# Fundamentals of Image Processing

# Review – Frequency Domain Techniques

- The name "filter" is borrowed from frequency domain processing
- Accept or reject certain frequency components
- <u>Fourier (1807):</u>
  Periodic functions could be represented as a weighted sum of sines and cosines

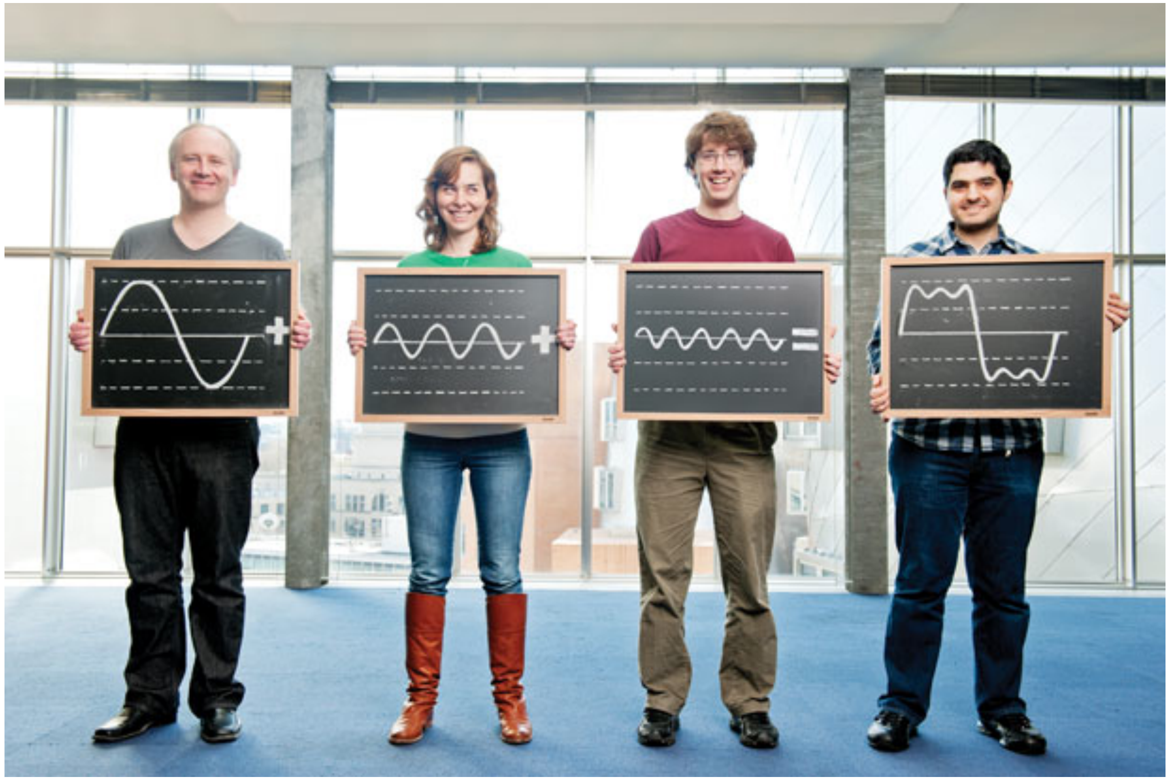


Image courtesy of Technology Review

# Review - Fourier Transform

We want to understand the frequency *w* of our signal. So, let's reparametrize the signal by *w* instead of *x*:

$$f(x) \longrightarrow \boxed{\text{Fourier Transform}} \longrightarrow F(w)$$

For every *w* from 0 to inf, ***F(w)*** holds the amplitude *A* and phase *f* of the corresponding sine $A\sin(\omega x+\phi)$

- How can *F* hold both? Complex number trick!

$$F(\omega) = R(\omega) + iI(\omega)$$

$$A = \pm\sqrt{R(\omega)^2 + I(\omega)^2} \qquad \phi = \tan^{-1}\frac{I(\omega)}{R(\omega)}$$

We can always go back:

$$F(w) \longrightarrow \boxed{\text{Inverse Fourier Transform}} \longrightarrow f(x)$$

Slide credit: A. Efros

# Review - The Discrete Fourier transform

- Forward transform

$$F[m,n] = \sum_{k=0}^{M-1}\sum_{l=0}^{N-1} f[k,l]\, e^{-\pi i\left(\frac{km}{M}+\frac{\ln}{N}\right)}$$

- Inverse transform

$$f[k,l] = \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1} F[m,n]\, e^{+\pi i\left(\frac{km}{M}+\frac{\ln}{N}\right)}$$

Slide credit: B. Freeman and A. Torralba

# Review - The Convolution Theorem

- The Fourier transform of the convolution of two functions is the product of their Fourier transforms

$$\mathrm{F}[g * h] = \mathrm{F}[g]\,\mathrm{F}[h]$$

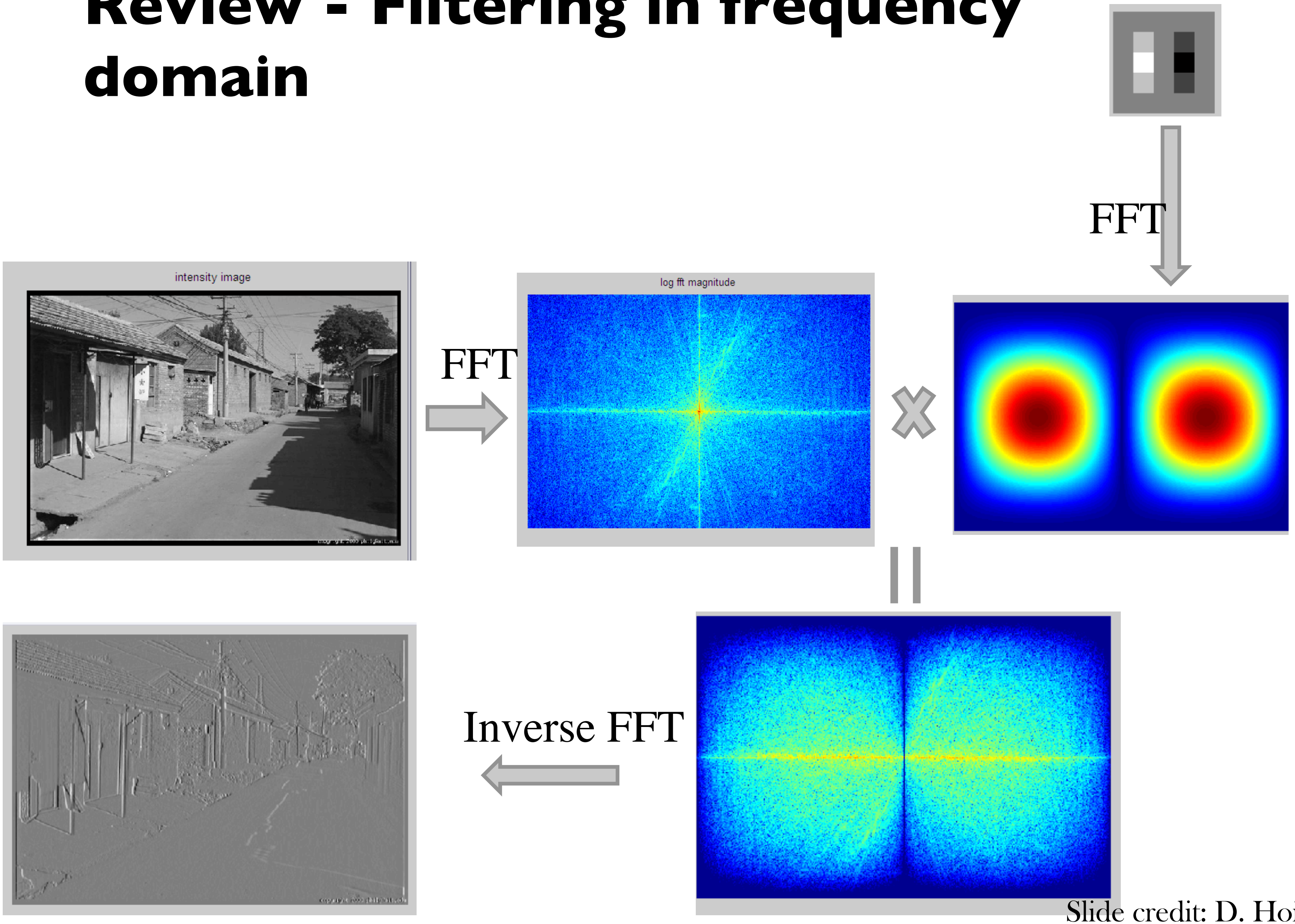
- The inverse Fourier transform of the product of two Fourier transforms is the convolution of the two inverse Fourier transforms

$$\mathrm{F}^{-1}[gh] = \mathrm{F}^{-1}[g] * \mathrm{F}^{-1}[h]$$

- **Convolution** in spatial domain is equivalent to **multiplication** in frequency domain!

Slide credit: A. Efros

# Review - Filtering in frequency domain




Slide credit: D. Hoiem

# Image pyramids

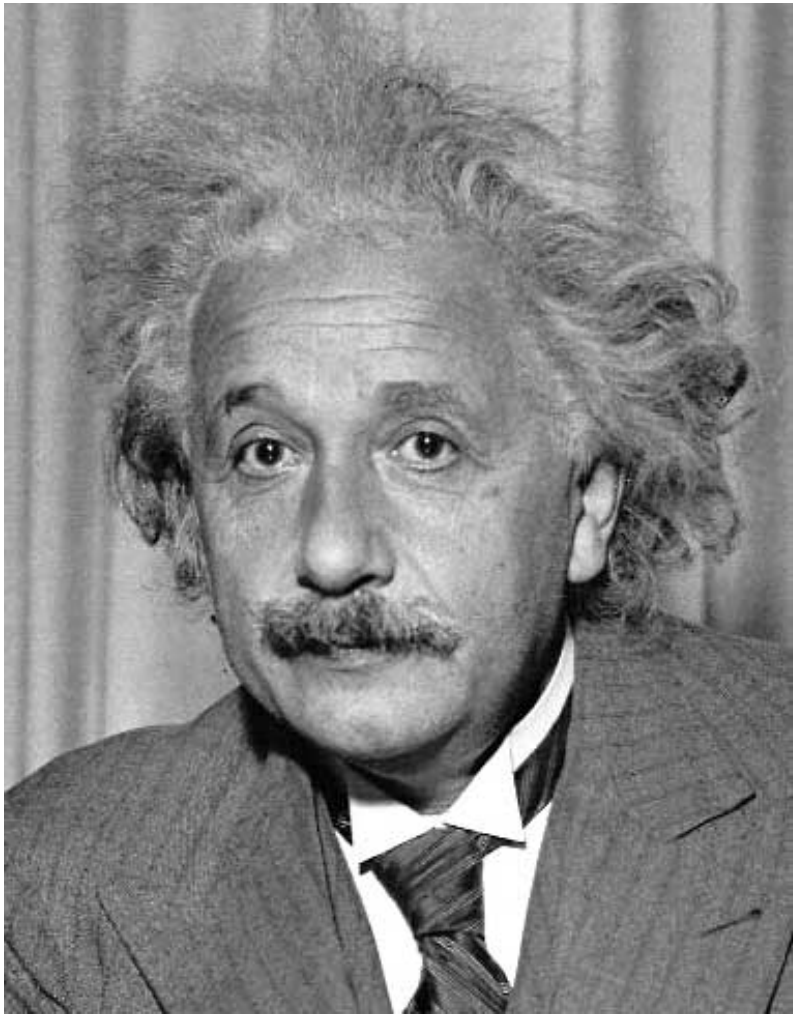
- Gaussian pyramid
- Laplacian pyramid
- Wavelet/QMF pyramid
- Steerable pyramid

Slide credit: B. Freeman and A. Torralba

# 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



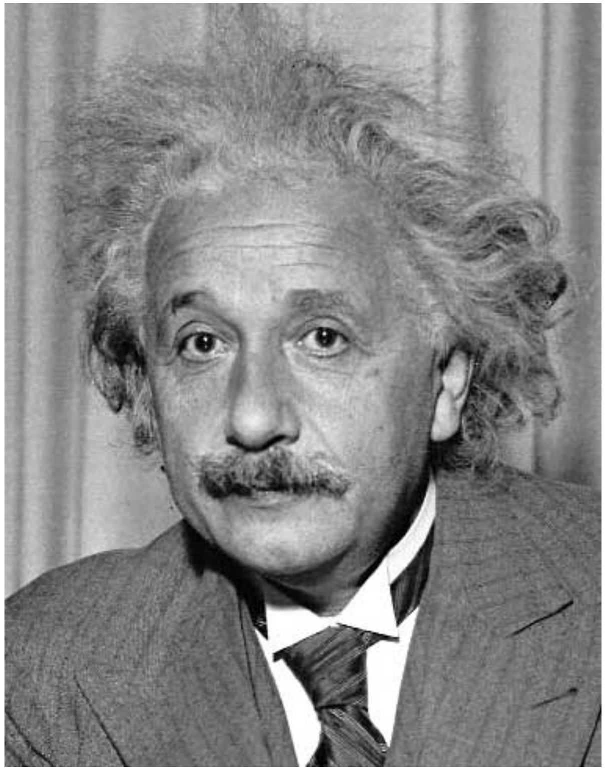
Slide: Hoiem

# Matching with filters

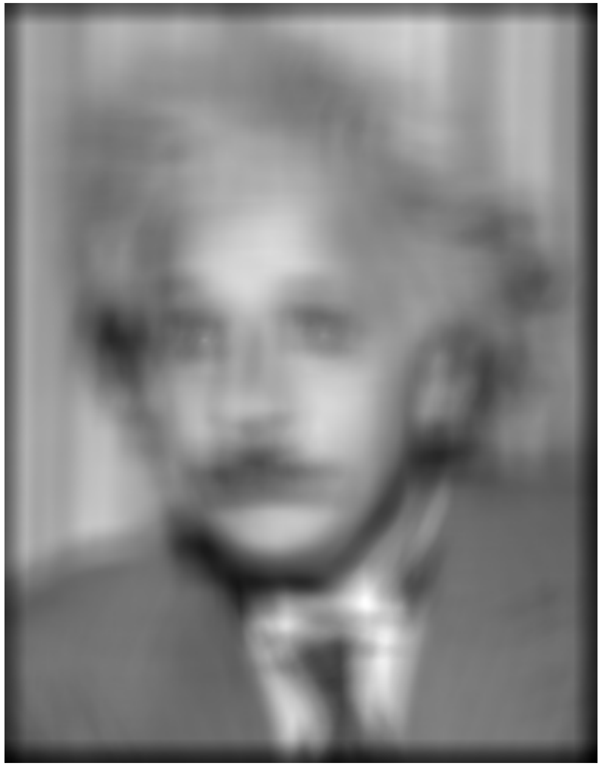
- Goal: find  in image
- Method 0: filter the image with eye patch

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

f = image
g = filter



Input



Filtered Image

What went wrong?

response is stronger for higher intensity

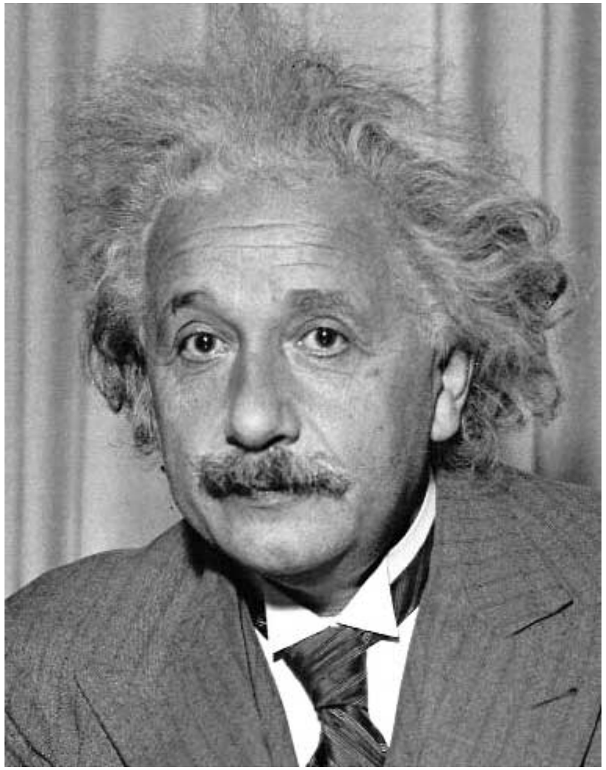
Slide: Hoiem

# Matching with filters

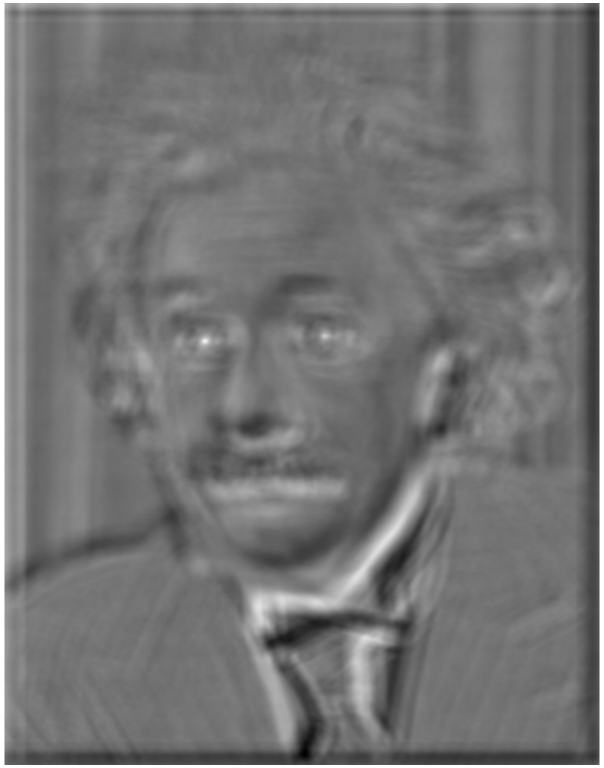
- Goal: find  in image
- Method 1: filter the image with zero-mean eye

