Wavelets on graphs for texture-based image segmentation

Application for VHR Pleiades images

Minh Tân Pham\textsuperscript{1} Grégoire Mercier\textsuperscript{1}
Julien Michel\textsuperscript{2}

\textsuperscript{1}TELECOM Bretagne - \textsuperscript{2}CNES

\{minh.pham,gregoire.mercier\}@telecom-bretagne.eu
julien.michel@cnes.fr

Pleiades Days - April 2, 2014
Contents

1 Introduction

2 Signal processing on graphs
   • Graph and characteristics
   • Graph wavelet transform
   • Graph for images

3 Texture by graph wavelets
   • Methodology
   • Experimental results

4 Conclusion
Contents

1 Introduction

2 Signal processing on graphs
   - Graph and characteristics
   - Graph wavelet transform
   - Graph for images

3 Texture by graph wavelets
   - Methodology
   - Experimental results

4 Conclusion
Objective of this work

An approach for local texture-based image segmentation applied for very high spatial resolution imagery

Challenges:
- Zones to be segmented become too small
- Do not respect the stationary hypothesis

Proposition
- Sparse approach for image segmentation
- A graph-based approach
- Texture characterization via spectral graph wavelet transform
A cut from Melbourne by Pleiades (25-02-2012) © CNES Copyright 2013

Downloaded form www.astrium-geo.com
Contents

1 Introduction

2 Signal processing on graphs
   • Graph and characteristics
   • Graph wavelet transform
   • Graph for images

3 Texture by graph wavelets
   • Methodology
   • Experimental results

4 Conclusion
**Signal processing on graphs**

**Graph**

A graph \( G = (V, E, w) \) consists of:

- \( V \) Vertex set (i.e. nodes)
- \( E \) Set of edges between vertices
- \( w \) Edge weights involving vertex similarity

![Graph diagram](image)

**Characteristics**

- Adjacency matrix

\[
A = \begin{pmatrix}
0 & w_{0,1} & \cdots & w_{0,N-1} \\
 w_{1,0} & 0 & \cdots & w_{1,N-1} \\
\vdots & \vdots & \ddots & \vdots \\
 w_{N-1,0} & w_{N-1,1} & \cdots & 0
\end{pmatrix}
\]
Graph

A graph $G = (V, E, w)$ consists of:

- $V$ Vertex set (i.e. nodes)
- $E$ Set of edges between vertices
- $w$ Edge weights involving vertex similarity

Characteristics

- Laplacian matrix

$$L = \begin{pmatrix}
\sum_\ell w_{0,\ell} & w_{0,1} & \cdots & w_{0,N-1} \\
\vdots & \vdots & \ddots & \vdots \\
w_{1,0} & \sum_\ell w_{1,\ell} & \cdots & w_{1,N-1} \\
\vdots & \vdots & \ddots & \vdots \\
w_{N-1,0} & w_{N-1,1} & \cdots & \sum_\ell w_{N-1,\ell}
\end{pmatrix}$$
Signal processing on graphs

Characteristics

- For \( f \in \mathbb{R}^N \)
  \[
  (Lf)(m) = \sum_{n=0}^{N-1} w_{m,n} (f(m) - f(n))
  \]

- \( L \) is symmetric and positive

- Eigen decomposition of \( L \):
  - Non-negative eigenvalues: \( 0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{N-1} \)
  - Orthogonal eigenvector basis: \( \{ \chi_k \}_{k=0,1,\ldots,N-1} \)
  - Graph spectrum: \( \sigma(L) = \{ \lambda_0, \lambda_1, \ldots, \lambda_{N-1} \} \)
Graph Fourier transform

- Fourier transform \( f \in \mathbb{R}^N \)

\[
\hat{f}(k) = < \chi_k, f > = \sum_{n=0}^{N-1} f(n) \chi_k^*(n)
\]

- Inverse transform

\[
f(n) = \sum_{k=0}^{N-1} \hat{f}(k) \chi_k(n)
\]

We use a function $g : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ which represents a transfer function of a band-pass filter (in frequency domain $\chi_k$)

$$\widehat{T_g f}(k) = g(\lambda_k) \hat{f}(k)$$

$$(T_g f)(m) = \sum_{k=0}^{N-1} \widehat{T_g f}(k) \chi_k(m) = \sum_{k=0}^{N-1} g(\lambda_k) \hat{f}(k) \chi_k(m)$$

