Finding Robust Vertices for 3D Synchronization Based on Euclidean Minimum Spanning Tree

N. Tournier[a, b], W. Puech[a], G. Subsol[a] and J-P. Pedeboy[b]

[a] Université de Montpellier II, LIRMM, UMR CNRS 5506,
161 rue Ada, 34095 Montpellier, France;

[b] Stratégies S.A., 41-43 rue de Villeneuve,
Parc d'affaires SILIC - BP 80429, 94583 Rungis Cedex, France

ABSTRACT

Synchronization in 3D data hiding is one of the main problems. We need to know where we can embed information, and be able to find this space in order to extract the message. Various algorithms propose synchronization techniques by triangle or vertex path in a 3D mesh. In this paper, we proposed a new synchronization technique based on Euclidean minimum spanning tree computing (EMST) and the analysis of the displacement of the vertices without moving the connections in the tree. Based on the analysis of the vertices, we select the most robust vertices and synchronize these areas by computing a new EMST called "robust EMST". Then, we analyze the robustness of the technique, i.e. the stability of the most robust vertices selection; and demonstrate the consistence of the criterion selection with the vertex displacement.

Keywords: Feature Vertices, 3D Synchronization, 3D Watermarking, Graph Theory

1. INTRODUCTION

3D multimedia content is now everywhere, in the industry for modeling and design; in the entertainment for video gaming, cinema, in cultural heritage for 3D interaction, archiving, etc. Thus, large quantities of multimedia data are exchanged which raises some research challenges, in particular the protection of 3D models.

There are different ways to protect multimedia file. On one hand, we can choose to encrypt the content (cryptography) and the communication will be secured until the customer decodes the informations. On the other hand, we can embed a hidden message (steganography and watermarking) in the host signal (i.e. 3D model).

The principle of data hiding is illustrated on Fig. 1. We can divide the insertion process in two parts: synchronization and embedding.

Synchronization allows one to know where the encoded message is embedded in the host signal. The aim is to be able to recognize the same subspace, called the insertion subspace, before and after the watermarking process.

One of the major problems of 3D data hiding is synchronization. At the opposite of audio or image watermarking, there is no manner to find a "natural" path in a 3D model. The idea is to define path of vertices or facets in a 3D mesh. Ohbuchi *et al.*[1] propose to double a band of triangles depending on the message to embed. This method is not secure because it is very easy to find the message by identifying the double triangles.

Other approaches[2–4] propose to scan the mesh by defining deterministic sequence of triangles, independent of the message and of the insertion process. But, the synchronization process requires to choose an initial triangle. Moreover these approaches are based on the connectivity of the mesh; therefore it is very fragile to topology modification such as remeshing. A recent survey of 3D embedding techniques was proposed by Wang *et al.*[5]

Further author information: (Send correspondence to N.T., W.P, or G.S)
E-mail: {nicolas.tournier, willam.puech, gerard.subsol}@lirmm.fr

Three-Dimensional Imaging, Interaction, and Measurement, edited by J. Angelo Beraldin, et al.,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 7864, 78640W · © 2011 SPIE-IS&T
CCC code: 0277-786X/11/$18 · doi: 10.1117/12.872384

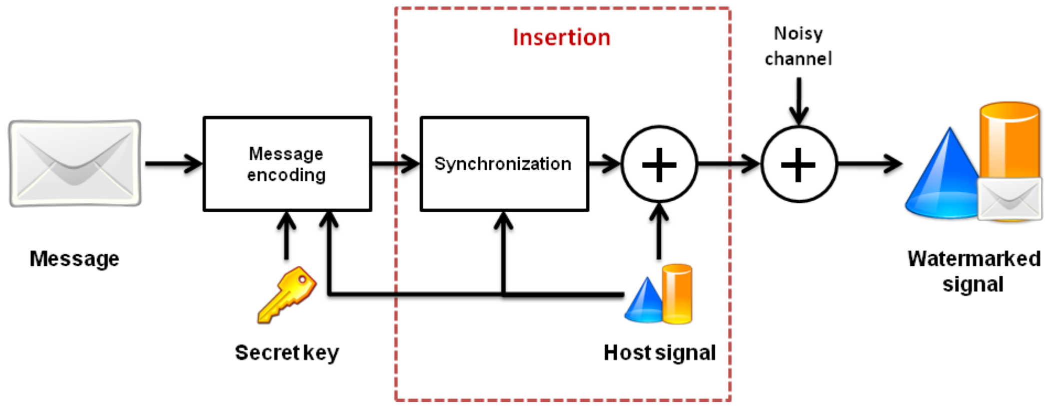

Figure 1. Watermarking Scheme.

