Lecture -14
Multi resolution & Wavelets

Image Scaling

This image is too big to fit on the screen. How can we generate a half-sized version?

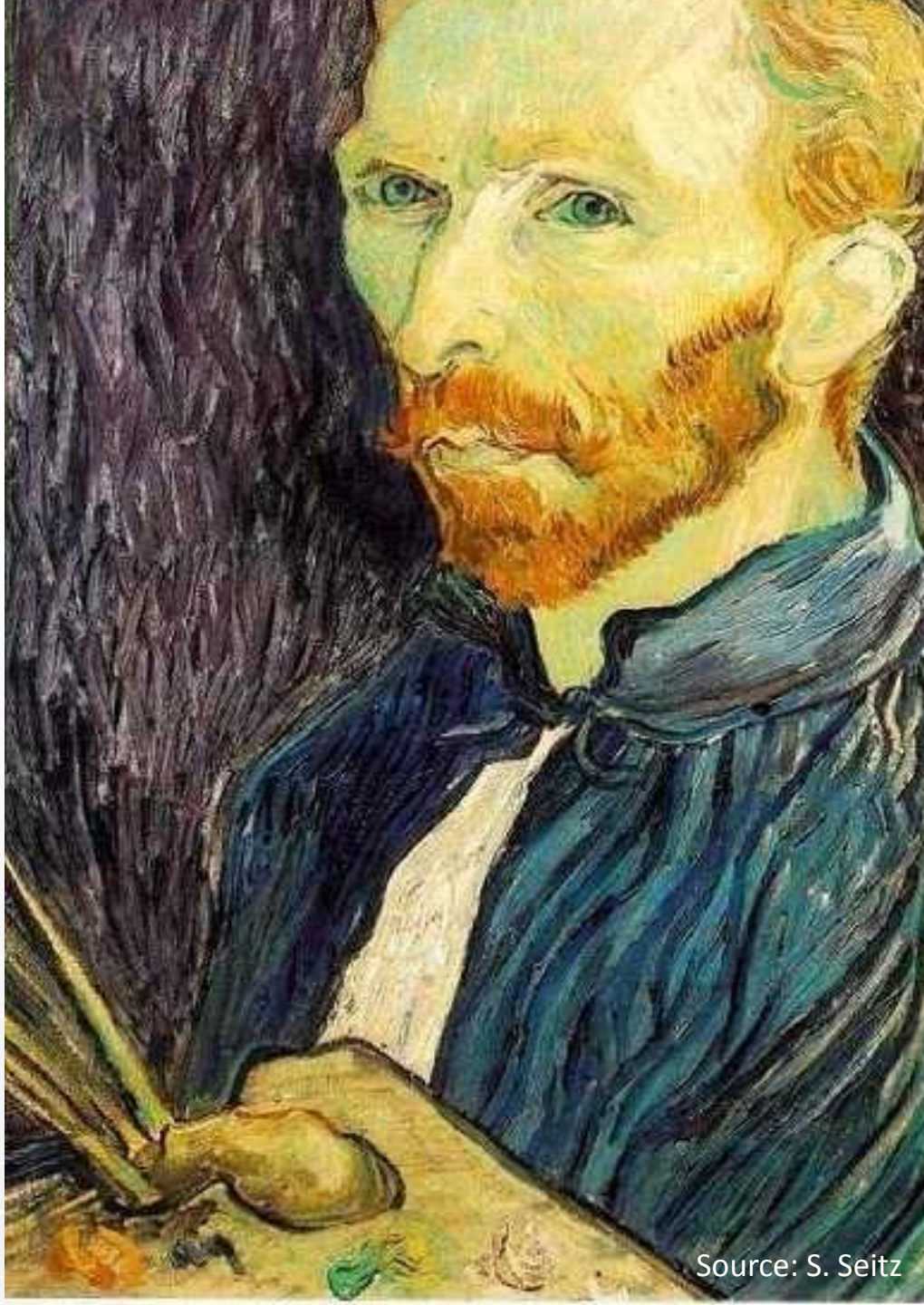
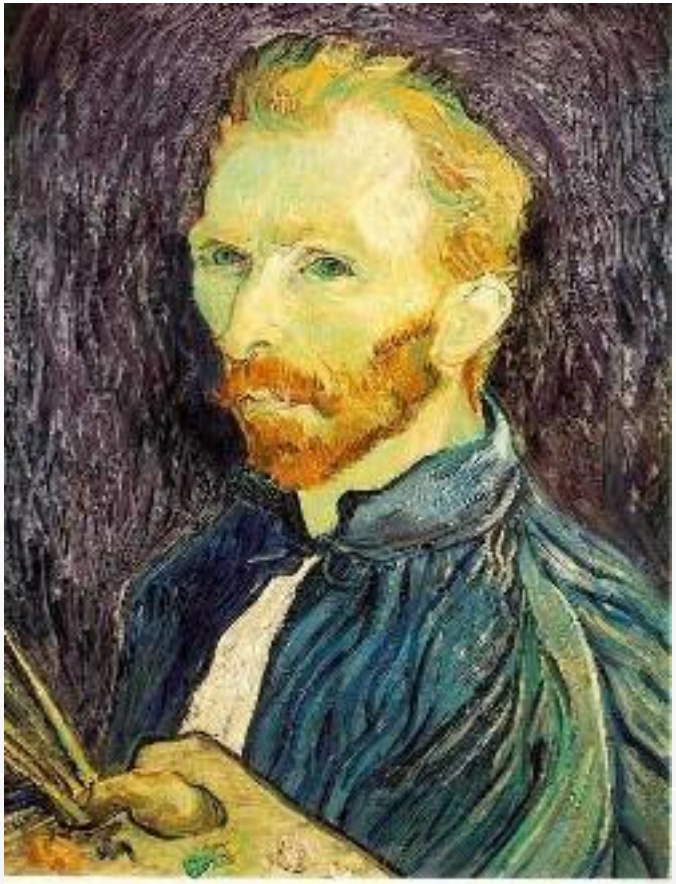


Image sub-sampling



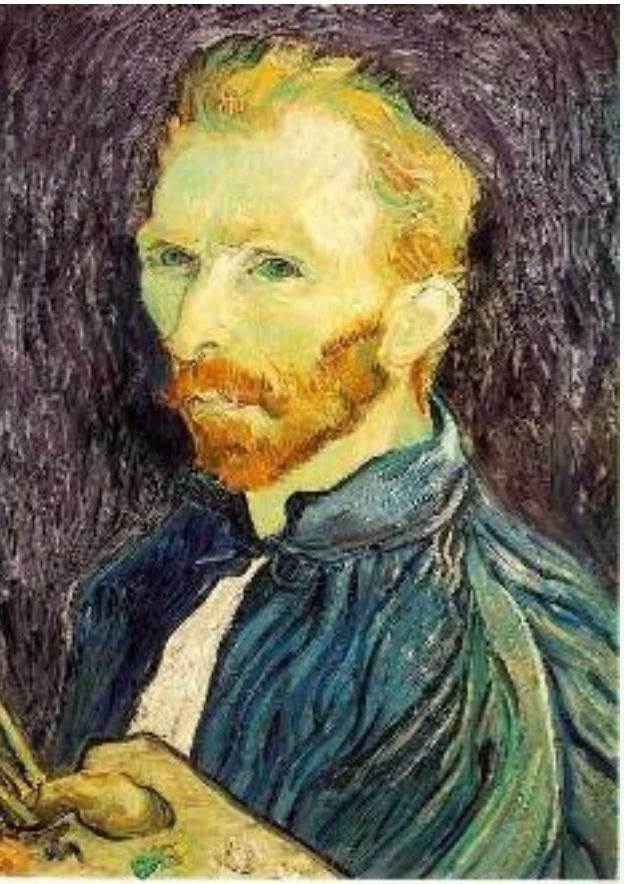
1/4

1/8

Throw away every other row and column to create a 1/2 size image
- called *image sub-sampling*

Source: S. Seitz

Image sub-sampling



1/2

1/4 (2x zoom)

1/8 (4x zoom)

Why does this look so crufty?

Source: S. Seitz

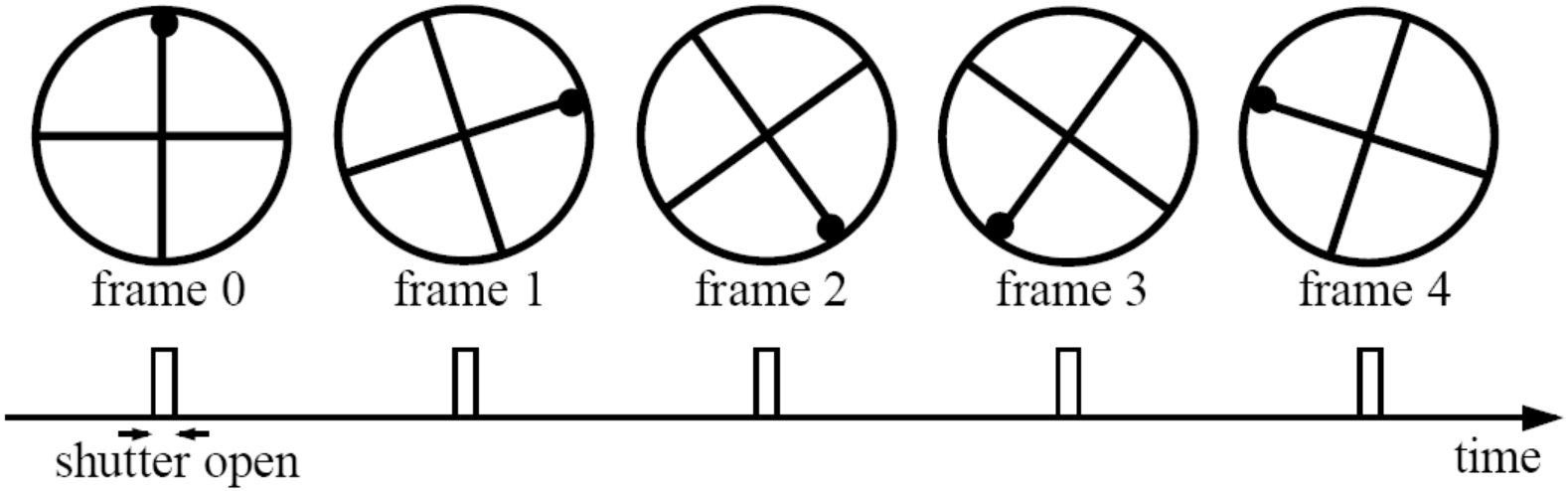
Image sub-sampling

Source: F. Durand

Wagon-wheel effect

Imagine a spoked wheel moving to the right (rotating clockwise). Mark wheel with dot so we can see what's happening.

