



# Dissimulation de Données par Changement de Connexité d'un Maillage 3D

Philippe Amat, William Puech, Sébastien Druon, Jean-Pierre Pedeboy

► **To cite this version:**

Philippe Amat, William Puech, Sébastien Druon, Jean-Pierre Pedeboy. Dissimulation de Données par Changement de Connexité d'un Maillage 3D. CORESA: COmpression et REprésentation des Signaux Audiovisuels, Nov 2007, Montpellier, France. pp.110-114. lirmm-00186897

**HAL Id: lirmm-00186897**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00186897>**

Submitted on 13 Nov 2007

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Dissimulation de données par changement de connexité d'un maillage 3D

P. Amat<sup>1,2</sup>

W. Puech<sup>1</sup>

S. Druon<sup>1</sup>

J-P Pedeboy<sup>2</sup>

<sup>1</sup> Laboratoire LIRMM, UMR CNRS 5506

Université Montpellier II  
161 rue Ada, 34392 Montpellier cedex 5

<sup>2</sup> Stratégies S.A., 41-43 rue de villeneuve

Parc d'affaires SILIC  
BP 80429, 94583 Rungis cedex 5

{philippe.amat, william.puech, sebastien.druon}@lirmm.fr, jp.pedeboy@cadwin.com

## Résumé

*Cet article présente une nouvelle méthode d'insertion de données cachées dans des modèles 3D surfacique. Les méthodes d'insertion de données se basent généralement sur la modification des sommets du modèle pour réaliser l'insertion. Dans notre méthode l'insertion des bits du message s'appuie sur une modification de la connexité du maillage. De ce fait nous ne modifions pas la position des sommets du modèle pour réaliser l'insertion. Notre approche privilégie la capacité d'insertion à la robustesse.*

## Mots clefs

Dissimulation de données, Maillage triangulaire 3D, ACP.

## 1 Introduction

Avec l'évolution des réseaux informatiques, le téléchargement d'objets 3D devient une transmission courante. Afin d'être universellement lisibles ces objets 3D doivent répondre à des normes et des standards internationaux. Il est donc important de respecter les formats classiques permettant de visualiser les objets 3D même si nous souhaitons apporter de nouvelles fonctionnalités à ces objets telles que la protection de ces objets durant la transmission et l'archivage mais également l'enrichissement des objets 3D avec des méta-données sans augmenter la taille des objets et en synchronisant ces méta-données de manière hiérarchique (qualité et spatiale). La représentation la plus utilisée pour les objets 3D est le maillage surfacique. Elle est composée de sommets, d'arêtes et de faces. Cette représentation est très répandue parce qu'elle est supportée par toutes les cartes de rendu graphique. La plupart des méthodes d'insertion de données sont basées sur cette représentation. Pour insérer des données cachées dans un objet 3D les deux domaines les plus couramment utilisés sont le domaine spatial avec des méthodes comme celle de Wagner [8], Benbens [1, 2]

ou Cayre et Macq [4] et le domaine spectral avec des méthodes telles que celle de Ohbuchi *et al.* [7] ou celle de Cayre *et al.* [5]. Notons qu'il existe d'autres domaines plus originaux comme l'espace des ondelettes [6] ou l'espace des NURBS [3]. Nous proposons dans ce travail une nouvelle méthode d'insertion de données cachées qui ne s'appuie pas sur la modification des sommets de l'objet 3D pour réaliser l'insertion. Cette méthode permet d'insérer des méta-données dans un objet 3D tels que le nom de l'auteur, la date de création de l'objet, un logo 2D ou 3D, la couleur de l'objet ou encore la texture de l'objet lui même. De plus, ces méthodes autorisent certaines modifications qui ne sont pas considérées comme des attaques telles que les rotations, les translations et les mises à l'échelle uniforme. Dans la Section 2 nous détaillons la méthode proposée et les résultats et analyses de cette méthode sont présentés en Section 3.

## 2 Méthode d'insertion basée sur la projection de quadrangles

Cette méthode est composée de trois parties. La première partie est la sélection des zones qui vont nous servir à insérer des données cachées. Ces zones sont choisies selon plusieurs contraintes. La seconde partie est la synchronisation du message avec le modèle 3D. Pour cela nous nous appuyons sur un axe construit dans un repère invariant. Ce repère est construit à l'aide d'une analyse en composante principale (ACP) et d'un axe généré à partir d'une clef secrète. Le centre de chaque quadrangle choisi pour l'insertion est alors projeté sur l'axe. La troisième étape est l'insertion du message. Pour cela la connexité des triangles des quadrangles sélectionnés est modifiée. La Figure 1 illustre le schéma général de la méthode.

