From Visual Confidentiality To Transparent Format-Compliant Selective Encryption Of 3D Objects

Sebastien Beugnon, William Puech, Jean-Pierre Pedeboy

FROM VISUAL CONFIDENTIALITY TO TRANSPARENT FORMAT-COMPLIANT SELECTIVE ENCRYPTION OF 3D OBJECTS

Sébastien BEUGNON[1,2]*, William PUECH*[1] *and Jean-Pierre PEDEBOY*[2]

[1] LIRMM, Univ Montpellier, CNRS, Montpellier, France
[2] STRATEGIES, Rungis, France

ABSTRACT

Unlike full encryption methods, for which the encryption does not preserve the internal structure of files, this paper presents an efficient format compliant selective encryption method for 3D object binary formats. The method encrypts selected bits of the 3D object geometry to visually protect the content without increasing the file size. Geometrical distortions are created in order to partially or fully protect the content, but they do not corrupt 3D scenes for preview purposes. As a function of the degradation level, we can obtain security from visual confidentiality to transparent encryption passing through sufficient encryption providing solutions for different application scenarios. Experimental results and analysis show the efficiency of the proposed method.

Index Terms— Visual Cryptography, Selective Encryption, 3D Object, Content Protection.

1. INTRODUCTION

In recent years, multimedia content is overflowing the Internet. 3D objects are used in a large number of applications including; medical visualization, simulation tools, games, online sales, animation and cinema rendering. Moreover, today fab labs and self-manufacturing with 3D printers is becoming a very popular consumer trend, which could evolve into a whole new economy. With the democratization of this technology, 3D object downloading platforms are developing and proprietary format usage becomes a huge obstacle for their development. Digital rights management (DRM) becomes more and more crucial for content producers. 3D objects represent financial assets and should be protected from piracy and counterfeiting like any critical data. 3D protection techniques can be carried out by encryption or watermarking.

During the last few years, work has been mainly concentrated on 3D watermarking [1, 2, 3]. Watermarking enhances 3D objects with visible or invisible information to bring new features like tamper detection, high capacity data hiding and traceability to identify the source of the object.

Meanwhile, protection against unauthorized access has recently received some attention [4, 5, 6, 7]. Full encryption does not take the content of encrypted data into consideration. Media contents are processed as binary files, destroying the internal structure of files during the encryption process. Koller *et al.* proposed to protect data by using a remote rendering system [4]. Cho *et al.* chose to use watermarking with additional random encryption of connectivity of 3D objects [5]. Gschwandtner and Uhl used progressive mesh representation with layers of refinement [6]. Usually employed to transmit 3D objects over a network with a low quality preview as a sufficient encryption method, they chose to encrypt the content of these layers with different strategies. Later, Éluard *et al.* presented multiple geometry-preserving encryption methods (GPE) based on the permutation of vertices or coordinates to protect the content [7]. Their methods preserve properties like the bounding box or the convex hull in order to minimize impact on rendering time.

In this paper, a new encryption method for 3D objects is proposed. The approach is based on selective encryption of the binary representation of coordinates. Thanks to this structural encryption based on the standardized representation of floating values, the method is able to protect 3D objects by completely protecting content from visual confidentiality to transparent encryption through sufficient encryption. Consequently the method allows us to choose the encryption level applied to the 3D object.

The rest of this paper is organized as follows. In Section 2, the method is described with necessary specifications. Then, experimental results are evaluated in Section 3 from a statistical point of view with metrics tailored for 3D objects and a security analysis based on previous work. Finally, we draw some conclusions in Section 4.

2. THE PROPOSED SELECTIVE ENCRYPTION METHOD

In this section, we develop the proposed method to selectively encrypt 3D objects with a control over geometrical impacts of the encryption, by encrypting only a selection of the binary representation of coordinates of vertices in the 3D object. As illustrated in Fig. 1, our encryption method requires 2 parameters along the 3D object, which are a secret key K and a degradation level D. In Section 2.1, we present a brief summary of floating value binary representation used in 3D object file formats. Then, the proposed method is divided into two main tasks, which are selection of the data to encrypt

