COPERNICUS MASTER
IN DIGITAL EARTH

Image Processing

Graph-based image processing (Part 2)

Minh-Tan Pham

IRISA-UBS, Vannes, France

minh-tan.pham@univ-ubs.fr

Table of contents

Lecture overview

The objective of this lecture & lab (2 $\times$ 3 hours) is to introduce some basic tools of graph-based image analysis and processing. The lecture includes 2 parts:

- Part 1 (4h of lecture & lab):
  - General introduction of graph
  - How to construct a graph from an image?
  - Application 1: Graph-cut for segmentation
  - Application 2: Keypoint local graph for change detection
- Part 2 (2h of lecture & demo):
  - The spectral domain of graph
  - ~~Graph Fourier transform and multiscale signal analysis on graph~~
  - Application 3: Spectral graph clustering for optical image classification
  - ~~Application 4: Non-local graph for image denoising~~

Table of contents

Graph spectral domain

Reminder some basic matrices from a given weighted graph $\mathcal{G} = \{V, E, w\}$:

- The adjacency matrix $\mathcal{W}$ (i.e. similarity matrix, matrix of weights):

$$\mathcal{W}_{ij} = \begin{cases} w(i,j) & \text{if } (v_i, v_j) \in E, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

- The degree matrix $\mathcal{D}$ which is diagonal:

$$\begin{cases} \mathcal{D}_{ii} = \sum_j \mathcal{W}_{ij}, \\ \mathcal{D}_{ij} = 0, \forall i \neq j. \end{cases} \tag{2}$$

- The Laplacian matrix:

$$\mathcal{L} = \mathcal{D} - \mathcal{W}. \tag{3}$$

- The normalized Laplacian matrix:

$$\mathcal{L}_{\text{nor}} = \mathcal{D}^{-\frac{1}{2}} \mathcal{L} \mathcal{D}^{-\frac{1}{2}} = \mathcal{I} - \mathcal{D}^{-\frac{1}{2}} \mathcal{W} \mathcal{D}^{-\frac{1}{2}}, \tag{4}$$

- The random walk matrix:

$$\mathcal{P} = \mathcal{D}^{-1} \mathcal{W}. \tag{5}$$

Graph spectral domain

Spectral graph analysis is usually performed based on **the eigen decomposition** of the Laplacian matrix $\mathcal{L}$ [1].

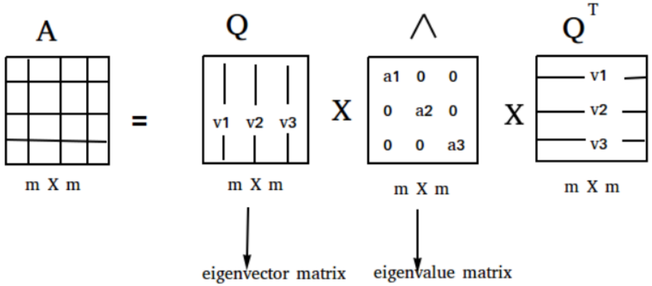
Some studies also exploit the adjacency matrix $\mathcal{W}$, the random walk matrix $\mathcal{P}$, or the normalized Laplacian matrix $\mathcal{L}_{\text{nor}}$ for the eigen decomposition to perform spectral analysis on graphs.

[1] Chung, Fan RK, and Fan Chung Graham. Spectral graph theory. No. 92. American Mathematical Soc., 1997. Shuman, David I., et al. "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains." IEEE signal processing magazine 30.3 (2013): 83-98.

Graph spectral domain Eigen decomposition

Reminder: (*Wikipedia*) Eigen decomposition (i.e. spectral decomposition) is the factorization of a matrix into a canonical form, whereby the matrix is represented in terms of its eigenvalues and eigenvectors[2].

$$A = Q \Lambda Q^T$$



- $v_1, v_2, \ldots, v_m$: the eigenvectors
- $a_1, a_2, \ldots, a_m$: the eigenvalues

[2]Image source: https://blog.paperspace.com/dimension-reduction-with-principal-component-analysis/

Graph spectral domain Eigen-decomposition

Okay, now let's perform the eigen decomposition of the Laplacian matrix $\mathcal{L}$ of a graph G with N nodes. (i.e. $\mathcal{L}$ has the size of $N \times N$).

$$\mathcal{L} = V \Lambda V^T$$

For each single eigen-vector v_k and eigenvalue λ_k, we can write the equation:

$$\mathcal{L} v_k = \lambda_k v_k, \quad k = 0, \ldots, N-1.$$

Graph spectral domain

For each single eigenvector v_k and eigenvalue λ_k, we can write the equation:

$$\mathcal{L}v_k = \lambda_k v_k, \quad k = 0, \ldots, N-1.$$

Since $\mathcal{L}$ is symmetric, positive and semi-definite, the following properties are considered:

- $\{\lambda_k\}_{k=0\ldots N-1}$ is a non-negative eigenvalue set where:

$$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{N-1}.$$

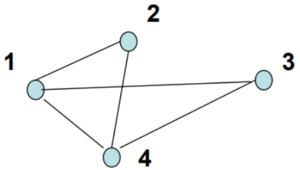
- $\{v_k\}_{k=0\ldots N-1}$ forms an orthogonal eigenvector basis.
- $\sigma(\mathcal{L}) = \{\lambda_0, \lambda_1, \ldots, \lambda_{N-1}\}$ refers to the entire graph spectrum in which a small eigenvalue λ_k represents a low frequency in graph spectral domain, and vice versa.