In this paper, we focus on the 3D synchronization and propose an improvement of a previous technique based on the Euclidean minimum spanning tree (EMST).[6] EMST defines a structure that depends only on the vertices. This synchronization method does not move any vertex; it is robust to the modification of the mesh topology (in particular, does not depend on the surface reconstruction process ; only the point position are necessary), but not to vertex decimation and remains very fragile to perturbation of the position of vertices.

In Section 2, we describe our synchronization technique and in Section 4, we present our experimental results and discuss then in Section 3.

2. FINDING VERTICES TO BUILD A ROBUST EMST

We give prominence to the possible displacement of a vertex without changing the connections in the Euclidean minimum spanning tree (EMST). The problem is also called the MST sensitivity problem. The aim is to know an interval of possible values on a graph edge without changing any connection in the MST.[7–9]

In our case we had analyzed the sensitivity of EMST.[10] The problem is really different because if one vertex is moved, all the incident edges are modified. From this study, we proposed a way to quantify the possible displacement of a vertex.

By this analysis, we need to know if these regions are robust, in the watermarking point of view, and how it is possible to synchronize them. We divide the problem in two parts. First, we need to characterize the most robust areas. So we need to find a criterion as a function of the displacement radius (r_i^+ and r_i^-) computed in.[10] Secondly, until the robust areas are identified, we need to synchronize them. In other words, we need to find a unique path between them. This path must also be robust to be able to find the correct message embedded.

We need to find a robust criterion to quantify the displacement of the vertices. Moreover we must have a good correlation with the perturbation of the vertices by the Gaussian noise addition with a standard deviation σ, in order to be able to find the most robust vertices again.

From r^+ and r^-, illustrated Figure 2, it is easy to deduce a magnitude that quantifies the displacement of the vertices along their straight line. Let us denote by r this magnitude:

$$r_i = \min\{r_i^+, r_i^-\}.$$

Nevertheless, in[10] we consider only one degree freedom displacement, along a straight line. So we need to verify if this criterion can be an approximation of the 3D vertex displacement. From r and σ, we need to find a criterion correlated to the Gaussian noise. We denote by $r_{x\%}$ the value such as $x\%$ of the vertices verify $r_i > r_{x\%}$. That represents a threshold to the most robust vertices. This value will be an estimation of the robustness of the vertices.

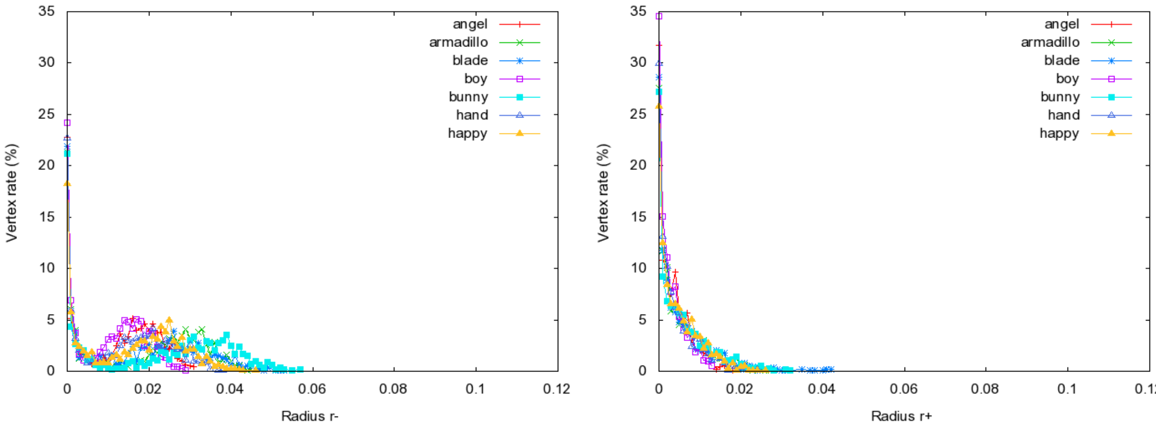

Figure 2. Distribution of the radius r^- and r^+. r^- stands for the distance which the vertex can be moved from its father and r^+ which the vertex can be moved away. More the distance is important, more the vertex is mobile.