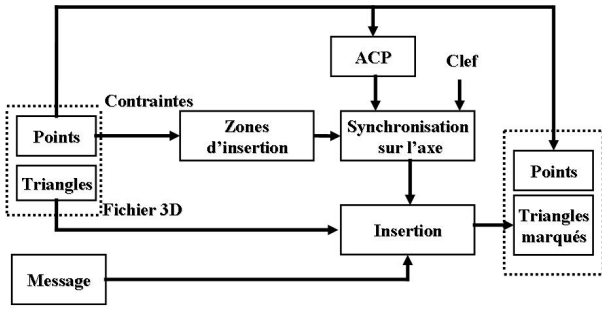


Figure 1 – Méthode d'insertion basée sur la projection de quadrangles.

## 2.1 Sélection des zones d'insertion

La première étape de notre méthode est de définir les zones qui vont nous servir pour l'insertion de données cachées. Ces zones, que nous appelons des quadrangles, sont formées de deux triangles du maillage ayant une arête commune. La recherche des quadrangles se fait en parcourant l'ensemble des triangles du modèle et en regardant pour chaque triangle les quadrangles formés avec les triangles voisins par une arête. L'ordre de parcours des triangles est donné par la liste de points de façon à rendre la méthode indépendante de la position des triangles dans la liste. En effet il est facile d'inverser deux triangles dans la liste sans que cela n'affecte l'objet 3D. Les quadrangles sont validés comme zones d'insertion dans la mesure où ils répondent aux contraintes de coplanarité, convexité, recouvrement et stabilité.

**Contrainte de coplanarité :** la première contrainte est la coplanarité des quadrangles. Le fait de modifier la connexité des deux triangles formant un quadrangle modifie aussi l'angle formé entre ces deux triangles. Par conséquent, l'insertion de données sur un quadrangle non coplanaire affecte la surface et donc le rendu visuel du modèle 3D. De manière idéale il ne faudrait insérer le message que dans les quadrangles strictement coplanaires. Cependant, la quantité de quadrangles répondant strictement au critère de coplanarité est très limitée. Afin d'augmenter la capacité d'insertion, nous insérons dans notre approche un seuil  $S_c$  de tolérance sur la coplanarité des quadrangles sélectionnés. Ce seuil nous permet d'obtenir un compromis entre la capacité d'insertion et la qualité du modèle 3D. Plus ce seuil sera grand, plus la capacité d'insertion sera grande au détriment du rendu visuel. Au contraire plus ce seuil sera petit, plus la capacité d'insertion sera petite mais moins sera dégradé l'objet 3D. Par l'intermédiaire du seuil  $S_c$ , un réglage est donc possible entre la capacité d'insertion et le rendu visuel du modèle 3D. Soit le quadrangle  $Q_{1234}$  formé des points  $P_1, P_2, P_3, P_4$ . Dans un premier temps, la normale  $N$  au plan  $P_{123}$  est calculée. Le point  $P_4$  est alors projeté orthogonalement sur la normale  $N$ . La distance qui sépare la projection du point  $P_4$  sur  $N$  du plan  $P_{123}$  est nommé  $d_4$ . Ce calcul de distance est effectué pour

les projections de  $P_1, P_2$  et  $P_3$  respectivement sur les normales aux plans  $P_{234}, P_{134}$  et  $P_{124}$ . Le quadrangle ne sera retenu que si la plus grande des distances calculées est inférieure au seuil  $S_c$  préalablement fixé :

$$Q_{1234} \text{ est retenu si et seulement si } \min_{i=\{1,2,3,4\}}(d_i) < S_c. \quad (1)$$

Un exemple de calcul de  $d_4$  pour la coplanarité d'un quadrangle est illustré Figure 2.

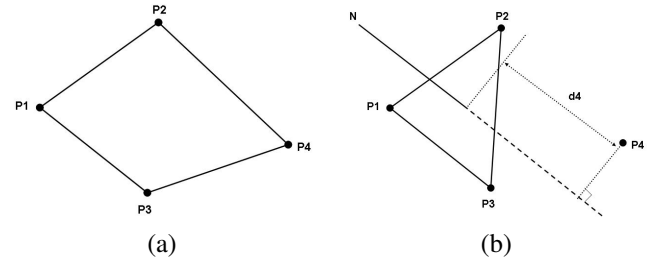


Figure 2 – a) Quadrangle  $Q_{1234}$ , b) Mesure de  $d_4$  pour la contrainte de coplanarité.

