Data Analysis and Manifold Learning
Lecture 10: Spectral Matching

Radu Horaud
INRIA Grenoble Rhone-Alpes, France
Radu.Horaud@inrialpes.fr
http://perception.inrialpes.fr/

Outline of Lecture 10

- Graph isomorphism problems
- Solving graph isomorphism with spectral matching
- Problems with standard algorithms
- The "signature" of an eigenvector
- Graph matching based on point registration

Material for this lecture

- A. Sharma, R. Horaud, and D. Mateus. "3D Shape Registration Using Spectral Graph Embedding and Probabilistic Matching". To appear soon as a book chapter. 2011.

The Graph Isomorphism Problem

- Let two graphs $\mathcal{G}_A$ and $\mathcal{G}_B$ have the same number of nodes, and let $\pi : \mathcal{V}_A \longrightarrow \mathcal{V}_B$ be a bijection;
- π is an isomorphism if and only if:

$$u \sim v \Longleftrightarrow \pi(u) \sim \pi(v)$$

- The notion of graph isomorphism allows to study the structure of graphs.
- The *graph isomorphism problem*: an algorithm that determines whether two graphs are isomorphic
- It is one of only two, out of 12 total, problems listed in Garey & Johnson (1979) whose complexity remains unresolved: It has not been proven to be included in, nor excluded from, P (polynomial) or NP-complete.

The Subgraph Isomorphism Problem

- Let two graphs $\mathcal{G}_A$ with n_A nodes and $\mathcal{G}_B$ with n_B nodes such that $n_A > n_B$.
- One must determine whether $\mathcal{G}_A$ contains a subgraph that is isomorphic to $\mathcal{G}_B$.
- The number of possible solutions is: $\begin{pmatrix} n_B \\ n_A \end{pmatrix} n_B!$
- The problem is NP-complete (Nondeterministic polynomial).

The Maximum Subgraph Matching Problem

- Let two graphs $\mathcal{G}_A$ with n_A nodes and $\mathcal{G}_B$ with n_B nodes such that $n_A > n_B$.
- Determine the largest pair of subgraphs $(\mathcal{G}'_A, \mathcal{G}'_B)$, with $\mathcal{G}'_A \subset \mathcal{G}_A$ and $\mathcal{G}'_B \subset \mathcal{G}_B$, such that $\mathcal{G}'_A$ and $\mathcal{G}'_B$ are isomorphic.
- The number of possible solutions is:

$$\sum_{i=1}^{n_B} \left(\begin{array}{c} i \\ n_A \end{array} \right) \left(\begin{array}{c} i \\ n_B \end{array} \right) i!$$

- The problem is NP-complete

How to Solve Graph Isomorphism Problems

- Let's consider, as above two undirected graphs (weights are all equal to 1) with the same number of nodes.
- Let's define a metric between the two graphs as:

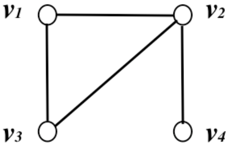
$$\arg\max_{\pi} \sum_{i=1}^{n} \sum_{j=1}^{n} (e_{ij} - e_{\pi(i)\pi(j)})^2$$

Using Graph matrices

- Let $\mathbf{W}_A$ and $\mathbf{W}_B$ be the adjacency matrices of two graphs with the same number of nodes n
- Let $\mathbf{P} \in \mathcal{P}_n$ be a permutation matrix: exactly one entry in each row and column is equal to 1, and all the other entries are 0:
- Left multiplication of $\mathbf{W}$ with $\mathbf{P}$ permutes the rows of $\mathbf{W}$ and
- Right multiplication of $\mathbf{W}$ with $\mathbf{P}$ permutes the columns of $\mathbf{W}$
- What does $\mathbf{PWP}^\top$?
- $\mathbf{W} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$
- $\mathbf{PWP}^\top = (\mathbf{PU})\,\mathbf{\Lambda}\,(\mathbf{PU})^\top$
- Think of the rows of $\mathbf{U}$ as the coordinates of the graph's vertices in spectral space ... the nodes are renamed

A Simple Example

$$\mathbf{W}_A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$



$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \mathbf{W}_B = \mathbf{P}\mathbf{W}_A\mathbf{P}^\top = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Which corresponds to:

$$v_1 \leftrightarrow u_2, \; v_2 \leftrightarrow u_1, \; v_3 \leftrightarrow u_4, \; v_3 \leftrightarrow u_4$$

