# GPBR Exercise 3

Name: Soi Zhi Wen

Matriculation Number: 22-132-245

The eigendecomposition of :

The eigendecomposition of :

The resulting matrix:

Assignment found by LSAP solver:

This indicates that

node in is matched with node in ,

node in is matched with node in ,

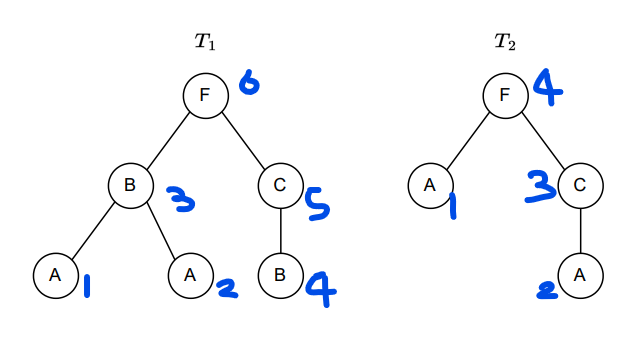
node in is matched with node in ,

node in is matched with node in ,

node in is matched with node in .

|  |  |
| --- | --- |
| **Spectral Graph Matching** | **Continuous Graph Matching** |
| 1. Compute the Laplacian matrix for each graph. | 1. Compute the distance metric between each pair of nodes in each graph. |
| 2. Compute the first k eigenvectors of each Laplacian matrix. | 2. Solve the continuous optimization problem to find the best correspondence between the graphs. |
| 3. Construct a matrix Y for each graph, where the columns are the k eigenvectors. | 3. Use the optimal correspondence to transform one graph into the other. |
| 4. Compute the orthogonal Procrustes transformation between the two Y matrices. |  |
| 5. Apply the transformation to one of the Y matrices. |  |
| 6. Compute the matrix product of the transformed Y matrix and the other original Y matrix. |  |

|  |  |
| --- | --- |
| **Spectral Graph Matching** | **Continuous Graph Matching** |
| **Similarities** | |
| Both approaches aim to find a permutation matrix that maps nodes in one graph to nodes in another graph, such that the two graphs are as similar as possible. | |
| Both approaches use optimization techniques to find the best permutation matrix that minimizes a given objective function. | |
| Both approaches are based on the idea of embedding graphs into a lower-dimensional space, where matching can be performed more efficiently. | |
| **Differences** | |
| Uses eigenvectors of the graph Laplacian to embed the graphs into a lower-dimensional space. | Uses a mapping function to embed the graphs into a continuous space. |
| Requires that the two graphs being matched have the same number of nodes. | Can handle graphs of different sizes. |
| More sensitive to noise in the eigenvectors. | Handle noisy data more effectively since it allows for a smooth mapping between the two graphs. |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  | 0 | 1 | 2 | 3 | 4 |
|  | 1 | 0 | 1 | 2 | 3 |
|  | 2 | 1 | 0 | 1 | 2 |
|  | 3 | 2 | 1 | 1 | 2 |
|  | 4 | 3 | 2 | 2 | 2 |
|  | 5 | 4 | 3 | 2 | 3 |
|  | 6 | 5 | 4 | 3 | 2 |