978-1-5386-1737-3/18/$31.00

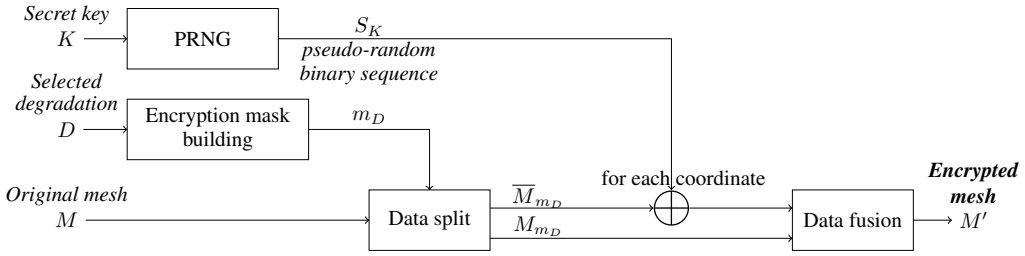


Fig. 1: Overview of our 3D object selective encryption method.

described in Section 2.2 and encryption of the vertex coordinates of the 3D object in Section 2.3. Finally, Section 2.4 presents the decryption process using a 3D object protected by our method.

2.1. Floating value representation

Coordinates of a vertex in a 3D object are defined by floating values, which are normalized in binary 3D object files by the *IEEE 754* Norm [8]. This norm is the most common representation of floating values on current machines. Based on 32 bits, this representation holds three types of distinct data: the sign, the exponent and the mantissa.

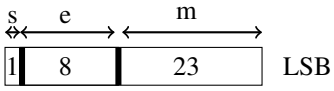


Fig. 2: Representation of a floating value by the *IEEE 754* Norm [8].

As illustrated in Fig. 2, each of the three types of information constituting a floating value has a specific quantity of bits: the Most Significant Bit (or MSB) indicates if the value is positive (0) or negative (1). The following 8 bits represent the exponent. Then, the next 23 bits correspond to the mantissa. The exponent and the mantissa allow us to represent any absolute floating values between $1.175494e^{-38}$ and $3.402823e^{+38}$ with a precision of 6 to 7 decimal places.

2.2. Encrypted data selection

This step consists of two specific tasks. First of all, the method generates a pseudo-random binary sequence S_k with a pseudo-random number generator (PRNG) and a secret key K. The sequence is used for the encryption of coordinates.

Then, our method builds an encryption mask based on the desired level of degradation D. This variable defines the strength of the selective encryption. The lower the level of degradation, the more recognizable the 3D object. The mask determines which parts to use of the floating value binary representation and how many bits are encrypted.

We have developed a strategy to generate a mask m_D, which consists in dragging a sliding window over the bits of the floating value to select which ones to encrypt. We call this strategy, D-Sliding Window mask (or D-SW mask) where the level of degradation D is defined by the pair:

$$D =< p, l >, \tag{1}$$

where the parameter p of D indicates the position of the first bit (between $\{1, ..., 31\}$) of the encryption mask, meanwhile l (between $\{1, ..., p\}$) defines the number of bits to encrypt. As a function of the encryption level parameter, we define three different types of encryption level:

- **Visual confidentiality**: the shape and the content of the 3D object are visually protected. Information such as format data, may be leaked, but an adversary is not able to compute any visual information;
- **Sufficient encryption**: the shape of the 3D object is recognizable, but the content is sufficiently protected visually;
- **Transparent encryption**: shape and content are recognizable, however high quality is protected. An adversary may only recover a low quality version of the 3D object.

As illustrated in Fig. 3, the sliding window allows the user to select the bits to encrypt more precisely. The higher the value of p, the greater the geometrical distortions.

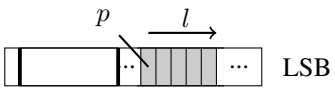


Fig. 3: Representation of selected bits to encrypt as a function of the selected degradation level D.

In a specific case, by setting the constraint $l = p$ it is possible to create a mask selecting all the least significant bits (LSB) for encryption, we call it D-LSB mask or $D =< p, p >$.