If camera shutter is only open for a fraction of a frame time (frame time = 1/30 sec. for video, 1/24 sec. for film):

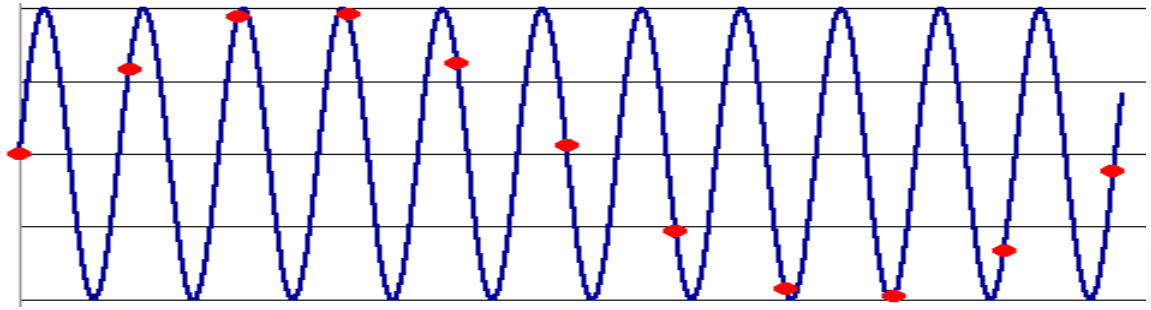


Without dot, wheel appears to be rotating slowly backwards! (counterclockwise)

(See http://www.michaelbach.de/ot/mot_wagonWheel/index.html)

Source: L. Zhang

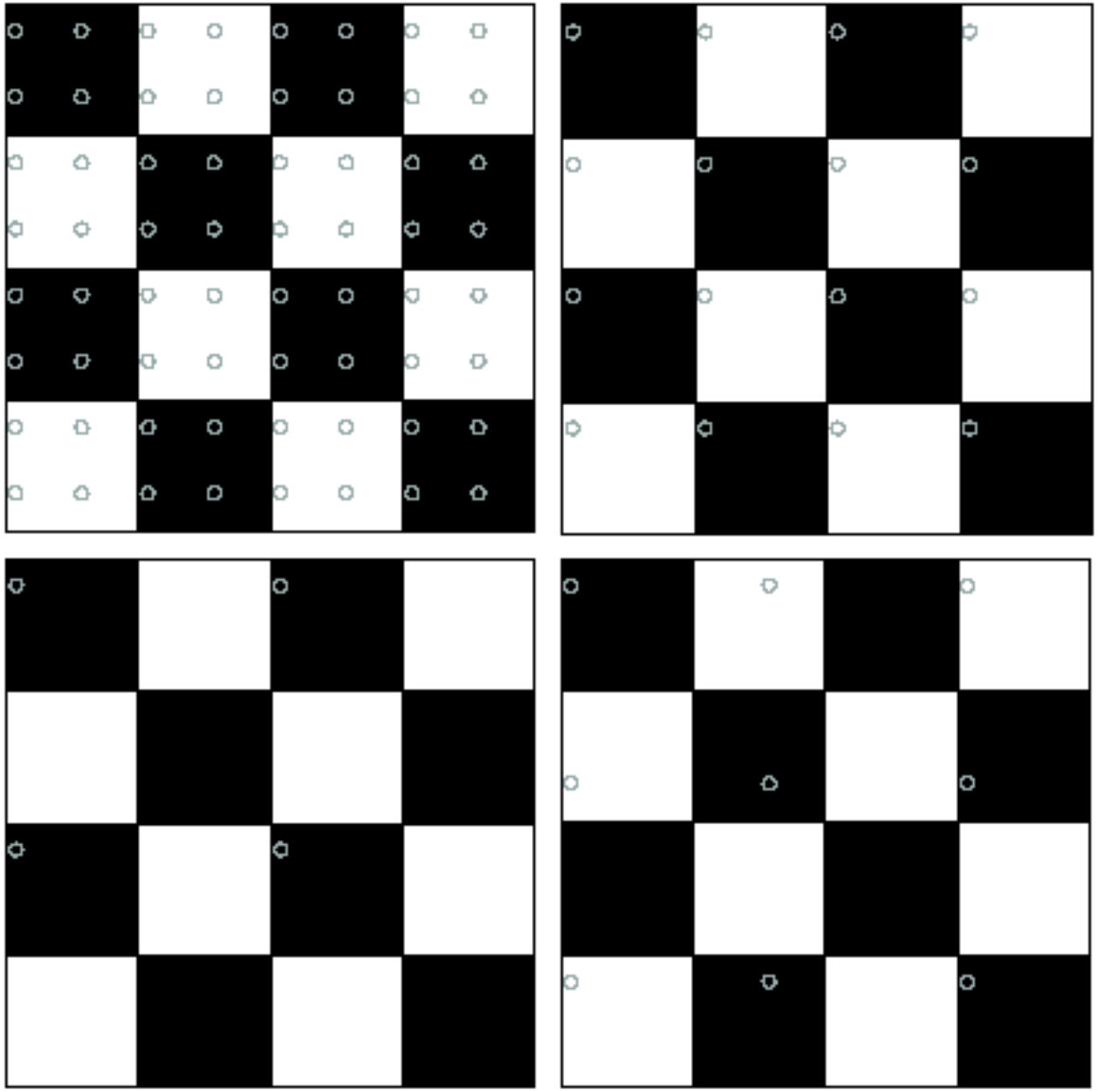
Aliasing



- Occurs when your sampling rate is not high enough to capture the amount of detail in your image
- Can give you the wrong signal/image—an *alias*
- To do sampling right, need to understand the structure of your signal/image
- To avoid aliasing:
  - sampling rate ≥ 2 * max frequency in the image
    - said another way: ≥ two samples per cycle
  - This minimum sampling rate is called the **Nyquist rate**

Source: L. Zhang

Good and Bad Sampling



Good sampling:
- Sample often or,
- Sample wisely

Bad sampling:
- Aliasing!

Aliasing

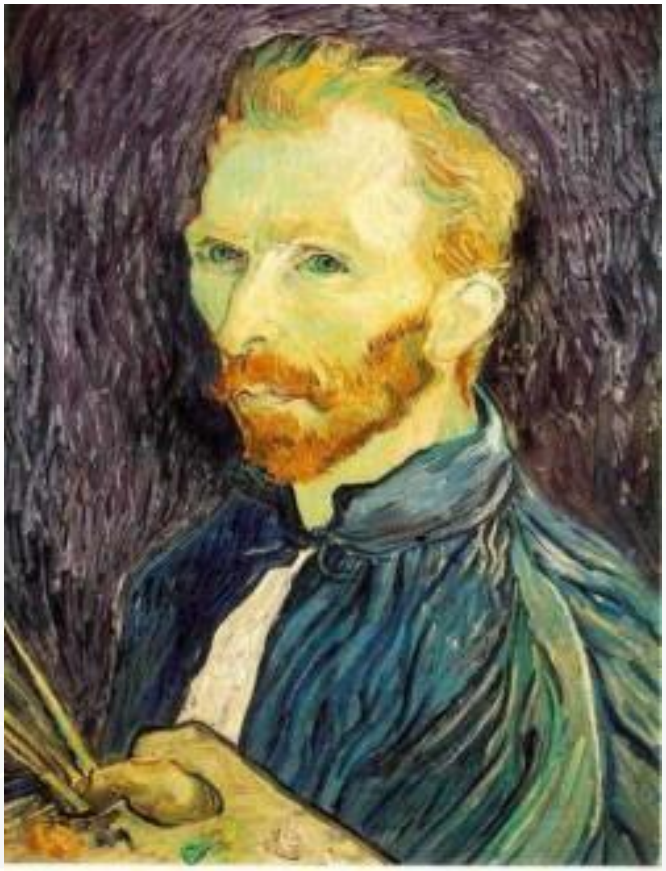
- When downsampling by a factor of two
  - Original image has frequencies that are too high

- How can we fix this?

Smoothing as low-pass filtering

- High frequencies lead to trouble with sampling.
- Suppress high frequencies before sampling !
  - truncate high frequencies in FT
  - or convolve with a low-pass filter
- Common solution: use a Gaussian
  - multiplying FT by Gaussian is equivalent to convolving image with Gaussian.

Gaussian pre-filtering



Gaussian 1/2

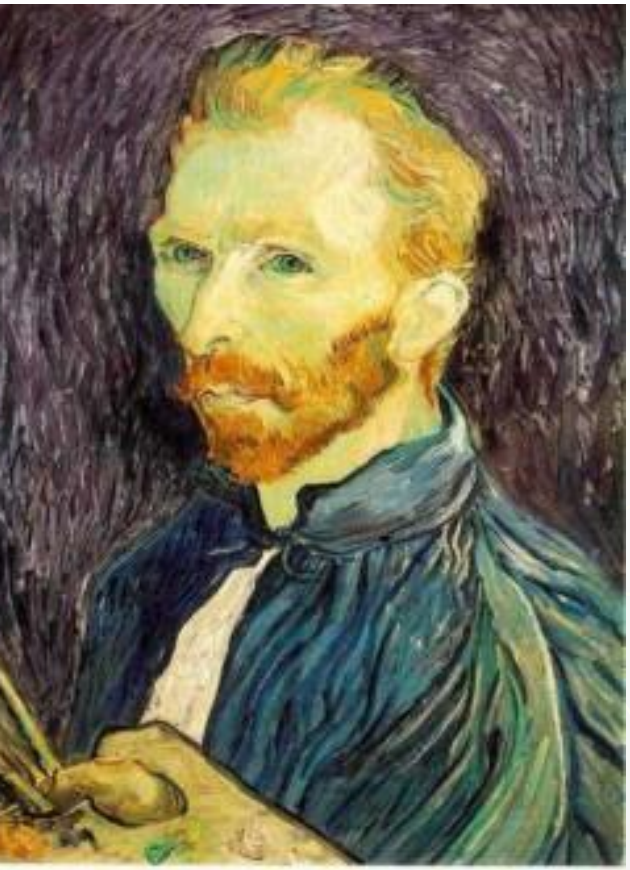
G 1/4

G 1/8

- Solution: filter the image, *then* subsample

Source: S. Seitz

Subsampling with Gaussian pre-filtering



Gaussian 1/2

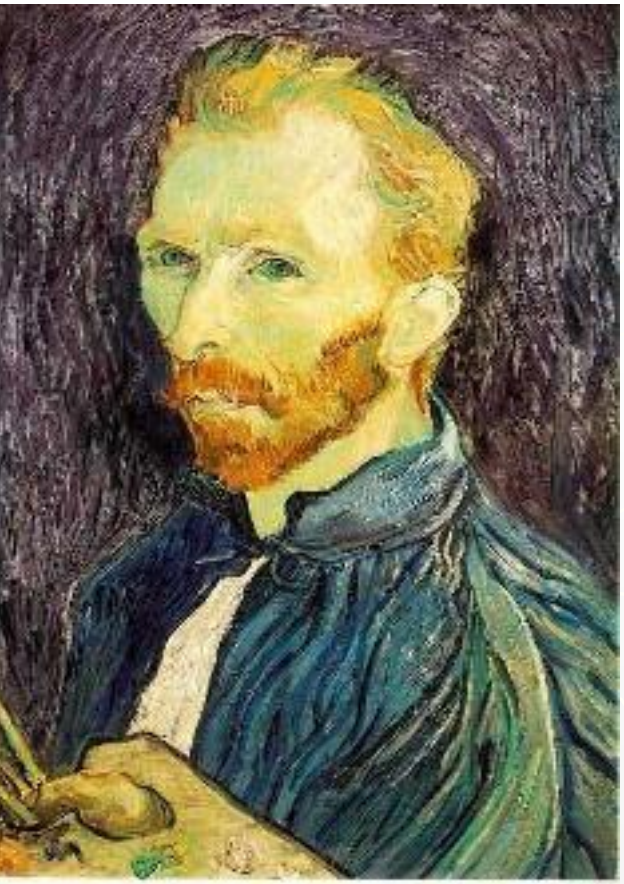
G 1/4

G 1/8

- Solution: filter the image, *then* subsample

Source: S. Seitz

Compare with...