The question is to know if the more robust vertices in the EMST are robust in the synchronization point of view. In other words, we want to be able to find the same vertices or the same areas after the perturbation by a Gaussian noise addition.

With this study, we are able to locate the most mobile vertices, and to synchronize these areas by find a path in the EMST of the robust vertices. The synchronization scheme is illustrated Figure 3 on the 3D mesh *Horse*, composed by 1000 vertices, and normalized such as a unitary bounding box can contain it. In the following Section, we present our experiment results that prove the improvement of synchronization techniques based on EMST.

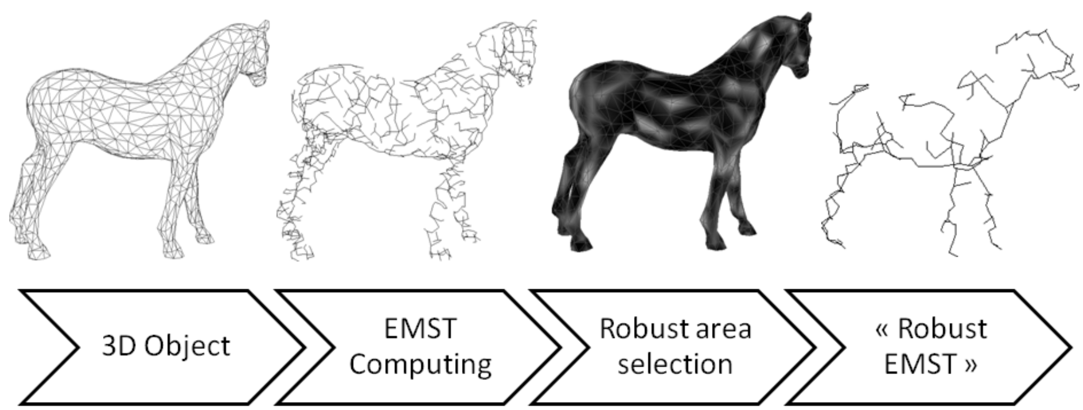

Figure 3. Representation of the synchronization process on *Horse*. We illustrate respectively from the left to the right: the 3D mesh, the EMST computed by Prim's algorithm,[11] the selection of the most mobile vertices represented in light colors, the synchronization of the most robust area by an other EMST computing. We called this tree, the "robust EMST".

3. EXPERIMENTAL RESULTS

We experiment our theory on one dozen of 3D meshes composed by 1000 vertices. We want to estimate the robustness of the synchronization process, i.e. the selection of the more robust vertices and in the same time the stability of the "robust EMST". Each mesh is normalized such as a unitary bounding box can contain it so the normalized factor equals to:

$$\max\{\Delta_x, \Delta_y, \Delta_z\},$$

with $\Delta_x = x_{max} - x_{min}$.

We illustrate on Fig. 4 the experimentation. From a 3D mesh, first we compute its EMST by the Prim's[11] algorithm. Then we are able to compute a quantification of the possible displacement of the vertex without changing any connection in the EMST.[10] The most mobile vertex are the more robust one. We select $x\%$ of the most robust vertices (in the experimentation we set x to 20%), the most mobile vertices in the EMST where $r_i > r_{x\%}$. And we compute the "robust EMST" on the set of the robust vertices.

We apply the same treatment on a mesh where each vertex position is disturbed by an additive Gaussian noise. Then we want to compare the "robust EMST" from the original mesh and the "robust EMST" from the perturbed mesh.

