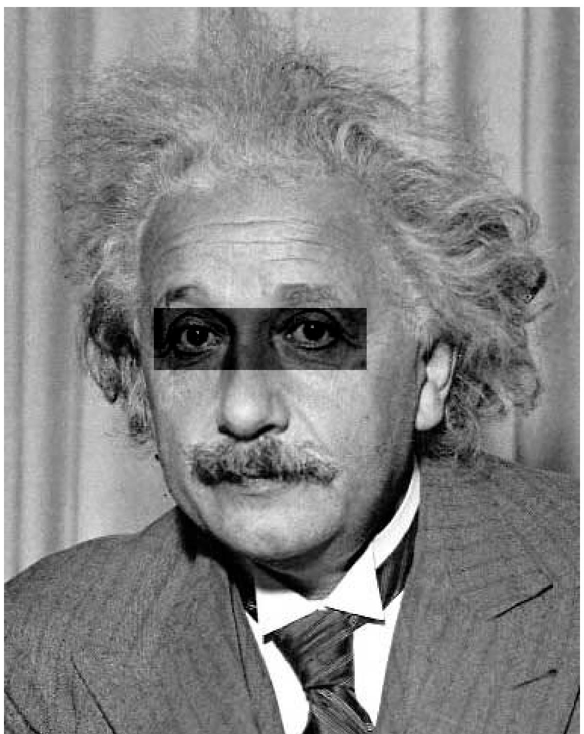


Thresholded Image

Slide: Hoiem

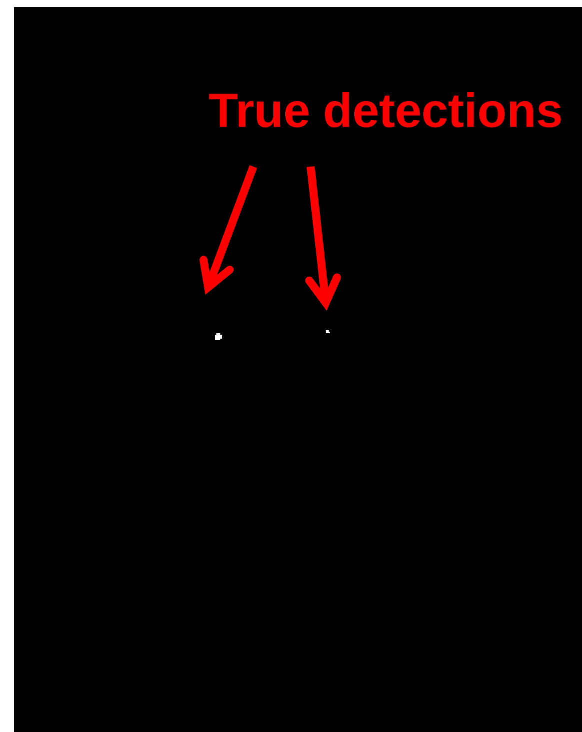
Matching with filters

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



Input

Normalized X-Correlation



Thresholded Image

Slide: Hoiem

Template Matching

Q: What is the best method to use?