**Contrainte de convexité :** la seconde contrainte impose que les quadrangles utilisés pour la dissimulation de données soient convexes. Pour cette contrainte nous partons du principe que les quadrangles retenus répondent déjà à la contrainte de coplanarité. La Figure 3 illustre un cas de quadrangle non convexe formé par les plans  $P_{123}$  et  $P_{234}$ . Nous constatons que la modification de la connexité des triangles lors de l'insertion d'un bit modifie la surface couverte par le précédent quadrangle. Cette modification ajoute une erreur sur la surface du quadrangle. Cette erreur est représentée par la surface  $P_{134}$  de la Figure 3. C'est pour éviter cette erreur sur la surface que la contrainte de convexité a été introduite.

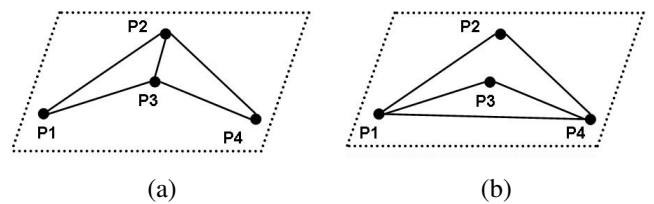


Figure 3 – a) Triangles  $T_{123}$  et  $T_{234}$  avant insertion, b) Triangles  $T_{124}$  et  $T_{134}$  après insertion.

Le calcul de cette contrainte combine successivement deux procédés :

- Les vecteurs  $V_{i,i+1 \bmod 4}$  entre les points  $P_i$  et  $P_{i+1}$  du quadrangle sont calculés pour  $i \in \{1, \dots, 4\}$ . Les angles  $\alpha_i$  entre deux vecteurs consécutifs sont alors calculés.

$$Q_{1234} \text{ est retenu si et seulement si} \quad (2)$$

$$\alpha_i < 180^\circ \text{ pour } i \in \{1, \dots, 4\}.$$

Ceci permet de retirer tous les quadrangles ayant des angles obtus ou trois points alignés.

- Le second calcul nous empêche d’avoir des quadrangles trop proche d’une forme triangulaire, ce qui correspond à trois points sur les quatre alignés. Dans ce cas l’insertion de données ferait apparaître des triangles trop disproportionnés. Un seuil de tolérance  $S_t$  est choisi afin que la proportion des deux triangles d’un quadrangle respecte une certaine valeur. Pour cela nous calculons la longueur des segments  $\overline{P_{23}}$  et  $\overline{P_{14}}$  ainsi que la distance  $D$  séparant les centres des deux segments. Un quadrangle est alors retenu pour l’insertion si et seulement si :

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

**Contrainte de stabilité et de recouvrement :** Le fait de modifier la connexité des triangles pour l’insertion de données cachées engendre de nouveaux triangles et donc de nouveaux quadrangles. Ces nouveaux quadrangles sont susceptibles de répondre aux contraintes demandées pour l’insertion et donc de venir interférer avec les quadrangles choisis au préalable. Il en est de même dans le cas où plusieurs quadrangles retenus contiennent directement des zones de recouvrements. Dans ces deux cas, des ambiguïtés seraient rencontrées au moment de l’extraction. Pour la contrainte de recouvrement les quadrangles sont évalués à l’aide des critères de coplanarité et de convexité, nous choisirons alors pour l’insertion de données le quadrangle ayant les meilleurs critères dans la zone de recouvrement. Pour qu’un quadrangle réponde au critère de stabilité il faut que tous les quadrangles formés lors de l’insertion d’un bit ne puissent pas interférer avec le quadrangle sélectionné pour l’insertion. De cette façon nous nous assurons que le quadrangle est stable. Finalement nous retenons uniquement les quadrangles satisfaisant les quatre contraintes présentées.