1/2

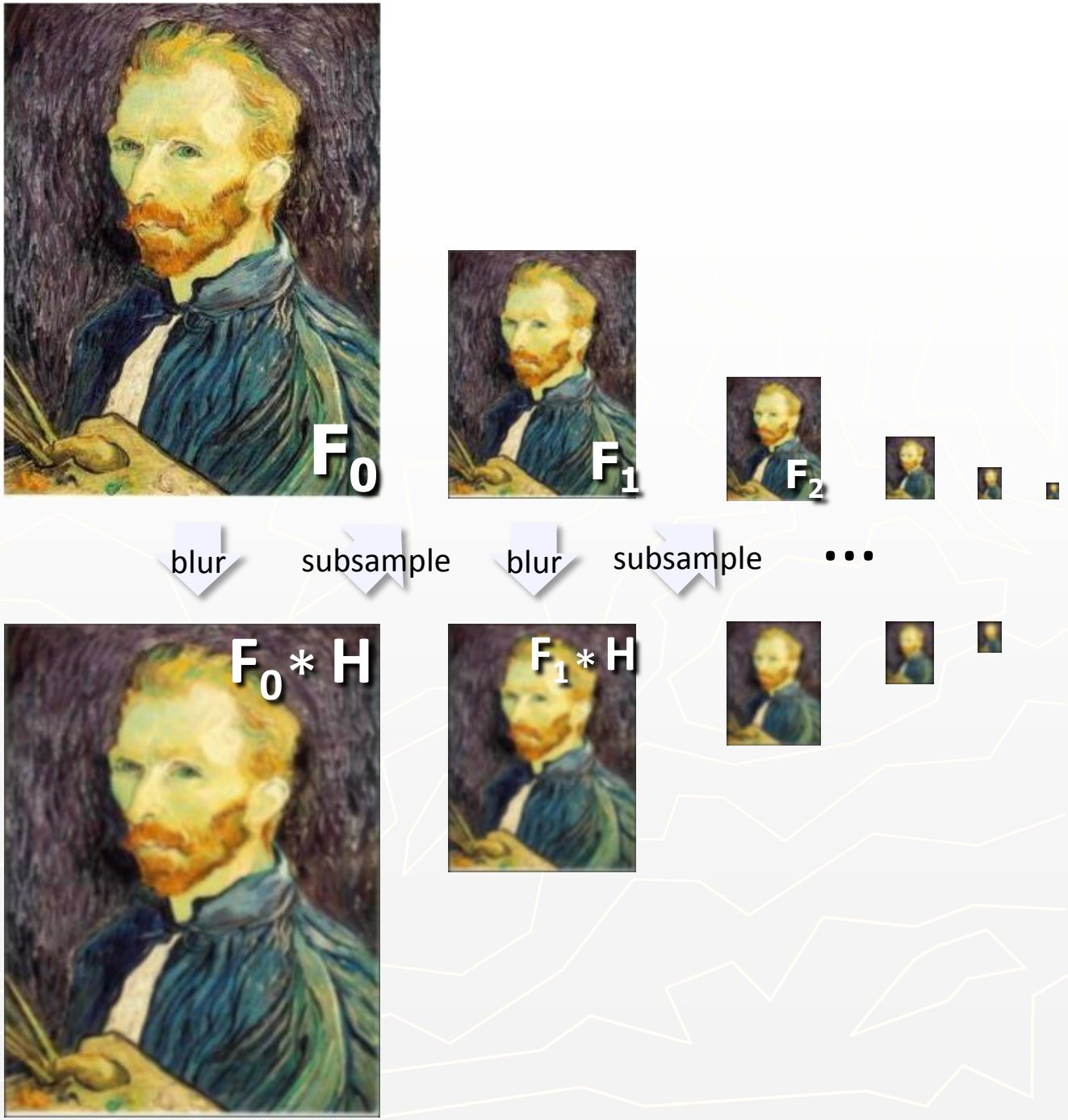
1/4 (2x zoom)

1/8 (4x zoom)

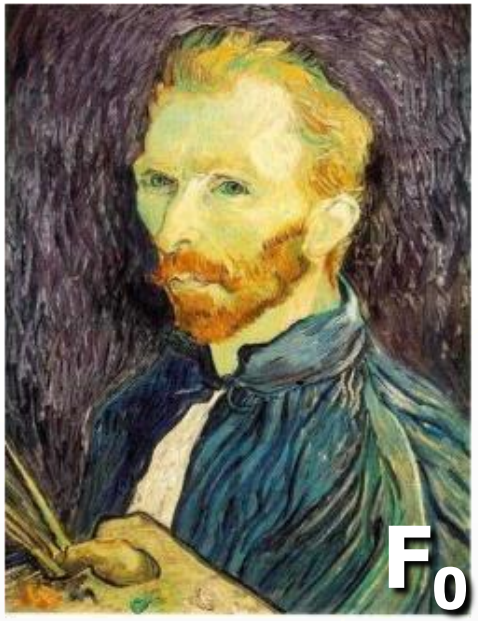
Source: S. Seitz

Gaussian pre-filtering

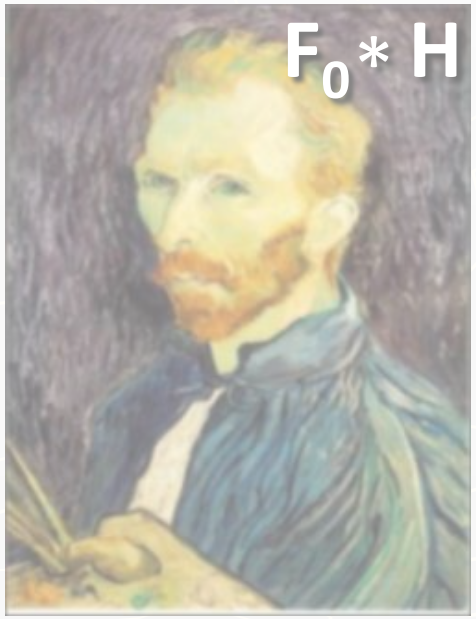
- Solution: filter the image, *then* subsample



Gaussian pyramid

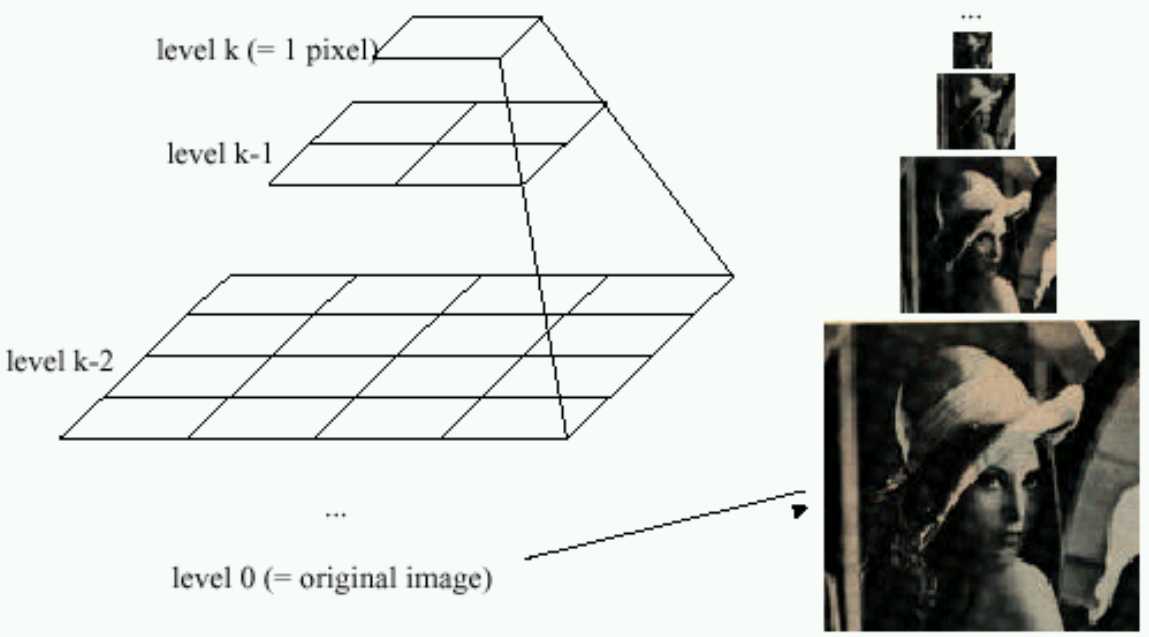


blur subsample blur subsample • • •



Gaussian pyramids [Burt and Adelson, 1983]

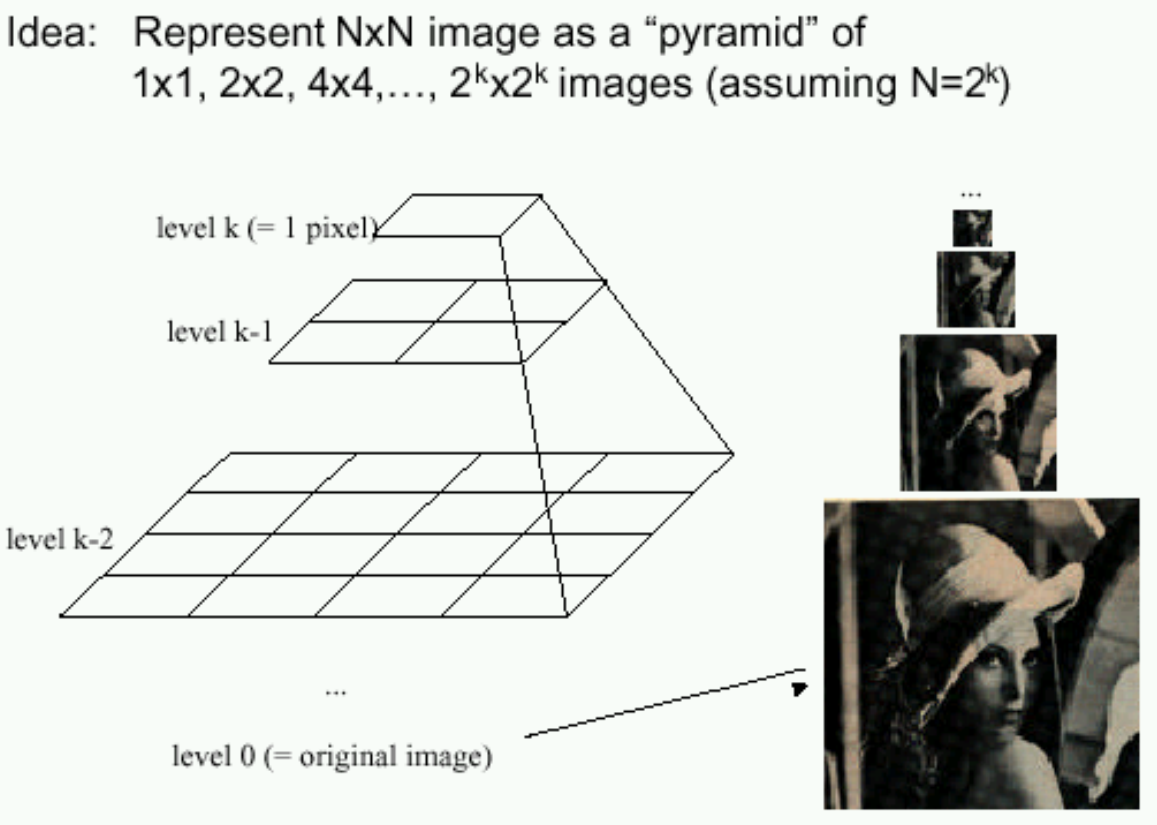
Idea: Represent NxN image as a "pyramid" of 1x1, 2x2, 4x4,…, 2^kx2^k images (assuming N=2^k)



Gaussian Pyramids have all sorts of applications in computer vision

Source: S. Seitz

Gaussian pyramids [Burt and Adelson, 1983]



- How much space does a Gaussian pyramid take compared to the original image?

Source: S. Seitz

Multi-Resolution Image Representation

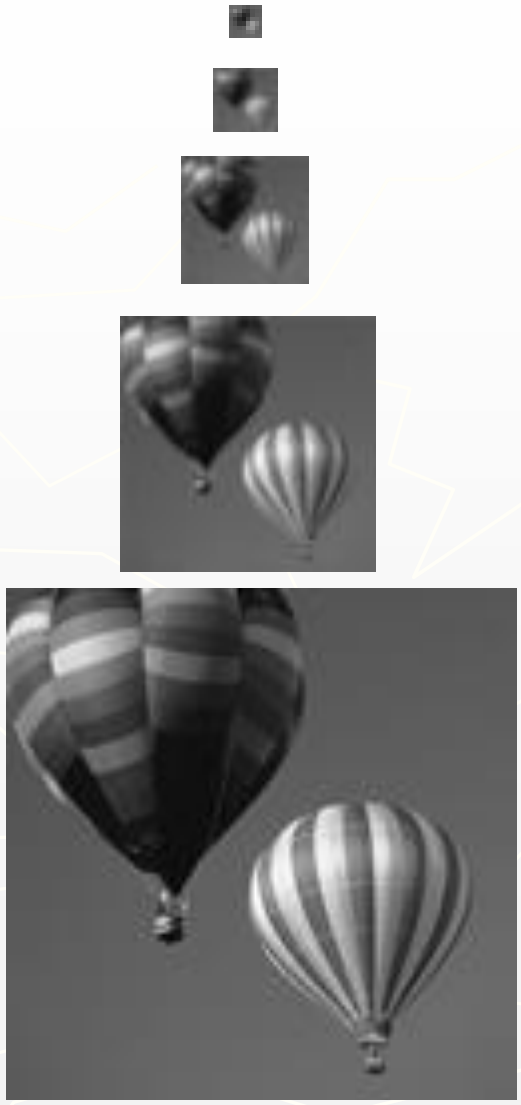
- Fourier domain tells us “what” (frequencies, sharpness, texture properties), but not “where”.
- Spatial domain tells us “where” (pixel location) but not “what”.
- We want a image representation that gives a local description of image “events” – what is happening where.
- Naturally, think about representing images across varying scales.