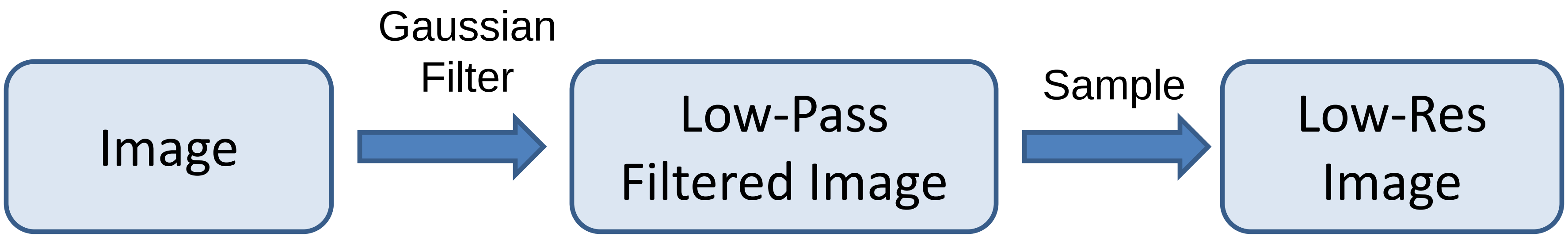
A: Depends

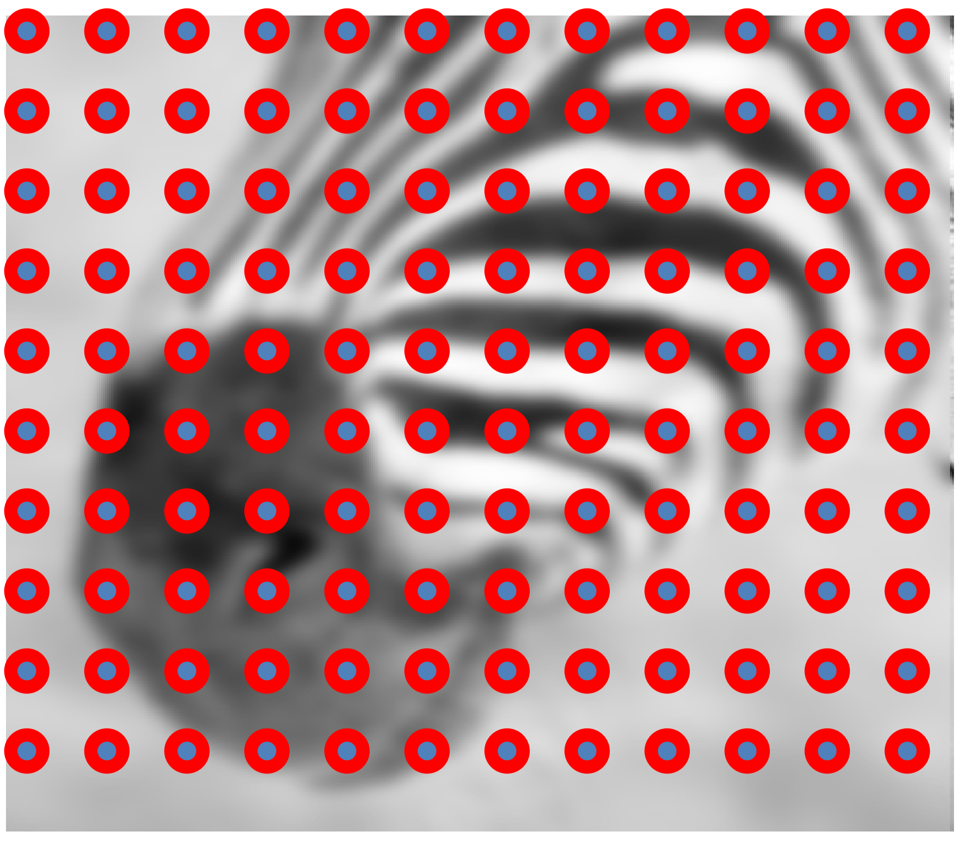
- SSD: faster, sensitive to overall intensity
- Normalized cross-correlation: slower, invariant to local average intensity and contrast

Q: What is the problem with template matching (as described so far)?

A: Template size is fixed, so we cannot find larger or smaller eyes.

Recall Sampling





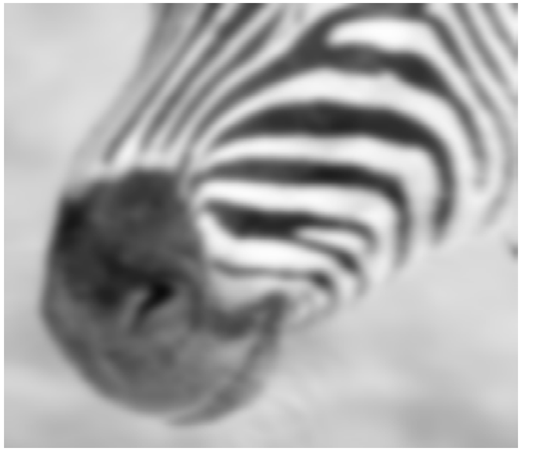


Image Pyramids

- Multi-scale image representation

- Widely used in CVPR
  - Template matching
  - Image registration
  - Hybrid Images

Gaussian pyramid

512 256 128 64 32 16 8

Template Matching with Image Pyramids

Input: Image, Template

1. Match template at current scale

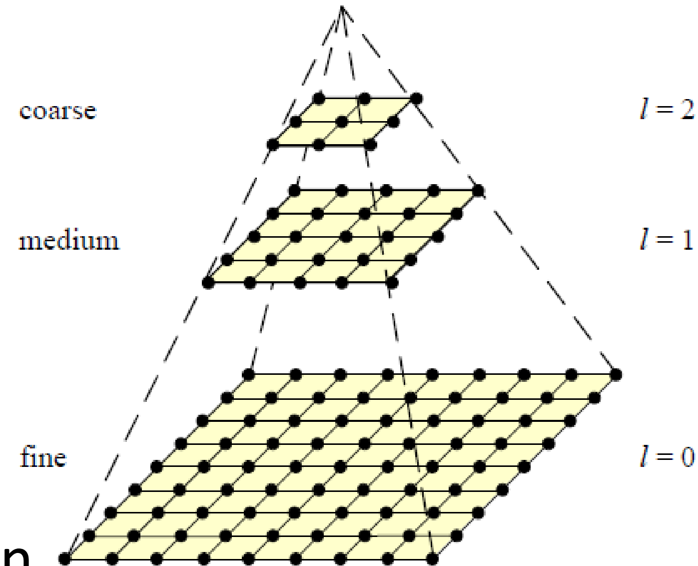
2. Downsample image

3. Repeat 1-2 until image is very small

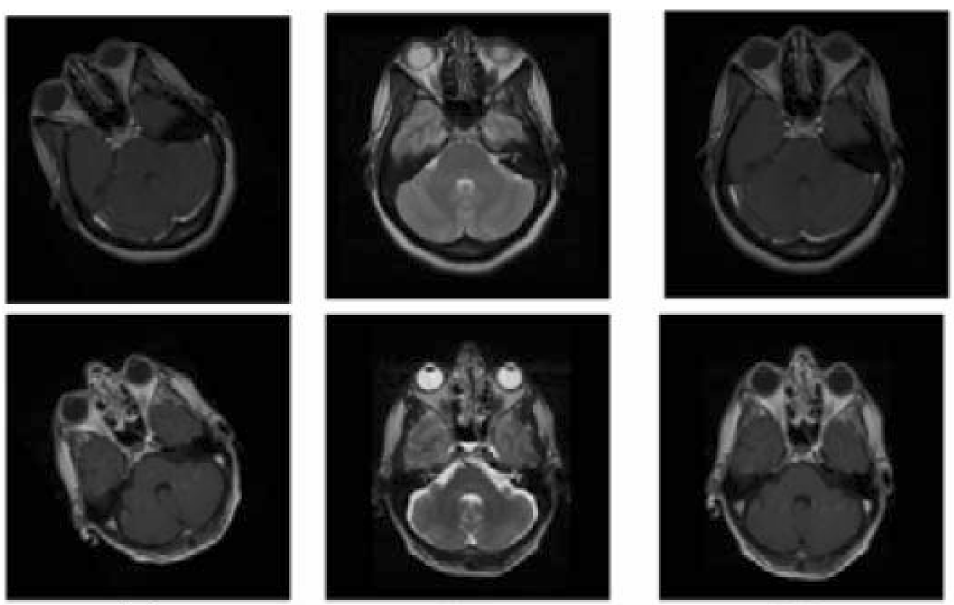
4. Take responses above some threshold, perhaps with non-maxima suppression