In graph spectral domain, the Laplacian eigenvalues and eigenvectors provide a similar notion to the classical Fourier frequency.

+ With $\lambda_0 = 0$, the associated eigenvector v_0 is a constant vector across the graph.
+ For a low frequency λ_k close to 0, the values v_k vary slowly and smoothly across the graph.
  $\Longrightarrow$ If two vertices v_i and v_j are connected with an important weight $w(i, j)$, corresponding values at their locations, i.e. $v_k(i)$ and $v_k(j)$, are likely to be similar.
+ For a high frequency λ_k far from 0, corresponding eigenvector v_k oscillates more rapidly.
  $\Longrightarrow$ If two vertices v_i and v_j are connected with a large weight $w(i, j)$, $v_k(i)$ and $v_k(j)$ are likely to have dissimilar values.

YES!!! We NEED example to understand!!!

Let's take a look on the following graph G (unweighted) and its matrices[3]



$$A_G = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \qquad D_G = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix} \qquad L_G = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & 0 & -1 \\ -1 & 0 & 2 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix}$$

[3] Image source: Spectral Graph Theory and its Applications, `http://web.mit.edu/6.454/www/www_fall_2004/lldai/slides.pdf`

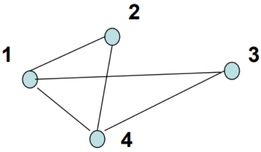
Eigen decomposition (spectral decomposition) of L_G:

$$L_G = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & 0 & -1 \\ -1 & 0 & 2 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix} = V \Lambda V^T$$

where

$$\Lambda = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} \text{ and } V = \begin{bmatrix} 1 & 0 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 0 & 3 & 0 \end{bmatrix}$$

So we obtain the eigenvalue set and the eigenvector basis as follows[4]



$$L_G = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & 0 & -1 \\ -1 & 0 & 2 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix}$$

$$\lambda = \{0, 2, 4, 4\}$$

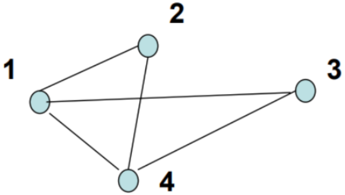
$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \begin{bmatrix} 0 \\ -1 \\ 1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} -1 \\ -1 \\ -1 \\ 3 \end{bmatrix} \quad \begin{bmatrix} -2 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

Let's analyze!!!!

[4] Image source: Spectral Graph Theory and its Applications, `http://web.mit.edu/6.454/www/www_fall_2004/lldai/slides.pdf`

Graph spectral domain Example

Let's analyze!!!! [5]



$$\lambda = \{0, 2, 4, 4\}$$

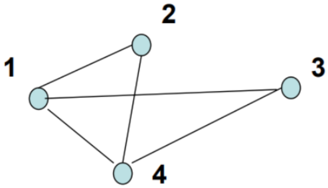
$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \quad \begin{bmatrix} 0 \\ -1 \\ 1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} -1 \\ -1 \\ -1 \\ 3 \end{bmatrix} \quad \begin{bmatrix} -2 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

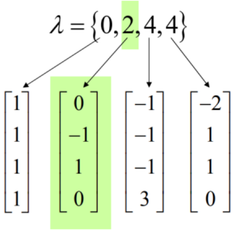
- All $\lambda_k \geq 0$
- $0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \lambda_3$
- v_0 is a constant vector

[5] Image source: Spectral Graph Theory and its Applications, `http://web.mit.edu/6.454/www/www_fall_2004/lldai/slides.pdf`

Graph spectral domain Example

Let's analyze!!!! [6]





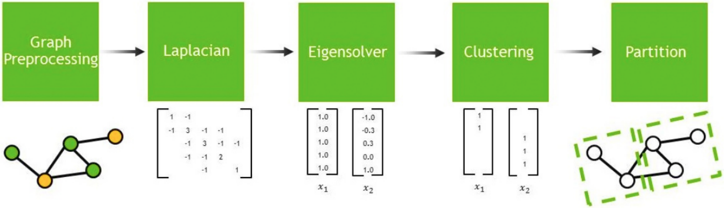
- values of v_1 vary slowly and smoothly across the graph
  - node_2 connects to node_1 and node_4 so $|v_1(2) - v_1(1)| = 1$ & $|v_1(2) - v_1(4)| = 1$
  - node_2 does not connect to node_3 $|v_1(2) - v_1(3)| = 2$

[6] Image source: Spectral Graph Theory and its Applications, `http://web.mit.edu/6.454/www/www_fall_2004/lldai/slides.pdf`