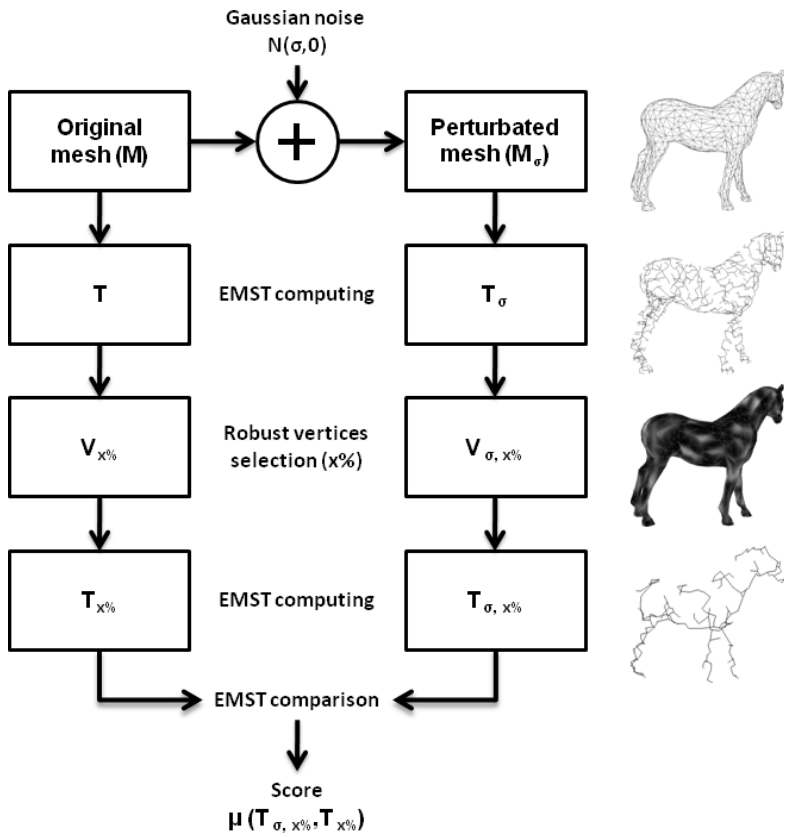

Figure 4. Overview of the experimental test of robustness.

We match each edge of T_σ to $T_{\sigma,x\%}$ such as we are able to compare the connections in the trees. Then we compute the number of common edges between T_σ and $T_{\sigma,x\%}$. We denote by $\bar{\mu} = \mu(T_\sigma, T_{\sigma,x\%})$ this measure.

We illustrate Figure 5, the results of the common edge rate ($\bar{\mu}$) of the "robust EMST" in average for (20 iterations) as a function of Gaussian noise intensity (σ).

For the majority of the tested models we preserve a good synchronization level until a Gaussian noise with a standard deviation σ around 10^{-5} or 10^{-4}. This is really interesting, at the opposite for a synchronization based only on the EMST[6] in which we are desynchronized for small intensity noise (around 10^{-7}, 10^{-6}) for the same models. Therefore we improve the synchronization robustness.

The robustness can be explain by the correlation between $r_{x\%}$ and the intensity of the Gaussian noise. Let $\sigma_{x\%}$ is the critical standard deviation of the Gaussian noise such as $\mu(T, T_\sigma) = x\%$ in average. By the experience, we take one dozen of 3D models composed of thousands vertices and fix x to compute for each mesh $r_{x\%}$ and $\sigma_{x\%}$ for around 20 experimentations.

