International Journal of Virtual Reality
Volume 19 Issue 2

IPI Press
ISSN: 1081-1451

Received: 04/2019

Published: 10/2019

3D Objects based Security in Virtual and Augmented Reality Systems

H. Djaghloul [1] and J-P. Jessel [2]

[1] University of Ferhat Abbes - Setif 1 (UFAS), Algeria
[2] Institut de Recherche en Informatique de Toulouse (IRIT), France

Abstract - **This paper presents a new method for public and blind watermarking of 3D objects and hiding information of various types in virtual reality and augmented reality environments. In particular, the the virtual scene is transformed to a tree-based model based using constructive solid geometry (CSG). The tree of the virtual 3D objects is composed of nodes and leafs. Each tree node contains a set operator (union, intersection or difference) whereas the tree leafs consist of geometric primitives such as spheres, cylinders and cubes. The proposed method allows embedding invisible watermarks that can hide different security information types. The proposed method ensures unlimited capacity and absolute invisibility of the hidden information with robustness to affine transformations. Thanks to these unique properties, it is possible to implement all the security infrastructures needed for a virtual and augmented reality system. Finally, the performances of the proposed method are shown using 3D models of different types and scenes.**

Index Terms - **public blind 3D objects watermarking; virtual reality; augmented reality; information hiding; CSG.**

I. INTRODUCTION

Scenes or 3D objects modeled using constructive solid geometry (CSG) are represented by trees where leaves are primitives such as spheres, cubes or cylinders, ... and nodes are operators (union, intersection and difference). Protecting and authenticating a 3D object without altering its structure or adding some kind of header files can be performed thanks to watermarking. Watermarking is the process of adding an invisible information that can be used as a proof of copyright protection which must be robust to a given set of attacks such as affine transformations.

Although the research in information hiding and digital watermarking is relatively young, it has been extensively studied and widely used for many data types including text, image, audio. In this study, we are interested in watermarking 3D objects and virtual scenes, particularly, with CSG modeling technique. Despite that the majority of researches in the literature have focused on digital images watermarking, one can find several research groups that have worked on 3D watermarking. To our knowledge, among the first researches in watermarking 3D models, one can distinguish the pioneering work of (Ohbuchi, Masuda, and Aono 1998)(Ohbuchi et al. 2001)(Praun, Hoppe, and Finkelstein 1999)(Benedens 1999) and (Yeo and Yeung 1999).(Ohbuchi, Masuda, and Aono 1998)(Ohbuchi et al. 2001) propose to watermark 3D polygonal models through geometric and topological modifications. This method does not the original model and it concerns the copyright protection of VRML models. The reported work incorporate certain robustness criteria against some transformations like local deformations but it provides little security to combat attempts to removal according to (Yeo and Yeung 1999). In their work, (Praun, Hoppe, and Finkelstein 1999) propose to watermark 3D objects composed of arbitrary triangle meshes by extending the spread spectrum approach applied to digital images by perturbing the coefficients of most significant DCT basis functions. The watermarking method proposed by (Yeo and Yeung 1999) enables the verification of polygonal models for detecting unauthorized alterations. (Bors 2006) suggest a blind watermarking algorithm for 3D models and objects by embedding a string of bits generated using a public key in the geometrical structure of the graphical object by changing the locations of certain vertices. In (Zafeiriou, Tefas, and Pitas 2005), the authors propose two methods suitable for blind 3D mesh object watermarking applications by applying restrictions in the spherical representation. One can also find other techniques primarly inspired from those applied to digital 2D images (Chou and Tseng 2006)(Fornaro and Sanna 2000)(Harte and Bors 2002)(Syed 2011)(Uccheddu, Corsini, and Barni 2004). However, they all addresses the embedding of 1D or 2D images in the original 3d model. A method used for hiding a 3d object inside another 3d object has been proposed by (Kishk and Javidi 2003). In their work, the authors proopsed to hide the 3d watermark in digital holograph of the 3d host object.

More recently, other research groups have proposed more enhanced solutions for 3D objects watermarking(Bader and Amara 2014)(Molaei, Ebrahimnezhad, and Sedaaghi 2013)(Mun et al. 2015)(Soliman, Hassanien, and Onsi 2016)(Su et al. 2013)(Wang et al. 2014a)(Wang et al. 2014b)(Yang et al. 2014)(Yu, Au, and Chiu 2016). A survey of 3D triangular mesh watermarking techniques can be found in (Medimegh, Belaid, and Werghi 2015).