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

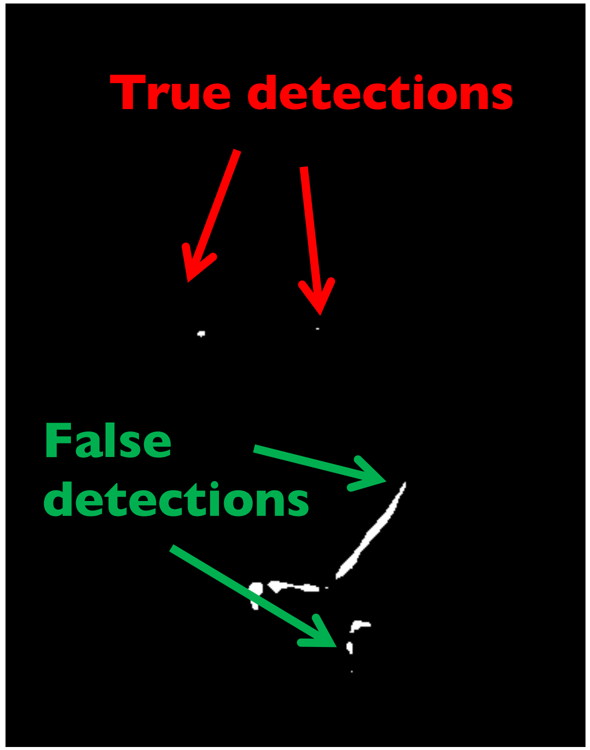
mean of f



Input



Filtered Image (scaled)



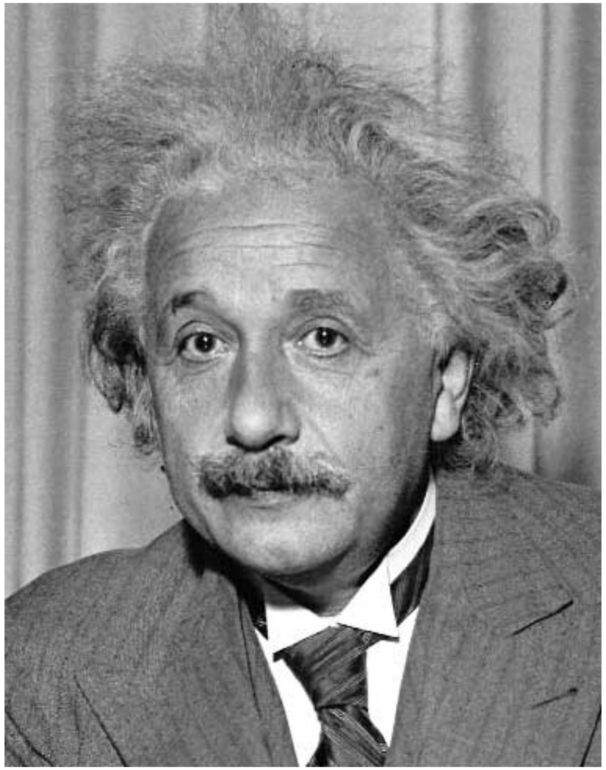

Thresholded Image

Slide: Hoiem

# Matching with filters

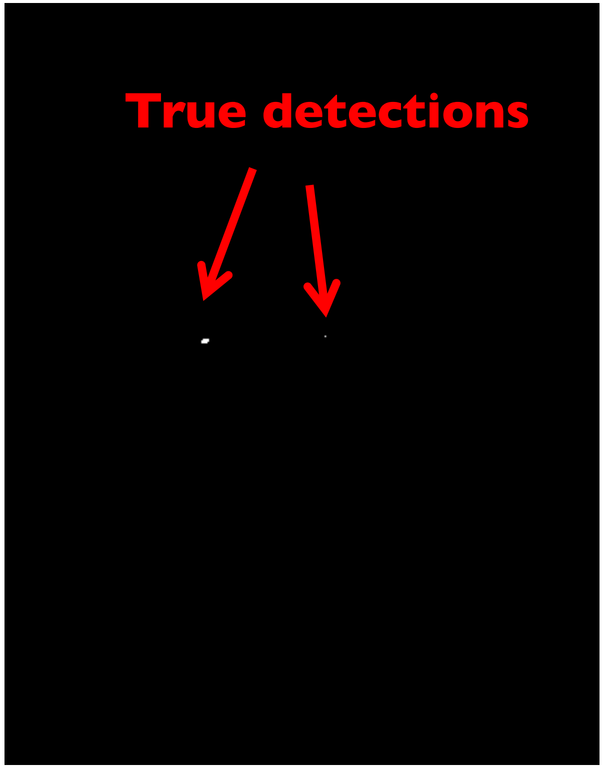
- Goal: find  in image
- Method 2: SSD

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



Input

1- sqrt(SSD)



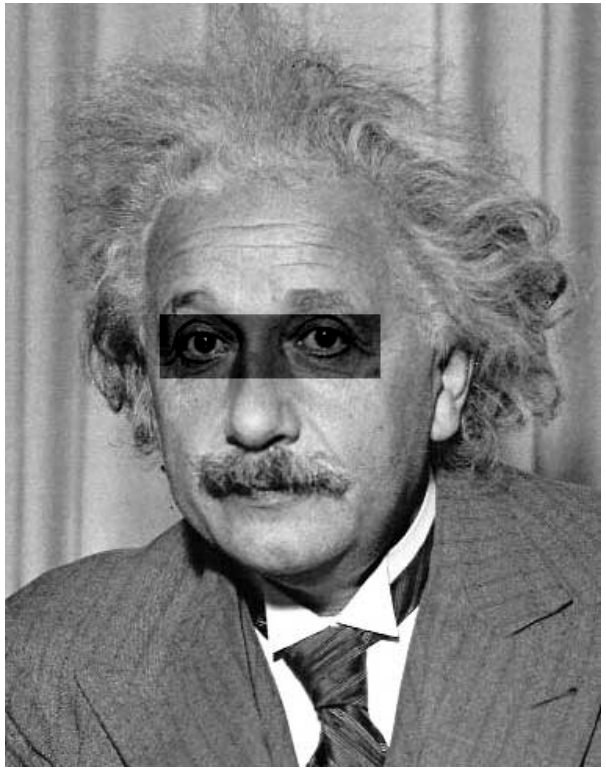

Thresholded Image

Slide: Hoiem

# Matching with filters

- Goal: find  in image
- Method 2: SSD

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



Input

1- sqrt(SSD)

What's the potential downside of SSD?

SSD sensitive to average intensity

# Matching with filters

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

mean template

mean image patch

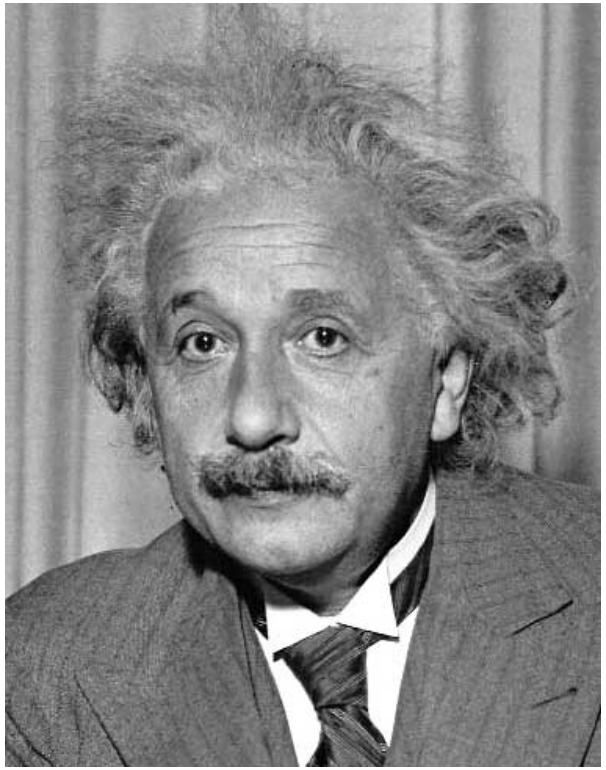
$$h[m,n] = \frac{\sum_{k,l}(g[k,l]-\bar{g})(f[m-k,n-l]-\bar{f}_{m,n})}{\left(\sum_{k,l}(g[k,l]-\bar{g})^2 \sum_{k,l}(f[m-k,n-l]-\bar{f}_{m,n})^2\right)^{0.5}}$$

standard deviation of
intensity values of the
image in the area overlaed
by template.

Slide: Hoiem

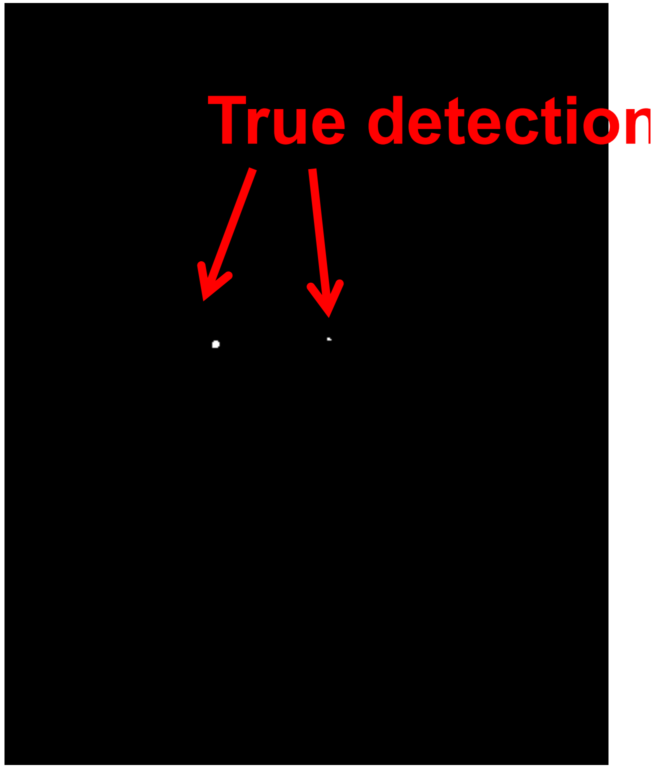
# Matching with filters

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



Input

Normalized X-Correlation



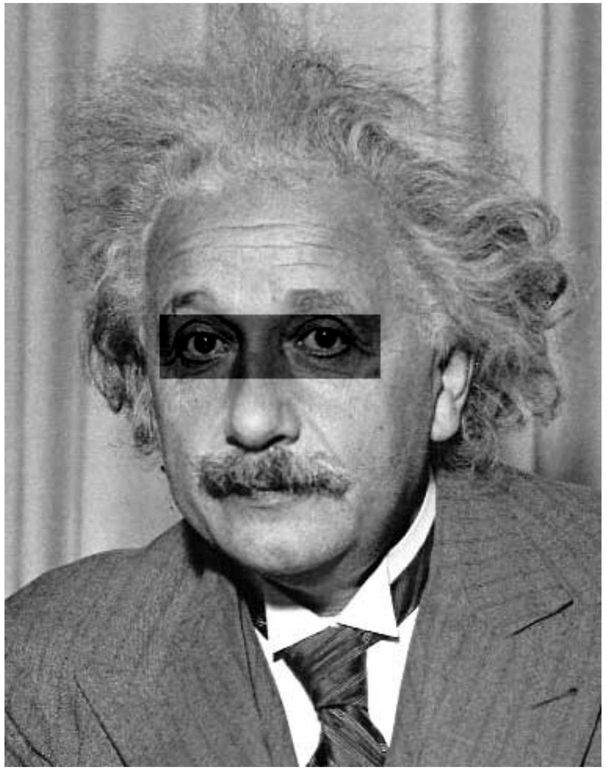

Thresholded Image

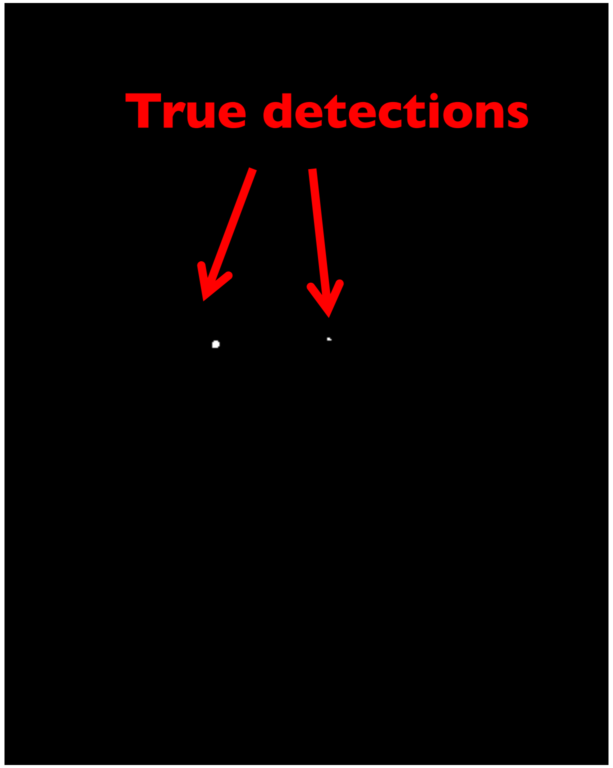
Slide: Hoiem

# Matching with filters

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

Invariant to mean and scale of intensity






Slide: Hoiem

Input

Normalized X-Correlation

Thresholded Image

# Q: What is the best method to use?

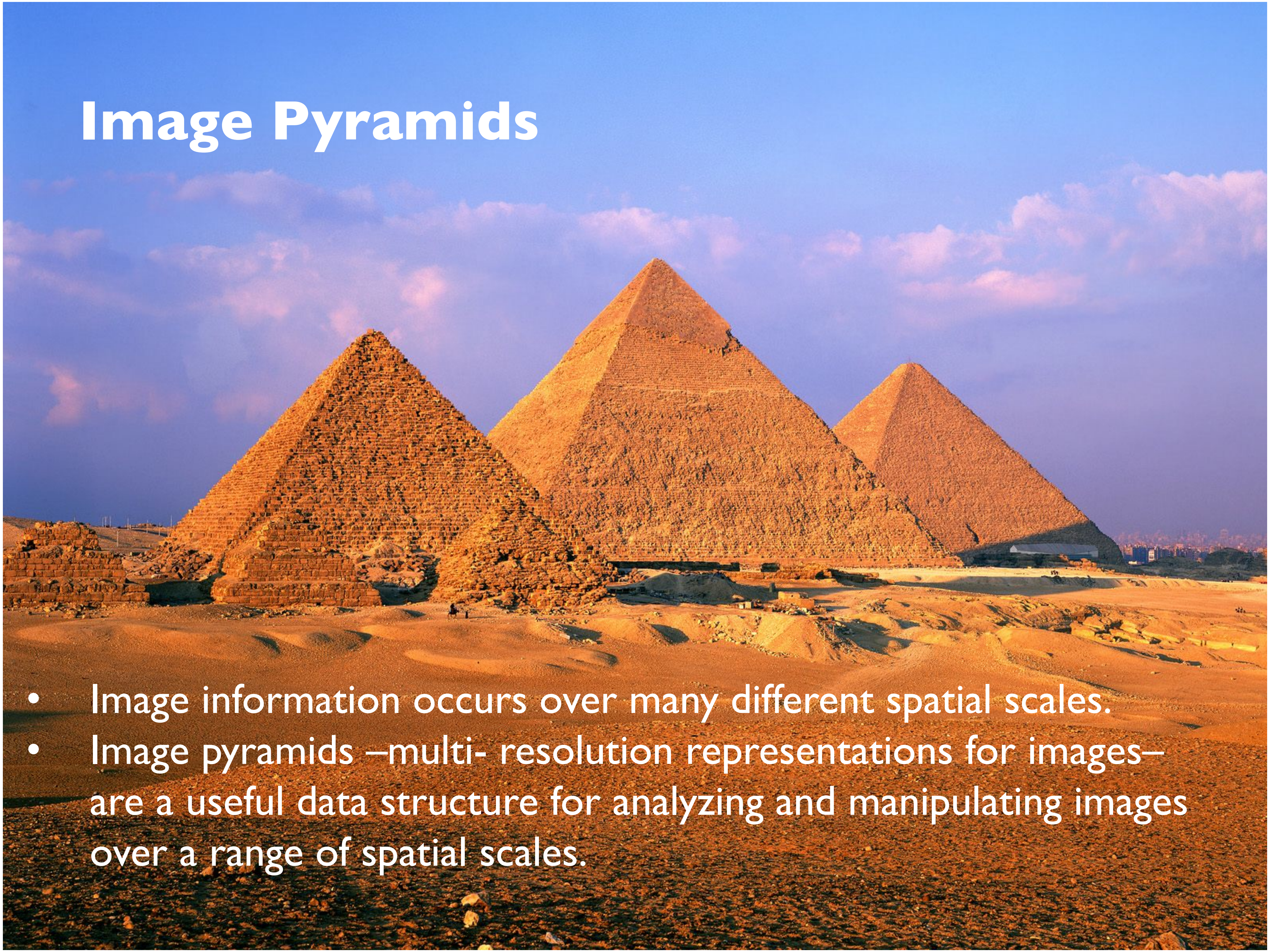
A: Depends

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

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

A: Image Pyramid

# Image Pyramids



- Image information occurs over many different spatial scales.
- Image pyramids –multi- resolution representations for images– are a useful data structure for analyzing and manipulating images over a range of spatial scales.

# Image pyramids

- Gaussian pyramid
- Laplacian pyramid
- Wavelet/QMF pyramid
- Steerable pyramid

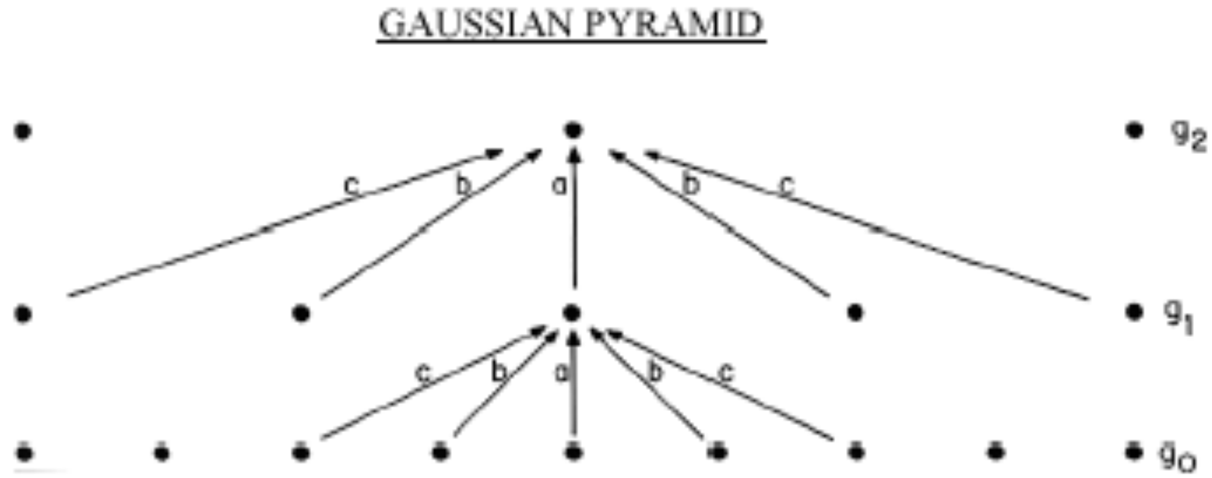
# Gaussian pyramid
# Review of Sampling

# The Gaussian pyramid

- Smooth with Gaussians, because
  - A Gaussian*Gaussian = another Gaussian
- Gaussians are low pass filters, so representation is redundant.
- Gaussian pyramid creates versions of the input image at multiple resolutions.
- This is useful for analysis across different spatial scales, but doesn't separate the image into different frequency bands.