The Graph Isomorphism Problem with Matrices

- The metric between two graphs becomes:

$$\mathbf{P}^\star = \arg\min_{\mathbf{P}} \|\mathbf{W}_A - \mathbf{P}\mathbf{W}_B\mathbf{P}^\top\|^2$$

- where the Frobenius norm is being used:

$$\|\mathbf{A}\|_F^2 = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}^2 = \mathrm{tr}(\mathbf{A}^\top\mathbf{A}).$$

An Exact Solution

- When the metric is equal to zero:

$$\mathbf{W}_A = \mathbf{P}\mathbf{W}_B\mathbf{P}^\top$$

- Two isomophic graphs have the same eigenvalues;
- The reverse is not always true.

Finding an Exact Solution in the Spectral Domain

- Let's write an equality:

$$\mathbf{W}_A = \mathbf{P}^\star \mathbf{W}_B {\mathbf{P}^\star}^\top$$

- Consider the spectral decompositions:

$$\mathbf{W}_A = \mathbf{U}_A \mathbf{\Lambda}_A \mathbf{U}_A^\top \text{ and } \mathbf{W}_B = \widetilde{\mathbf{U}}_B \mathbf{\Lambda}_B \widetilde{\mathbf{U}}_B^\top$$

- with the notation:

$$\widetilde{\mathbf{U}}_B = \mathbf{U}_B \mathbf{S} \text{ and } \mathbf{S} = \text{Diag}[s_i], s_i = \pm 1$$

- By substitution in the first equation, we obtain:

$$\mathbf{P}^\star = \mathbf{U}_B \mathbf{S} \mathbf{U}_A^\top$$

Short Discussion

- There are as many solutions as the number of possible $\mathbf{S}$-matrices, i.e., 2^n.
- Not all of these solutions correspond to a valid permutation matrix.
- There exist some $\mathbf{S}^\star$ that exactly make $\mathbf{P}^\star$ a permutation: these are *valid* solutions to the graph isomorphism problem
- Eigenspace alignment:

$$\mathbf{U}_A = \mathbf{P}^\star \mathbf{U}_B \mathbf{S}^\star$$

  The rows of $\mathbf{U}_A$ can be interpreted as the coordinates of the graph's vertices in the eigenspace of $\mathbf{W}_A$. The above equation can be interpreted as a registration between the embedding of the two graphs. Hence:
- *The graph isomorphism problem can be viewed as a rigid registration problem in embedded space.*

The Hoffman-Wienlandt Theorem

Theorem

(Hoffman and Wielandt) If $\mathbf{W}_A$ *and* $\mathbf{W}_B$ *are real-symmetric matrices, and if* α_i *and* β_i *are their eigenvalues arranged in increasing order,* $\alpha_1 \leq \ldots \leq \alpha_i \leq \ldots \leq \alpha_n$ *and* $\beta_1 \leq \ldots \leq \beta_i \leq \ldots \leq \beta_n$*, then*

$$\sum_{i=1}^{n} (\alpha_i - \beta_i)^2 \leq \|\mathbf{W}_A - \mathbf{W}_B\|^2 \quad (1)$$

- This theorem is the fundamental building block of spectral graph matching.

Additional Results

Corollary

The inequality (1) becomes an equality when the eigenvectors of $\mathbf{W}_A$ are aligned with the eigenvectors of $\mathbf{W}_B$ up to a sign ambiguity:

$$\mathbf{U}_B = \mathbf{U}_\mathbf{A}\mathbf{S}. \tag{2}$$

Corollary

If $\mathbf{Q}$ is an orthogonal matrix, then

$$\sum_{i=1}^{n}(\alpha_i - \beta_i)^2 \leq \|\mathbf{W}_A - \mathbf{Q}\mathbf{W}_B\mathbf{Q}^\top\|^2. \tag{3}$$

- Indeed, matrix $\mathbf{Q}\mathbf{W}_B\mathbf{Q}^\top$ has the same eigenvalues as matrix $\mathbf{W}_B$.