However, to our knowledge and outside of our work, the

E-mail: djaghloul@univ-setif.dz, jessel@irit.fr

literature on CSG objects watermarking is absolutely very limited (Fornaro and Sanna 2000). This can be explained by the difficulty of the task due to the fact that there are very little parameters governing primitives shape and behavior comparatively to other modeling techniques such as with meshes or voxels. For instance, the geometry of a sphere is solidly defined using only by its center and ray, resulting in a dataset of four scalars only which is very few relatively to the millions of vertices and faces used in the case of mesh-based modeling.

The method proposed by (Fornaro and Sanna 2000) is a method for public key watermarking of CSG models but it is restricted to authentication purposes. In other words, it allows the detection and verification of any model modification or deterioration. This is accomplished primarily by adding fictive attributes to those of the primitives containing some computed information based on primitive properties or by inserting a duplicate of a given node by using the union operator. Because of the public key scheme of this method, it does not need some external secret keys known by both the author and the users of the model. However, it does not allow the insertion of additional information as for copyright protection for instance which needs embding of totally different model or dataset with respect to the original model. These lead some authors to consider it further as a labeling method more than a watermarking one. Another major con of this method is that is proprietary format dependent as it requires the insertion of hidden information in an augmented properties set of the primitives. Moreover, no description has been given about the robustness of the method against some kind of attacks such as geometric and topological attacks.

Here, We propose a method for information hiding and high capacity watermarking of 3D scenes modeled using constructive solid geometry (CSG) which allows absolute invisibility of the marks and robustness to affine transformation attacks. In particular, we propose to hide information inside the CSG tree by applying topological transformations and geometry based scene tree augmentation. Here, we mean by topology of the CSG tree, the set of operators used to link the primitives. On the other hand, the geometry of the CSG tree concerns the types and properties of the primitives.

In the following section, we provide the mathematical formula and basic background necessary to explain the proposed watermarking method. Then, in section 3, we show some experiments that demonstrate the effectiveness of the proposed method. Finally, we present our conclusions and perspective in the last section.

II. Proposal

Let consider an object or a scene that is implemented as a CSG tree. In the case of a simple object, one can find different set/logical operators linking between different primitives such as spheres, cylinders and so on. In the case of a whole scene, one can represent in addition to different 3D objects the other elements of the scene such as light sources, materials, cameras or view points.

By watermarking a node, we mean the insertion of the mark at the link between this node and its unique predecessor using another topological operator $[+]$. The operator can be one of any of the basic set/algebraic operations used to define a CSG tree, such as, the union, the intersection and the difference. Note that the mark may be another CSG model of a completely different 3D object. Hereafter, we explain the three cases of the watermarks insertion scheme for each set/logical operator, namely, the difference, the union and the intersection operation. We denote by N the subtree to be watermarked. N can be the root of the whole CSG tree of the original 3D model. The basic idea of the watermarks insertion is that the result of the set operations application will always produce the original subtree whatever the set operator used. Thus, if we consider ψ as the subtree to be watermarked, ω the subtree of the watermark and $[+]$ the set operator used to link between the original model and watermark model subtrees, then we will always have

$$\psi[+]\omega = \psi$$

. This fundamental equation guaranties the absolute invisibility of the watermark in all cases whatever the rendering algorithm used. Here, we apply the fundamental equation considering the three cases namely, the difference, the union and the intersection set operators used for building the watermarked CSG tree.

Here, we define a 3D object modeled by a CSG tree ψ as a set of points in 3D space that can be expressed as follows,

$$\psi = p = (x, y, z)|(x, y, z) \in \Re^3 \wedge (x, y, z) \in \psi$$

. In this case, the union $(\cup/+)$ allows to hide the information in an empty CSG deduced from a set of objects with empty intersection or any object included in N so that the result of the rendering is always N itself) or else the difference between an object and its copy that occupies the same volume.