2.3. Selective encryption of vertex coordinates

During the 3D object encryption step, as shown in Fig 1, for each coordinate of each vertex of the 3D object, the current coordinate is divided into two values, thanks to the mask m_D. For example for x_i:

- $x_{i_{m_D}}$, is the preserved part of the coordinate x_i as: $x_{i_{m_D}} = x_i \wedge \neg m_D$;
- $\overline{x_{i_{m_D}}}$, is the part to encrypt of the coordinate x_i as: $\overline{x_{i_{m_D}}} = x_i \wedge m_D$.

By using the pseudo-random sequence S_K created by the PRNG, a value r is pseudo-randomly generated and the mask m_D is applied to it:

$$r_{m_D} = r \wedge m_D. \tag{2}$$

The intermediate value $e = r_{m_D} \oplus \overline{x_{i_{m_D}}}$ is computed and then, the result is merged with $x_{i_{m_D}}$ to give the selectively encrypted coordinate $x'_i = e \vee x_{i_{m_D}}$. We repeat this same process for y_i and z_i. These different operations, which constitute selective encryption, can be summarized using the following equation where c_i is the variable to replace the coordinates of the $i-$vertex:

$$c'_i = ((c_i \oplus r) \wedge m_D) \vee (c_i \wedge \neg m_D). \quad (3)$$

2.4. 3D object decryption

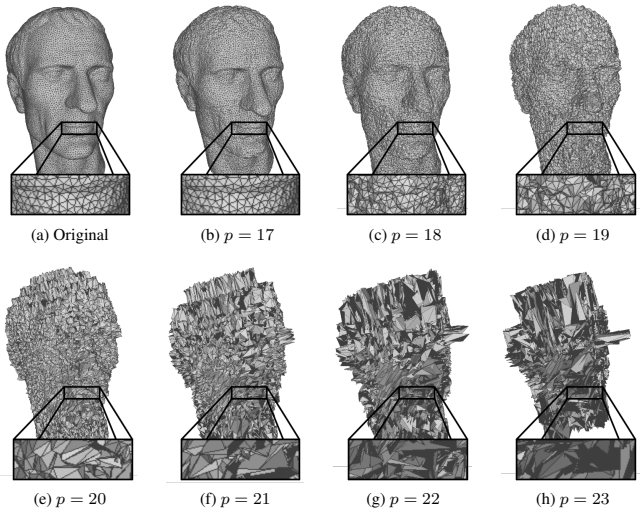
To decrypt the 3D object encrypted by our method, we need to reapply the encryption algorithm presented in Fig. 1. Indeed, by using the same secret key and the same degradation level D, as input, we use the encrypted 3D object and in the output we obtain a decrypted 3D object which corresponds to the original one.

3. EXPERIMENTAL RESULTS

This section presents experimental results of our selective encryption method of 3D objects. Firstly, in Section 3.1, we present an application of our method on a 3D object for two different values of degradation level. Then in Section 3.2, we realize a statistical analysis and a comparison between the results of our two parameters. We analyze the security of our scheme in Section 3.3. Finally, in Section 3.4 we compare our proposed 3D selective encryption method to previous methods.

3.1. Application of our method

Fig. 4 presents our method using the D-SW mask with the degradation level $D =< p, l >$ with $p \in \{17, ..., 23\}$ where p is the position of the first bit to encrypt and $l = 1$, the number of bits to encrypt.

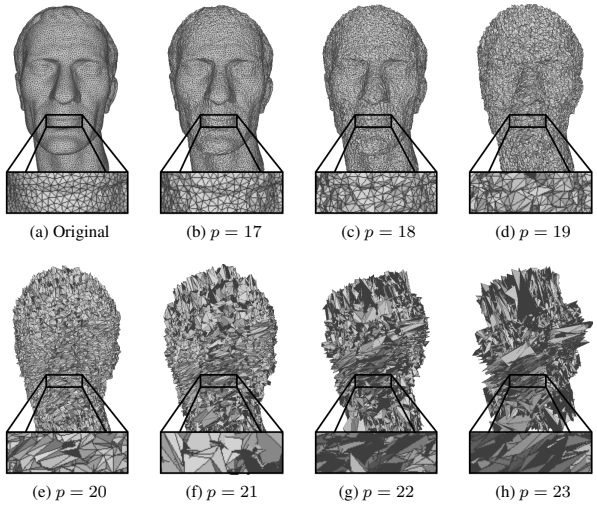


(a) Original (b) $p = 17$ (c) $p = 18$ (d) $p = 19$

(e) $p = 20$ (f) $p = 21$ (g) $p = 22$ (h) $p = 23$

Fig. 4: Selective encryption as a function of the degradation level $D =< p, 1 >$.

