Image Processing – lesson 8

Image Representation

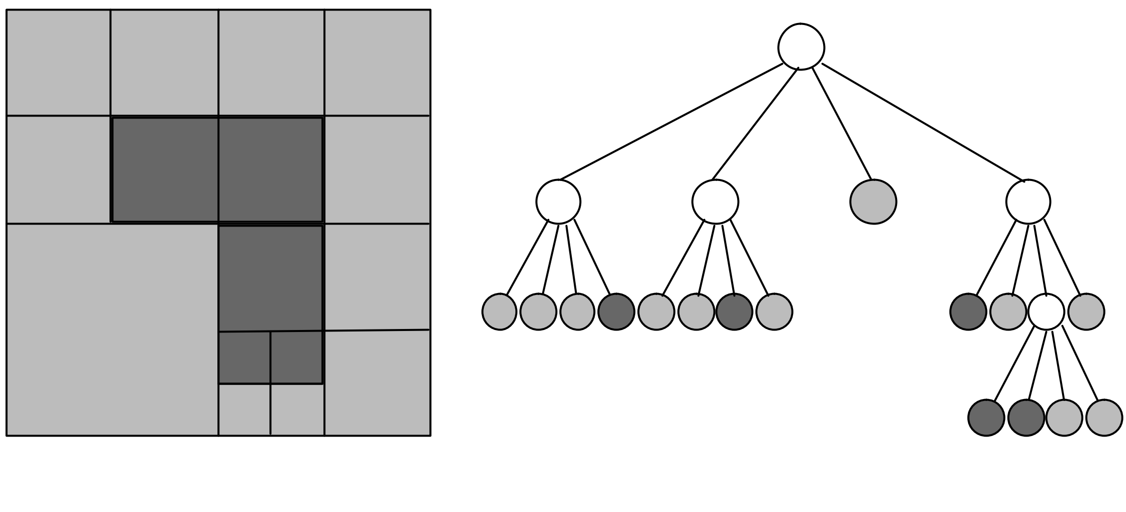
- Quad Trees
- Gaussian pyramids
- Laplacian Pyramids
- Wavelet Pyramids
- Applications

Quad Trees

Quad tree image representation = a tree representation which represents recursive subdivisions of an image.

Example:

Quad tree representation of an image



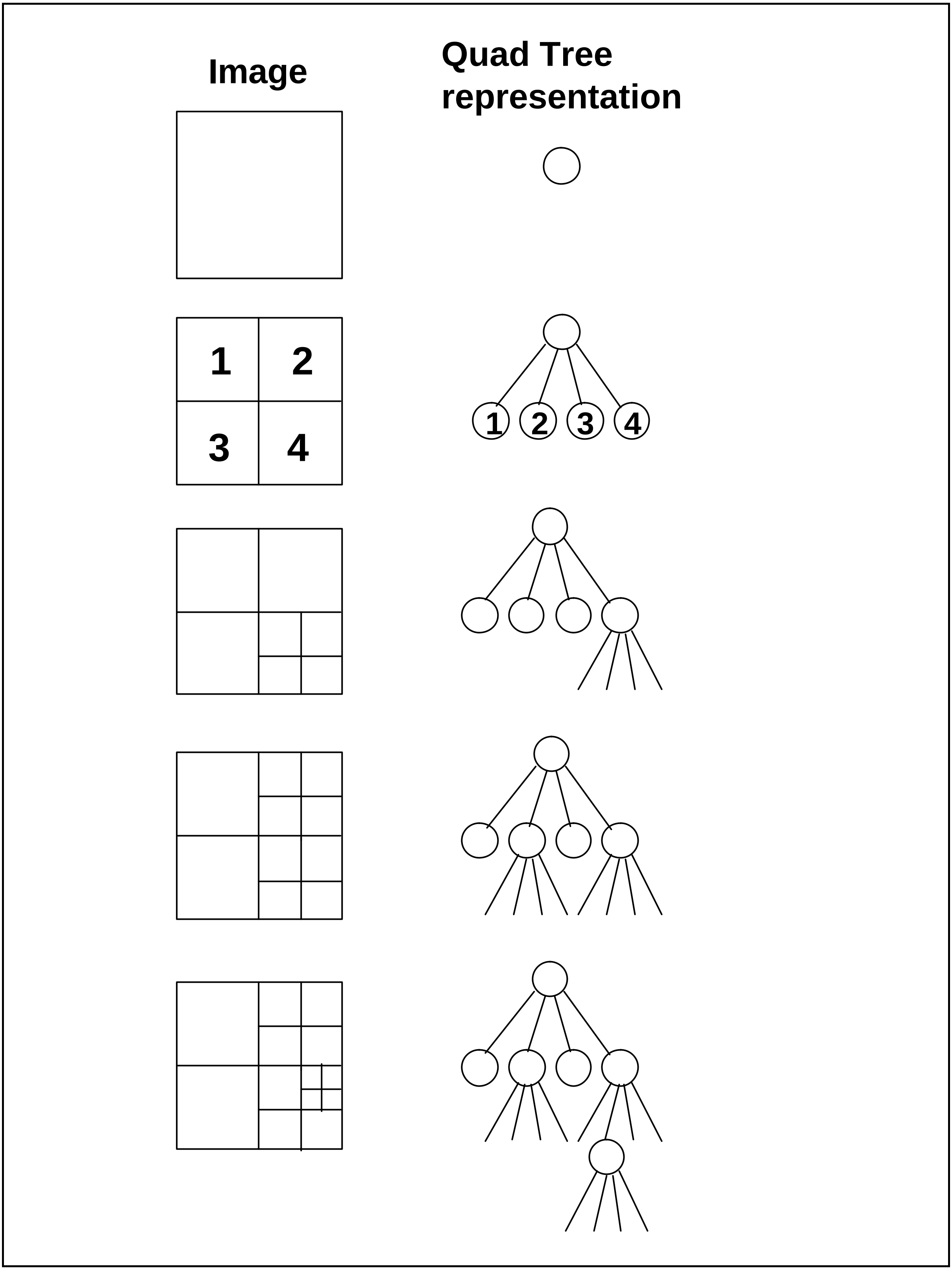
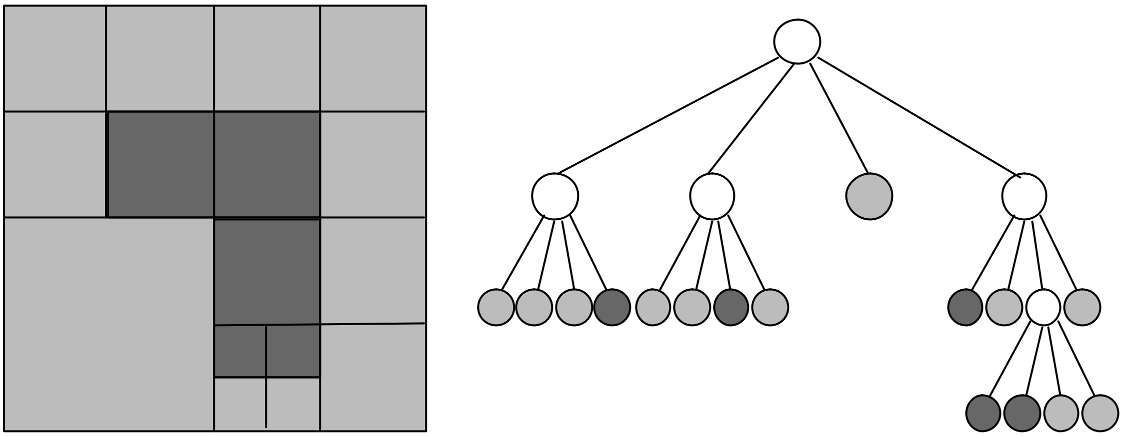


Image
Quad Tree representation
1
2
3
4
1
2
3
4

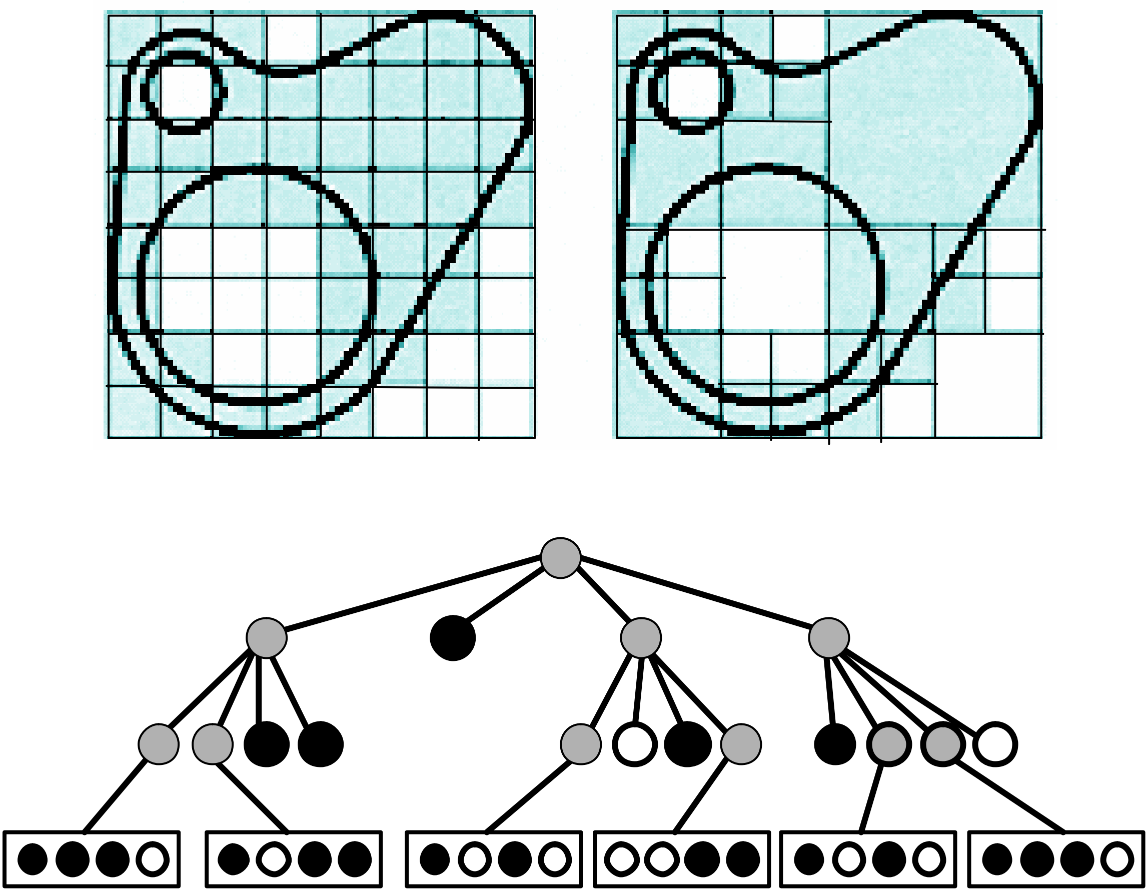
Quad Tree Applications:

- Compression
- Segmentation (Split & Merge)
- Smoothing
- Binary Image Operations (“And” “Or” “Not”)

Quad Tree Representation - Example

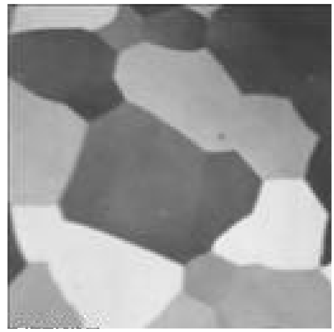


Quad Tree Representation - Example

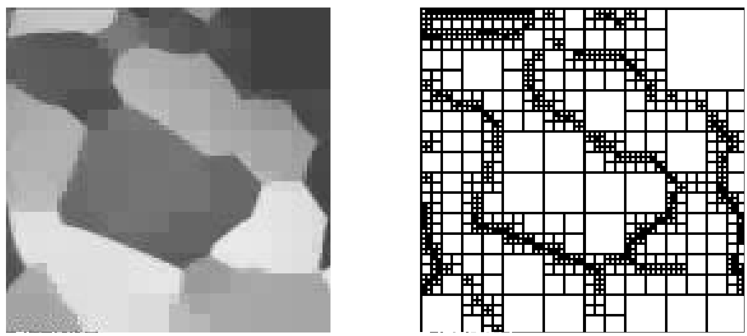


Quad Tree Representation

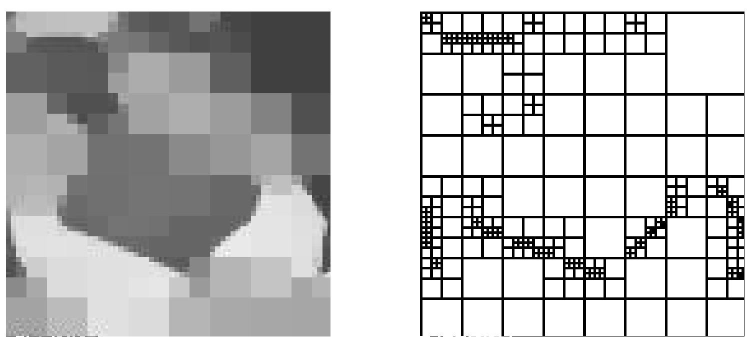
Original



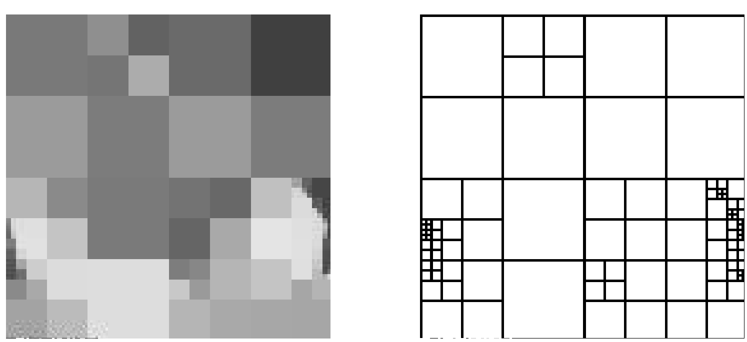
Thresh = 0.20



Thresh = 0.40



Thresh = 0.55



Binary Operations Using Quad Trees

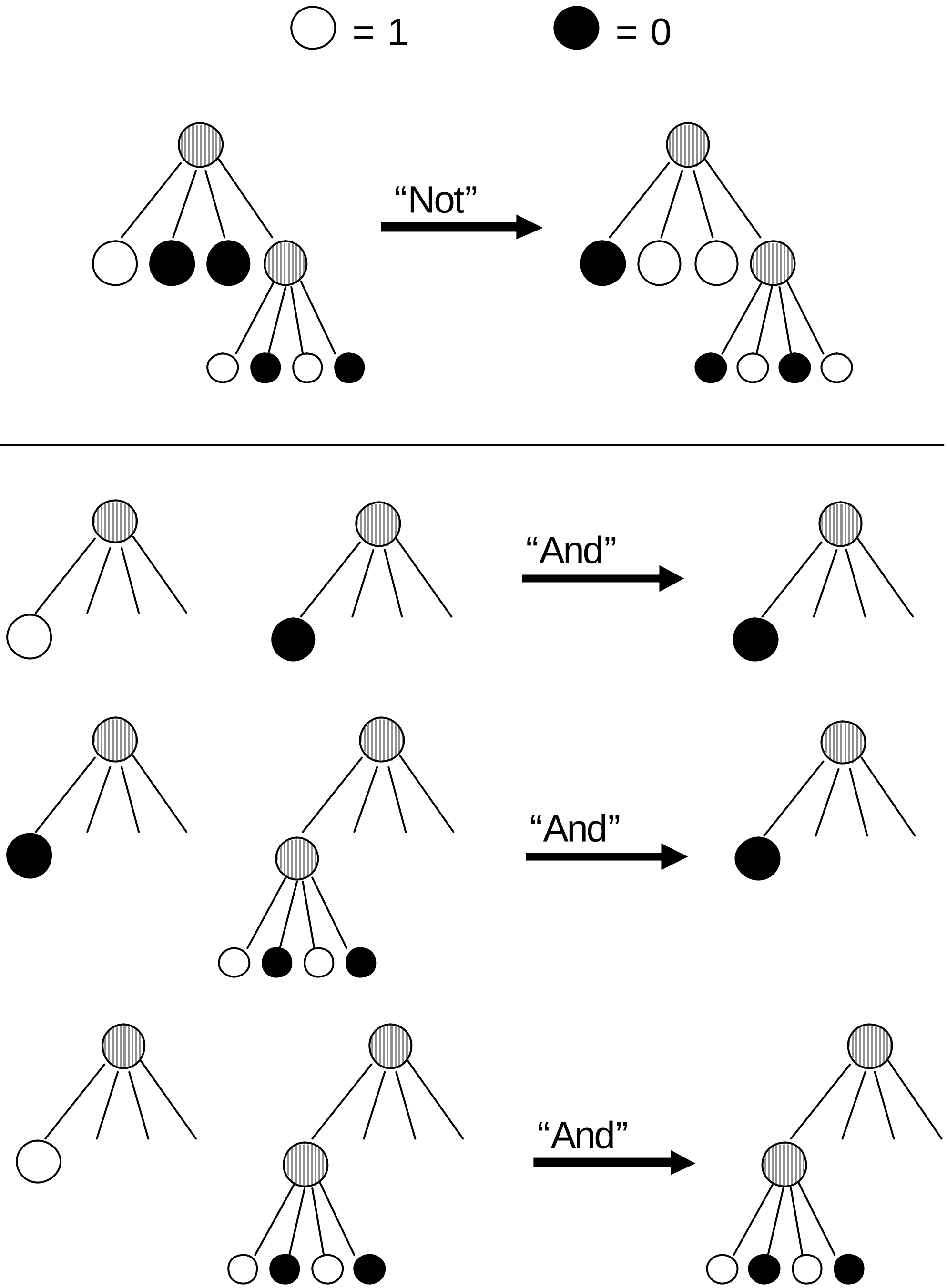
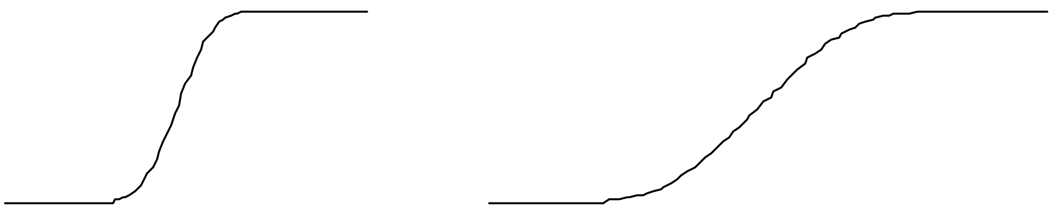


Image Pyramids

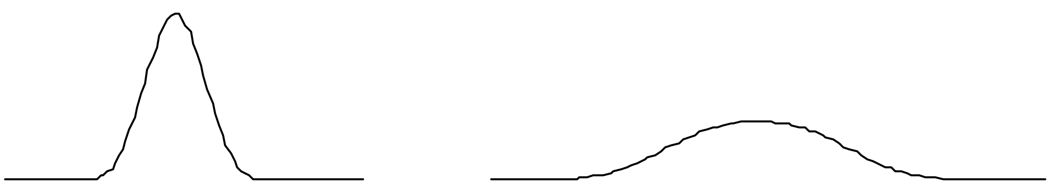
Image features at different scales require filters at different scales.