512 256 128 64 32 16 8

Figure from David Forsyth

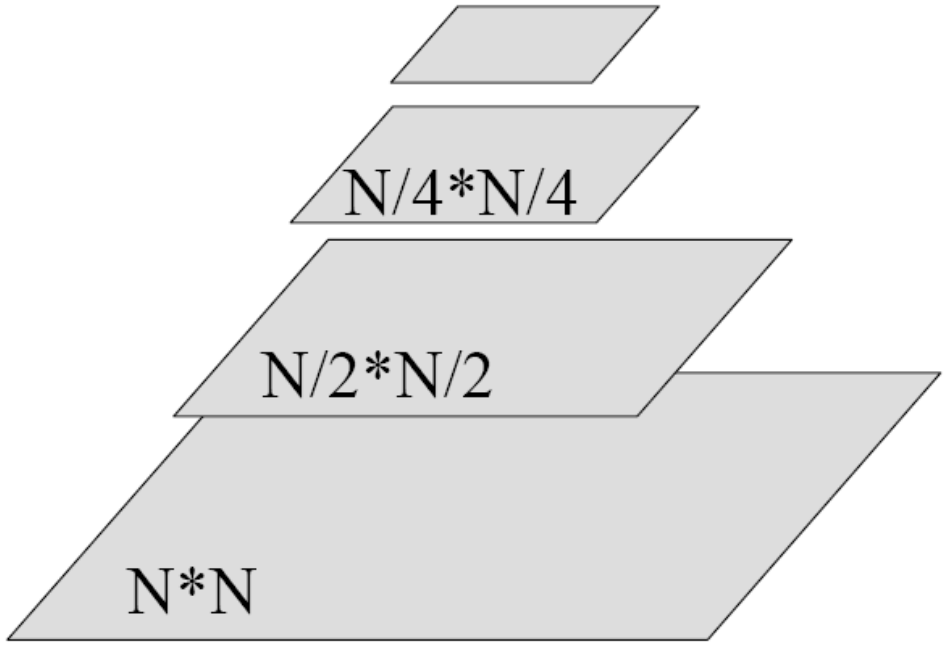
Multi-resolution Image Pyramids

Low resolution



High resolution

Space Required for Pyramids



$$N^2 + \frac{1}{4}N^2 + \frac{1}{16} + \cdots = 1\frac{1}{3}N^2$$

Decimation

*

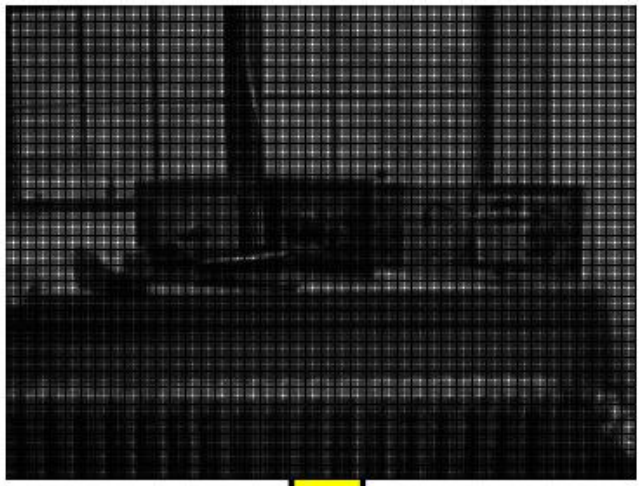
1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1

Filter

Subsample

Expansion

Expand Image



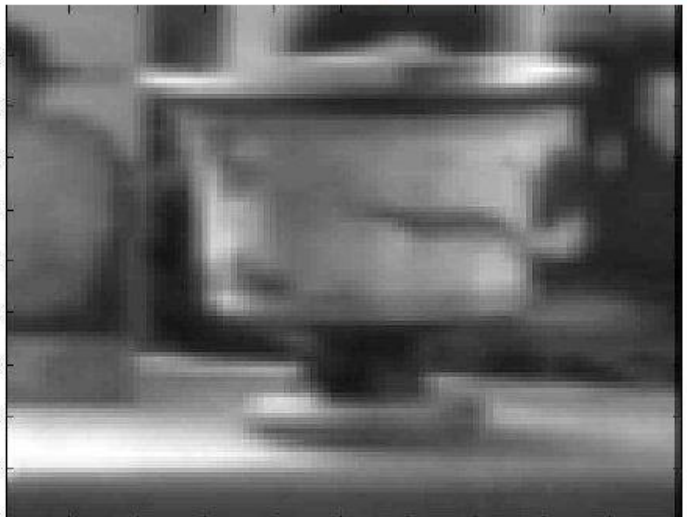
Interpolation

Interpolation Results

Original Image

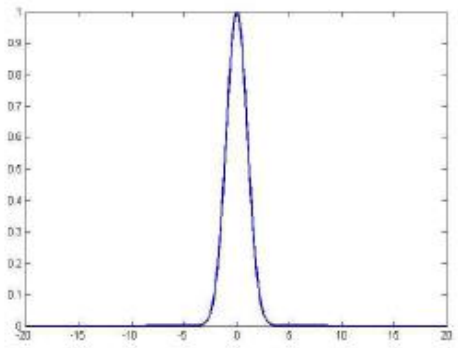
Nearest Neighbor

Bilinear Interpolation

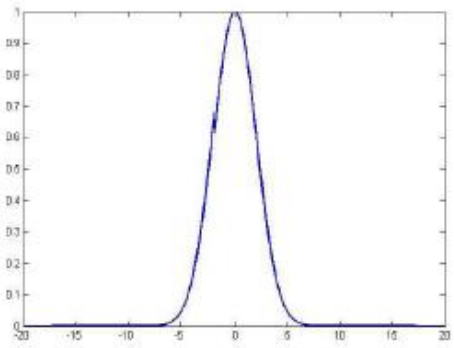


The Gaussian Pyramid

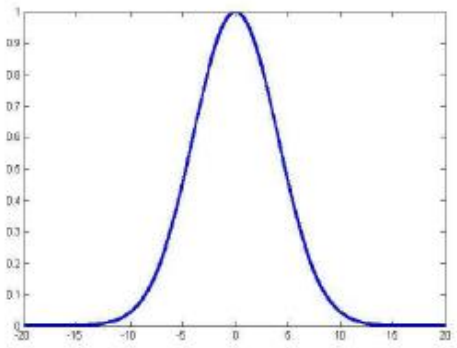
- Smooth with Gaussians because
  - a Gaussian*Gaussian=another Gaussian
- Synthesis
  - smooth and downsample
- Gaussians are low pass filters, so repetition is redundant
- Kernel width doubles with each level