- Dilation property: $T_t^g = g(tL)$
- Generation of wavelet function $\psi_{t,n}(m)$

$$\psi_{t,n}(m) = \sum_{k=0}^{N-1} g(t\lambda_k) \chi_k^*(n) \chi_k(m)$$

Graph wavelet transform

- Wavelet coefficients of $f$

$$W_f(t, n) = (T^t_g f)(n) = <\psi_{t,n}, f> = \sum_{k=0}^{N-1} g(t\lambda_k)\hat{f}(k)\chi_k(n)$$

- A function $h: \mathbb{R}^+ \rightarrow \mathbb{R}^+$ which represents the transfer function of a low-pass filter (in frequency domain $\chi_k$) is used for generating scaling function $\phi(n)$ and corresponding coefficients:

$$\phi_n(m) = \sum_{k=0}^{N-1} h(\lambda_k)\chi_k^*(n)\chi_k(m)$$

$$S_f(n) = (T_h f)(n) = <\phi_n, f> = \sum_{k=0}^{N-1} h(\lambda_k)\hat{f}(k)\chi_k(n)$$
Example of filters for scaling function and wavelets

Example of filter design with 4 wavelet scales

Cubic spline filters

Meyer-like filters
Signal processing on graphs

Graphs for images

Local graph

Graphs for images

Local graph

Graphs for images

Local graph

Signal processing on graphs

Graphs for images

Local graph

Signal processing on graphs

Graphs for images

Local graph

Signal processing on graphs

Graphs for images

Non-local graph

Graphs for images

Non-local graph

Signal processing on graphs

Graphs for images

Non-local graph

Contents

1 Introduction

2 Signal processing on graphs
   - Graph and characteristics
   - Graph wavelet transform
   - Graph for images

3 Texture by graph wavelets
   - Methodology
   - Experimental results

4 Conclusion
Texture by graph wavelets

Methodology

Proposed processing chain

Initial Image → Extraction of representative pixels

Vertex set $E$ → Weighted graph construction

Graph $G$ → Multiscale decomposition via SGWT

Wavelet coeffs $W$ → Segmentation

Segmented Image
Texture by graph wavelets

Extraction of representative pixels

Texture can be represented and characterized by a set of points of interest

$\Rightarrow$ Sparse texture representation of images

- Representative pixels are extracted
- Compromise between loss of information and processing time

Suitable for huge data

Our approach:

$\{ (i, j) \in S_{\text{max}} \iff I(i, j) = \max_{(k, \ell) \in N_{\omega \times \omega}(i, j)} I(k, \ell) \}$

$S_{\text{max}}$ : set of local maximum pixels

$N_{\omega \times \omega}(i, j)$ : set of neighboring pixels of $(i, j)$ within the window size $\omega \times \omega$
Graph construction

**Graph vertices**  ⟷ extracted representative pixels (i.e. local maximum pixels in our work)

**Vertex description** For each vertex \( n \), create a vector of signatures \( v(n) \) which describes the environment around it.

\[ v(n) \text{ consists of some measures of intensity, distance and direction given by vertex } n \text{ and a number of its local neighboring maxima and minima} \]

**Weighted edge creation**

\[
w(i, j) = \begin{cases} 
\exp(-\gamma \left[ \text{dist}(v(i), v(j)) \right]^2) & \text{if } j \in \mathcal{N}_k(i), \\
0 & \text{otherwise}
\end{cases}
\]

- \( \text{dist}(v(i), v(j)) \) : similarity measure between vectors of signatures
- \( \mathcal{N}_k(i) \) : set of \( k \)-closest neighbors of \( i \) in terms of signature distances
- \( \gamma \) : a free parameter
Texture by graph wavelets

Results

- A ROI of $1200 \times 1500$ pixels
- Graph vertices = local maximum pixels inside a search window $15 \times 15$
- Graph wavelets performed with 3 scales
- Segmentation by *K-means clustering* with 4 classes on wavelet coefficients
Texture by graph wavelets

Results

- A ROI of 1500 × 2000 pixels
- Graph vertices = local maximum pixels inside a search window 11 × 11
- Graph wavelets performed with 3 scales
- Segmentation by *K-means clustering* with 4 classes on wavelet coefficients
1 Introduction

2 Signal processing on graphs
   - Graph and characteristics
   - Graph wavelet transform
   - Graph for images

3 Texture by graph wavelets
   - Methodology
   - Experimental results

4 Conclusion
Texture segmentation by graph wavelets

- Sparse image representation jointly with a graph wavelet-based image segmentation
- Suitable for huge size images in case of VHR imagery
- Graph-based approach still relevant with non stationary images
- Interesting and promising preliminary results