Umeyama's Theorem

Theorem

(Umeyama'1988) If $\mathbf{W}_A$ and $\mathbf{W}_B$ are real-symmetric matrices with n distinct eigenvalues (that can be ordered), $\alpha_1 < \ldots < \alpha_i < \ldots < \alpha_n$ and $\beta_1 < \ldots < \beta_i < \ldots < \beta_n$, the minimum of :

$$J(\mathbf{Q}) = \|\mathbf{W}_A - \mathbf{Q}\mathbf{W}_B\mathbf{Q}^\top\|^2$$

is achieved for:

$$\mathbf{Q}^\star = \mathbf{U}_A\mathbf{S}\mathbf{U}_B^\top \tag{4}$$

and hence (3) becomes an equality:

$$\sum_{i=1}^{n}(\alpha_i - \beta_i)^2 = \|\mathbf{W}_A - \mathbf{Q}^\star\mathbf{W}_B\mathbf{Q}^{\star\top}\|^2.$$

Method Proposed by Umeyama in 1988

- Notice that (4) can be written as:

$$\mathbf{U}_A = \mathbf{Q}^\star \mathbf{U}_B \mathbf{S}$$

  which is a *relaxed* version of the the permutation matrix in the exact isomorphism case (permuation is replaced by an orthogonal matrix).
- For each sign matrix $\mathbf{S}$ (remember that there are 2^n such matrices) there is an orthogonal matrix that satisfies Umeyama's theorem, but not all these matrices can be easily relaxed to a permutation, and not all these permutations correspond to an isomorphism.

Umeyama's Heuriststic

- Take the absolute values of the eigenvector's components:

$$\overline{\mathbf{U}}_A(i,j) = |u_{ij}|$$

- It can be shown that the problem can be written as the following maximization problem:

$$\max_{\mathbf{Q}} \operatorname{tr}(\overline{\mathbf{U}}_A \overline{\mathbf{U}}_B^\top \mathbf{Q}^\top)$$

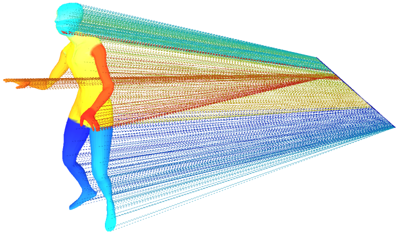
  This is not, however, such an easy problem to solve (See Umeyama'1988 for more details).
- It can be looked at as an assignment problem, namely extracting a permutation matrix $\mathbf{P}$ from the nonnegative matrix $\overline{\mathbf{Q}} = \overline{\mathbf{U}}_A \overline{\mathbf{U}}_B^\top$.

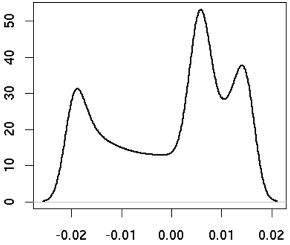
Discussion

The method just described has serious limitations:

- It applies to graphs with the same number of nodes;
- It assumes that there are no eigenvalue multiplicities and that the eigenvalues can be reliably ordered;
- The heuristic proposed is weak and it does not necessarily lead to a simple algorithm;
- Other heuristics were proposed.

Eigenvector Histogram





- The Laplacian eigenvector associated with the smallest non-null eigenvalue is the direction of maximum variance of a graph (principal component)
- The histogram of this eigenvector's entries is invariant to vertex ordering.

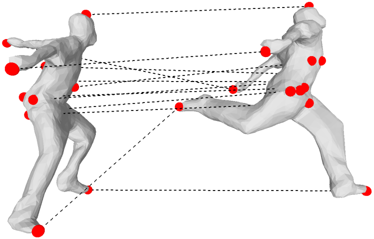
Characterization of These Histograms

- If $\mathbf{L}\boldsymbol{u} = \lambda\boldsymbol{u}$ then: $\mathbf{P}\mathbf{L}\mathbf{P}^\top\,\mathbf{P}\boldsymbol{u} = \lambda\mathbf{P}\boldsymbol{u}$
- The vectors $\boldsymbol{u}$ and $\mathbf{P}\boldsymbol{u}$ have the same histograms;
- Remind that for each eigenvector $\boldsymbol{u}_i$ of $\mathbf{L}$ we have $-1 < u_{ik} < +1$, $\overline{u}_k = 0$, and $\sigma_k = 1/n$.
- The number of bins and the bin-width are invariant:

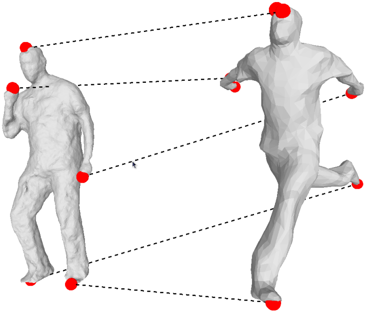
$$\begin{aligned} w_k &= \frac{3.5\sigma_k}{n^{1/3}} = \frac{3.5}{n^{4/3}} \\ b_k &= \frac{\sup_i u_{ik} - \inf_i u_{ik}}{w_k} \approx \frac{n^{4/3}}{2} \end{aligned}$$

- The histogram is not invariant to the sign change, i.e., $H\{\boldsymbol{u}\} \neq H\{-\boldsymbol{u}\}$.