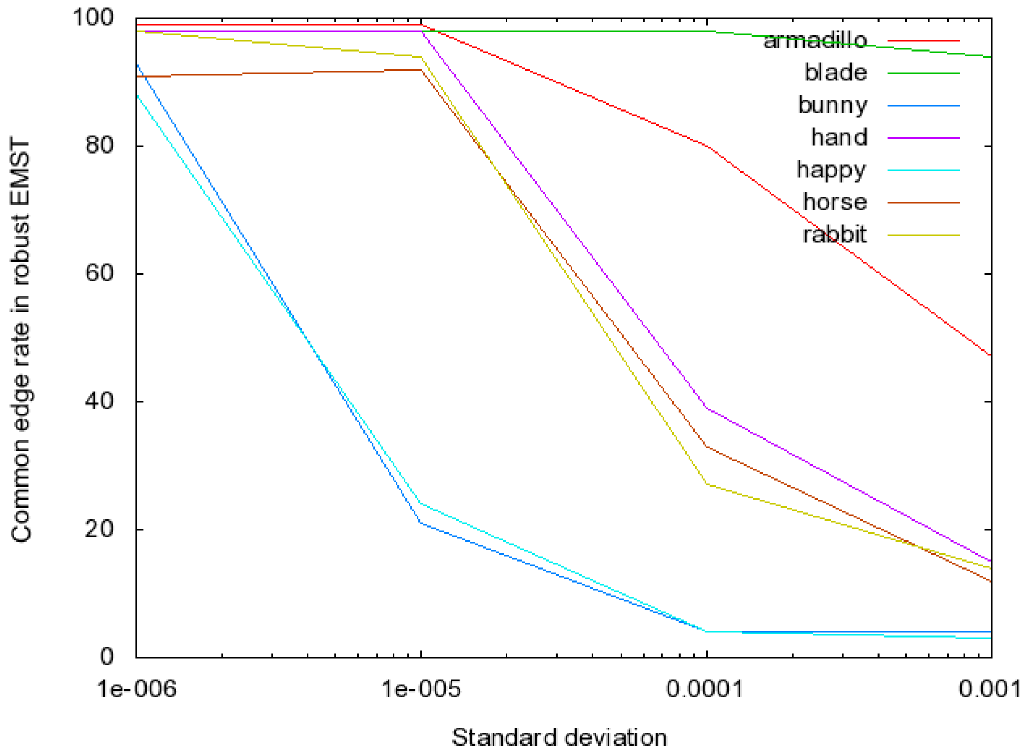

Figure 5. Representation of the common edge rate in robust EMST $\bar{\mu}$ as a function of the standard deviation σ of the Gaussian noise realized for 20 experiments with $x = 20\%$.

According to Figure 6, we represent, for different selection rate $x\%$, the critical standard deviation as a function of our theoretical criterion. For each selection rate we obtain a straight line. Indeed, it exists a linear relation between $r_{x\%}$ and $\sigma_{x\%}$ that does not depend on x:

$$\sigma_{x\%} = k \cdot r_{x\%}.$$

By linear regression we estimate k the value of the coefficient of the linear correlation around $1/3$ for $x \in \{10; 20; 30\}$. In future work, it will be interesting to formalize this study in theoretical context. This kind of techniques should be interesting with various criteria that we are studying, such as the estimation of the discrete curvature. The aim is to find a stable criterion in order to use the synchronization in a robust watermarking scheme.

4. CONCLUSION

We propose an improvement of the synchronization by EMST.[6] Thanks to the analysis of the displacement of the vertices without changing the connections in the tree,[10] we propose a theoretical criterion in consistence with the Gaussian noised intensity in order to select robust areas.

To synchronize them, we compute a new EMST based only on the most robust vertices. This EMST is more robust to noise addition attack. So, from the previous method,[6] the synchronization process is more robust. But if we used the same embedding techniques in the most robust areas only, obviously the capacity decreases because we select only $x\%$ of vertices (with $x \in [0; 30]$).

Nevertheless the method is too complex, time computation is quadratic as a function of the number of vertices. We need to work on low resolution of multi-resolution models or to find criterion easier to compute for voluminous 3D meshes.

