Dimensionality Reduction of Laplacian Embedding for 3D Mesh Reconstruction

I Mardhiyah\textsuperscript{1}, S Madenda\textsuperscript{2}, R A Salim\textsuperscript{3}, I M Wiryana\textsuperscript{4}.

\textsuperscript{1,2,3,4} Universitas Gunadarma, Depok, Jawa Barat, Indonesia. Email: \{iffatul\textsuperscript{1}, sari\textsuperscript{2}, ravi\textsuperscript{3}, mwiryana\textsuperscript{4}\}@staff.gunadarma.ac.id

Abstract. Laplacian eigenbases are the important thing that we have to process from 3D mesh information. The information of geometric 3D mesh are include vertices locations and the connectivity of graph. Due to spectral analysis, geometric 3D mesh for large and sparse graphs with thousands of vertices is not practical to compute all the eigenvalues and eigenvector. Because of that, in this paper we discuss how to build 3D mesh reconstruction by reducing dimensionality on null eigenvalue but retain the corresponding eigenvector of Laplacian Embedding to simplify mesh processing. The result of reducing information should have to retained the connectivity of graph. The advantages of dimensionality reduction is for computational efficiency and problem simplification. Laplacian eigenbases is the point of dimensionality reduction for 3D mesh reconstruction. In this paper, we show how to reconstruct geometric 3D mesh after approximation step of 3D mesh by dimensionality reduction. Dimensionality reduction shown by Laplacian Embedding matrix. Furthermore, the effectiveness of 3D mesh reconstruction method will evaluated by geometric error, differential error, and final error. Numerical approximation error of our result are small and low complexity of computational.

1. Introduction
The 3D shape is a graph which has all the information of vertices. 3D mesh as a surface is viewed by its connected graph. In this case, we discuss about the undirected graph of 3D mesh which has information of the vertices and the faces. Geometric reconstruction is one of application of 3D mesh processing. Due to spectral analysis, geometric 3D mesh for large and sparse graphs with thousands of vertices is not practical to compute all the eigenvalues and eigenvector. Because of that, our new method for 3D mesh reconstruction reduce dimensionality to simplify mesh processing. Reduce dimensionality by reducing spectral will reduce the information, but the result of reducing information should have to retained the connectivity of graph. The advantages of dimensionality reduction is for computational efficiency and problem simplification. Laplacian eigenbases is the point of dimensionality reduction for 3D mesh reconstruction. In this paper, we show how to reconstruct geometric 3D mesh after approximation step of 3D mesh by dimensionality reduction. Dimensionality reduction shown by Laplacian Embedding matrix. Furthermore, the effectiveness of 3D mesh reconstruction method will evaluated by geometric error, differential error, and final error. Numerical approximation error of our result are small and low complexity of computational.
reconstruction method will be evaluated by geometric error, differential error, and final error. Final error is average of geometric error and differential error.

[12] used a small set of eigenvalues and eigenvectors to match 3D shape. Laplacian mesh processing in [12] was done by reducing spectral information but still hold the original 3D mesh geometry. [8] introduced spectral compression of 3D mesh geometry by spectral method using a linear combination of a number of basis vectors. [2] compressed the 3D triangle mesh geometry on 3D shape geometric representation problem, by projecting to $N$ dimension to build spectral decomposition. They obtained three vectors of dimension $N$ which are called spectra or pseudo frequencies. [13] represents the surface by adding anchors on Laplacian matrix. [11], [13]-[15] introduced the new transformation of Laplacian matrix by differential coordinates called $\hat{\delta}$-coordinates and adding anchors on rows of Laplacian matrix. Adding rows on Laplacian matrix made the size of row of matrix growth. Therefore the complexity of their computation was take a lot of time. [3] used TAUCS library [16] as a tools for linear solving on approximation problems. Other methods on 3D mesh reconstruction are Laplace-Beltrami eigenfunctions to understands the geometry of 3D mesh on [10], an ellipsoid shape approximation approach on [9], and Spectral Graph Wavelets (SGW) on [17].

