A DATA HIDING METHOD BASED ON THE TOPOLOGY CHANGE OF A 3D TRIANGULAR MESH

P. Amat [a,b], *W. Puech* [a], *S. Druon* [a] *and J.P. Pedeboy* [b]

[a] LIRMM Laboratory, UMR CNRS 5506, University of Montpellier II
161, rue Ada, 34392 MONTPELLIER CEDEX 05, FRANCE
[b] Stratégies S.A., 41-43 rue de Villeneuve, Parc d'affaires SILIC - BP 80429
94583 RUNGIS CEDEX, FRANCE
philippe.amat@lirmm.fr, william.puech@lirmm.fr, sebastien.druon@lirmm.fr, jp.pedeboy@cadwin.com

ABSTRACT

In this paper we present a new method of data hiding in 3D objects. This method is based on a 3D model represented by a cloud of vertices and a list of edges corresponding to the triangular mesh of the surface. The main idea of this method is to find and to synchronize particular areas that can be used to embed the message. The data hiding relies on the modification of the topology of edges in chosen areas. These modifications, as a consequence, have to change the structure of triangles constructed in these areas. This method has the advantage of not changing the position of the vertices of the 3D model. This invariant on the vertex positions allows us to carry out an embedding that is robust to the affine transformations of type rotation, translation or zoom.

1. INTRODUCTION

To embed hidden data in a 3D object the two most frequently used domains are the spatial domain and the spectral domain. It is worthwhile to note that there are other more original domains like the wavelet space [5] or the NURBS space [2]. The approach of data hiding presented in this article has the constraint of not changing the position of vertices in the 3D space. Because of this, we cannot depend on methods of spectral embedding such as that of Ohbuchi *et al.* [8] or that of Cayre *et al.* [4]. Spatial methods can be classified into two categories. The first one encompasses the spatial methods not using a geometric shape for data hiding. The method of Wagner [10] aims at embedding information in the relating length of Laplacian local vectors. For this, Wagner changes the local curve of the object. The *Vertex Flood* algorithm of Benebens [1] also belongs to this first category. This method rests on the modification of the distance of the acceptable vertices in the center of mass of the object. Method proposed by Bors uses the local geometric shape of vertices to choose embedding areas [3]. The area is represented by a vertex and all closely related triangles. In his approach Bors calculates an ellipsoidal neighborhood for this vertex. To encode a value in 0, the vertex is displaced outside its neighborhood and to encode a value in 1 it is brought inside. These methods are in most cases founded on the position of vertices, even if they sometimes utilize their neighborhood. This approach goes contrary to our restriction of not changing the position of vertices. The second category comprises the spatial methods which use the geometric shape of the object for data hiding. In this second category we find method of Ohbuchi *et al.* called *Triangle Similarity Quadruple* which offers the realization of an index-linked arrangement [7]. The index of the information hidden in a shape is explicitly encoded. The geometric shape is constituted of four triangles amongst which one is central and the three others are its neighbors. The central triangle, called M, serves for signaling the presence of hidden data. One of its neighbors, named S, contains the index of the triangle used to embed the bit of the message. The two other neighbors, named D_1 and D_2, serve alternatively for recording useful information. The data hiding is performed by changing pairs of comparisons. In the same article Ohbuchi *et al.* offers a second method called *Tetrahedral Volume Ratio* which uses the report of the volumes of two tetrahedrons [7]. For this method a reference tetrahedron is necessary to be able to bring back the volume of others. The used tetrahedrons are formed by an edge and both incident triangles. The traversing of the connectivity graph thus rests on the spanning tree of the vertices. The edges of the constructed tree serve for generating and classifying tetrahedrons as well as their traversal order. In this second category of spacial methods with geometric shape we can also present the method of Cayre and Macq in which the projection of the top of a triangle on its foundation is used to perform watermarking [6]. The selection of the embedding area is then made by a unique path of 3D mesh with the aid of a secret key. These second category algorithms present feature to inscribe, along with useful information, information of localization of mark on 3D mesh.

