Lecture 8
Texture

Dr. Lin ZHANG
School of Software Engineering
Tongji University
Fall 2011
Content

• General Overview
• Texture Classification
  • MR8
  • LBP
  • M-LBP
• Texture Synthesis
What is texture?

• Easy to recognize, hard to define
What is texture?

- Provides information in the spatial arrangement of colors or intensities in an image
- Characterized by the spatial distribution of intensity levels in a neighborhood
- Repeating pattern of local variations in image intensity
- Often has some degree of randomness
- Cannot be defined for a point
What is texture?

- Whether an effect is a texture or not depends on the scale at which it is viewed

* A leaf that occupies most of the image – object
* Foliage of a tree - texture
Applications of Texture

• Industrial inspection
• Biomedical image analysis
  - dermatosis
  - stomach ulcer
  - retina of a diabetes patient
  - tongue of a patient with kidney disease
• Fire or fume detection
Applications of Texture

- Remote sensing

- Entertainment
Topics for Texture Research

- Texture classification
  - identifying a given textured region from a given set of texture classes

*They belong to different kinds of textures*
Topics for Texture Research

• Texture segmentation
  • automatically determining the boundaries between various texture regions in an image
Topics for Texture Research

• Texture synthesis
  • construct large regions of texture from small example images
Texture Representation

- Image textures generally consist of organized patterns of regular sub-elements.
- One natural way to represent texture is to find the textons and then describe the way in which they are laid out.
- It generally required a human to look at the texture in order to decide what those fundamental units were.
A set of texture examples used in experiments with human subjects to tell how easily various types of textures can be discriminated.
Texture Representation

- Instead of looking for patterns at the level of arrowheads or triangles, look for simpler pattern elements (e.g. dots and bars)
- Easy: Find patterns by filtering the image
- Remember: There is a strong response to the filter when the image pattern in a neighborhood looks similar to the filter kernel
- Represent a texture in terms of the response of a collection of filters in different scales and orientations
Texture Representation

- Spot filters – they respond strongly to small regions that differ from their neighbors
- Bar filters – respond to oriented structure
- To obtain these filters – form a weighted difference of Gaussian filters at different scales
Content

• General Overview
• Texture Classification
  • MR8
  • LBP
  • M-LBP
• Texture Synthesis
Maximum-Response 8

- It is a learning based texture classification method
- Consists of a training stage and a classification stage
Learning stage I: Generating the texton dictionary. Multiple, unregistered images from the training set of a particular texture class are convolved with a filter bank. The resultant filter responses are aggregated and clustered into textons using the *K-Means* algorithm. Textons from different texture classes are combined to form the texton dictionary.
Learning stage II: Model generation. Given a training image, its corresponding model is generated by first convolving it with a filter bank and then labeling each filter response with the texton which lies closest to it in filter response space. The histogram of textons, i.e. the frequency with which each texton occurs in the labeling, forms the model corresponding to the training image.
Classification stage. A novel image is classified by forming its histogram and then using a nearest neighbor classifier to pick the closest model to it (in the $\chi^2$ sense). The novel image is declared as belonging to the texture class of the closest model.
• $\chi^2$ to compute the dissimilarity of two histograms

$$D(T, L) = \sum_{n=1}^{N} \frac{(T_n - L_n)^2}{(T_n + L_n)}$$

where $N$ is the number of bins, and $T_n$ and $L_n$ are the values of the sample and model histogram at the $n^{th}$ bin, respectively.
Maximum-Response 8

- Filter banks used in MR8
  - A set of first and second order Gaussian derivatives (three scales and 6 orientations); 1 Gaussian and 1 Laplician of Gaussian (LoG); altogether 38 filters
  - taking maximum responses of an anisotropic filter bank
  - At each location, a 8D response vector is obtained
Content

- General Overview
- Texture Classification
  - MR8
  - LBP
  - M-LBP
- Texture Synthesis
Local Binary Pattern

• It is totally decided by the **signs** of the differences between neighborhood pixels and the center pixel
• It is a simple yet powerful texture descriptor
• Now it has been widely used in image related CV and PR applications
Local Binary Pattern