In order to allow most robustness to both geometric attacks such as affine transformations and intentional targeted topological attacks, we propose to insert the mark into the CSG tree using various insertion scenari. First, the mark can be inserted into the empty volume of the empty bounding box of the primitive or the scene. The second approach consist in putting elements of the mark support within the empty space created by the application of construction operators such as the difference, the union, the intersection and the xor operators. To render the task of the attacker difficult, one can duplicate the nodes used to model the object or the scene. In this case, the mark is inserted in the CSG node of the object but with positioning the mark support elements into other various parts within the scene. Then, as for crypting algorithms, the task of the detection and the elimination of the mark by the attacker will be so difficult and, thus, the temporal cost so high. Another proposition consists in the insertion of the mark support into well known positions with respect to the structure of the CSG tree. for example, to put the mark nodes at a given position of the CSG tree of the scene with respect to an order of (walking).

Possible strategy to render the topological attack of changing the operator without interest is to consider prim-

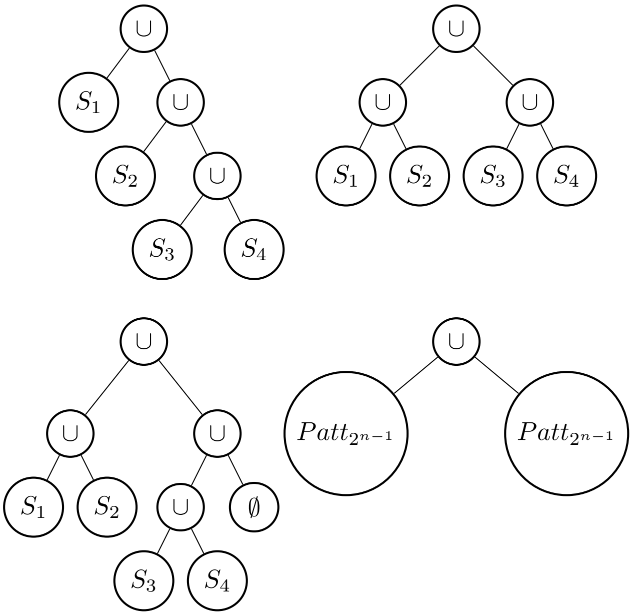


Figure 1: Watermark containers topologies

itives with exotic geometrical properties in the case of constructive difference for example.

Linking with minus operator the scene and the mark means linking the *root* of the scene and the root of the mark.

In order to determine a root of the scene, we suppose that it is a *super CSG object* with a multitude of objects linked by union operators.

We propose also to insert the mark inside a primitive using the union operator and the same photometric parameters as the primitive if it is plain.

Hidding information container

The mark may be an object of any type including another CSG object. We can also use a set of 2^n fictive spheres with arbitrary number as long as length of the hidden binary sequence has a length equal to the number of considered distances between the marking spheres. The set of spheres used to contain the included information can follow different patterns.

If the scene is modeled using a set of 2^n fictive spheres which we define randomly in a fictive bounding box B_w with same dimension as B_s. Watermarking spheres in B_w are inserted in the empty volumes of B_s. Note that the spheres are sufficiently small to be contained in the founded empty bounding boxes.

According to the fact that the mark is a collection of 2^n spheres, we directly insert the mark into the distances between the primitives composing the added CSG tree.

These distances are perturbed to hide the mark by using the binary string coding any text or image. For all mark primitives, we compute the distances between them according to a given point in the space such as the gravity center of the bounding spheres set. Inside each of this bounding volumes, we can put either a primitive, an object of the scene or even the entire scene with a correspondent scaling factor.

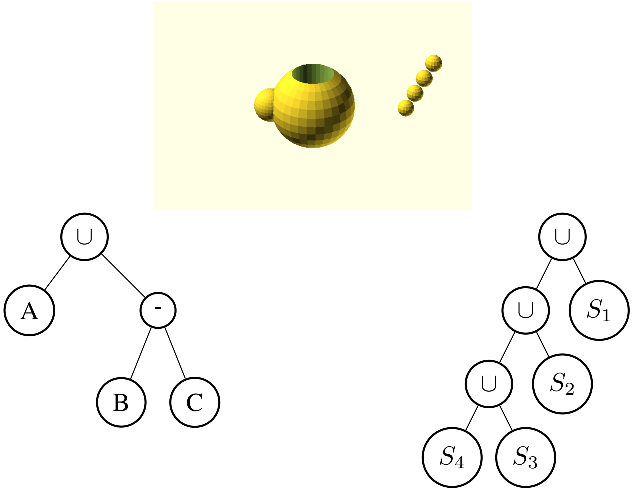


Figure 2: Example of CSG watermarks

Robustness to attacks

We limit our tests to affine transformations attacks. The distances between the spheres are not affected. Among topological attacks, one can consider the permutation of the branches of a given node if the operator is commutative such as intersection, union and xor. It is also possible to have the same object or scene with changing the hierarchy of the CSG tree using the same operator such as the union, intersection and the xor operators.