2. Spectral graph
In this study, before 3D mesh reconstruction, we need to graph analysis, dimensionality reduction, and approximate 3D that show on Figure 1. Graph analysis show the geometry of 3D mesh by Laplacian matrix. Laplacian matrix needs vertices location on Euclidean space and connected vertices informations. To simplify a large graph problems which has thousands of vertices, we need to reduce some information but still retain the geometry connectivity, so it is dimensionality reduction purpose on second step. 3D mesh approximation will represent the 3D mesh geometry on other way, so after that we can reconstruct the new 3D mesh.

![Figure 1. 3D mesh reconstruction steps.](image)

2.1 Graph and the corresponding matrix
Laplacian eigenbases are the important thing that we have to process from 3D mesh information. The information of geometric 3D mesh are include vertices locations and the connectivity of graph. In this paper, $V = \{v_1, v_2, ..., v_n\}$ on a graph $G(V, E)$ is called by vertices set and the number of vertices is denoted by $n$. Besides vertices set, on graph $G(V, E)$, set of edge is denoted by $E = \{e_1, e_2, ..., e_m\}$ and $|E| = m$. The edges shows the connectivity of vertex $i$-th ($v_i$) and vertex $j$-th ($v_j$). A graph can be represent by a matrix that we called it as adjacency matrix, it is a symmetric matrix with size $n \times n$, where $n$ is number of vertex in a graph. In general, we can define the matrix adjacency as follow:

$$A = \begin{cases} a_{ij} = 1 & \text{if there is an edge } e_{ij} \\ a_{ij} = 0 & \text{if there is no edge} \end{cases}$$

Furthermore, a matrix adjacency can be transformed to weighted matrix by calculating weight for every edge on the graph, weight is calculated using Euclidean distance of $v_i$ ($i^{th}$ vertex) and $v_j$ ($j^{th}$ vertex) which is denoted by $c_{ij}$. $c_{ij}$ shows the distance of $v_i$ ($i^{th}$ vertex) and $v_j$ ($j^{th}$ vertex), so $c_{ij}^2 = \sum_{i=1}^{n}(v_i - v_j)^2$. Then, weighted matrix can be written as (2):

$$c_{ij} = \sum_{i=1}^{n}(v_i - v_j)^2$$
Let $d_v$ denote the degree of vertex $v$ which represent the number of edge on vertex $v$. The set of degree of vertex can be showed as a matrix, then it called by degree matrix, denoted by $D$, $d(v_i) = \sum_{v_j \sim v_i} e_{ij}$ and $D = \text{diag}(d(v_1), ..., d(v_n))$ or $D = \sum_{v_i \sim v_j} w_{ij}$. $W$ is a weighted matrix of graph.

### 2.2 Graph Laplacian

Graph Laplacian is an important role for 3D mesh reconstruction, so we will see about Laplacian operator. In this paper, 3D mesh geometric reconstruction represented by Laplacian Embedding which the main idea to simplify 3D mesh reconstruction problem. A large graph with many vertices on 3D mesh must be complicated on Laplacian eigenbases calculating problems. Therefore, dimensionality reduction is selected to simplify our problems. 3D mesh reconstruction result must retain the connectivity of graph.

Let $I \in \mathbb{R}^{n\times n}$ is identity matrix, $D \in \mathbb{R}^{n\times n}$ is degree matrix, $W \in \mathbb{R}^{n\times n}$ is weighted matrix, the random-walk Laplacian matrix [4] is given by (3):

$$W = \begin{cases} \frac{1}{c_{ij}^2}, & \text{there is an edge between vertex } i^{th} \text{ and vertex } j^{th}, e_{ij} \\ 0, & \text{there is no edge} \end{cases}$$  \hspace{1cm} \text{(2)}

or it can be written as (4),

$$L = \begin{cases} 1 - \frac{w_{ii}}{d_i}, & i=j \text{ and } d_{ii} \neq 0 \\ \frac{w_{ij}}{d_i}, & \text{there is an edge} \\ 0, & \text{others} \end{cases}$$  \hspace{1cm} \text{(4)}