Texture at $g_c$ is modeled using a local neighborhood of radius $R$, which is sampled at $P$ (8 in the example) points:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p$$
Local Binary Pattern

\[
LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) \ 2^p
\]

1. Sample

2. Difference

3. Threshold

\[
1 \times 1 + 1 \times 2 + 1 \times 4 + 1 \times 8 + 0 \times 16 + 0 \times 32 + 0 \times 64 + 0 \times 128 = 15
\]
Local Binary Pattern

Texture primitives ("micro-textons") detected by the uniform patterns of LBP

\[ 1 = \text{black} \]
\[ 0 = \text{white} \]

- Spot
- Spot / flat
- Line end
- Edge
- Corner
Local Binary Pattern

Spatial rotation of the binary pattern changes the LBP code:

![Diagram showing spatial rotation of binary pattern](image-url)
Local Binary Pattern

- Rotation invariant local binary patterns

Formally, rotation invariance can be achieved by defining:

\[
\text{LBP}_{P,R}^{ri} = \min\{\text{ROR}(\text{LBP}_{P,R}, i) \mid i = 0, \ldots, P-1\}
\]

ROR\((x, i)\) performs a circular bit-wise right shift on the \(P\)-bit number \(x\) \(i\) times.
Local Binary Pattern

'Uniform' patterns (P=8)

Examples of 'nonuniform' patterns (P=8)

\[ U(LBP_{p,R}) = \left| s(g_{p-1} - g_c) - s(g_0 - g_c) \right| + \sum_{p=1}^{P-1} \left| s(g_p - g_c) - s(g_{p-1} - g_c) \right| \]
Local Binary Pattern

The rotation invariant uniform LBP is defined as

\[ LBP_{P,R}^{rnu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c) & \text{if } U(LBP_{P,R}) \leq 2 \\ P + 1 & \text{otherwise} \end{cases} \]
Local Binary Pattern

$LBP_{8,1}^{riu^2}$
LBP-based Texture Classification

- Texture dataset

LBP operator

\( h_1, h_2, h_3, \ldots, h_n \)

Models

Matching

Test image

LBP operator

\( h_t \)

Output its class label

- Chi-square distance can be used for matching
- Nearest neighborhood can be used as the classifier
LBP-based Texture Classification

• Usually, a multi-scale extension can get better results

Information provided by N operators can be combined simply by summing up operatorwise similarity scores into an aggregate similarity score:

\[ L_N = \sum_{n=1}^{N} L_n \]

* e.g. \( LBP_{8,1}^{riu2} + LBP_{8,3}^{riu2} + LBP_{8,5}^{riu2} \)

Effectively, the above assumes that distributions of individual operators are independent
Content

• General Overview
• Texture Classification
  • MR8
  • LBP
  • M-LBP
• Texture Synthesis
Monogenic-LBP (M-LBP)

Motivation

- LBP is simple, powerful, and non-training
- LBP tends to oversimplify the local structure
- If some other rotation invariant features are incorporated, performance may be improved

What?

- Local phase
- Local surface type

How to extract?

- Monogenic signal
- Monogenic curvature tensor
Monogenic Signal—Definition

For a 2-D image $f(x)$, the monogenic signal is defined as

$$f_M(x) = \left(f, R_x \{f\}(x), R_y \{f\}(x)\right)$$

where

$$R_x(x) = \frac{x}{2\pi|x|^3}, \quad R_y(x) = \frac{y}{2\pi|x|^3}, \quad x = (x, y) \in \mathbb{R}^2$$

Transfer function of the Riesz kernel in the frequency domain:

$$\left(H_u(u), H_v(u)\right) = \left(-i\frac{u}{|u|}, -i\frac{v}{|u|}\right), \quad u = (u, v) \in \mathbb{R}^2$$
Monogenic Signal—Interpretation

\[ f_M(x) = \left( f, R_x \{f\}(x), R_y \{f\}(x) \right) \]

Local amplitude
\[ A = \sqrt{f^2 + (R_x \{f\})^2 + (R_y \{f\})^2} \]