Hidden information extraction

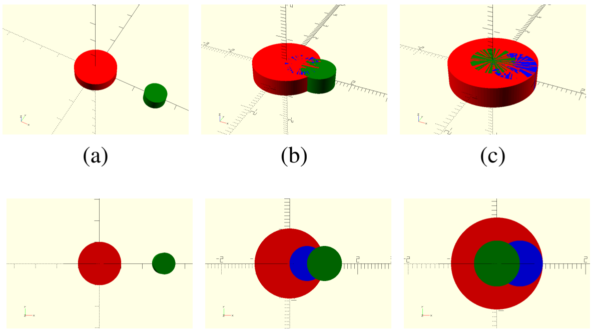
In order to be able to distinguish between the branches of the mark and those used in the construction of the CSG scene, whatever, the number of primitives and operators used, is to ensure that the mark is always invisible and in the same time the other construction are visible if there is an attack.

For example, one has to consider the result of the topological expression is always an empty set even in the case of internal sub trees using the scheme presented above $A+B = \emptyset$.

Experimental results

To assess the performance of the 3D based information hiding procedure, we first conduct an experiment on the original synthetic model using the watermarking method defined in section 2. Our synthetic model consists of a cylindrical primitive (ϕ) consisting of a 3D disk having a unit radius ($\phi_r = 1$) and a height equals to a half of its radius ($\phi_h = 0.5$) and with no applied texture as shown in Fig. (3a). On the other hand, the watermarking 3D model (ω) is a cylindrical primitive similar to the original model (ϕ) but with a radius equals to its half ($\omega_r = 0.5$). As with the original 3D primitive, no texture is applied to the watermark 3D model (ω) but has the same color as well as the same material photometric properties of the original model (ϕ) to be watermarked.

Initially, the distance between the watermark and original 3D model centers is three times the radius of the original model ($\|\phi_c\omega_c\| = 3 * \phi_r$) as shown in Fig. (3a). Next, the watermarking 3D model is located with the center belonging to the cylindrical boundary of the original 3D disk model, as shown in Fig. (3b). Here, we consider a volume subset (ρ) of the original model as a disk with a center belonging to the line segment $[\phi_c\omega_c]$ linking the centers of the original and watermark models ($\rho_r = \|\rho_c\phi_c\|$), the radius of the inner volumetric subset disk (ρ) is given by the half of the original model radius ($\rho_r = \phi_r/2$). In the third case (Fig. (3c)), the 3D model of the watermark (ω) is located so that it is completely included in the inner volume set of the 3D model of the original object (ϕ) to be watermarked. Figs (3d), (3e) and (3f) show the top views of the original model (ϕ), the subset volume of the original model (ρ) and the watermark model (ω) in the different cases of geometric relationship between the original model and the watermark model (Out, Intersect, In), respectively. Note that in this experiment, the scene is composed directly using the primitives without using ony topological set operator to link the primitives (ϕ, ρ, ω). We can observe that the watermark 3d model is only invisible in the case where it is totally included in the volume set of the original model.

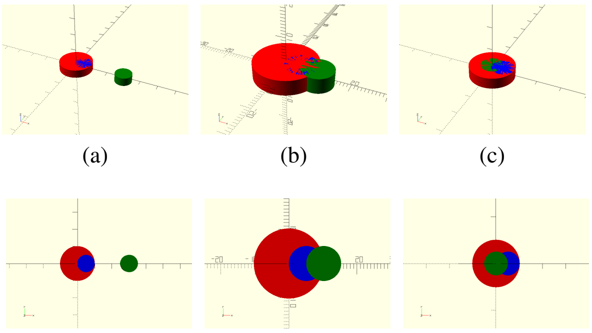


(a) (b) (c)