Edges (derivatives):

$f(x)$



$f'(x)$



Objects (correlation):

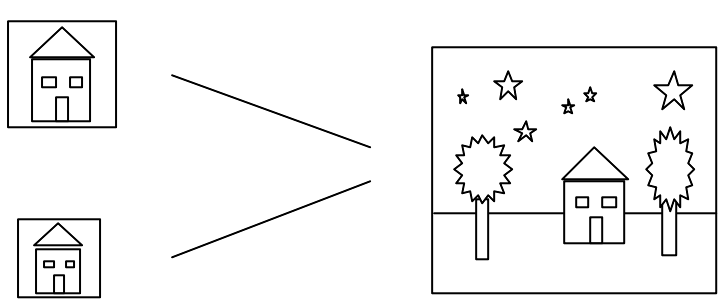


Image Pyramids

Image Pyramid = Hierarchical representation of an image

Image Pyramid = A collection of images at different resolutions.

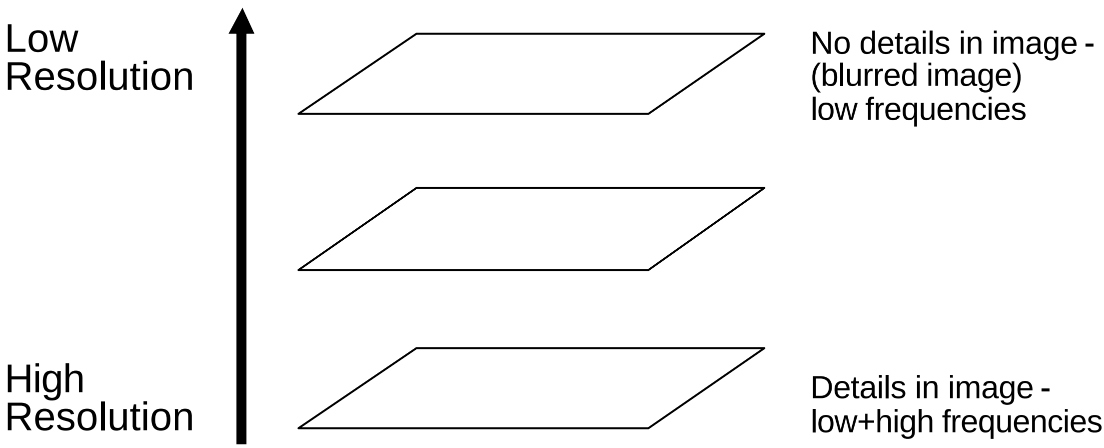
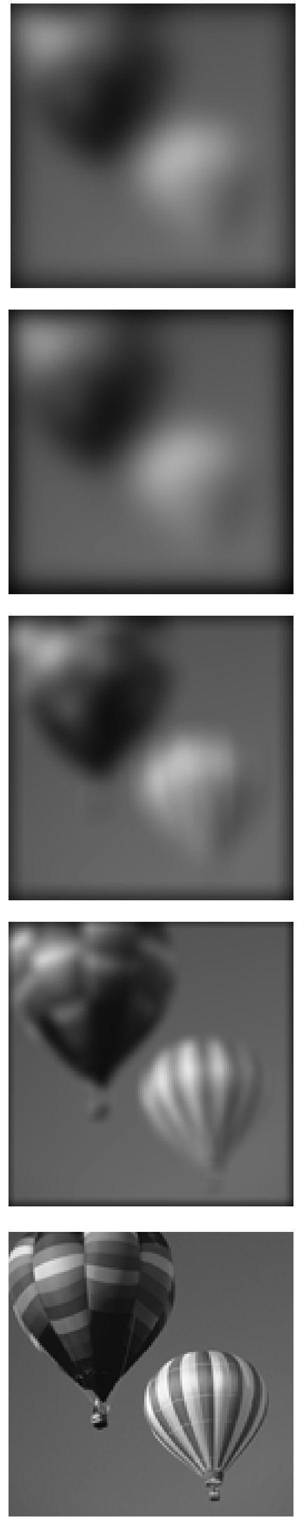


Image Pyramid

Low resolution



High resolution

Image Pyramid
Frequency Domain

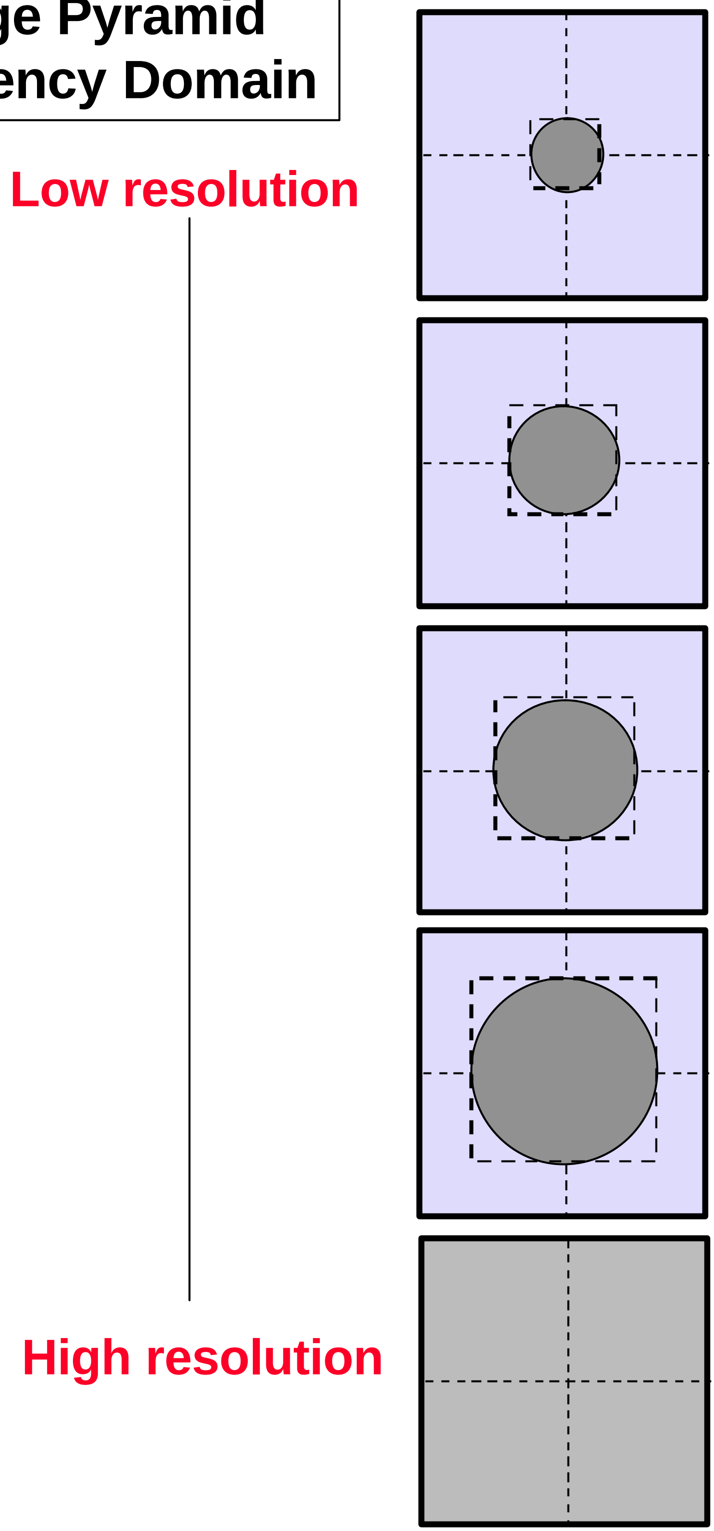
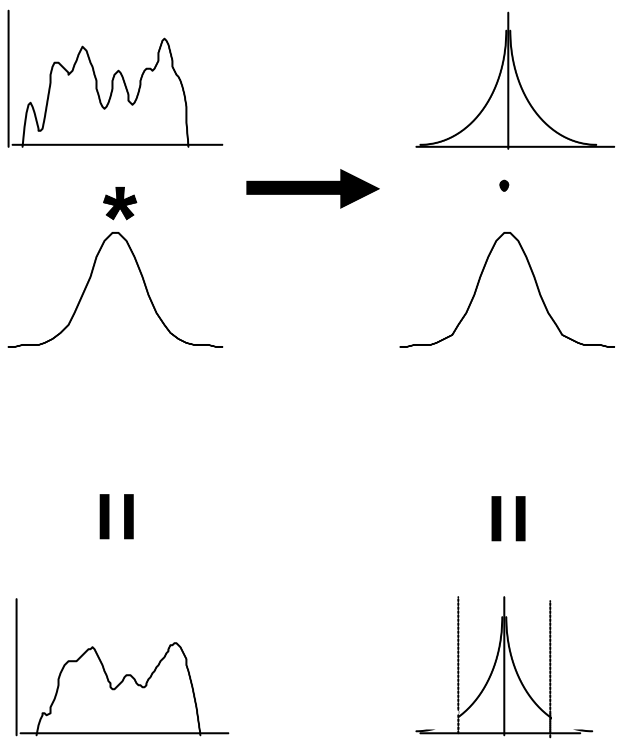