The weighted matrix and the degree matrix of geometry 3D mesh is shown on (5) with the number of vertex of geometry 3D mesh is $n$, and the number of faces of geometry 3D mesh is $m$.

$$W = \begin{bmatrix} w_{11} & \cdots & w_{1n} \\ w_{21} & \cdots & w_{2n} \\ \vdots & \ddots & \vdots \\ w_{n1} & \cdots & 0 \end{bmatrix} \quad \text{and} \quad D = \begin{bmatrix} d_{11} & 0 & \cdots & 0 \\ 0 & d_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_{nn} \end{bmatrix}$$  \hspace{1cm} \text{(5)}

### 2.3 Eigenvalues and Eigenvectors

Vertices on 3D mesh is important to knowing the right positions of the 3D mesh. 3D mesh can represented by vertex and face. The vertices on 3 dimension represented by $3 \times n$ matrix, that rows show XYZ axis of Cartesian coordinates, columns show vertex indices. The face contains indices of vertex that build face in sequence. Laplacian matrix can be obtained by (6)

$$L \mathbf{u} = \lambda \mathbf{u}$$  \hspace{1cm} \text{(6)}

Where, $\lambda$ is an eigenvalue and $\mathbf{u}$ is an eigenvector. For every $\mathbf{f} \in \mathbb{R}^n$, we have $f^t L f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} (f_i - f_j)^2$, where $w_{ij}$ is scalar. $L$ is symmetric and positive semi-definite, the smallest eigenvalue of $L$ is 0, and the corresponding of eigenvector is the constant one vector. $L$ has non-negative, real valued eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$. $A$ is a symmetric matrix which shows the connectivity of all the vertices from the connected graph. A called by adjacency matrix. 3D mesh with $n$ vertices has $n$ real eigenvalues and $n$ real eigenvectors form an orthonormal basis. $\{\lambda_1, \lambda_2, ..., \lambda_n\}$ are eigenvalues of the adjacency matrix and $\{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_n\}$ are eigenvectors. All the eigenvectors corresponding with eigenvalues. The eigenspace $S_i = \{\mathbf{u} \in \mathbb{R}^n | A \mathbf{u} = \lambda_i \mathbf{u}\}$ where $1 \leq i \leq n$ represent set of the eigenvectors [7]. Then (7), $A$ shows set of the eigenvalue of $L$ and $U$ shows set of the eigenvectors.
2.4 Dimensionality Reduction

In this paper, we will reduce the dimension of Laplacian matrix by reducing the dimensionality of spectral graph. Spectral dimensionality reduction step is for reducing the number of spectral on 3D mesh approximation. The reduction of computational of approximation should retains the connectivity graph. The 3D mesh geometric reconstruction results must be show the 3D mesh approximation that represent the original 3D mesh. By embedding the graph into the spectral space, we can get 3D mesh reconstruction. By projecting all vertices in the graph into \( k \) dimension of spectral space. Graph that represent 3D mesh as a geometric form is a finite graph. On [1], every finite graph can be embedded in \( k \) Euclidean space \((\mathbb{R}^k)\) if \( k \geq 3 \). We use \( k \) spectral information of 3D mesh for 3D mesh reconstruction purposing. \( k \) is the independent variable, denotes how many spectral information that used for 3D mesh reconstruction purposing by spectral dimensionality reduction. Because of reducing dimension (spectral), it will makes efficient on computational 3D mesh reconstruction processing.