## 2.2 Synchronisation des quadrangles retenus

Pour réaliser la synchronisation nous nous appuyons sur un axe. Cet axe est généré à partir d’une clef en construisant un point et un vecteur. Cet axe est construit dans le repère de l’objet obtenu à partir d’une analyse en composante principale (ACP). L’ACP nous permet de rendre notre méthode d’insertion robuste aux transformations affines telles que les rotations, les translations ou les changements d’échelles. Pour synchroniser le message nous effectuons une projection orthogonale des centres des quadrangles retenus sur cet axe. Nous classons alors les points projetés en prenant le point initial de l’axe comme origine. L’ordonnancement de ces points sur l’axe donnera le sens d’insertion des bits du message sur les quadrangles. La Figure 4 illustre la projection et la synchronisation de cinq quadrangles sur l’axe dans le repère de l’ACP de l’objet 3D.

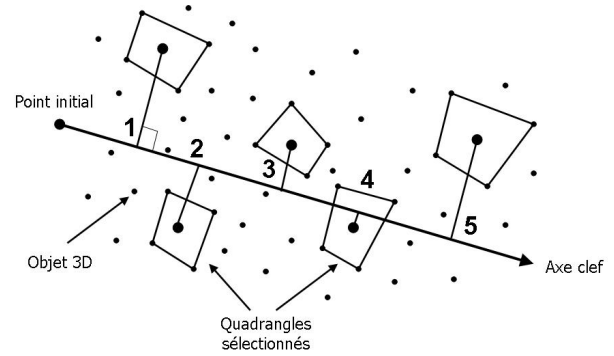


Figure 4 – Exemple de synchronisation de quadrangles.

## 2.3 Insertion du message

Comme pour la synchronisation, l’insertion va également s’appuyer sur l’axe construit à l’aide de la clef. Pour chaque quadrangle retenu nous projetons sur l’axe les quatre points formant ce quadrangle. Deux de ces quatre points appartiennent à la base commune aux deux triangles formant le quadrangle. Le code  $b$  est attribué à la projection des points appartenant à la base commune et le code  $a$  est attribué à la projection des deux autres points du quadrangle. Le sens de lecture de l’axe va nous permettre d’obtenir un ordre particulier sur la projection des points  $a$  et  $b$ . Nous considérons que si un point  $b$  est lu en premier sur l’axe alors le quadrangle contient un bit à 1 du message caché. Respectivement si un point  $a$  est lu en premier alors nous considérons que c’est un bit à 0 qui est caché. Pour changer la valeur du bit caché il suffit de changer la base commune aux deux triangles formant le quadrangle. La Figure

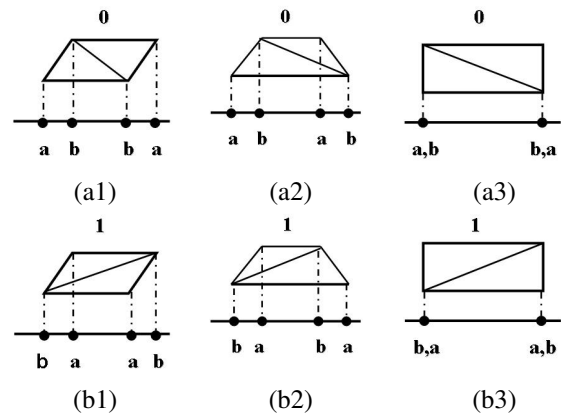


Figure 5 – a1, a2, a3) Configuration permettant d’insérer un bit du message à 0, b1, b2, b3) Configuration permettant d’insérer un bit du message à 1.

5 montre l’ensemble des solutions possible pour réaliser une insertion ainsi que les codes correspondants. Les Figures 5.a montrent des quadrangles marqués à 0. Si nous souhaitons marquer ces quadrangles avec un bit à 1 il suffit

d'inverser la base commune des deux triangles afin d'obtenir les configurations présentées Figures 5.b. Dans le cas particulier où un point de la base et un autre point du quadrangle se projettent en premier et au même endroit sur l'axe nous prenons alors en compte la distance séparant ces points de l'axe. Si le point le plus proche des deux est le point appartenant à la base commune alors le quadrangle est considéré porteur d'un bit à 1, sinon le quadrangle dissimule un bit à 0 comme présenté Figures 5.a3 et 5.b3.

## 2.4 Extraction du message

La première étape pour l'extraction du message est de rechercher les zones d'insertion répondant aux critères de convexité, coplanarité, stabilité et recouvrement. Ces zones sont ensuite synchronisées sur l'axe construit avec la clef et l'ACP réalisé sur l'objet 3D. Une fois les zones d'insertion synchronisées nous projetons les quatre points de chaque quadrangle retenu afin d'en extraire le code. Celui-ci va nous donner la valeur du bit inséré dans le quadrangle. La Figure 6 illustre le schéma d'extraction du message.