2. THE PROPOSED METHOD BASED ON THE PROJECTION OF QUADRUPLES

In this section we detail a new method of data hiding in 3D objects. This method is based on a 3D model represented by a cloud of vertices, as illustrated in Figure 1 and a list of edges corresponding to the triangular mesh of the surface.

The main idea of this method is to find and to synchronize particular areas that can be used to embed the message. The data hiding relies on the modification of the topology of edges in chosen areas. These modifications, as a consequence, have to change the structure of triangles constructed in these areas. This method has the advantage of not changing the position of the vertices of the 3D model. This invariant on the vertex positions allows us to carry out an embedding that is robust to the affine transformations of type rotation, translation or zoom.

This data hiding method is based on the projection of the centers of embedding areas on a key axis. We call quadruple an area formed by two triangles of mesh having a common edge. This method is made up of three parts. The first part is the selection of the areas which are going to serve us to embed the data. These areas are chosen according to sev-

Figure 1: Cloud of initial vertices (453 vertices).

eral constraints. The second part is the synchronization of the message with the 3D model. For this, we depend on a key axis constructed in an invariant coordinate system. This coordinate system is constructed with the aid of a principal component analysis (PCA) and with an axis generated from a secret key. The center of every quadruple chosen for the data hiding is then projected on the axis. The third stage is the embedding of the message. For that, the topology of the triangles of the chosen quadruple is changed. The Figure 2 illustrates the overview of the method.

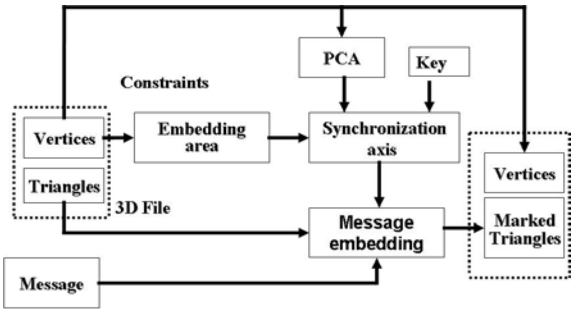


Figure 2: Overview of the proposed data hiding method based on the projection of quadruples.

2.1 Selection of the embedding areas

The search of the quadruples is done by going through all the triangles of the 3D model and by looking for every triangle the quadruple formed with the neighboring triangles by an edge. The quadruples will be validated as embedding zones as long as they comply to the constraints of coplanarity, convexity, overlapping and stability.

- Coplanarity constraint: the first constraint is the coplanarity of the quadruples. The fact to change the topology of both triangles forming one quadruple also changes the angle formed between these two triangles. Consequently, the data hiding on a non-coplanar quadruple affects the surface and therefore the visual aspect of the 3D model. In an ideal situation, the message should be embedded only in quadruples which are strictly coplanar. However, the number of quadruples complying strictly to this coplanar criterion is very limited. To increase the embedding capacity, we introduce in our approach a threshold S_c of tolerance on the coplanarity of the chosen quadruples. This threshold allows us to have a compromise between the embedding capacity and the quality of the 3D model. The higher the threshold, more will be the embedding capacity at the cost of visual degradation. Conversely, a lower threshold will mean a smaller embedding capacity and hence lower degradation in the visual quality of the 3D object. Through the threshold S_c, an adjustment is therefore possible between the embedding capacity and visual aspect of the 3D model. Let Q_{1234} be the quadruple formed by vertices P_1, P_2, P_3, P_4. At first, normal N to the plane P_{123} is calculated. The vertex P_4 is then orthogonally projected on the normal N. The distance which separates the projection of the vertex P_4 on N from plane P_{123} is called d_4. This distance is obtained for the projections of P_1, P_2 and P_3, respectively, on normals on planes P_{234}, P_{134} and P_{124}. The quadruple will be retained only if the highest calculated distance is less than the fixed threshold S_c:

$$Q_{1234} \text{ is retained if and only if } \min_{i=\{1,2,3,4\}}(d_i) < S_c. \quad (1)$$

An example of the measurement of d_4 for the coplanarity of a quadruple is illustrated in Figure 3.