(d) Out (e) Intersect (f) In

Figure 3: Rendering primitives without CSG tree

The next experiment shows the different primitives (ϕ, ρ and ω) placed inside a CSG tree and linked with the union (+) topological set operations so that all the primitives will be visible only if they are intersecting. Figs. (4a)-(4c) show the original model and the watermarking model in different positions, namely, separated (Out), merged (Intersecting) and included (In). For each case, the figures of the second line (Figs. (4d)-(4f)) show top X-Y views of the 3D rendered images of the corresponding CSG model. Here, we note that topology is preserved but only geometric properties are affected (the disc center of the watermarking model ω).As can be seen, the rendered images are identical to those of the precedent section without solid geometry construction using topological set operators. We can also observe that the only case in this construction (CSG union) where the watermark 3d model (ω) is completely invisible is the case where it is totally included within the volume set of the original 3d primitive (ϕ) (see Figs. (3f) and (4f)).

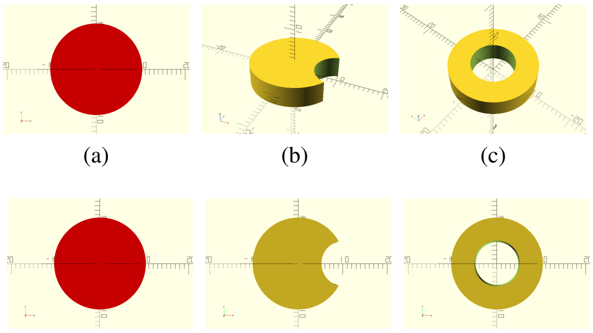


(a) (b) (c)

(d) Out (e) Intersect (f) In

Figure 4: Rendering all primitives within a Union-CSG tree (ϕ+ω)

For comparison, we have considered the same scene geometric primitives (ϕ, ρ and ω) taking place at the same positions but with applying the difference (-) topological set operator between the original 3D model (ϕ) and the 3D model of the watermark (ω). In



(a) (b) (c)

(d) Out (e) Intersect (f) In

Figure 5: Rendering all primitives within a Difference-CSG tree (ϕ-ω)

In the following tests, we consider a virtual of the Alice tea table scene (Fig. 6) with a variety of objects and watermarked using a set of spheres. These spheres may have different photometric properties such as ambient, specular, reflective and transparent coefficients. Also, some of them can be outside or inside an object.

In this figure , we use union operator to render the scene, all the marks and shadows are totally visible. Here, we use the operator to hide the container model inside the bounding box of the whole virtual scene. Thus, we can see that the sphere that has intersection with the table produces a hole, while the cube which has no intersection with the table produces a void.

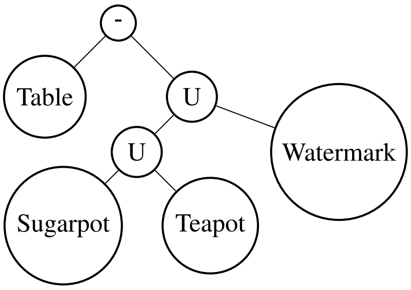
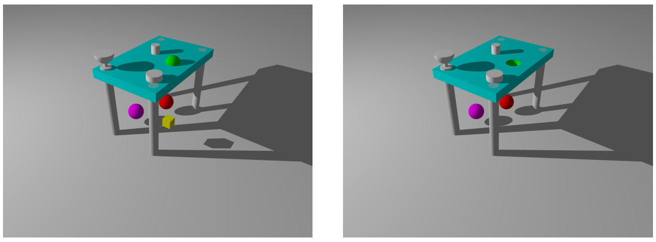


Figure 6: Example of CSG watermarks



(a) before watermarking (b) After watermarking

Figure 7: scene table with different insertion scenari

III. Conclusion

In this paper, we proposed a new method for watermarking 3D scenes that can contain any kind of 3D objects including CSG models, Octrees and so on. The proposed method allows high level of robustness to affine transformation attacks with unlimited watermarking capacity using different types of marks. Thanks, to the insertion of of different types of data in the distances between the primitives of the watermark container, the method provide robustness to different types of attacks such as affine transformations. However, despite of the unique properties of the proposed hiding method method, it shows some weaknesses to some attacks such as topological attacks. So, we propose as perspective works to study and enhance the robustness of the proposed method to address these limitations.

References

[1] Bader, S., and Amara, N. E. B. 2014. A securing access approach to virtual worlds based on 3d mesh watermarking of avatar's face. In *2014 4th International Conference on Image Processing Theory, Tools and Applications (IPTA)*, 1–6. IEEE.