Slide adapted from: B. Freeman and A. Torralba

# The computational advantage of pyramids




$g_0$ = IMAGE

$g_L$ = REDUCE [$g_{L-1}$]

Fig 1. A one-dimensional graphic representation of the process which generates a Gaussian pyramid Each row of dots represents nodes within a level of the pyramid. The value of each node in the zero level is just the gray level of a corresponding image pixel. The value of each node in a high level is the weighted average of node values in the next lower level. Note that node spacing doubles from level to level, while the same weighting pattern or "generating kernel" is used to generate all levels.

[Burt and Adelson, 1983]

Slide credit: B. Freeman and A. Torralba

# The Gaussian Pyramid

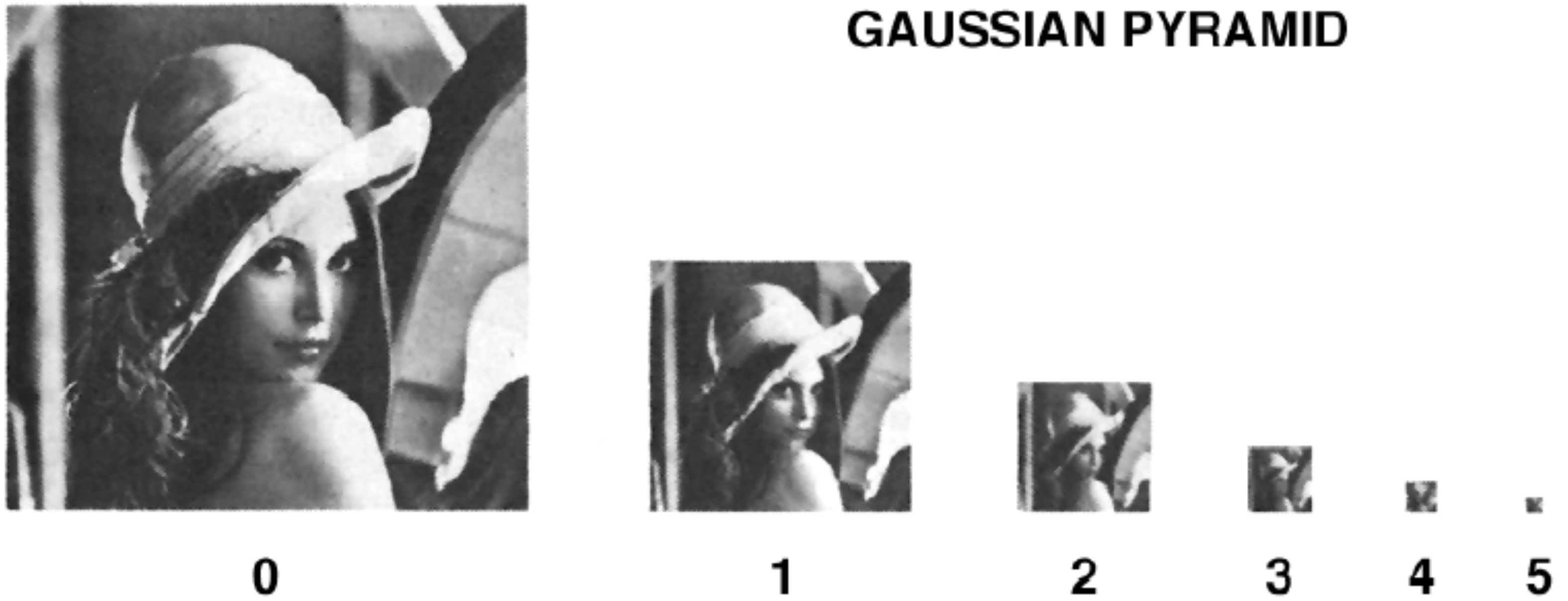



Fig. 4. First six levels of the Gaussian pyramid for the "Lady" image The original image, level 0, meusures 257 by 257 pixels and each higher level array is roughly half the dimensdons of its predecessor. Thus, level 5 measures just 9 by 9 pixels.

[Burt and Adelson, 1983]

Slide credit: B. Freeman and A. Torralba

512 256 128 64 32 16 8

Slide credit: B. Freeman and A. Torralba

# Convolution and subsampling as a matrix multiply (1D case)

$$x_2 = G_1 x_1$$

$$G_1 =$$

$$\begin{matrix}
1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0
\end{matrix}$$

Slide credit: B. Freeman and A. Torralba

(Normalization constant of 1/16 omitted for visual clarity.)

# Next pyramid level

$$x_3 = G_2 x_2$$

$$G_2 = \begin{matrix} 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 \end{matrix}$$

Slide credit: B. Freeman and A. Torralba

# The combined effect of the two pyramid levels

$$x_3 = G_2 G_1 x_1$$

$$G_2 G_1 =$$

$$\begin{matrix}
1 & 4 & 10 & 20 & 31 & 40 & 44 & 40 & 31 & 20 & 10 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 4 & 10 & 20 & 31 & 40 & 44 & 40 & 31 & 20 & 10 & 4 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 10 & 20 & 31 & 40 & 44 & 40 & 30 & 16 & 4 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 10 & 20 & 25 & 16 & 4 & 0
\end{matrix}$$

Slide credit: B. Freeman and A. Torralba

# Gaussian pyramids used for

- up- or down- sampling images.
- Multi-resolution image analysis
  - Look for an object over various spatial scales
  - Coarse-to-fine image processing: form blur estimate or the motion analysis on very low-resolution image, upsample and repeat. Often a successful strategy for avoiding local minima in complicated estimation tasks.

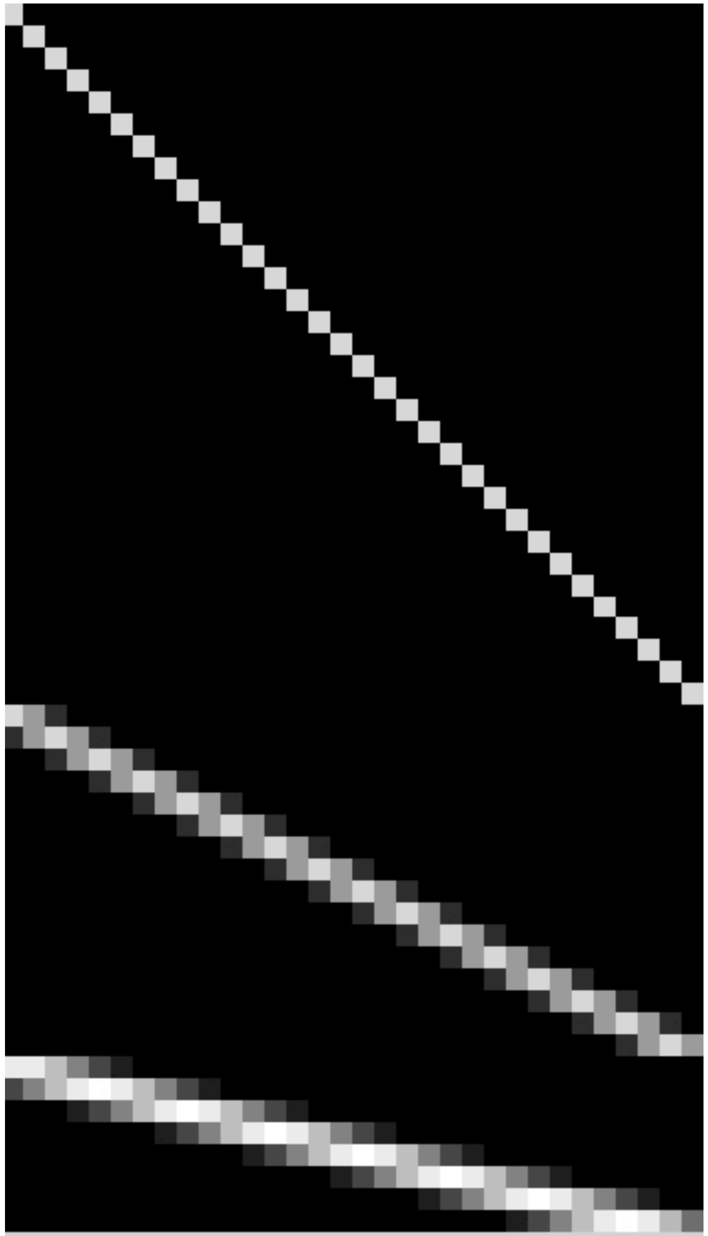
Slide credit: B. Freeman and A. Torralba

# 1D Gaussian pyramid matrix, for [1 4 6 4 1] low-pass filter

full-band image,
highest resolution

lower-resolution
image

lowest resolution
image



Slide credit: B. Freeman and A. Torralba

# 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

Slide: Hoiem

# Image pyramids

- Gaussian pyramid
- Laplacian pyramid
- Wavelet/QMF pyramid
- Steerable pyramid

Slide credit: B. Freeman and A. Torralba

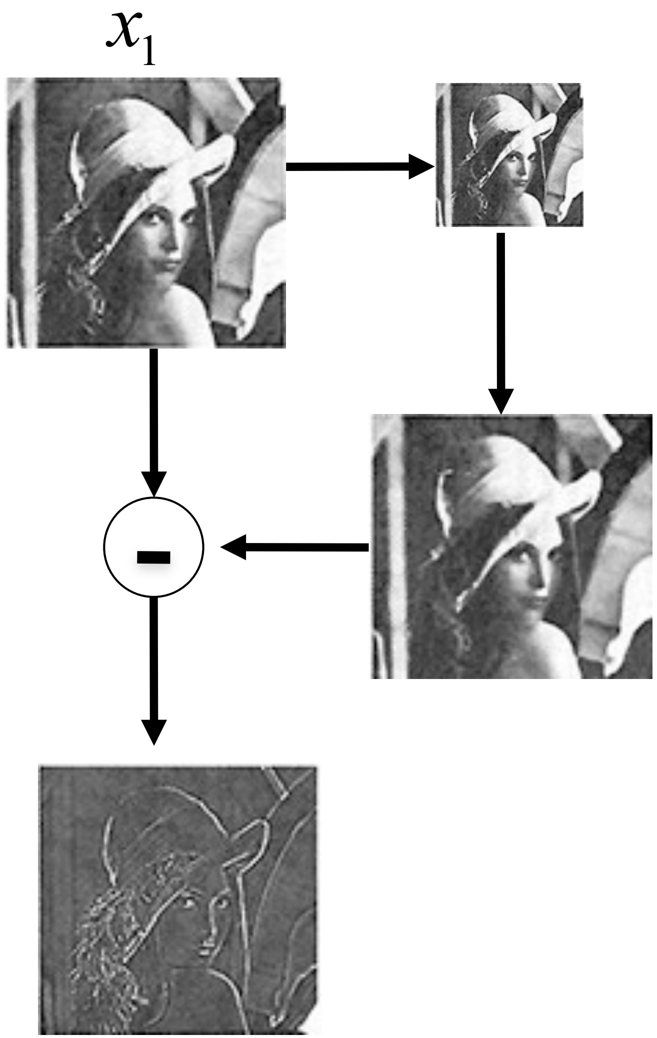
# The Laplacian Pyramid

- Synthesis
  - Compute the difference between upsampled Gaussian pyramid level and Gaussian pyramid level.
  - band pass filter - each level represents spatial frequencies (largely) unrepresented at other level.

- Laplacian pyramid provides an extra level of analysis as compared to Gaussian pyramid by breaking the image into different isotropic spatial frequency bands.

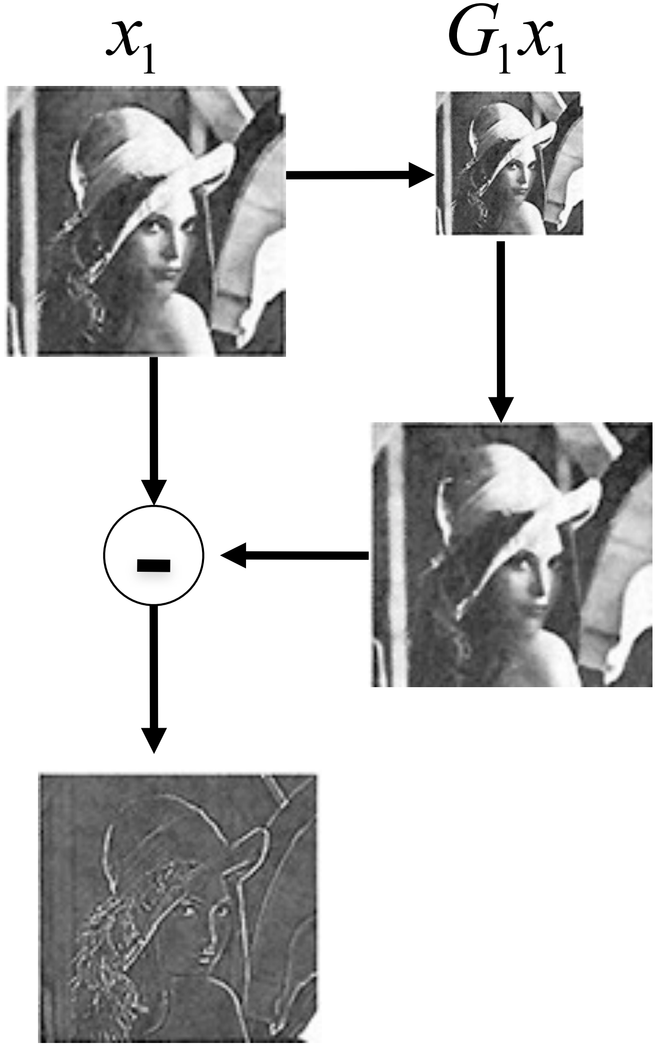
Slide adapted from: B. Freeman and A. Torralba