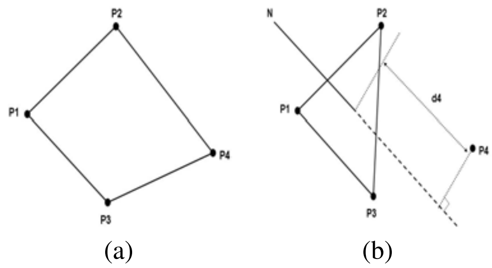


Figure 3: a) A quadruple, b) Measurement of d_4 for the coplanarity constraint.

- Convexity constraint: the second constraint imposes that the quadruples used for the data hiding are convex. For this constraint, we assume that the retained quadruples already obey the coplanarity constraint. The Figure 4 illustrates the case of a non-convex quadruple formed by the planes P_{123} and P_{234}. We note that the modification of the topology of triangles during the embedding of one bit changes the surface covered by the original quadruple. This modification adds an error on the surface of the quadruple. This error is represented by the surface P_{134} in Figure 4. It is to avoid this kind of error on the surface that the convexity constraint was introduced.

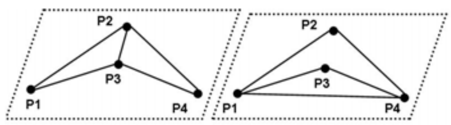


Figure 4: Example of embedding in a not convex area.

The measurement of this constraint is a combination of the following two procedures, respectively:

- Vectors $V_{i,i+1 \bmod 4}$ between vertices P_i et P_{i+1} of the quadruples are calculated for $i \in \{1,...,4\}$. Angles α_i between two successive vectors are then calculated.

  $$Q_{1234} \text{ is retained if and only if } \alpha_i < 180° \text{ for i } \in \{1,...,4\}. \quad (2)$$

  This allows to eliminate all quadruples having obtuse angles or three colinear vertices.
- The second measure prevents us from having quadruples too close to a triangular form when three out of four vertices are aligned. In that case the data hiding would make reveal too disproportional triangles. A threshold of tolerance S_t is chosen so that the proportion of both triangles of one quadruple respects a certain value. For this we calculate the length of segments $\overline{P_{23}}$ and $\overline{P_{14}}$ as well as distance D separating the centers of these two segments. A quadruple is then retained for the embedding if an only if:

  $$D \leq S_t \times \overline{P_{14}} \ AND \ D \leq S_t \times \overline{P_{23}}. \quad (3)$$

- Overlapping constraint: the third constraint concerns the overlapping of the quadruples. From the quadruples selected with the two previous constraints, there may remain quadruples with common vertices. When quadruples have at most one common vertex it poses no problem and in that case the quadruples are retained for data hiding. On the contrary if quadruples have more than one common vertex, the modification of topology of one of these quadruples during the embedding step, may disturb the second. To avoid this problem, a choice should be made among all the overlapped quadruples to retain only one of these. This choice is made in order to keep the quadruple which will affect less the 3D model. As a result the most coplanar quadruple will be chosen to embed the data.
- Stability constraint: the stability is a strong constraint for the hiding of data. In fact the change of topology of triangles for the data hiding creates new triangles and therefore new quadruples. These new quadruples may well be susceptible to the constraints imposed and thus create new candidates for embedding which should interfere with the data hiding process. We can see Figure 5 as an example of interference. At first in the Figure 5.a the quadruple Q_{1234} chosen for the embedding is represented by triangles T_{124} and T_{234}. We remark that triangles T_{234} and T_{235} also form the quadruple Q_{2534} which does not conform to the convexity constraint. The Figure 5.b presents the same area after an embedding in the quadruple Q_{1234} with a change in common edge. We also anticipate the possibility of the formation of the quadruple Q_{1253} with triangles T_{123} and T_{253} which comply with the constraints. This quadruple can then interfere with the previously chosen quadruple and prevents us from recovering the bit of the message. In fact the embedding of a bit of the message in one quadruple can generate a new quadruple conforming to the four constraints. In that case, the ambiguity would be done away with at the time of extraction. Thus for a quadruple to comply with the stability constraint, it is necessary that all quadruples formed during the embedding of a bit do not interfere with the chosen quadruples. The quadruples are evaluated with the aid of constraints of coplanarity and convexity, we shall then choose those quadruples as carriers having the best criteria in the zone of disturbance. In that way we are sure that a selected quadruple is stable. Finally we keep only the quadruples satisfying the four presented constraints.

