Random Walks for Image Segmentation

Group Learning Presentation
Xiaowei Zhou, 2010-12-10
Outline

• Problem Description and Basic Idea
• Solution of Random Walk Probability
• Properties and Results
• Relation to Other Methods
• Discussion
Problem Description

• Interactive Segmentation

![Partially labeled image](image1.png)  ![Segmented image](image2.png)

- Requirements:
  - fewer interactions
  - fast computation, fast editing
Previous Methods

• Interactive segmentation
  ▫ Intelligent Scissors (live wire)
    • fail in the presence of noise and weak boundary
    • too much interaction needed for high accuracy
  ▫ Active Contour
    • local minimum solution
    • inconvenient for editing
  ▫ Graph Cuts
    • small cut bias
    • no global optimal solution for multi-way cut
Basic Idea

A pixel $p$ belongs to which class?

A random walker starts from $p$, which seeds he will arrive first?

Green

Red

Yellow

Blue

The pixel intensity means: starting at this pixel the probability that a random walker will arrive the target seed before arriving other three seeds.
Why formulate in this way?

1. Fast solution
   - solving a sparse linear system
   - ability of multiple segmentation
   - fast editing

2. Nice properties
   - location of weak/missing boundaries
   - noise robustness
   - avoidance of trivial solutions
Algorithm Overview

1. Allocate region seeds $s_i$ for each region $i$

2. Calculate $u_i(x,y)$:
   the probability of first arriving $s_i$ for a random walker starting from $(x,y)$

3. Assign $(x,y)$ to Label $k$
   if $u_k(x,y)$ is the largest among $u_i(x,y)$
   for $i = 1 \ldots N$
Outline

• Problem Description and Basic Idea
• Solution of Random Walk Probability
• Properties and Results
• Relation to Other Methods
• Discussion
Definition of Image Graph

Graph nodes: pixels
Edge weights: similarity between neighboring pixels

\[ W_{ij} = e^{-\frac{(I_i - I_j)^2}{\sigma^2}} \]

Figures are borrowed from
http://www.cs.yale.edu/homes/spielman/sgta/
Graph Terminology

- **Adjacency Matrix**
  \[ W = [w_{ij}]_{n \times n} \]
  \[ w_{ij} = \begin{cases} 
  0 & i = j \\
  \frac{(I_i - I_j)^2}{\sigma^2} & i \neq j 
  \end{cases} \]

- **Degree Matrix**
  \[ D = \begin{bmatrix} 
  d_1 \\
  \vdots \\
  d_n 
  \end{bmatrix} \]
  \[ d_i = \sum_j w_{ij} \]

- **Laplacian Matrix**
  \[ L = D - W \]
  \[ l_{ij} = \begin{cases} 
  d_i & i = j \\
  -w_{ij} & i \neq j 
  \end{cases} \]
Random walk probability

Let \( p_{ij} \) be the probability of walking from node \( i \) to node \( j \):

\[
p_{ij} = \frac{w_{ij}}{\sum_{(i,k) \in E} w_{ik}} = \frac{w_{ij}}{d_i}
\]

Let \( x_i \) be the probability, starting at node \( i \), of a random walker reaching the red node before reaching the blue nodes.
Random walk probability

• How to compute $X = [x_1 \ x_2 \ \ldots \ x_n]^T$?