# The Laplacian Pyramid



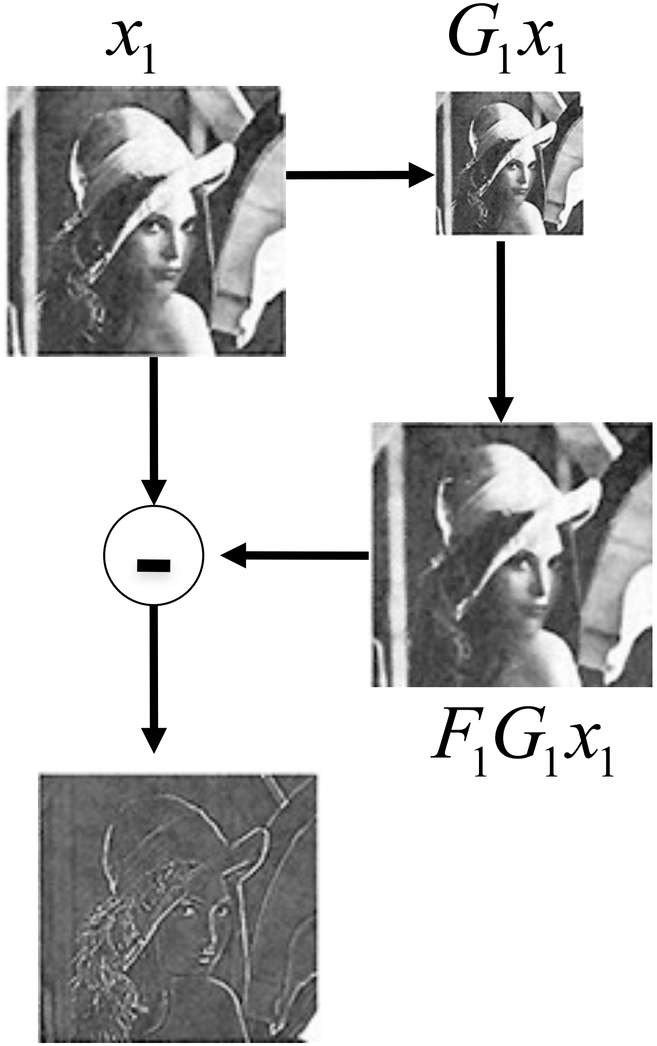

Slide credit: B. Freeman and A. Torralba

# The Laplacian Pyramid



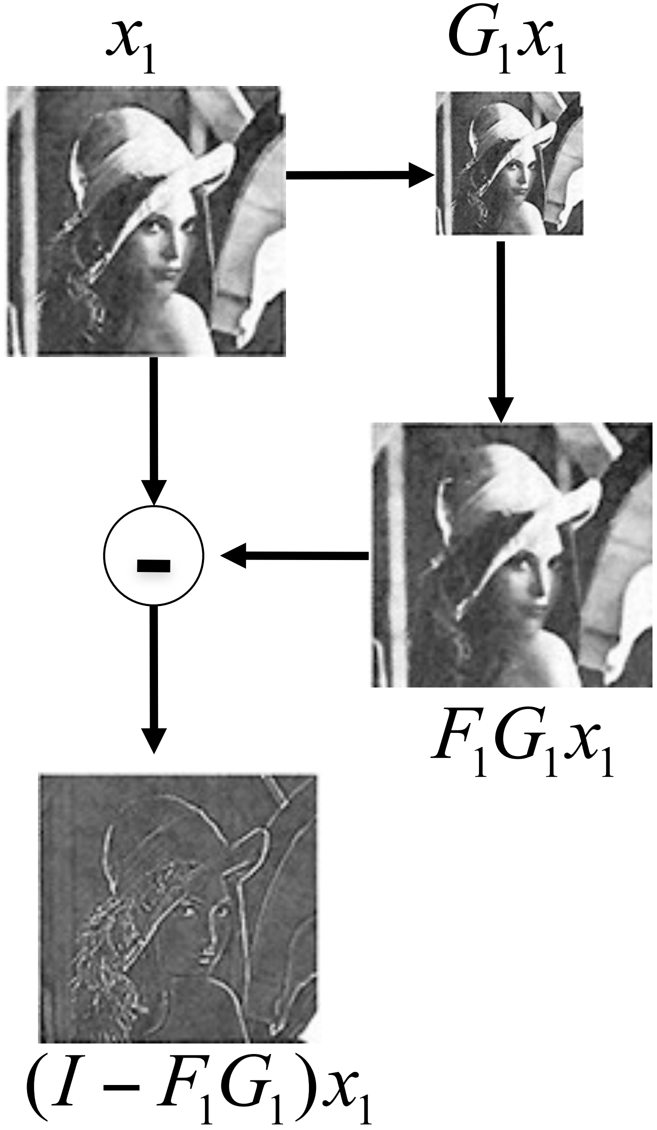

Slide credit: B. Freeman and A. Torralba

# The Laplacian Pyramid



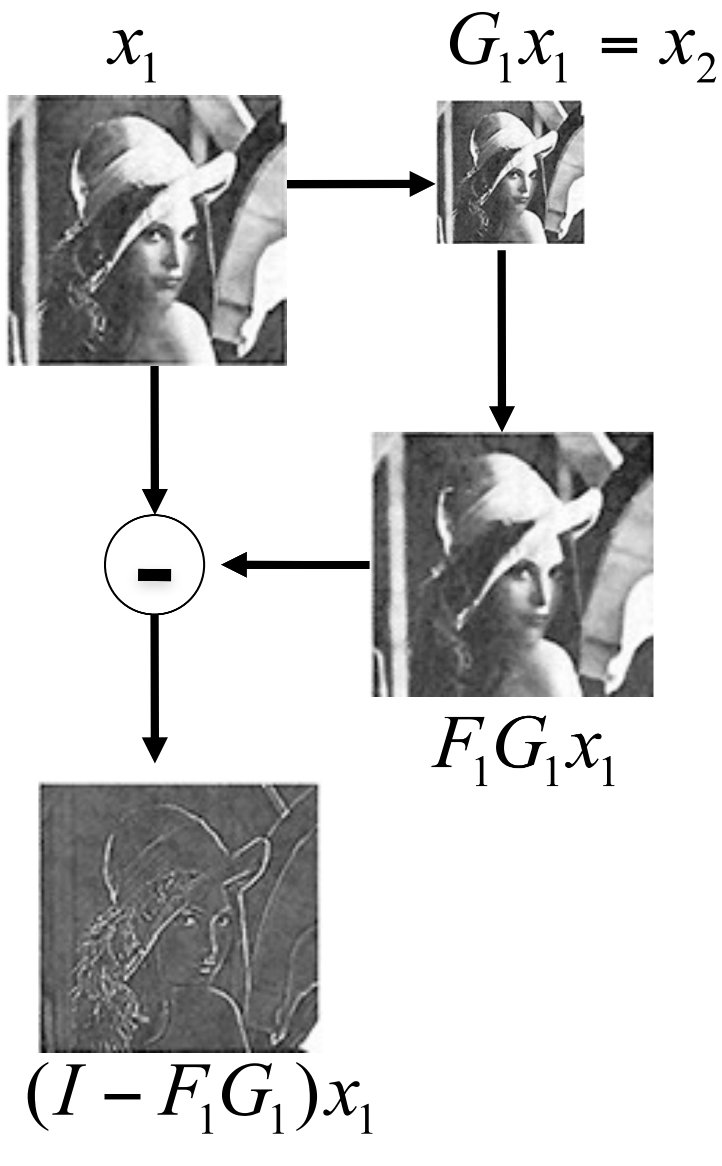

Slide credit: B. Freeman and A. Torralba

# The Laplacian Pyramid



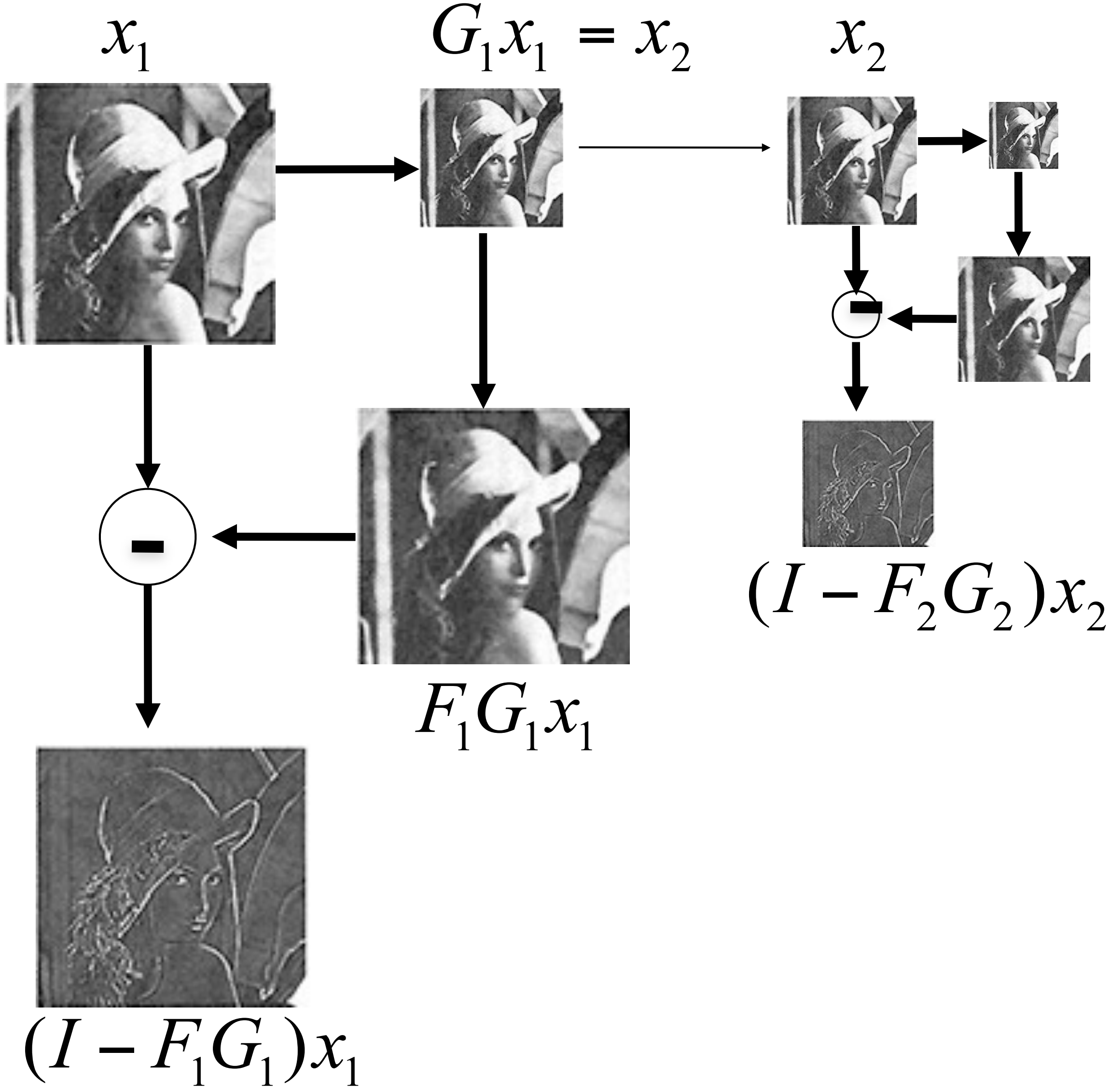

Slide credit: B. Freeman and A. Torralba

# The Laplacian Pyramid



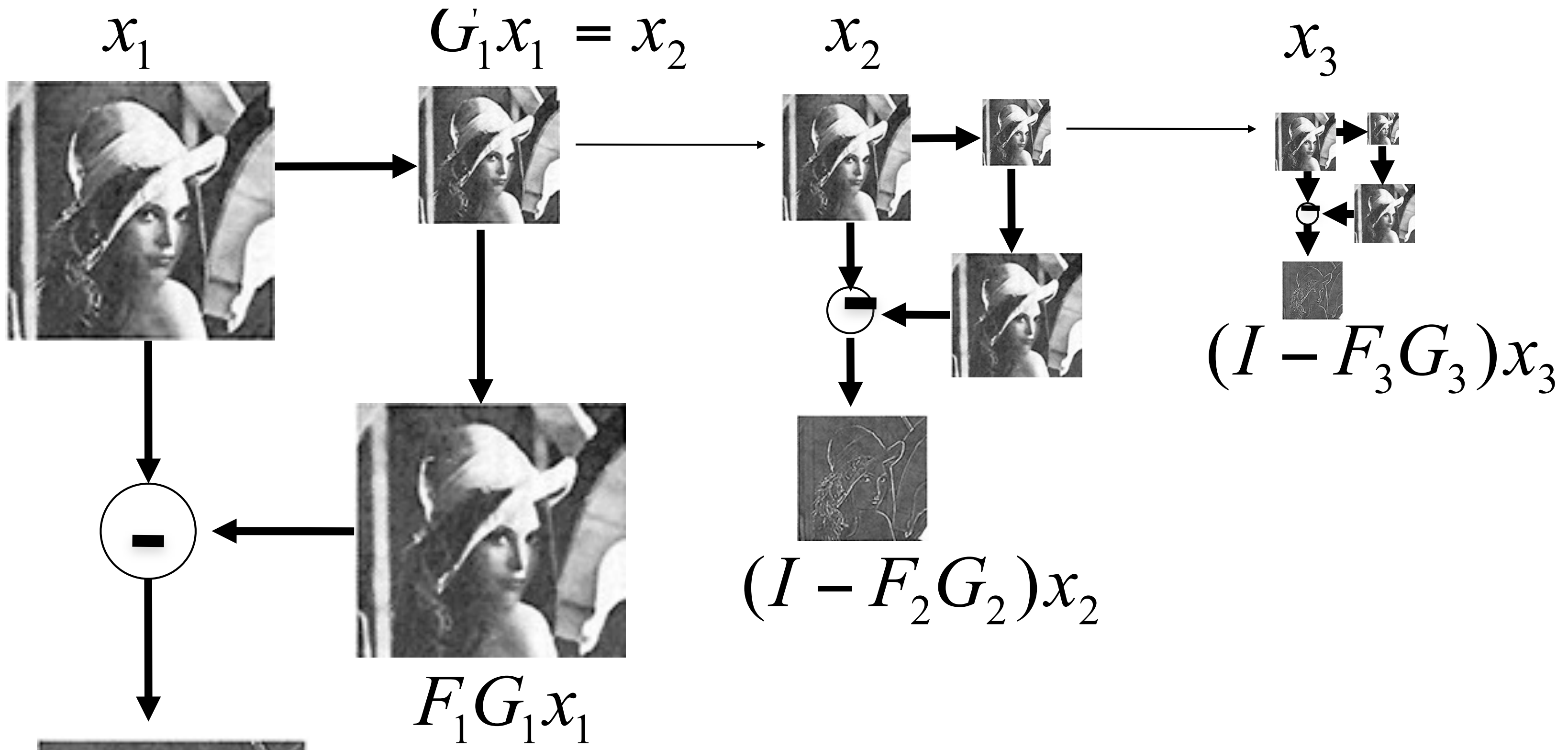

Slide credit: B. Freeman and A. Torralba

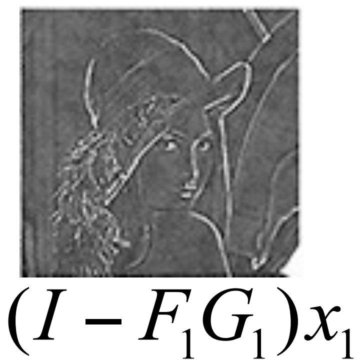
# The Laplacian Pyramid




Slide credit: B. Freeman and A. Torralba

# The Laplacian Pyramid




The coefficients $h_k$ at each level $k$ of the Laplacian pyramid $L(I)$ are constructed by taking the difference between adjacent levels in the Gaussian pyramid, upsampling the smaller one with $u(.)$ so that the sizes are compatible:

$$h_k = \mathcal{L}_k(I) = G_k(I) - u(G_{k+1}(I)) = I_k - u(I_{k+1})$$

Slide credit: B. Freeman and A. Torralba

# Upsampling

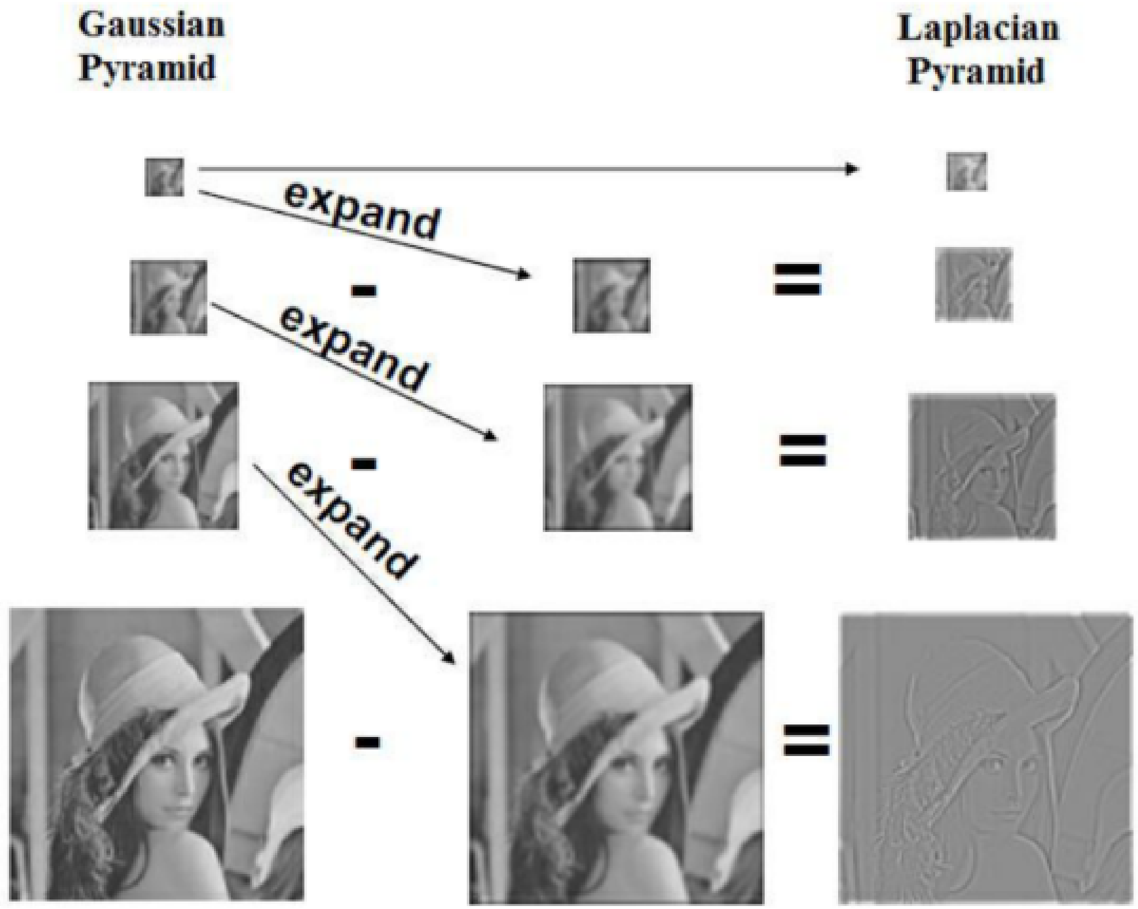
$$y_2 = F_3 x_3$$

Insert zeros between pixels, then apply a low-pass filter, [1 4 6 4 1]

$$F_3 = \begin{matrix} 6 & 1 & 0 & 0 \\ 4 & 4 & 0 & 0 \\ 1 & 6 & 1 & 0 \\ 0 & 4 & 4 & 0 \\ 0 & 1 & 6 & 1 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 1 & 6 \\ 0 & 0 & 0 & 4 \end{matrix}$$