Coarse-to-fine Image Registration

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

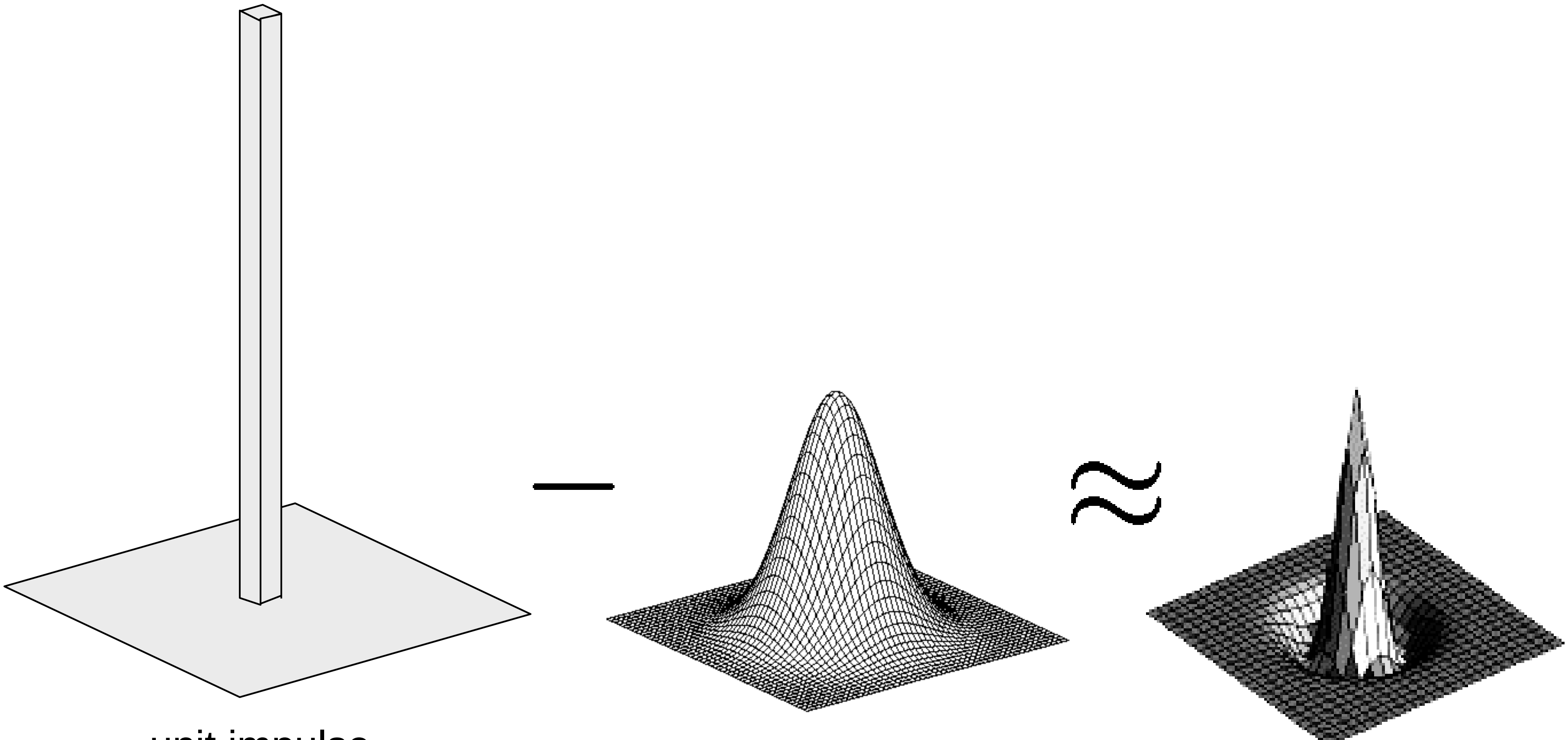


- Example: Medical image registration



Learning Based Coarse-to-fine Image Registration, Jiang et al., CVPR 2008

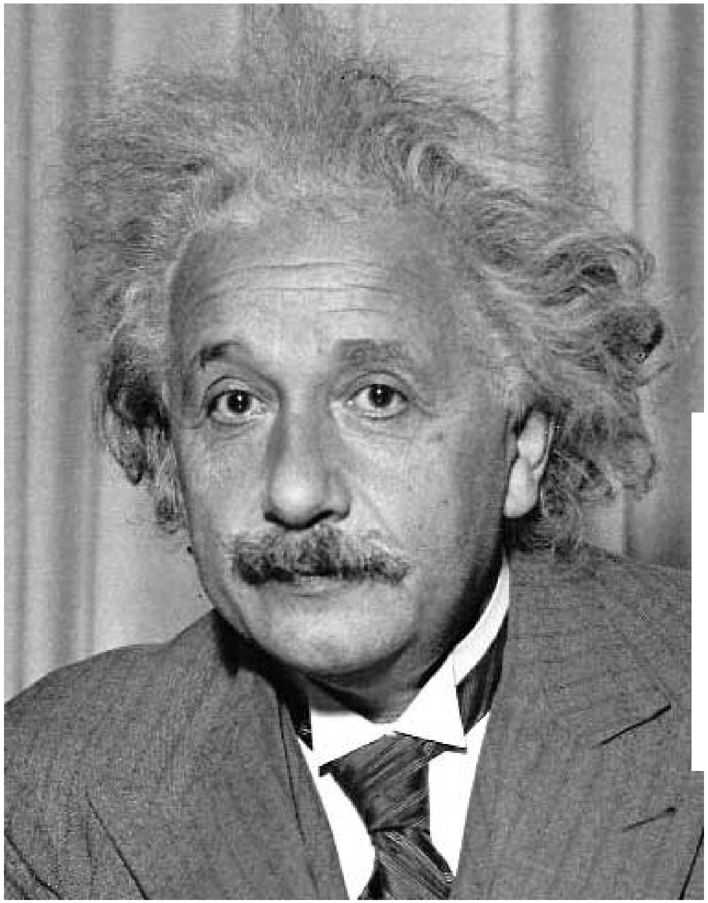