Local orientation
\[ \theta = \arctan \frac{R_y \{f\}}{R_x \{f\}}, \theta \in [0, \pi) \]

For i1D signal \( f(x) \)

\[ \sqrt{R_x^2 \{f\}(0,0) + R_y^2 \{f\}(0,0)} = \left| (h_1 \ast f_\theta)(0) \right| = \left| \frac{1}{\pi} \int_{t \in \mathbb{R}} \frac{f(t \cos \theta, t \sin \theta)}{t} dt \right| \]

Partial Hilbert transform along the orientation \( \theta \)
Thus, the local phase of the 1D signal $f(x)$ can be defined analogously to the 1-D analytic signal as

$$\varphi = \text{atan} 2\left(\sqrt{R_x^2 \{f\} + R_y^2 \{f\}}, f\right), \varphi \in [0, \pi)$$

$\varphi$ is rotation invariant
Monogenic Curvature Tensor

- It is based on the 2\textsuperscript{nd}-order Riesz transforms

\[
T = \begin{bmatrix}
R_x \{ R_x [f] \} & R_x \{ R_y [f] \} \\
R_x \{ R_y [f] \} & R_y \{ R_y [f] \}
\end{bmatrix}
\]

\[
\det(T) = \left( R_x \{ R_x \{ f \} \} \right) \left( R_y \{ R_y \{ f \} \} \right) - \left( R_x \{ R_y \{ f \} \} \right)^2
\]
Monogenic Curvature Tensor (Cont.)

$f(x, y) \xrightarrow{\text{emebded into}} \text{Monge Patch: } s(x, y) = (x, y, f(x, y))$

Gaussian Curvature: 

$$K = \frac{f_{xx}f_{yy} - (f_{xy})^2}{\left(1 + f_x^2 + f_y^2\right)^2}$$

- If $K > 0$, $s$ is an elliptic patch
- If $K < 0$, $s$ is a hyperbolic patch

$\det(T)$ has the same sign as $K$

The sign $\det(T)$ partially determines the local surface type of $s$ and it is rotation invariant [1]

M-LBP [1]

- Combine the local phase information, the local surface type, with the LBP

Incorporating local phase information:

$$\varphi_c = \left\lceil \frac{\varphi}{(\pi / M)} \right\rceil \quad (M \text{ is set as } 5)$$

Incorporating local surface type information:

$$S_c = \begin{cases} 
0, & \det(T) \leq 0 \\
1, & \text{else}
\end{cases}$$

Then, we can obtain a new 3-D texton feature vector: M-LBP

$$\left( \varphi_c, S_c, LBP_{P,R}^{\text{riu}^2} \right)$$

In practice, band-pass filtering is needed before performing Riesz transform. Laplacian of Poisson (LOP) is used, whose transfer function in the Fourier domain is:

\[ \mathcal{F}\{LOP\}(\mathbf{u}) = -4\pi^2 |\mathbf{u}|^2 \exp(-2\pi |\mathbf{u}| \lambda), \mathbf{u} \in \mathbb{R}^2 \]
Examples for $\varphi_C$ and $S_C$

\[(\lambda = 3.5)\]
• Multiresolution analysis
  • In real applications, a multiresolution analysis can lead to better results
  • With our scheme, this can be achieved by combining information provided by multiple operators of varying

  \[(P, R, \lambda)\]

We use 3 M-LBP operators, \(M-LBP_i (i = 1, 2, 3)\) and the parameters are set as (8, 1, 3.5), (16, 3, 7), and (24, 5, 14).
A 3D normalized histogram $h_i$ (i= 1, 2, 3) can be constructed by counting the frequencies of textons $M-LBP_i$

$h_1$, $h_2$, and $h_3$ are concatenated as $h$

The bins of $h$ is 540 ($5*2*10 + 5*2*18 + 5*2*26$)

$\chi^2$ distance is used to measure the dissimilarity of the sample and model histograms

$$D(T, L) = \sum_{n=1}^{N} (T_n - L_n)^2 / (T_n + L_n)$$

Nearest neighbor classifier is used
Experimental Results & Comparison

Database
• We use CUReT database, which contains 61 textures and each texture has 92 images.