Slide credit: B. Freeman and A. Torralba

# Showing, at full resolution, the information captured at each level of a Gaussian and Laplacian pyramid.




Slide credit: B. Freeman and A. Torralba

# Laplacian pyramid reconstruction algorithm: recover $x_1$ from $L_1$, $L_2$, $L_3$ and $x_4$

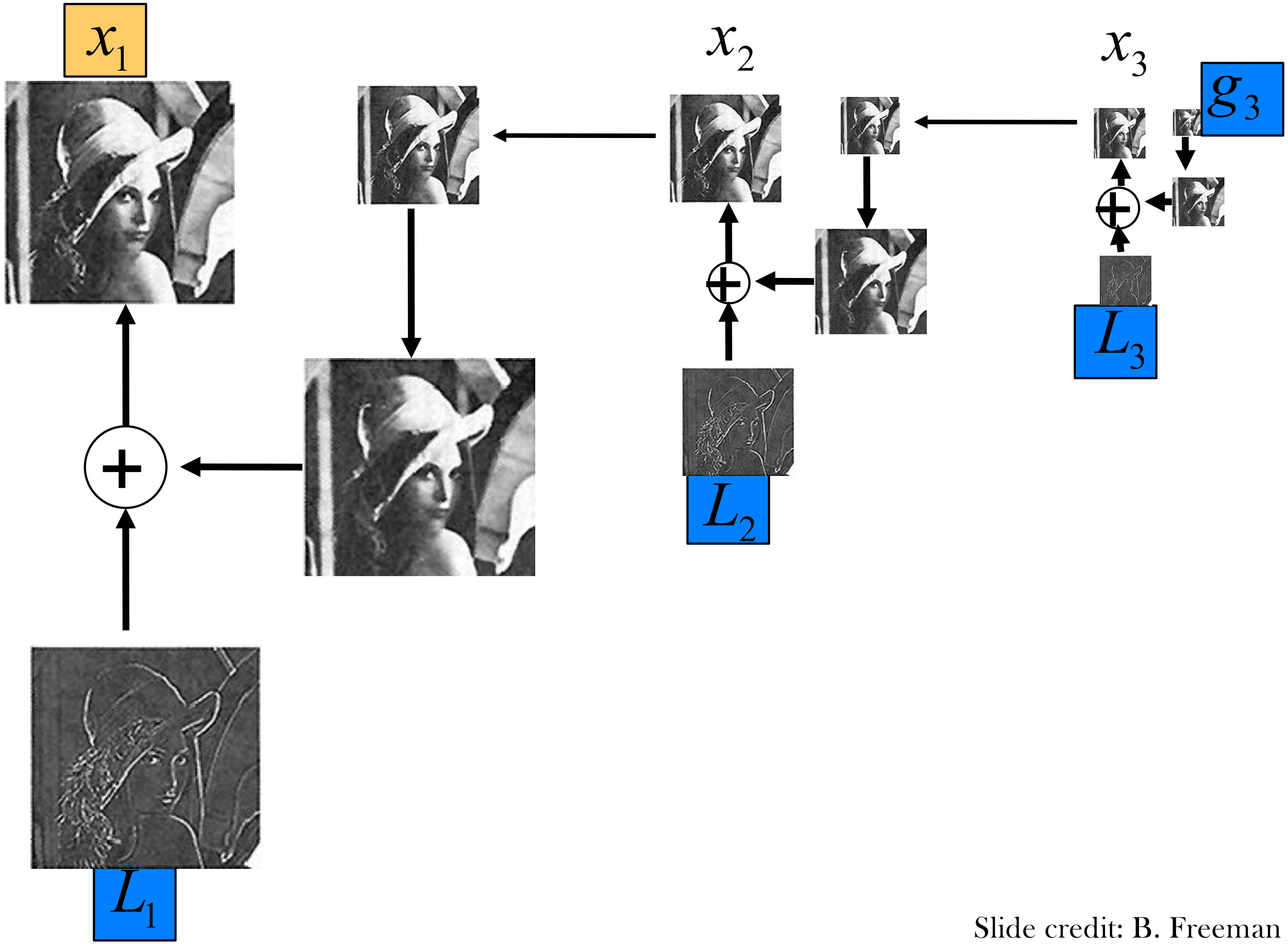
G# is the blur-and-downsample operator at pyramid level #
F# is the blur-and-upsample operator at pyramid level #

Laplacian pyramid elements:
L1 = (I – F1 G1) x1
L2 = (I – F2 G2) x2
L3 = (I – F3 G3) x3
x2 = G1 x1
x3 = G2 x2
x4 = G3 x3

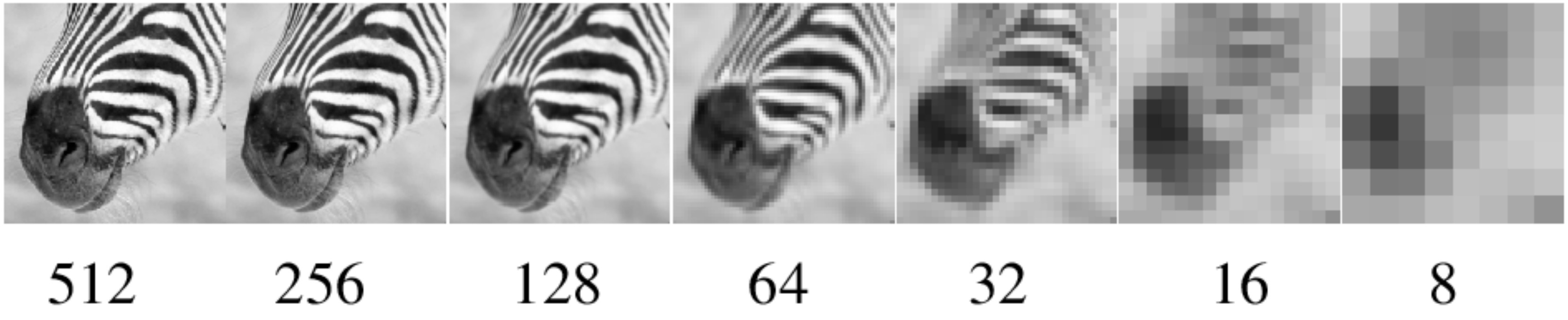
Reconstruction of original image (x1) from Laplacian pyramid elements:
x3 = L3 + F3 x4
x2 = L2 + F2 x3
x1 = L1 + F1 x2

Slide credit: B. Freeman and A. Torralba

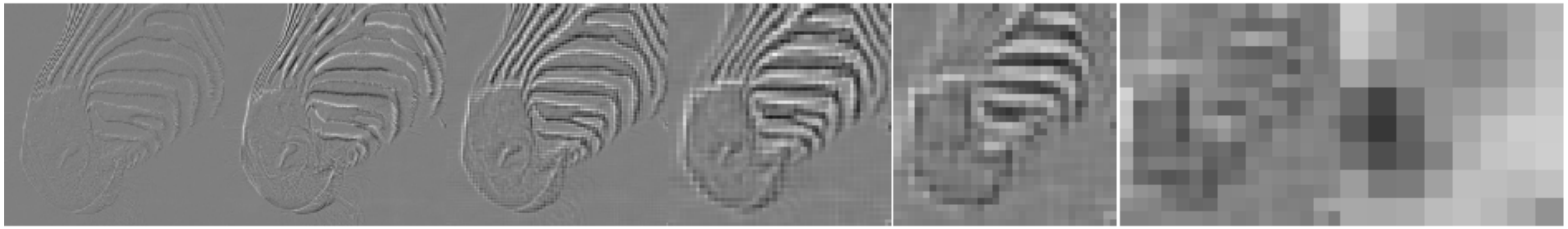
# Laplacian pyramid reconstruction algorithm: recover $x_1$ from $L_1$, $L_2$, $L_3$ and $g_3$



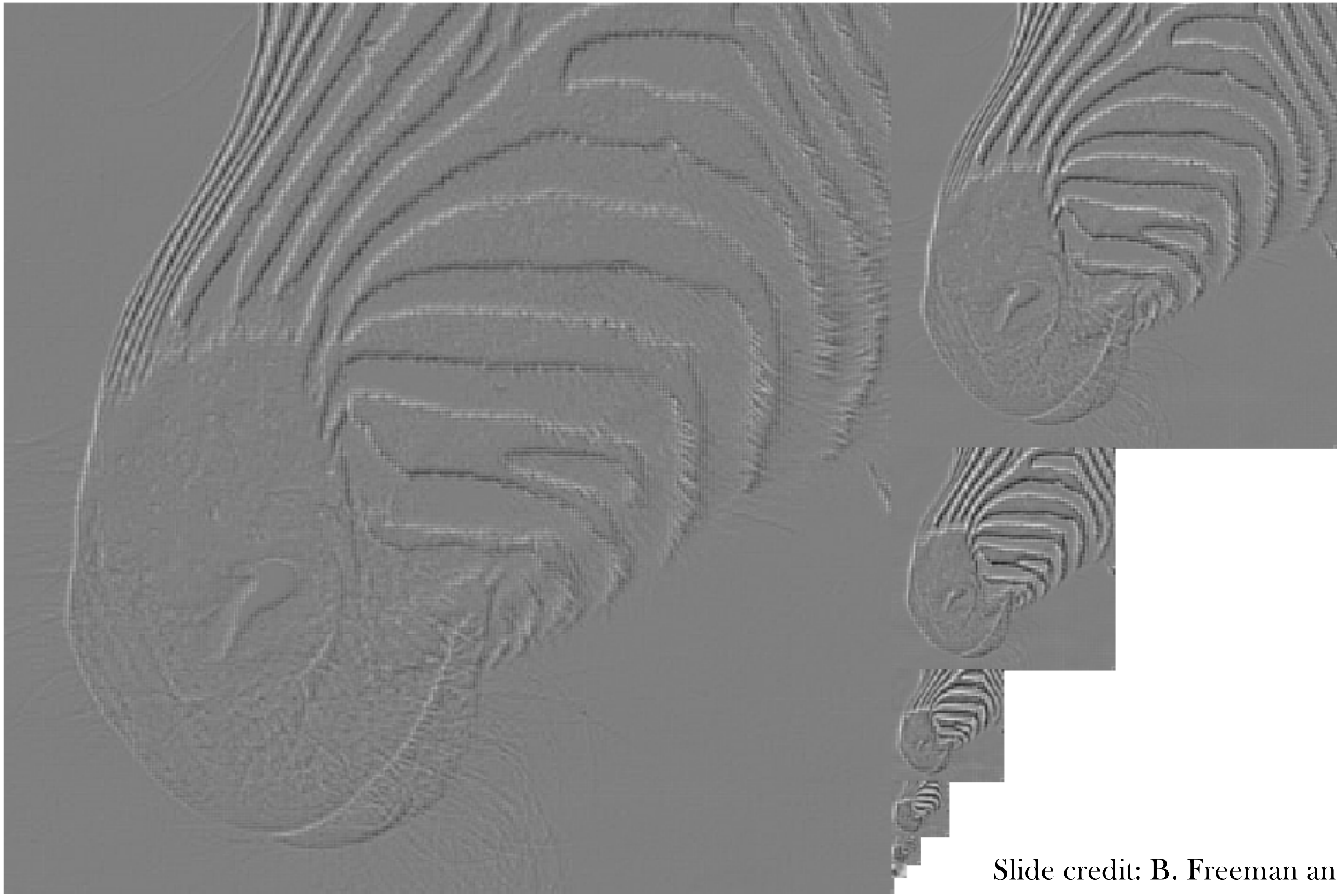

Slide credit: B. Freeman and A. Torralba




Slide credit: B. Freeman and A. Torralba



512 256 128 64 32 16 8



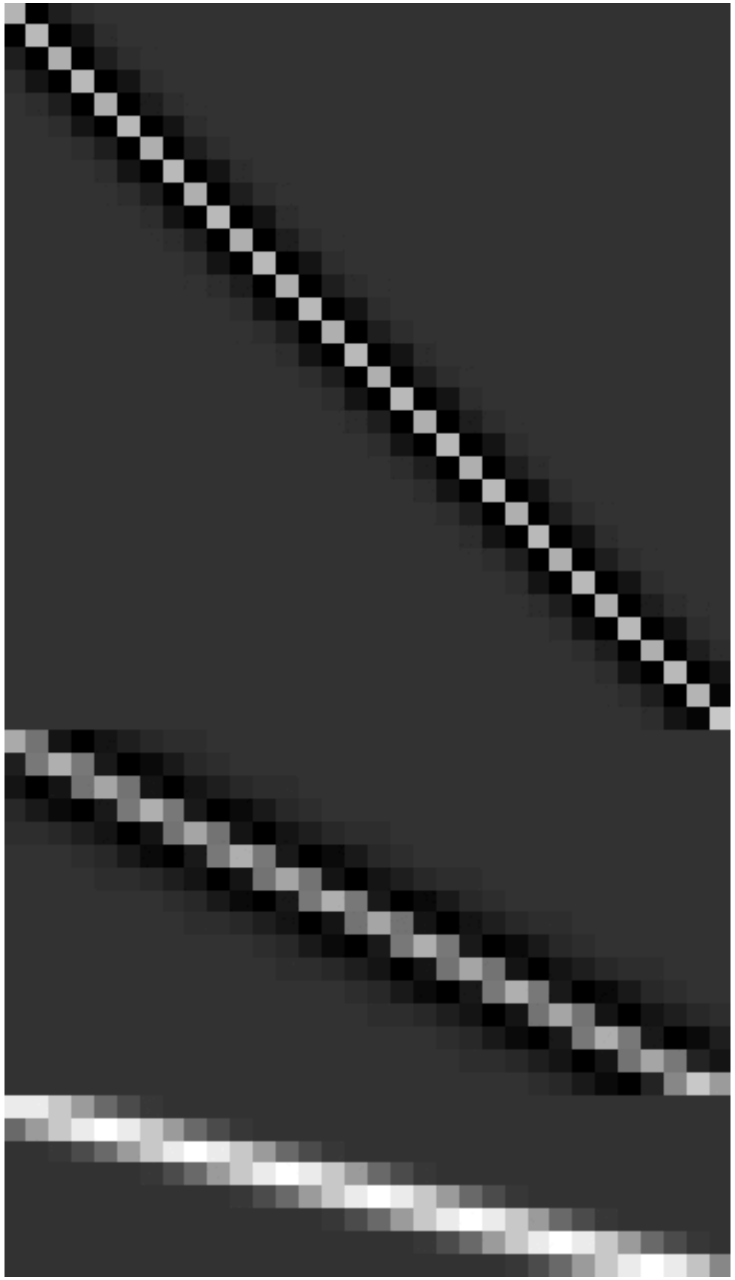
Slide credit: B. Freeman and A. Torralba

# 1D Laplacian pyramid matrix, for [1 4 6 4 1] low-pass filter

high frequencies

mid-band frequencies

low frequencies



Slide credit: B. Freeman and A. Torralba

# Laplacian pyramid applications

- Texture synthesis
- Image compression
- Noise removal

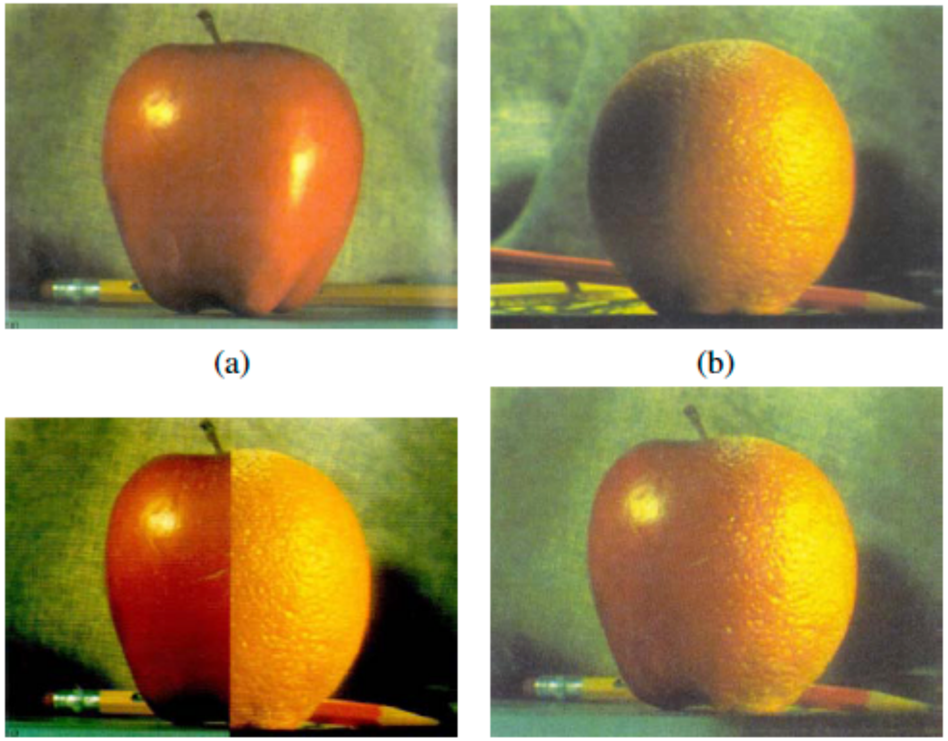
IEEE TRANSACTIONS ON COMMUNICATIONS, VOL. COM-31, NO. 4, APRIL 1983

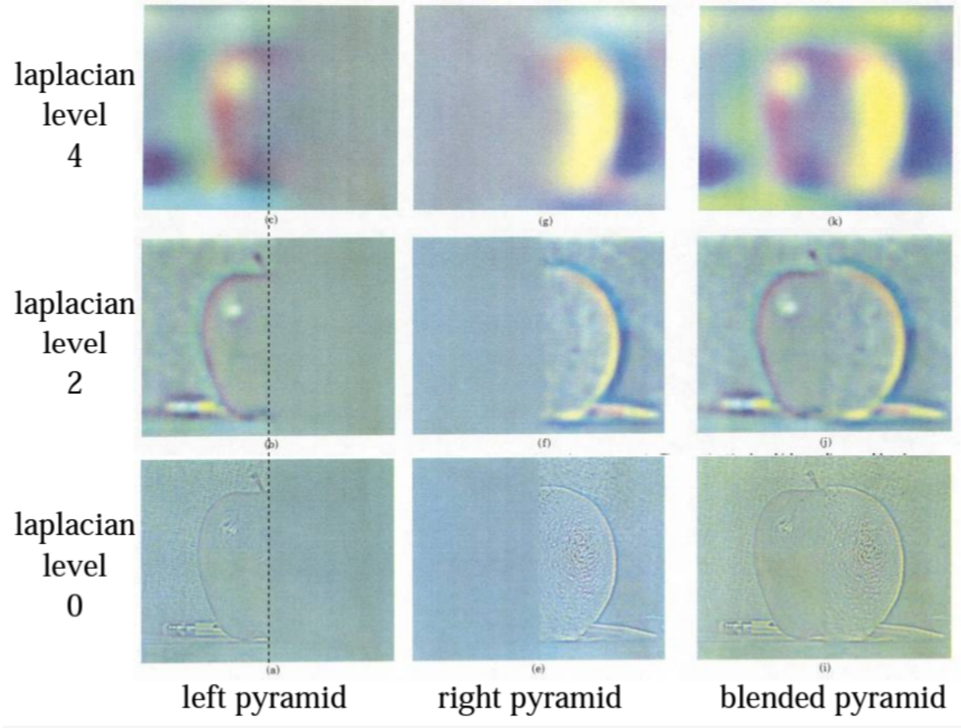
**The Laplacian Pyramid as a Compact Image Code**

