Image pyramids

Aykut Erdem
Dept. of Computer Engineering
Hacettepe University

Image Scaling

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

Image Sub-Sampling

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

Image Sub-Sampling

1/4

1/2

1/4 (2x zoom)

1/8 (4x zoom)
Good and Bad Sampling

Good sampling:
- Sample often or,
- Sample wisely

Bad sampling:
- Aliasing!

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
- Enter Monsieur Fourier...

- To avoid aliasing:
  - sampling rate $\geq 2 \times$ max frequency in the image
    - said another way: $\geq$ two samples per cycle
  - This minimum sampling rate is called the Nyquist rate

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

• How can we fix this?
Gaussian pre-filtering

• Solution: filter the image, then subsample

Subsampling with Gaussian pre-filtering

• Solution: filter the image, then subsample

Compare with...

Gaussian pre-filtering

• Solution: filter the image, then subsample
Image pyramids

Image information occurs at all spatial scales

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

The Gaussian pyramid

- Smooth with gaussians, because
  - a gaussian * gaussian = another gaussian
- Gaussians are low pass filters, so representation is redundant.

The computational advantage of pyramids

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.

[Institute, 1983]
The Gaussian Pyramid

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

\[ x_2 = G_1 x_1 \]

\[ G_1 = \begin{bmatrix} 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 \end{bmatrix} \]

\[ G_2 = \begin{bmatrix} 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 6 & 4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 4 & 6 & 4 \\ 0 & 0 & 0 & 0 & 0 & 1 & 4 & 6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \]

(Normalization constant of 1/16 omitted for visual clarity.)
The combined effect of the two pyramid levels

\[ x_3 = G_2 G_1 x_1 \]

\[ G_2 G_1 = \]

\[
\begin{array}{cccccccccccccccccccc}
1 & 4 & 10 & 20 & 31 & 40 & 44 & 40 & 31 & 20 & 10 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 4 & 10 & 20 & 31 & 40 & 44 & 40 & 31 & 20 & 10 & 4 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 4 & 10 & 20 & 31 & 40 & 44 & 40 & 30 & 16 & 4 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 10 & 20 & 25 & 16 & 4 & 0 & 0 & 0 & 0 \\
\end{array}
\]

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.
Image pyramids

Image information occurs at all spatial scales

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

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.

Upsampling

\[ y_2 = F_3 x_3 \]

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

\[ F_3 = \begin{bmatrix} 6 & 1 & 0 & 0 \\ 4 & 4 & 0 & 0 \\ 1 & 6 & 1 & 0 \\ 0 & 4 & 4 & 0 \\ 0 & 1 & 6 & 1 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 1 & 6 \\ 0 & 0 & 0 & 4 \end{bmatrix} \]
Showing, at full resolution, the information captured at each level of a Gaussian (top) and Laplacian (bottom) pyramid.

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:
$L_1 = (I - F_1 G_1) x_1$
$L_2 = (I - F_2 G_2) x_2$
$L_3 = (I - F_3 G_3) x_3$
$x_2 = G_1 x_1$
$x_3 = G_2 x_2$
$x_4 = G_3 x_3$

Reconstruction of original image ($x_1$) from Laplacian pyramid elements:
$x_3 = L_3 + F_3 x_4$
$x_2 = L_2 + F_2 x_3$
$x_1 = L_1 + F_1 x_2$
**Laplacian pyramid applications**

- Texture synthesis
- Image compression
- Noise removal

---

**Image blending**

(a)  
(b)
Image blending

- 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

Image pyramids

Image information occurs at all spatial scales

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

Linear transforms

\[ \mathbf{F} = \mathbf{Uf} \quad \Rightarrow \quad \mathbf{f} = \mathbf{U}^{-1}\mathbf{F} \]

Note: not all important transforms need to have an inverse
Linear transforms \[ \hat{F} = U \hat{f} \]

Digits

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

Pixels

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[ U = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \]

- No locality for reconstruction
- Needs boundary

Linear transforms

\[ \hat{F} = U \hat{f} \]

Derivative

\[
\begin{bmatrix}
1 & -1 & 0 & 0 \\
0 & 1 & -1 & 0 \\
0 & 0 & 1 & -1 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[ U = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \]

Integration

\[
\begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[ U^{-1} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \]

Haar transform

The simplest set of functions:

\[ U = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \]

\[ U^{-1} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \]
The simplest set of functions:
\[ U = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \]
\[ U^T = \begin{bmatrix} 0.5 & 0.5 \\ -0.5 & -0.5 \end{bmatrix} \]

To code a signal, repeat at several locations:

The entire process can be written as a single matrix:

\[ \tilde{F} = U \tilde{f} \]

Properties:
- Orthogonal decomposition
- Perfect reconstruction
- Critically sampled
2D Haar transform

Basic elements:

\[
\begin{bmatrix}
1 & 1 \\
1 & -1
\end{bmatrix}
\]

Low pass

\[
\frac{1}{2}
\begin{bmatrix}
1 & 1 \\
1 & 1
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 1 \\
1 & -1
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 1 \\
1 & -1
\end{bmatrix}
\]
2D Haar transform

Basic elements:

\[
\begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 \\
1 & 1 & -1 & -1 \\
-1 & -1 & -1 & 1 \\
\end{bmatrix}
\]

Low pass

High pass vertical

High pass horizontal

High pass diagonal

Pyramid cascade

Wavelet/QMF representation

Sketch of the Fourier transform

Same number of pixels!
Image pyramids

Image information occurs at all spatial scales

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

Steerable Pyramid

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

Decomposition  Reconstruction

Steerable Pyramid

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

Decomposition  Reconstruction
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 low-pass subband). For example, the shaded region corresponds to the spectral support of a single (vertically-oriented) subband.


There is also a high pass residual...

Filter Kernels

Image

Coarsest scale

Finest scale

Dog or cat?

Almost no dog information

Summary of pyramid representations

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
  - Bandpass 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.
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.

Fourier transform, or Wavelet transform, or Steerable pyramid transform

\[ \vec{F} = \vec{U} \vec{f} \]

Transformed image

Vectorized image

Gaussian pyramid

\[ = \]

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

Pixel image

Laplacian pyramid

\[ = \]

Overcomplete representation. Transformed pixels represent bandpassed image information.

Pixel image

Wavelet (QMF) transform

\[ = \]

Ortho-normal transform (like Fourier transform), but with localized basis functions.

Pixel image

Slide credit: B. Freeman and A. Torralba
Steerable pyramid

Multiple orientations at one scale

Multiple orientations at the next scale

Over-complete representation, but non-aliased subbands.

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.