Assume that we have one connected graph of 3D mesh. 3D mesh have the Laplacian matrix and we can get the eigenvalue matrix and eigenvector matrix. \( \Lambda \) shows the original eigenvalues matrix after spectral dimensionality reduction by \( k \) information on (8). And \( U \) shows the original eigenvectors matrix after spectral dimensionality reduction by \( k \) information on (9).

\[
\Lambda = \begin{bmatrix}
\lambda_1 & 0 & \cdots & 0 \\
0 & \lambda_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \lambda_n 
\end{bmatrix}
\quad \text{and} \quad
U = \begin{bmatrix}
u_1(v_1) & u_2(v_1) & \cdots & u_n(v_1) \\
u_1(v_2) & u_2(v_2) & \cdots & u_n(v_2) \\
\vdots & \vdots & \ddots & \vdots \\
u_1(v_n) & u_2(v_n) & \cdots & u_n(v_n)
\end{bmatrix}
\]

(7)

(8)

(9)

We obtain \( n \times k \) matrix which \( u_i^T u_j = \alpha_{ij} \) (ortthonormal vectors, \( i, j = 1, \ldots, k \)), hence \( U^TU = I_k \). The eigenvalue of Laplacian matrix is called by spectral Laplacian. The spectral Laplacian considered approximate the mesh using linear combination of a number of basis vectors. On [6] \( u_2 \) is called Fiedler vector. The first eigenvalue is zero that corresponding by first eigenvector \( u_1 = [1, 1, 1, \ldots, 1] \) (size \( n \)). Eigenvalues of Laplacian matrix corresponding by geometry of 3D mesh are called spectral graph for 3D mesh. The first eigenvalue (\( \lambda_1 \)) is 0, the first column of eigenvector reduced and we have \( \{u_1, u_2, \ldots, u_k\} \) as an eigenvector. Therefore, to build Laplacian Embedding, spectral dimensionality reduction of Laplacian eigenbases by \( k \) information are shown on (8).

3. 3D Mesh Reconstruction

In the 3D mesh reconstruction, we approximate all vertices to get the new vertices which will reconstruct 3D mesh. We project the original geometry of 3D mesh into \( k - 1 \) dimension, and vertices approximation obtained by back projection into 3 dimension on approximation stage. The vertices approximation are reconstructed. Inverse the L-embedding is our work to reconstruct 3D mesh so that we should not have to adding anchors to the original Laplacian matrix like [15]. The set \( V \) (vertices)
represents the geometrical information of the mesh and the set \( E \) (edges) represents the connectivity information. \( \mathbf{v} = (x, y, z) \in \mathbb{R}^3 \) in Cartesian coordinates shows the geometry of 3D mesh. The 3D mesh embed into \( k - 1 \) dimension, so we obtain three vectors of \( k - 1 \) dimension which called spectra or pseudo-frequencies \((x_p, y_p, z_p)\) correspond to the 3D mesh original vertices. Since \( \lambda_1 \) is zero, \( \lambda_1 \) is eliminated from the eigenvalues matrix. On this 3D mesh reconstruction, the embedding method is reducing dimensionality by eliminate null eigenvalue (first eigenvalue) but retain the corresponding eigenvector (first eigenvector). Besides that, rearrange the column of eigenvectors matrix with the result that \( \{u_1, u_{k-1}, \ldots, u_1\} \) and eliminated the first column, so eigenvector \( u_k \) is eliminated. (10) shows the new eigenvalues matrix, and (11) shows the new eigenvector matrix.

\[
Y = \begin{bmatrix}
\lambda_2 & 0 & \cdots & 0 \\
0 & \lambda_3 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \lambda_k
\end{bmatrix}
\]

\[
M = \begin{bmatrix}
u_{k-1}(v_1) & u_{k-2}(v_1) & \cdots & u_1(v_1) \\
u_{k-1}(v_2) & u_{k-2}(v_2) & \cdots & u_1(v_2) \\
\vdots & \vdots & \ddots & \vdots \\
u_{k-1}(v_n) & u_{k-2}(v_n) & \cdots & u_1(v_n)
\end{bmatrix}
\]

The number of eigenfunctions must be \( 3 \leq k \leq n - 1 \) which can be implied to embed the graph. The Laplacian Embedding of a graph as an operator to embed the original 3D mesh into \( k - 1 \) dimension can be viewed as (12).

\[
\ell = Y^{-1/2} M^T
\]