**PETER J. BURT,** MEMBER, IEEE, AND **EDWARD H. ADELSON**

Slide credit: B. Freeman and A. Torralba

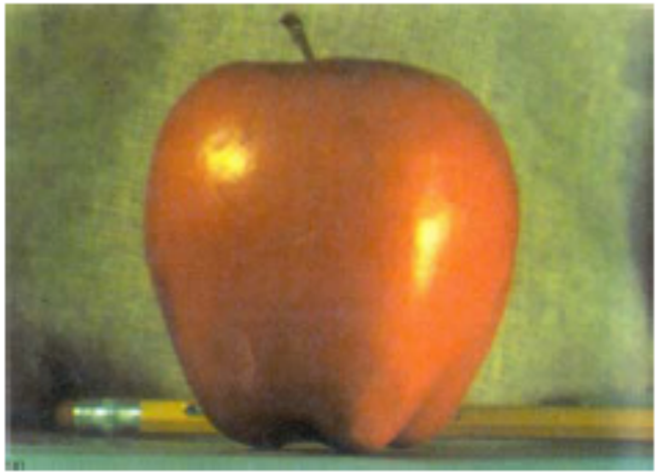
# Image blending



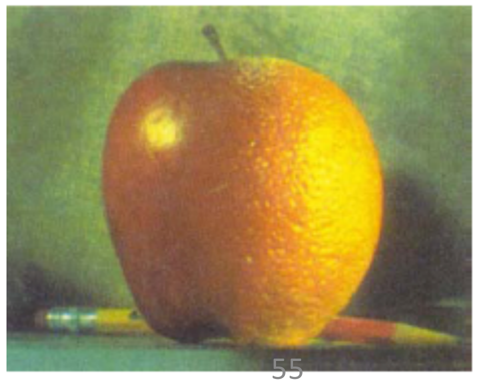




Slide credit: B. Freeman and A. Torralba

# Image blending




- General Approach
- Build Laplacian pyramid for both images: LA, LB
- Build Gaussian pyramid for mask: G
- Build a combined Laplacian pyramid:
  $L(j) = G(j)\ LA(j) + (1-G(j))\ LB(j)$
- Collapse L to obtain the blended image





Slide credit: B. Freeman and A. Torralba

# Image pyramids

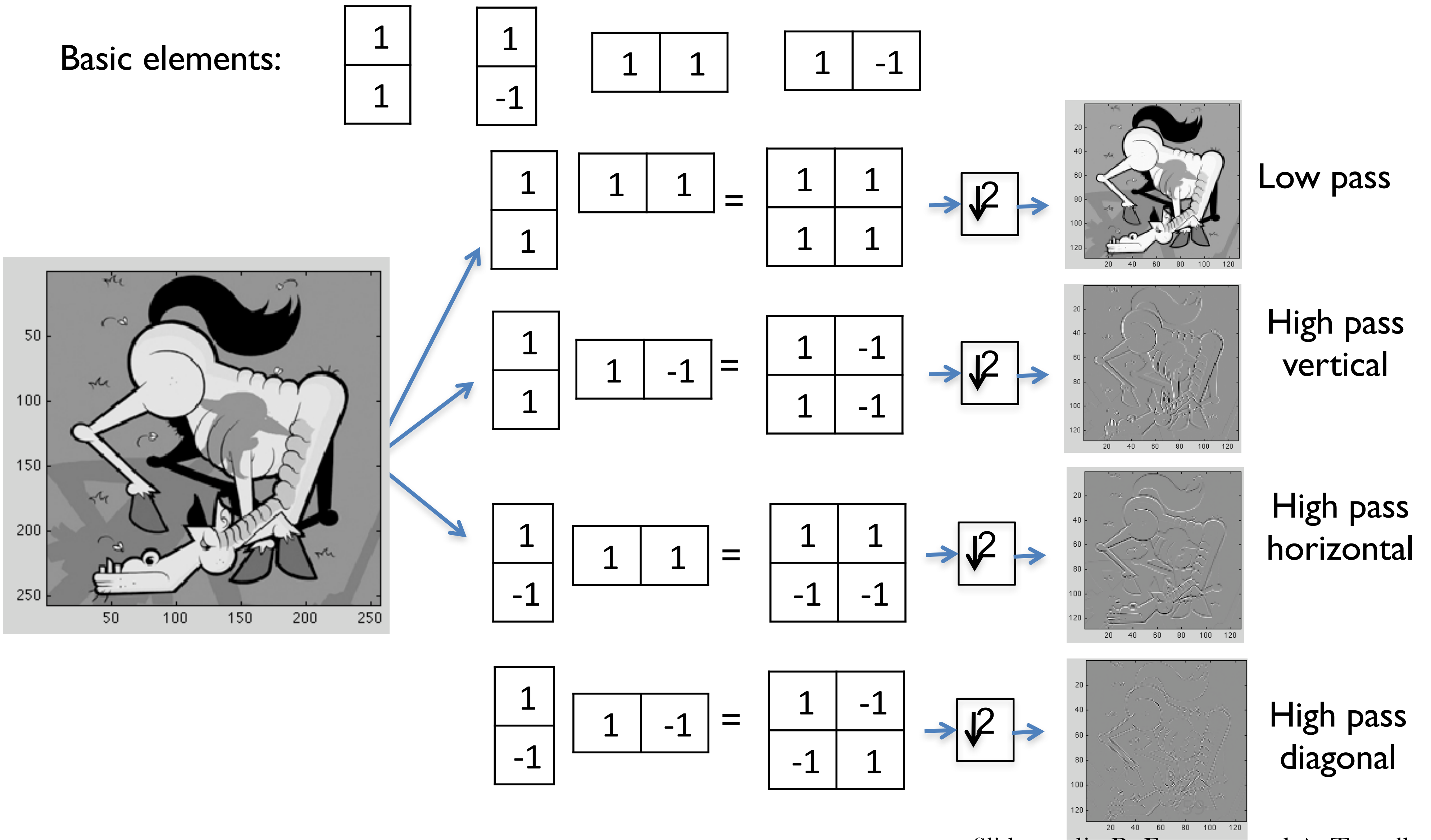
- Gaussian pyramid
- Laplacian pyramid
- Wavelet/QMF pyramid
- Steerable pyramid

Slide credit: B. Freeman and A. Torralba

# Wavelet/QMF pyramid

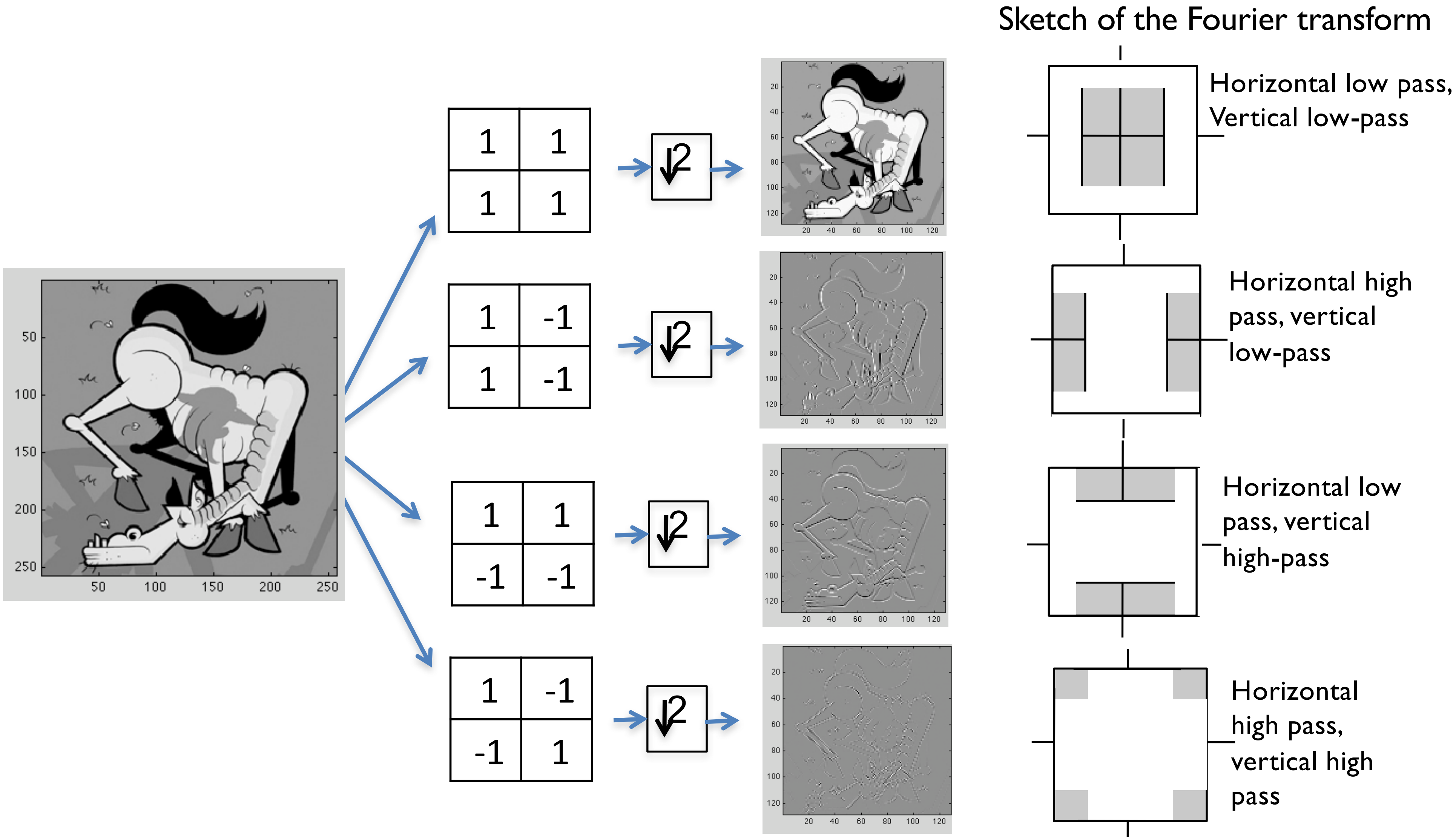
- Subband coding
- Wavelet or QMF (quadrature mirror filter) pyramid provides some splitting of the spatial frequency bands according to orientation (although in a somewhat limited way).
- Image is decomposed into a set of band-limited components (subbands).
- Original image can be reconstructed without error by reassemblying these subbands.

# 2D Haar transform



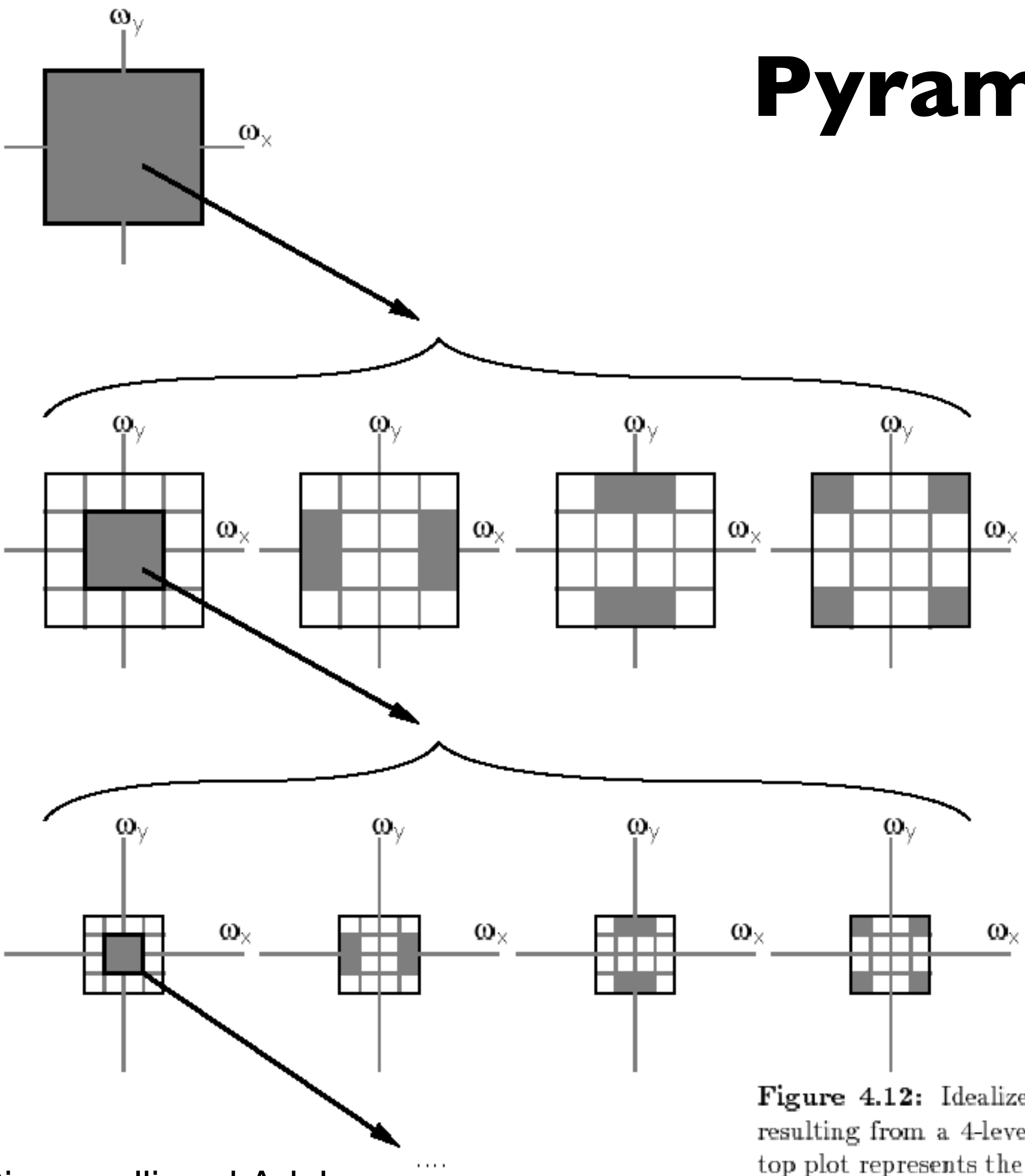

Slide credit: B. Freeman and A. Torralba

# 2D Haar transform




Slide credit: B. Freeman and A. Torralba

# Pyramid cascade

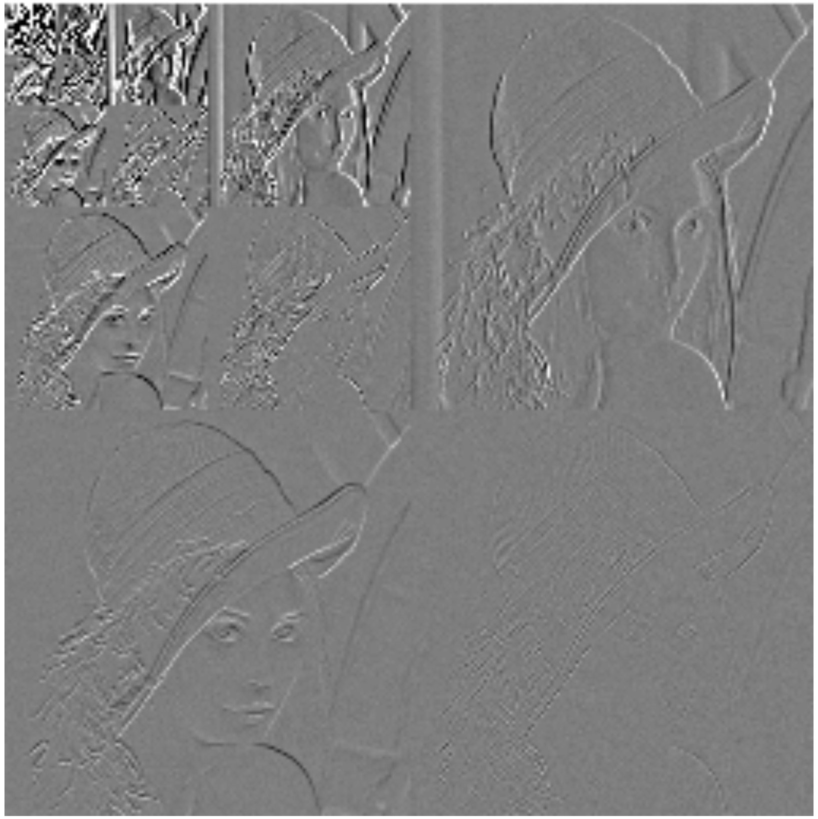



Simoncelli and Adelson,
in "Subband coding", Kluwer, 1990.

**Figure 4.12:** Idealized diagram of the partition of the frequency plane resulting from a 4-level pyramid cascade of separable 2-band filters. The top plot represents the frequency spectrum of the original image, with axes ranging from $-\pi$ to $\pi$. This is divided into four subbands at the next level. On each subsequent level, the lowpass subband (outlined in bold) is subdivided further.

Slide credit: B. Freeman and A. Torralba

# Wavelet/QMF representation



| | |
|---|---|
| 1 | -1 |
| 1 | -1 |

| | |
|---|---|
| 1 | 1 |
| -1 | -1 |

| | |
|---|---|
| 1 | -1 |
| -1 | 1 |

Same number of pixels!

Slide credit: B. Freeman and A. Torralba

# Image pyramids

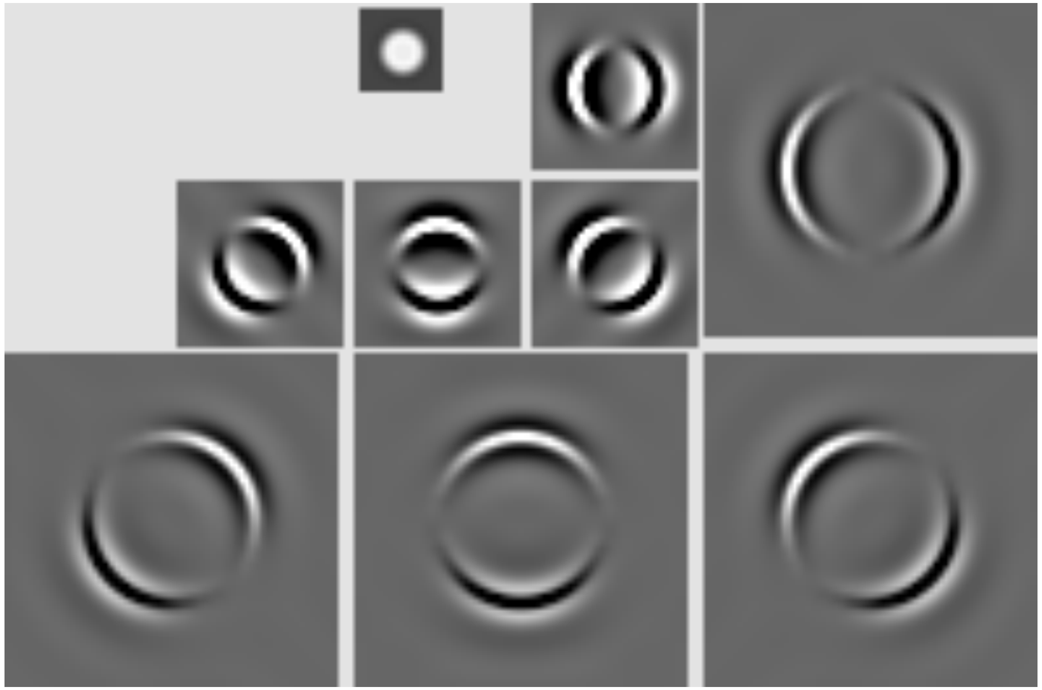
- Gaussian pyramid
- Laplacian pyramid
- Wavelet/QMF pyramid
- Steerable pyramid

Slide credit: B. Freeman and A. Torralba

# Steerable Pyramid

Low pass
residual

2 Level decomposition
of white circle example:



Subbands

- The Steerable pyramid provides a clean separation of the image into different scales and orientations.