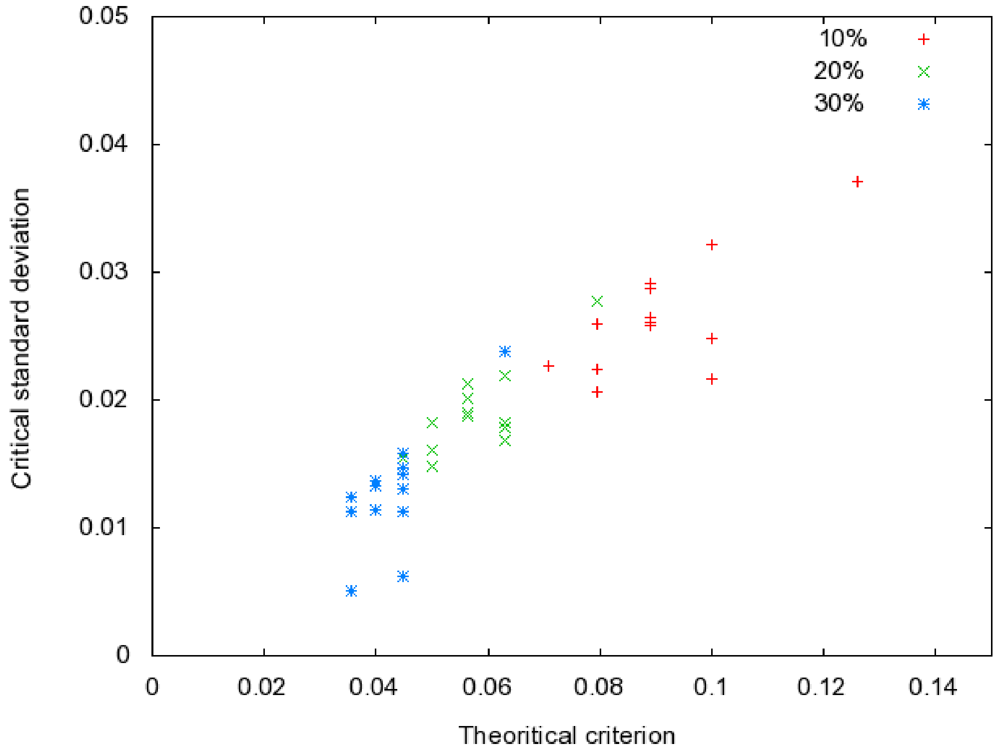

Figure 6. Correlation between our theoretical criterion $r_{x\%}$ and $\sigma_{x\%}$ the critical standard deviation.

For the first lead, we apply our criterion on different low resolution of semi-regular meshes. And our first results are quite encouraging. The most robust areas seems to be in the same places for a set of resolution. Inspired by the discrete LOD (Level Of Detail),[12, 13] we will decimate the mesh in a lower resolution to synchronize the watermark in this new resolution domain.

Our other lead is to find an other criterion based on the discrete curvature[14] computing. As we find the most mobile vertices in the EMST, we can study the vertices with high value of the average curvature. For complexity reason, the computing of the curvature is fast at the opposite of our criterion (quadratic time computing). With this criterion it will be easy to treat voluminous 3D mesh.

REFERENCES

[1] Ohbuchi, R., Masuda, H., and Aono, M., "Watermaking Three-Dimensional Polygonal Models," in [*Proceedings of the Fifth ACM International Conference on Multimedia*], 261–272, ACM New York, NY, USA (1997).

[2] Cayre, F. and Macq, B., "Spatial Watermarking of 3D Triangle Meshes," in [*Proceedings of SPIE*], **4472**, 155 (2001).

[3] Mao, X., Shiba, M., and Imamiya, A., "Watermarking 3D Geometric Models Through Triangle Subdivision," in [*Proc. SPIE*], **4314**, 253–260 (2001).

[4] Rossignac, J., "Edgebreaker: Connectivity Compression for Triangle Meshes," *IEEE transactions on visualization and computer graphics* **5**(1), 47–61 (1999).

[5] Wang, K., Lavoué, G., Denis, F., and Baskurt, A., "A Comprehensive Survey on Three-Dimensional Mesh Watermarking," *IEEE Transactions on Multimedia* **10**, 1513–1527 (2008).

[6] Amat, P., Puech, W., Druon, S., and Pedeboy, J.-P., "Lossless 3D Steganography Based on MST and Connectivity Modification," *Signal Processing: Image Communication* **25**, 400–412 (July 2010).

[7] Dixon, B., Rauch, M., and Tarjan, R., "Verification and Sensitivity Analysis of Minimum Spanning Trees in Linear Time," *SIAM Journal on Computing* **21**, 1184–1992 (1992).

[8] Gordeev, E., "Stability Analysis of the Minimum Spanning Tree Problem," *Computational Mathematics and Mathematical Physics* **39**(5), 738–746 (1999).

[9] Gordeev, E., "Stability Analysis in Optimization Problems on Matroids in the Metric l1," *Cybernetics and Systems Analysis* **37**, 251–259 (2001).

[10] Tournier, N., Puech, W., Subsol, G., and Pedeboy, J., "Sensitivity Analysis of Euclidean Minimum Spanning Tree," in [*Proceedings of SPIE*], **7526**, 75260G (2010).

[11] Prim, R., "Shortest Connection Networks and Some Generalizations," *Bell System Technical Journal* **36**, 1389–1401 (1957).

[12] Clark, J. H., "Hierarchical geometric models for visible surface algorithms," *Communications of the ACM* **19**(10), 547–554 (1976).

[13] Luebke, D., [*Level of detail for 3D graphics*], Morgan Kaufmann Pub (2003).

[14] Dong, C. and Wang, G., "Curvatures estimation on triangular mesh," *Journal of Zhejiang University-Science A* **6**, 128–136 (2005).