Level 1



Level 2

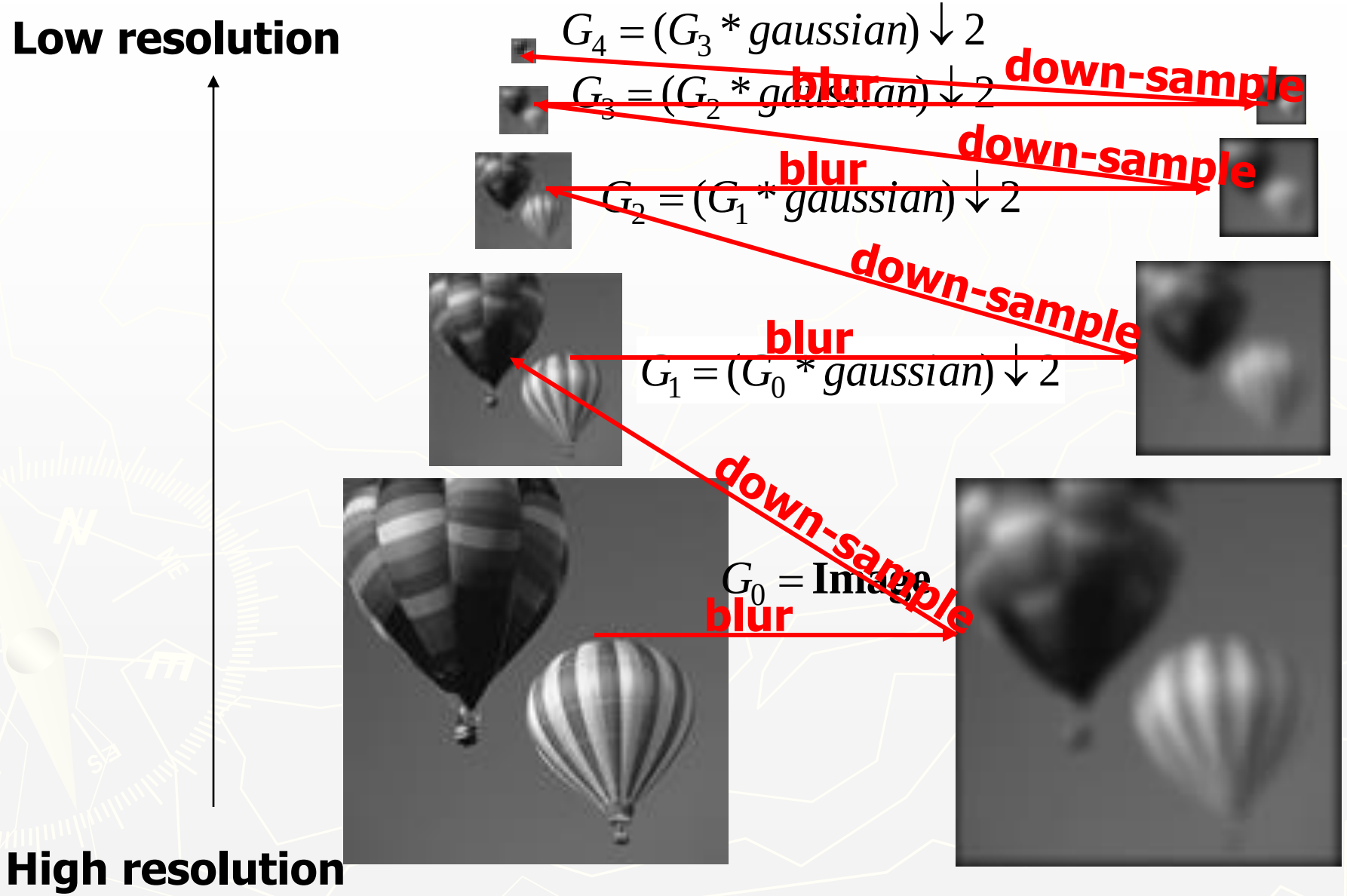


Level 3

Gaussian pyramids used for

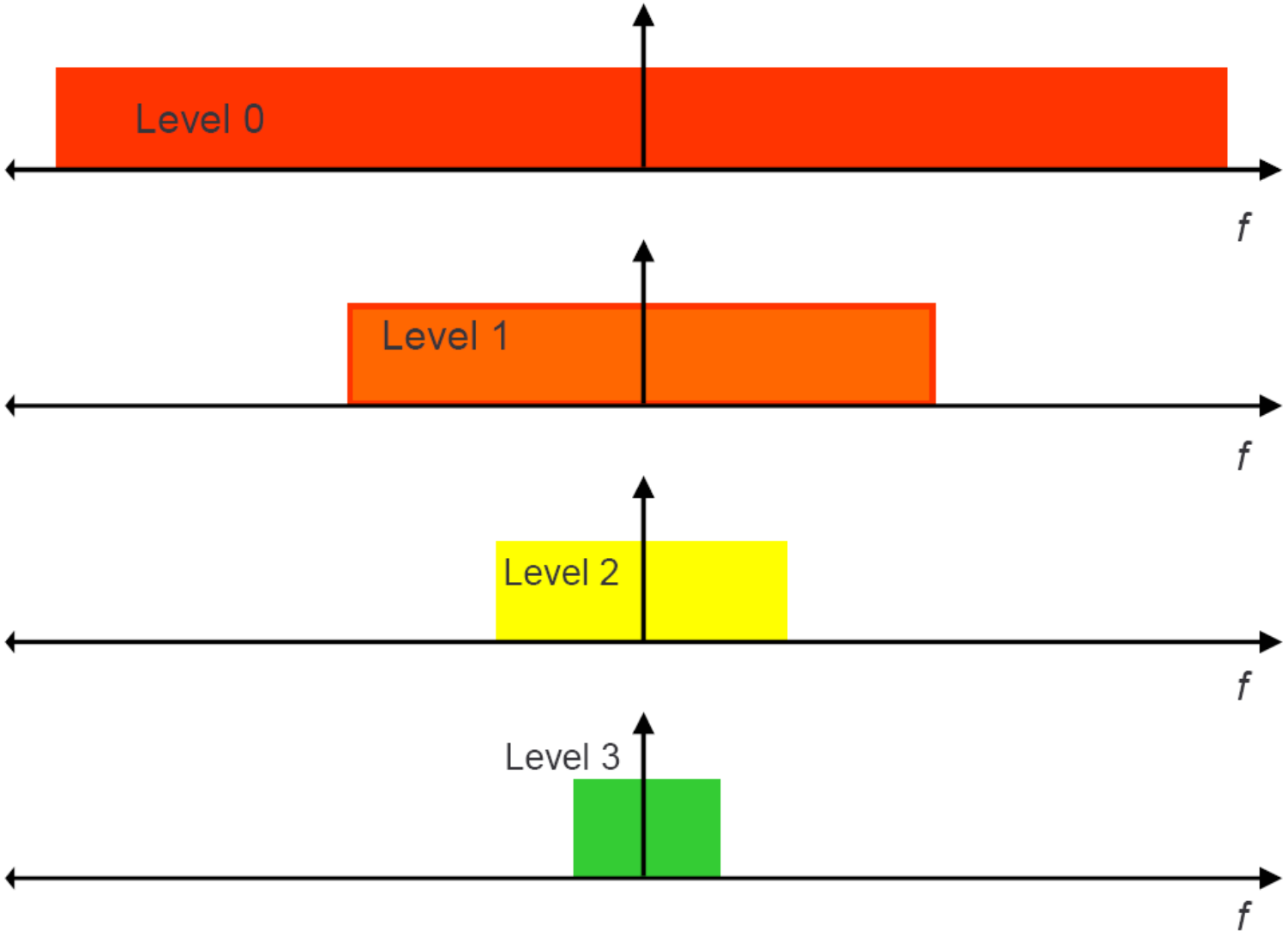
- up- or down- sampling images.
- Multi-resolution image analysis
  - Look for an object over various spatial scales
  - Coarse-to-fine image processing: form blur estimate or the motion analysis on very low-resolution image, upsample and repeat. Often a successful strategy for avoiding local minima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