Image Blurring = low pass filtering



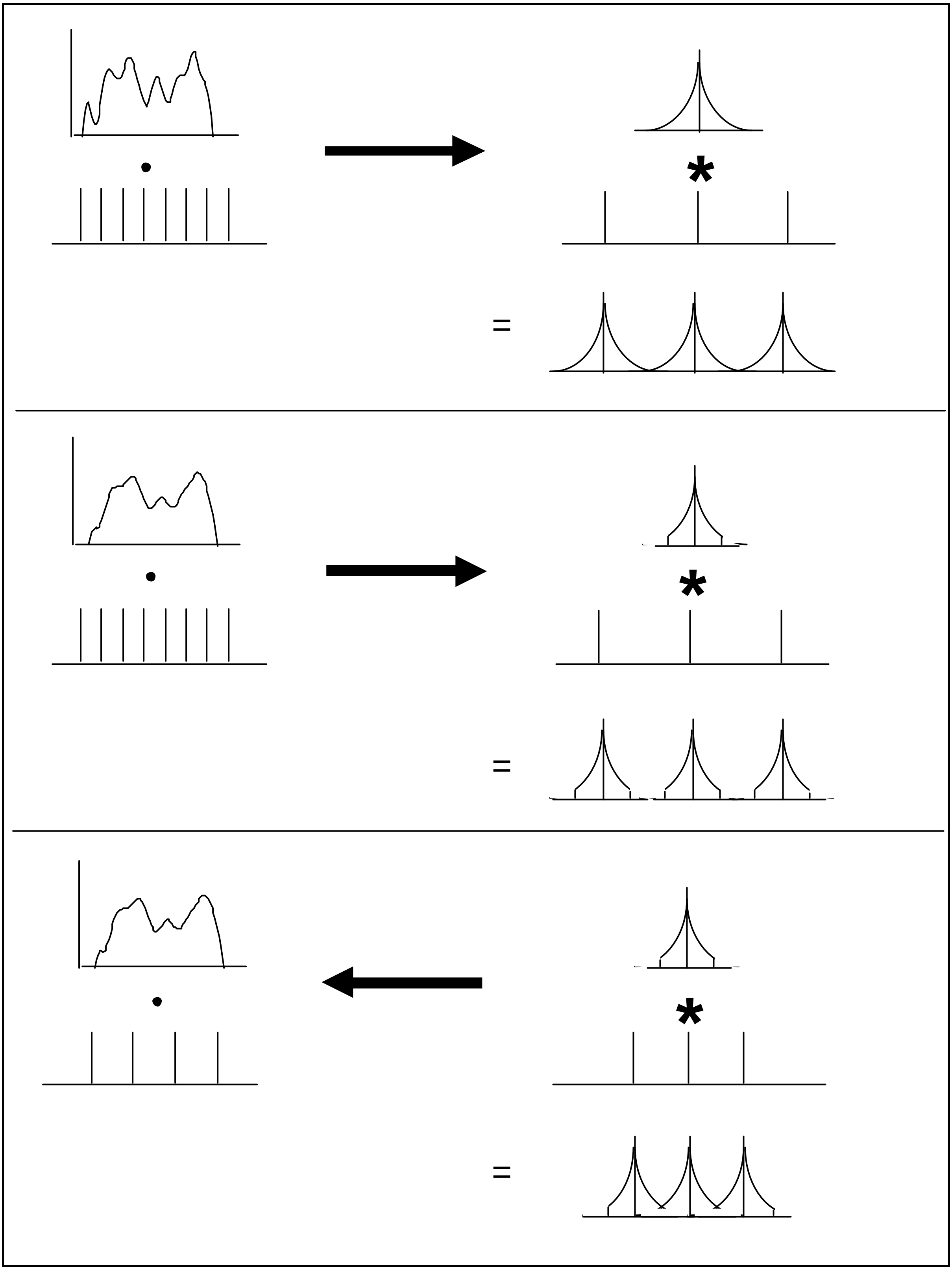
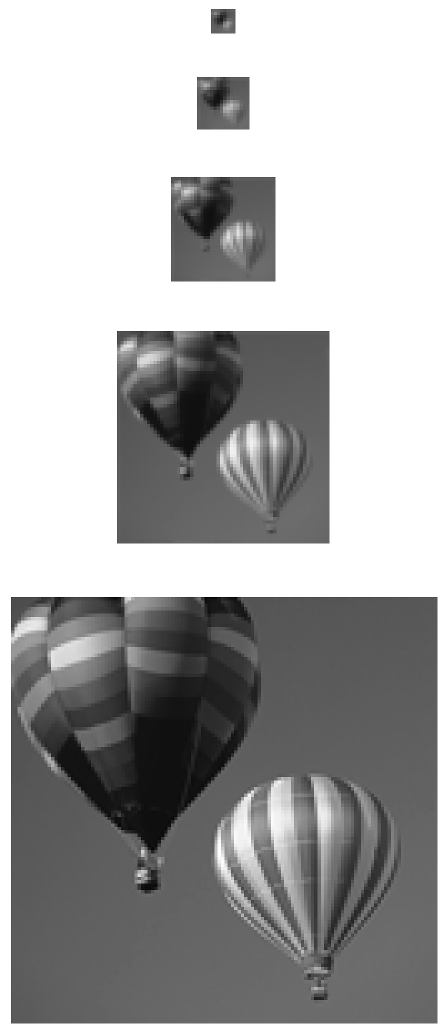


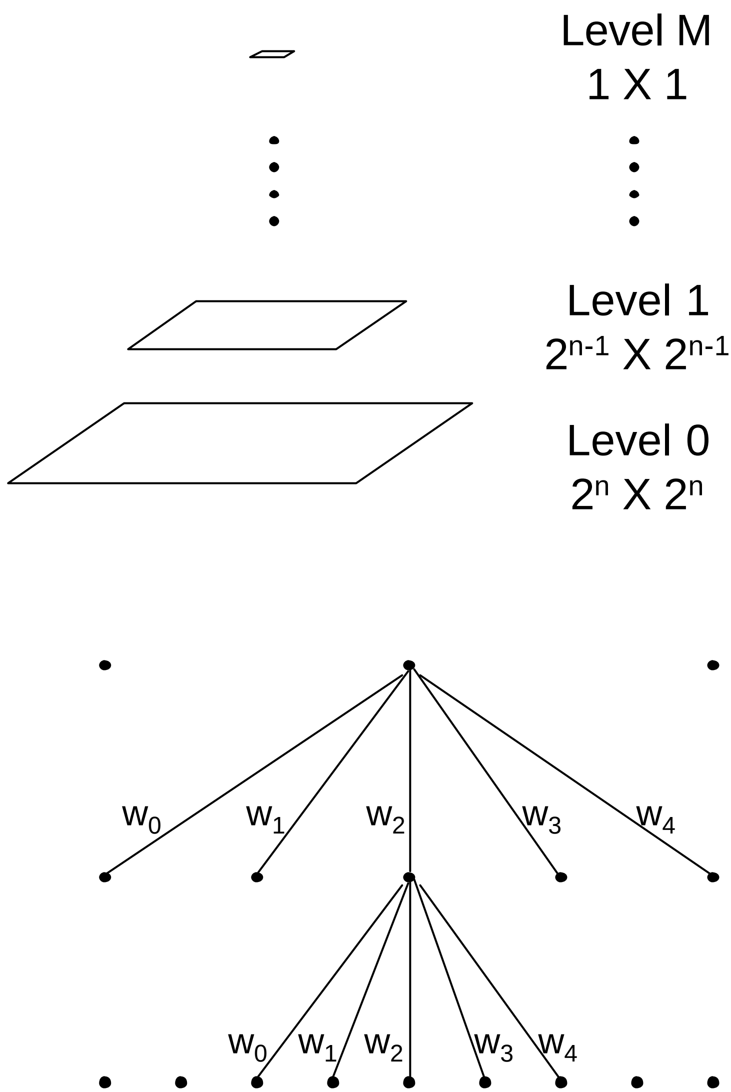
Image Pyramid

Low resolution



High resolution

Gaussian Pyramid



Gaussian Pyramid

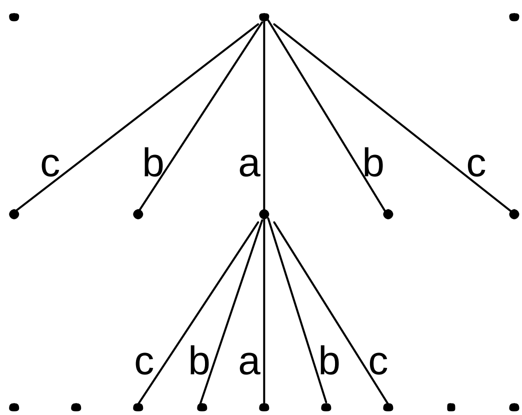
Burt & Adelson (1981)