Laplacian filter

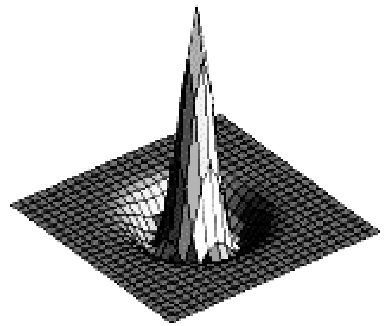


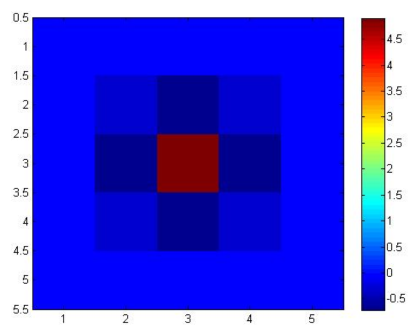
unit impulse — Gaussian ≈ Laplacian of Gaussian

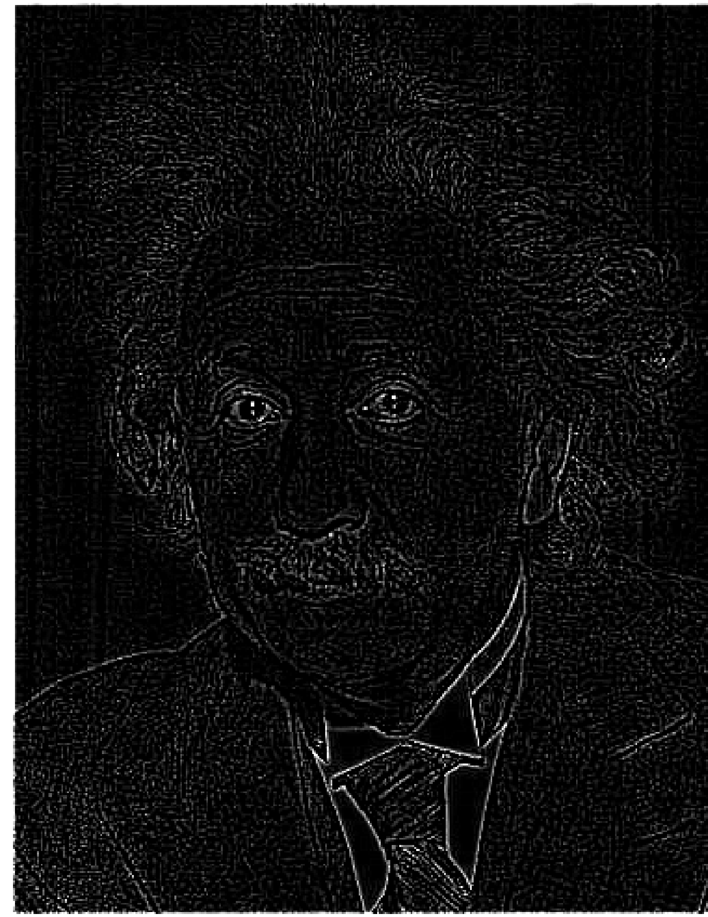
Laplacian Filter

- What does the Laplacian Filter do?