Therefore, the embedded coordinates of a vertex \( v_j \) are (13):

\[
\ell_j = \begin{bmatrix}
u_{k-1}(v_j) \\
\sqrt{\lambda_2} \\
\vdots \\
\sqrt{\lambda_k}
\end{bmatrix}
\]

The projected vertices of \( k \)-dimension can be obtained by multiplying the Laplacian Embedding matrix with the coordinates (14):

\[
x_p = \ell x \\
y_p = \ell y \\
z_p = \ell z
\]

The Laplacian embedding is not symmetric matrix, so we calculate its pseudo-inverse for equation (12) to get the approximation of 3D mesh information. The new vertices or an exact reconstruction are \( \hat{\mathbf{x}} = x_p^T \ell^+, \hat{\mathbf{y}} = y_p^T \ell^+, \hat{\mathbf{z}} = z_p^T \ell^+ \). Where \( \ell^+ \) represents the pseudo-inverse of (12). That new vertices are used to reconstruct 3D mesh, such that the numerical approximation error of 3D mesh reconstruction are small and low complexity of algorithm.

4. Experimental Results

In this study, we show three kinds of 3D mesh that has 4315, 5103, and 24955 vertices. We use OFF file format for vertices and faces detail information. In Figure 2 we approximate the 3D mesh of cow. The geometry of 3D mesh cow has 4315 vertices and 8626 faces, the original mesh shows in Figure 2.
Figure 2. Cow original mesh has 4315 vertices.

The 3D mesh geometric reconstruction of cow by reducing spectral $k = 216, 863, 1726$, and $2158$. These $k$ parameters are equal to 5%, 20%, 40%, and 50% as approximation ratio. Therefore, the reconstruction of 3D mesh of cow results show in Figure 3.

![Cow reconstructions](image)

Figure 3. Geometric reconstructions of 3D mesh cow with $k=216$ (a), 863 (b), 1726 (c), and 2158 (d) the increasing number of eigenfunctions.

Hereafter we will see the other 3D mesh, such as homer and elephant. Homer has 5103 vertices and 10202 faces. Figure 4 shows the original 3D mesh of homer with 5103 vertices.

![Homer](image)

Figure 4. Homer original mesh has 5103 vertices.

The 3D mesh geometric reconstruction of homer by reducing spectral $k = 1021, 2041$, and $2552$. These $k$ parameters are equal to 20%, 40%, and 50% as approximation ratio. Then, the reconstruction of 3D mesh of homer results show in Figure 5.

![Homer reconstructions](image)

Figure 5. Geometric reconstructions of 3D mesh homer with $k=1021$ (a), 2041(b), and 2552 (c) the increasing number of eigenfunctions.

Figure 6 shows the original 3D mesh of elephant that has 24955 vertices and 49918 faces.

![Elephant](image)

Figure 6. Elephant original mesh has 24955 vertices.
The 3D mesh geometric reconstruction of elephant by reducing spectral \( k = 250, 499, 749 \) and 998. These \( k \) parameters are equal to 1%, 2%, 3%, and 4% as approximation ratio. Then, the reconstruction of 3D mesh of homer results show in Figure 7.

![Geometric reconstructions of 3D mesh elephant with \( k = 250 \) (a), 499 (b), 749 (c), and 998 (d) the increasing number of eigenfunctions.](image)

These reconstruction results are closely enough to described the original 3D mesh as the approximation of all 3D mesh. The connected graph on the 3D mesh are retained on 3D mesh reconstruction.

### 4.1 Evaluation methods

On 3D mesh geometric reconstruction, certainly there is error on approximation results, since generated 3D mesh is just simply a fiction of the original object. [15] evaluate the approximation result by metro tool [5]. In this paper discussed error calculation that occurred in the approximation of geometric 3D mesh with three kinds of error. Three kinds of error are geometric error, differential error, and final error. On error calculation, there are \( M_1 \) which represents the original 3D mesh, while \( M_2 \) represents the approximation 3D mesh. (15) shows geometric error formula.