[2] Benedens, O. 1999. Geometry-based watermarking of 3d models. *IEEE computer Graphics and Applications* 19(1):46–55.

[3] Bors, A. G. 2006. Watermarking mesh-based representations of 3-d objects using local moments. *Image Processing, IEEE Transactions on* 15(3):687–701.

[4] Chou, C.-M., and Tseng, D.-C. 2006. A public fragile watermarking scheme for 3d model authentication. *Computer-Aided Design* 38(11):1154–1165.

[5] Fornaro, C., and Sanna, A. 2000. Public key watermarking for authentication of csg models. *Computer-Aided Design* 32(12):727–735.

[6] Harte, T., and Bors, A. G. 2002. Watermarking 3d models. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 3, 661–664. IEEE.

[7] Kishk, S., and Javidi, B. 2003. 3d object watermarking by a 3d hidden object. *Optics express* 11(8):874–888.

[8] Medimegh, N.; Belaid, S.; and Werghi, N. 2015. A survey of the 3d triangular mesh watermarking techniques. *Int J Multimed* 1(1).

[9] Molaei, A. M.; Ebrahimnezhad, H.; and Sedaaghi, M. H. 2013. A blind fragile watermarking method for 3d models based on geometric properties of triangles. *3D Research* 4(4):1–9.

[10] Mun, S.-M.; Jang, H.-U.; Kim, D.-G.; Choi, S.; and Lee, H.-K. 2015. A robust 3d mesh watermarking scheme against cropping. In *3D Imaging (IC3D), 2015 International Conference on*, 1–6. IEEE.

[11] Ohbuchi, R.; Takahashi, S.; Miyazawa, T.; and Mukaiyama, A. 2001. Watermarking 3d polygonal meshes in the mesh spectral domain. In *Graphics interface*, volume 2001, 9–17. Citeseer.

[12] Ohbuchi, R.; Masuda, H.; and Aono, M. 1998. Watermarking three-dimensional polygonal models through geometric and topological modifications. *Selected Areas in Communications, IEEE Journal on* 16(4):551–560.

[13] Praun, E.; Hoppe, H.; and Finkelstein, A. 1999. Robust mesh watermarking. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 49–56. ACM Press/Addison-Wesley Publishing Co.

[14] Soliman, M. M.; Hassanien, A. E.; and Onsi, H. M. 2016. Watermarking 3d triangular mesh models using intelligent vertex selection. In *Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015*, 617–627. Springer.

[15] Su, Z.; Li, W.; Kong, J.; Dai, Y.; and Tang, W. 2013. Watermarking 3d capd models for topology verification. *Computer-Aided Design* 45(7):1042–1052.

[16] Syed, A. A. 2011. Digital watermarking.

[17] Uccheddu, F.; Corsini, M.; and Barni, M. 2004. Wavelet-based blind watermarking of 3d models. In *Proceedings of the 2004 workshop on Multimedia and security*, 143–154. ACM.

[18] Wang, J. T.; Fan, C. M.; Huang, C. C.; and Li, C. C. 2014a. Error detecting code based fragile watermarking scheme for 3d models. In *Computer, Consumer and Control (IS3C), 2014 International Symposium on*, 1099–1102. IEEE.

[19] Wang, J. T.; Yang, W. H.; Wang, P. C.; and Chang, Y. T. 2014b. A novel chaos sequence based 3d fragile watermarking scheme. In *Computer, Consumer and Control (IS3C), 2014 International Symposium on*, 745–748. IEEE.

[20] Yang, Y.; Pintus, R.; Rushmeier, H.; and Ivrissimtzis, I. 2014. A steganalytic algorithm for 3d polygonal meshes. In *2014 IEEE International Conference on Image Processing (ICIP)*, 4782–4786. IEEE.

[21] Yeo, B.-L., and Yeung, M. M. 1999. Watermarking 3d objects for verification. *Computer Graphics and Applications, IEEE* 19(1):36–45.

[22] Yu, K.; Au, K.; and Chiu, W. 2016. Watermarking scheme for geometric data protection and detection on 3d cad assembly model. *Computer-Aided Design and Applications* 1–10.

[23] Zafeiriou, S.; Tefas, A.; and Pitas, I. 2005. Blind robust watermarking schemes for copyright protection of 3d mesh objects. *Visualization and Computer Graphics, IEEE Transactions on* 11(5):596–607.