• $x_i$ has following property
  - $x_{\text{red}} = 1$ for the red node
  - $x_{\text{blue}} = 0$ for the blue nodes
  - $x_i = \sum_j p_{ij} x_j$ for other nodes
• If we define the transition matrix to be

\[
P = \left[ p_{ij} \right]_{n \times n} = \frac{W_{ij}}{\sum_{(i,k) \in E} W_{ik}} = D^{-1}W
\]

• \(PX = X\) except for labeled nodes.
• Reorder the node vector \(X\) to make labeled nodes in front of unlabeled:

\[
X = \begin{bmatrix} X_L \\ X_U \end{bmatrix}, \quad D = \begin{bmatrix} D_L & 0 \\ 0 & D_U \end{bmatrix}, \quad W = \begin{bmatrix} W_L & B \\ B^T & W_U \end{bmatrix}
\]

\[
P = \begin{bmatrix} D_L^{-1}W_L & D_L^{-1}B \\ D_U^{-1}B^T & D_U^{-1}W_U \end{bmatrix}
\]
• The properties can be described by:

\[
\begin{bmatrix}
X_L \\
X_U
\end{bmatrix} = \begin{bmatrix}
I & 0 \\
D_U^{-1}B^T & D_U^{-1}W_U
\end{bmatrix} \begin{bmatrix}
X_L \\
X_U
\end{bmatrix}
\]

• \(X_L\) are known and we need to calculate \(X_U\)

\[
X_U = D_U^{-1}B^T X_L + D_U^{-1}WX_U
\]

\[(I - D_U^{-1}W_U)X_U = D_U^{-1}B^T X_L\]

• Finally we can compute \(X_U\) by solving:

\[
(D_U - W_U)X_U = B^T X_L
\]
• $D_U - W_U$ is positive definite for fully connected graph, thus the linear system is nonsingular.

• Proof:

$$x_U^T (D_U - W_U) x_U$$

$$= x_U^T (\Delta D_U + D_U^s - W_U^s) x_U$$

$$= x_U^T \Delta D_U x_U + x_U^T (D_U^s - W_U^s) x_U$$

$$= \sum_{i \in U} (\sum_{j \in L} w_{ij}) x_i^2 + \sum_{i,j \in U} w_{ij} (x_i - x_j)^2$$

$\Delta D_U = \text{diag} \left[ \sum_{j \in L} w_{ij} \right]_{i \in U}$

$D_U^s = \text{diag} \left[ \sum_{j \in U} w_{ij} \right]_{i \in U}$

$W_U^s = W_U$
Minimization Point of View

• It can be shown that the random walk solution also minimizes following energy:

\[ \frac{1}{2} X^T L X = \frac{1}{2} \sum_{(i,j) \in E} w_{ij} (x_i - x_j)^2 \]

s.t. \( x_{red} = 1, \ x_{blue} = 0 \)
Continuous case

- In continuous case, it is called the Dirichlet integral

\[ D(u) = \frac{1}{2} \int_{\Omega} |\nabla u|^2 \, d\Omega \]

- The Eular-Lagrange equation of the above functional is the Laplace equation. Thus the solution of random walk probability in the continuous case is a harmonic function with predefined boundary conditions

\[ \nabla^2 u = 0 \quad u(s_{\text{red}}) = 1 \quad u(s_{\text{blue}}) = 0 \]
Outline

• Problem Description and Basic Idea
• Solution of Random Walk Probability
• Properties and Results
• Relation to Other Methods
• Discussion
Locate weak boundary
Noise Robustness
The secrets of nice properties

- **Property of Harmonic function**
  - *Maximum Principle*: the maximum and minimum are only attained on the boundary.
  - *Mean value property*: the function value is an average (or weighted average) of neighboring values.
Outline

• Problem Description and Basic Idea
• Solution of Random Walk Probability
• Properties and Results
• Relation to Other Methods
• Discussion
Graph Cut / Min-Cut

• The energy for Graph Cut segmentation is:

\[ \sum_{(i,j) \in E} w_{ij} |x_i - x_j| \]

\[ s.t. \; x = \{0,1\}, \; x_{\text{red}} = 1, \; x_{\text{blue}} = 0 \]

• It can be proved that the constraint \( x = \{0,1\} \) is slack, which means the Min-cut problem is a linear programming problem which minimize the energy with \( L1 \) penalty.
Normalized Cut (I)