Images from: http://www.cis.upenn.edu/~eero/steerpyr.html

Slide credit: B. Freeman and A. Torralba

# Steerable Pyramid

We may combine Steerability with Pyramids to get a Steerable Laplacian Pyramid as shown below.

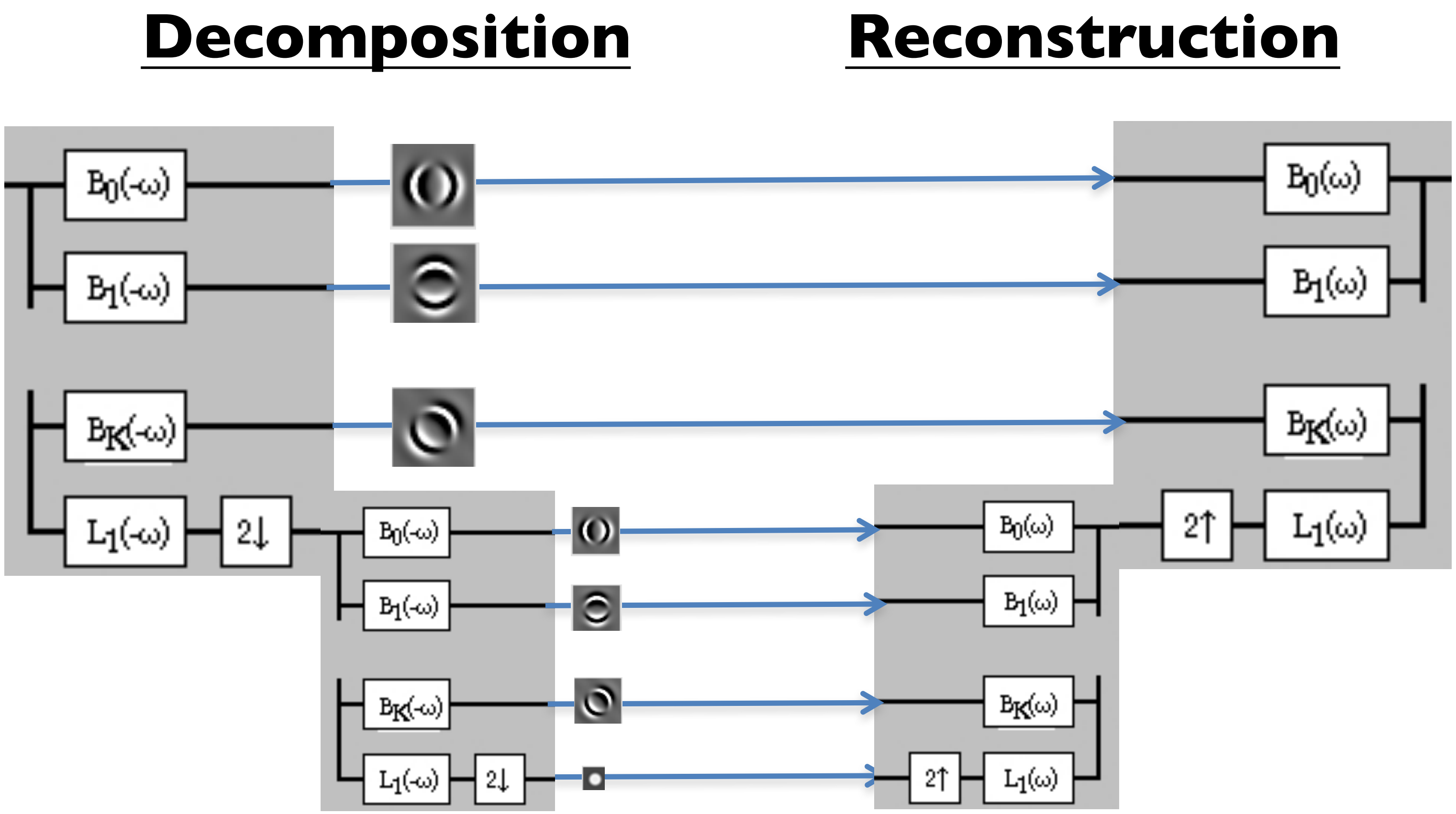
**Decomposition** **Reconstruction**


Images from: http://www.cis.upenn.edu/~eero/steerpyr.html

Slide credit: B. Freeman and A. Torralba

# Steerable Pyramid

We may combine Steerability with Pyramids to get a Steerable Laplacian Pyramid as shown below




Images from: http://www.cis.upenn.edu/~eero/steerpyr.html

Slide credit: B. Freeman and A. Torralba

# Steerable Pyramid

But we need to get rid of the corner regions before starting the recursive circular filtering

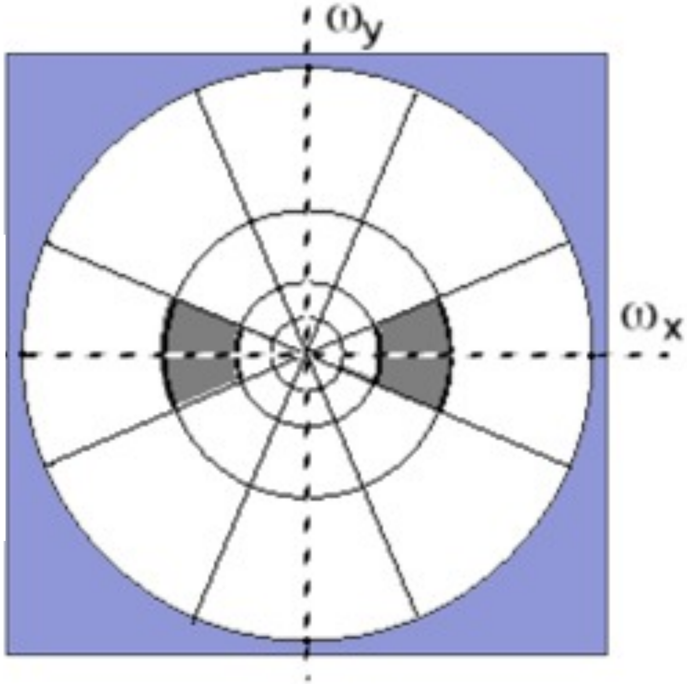
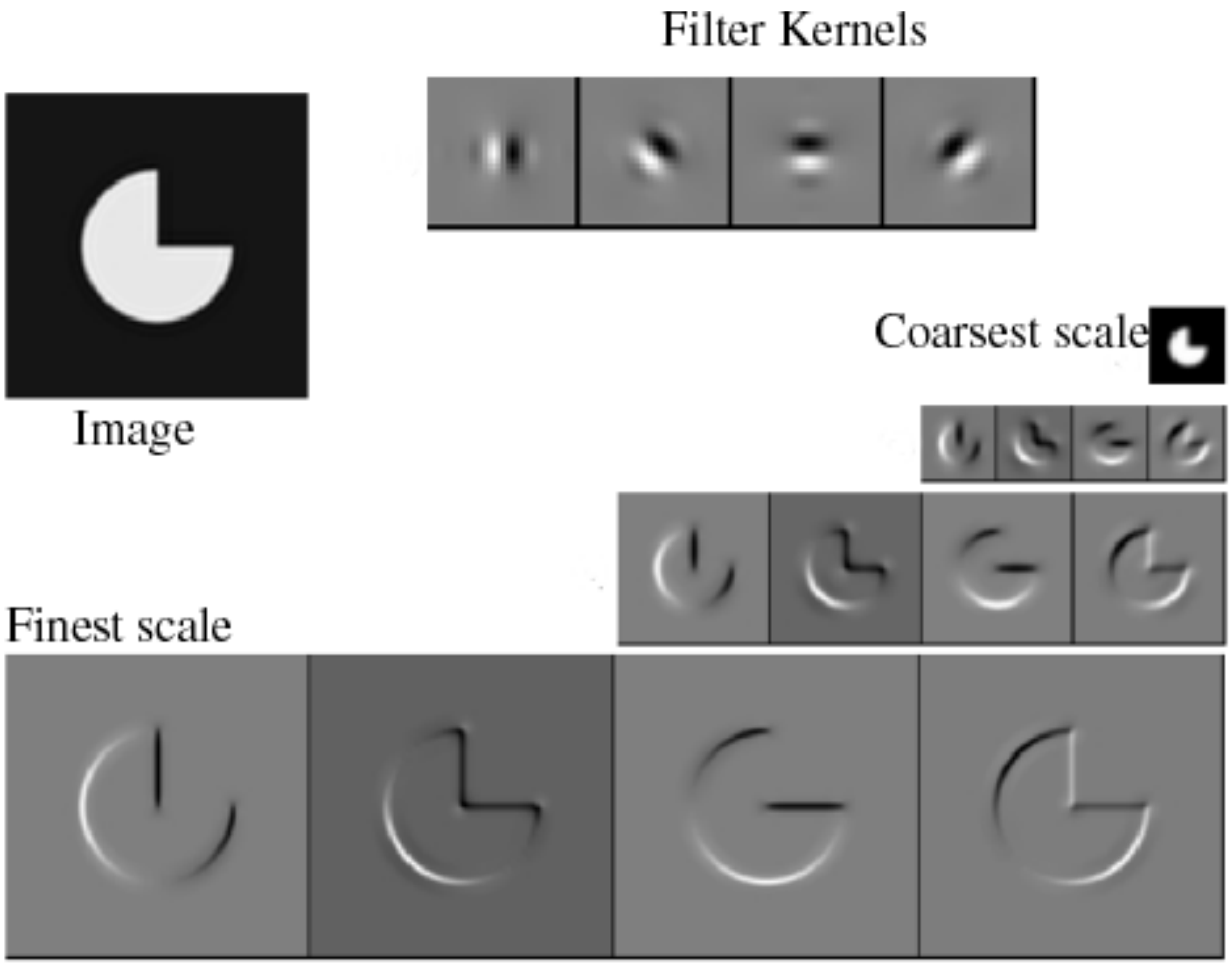



**Figure 1.** Idealized illustration of the spectral decomposition performed by a steerable pyramid with $k = 4$. Frequency axes range from $-\pi$ to $\pi$. The basis functions are related by translations, dilations and *rotations* (except for the initial highpass subband and the final lowpass subband). For example, the shaded region corresponds to the spectral support of a single (vertically-oriented) subband.

Simoncelli and Freeman, ICIP 1995

Slide credit: B. Freeman and A. Torralba




Reprinted from "Shiftable MultiScale Transforms," by Simoncelli et al., IEEE Transactions on Information Theory, 1992, copyright 1992, IEEE

There is also a high pass residual…

Slide credit: B. Freeman and A. Torralba

# Phase-based Video Magnification

https://www.youtube.com/watch?v=W7ZQ-FG7Nvw

# Image pyramids

- Gaussian
- Laplacian
- Wavelet/QMF
- Steerable pyramid

Slide credit: B. Freeman and A. Torralba

# Image pyramids

- Gaussian

Progressively blurred and subsampled versions of the image. Adds scale invariance to fixed-size algorithms.

- Laplacian

- Wavelet/QMF

- Steerable pyramid

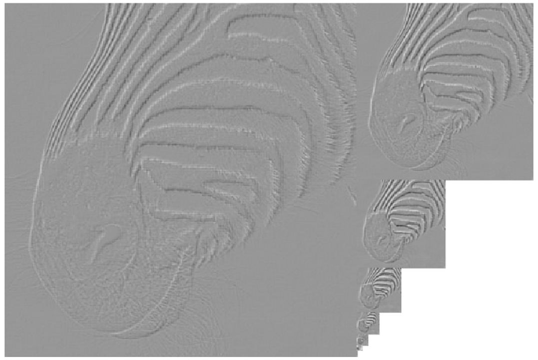
Slide credit: B. Freeman and A. Torralba

# Image pyramids

- Gaussian

Progressively blurred and subsampled versions of the image. Adds scale invariance to fixed-size algorithms.

- Laplacian



Shows the information added in Gaussian pyramid at each spatial scale. Useful for noise reduction & coding.

- Wavelet/QMF

- Steerable pyramid

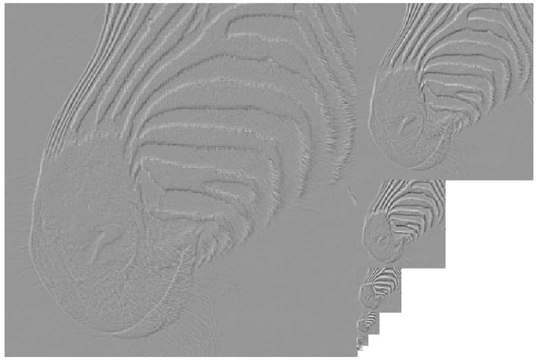
Slide credit: B. Freeman and A. Torralba

# Image pyramids

- Gaussian

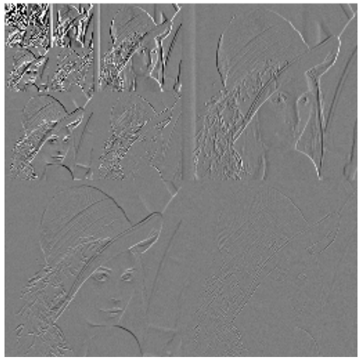
Progressively blurred and subsampled versions of the image. Adds scale invariance to fixed-size algorithms.

- Laplacian



Shows the information added in Gaussian pyramid at each spatial scale. Useful for noise reduction & coding.

- Wavelet/QMF



Bandpassed representation, complete, but with aliasing and some non-oriented subbands.

- Steerable pyramid

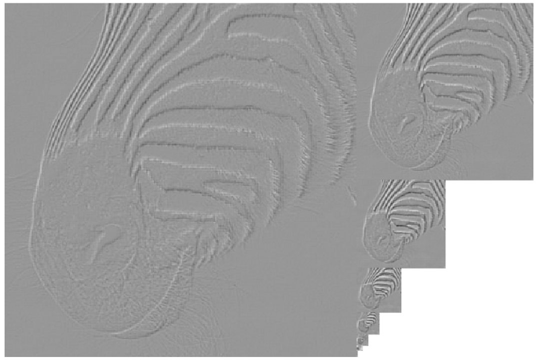
Slide credit: B. Freeman and A. Torralba

# Image pyramids

- Gaussian

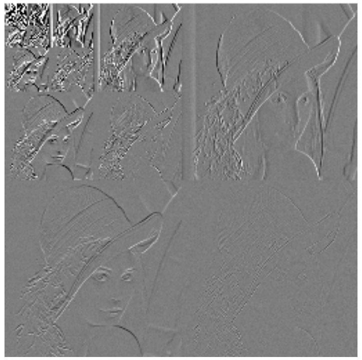
Progressively blurred and subsampled versions of the image. Adds scale invariance to fixed-size algorithms.

- Laplacian



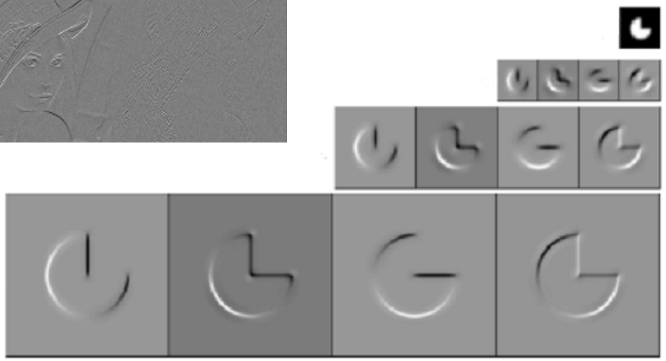
Shows the information added in Gaussian pyramid at each spatial scale. Useful for noise reduction & coding.

- Wavelet/QMF



Bandpassed representation, complete, but with aliasing and some non-oriented subbands.

- Steerable pyramid



Shows components at each scale and orientation separately. Non-aliased subbands. Good for texture and feature analysis. But overcomplete and with HF residual.

