Texture Analysis and Synthesis
Texture

- Texture: pattern that "looks the same" at all locations
- May be structured or random

[Wei & Levoy]
Applications of Textures

- Texture analysis
  - Determining statistical properties of textures
  - Segmentation
  - Recognition
  - Shape from texture
- Texture synthesis
Image Pyramids

Idea: Represent N x N image as a “pyramid” of 1x1, 2x2, 4x4, ..., 2^k x 2^k images (assuming N = 2^k)

level k (= 1 pixel)

level k-1

level k-2

... level 0 (= original image)
Pyramid Creation

- “Gaussian” Pyramid
- “Laplacian” Pyramid
  - Created from Gaussian pyramid by subtraction
    \[ L_i = G_i - \text{expand}(G_{i+1}) \]
Octaves in the Spatial Domain

Lowpass Images

Bandpass Images

Szeliski
Oriented Filter Banks

Multiresolution Oriented Filter Bank

Original Image

Steerable Pyramid
Steerable Pyramid Texture Analysis

- Pass image through filter bank
- Compile histogram of intensities output by each filter
- To synthesize new texture:
  - Start with random noise image
  - Adjust histograms to match original image
  - Re-synthesize image from filter outputs
Histogram Equalization

- **Given:** two histograms of intensity $H_1$ and $H_2$

- **Goal:** function that remaps intensities to make new histogram $H_1'$ equal $H_2$
Histogram Equalization

1. Compute CDFs of histograms

2. For each intensity, map through CDF 1 then look up inverse in CDF 2
Texture Analysis / Synthesis

Original Texture

Synthesized Texture

Heeger and Bergen
Textons

- Elements ("textons") either identical or come from some statistical distribution
- Can analyze in natural images

Olhausen & Field
Clustering Textons

- Output of bank of $n$ filters can be thought of as vector in $n$-dimensional space
- Can cluster these vectors using $k$-means [Malik et al.]
- Result: dictionary of most common textures
Clustering Textons

Image

Clustered Textons

Texton to Pixel Mapping

[Malik et al.]
Using Texture in Segmentation

- Compute histogram of how many times each of the $k$ clusters occurs in a neighborhood
- Define similarity of histograms $h_i$ and $h_j$ using $\chi^2$

$$\chi^2 = \frac{1}{2} \sum_k \frac{(h_i(k) - h_j(k))^2}{h_i(k) + h_j(k)}$$

- Different histograms $\rightarrow$ separate regions
Texture Segmentation

[Malik et al.]
Markov Random Fields

- Different way of thinking about textures
- Premise: probability distribution of a pixel depends on values of neighbors
- Probability the same throughout image
  - Extension of Markov chains
Texture Synthesis Based on MRF

- For each pixel in destination:
  - Take already-synthesized neighbors
  - Find closest match in original texture
  - Copy pixel to destination

- Efros & Leung 1999,
  speedup by Wei & Levoy 2000

- Extension to copying whole blocks
  by Efros & Freeman 2001
  - Let’s look at their talk…