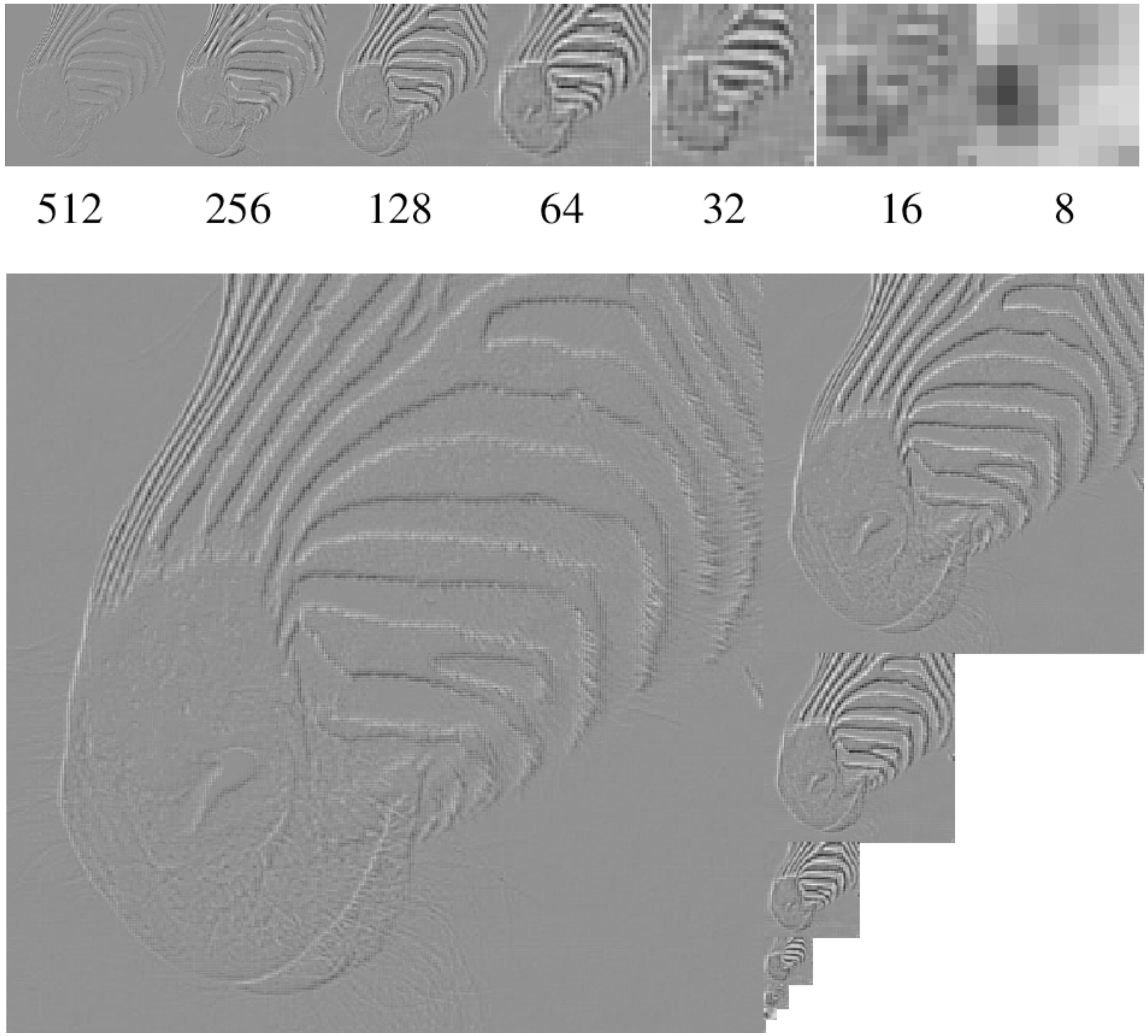




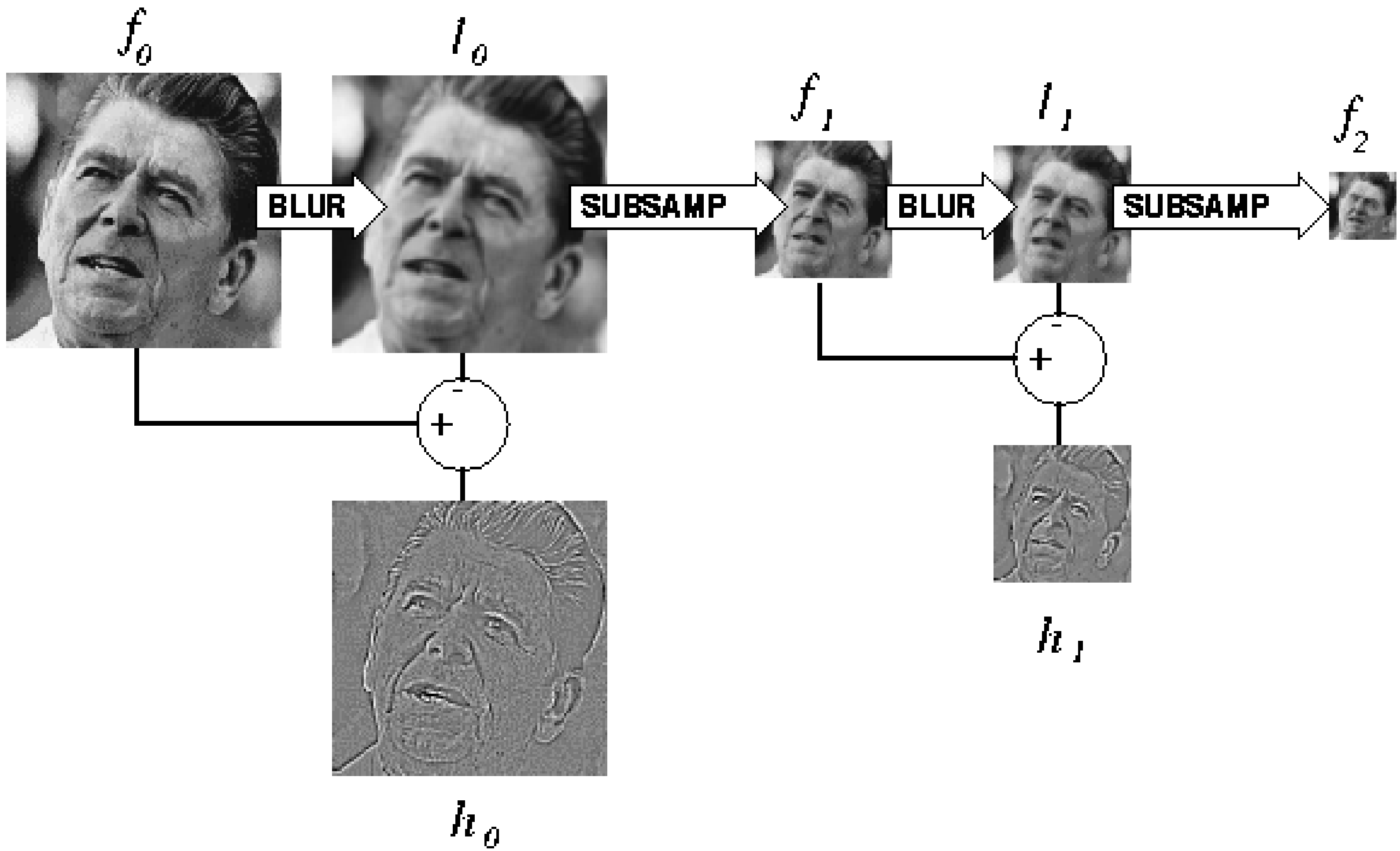


We will cover this filter (and edges, in general) next class.

Laplacian pyramid

