Analysis of Smoothly Varying Textures
with Applications to Segmentation

Jason Chang

CSAIL, MIT
jchang7@csail.mit.edu

June 2, 2010
Outline

1. Introduction
   - Problem Statement

2. Texture Modelling
   - Motivation
   - Feature Extraction
   - Boundary Effects
   - Spatial Smoothness

3. Applications
   - Segmentation
   - Gamma Estimation
   - Shading Reflectance
Outline

1. Introduction
   - Problem Statement

2. Texture Modelling
   - Motivation
   - Feature Extraction
   - Boundary Effects
   - Spatial Smoothness

3. Applications
   - Segmentation
   - Gamma Estimation
   - Shading Reflectance
Image Segmentation

- Separate an image into disjoint regions
- Assume regions have common statistical properties
Simple Images with i.i.d. Pixel Intensities [6]

**When are results good?**

Minimal spatial dependencies conditioned on the label

- [movies/mit-unimodal.mp4](movies/mit-unimodal.mp4)
- [movies/mit-bimodal.mp4](movies/mit-bimodal.mp4)

**When does this approximation fail?**

Strong spatial structures within regions

- [movies/t-orientation-scalar.mp4](movies/t-orientation-scalar.mp4)
Textured Image Segmentation

- Ill-posed nature - Segment the stripes or the zebra

- Must consider local neighborhoods instead of pixels
Outline

1. Introduction
   - Problem Statement

2. Texture Modelling
   - Motivation
   - Feature Extraction
   - Boundary Effects
   - Spatial Smoothness

3. Applications
   - Segmentation
   - Gamma Estimation
   - Shading Reflectance
A Motivating Example

Orientation
Scale
Contrast
Bias

A Motivating Example

Orientation
Scale
Contrast
Bias

A Motivating Example

Orientation
Scale
Contrast
Bias

A Motivating Example

Orientation
Scale
Contrast
Bias

A Motivating Example

Orientation
Scale
Contrast
Bias
Modelling Goals

Approach

Find the best scale to represent a texture at each pixel

- Measure a notion of contrast, bias, and orientation at that scale
- Features should be not vary much within a constant texture
- Estimate spatial dependencies in features
Outline

1. Introduction
   - Problem Statement

2. Texture Modelling
   - Motivation
   - Feature Extraction
   - Boundary Effects
   - Spatial Smoothness

3. Applications
   - Segmentation
   - Gamma Estimation
   - Shading Reflectance
Steerable Pyramids [8]
Steerable Pyramids [8]
Steering the Filters [8]

**Theorem**

*With bounded error, the output at any orientation can be computed from a linear combination of the basis*

\[ y_i^s (\theta) = \sum_{\phi \in \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}} b_\phi (\theta) y_i^s (\phi) \]

\[ b_\phi (\theta) = \frac{\cos (\theta - \phi) + \cos (3 (\theta - \phi))}{2} \]
Feature Extraction - Local Energy

Observation

Filter outputs go positive and negative, and vary in between modes. To get a location independent measure, consider the local energy of the filter output.

Feature Extraction - Local Energy

Filter outputs go positive and negative, and vary in between modes. To get a location independent measure, consider the local energy of the filter output.
Feature Extraction

For each pixel, $i$, we consider a neighborhood around it, $R_i$

Analyzing Angular Energy

In scale, $s$, we consider the following local angular energy:

$$E_i^s(\theta) = \frac{1}{|R_i^s|} \sum_{j \in R_i^s} y_j^s(\theta)^2$$
Local Angular Energy

movies/energy-sweep.mp4
Feature Extraction

Feature Set

Scale
\( \eta_i = \arg \max_s \max_\theta E^s_i(\theta) \)

Orientation
\( \theta_i = \arg \max_\theta E^\eta_i(\theta) \)

Contrast Energy
\( E_i = E^\eta_i(\theta_i^\eta_i) \)

Residual Energy
\( \epsilon_i = E^\eta_i(\theta_i + \pi/2) \)

Bias
\( \mu_i = \frac{1}{|R^\eta_i|} \sum_{j \in R^\eta_i} x_j \)
Visualizing Features
Boundary Effects

Object boundaries affect features in two ways:

1. Filtered outputs are corrupted near boundaries
2. Local neighborhood, $R_i$, in angular energy can span boundaries
Object boundaries affect features in two ways:

1. **Filtered outputs** are corrupted near boundaries
2. Local neighborhood, $R_i$, in **angular energy** can span boundaries
Corrupted Filter Outputs

- Similar to image boundary effects, zero padding regions creates artifacts
- Conditioned on a segmentation, we reflect each region across the object boundary and re-filter the image
Local Region in Angular Energy

\[ E_i^s (\theta) = \frac{1}{|R_i^s|} \sum_{j \in R_i^s} y_j^s (\theta)^2 \]

Instead of using \( R_i \), we use \( R_i' = R_i \cap R^\pm \)
Boundary Effects

- We call these steps the border refinement step.
- The refinement step is computationally expensive and creates many more local extrema.
- We first segment an image without refinement (until convergence) and then perform refinement.
We want to capture smooth changes in the features

Model feature as output of intrinsic feature (*) subject to smooth, additive Markov random field

Intrinsic feature distributions are estimated non-parametrically (using a kernel density estimate[7])

\[
\theta = \tilde{\phi} + \theta^* \\
\eta = \tilde{\nu} + \eta^* \\
x = g \circ (b + R) \\
\Rightarrow \log E = \log g + \log E^* \\
\Rightarrow \mu = g \circ (b + \mu^*)
\]
MRF Estimation

- Perform MAP estimation of the smooth fields:
  $$\tilde{\phi} = \arg \max_{\phi} p(\phi|\theta)$$

- Using Bayes rule and differentiating:
  $$\tilde{\phi}^{(k+1)} = F^{-1} \left( \theta - w^\theta_p \left( \theta - \tilde{\phi}^{(k)} \right) \right)$$

  $$w^\theta_p (\cdot) = \frac{\sum_s \left( \theta_s - \tilde{\phi}_s \right) K \left( \cdot - \theta_s + \tilde{\phi}_s \right)}{\sum_s K \left( \cdot - \theta_s + \tilde{\phi}_s \right)}$$

  $$F = \left( \frac{2}{h^2} \Lambda_\phi \right)^{-1} + I$$
MRF Estimation

- We show here that $F$ performs highpass filtering with unity DC gain
  \[ F = \left( \frac{2}{h^2 \Lambda_{\phi}} \right)^{-1} + I \]

- Treat $\Lambda_{\phi}$ as lowpass filtered i.i.d. noise
  \[ F = \left( \frac{2}{h^2 L \sigma_{\phi}^2 I L^T} \right)^{-1} + I = \frac{h^2}{2 \sigma_{\phi}^2} H^T H + I \]

- The lowpass filter operator, $L$, has unity DC gain. $H$ much be a highpass filter operator with unity DC gain.
- Assuming $h \ll \sigma_{\phi}$, $F_1$ has a DC gain close to zero
MRF Estimation

\[ F = F_1 + I \]

Equivalent Fourier Operation of \( F_1 \)

Equivalent Fourier Operation of \( I \)

Equivalent Fourier Operation of \( F \)

Equivalent Fourier Operation of \( F^{-1} \)
Feature-Specific Considerations

- Orientation is periodic
- Gain field \((g)\) assumed to be smooth in log domain
- Gain and bias fields coupled in bias feature
  - Hard to estimate jointly
  - Estimate \(g\) via \(E\) and treat as point estimate in \(\mu\)

\[
\frac{\mu}{g} = b + \mu^*
\]
Once the smooth fields are estimated, we remove their effects to obtain the intrinsic features.
Introduction

Problem Statement

Texture Modelling

Motivation

Feature Extraction

Boundary Effects

Spatial Smoothness

Applications

Segmentation

Gamma Estimation

Shading Reflectance
Image Segmentation

- Maximize $I(E, \mu, \theta, \eta; L)$ [6]
- Treat features as independent
- Estimate distributions using kernel density estimate

## Algorithm Overview

1. Segment an image without regard to object boundaries
2. Refine the segmentation considering boundary effects
3. Estimate smooth fields conditioned on segmentation
4. Repeat Steps 1-3 until convergence
Our features are invariant to smooth changes, but able to distinguish abrupt changes.
Image Segmentation Examples

|----------|------------|-----|-----|-----|

- Analysis of Smoothly Varying Textures
- Jason Chang
- Introduction
- Problem Statement
- Texture Modelling
- Motivation
- Feature Extraction
- Boundary Effects
- Spatial Smoothness
- Applications
- Segmentation
- Gamma Estimation
- Shading Reflectance
- Contributions
Our appearance model does not always hold in natural images. Consider the following examples.
Introduction