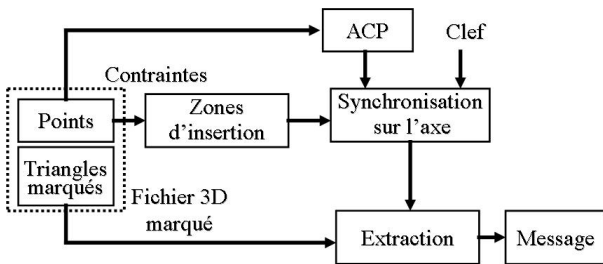


Figure 6 – Méthode d'extraction du message.

## 3 Résultats

La Figure 7.a représente le modèle Bunny de Stanford sous la forme d'un maillage triangulé. Sur la Figure 7.b nous pouvons voir l'ensemble des quadrangles retenus pour l'insertion de données. Afin d'illustrer le résultat d'une insertion d'un bit du message dans un quadrangle, nous avons effectué un zoom sur une partie du maillage. La Figure 7.c montre un quadrangle avant insertion, la Figure 7.d nous montre le même quadrangle marqué avec un bit à 0. Deux critères sont à prendre en compte dans l'analyse des résultats. Le premier est la capacité d'insertion, en bits par points, qui nous permet d'évaluer notre méthode d'insertion (taille du message inséré en bit). Sur la Figure 8.a nous pouvons voir la courbe représentant la capacité d'insertion de notre méthode en fonction du seuil de coplanarité  $S_c$  et cela pour différentes résolutions d'un modèle 3D. Le second critère d'analyse sert à évaluer l'erreur faite sur la surface en fonction du même seuil de coplanarité  $S_c$ . La Figure 8.b montre l'erreur commise sur la surface en fonction du même seuil de coplanarité  $S_c$  et sur les mêmes objets 3D. On constate que, pour un objet à une même résolution, plus le seuil de coplanarité augmente

plus le nombre de quadrangles retenus pour l'insertion de données cachées augmente et plus l'erreur augmente. Par contre pour deux objets à deux résolutions différentes et à un même seuil, la capacité d'insertion augmente alors que l'erreur semble elle rester constante. Ce qui veut dire, qu'à un même seuil de coplanarité, plus l'objet est précis plus on peut lui insérer d'information sans pour autant augmenter l'erreur faite sur la surface. Expérimentalement, en fonction de la valeur du seuil  $S_c$  nous avons réussi à obtenir une capacité d'insertion de  $N/10$  bits, avec  $N$  le nombre de sommets de l'objet 3D. Cette capacité d'insertion est plus faible que pour d'autres méthodes ou par exemple  $N - 2$  bits sont insérés. Ceci est dû à une contrainte forte sur la position des sommets, puisque dans notre méthode la position des sommets de l'objet reste inchangée. De manière théorique, sans aucune contrainte la capacité d'insertion maximale est de  $N/4$  bits puisque nous insérons 1 bit par quadrangle.

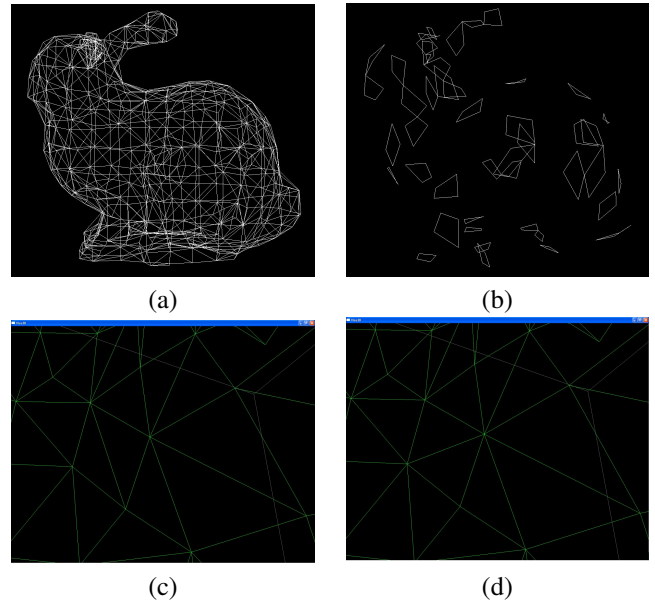
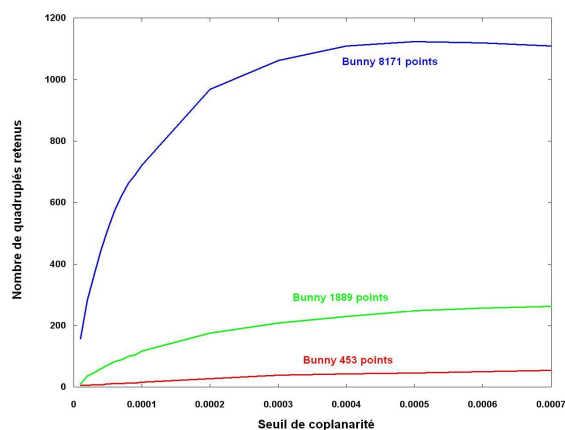