Fig. 4.a represents the original 3D object, when Fig.4.b-i are the selectively encrypted 3D objects with a specific level of degradation D. We find that our method allows us to moderately encrypt a 3D object in order to keep it recognizable, but geometrically distorted. We also note that from a specific value of degradation level (around $p = 21$), it is no longer possible to recognize the content of the encrypted 3D object by the human visual system. In fact, for this example, with p less than 19 we have a transparent encryption, with p between 20 and 21 we have sufficient encryption and with p up to 22 we succeed in obtaining visual confidentiality.



(a) Original (b) $p = 17$ (c) $p = 18$ (d) $p = 19$

(e) $p = 20$ (f) $p = 21$ (g) $p = 22$ (h) $p = 23$

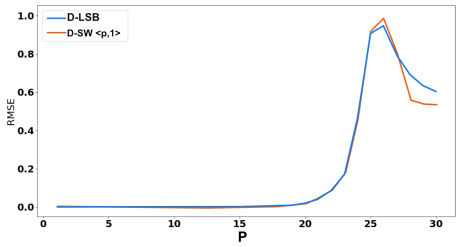
Fig. 5: Selective encryption as a function of the degradation level $D =< p, p >$.

Fig. 5 illustrates the application of our selective encryption method, with a D-LSB mask or $D =< p, p >$ for several levels of degradation using $p \in \{17, ..., 23\}$. Fig. 5.a represents the original 3D object, Fig.5.b-i show selectively encrypted 3D objects with a specific level of degradation D. We observe that using this mask which consists of encrypting the p least significant bits, this approach shows very similar results to the previous approach to the human visual system. We note experimentally that geometrical distortions become really visible starting from $p = 17$.

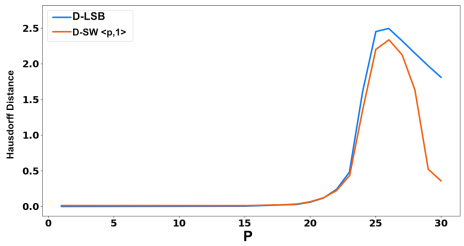
3.2. Statistical analysis

In these experiments, we compare encrypted 3D objects of both masks to the original 3D object. We used 380 3D objects from the Princeton Mesh Segmentation database [9]. Then, we set different values for the parameters as described in Section 2. The metrics used to compare 3D objects are *Root Mean Square Error* (*RMSE*) and *Hausdorff Distance* (*HD*) [10, 11].

With the D-SW mask, Fig 6.a and 6.b show the results for *RMSE* and *HD*. As we can observe in Fig. 6, even if we encrypt a small part of the 3D object geometry, 3 bits out of 96 by vertex, encrypted 3D objects can have their content fully protected from visual confidentiality to transparent encryption through sufficient encryption. We provided metric results from the selective encryption method with the D-LSB mask. The results are presented in Fig.6.a and 6.b. *RMSE* and *HD* show the same dynamic as for the D-SW mask. Both metrics have their values near 0 when $p \in \{1, ..., 18\}$. *RMSE* and



(a) RMSE Curve



(b) HD Curve

Fig. 6: Results of metrics (*RMSE* and *HD*) on Princeton Database [9] as a function of the selected degradation level parameter p for D-LSB and D-SW masks.

HD rise rapidly between $p = 20$ and $p = 25$. The mean *HD* reaches a maximum at around 2.5 at $p = 26$, the value drops quickly. When the method starts to encrypt exponent bits. The same happens for *RMSE*. By using a D-SW mask, we denote the impact of the most significant selected bit. Indeed, when the same mask starts at the same bit as D-LSB mask, geometrical distortions over a 3D object are very similar to the human visual system regardless of the number of encrypted bits. For example, Fig. 4.e and Fig. 5.e represents two 3D objects encrypted using the degradation level $D =< 19, 1 >$ and $D =< 19, 19 >$ respectively. Since the method starts the encryption from the 19^{th} bit of the coordinates then, for the human visual system, both of them are visually similar after encryption. But the first 3D object has only one encrypted bit, this means our method can encrypt only 3.125% of the geometry of the 3D object to hide the content.