- Revisit the Normalized Cut

\[ Ncut(A, B) = \frac{cut(A, B)}{\deg(A)} + \frac{cut(A, B)}{\deg(B)} \]

\[ = \ldots = \frac{y^T (D - W)y}{y^T Dy} \quad \text{with } y_i \in \{1, -1\}, \ y^T D 1 = 0 \]

- Relax \( y \) to be real numbers, the problem becomes:

\[ \min_{y \perp D1} \frac{y^T (D - W)y}{y^T Dy} \]
Normalized Cut (II)

• Let $z = D^{-2} y$, finally we want to solve

$$z^* = \arg \min_{z \perp D^{1/2} 1} z^T D^{-1/2} (D - W) D^{-1/2} z$$

$$= \arg \min_{z \perp D^{1/2} 1} z^T D^{-1/2} LD^{-1/2} z$$

• $D^{-1/2} LD^{-1/2}$ is called normalized Laplace matrix.

• The energy minimized in the Normalized Cut is similar to the random walk energy. They both have quadratic form of a Laplace matrix, while the constraints are different.
## Comparison

<table>
<thead>
<tr>
<th></th>
<th>Random Walk</th>
<th>Graph Cut</th>
<th>Normalized Cut</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Objective function</strong></td>
<td>$X^T LX$</td>
<td>$\sum_{(i,j) \in E} w_{ij}</td>
<td>x_i - x_j</td>
</tr>
<tr>
<td><strong>Constraints</strong></td>
<td>$x_{red} = 1$, $x_{blue} = 0$</td>
<td>$x_{red} = 1$, $x_{blue} = 0$</td>
<td>$z \perp D^{1/2} 1$</td>
</tr>
<tr>
<td><strong>Solution. (k=2)</strong></td>
<td>Linear system</td>
<td>Maximal Flow</td>
<td>Eigenvector of the normalized Laplace</td>
</tr>
<tr>
<td><strong>Solution. (k&gt;2)</strong></td>
<td>Linear system</td>
<td>Alpha-expansion (approximated)</td>
<td>Spectral clustering</td>
</tr>
<tr>
<td><strong>Uniqueness</strong></td>
<td>Unique</td>
<td>Not unique</td>
<td>Unique</td>
</tr>
<tr>
<td><strong>Result bias</strong></td>
<td>Shift related to seed placement</td>
<td>Small-cut bias</td>
<td>No bias</td>
</tr>
</tbody>
</table>
Outline

• Problem Description and Basic Idea
• Solution of Random Walk Probability
• Properties and Results
• Relation to Other Methods
• Discussion
Conclusions

• The advantages of random walker segmentation are mainly due to
  1. It describe the segmentation problem with a simple physical process, whose solution has well-studied probabilistic interpretation and nice mathematical properties.
  2. The discrete calculus formulation of the problem can be solved analytically and efficiently.
Why Use Graph?

- Capture the discrete nature of digital images
  - naturally model the spatial coherence of pixels
  - can be easily extended to high dimension
- Continuous differential operators can be replaced by combinatorial operators, which can be directly implemented.
- The linear algebra representation of the energy often has analytical solution.
### Discrete calculus

<table>
<thead>
<tr>
<th>Operator</th>
<th>Vector calculus</th>
<th>Combinatorial</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gradient</td>
<td>$\nabla$</td>
<td>$A$</td>
</tr>
<tr>
<td>Divergence</td>
<td>$\nabla \cdot$</td>
<td>$A^T$</td>
</tr>
<tr>
<td>Curl</td>
<td>$\nabla \times \nabla$</td>
<td>$K$</td>
</tr>
<tr>
<td>Laplacian</td>
<td>$\nabla \cdot \nabla$</td>
<td>$A^T A$</td>
</tr>
<tr>
<td>Beltrami</td>
<td>$\nabla C \cdot \nabla$</td>
<td>$A^T CA$</td>
</tr>
</tbody>
</table>
Key References