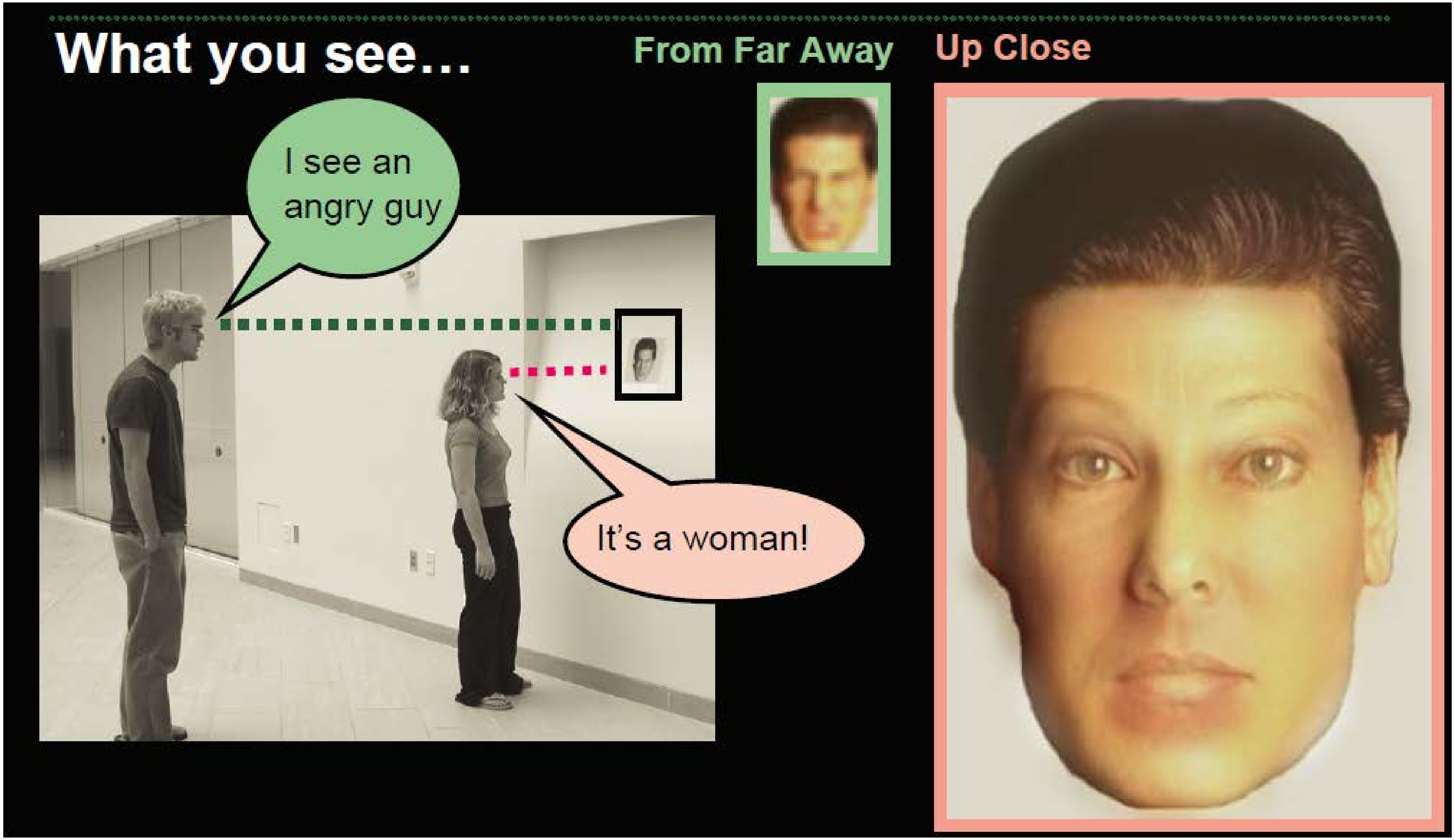


Computing Gaussian/Laplacian Pyramid



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

Hybrid Images (hw2)