Problem Statement

Texture Modelling

Motivation

Feature Extraction

Boundary Effects

Spatial Smoothness

Applications

Segmentation

Gamma Estimation

Shading Reflectance
Cameras typically have a nonlinear intensity response
Linear imaging is desired for some common computer vision tasks

If there was no $\gamma$ correction (i.e. $\gamma = 1$), the models are equivalent when $b = 0$ and $S = g$. 
Gamma Estimation

Algorithm Overview

- Given a $\gamma$, estimate $g(\gamma)$, $b(\gamma)$, $R(\gamma)$, image:
  \[
  R(\gamma) = \frac{x^{1/\gamma}}{g(\gamma)} - b(\gamma)
  \]
- Reconstruct the image without a bias field
  \[
  \hat{x} = (R(\gamma) \cdot g(\gamma))^{\gamma}
  \]
- Find the $\gamma$ that minimizes reconstruction error using golden section search [5]
  \[
  \gamma^* = \arg \min_{\gamma} \|\hat{x} - x\|_1
  \]
Gamma Estimation Results

Various Scenes

Generating the Data
Each scene was photographed using linear imaging ($\gamma = 1$) and post-processed with nine different $\gamma$ values. One set of $\gamma$ values is shown below.
## Gamma Estimation Results

<table>
<thead>
<tr>
<th>Image</th>
<th>Our RMSE</th>
<th>RMSE of [2]</th>
<th>Our $|\hat{\gamma} - \gamma|_1$</th>
<th>$|\hat{\gamma} - \gamma|_1$ of [2]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bookshelf</td>
<td>3.673</td>
<td>46.640</td>
<td>0.025</td>
<td>0.431</td>
</tr>
<tr>
<td>Glass Ceiling</td>
<td>4.771</td>
<td>37.468</td>
<td>0.040</td>
<td>0.302</td>
</tr>
<tr>
<td>Bricks &amp; Wood</td>
<td>5.381</td>
<td>39.411</td>
<td>0.053</td>
<td>0.288</td>
</tr>
<tr>
<td>Wood Cabinet</td>
<td>7.753</td>
<td>45.579</td>
<td>0.076</td>
<td>0.448</td>
</tr>
<tr>
<td>Keyboard</td>
<td>8.938</td>
<td>40.228</td>
<td>0.075</td>
<td>0.327</td>
</tr>
<tr>
<td>Floor Squares</td>
<td>13.028</td>
<td>41.607</td>
<td>0.176</td>
<td>0.343</td>
</tr>
<tr>
<td>Chair</td>
<td>17.516</td>
<td>29.170</td>
<td>0.296</td>
<td>0.236</td>
</tr>
<tr>
<td>Railing</td>
<td>18.262</td>
<td>20.401</td>
<td>0.231</td>
<td>0.156</td>
</tr>
<tr>
<td><strong>Mean</strong></td>
<td><strong>9.915</strong></td>
<td><strong>37.563</strong></td>
<td><strong>0.122</strong></td>
<td><strong>0.316</strong></td>
</tr>
</tbody>
</table>
## Gamma Estimation Results

| Image            | Our RMSE | RMSE of [2] | Our $||\hat{\gamma} - \gamma||_1$ | $||\hat{\gamma} - \gamma||_1$ of [2] |
|------------------|----------|-------------|-----------------------------------|--------------------------------------|
| Bookshelf        | 3.673    | 46.640      | 0.025                             | 0.431                                |
| Glass Ceiling    | 4.771    | 37.468      | 0.040                             | 0.302                                |
| Bricks & Wood    | 5.381    | 39.411      | 0.053                             | 0.288                                |
| Wood Cabinet     | 7.753    | 45.579      | 0.076                             | 0.448                                |
| Keyboard         | 8.938    | 40.228      | 0.075                             | 0.327                                |
| Floor Squares    | 13.028   | 41.607      | 0.176                             | 0.343                                |
| Chair            | 17.516   | 29.170      | 0.296                             | 0.236                                |
| Railing          | 18.262   | 20.401      | 0.231                             | 0.156                                |
| **Mean**         | **9.915**| **37.563**  | **0.122**                         | **0.316**                            |
Introduction
  Problem Statement

Texture Modelling
  Motivation
  Feature Extraction
  Boundary Effects
  Spatial Smoothness