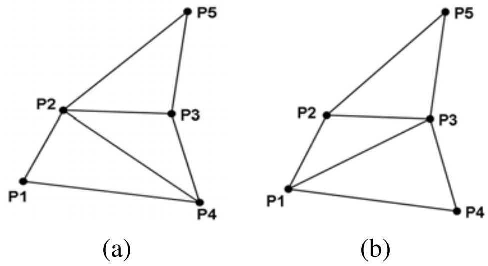


Figure 5: a) Initial triangles, b) Triangles after the embedding.

2.2 Synchronization of quadruples retained

To accomplish synchronization we depend on a key axis. This axis is generated from a key by constructing a 3D point and a vector. This axis is constructed in the coordinate system of the object acquired from a principal component analysis (PCA). PCA enables us to make our method robust to the transformations such as rotations, translations or zoom. To synchronize the message we perform an orthogonal projection of the centers of the selected quadruples retained on this axis. We then classify the projected vertices by taking the initial point of the key axis as origin. The position of these points on the axis will give the sense of embedding of the bits of the message on the quadruples. The Figure 6 illustrates projection and synchronization of five quadruples on the key axis in the coordinate system of PCA of the 3D object.

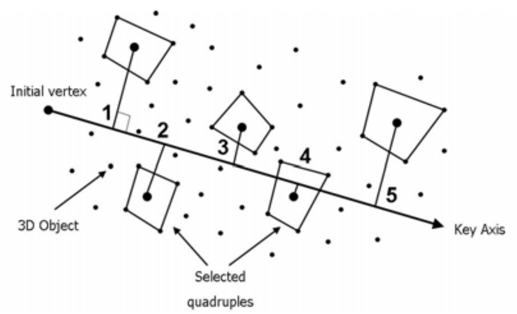


Figure 6: Synchronization of the projections of the quadruples centers on the key axis.

2.3 Data embedding step

As for synchronization, the data embedding step is also based on the key axis. For each retained quadruple we project the four vertices of the quadruple on the key axis. Two of these four vertices belong to the common edge of the two triangles forming the quadruple. Code b is allocated to the projection of vertices belonging to the common edge and code a

is allocated to the projection of the two other vertices of the quadruple. The order of reading of the key axis is going to allow us to acquire a particular order on the projection of vertices a and b. We consider that if a point b is read at first on the key axis then the quadruple contains a bit 1 of the hidden message. Conversely if a point a is read at first then we consider that it is the bit 0 that is hidden. To change the value of hidden bit it is enough to change the common edge of the two triangles forming the quadruple.

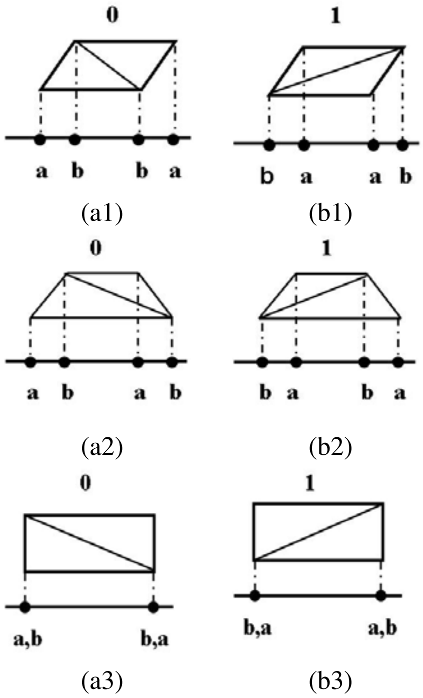


Figure 7: a1), a2), a3) Configuration allowing to embed a bit of the message to 0, b1), b2), b3) Configuration allowing to embed a bit of the message to 1.