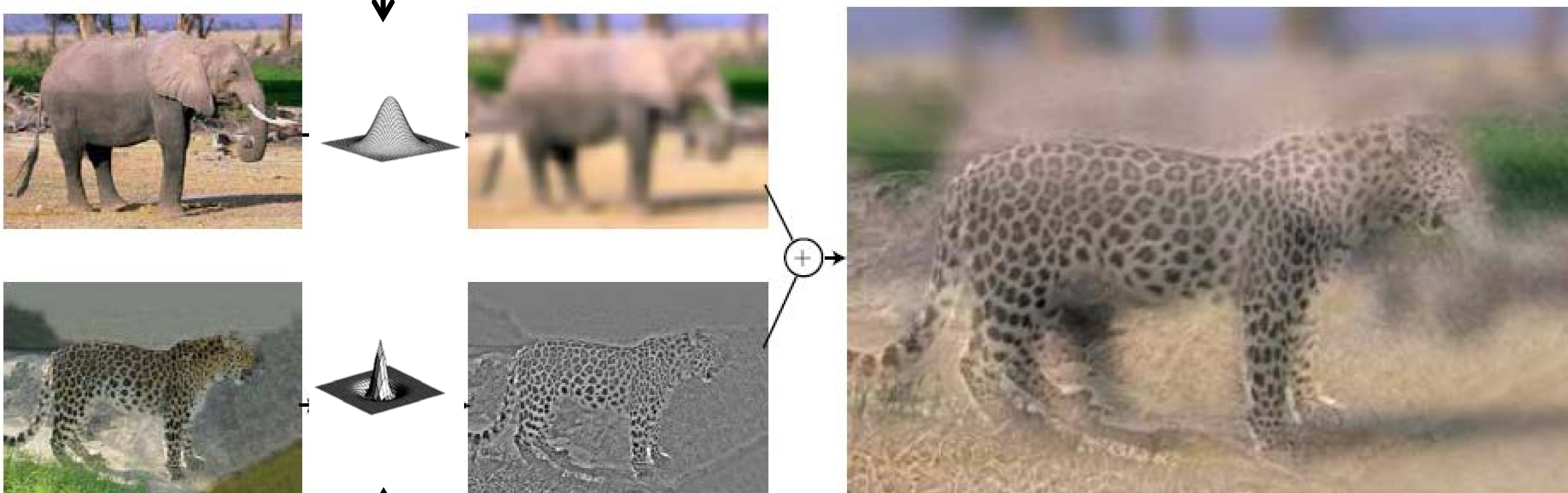


Aude Oliva & Antonio Torralba & Philippe G Schyns, SIGGRAPH 2006

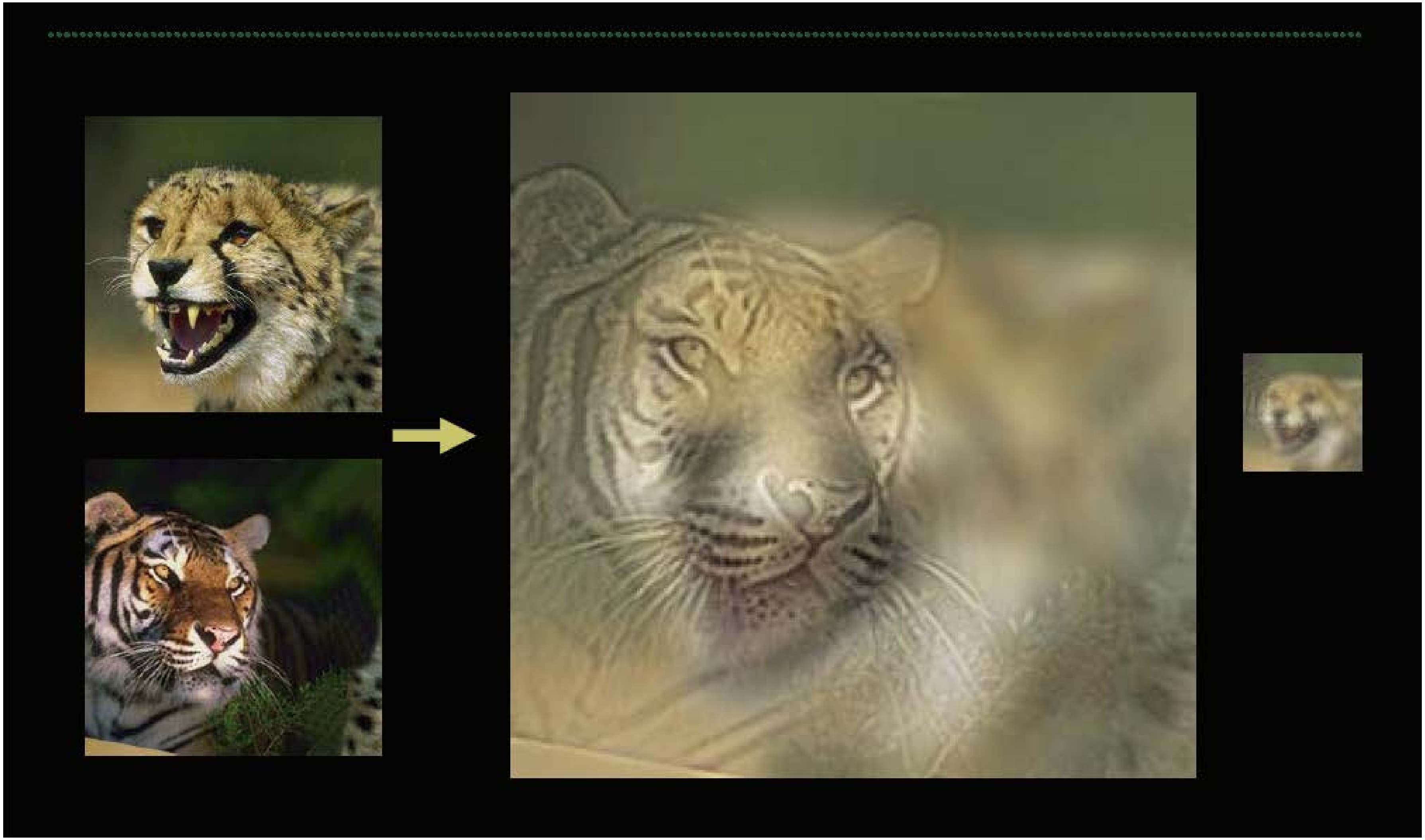
Application: Hybrid Images

A. Oliva, A. Torralba, P.G. Schyns, [“Hybrid Images,”]() SIGGRAPH 2006

Gaussian Filter



Laplacian Filter



Aude Oliva & Antonio Torralba & Philippe G Schyns, SIGGRAPH 2006

Changing expression

Sad ⟷ Surprised

Aude Oliva & Antonio Torralba & Philippe G Schyns, SIGGRAPH 2006

Image representation

- Pixels: great for spatial resolution, poor access to frequency
- Fourier transform: great for frequency, not for spatial info
- Pyramids/filter banks: balance between spatial and frequency information