The Gaussian Pyramid

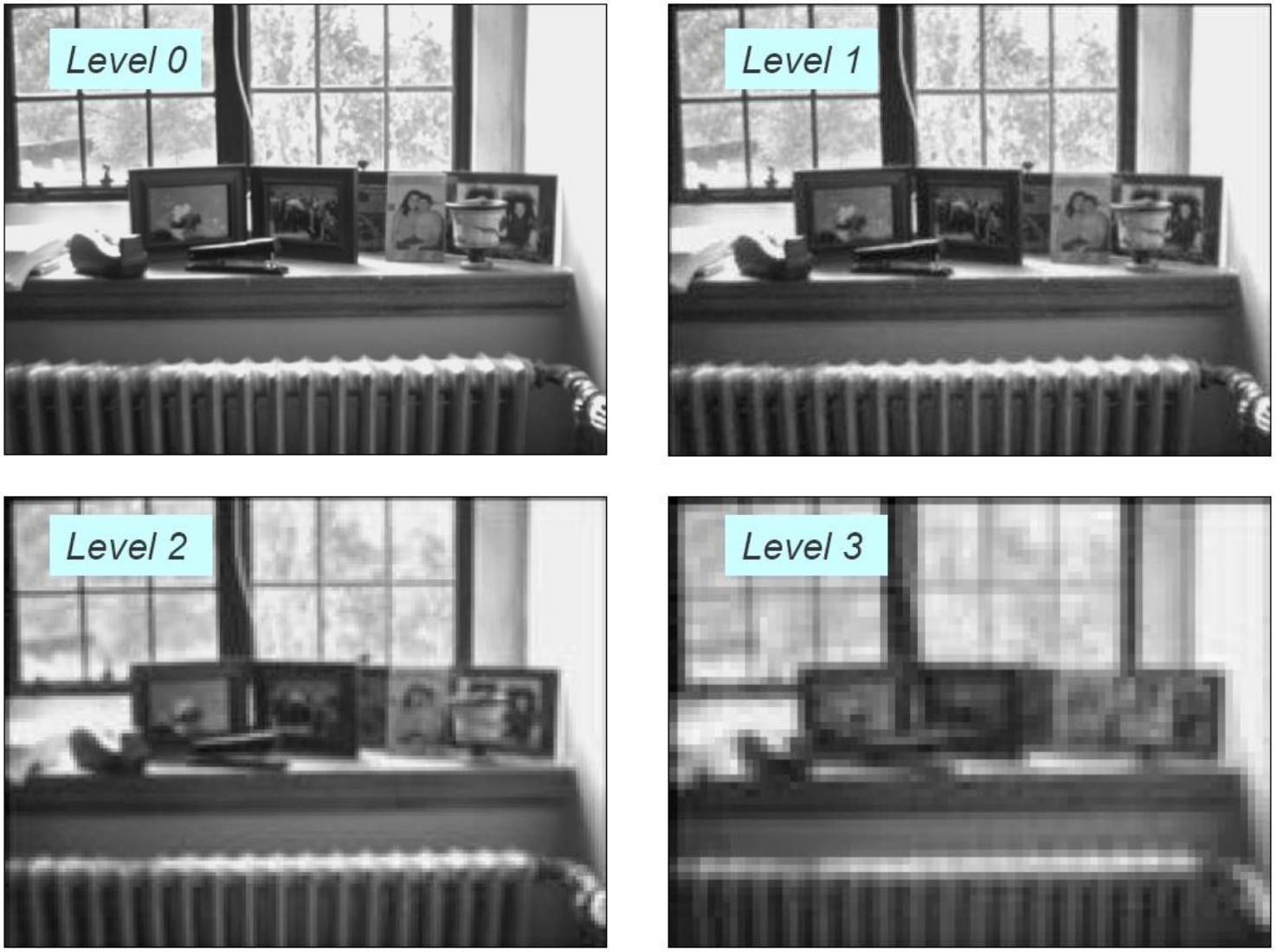


512 256 128 64 32 16 8

Gaussian Pyramid
Frequency Composition



Pyramids at Same Resolution



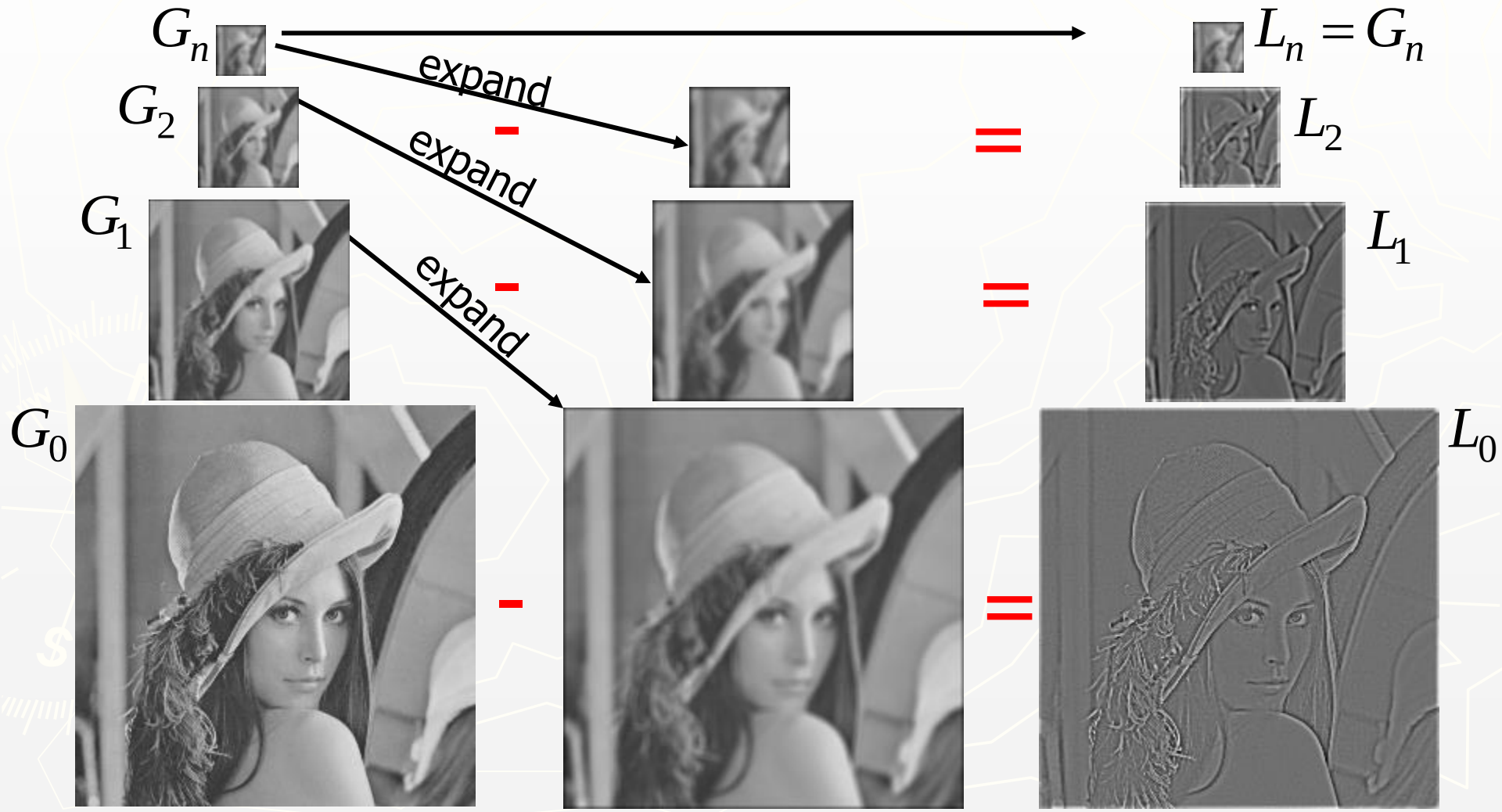
The Laplacian Pyramid

$$L_i = G_i - \text{expand}(G_{i+1})$$

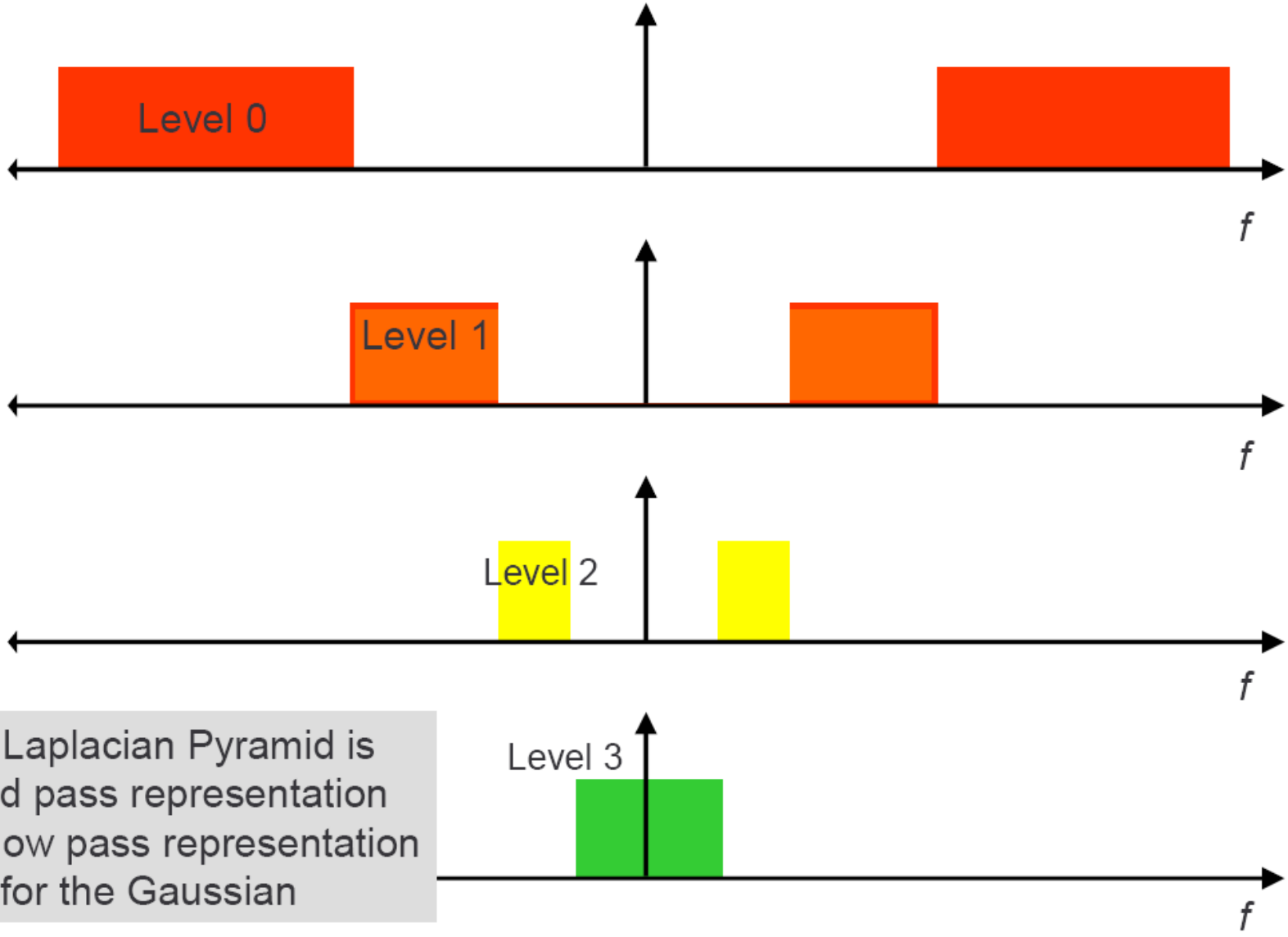
$$G_i = L_i + \text{expand}(G_{i+1})$$

Gaussian Pyramid

Laplacian Pyramid



Laplacian Pyramid Frequency Composition



The Laplacian Pyramid is a band pass representation vice a low pass representation for the Gaussian

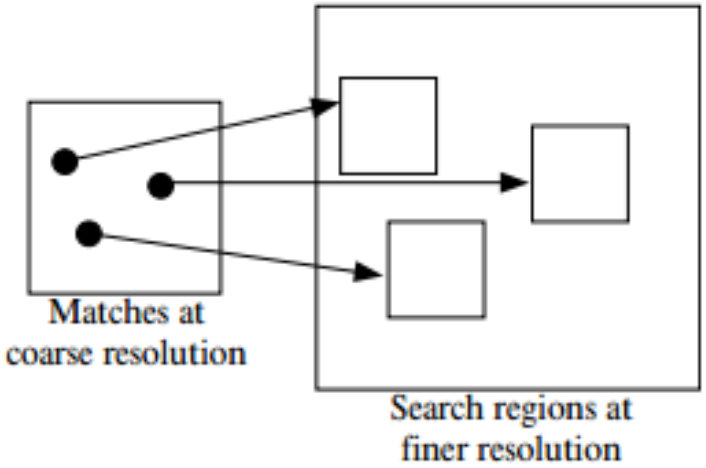
Applications of Image Pyramids

- Coarse-to-Fine strategies for computational efficiency.
- Search for correspondence
  - look at coarse scales, then refine with finer scales
- Edge tracking
  - a "good" edge at a fine scale has parents at a coarser scale
- Control of detail and computational cost in matching
  - e.g. finding stripes
  - very important in texture representation
- Image Blending
- Data compression (laplacian pyramid)

Fast Template Matching

Idea: Reduce computational complexity by using a *pyramid*

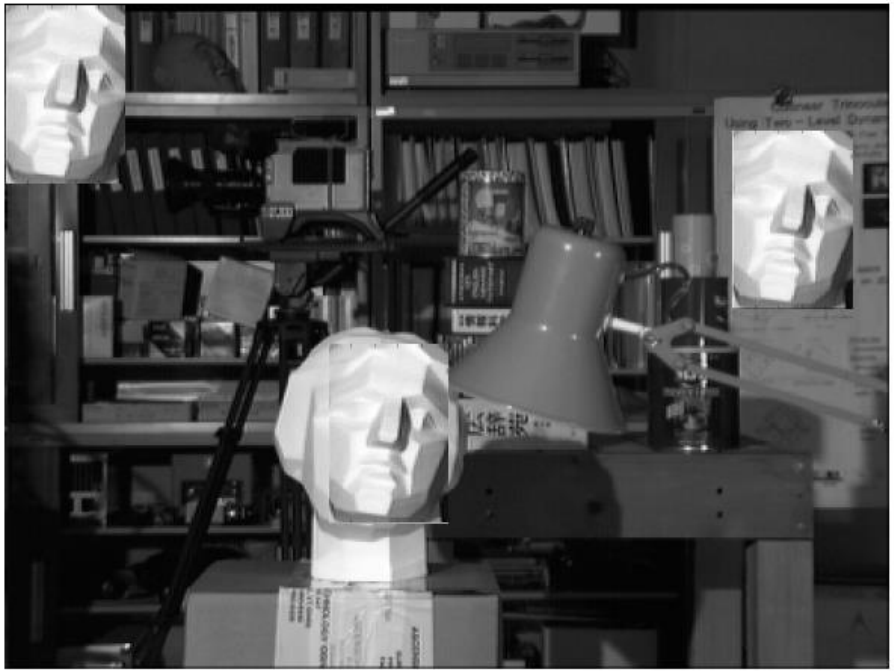
- Matching a coarser template to a coarser level of the pyramid requires fewer comparisons.
- Only positions/transformations that exceed some threshold need to be evaluated/compared for the next-finer resolution.
- Proceeds until the finest resolution is reached.



Fast Template Matching

Template

Search Region



- For an m x n image…
- For a p x q template…
- The complexity of the 2D pattern recognition task is O(mnpq) ☹
- This gets even worse for a family of templates (*e.g.*, to address scale and/or rotational effects)

Fast Template Matching

Template

Search Region

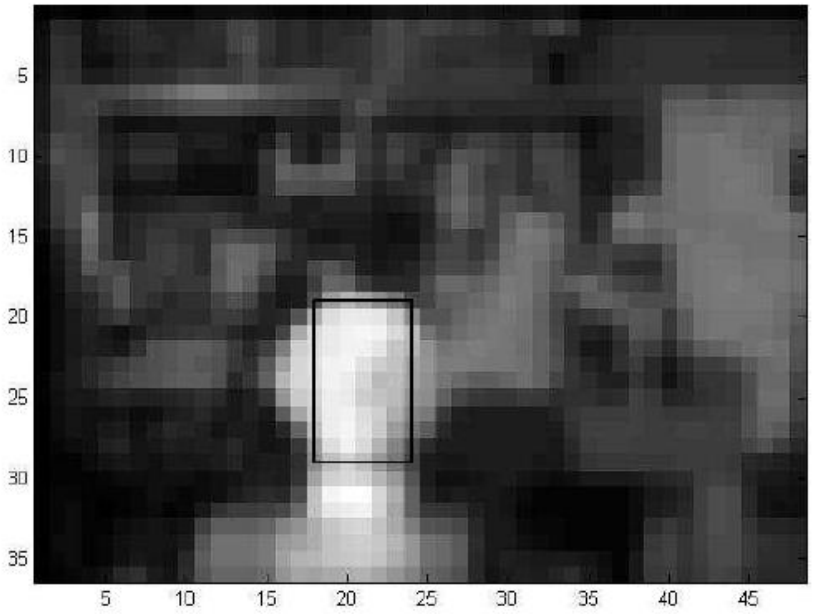
Original Image

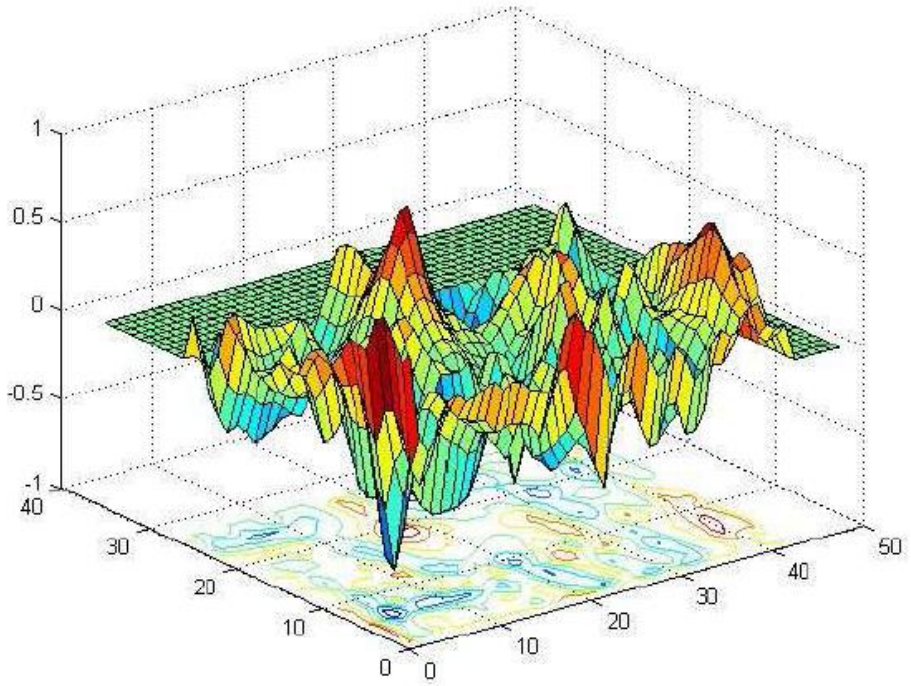
Multi-Resolution Correlation

- Multi-resolution template matching
  - reduce resolution of both template and image by creating an **image pyramid**
  - match small template against small image
  - identify locations of strong matches
  - expand the image and template, and match higher resolution template selectively to higher resolution image
  - iterate on higher and higher resolution images
- Issue:
  - how to choose detection thresholds at each level
    - too low will lead to too much cost
    - too high will miss match

Level 3 Search

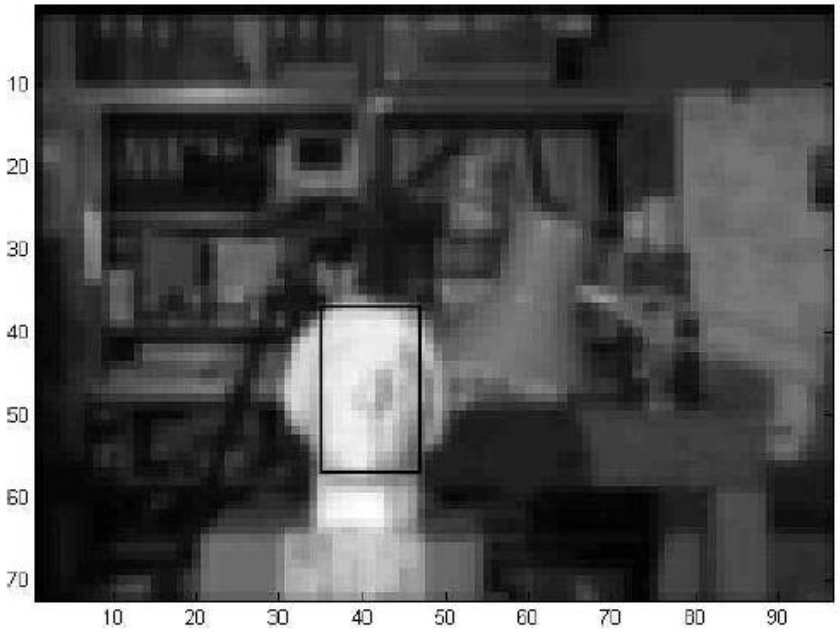
- At the lowest pyramid level, we search the entire image with the correlation template