Spectral graph clustering

Spectral (graph) clustering: (Wikipedia) In multivariate statistics and the clustering of data, spectral clustering techniques make use of the spectrum (eigenvalues) of the similarity matrix (weight matrix) of the data to perform dimensionality reduction before clustering in fewer dimensions.[7]

The spectral (graph) clustering scheme constructs the **graph Laplacian matrix**, solves an associated **eigenvalue problem**, and extracts splitting information from the calculated **eigenvectors**.[8]



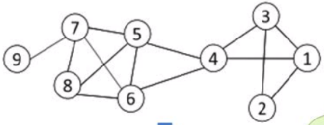
[7] https://en.wikipedia.org/wiki/Spectral-clustering
[8] Image source: https://developer.nvidia.com/discover/cluster-analysis

Spectral graph clustering

Algorithm Given a set of N data points $x_1, \ldots, x_N$ and a distance measure metric, following steps are done for spectral clustering to divide the data into C clusters:

- Construct a weighted graph $\mathcal{G}$ connecting N given points where each data point x_i corresponds to a graph node n_i and the adjacency matrix $\mathcal{W} \in \mathbb{R}^{N \times N}$ is calculated based on the similarity between data points using the given distance measure.
- Compute the graph Laplacian matrix $\mathcal{L}$.
- Compute the first C eigenvectors $\{v_k\}_{k=0,\ldots,C-1}$ corresponding to the C smallest eigenvalues $\{\lambda_k\}_{k=0,\ldots,C-1}$ via the eigen decomposition of $\mathcal{L}$.
- Consider the matrix $U = [v_0, \ldots, \chi_{C-1}] \in \mathbb{R}^{N \times C}$ as feature space of N data points where each point associates with a feature vector in $\mathbb{R}^C$.
- Cluster the above N points into C clusters by performing the *K-means clustering* algorithm on their feature vectors.

Example Let's take a look on the following example[9]



Two communities:
{1, 2, 3, 4} and {5, 6, 7, 8, 9}

The 1st eigenvector means all nodes belong to the same cluster, no use

k-means

$$D = diag(3, 2, 3, 4, 4, 4, 4, 3, 1)$$

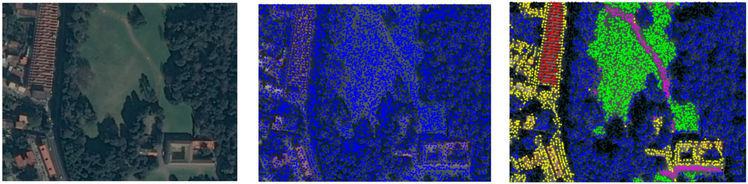
$$\tilde{L} = D - A = \begin{bmatrix} 3 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 4 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 4 & -1 & -1 & -1 & 0 \\ 0 & 0 & 0 & -1 & -1 & 4 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 4 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & -1 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix} \Rightarrow S = \begin{bmatrix} 0.33 & -0.38 \\ 0.33 & -0.48 \\ 0.33 & -0.38 \\ 0.33 & -0.12 \\ 0.33 & 0.16 \\ 0.33 & 0.16 \\ 0.33 & 0.30 \\ 0.33 & 0.24 \\ 0.33 & 0.51 \end{bmatrix}$$

[9] Community Detection and Graph-based Clustering, `https://slideplayer.com/slide/2309634/`

Spectral graph clustering

Application to remote sensing image [10]

- Extract keypoints from the image (local maximum keypoints)
- Construct a weighted graph to connect keypoints
- Perform spectral graph clustering with the number of clusters C equal to the number of expected classes

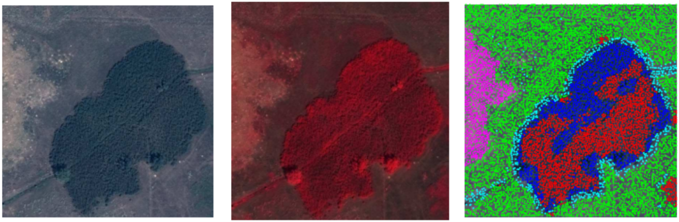


[10] PhD of M-T Pham

Spectral graph clustering

Application to remote sensing image [11]

- Extract keypoints from the image (local maximum keypoints)
- Construct a weighted graph to connect keypoints
- Perform spectral graph clustering with the number of clusters *C* equal to the number of expected classes



[11] PhD of M-T Pham

References I

Books & articles:

1. Lézoray, Olivier, and Leo Grady, eds. **Image processing and analysis with graphs: theory and practice.** CRC Press, 2012.
2. Chung, Fan RK, and Fan Chung Graham. **Spectral graph theory.** No. 92. American Mathematical Soc., 1997.
3. Shuman, David I., et al. **The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains.** IEEE signal processing magazine 30.3 (2013): 83-98.
4. Pham, Minh-Tan, et al. **Pointwise graph-based local texture characterization for very high resolution multispectral image classification."** IEEE JSTARS 8.5 (2015): 1962-1973.

Courses:

1. Spectral Graph Theory and its Applications
   `http://web.mit.edu/6.454/www/www_fall_2004/lldai/slides.pdf`
2. Community Detection and Graph-based Clustering
   `https://slideplayer.com/slide/2309634/`