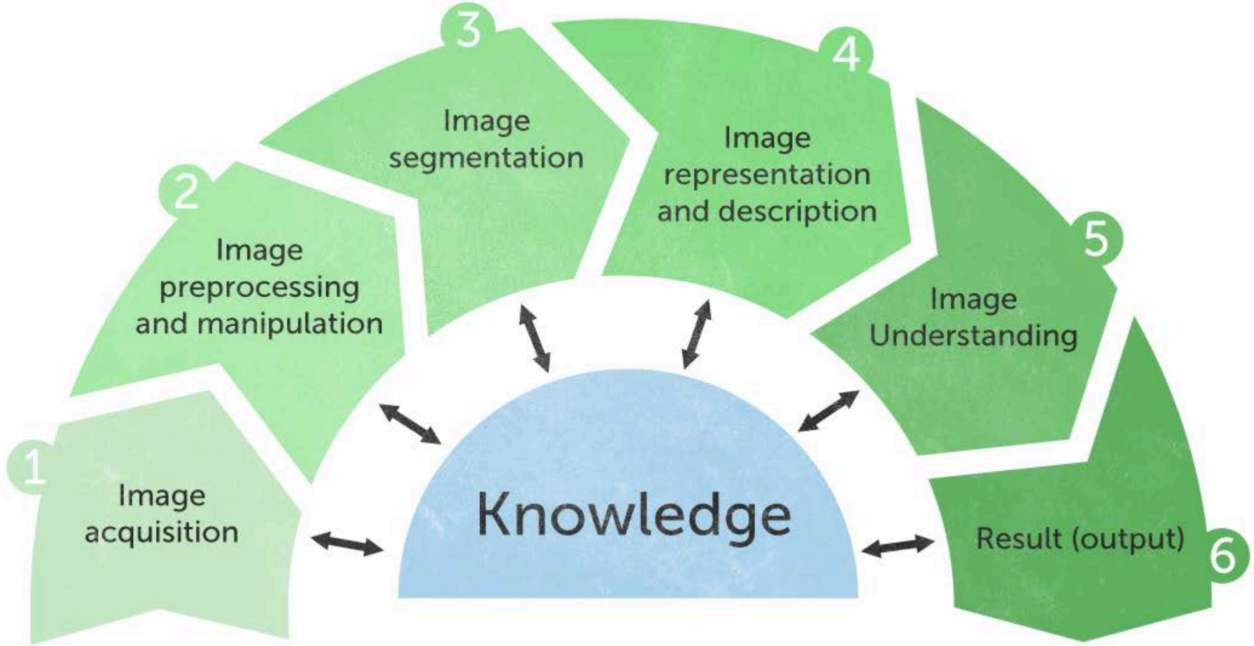
Major uses of image pyramids

- Compression
- Object detection
  - Scale search
  - Features
- Detecting stable interest points
- Registration
  - Course-to-fine

Applications of Filters

- Template matching (SSD or Normxcorr2)
  - SSD is sensitive to overall intensity
- Gaussian pyramid
  - Coarse-to-fine search, multi-scale detection
- Laplacian pyramid
  - More compact image representation
  - Can be used for compositing in graphics
- Downsampling
  - Need to sufficiently low-pass before downsampling

Coming Up…



- Continue the transition from Image Processing (Phase 2) to Computer Vision (Phases 3 & 4)
  - Interest Points
  - Finding Edges & Lines
  - Image Segmentation