Methods used for comparison
• MR8, Joint, and LBP

Experiment settings
• T46A: The training set for each class was selected by taking one from every two adjacent images. This setting was used to simulate the situation of large and comprehensive training set.
• T23A: The training set for each class was selected by taking one from every four adjacent images. This setting was used to simulate the situation of small but comprehensive training set.
• T46F: The training set for each class was selected as the first 46 images. This setting was used to simulate the situation of large but less comprehensive training set.
• T23F: The training set for each class was selected as the first 23 images. This setting was used to simulate the situation of small and less comprehensive training set.
### Table 1. Classification accuracies (%) and feature sizes

<table>
<thead>
<tr>
<th></th>
<th>Histogram Bins</th>
<th>T46A</th>
<th>T23A</th>
<th>T46F</th>
<th>T23F</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBP</td>
<td>54</td>
<td>95.47</td>
<td>93.09</td>
<td>85.64</td>
<td>78.50</td>
</tr>
<tr>
<td>MR8</td>
<td>2440</td>
<td>97.65</td>
<td>96.15</td>
<td>88.70</td>
<td>77.83</td>
</tr>
<tr>
<td>Joint</td>
<td>2440</td>
<td>97.36</td>
<td>95.37</td>
<td>87.13</td>
<td>78.55</td>
</tr>
<tr>
<td>M-LBP</td>
<td>540</td>
<td>97.86</td>
<td>96.72</td>
<td>89.67</td>
<td>81.21</td>
</tr>
</tbody>
</table>
### Table 2. Time consumption (msec) at the classification stage

<table>
<thead>
<tr>
<th></th>
<th>Histogram construction</th>
<th>One matching</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBP</td>
<td>87</td>
<td>0.022</td>
</tr>
<tr>
<td>MR8</td>
<td>4960</td>
<td>0.089</td>
</tr>
<tr>
<td>Joint</td>
<td>13173</td>
<td>0.089</td>
</tr>
<tr>
<td>M-LBP</td>
<td>221</td>
<td>0.035</td>
</tr>
</tbody>
</table>
Summary

M-LBP integrates three rotation invariant measures, LBP, local phase information and local surface type information together. It maintains the simplicity of LBP and is non-training. It can achieve higher classification accuracy than the other representative competitors. It has a moderate feature size, bigger than LBP but much smaller than MR8 and Joint. It is more suitable for real applications where training samples are limited and not comprehensive. It works fast. Source code: [http://sse.tongji.edu.cn/linzhang/](http://sse.tongji.edu.cn/linzhang/)
Content

• General Overview

• Texture Classification
  • MR8
  • LBP
  • M-LBP

• Texture Synthesis
Texture Synthesis

- Texture synthesis is the process of algorithmically constructing a large digital image from a small digital sample image by taking advantage of its structural content.
- Texture synthesis can be used to fill in holes in images (as in inpainting), create large non-repetitive background images and expand small pictures.
Non-parametric Texture Synthesis

- Proposed in A. Efros and K. Leung, Texture Synthesis by Non-parametric Sampling, ICCV, 2009
- Very simple, yet effective; the most famous texture synthesis algorithm
Non-parametric Texture Synthesis

To synthesize a pixel $p$, search the sample image for pixels with similar neighborhood to $p$, construct a histogram for the distribution of these pixels, finally sample this distribution to obtain a value for $p$.

Similarity is based on the Gaussian-weighted sum squared difference, to preserve local structure.
Non-parametric Texture Synthesis

Growing texture on pixel at the time

- User defined window size indicates the randomness of the texture
- To grow from from scratch a 3x3 random seed from the sample is used
- Unless no close match is found pixels with most neighbors are synthesized first
- Importance of Gaussian-weighted similarity measure
Non-parametric Texture Synthesis

More results

French canvas

rafiia weave
Non-parametric Texture Synthesis

More results

- wood

- granite

Lin ZHANG, SSE, 2011
Non-parametric Texture Synthesis

More results

white bread

brick wall
Non-parametric Texture Synthesis

Constrained synthesis