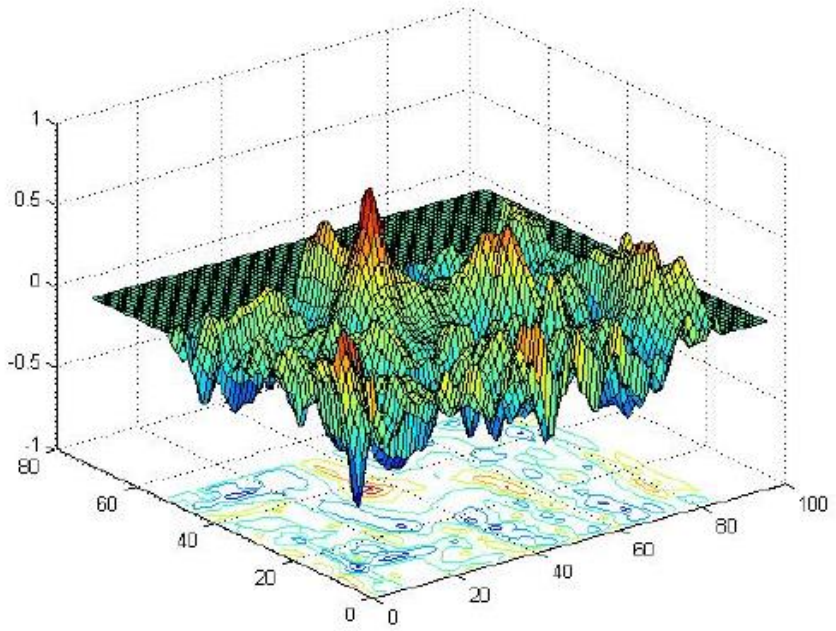




Level 2 Search

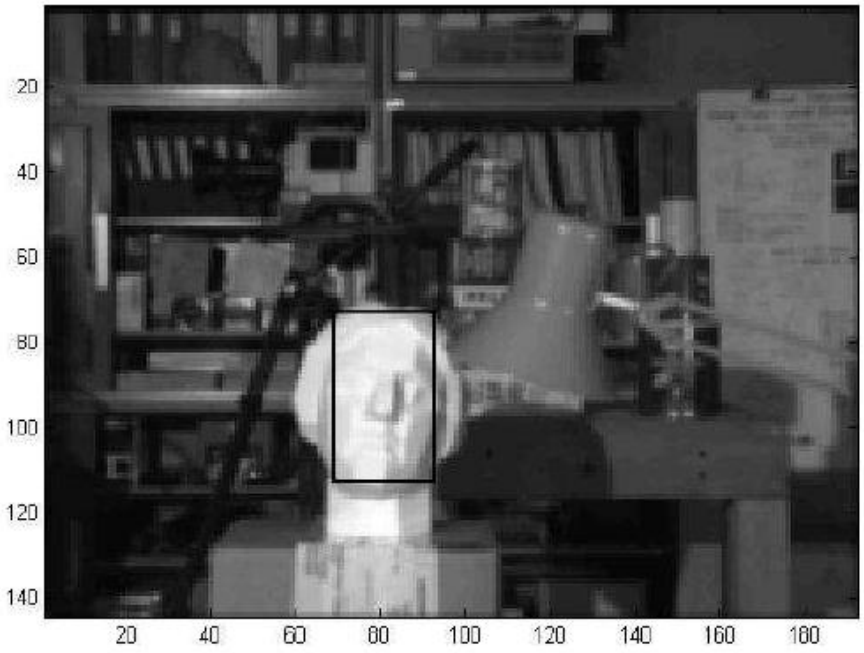
- Subsequent searches are constrained to a neighborhood of only several pixels in the x and y directions

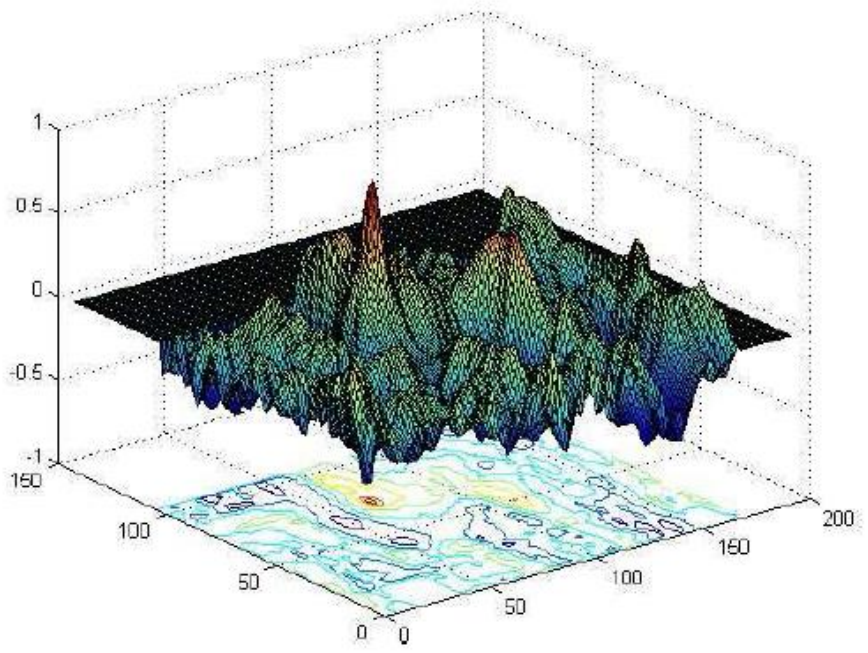




Level 1 Search

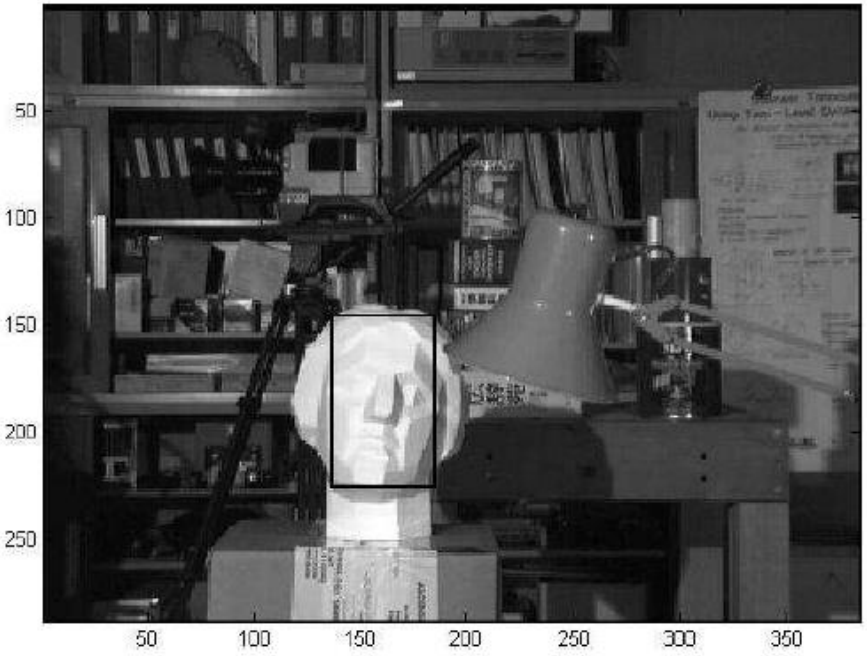
- Subsequent searches are constrained to a neighborhood of only several pixels in the x and y directions

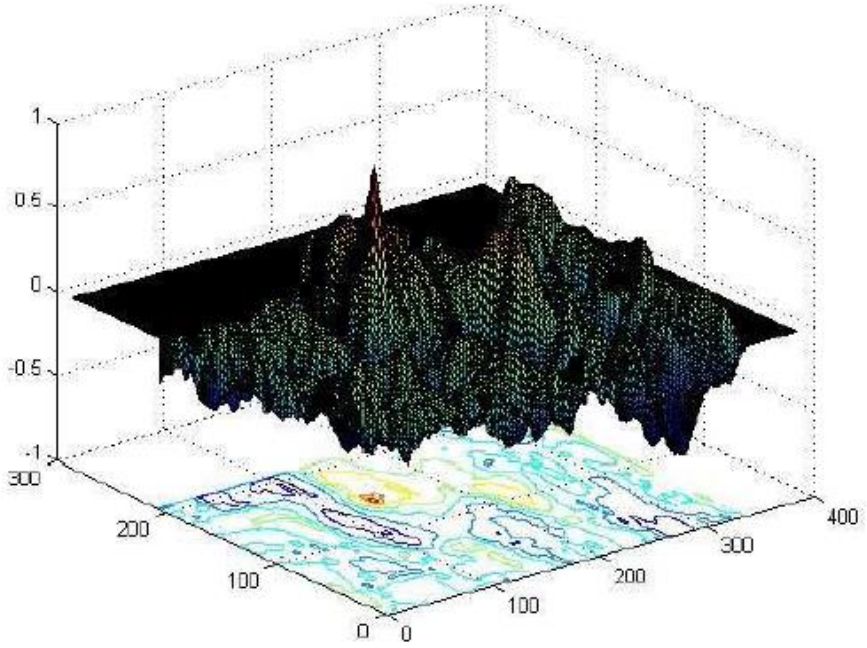




Level 0 Search

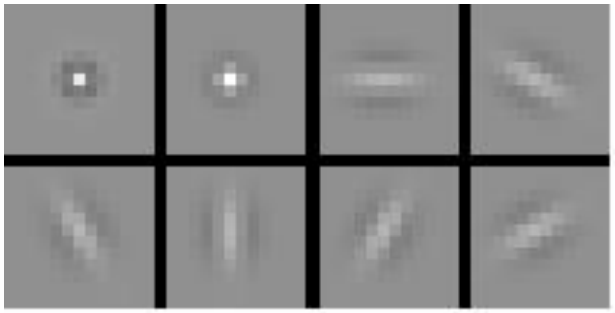
- In the end, the total time (in Matlab) was reduced from ≈ 31 seconds to ≈ 0.5 seconds while obtaining the same template match





Measuring response to selected filters (example)

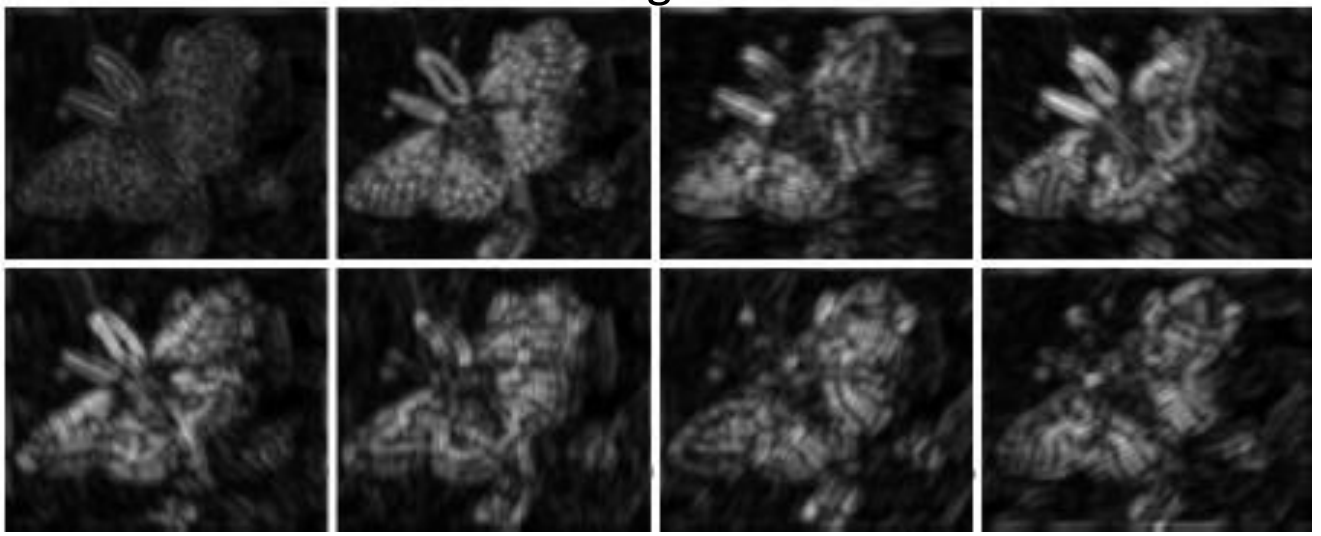
See David section 9.1.1 for the construction of spot and bar filters