Normalized: $\Sigma w_i = 1$

Symmetry: $w_i = w_{-i}$

Unimodal: $w_i \geq w_j$ for $0 < i < j$

Equal Contribution: for all j Σw_{j+2i} = constant



$a + 2b + 2c = 1$
$a + 2c = 2b$

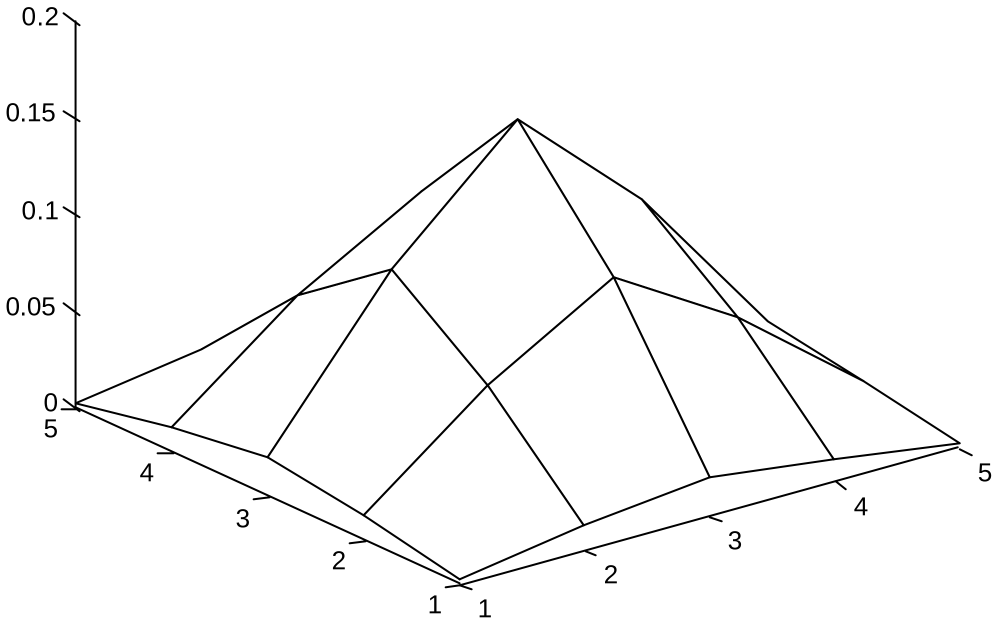
$a > 0.25$
$b = 0.25$
$c = 0.25 - a/2$

For a = 0.4 most similar to a Gauusian filter

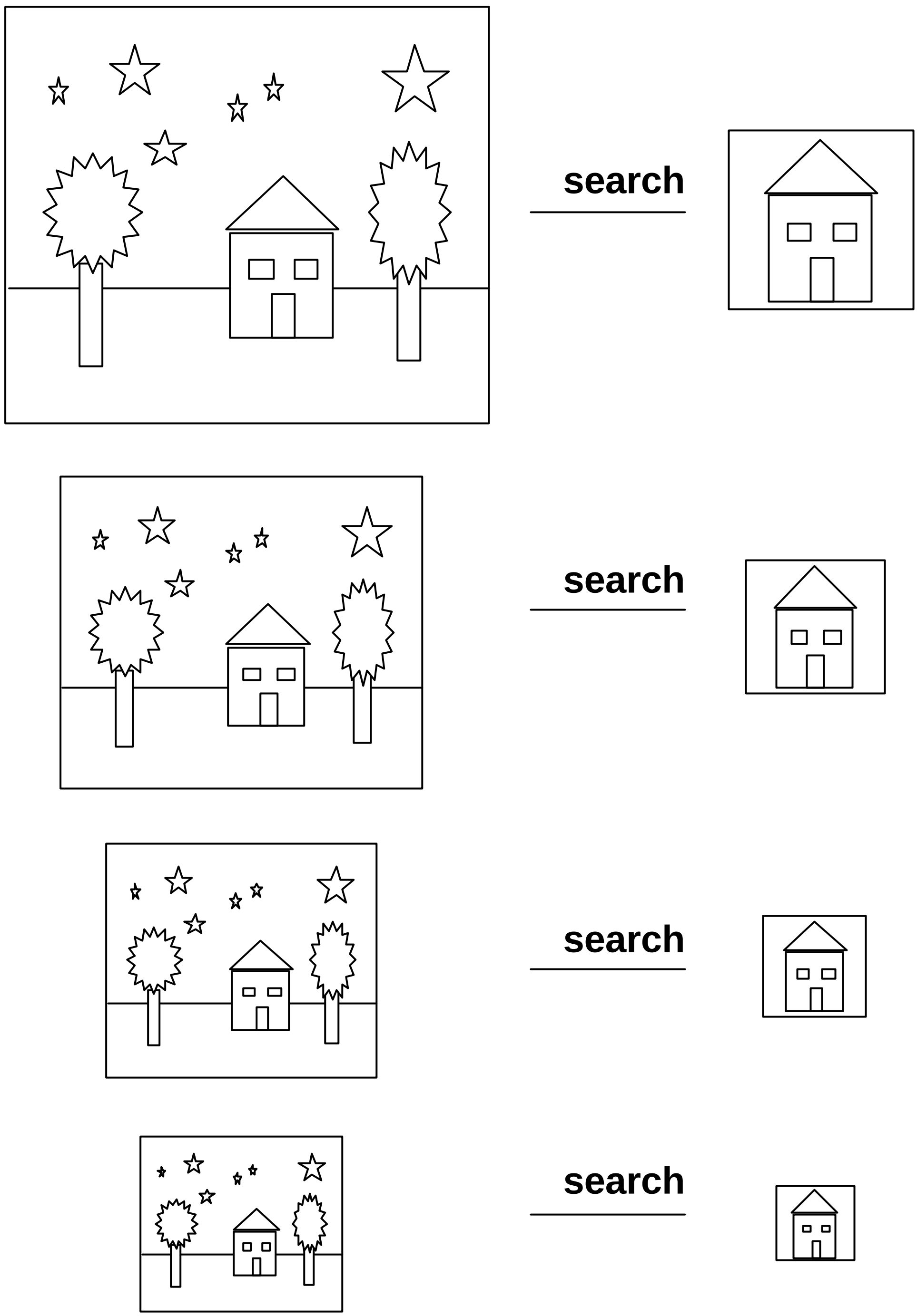
g = [0.05 0.25 0.4 0.25 0.05]

low_pass_filter = g * g' =

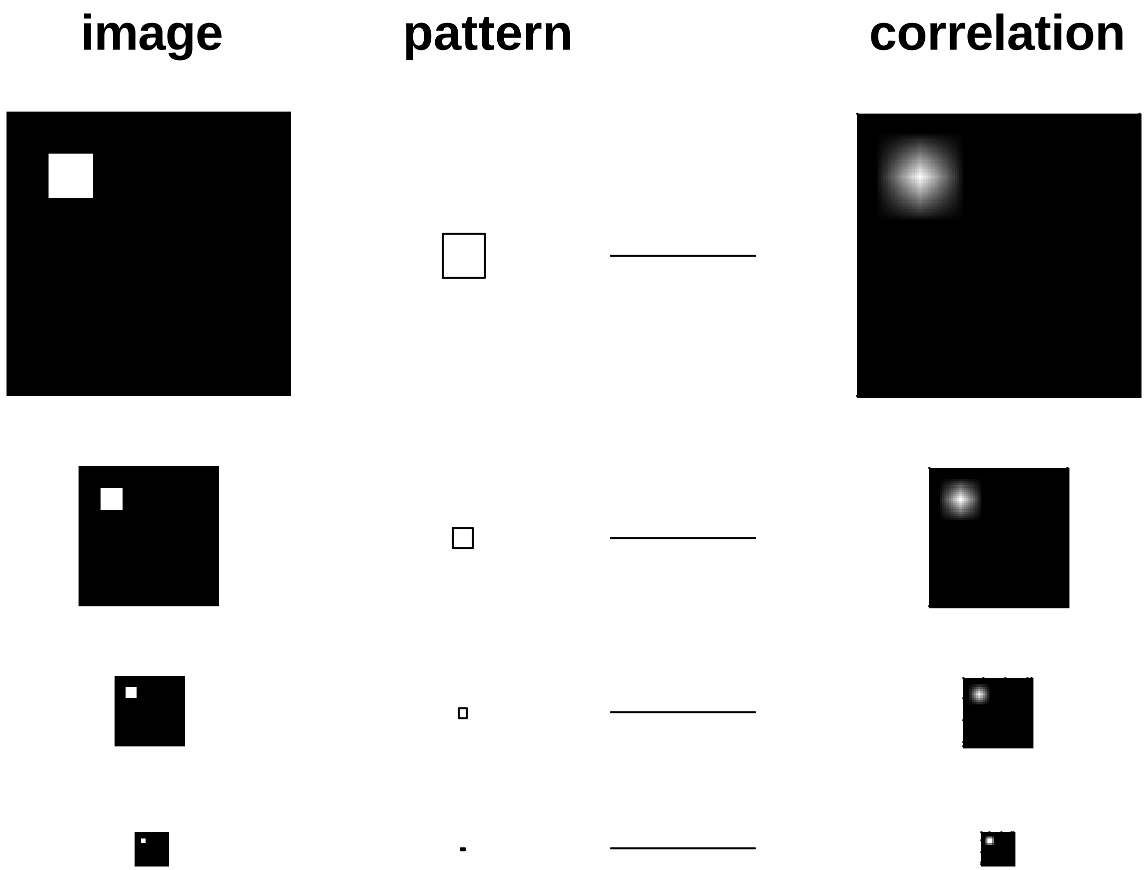
$$\begin{bmatrix} \mathbf{0.0025} & \mathbf{0.0125} & \mathbf{0.0200} & \mathbf{0.0125} & \mathbf{0.0025} \\ \mathbf{0.0125} & \mathbf{0.0625} & \mathbf{0.1000} & \mathbf{0.0625} & \mathbf{0.0125} \\ \mathbf{0.0200} & \mathbf{0.1000} & \mathbf{0.1600} & \mathbf{0.1000} & \mathbf{0.0200} \\ \mathbf{0.0125} & \mathbf{0.0625} & \mathbf{0.1000} & \mathbf{0.0625} & \mathbf{0.0125} \\ \mathbf{0.0025} & \mathbf{0.0125} & \mathbf{0.0200} & \mathbf{0.0125} & \mathbf{0.0025} \end{bmatrix}$$



Hierarchical Pattern Matching



Pattern matching using Pyramids - Example



Gaussian Pyramid - Computational Aspects

Memory:

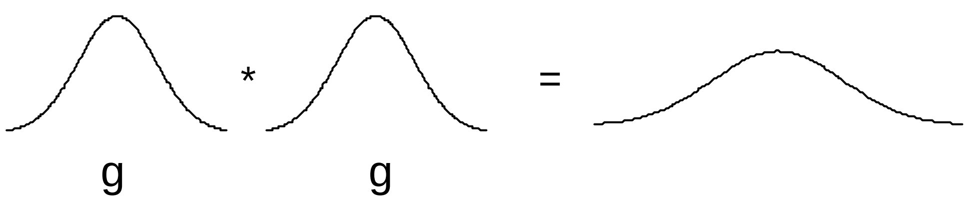
$$2^N X 2^N (1 + 1/4 + 1/16 + \dots) = 2^N X 2^N * 4/3$$

Computation:

Level i can be computed with a single convolution with filter: $h_i = \underbrace{g * g * g}_{\text{i times}} * \dots\dots$

Example:

h_2 =

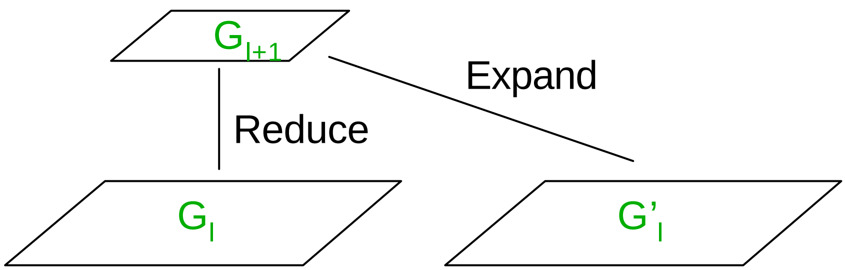


Laplacian Pyramid

Compression -
compression rates are higher for predictable values.
e.g. values around 0.