Applications
  Segmentation
  Gamma Estimation
  Shading Reflectance

Contributions
Shading and Reflectance Decomposition

- $\mathcal{I} = \mathcal{R} \times \mathcal{S}$

- Smooth MRF estimation of shading
  - No bias field $\Rightarrow$ shading affects contrast and bias
  - MRF estimation for a set of parameters

$$f^{(k+1)} = \sum_{\theta \in \Theta} F^{-1}_\theta \left( \theta - w^\theta_p \left( \theta - f^{(k)} \right) \right)$$

DC Gain $\left( F^{-1}_\theta \right) = \frac{\prod_{\theta_1 \neq \theta} h^2_{\theta_1}}{\sum_{\theta_1 \in \Theta} \prod_{\theta_2 \neq \theta_1} h^2_{\theta_2}}$
Shading and Reflectance Decomposition

Algorithm Overview

1. Segment the image
2. Estimate the $\gamma$ factor of the camera and obtain the irradiance image
3. Estimate the smooth shading image from the irradiance image
4. Estimate the shape from shading using [9]
Shading and Reflectance Results
Shading and Reflectance Results
Shading and Reflectance Results

movies/segmentation-shading.mp4
Contributions

- Developed a texture model that captures scale, orientation, contrast, and bias
- Modelled smooth spatial changes in features
- Achieved robust texture segmentation, estimation of an unknown camera response, and shading/reflectance decomposition

Possible Future Directions

- Better boundary effect handling
- Probabilistic feature measurements
- Using the shading / shape estimation to improve segmentation
- Speed improvements
P. Brodatz.
*Textures: A Photographic Album for Artists and Designers.*

H. Farid.
Blind inverse gamma correction.

M. Heiler and C. Schnorr.
Natural image statistics for natural image segmentation.

N. Houhou, J.-P. Thiran, and X. Bresson.
Fast texture segmentation model based on the shape operator and active contour.

J. Kiefer.
Sequential minimax search for a maximum.

A nonparametric statistical method for image segmentation using information theory and curve evolution.
E. Parzen.

On estimation of a probability density function and mode. 


Shiftable multi-scale transforms. 

P. sing Tsai and M. Shah.

Shape from shading using linear approximation. 
Level Set Methods

- Implicitly define the curve within a 3D surface
- Define a height at every pixel in the image

The Surface $\varphi$

The Level Sets of $\varphi$
Level Set Methods

- The zero level set of $\varphi$ implicitly represents the 2D curve
- Variational calculus is used to perform gradient descent on some energy functional
Kernel Density Estimate

\[ p_x(x) \approx \frac{1}{Nh} \sum_{s=1}^{N} K \left( \frac{x - x_s}{h} \right) \]

\[ K(x) = -\frac{1}{\sqrt{2\pi}} \exp \left[ -x^2 \right] \]
Optimizing Mutual Information

\[
\arg \max_L |\Omega| I(E, \mu, \theta, \eta; L) - \alpha \int_C ds
\]

\[
= \arg \max_L |\Omega| [H(E, \mu, \theta, \eta) - H(E, \mu, \theta, \eta|L)] - \alpha \int_C ds
\]

\[
= \arg \max_L - |\Omega| H(E, \mu, \theta, \eta|L) - \alpha \int_C ds
\]

\[
= \arg \max_L - |\Omega| \sum_{\ell \in L} p_L(\ell) H(E, \mu, \theta, \eta|L = \ell) - \alpha \int_C ds
\]

\[
\approx \arg \max_L - |\Omega| \sum_{\ell \in L} \frac{|R^\ell|}{|\Omega|} \frac{1}{|R^\ell|} \int_{R^\ell} \log p(E_i, \mu_i, \theta_i, \eta_i|\ell) \, di - \alpha \int_C ds
\]

\[
= \arg \max_L - \sum_{\ell \in L} \int_{R^\ell} \log p^\ell_E(E_i) p^\ell_\mu(\mu_i) p^\ell_\theta(\theta_i) p^\ell_\eta(\eta_i) \, di - \alpha \int_C ds
\]
Level-set Gradient Descent of $I(X; L)$

$$\frac{\partial \varphi_i}{\partial t} = \log \frac{p^+_X(x_i)}{p^-_X(x_i)} - \alpha \kappa_i, \ \forall i \in \mathcal{C}$$
Brodatz Classification [1]

- 100% correct classification on Brodatz textures
- Able to segment Brodatz mosaics