\[
\begin{align*}
g_e &= \|M_2 - M_1\|_g = \sum_{i=1}^{n} \frac{1}{2} \left\| v_i^1 - v_i^2 \right\|^2_2 \\
\end{align*}
\]

with

- \( n \) shows the number of 3D mesh vertices
- \( v_i^1 \) shows the original vertex of 3D mesh \( (v) \) \( i \)th, \( 1 \leq i \leq n \)
- \( v_i^2 \) shows the original vertex of 3D mesh \( (\overline{v}) \) \( i \)th, \( 1 \leq i \leq n \)

[8] introduce Geometric Laplacian (GL) then calculate differential error between both vertex (original vertex and approximation vertex) that showed on (16).

\[
\begin{align*}
GL(v_i^1) &= v_i^1 - \frac{\sum_{j \in \mathcal{N}(i)} l_{ij} v_j^1}{\sum_{j \in \mathcal{N}(i)} l_{ij}^2}, \\
\end{align*}
\]

\( l_{ij}^{-1} \) shows inverse of the edge length between \( v_i \) and \( v_j \), such that differential error can be written as (17).

\[
\begin{align*}
d_d &= \|M_1 - M_2\|_d = \sum_{i=1}^{n} \frac{1}{2} \left\| GL(v_i^1) - v_i^2 \right\|^2_2 \\
\end{align*}
\]

[17] used evaluation method of their compression results by the average of geometric error and differential error which are called final error \( (f_e) \) on (18).

\[
\begin{align*}
f_e &= \|M_1 - M_2\| = \frac{1}{2} \left( \|M_1 - M_2\|_g + \|M_1 - M_2\|_d \right) \\
\end{align*}
\]

All our experiments run on computer with core i7, 2.20 GHz, and RAM 4 GB as specifications. To compute the eigenvalue and the eigenvector, we used MATLAB. We compare three kinds of 3D mesh in this study, such as cow, homer, and elephant that are .OFF file format. On .OFF file format, one can know about the number of vertices of the object of 3D mesh. Table 1 shows error approximation for 3D mesh cow reconstructions, which are showed in Figure 2 with 4315 vertices by some ratio, 5%, 20%, 40%, and 50%. The evaluation dimensionality reduction method of 3D mesh cow reconstruction by Laplacian Embedding include geometric error, differential error, and final error.
Table 1. Error approximation for 3D mesh cow with 4315 vertices.

| Ratio | Geometric Error | Differential Error | Final Error   |
|-------|-----------------|-------------------|---------------|
| 5%    | 1.47e-4         | 2.24e-4           | 1.86e-4       |
| 20%   | 0.94e-4         | 1.47e-4           | 1.21e-4       |
| 40%   | 1.08e-4         | 1.31e-4           | 1.19e-4       |
| 50%   | 1.05e-4         | 1.20e-4           | 1.12e-4       |

Table 2 shows error approximation for 3D mesh homer reconstructions, that are showed in Figure 4 with 5103 vertices by some ratio 20%, 40%, and 50%. The evaluation dimensionality reduction method of 3D mesh homer reconstruction include geometric error, differential error, and final error.

Table 2. Error approximation for 3D mesh for 3D mesh homer with 5103 vertices

| Ratio | Geometric Error | Differential Error | Final Error   |
|-------|-----------------|-------------------|---------------|
| 20%   | 2.62e-5         | 2.50e-5           | 2.56e-5       |
| 40%   | 2.16e-5         | 2.01e-5           | 2.08e-5       |
| 50%   | 3.38e-5         | 2.89e-5           | 3.13e-5       |

Table 3 shows error approximation for 3D mesh elephant reconstructions, which are showed in Figure 6 with 24955 vertices by some ratio, 1%, 2%, 3%, and 4%. The evaluation dimensionality reduction method of 3D mesh elephant reconstruction include geometric error.