$G_0, G_1, \ldots$. = the levels of a Gaussian Pyramid.

Predict level G_l from level G_{l+1} by Expanding G_{l+1} to obtain G'_l

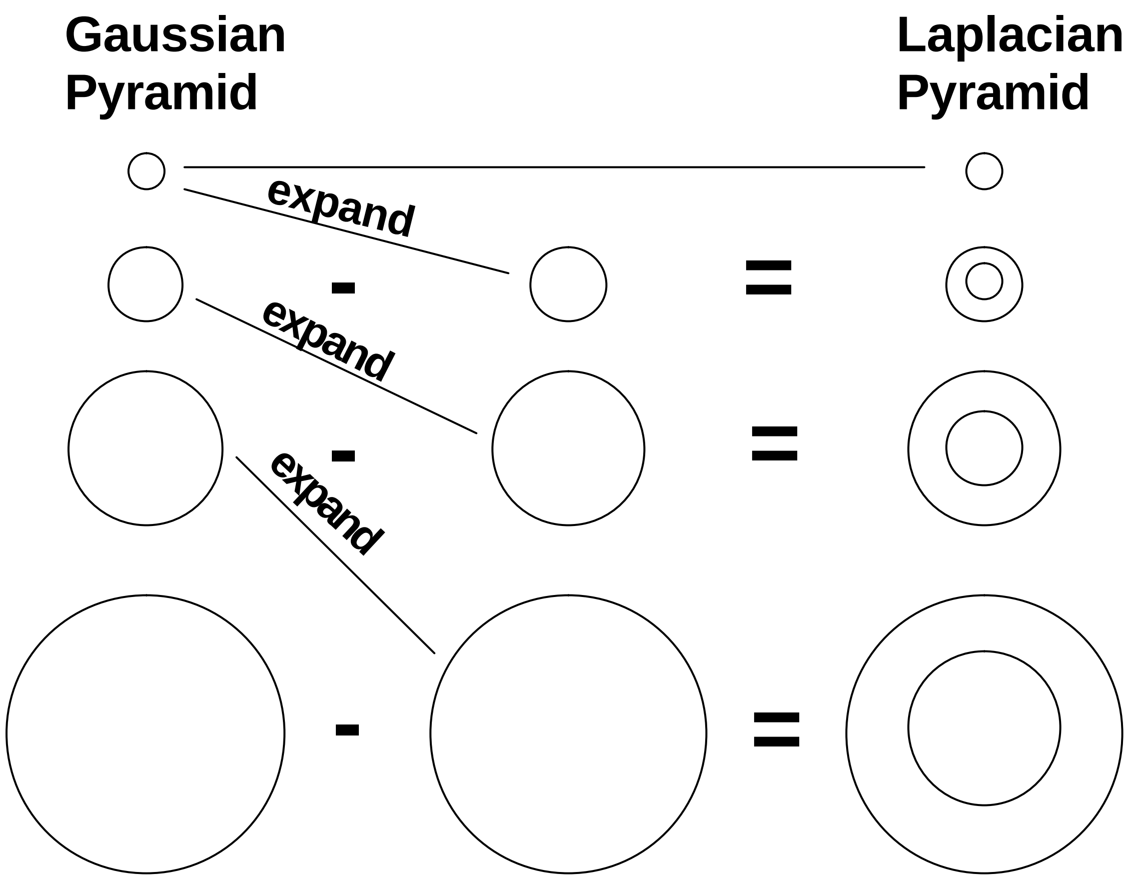


Denote by L_l the error in prediction:

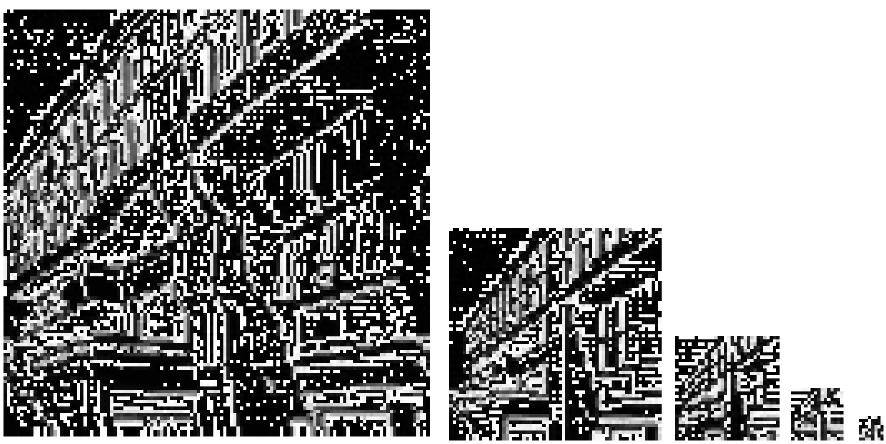
$$\boxed{L_l = G_l - G'_l}$$

$L_0, L_1, \ldots$. = the levels of a Laplacian Pyramid.