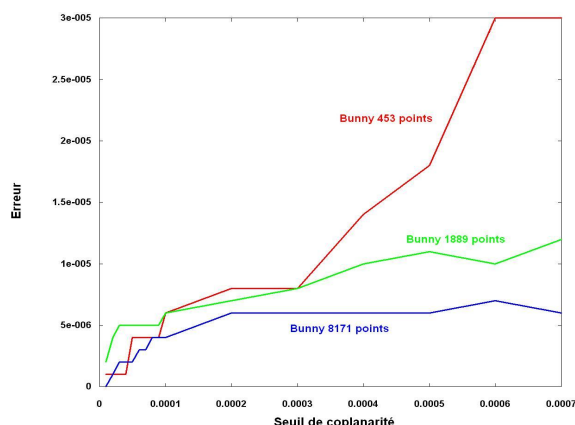
Figure 7 – a) Maillage triangulé (947 triangles), b) Quadrangles retenus avec la méthode par projection avec un seuil  $S_c$  de 0.005 (46 quadrangles retenus), c) Maillage avant insertion, d) Maillage après insertion.

## 4 Conclusion

Cet article présente une nouvelle méthode d'insertion de données cachées dans des objets 3D. Le système est basé sur la modification de la connectivité du maillage dans des zones particulières. Ces zones appelées quadrangles doivent répondre à certaines contraintes pour assurer l'invisibilité et la stabilité de l'insertion. Une fois les quadrangles sélectionnés ils sont projetés sur un axe obtenu à l'aide d'une clef secrète. L'ordonnement de ces projections sur l'axe donne le sens d'insertion du message.



(a)



(b)

Figure 8 – a) Nombre de quadrangle retenus en fonction du seuil de coplanarité, b) Erreur mesurée en fonction du seuil de coplanarité.

Le repère du système est obtenu à partir d'une ACP de façon à rendre l'insertion robuste aux transformations affines. Avec la méthode proposée nous avons montré qu'il était possible d'insérer une quantité de données pouvant atteindre une taille représentant 10% du nombre de sommets de l'objet 3D en fonction du seuil de coplanarité choisi. Dans nos perspectives nous envisageons de tester la robustesse de l'insertion face au bruit. En effet, l'insertion de bruit sur les sommets peut perturber la récupération des données mais aussi le repère créé par l'ACP. Dans le cas d'une trop grande perturbation d'autres méthodes peuvent être envisagées comme le calcul de moments surfaciques ou volumiques. Nous étudions aussi une nouvelle approche pour la synchronisation des données en utilisant un graphe de parcours des sommets. Cette approche nous permettrait d'augmenter la stabilité des quadrangles utilisés pour l'insertion.

## Références

- [1] O. Benedens. Geometry-Based Watermarking of 3D Models. *IEEE Comput. Graph. Appl.*, 19(1) :46–55, 1999.
- [2] O. Benedens. Two High Capacity Methods for Embedding Public Watermarks into 3d Polygonal Models. In *Proceedings of the Multimedia and Security-Workshop at ACM Multimedia*, pages 95–99, 1999.
- [3] 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.
- [4] F. Cayre and B. Macq. Data Hiding on 3D Triangle Meshes. *IEEE Transactions on Signal Processing*, 51(4) :939–949, 2003.
- [5] 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.
- [6] S. Kanai, H. Date, and T. Kishinami. Digital Watermarking for 3D Polygons using Multiresolution Wavelet Decomposition. in *Proc. int. Workshop Geometric Modeling : Fundamentals and Applications*, pages 296–307, Decembre 1998.
- [7] 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.
- [8] 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.