The Figure 7 shows all possible solutions to accomplish an insertion as well as the corresponding codes. Figures 7.a show quadruples embedded with 0. If we like to embed these quadruples with a bit 1 it is enough to reverse the common edge of the triangles to get the results presented in Figures 7.b. In particular case where a vertex of the common edge and another vertex of the quadruple are projected in first and in the same place on the key axis we take into account distance between these vertices and the key axis. If the closest vertex is the vertex belonging to the common edge then the quadruple is considered hiding a bit 1, otherwise the quadruple hides a bit 0 as presented in Figures 7.a3 and 7.b3.

3. RESULTS

The Figure 1.a presents the 3D model Bunny of Stanford [9] by a cloud of vertices. To illustrate our results we rely on this model in low definition (453 vertices). The Figure 8 illustrates the triangulated mesh of the same 3D model which includes 947 triangles. In the Figure 9 we can see all the retained quadruples for data hiding. To illustrate the result of an insertion of a bit of the message in one quadruple, we performed a zoom on one part of mesh. The Figure 10.a shows one quadruple before embedding, the Figure 10.b shows the same quadruple embedded with a bit 0. If we consider that the retained quadruple is convex and coplanar as represented in the Figures 10, then the change of topology performed during the embedding of this quadruple will not going to change the surface of the 3D model.

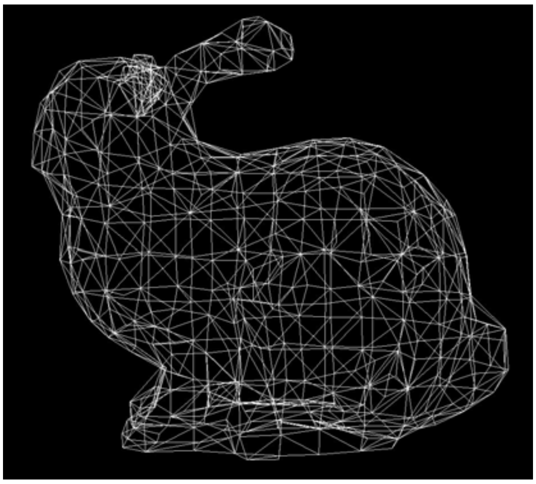
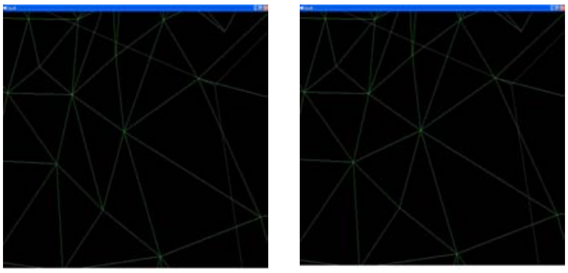


Figure 8: Triangulated mesh (947 triangles).

Figure 9: Retained quadruples with a threshold S_c of 0.005 (46 retained quadruples).

Two criteria are to be taken into account in the analysis of results. The first is the embedding capacity, in bits per vertex, which allows us to evaluate our method (size of the embedded message in bits). In the Figure 11 we can see the embedding capacity of our method according to the number of vertices of the 3D model used. The increase in the threshold allows us to increase the embedding capacity, as shows in the Figure 12, at the cost of the visual quality of the surface. The second criterion of analysis serves for estimating the error made on the surface according to the threshold of coplanarity S_c. The Figure 13 shows the error made on the surface according to the chosen threshold of coplanarity and that for 3D models with various resolutions.



(a) (b)

Figure 10: a) Mesh before the embedding, b) Mesh after the embedding.

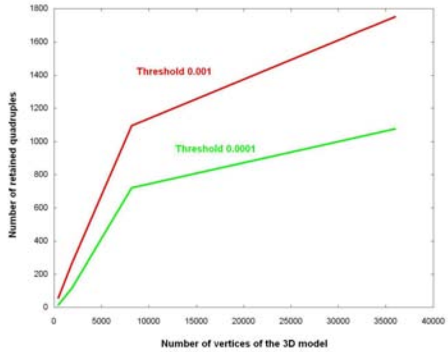


Figure 11: Number of retained quadruples in function of the number of vertices of the 3D model.