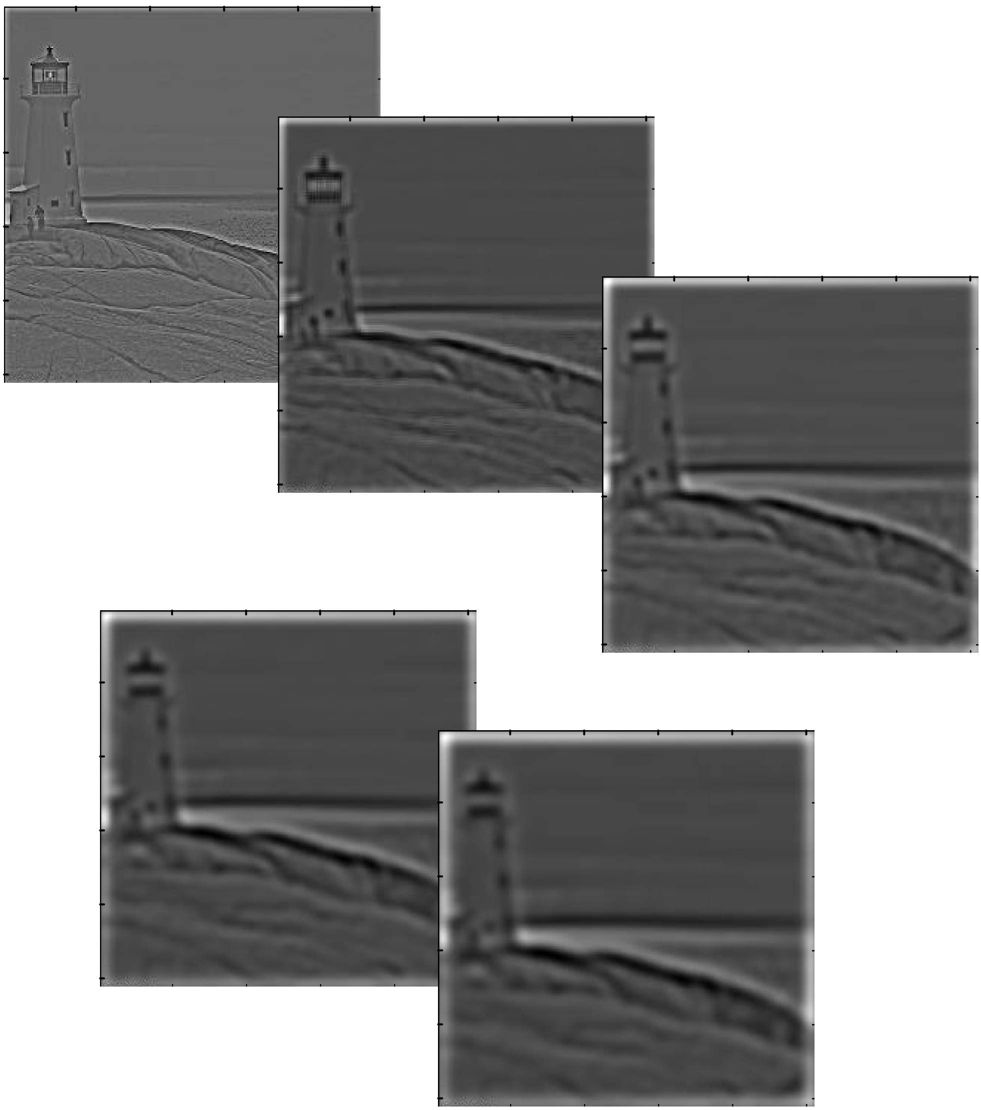
Laplacian Pyramid

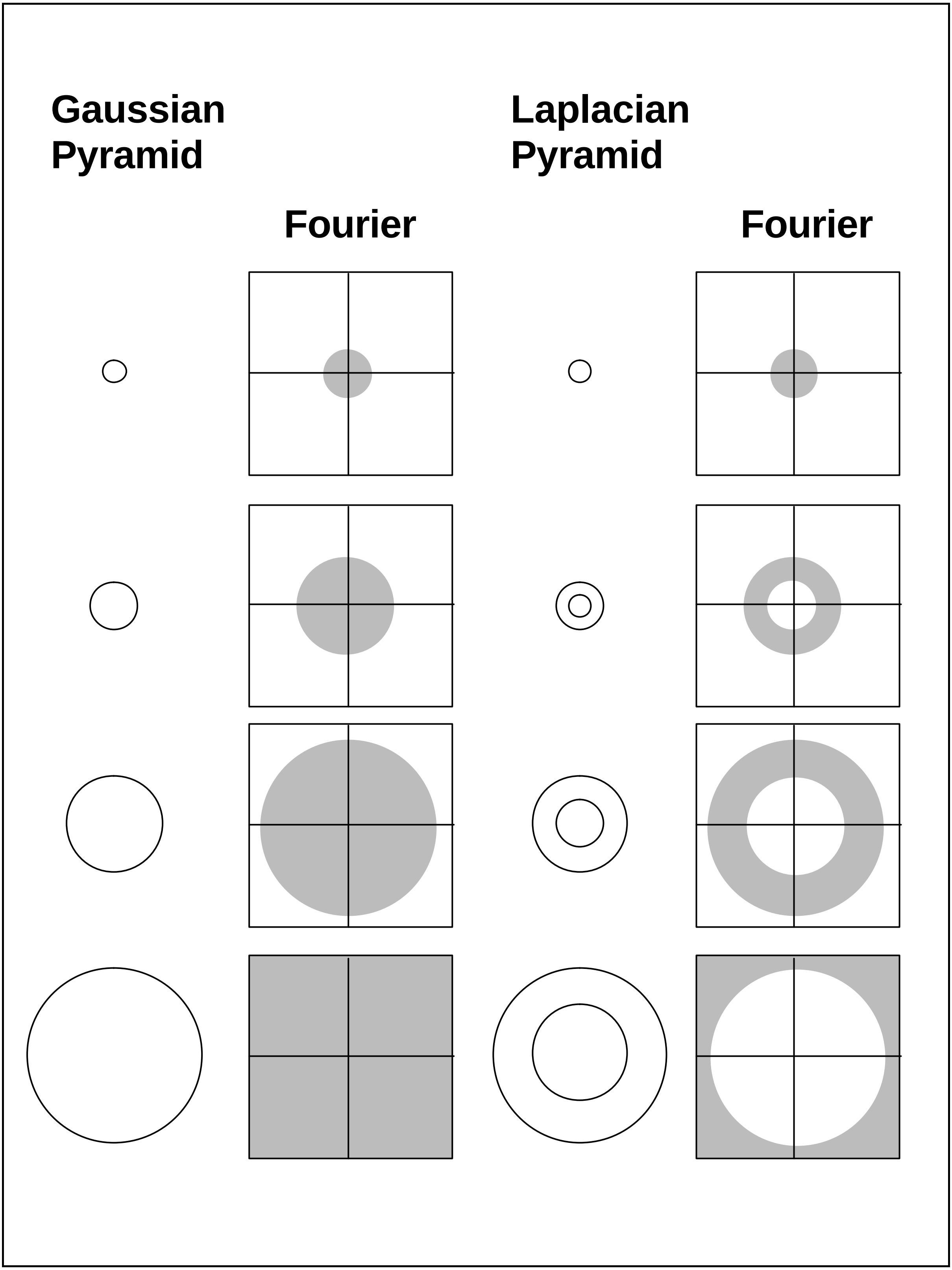


Laplacian Pyramid - Example

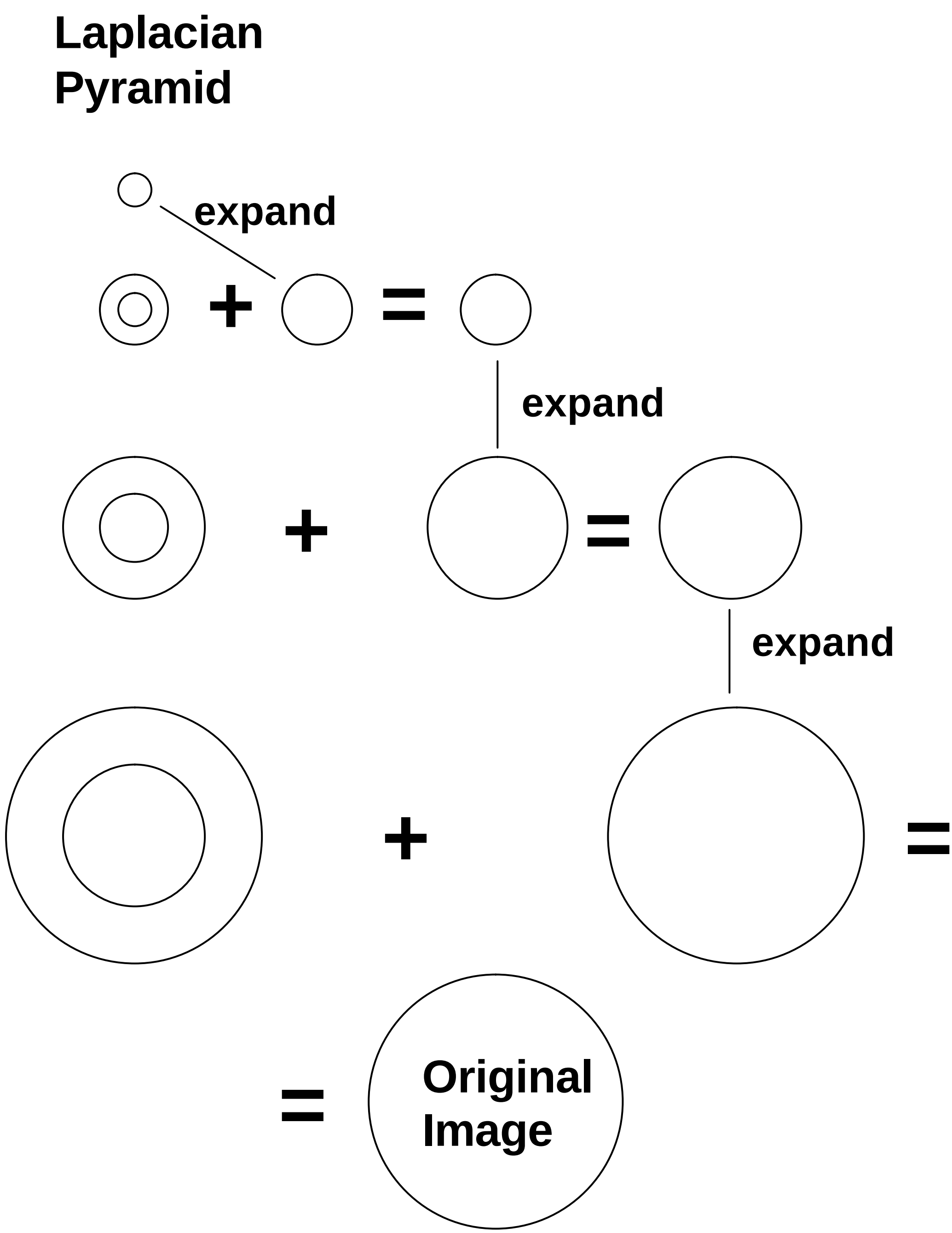


Laplacian Pyramid - No scaling




Gaussian Pyramid
Fourier
Laplacian Pyramid
Fourier

Reconstruction of the original image from the Laplacian Pyramid



Laplacian Pyramid - Computational Aspects

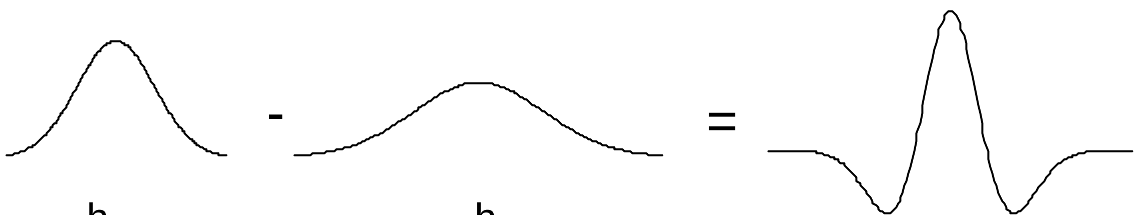
Memory:

$$2^N X 2^N (1 + 1/4 + 1/16 + \ldots) = 2^N X 2^N * 4/3$$

However coefficients are highly compressable.

Computation:

L_i can be computed from G_0 with a single convolution with filter: $k_i = h_{i-1} - h_i$

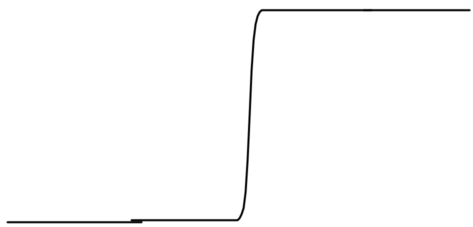


h_{i-1} h_i k_i

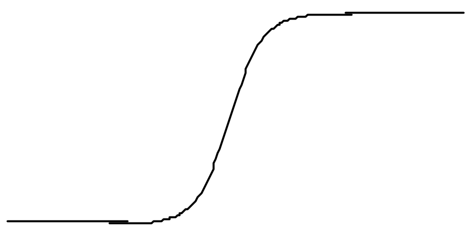
k_1 k_2 k_3

Multiresolution Spline

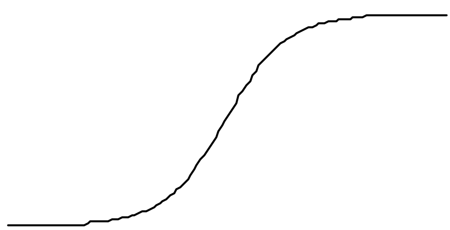
When splining two images, transition from one image to the other should behave:



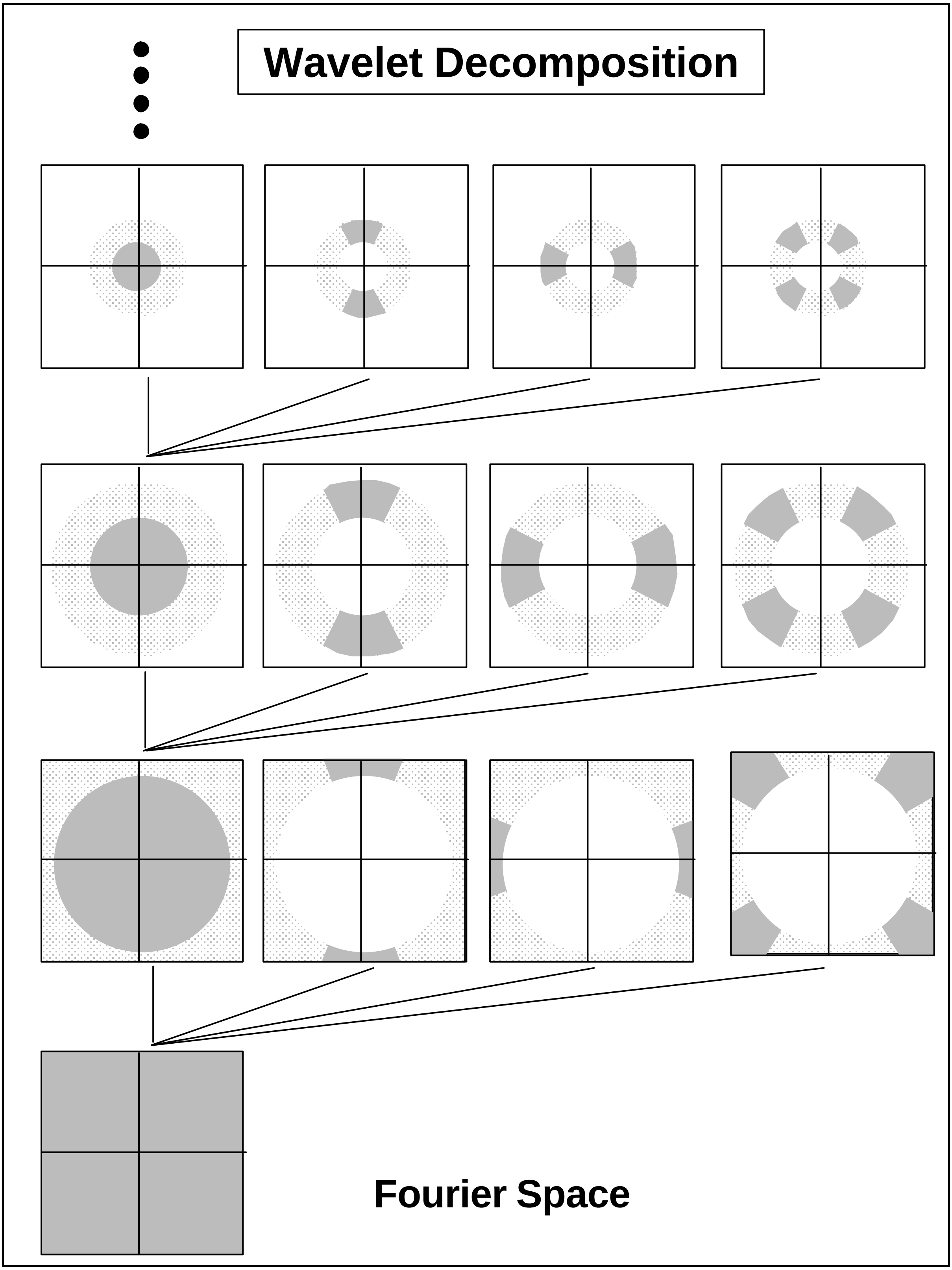
High Frequencies



Middle Frequencies

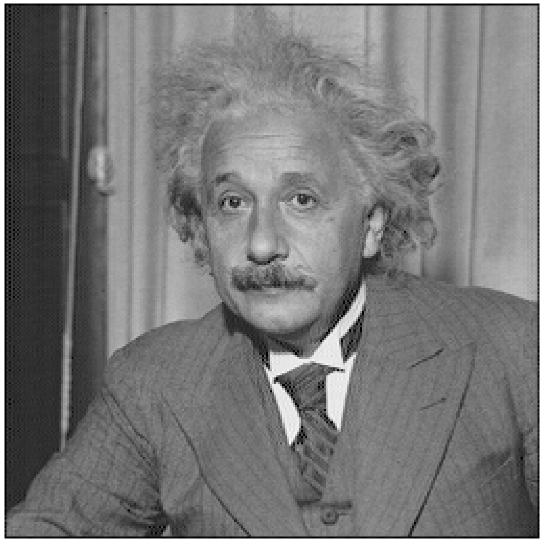


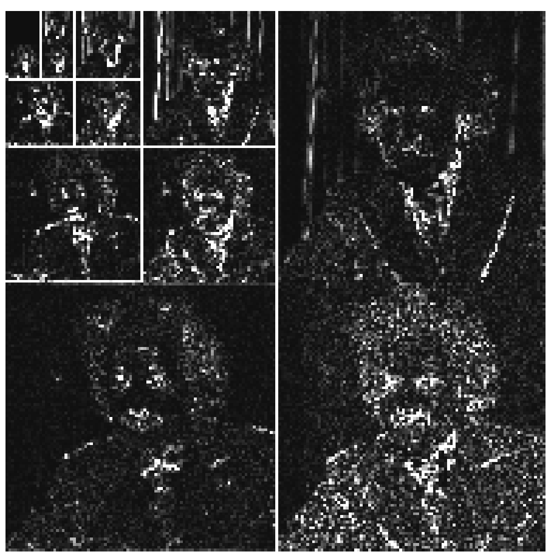
Low Frequencies



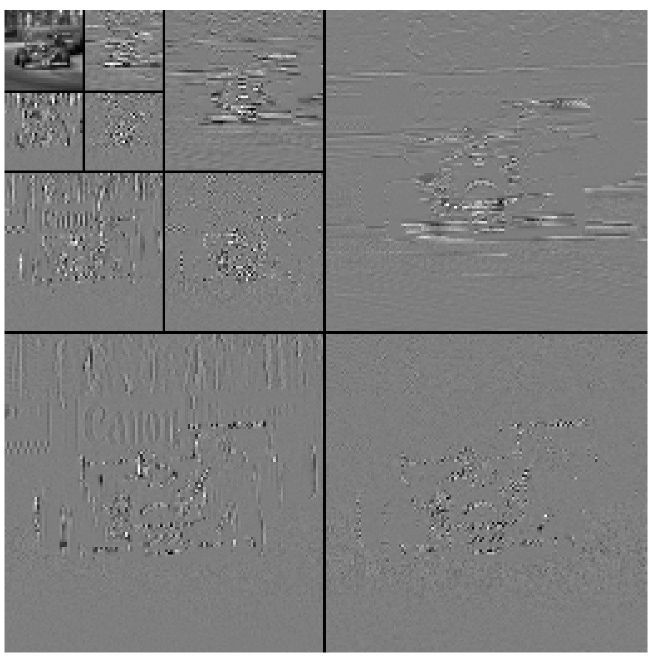
Wavelet Decomposition
Fourier Space

Wavelet Transform - Example





Wavelet Transform - Example



Wavelet Transform - Example

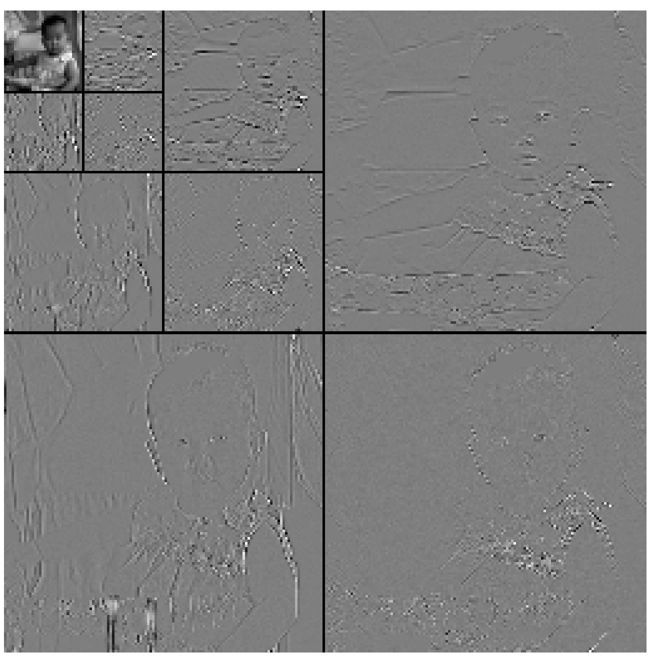


Image Pyramids - Comparison

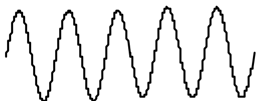
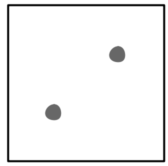
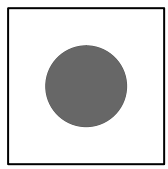
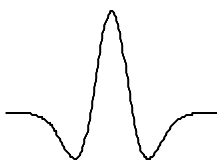
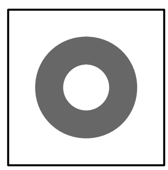
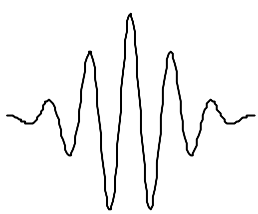
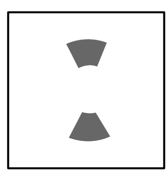
Transform	Basis	Frequency	Characteristics
Fourier	 Sines+Cosines		Not localized in space Localized in Frequency
Gaussian Pyramid	 Gaussian Filters		Localized in space Not localized in Frequency
Laplacian Pyramid	 Laplacian Filters		Localized in space Not localized in Frequency
Wavelet Pyramid	 Wavelet Filters		Localized in space Localized in Frequency

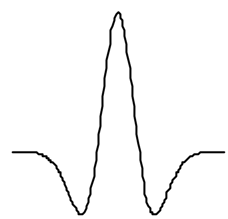
Image Pyramids - Comparison

Image pyramid levels = Filter then sample.

Filters:

Gaussian Pyramid

Laplacian Pyramid



Wavelet Pyramid