Table 3. Error approximation for 3D mesh elephant

| Ratio | Geometric Error |
|-------|-----------------|
| 1%    | 1.75e-5         |
| 2%    | 1.26e-5         |
| 3%    | 8.74e-6         |
| 4%    | 7.60e-6         |

5. Conclusions
We have developed the new method to 3D mesh reconstruction which derive basis from mesh connectivity by reducing dimensionality on null eigenvalue but retain the corresponding eigenvector. The main idea is dimensionality reduction of Laplacian Embedding by reducing any spectral but retain the connectivity of graph as 3D mesh. Built the Laplacian embedding by eliminating the first eigenvalue, rearrange the column of eigenvector matrix, and eliminating the first column of the rearrange eigenvector matrix result. By embedding all the original vertices, the approximation vertices can be obtained by multiplying the projected vertices and pseudo-inverse of Laplacian embedding. The result of dimensionality reduction on 3D mesh reconstruction is for simplify reconstruction problems and computational eficiency. Since reducing dimension ($k$ spectral) which $3 \leq k \leq n - 1$, we made efficient on computational by Laplacian embedding without adding anchors on 3D mesh geometric reconstruction processing.
References

[1] Brouwer, A.E., Haemers, W.H. 2011. Spectra of Graph. New York : Universitext-Springer
[2] Cayre, F., Rondao-Alface, P., Schmitt, F., Macq, B., Malître, H. 2003. Application Of Spectral Decomposition To Compression And Watermarking Of 3D Triangle Mesh Geometry. Elsevier Science, 18:309-319.
[3] Chen, D., Cohen-Or, D., Sorkine, O., Toledo, S. 2005. Algebraic Analysis of High-pass Quantization, ACM Transaction on Graphics, Vol.24(4), pp. 1259-1282.
[4] Chung, F.R.K. 2000. Spectral Graph Theory. CBMS. American Mathematical Society.
[5] Cignoni, P., Rocchini, C., Scopigno, R. 1998. Metro : Measuring Error on Simplified Surfaces. Computer Graphics Forum, Vol. 17, no. 2, pp. 167-174.
[6] Fiedler, M. 1973. Algebraic Connectivity of Graphs. Czechoslovak Mathematical Journal, Vol. 23, No. 2, pp. 298-305.
[7] Horaud, R. 2012. A Short Tutorial On Graph Laplacians, Laplacian Embedding, And Spectral Clustering. INRIA.
[8] Karni, Z., Gotsman, C. 2000. Spectral Compression of Mesh Geometry. Proceedings of ACM SIGGRAPH 2000, pp. 279-286.
[9] Khatun, A., Chai, W. Y., Iskandar, DNF. A., Islam, M.R. 2012. An Ellipsoid Shape Approximation Approach for 3D Shape Representation. Proceeding of the IIEJ Image Electronics and Visual Computing Workshop. Kuching, Malaysia.
[10] Lévy, B. 2006. Laplace-Beltrami Eigenfunctions Towards An Algorithm That “Understands” Geometry. INRIA.
[11] Lipman, Y., Sorkine, O., Cohen-Or, D., Levin, D., Rössl, C., Seidel, H.P. 2004. Differential Coordinates for Interactive Mesh Editing. Proceedings of EUROGRAPHICS/ACM SIGGRAPH Symposium on Geometry Processing (SGP), pp. 42-51.
[12] Sharma, A., Horaud, R., Mateus, D. 2012. Chapter 13 : 3D shape registration using spectral graph embedding and probabilistic matching. Image Processing and Analysis with Graphs : Theory and Practice. CRC Press
[13] Sorkine, O. 2005. State-of-the-art : Laplacian Mesh Processing. Proceeding of EUROGRAPHICS.
[14] Sorkine, O. 2006. Differential Representations For Mesh Processing. Computer Graphics Forum, Vol. 25(4), pp. 789-807.
[15] Sorkine, O., Cohen-Or, D., Irony, D., Toledo, S. 2005. Geometry-aware Bases for Shape Approximation. IEEE, vol 11., No. 2.
[16] Toledo, S. 2003. TAUCS : A Library of Sparse Linear Solvers, version 2.2. Tel-Aviv University.
[17] Zhong, M., Qin, H., 2014. Sparse Approximation of 3D Shapes via Spectral Graph Wavelets. USA : Springer.