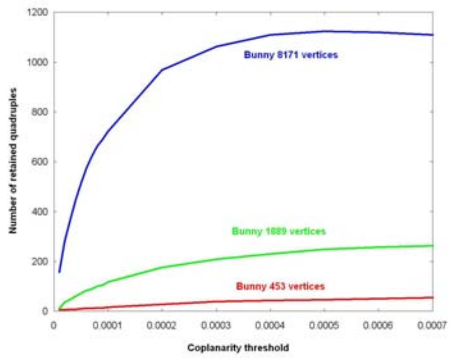


Figure 12: Embedding capacity in function of coplanarity threshold.

4. CONCLUSION

In this paper we presented a new method of data hiding in 3D objects. The main idea of the proposed method is to find and to synchronize particular areas that can be used to embed the message. The data hiding relies on the modification of the topology of edges in chosen zones. These modifications, as a consequence, have to change the structure of triangles constructed in these areas. This method presents the advantage of not altering the position of the vertices of the 3D model. We have applied our method on 3D models with various sizes and various coplanarity thresholds.

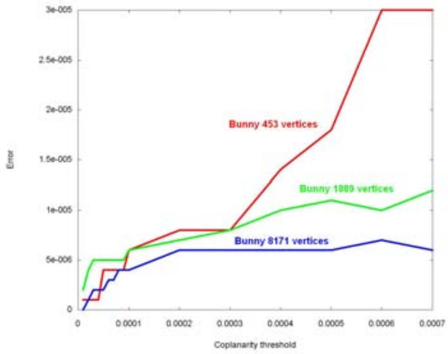


Figure 13: Error in function of coplanarity threshold.

REFERENCES

[1] O. Benedens. Geometry-Based Watermarking of 3D Models. *IEEE Comput. Graph. Appl.*, 19(1):46–55, 1999.

[2] O. Benedens. Affine Invariant Watermarks for 3D Polygonal and NURBS Based Models. In *ISW '00: Proceedings of the Third International Workshop on Information Security*, pages 15–29, London, UK, 2000. Springer-Verlag.

[3] A. G. Bors. Watermarking Mesh-Based Representations of 3-D Objects Using Local Moments. *IEEE Trans. on Image Processing*, 15(3):687–701, mar 2006.

[4] F. Cayre, P. Rondao-Alface, F. Schmitt, B. Macq, and H. Maître. Application of spectral Decomposition to Compression and Watermarking of 3D Triangle Mesh Geometry. *Signal Processing: Image Communication*, 18(4):309–319, April 2003.

[5] S. Kanai, H. Date, and T. Kishinami. Digital Watermarking for 3D Polygons using Multiresolution Wavelet Decomposition. *in Proc. int. Workshop Gemetric Modeling : Fondamentals and Applications*, pages 296–307, Decembre 1998.

[6] B. Macq and F. Cayre. Data Hiding on 3D Triangle Meshes. *IEEE Transactions on Signal Processing*, 51(4):939–949, 2003.

[7] R. Ohbuchi, H. Masuda, and M. Aono. Watermaking Three-Dimensional Polygonal Models. In *MULTIMEDIA '97: Proceedings of the fifth ACM international conference on Multimedia*, pages 261–272, New York, NY, USA, 1997. ACM Press.

[8] R. Ohbuchi, S. Takahashi, T. Miyazawa, and A. Mukaiyama. Watermarking 3D Polygonal Meshes in the Mesh Spectral Domain. In *GRIN'01: No description on Graphics interface 2001*, pages 9–17. Canadian Information Processing Society, 2001.

[9] Stanford University. Stanford 3D Scanning Repository. 2002. http://www-graphics.stanford.edu/data/3Dscanrep.

[10] M. G Wagner. Robust Watermarking of Polygonal Meshes. In *GMP '00: Proceedings of the Geometric Modeling and Processing 2000*, pages 201–208, Washington, DC, USA, 2000. IEEE Computer Society.