Shape Matching (1)

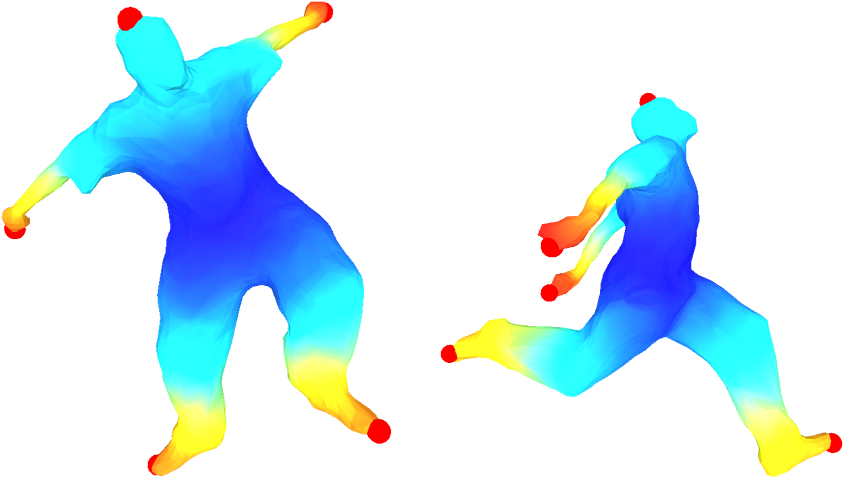


$t = 200,\ t' = 201.5$

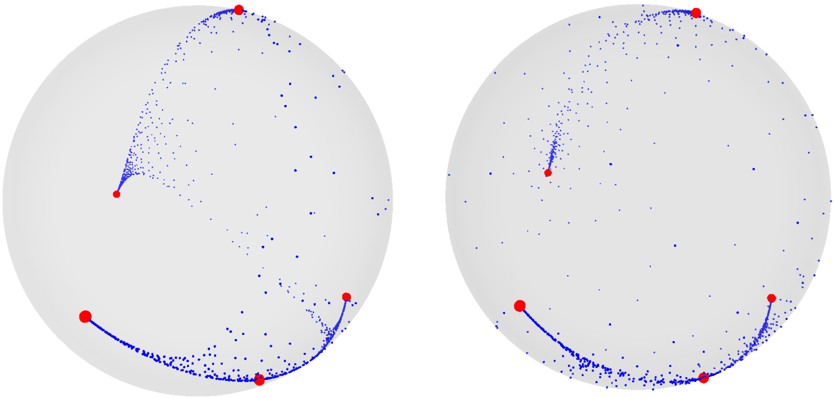


$t = 90,\ t' = 1005$

Shape Matching (2)



Shape Matching (3)



Sparse Shape Matching

- Shape/graph matching is equivalent to matching the embedded representations [Mateus et al. 2008]
- Here we use the projection of the embeddings on a unit hyper-sphere of dimension K and we apply rigid matching.
- How to select t and t', i.e., the scales associated with the two shapes to be matched?
- How to implement a robust matching method?

Scale Selection

- Let $\mathbf{C}_X$ and $\mathbf{C}_{X'}$ be the covariance matrices of two different embeddings $\mathbf{X}$ and $\mathbf{X}'$ with respectively n and n' points:

$$\det(\mathbf{C}_X) = \det(\mathbf{C}_{X'})$$

- $\det(\mathbf{C}_X$ measures the volume in which the embedding X lies. Hence, we impose that the two embeddings are contained in the same volume.
- From this constraint we derive:

$$t' \operatorname{tr}(\mathbf{L}') = t \operatorname{tr}(\mathbf{L}) + K \log n/n'$$

Robust Matching

- Build an association graph.
- Search for the largest set of mutually compatible nodes (maximal clique finding).
- See [Sharma and Horaud 2010] (Nordia workshop) for more details.