Slide credit: B. Freeman and A. Torralba

# Schematic pictures of each matrix transform

Shown for 1-d images

The matrices for 2-d images are the same idea, but more complicated, to account for vertical, as well as horizontal, neighbor relationships.

transformed image

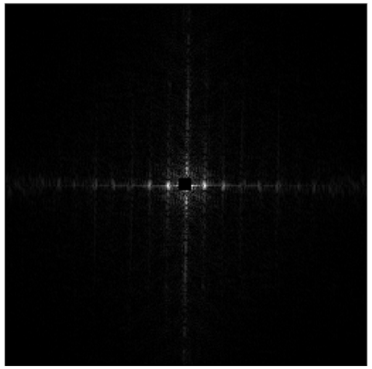
$$\vec{F} = U\vec{f}$$

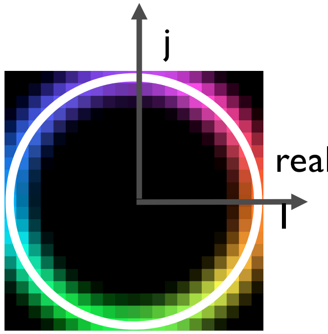
Vectorized image

Fourier transform, or
Wavelet transform, or
Steerable pyramid transform

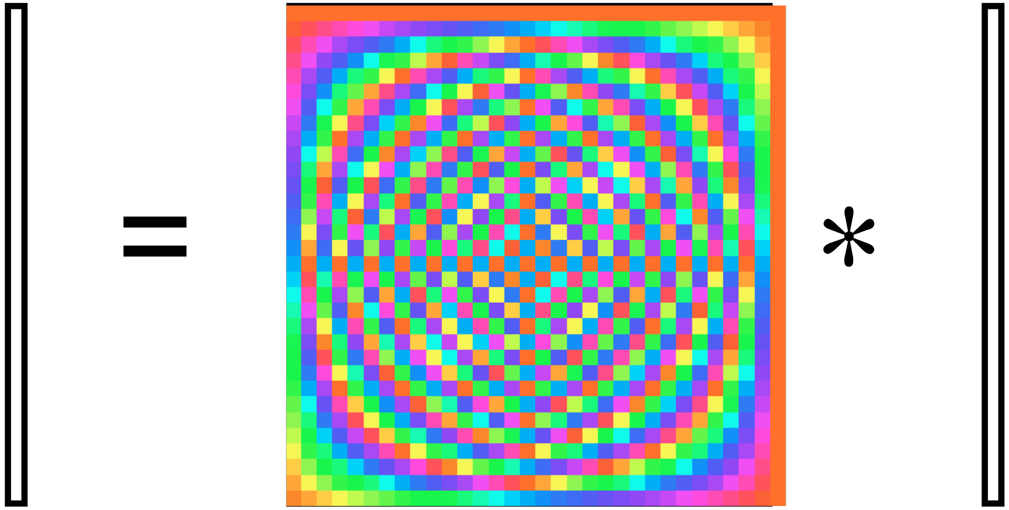
Slide credit: B. Freeman and A. Torralba

# Fourier transform






color key



Fourier transform

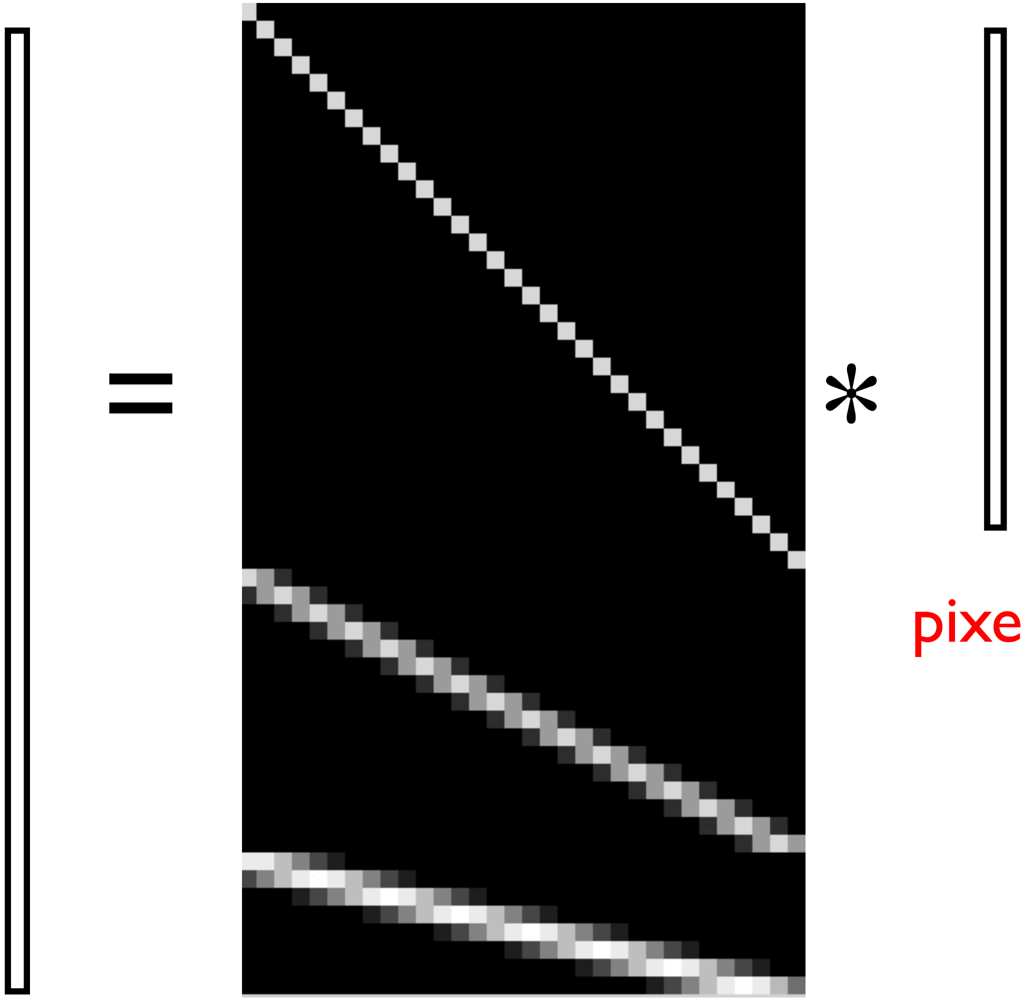
Fourier bases are global: each transform coefficient depends on all pixel locations.

pixel domain image

Slide credit: B. Freeman and A. Torralba

# Gaussian pyramid

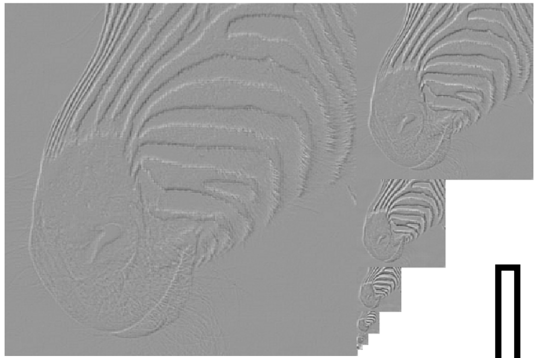
Gaussian pyramid



pixel image

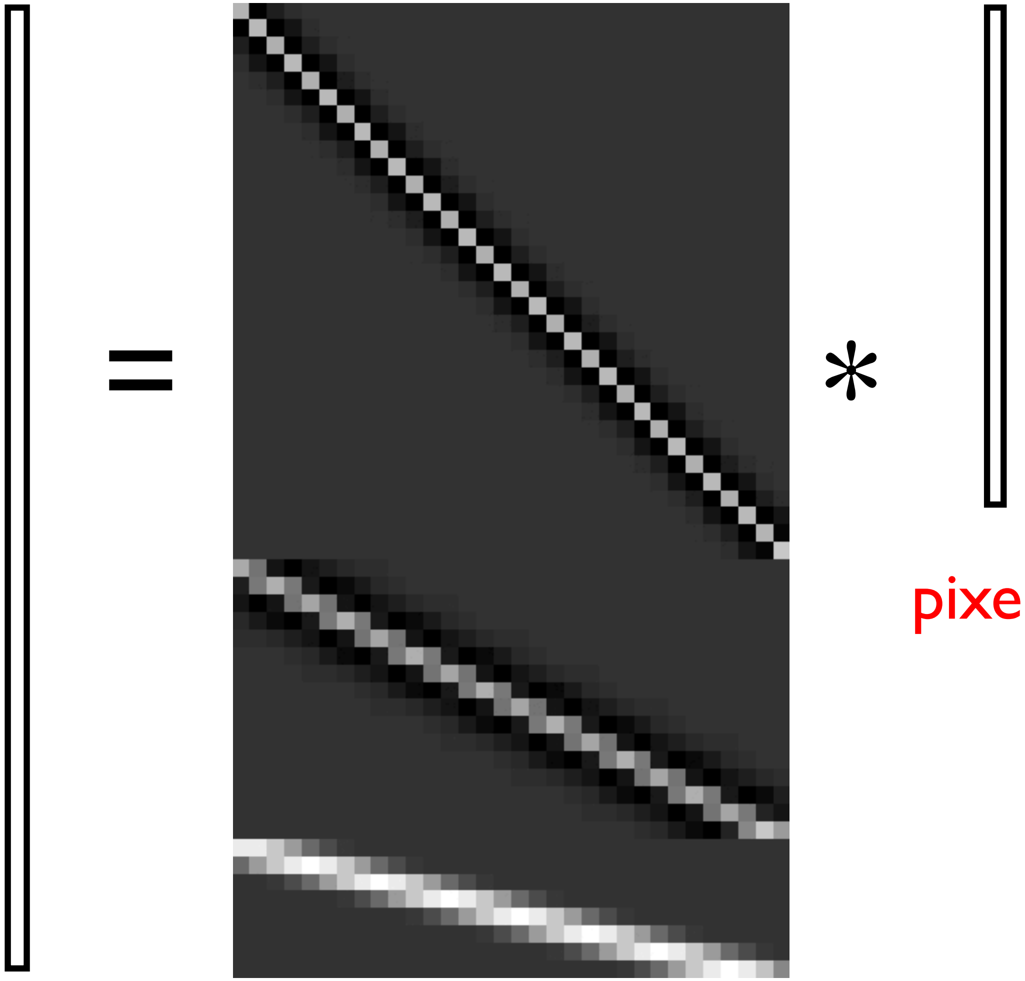
Overcomplete representation.
Low-pass filters, sampled appropriately for their blur.

Slide credit: B. Freeman and A. Torralba



# Laplacian pyramid

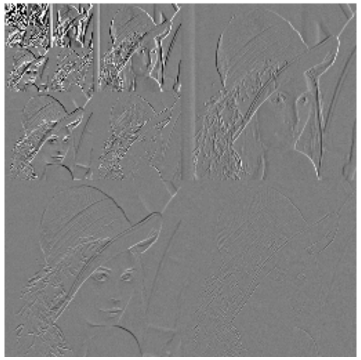
Laplacian
pyramid



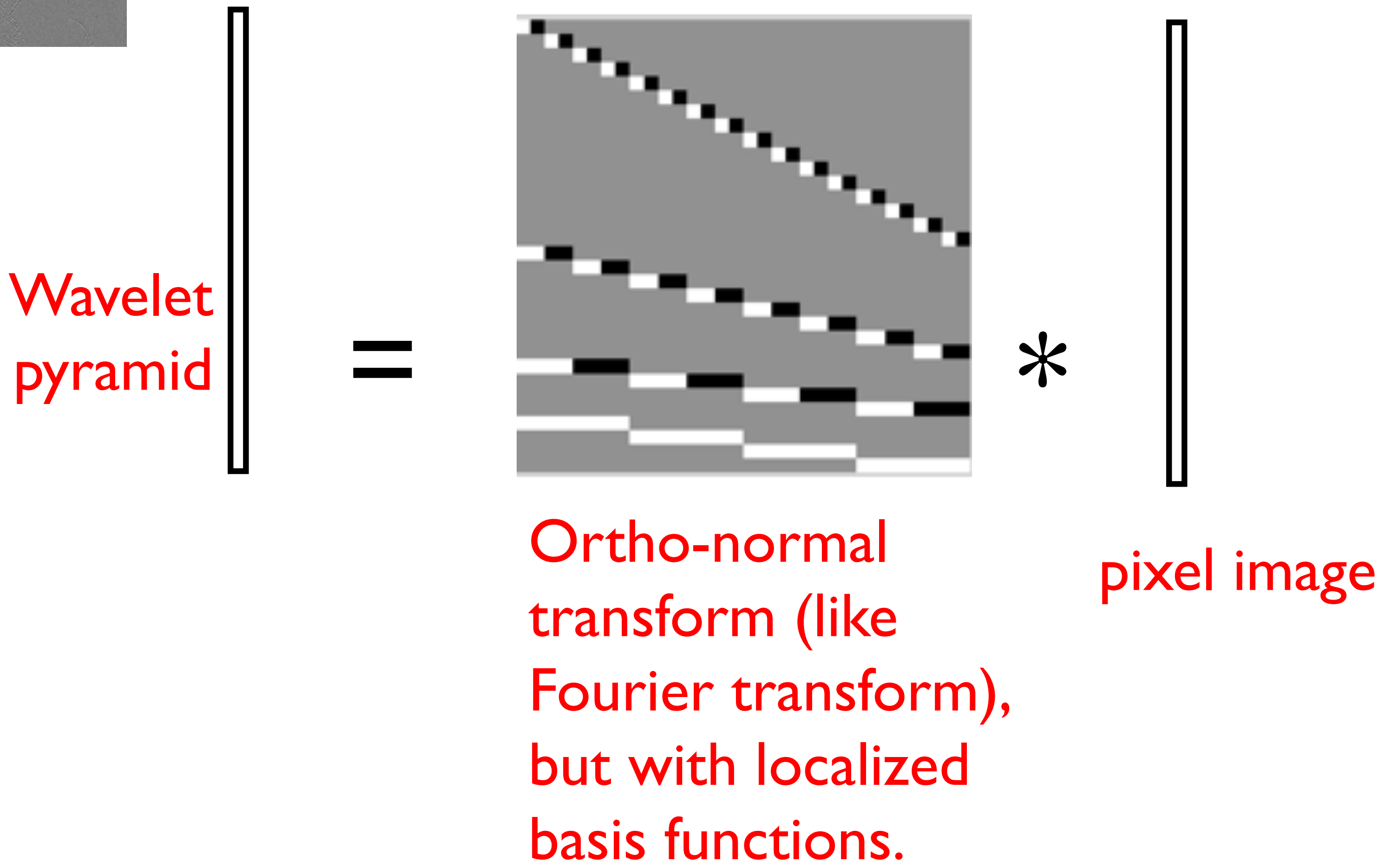
pixel image

Overcomplete representation.
Transformed pixels represent
bandpassed image information.

Slide credit: B. Freeman and A. Torralba

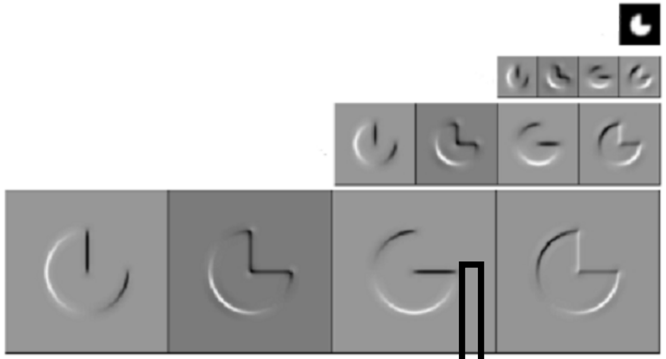


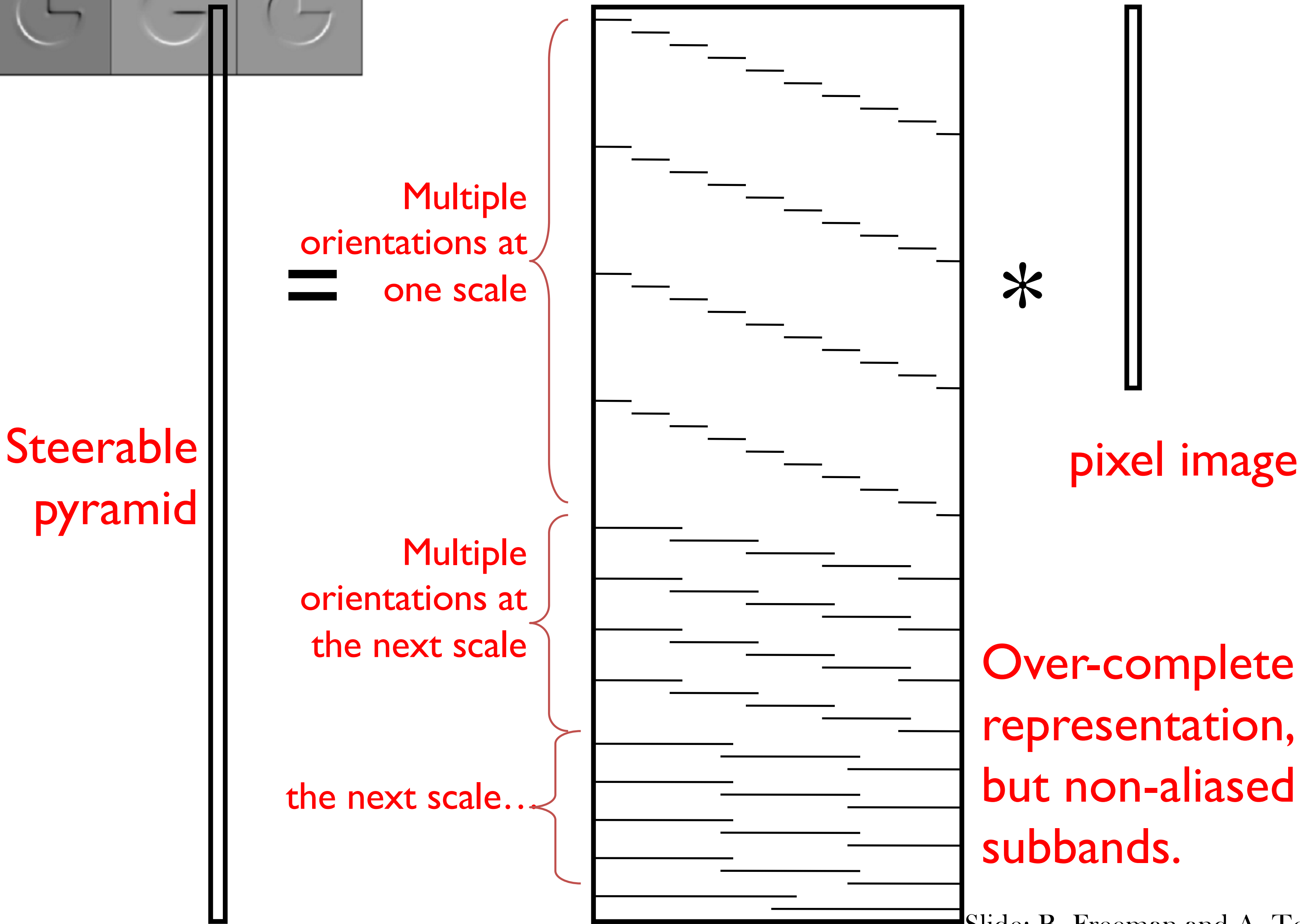
# Wavelet (QMF) transform




Slide credit: B. Freeman and A. Torralba

# Steerable pyramid






Slide: B. Freeman and A. Torralba

# Why use image pyramids?

- Handle real-world size variations with a constant-size vision algorithm.
- Remove noise
- Analyze texture
- Recognize objects
- Label image features
- Image priors can be specified naturally in terms of wavelet pyramids.

Slide credit: B. Freeman and A. Torralba

# Reading Assignment #2 – Hybrid Images

- A. Oliva, A. Torralba, P.G. Schyns (2006). Hybrid Images. ACM Transactions on Graphics, ACM SIGGRAPH, 25-3, 527-530.
- Due on 1 of April