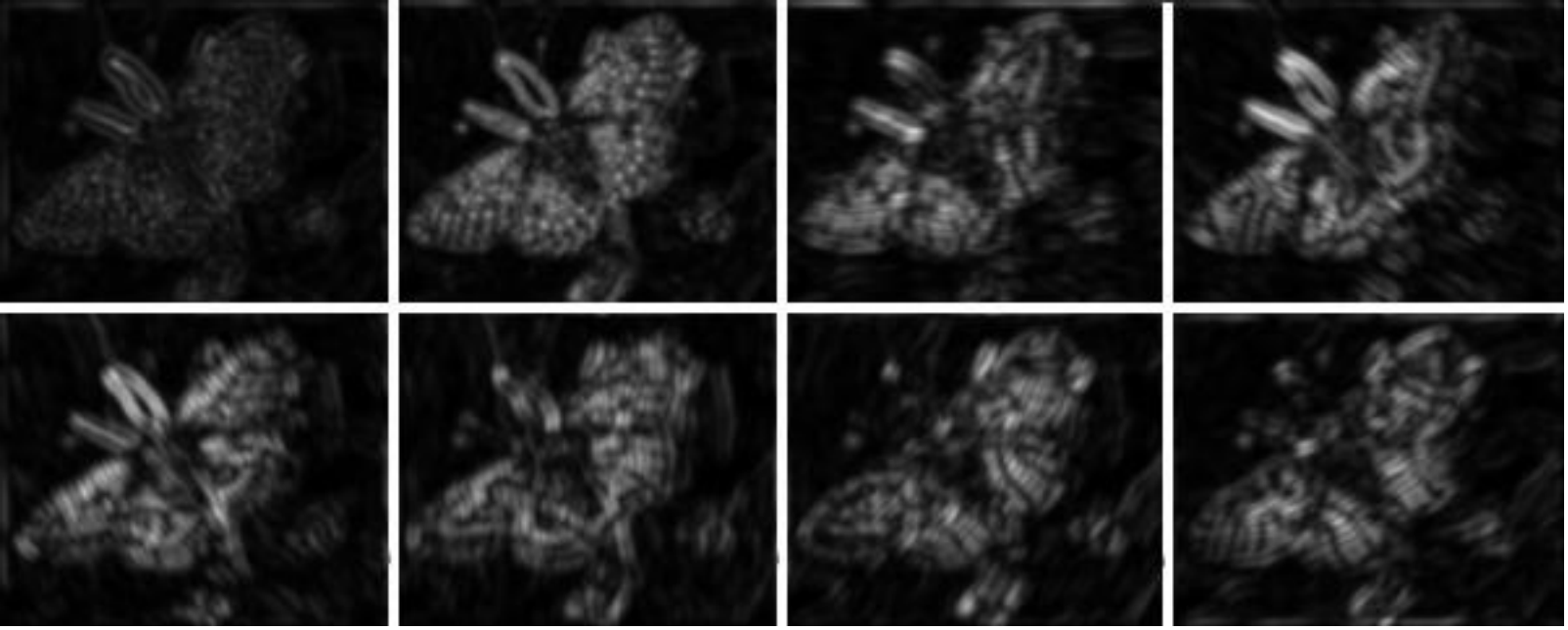


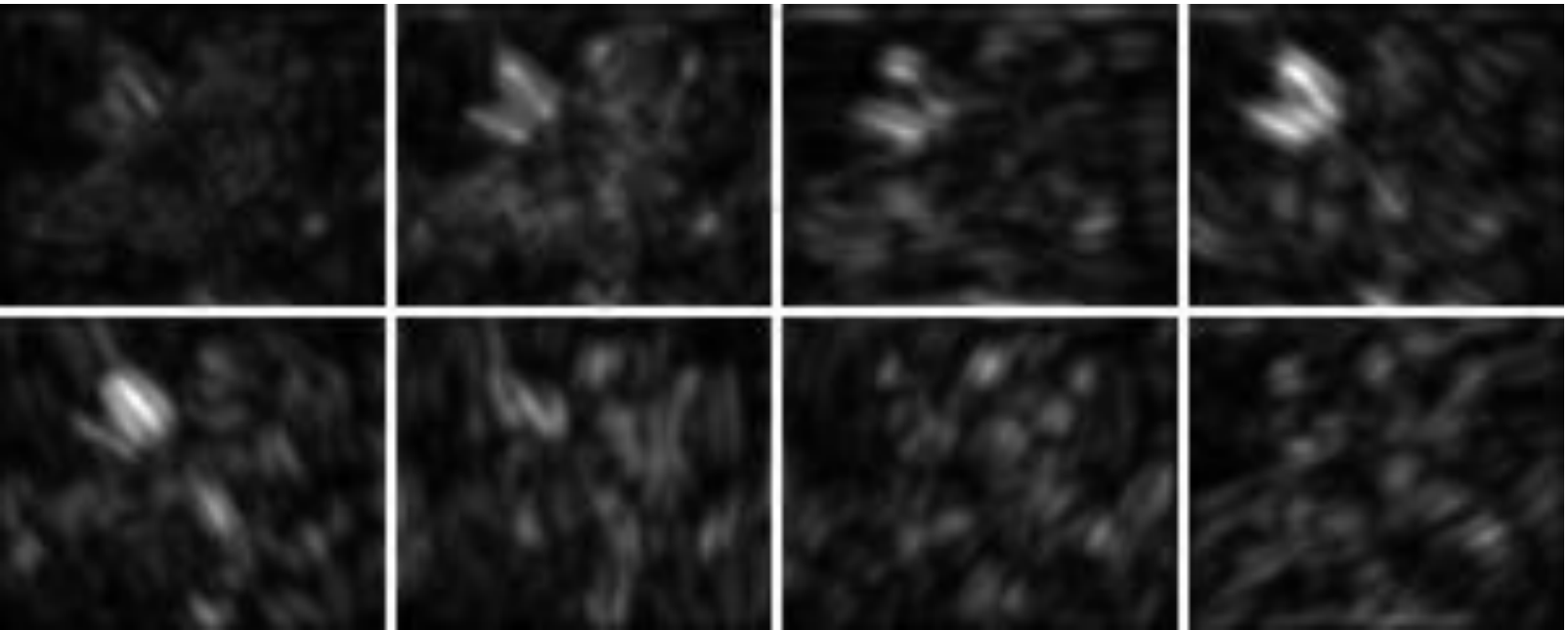
Eight Spot and bar filters

Input image

Filtered images



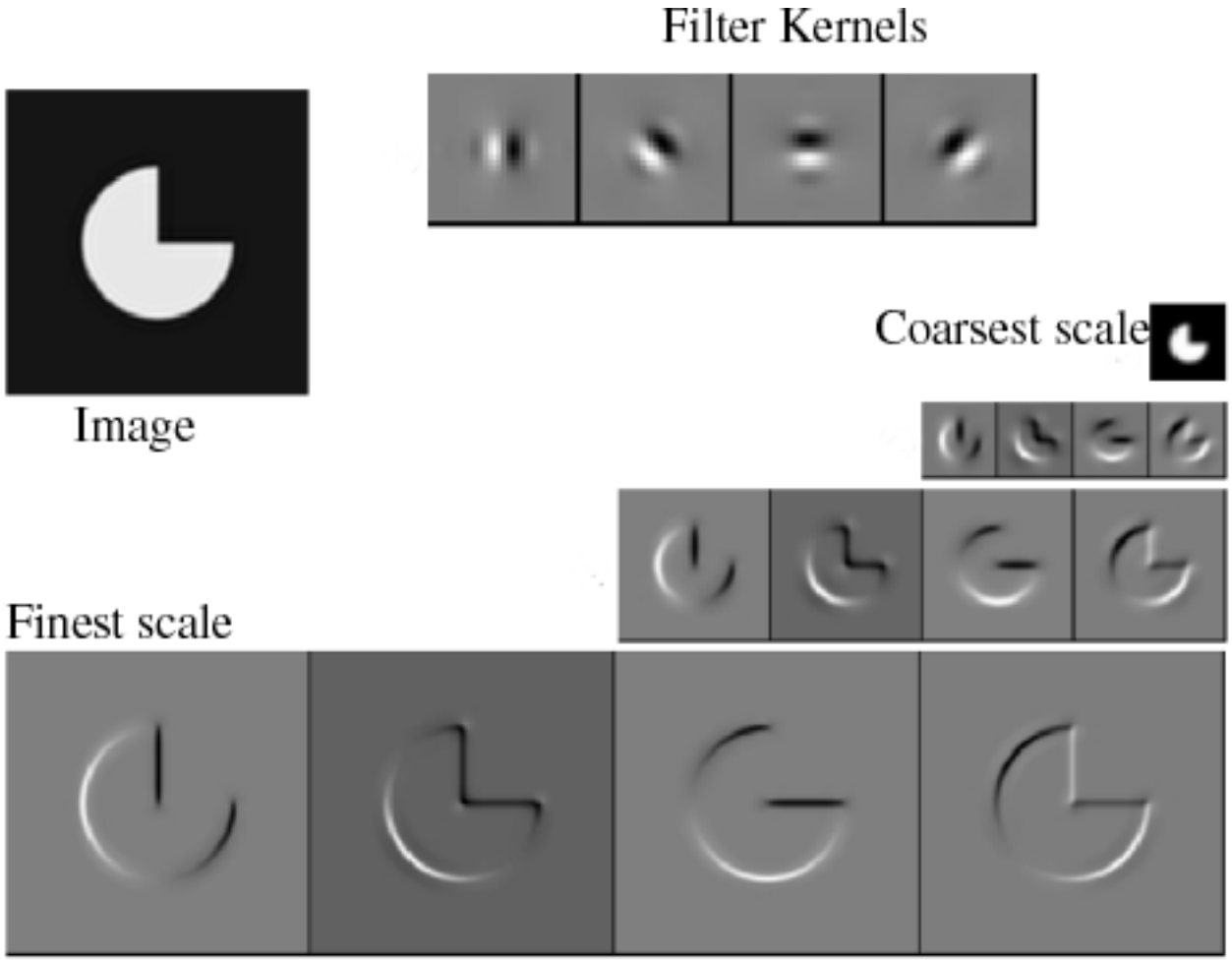




Oriented pyramids

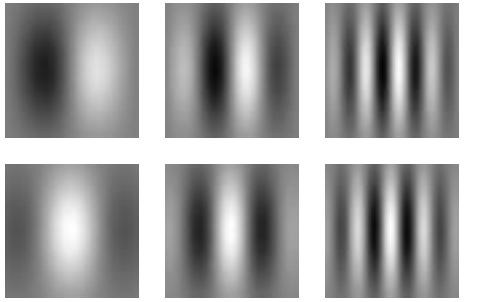
- Laplacian pyramid is orientation independent
- Apply an oriented filter to determine orientations at each layer
  - This represents image information at a particular scale and orientation.

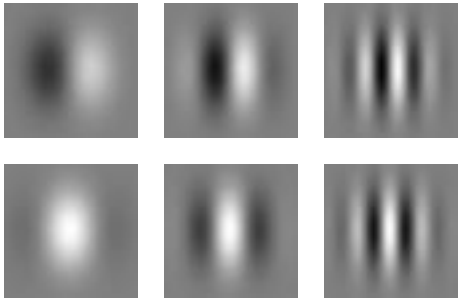
Oriented Pyramids



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

Alternative: Gabor filters





Gabor filters: Product of a Gaussian with sine or cosine

Top row shows anti-symmetric (or odd) filters, bottom row the symmetric (or even) filters.

Acknowledgements

Chapter-7 and chapter-9 from David A. Forsyth and Jean Ponce, "*Computer Vision A Modern Approach", 2nd edition, Prentice Hall, Inc., 2003.*

Material in these slides has been taken from, the following resources