3.3. Security analysis

In this section, we discuss the security of our method. First of all, in Section 3.3.1, we study the sensitivity of the secret key in our scheme. In Section 3.3.2, we analyze visual sensitivity of the degradation level. Then, in Section 3.3.3 we argue about the fragility linked to the nature of our method, under intelligent brute force attacks. In section 3.3.4, we discuss attacks based on mesh processing algorithms.

3.3.1. Key sensitivity

To test the sensitivity of the secret key, we encrypt a 3D object M using secret key K. Then, we generate a set of keys K' by changing only one bit of the original key and we use these set of keys to decrypt the encrypted 3D object M'. At last, we compare these decrypted 3D objects to the original M. In Fig. 7, we observe that only the original key returns a 3D object having an RMSE equals to 0. While the decryption

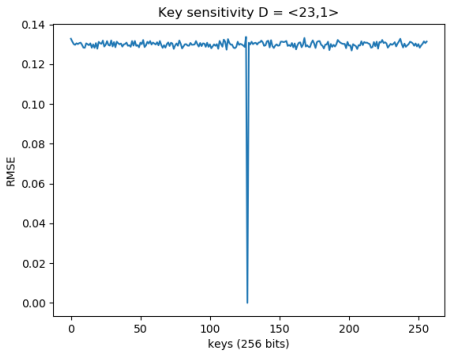


Fig. 7: Results of RMSE metric for decrypted 3D objects using 256 secret keys K' by changing only one bit of the secret key K.

using all the keys only generates 3D objects having a similar RMSE, near 0.13 and remaining remote from the original model.

3.3.2. Visual sensitivity of degradation level

When someone tries to decrypt a 3D object with the correct secret key K, but an incorrect degradation level D_{bad}, the 3D object is not revealed. In the case of a D-LSB mask, when $D_{bad} \leq D$ where D is the correct degradation level for decryption, D_{bad} bits are decrypted (if the user successes to resynchronize it), but not the most significant ones from the mask. But, as explained in Section 3.2 with the D-SW mask, the encryption of only one bit, is enough to protect the content as much as the full D-LSB mask.

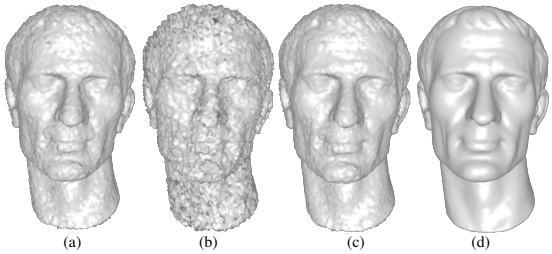


Fig. 8: Decryption with the incorrect degradation level: a) Encrypted 3D object with $D =< 18, 18 >$, b) Decrypted 3D object with $D =< 19, 19 >$, c) Decrypted 3D object with $D =< 17, 17 >$ and d) Original 3D object.

Furthermore if $D_{bad} \geq D$, the D-LSB mask is decrypted, but $D_{bad} - D$ more significant bits of the mask are still encrypted then, the 3D object is more or less unrecognisable. The same happens for D-SW mask.

3.3.3. Fragility of selective encryption schemes

As a selective encryption scheme, the quantity of encrypted bits is lower than full encryption. This makes the method faster for encryption and decryption, but also sensitive to diverse attacks. Instead of looking for secret key K, one strategy is to attempt to recover the encrypted content by brute force attacks. Naively attacking our method requires the attacker to find true values of bits for each coordinate of each

vertex in the encrypted 3D object, which corresponds to finding the correct combination among $2^{3\times N\times l}$.

As shown in [12], the author has explained clearly the weakness of a partial encryption approach. He has designed a cryptanalysis system for partially encrypted images by using information around the encrypted data. If we take into consideration his approach for our method, and use some public and relevant information such as the degradation level or the connectivity of the 3D object, an attacker can design some heuristics guiding the choice of combination. It can reduce by one the degradation level D parameter p of an encrypted 3D object by computing the right combination among $2^{3\times N}$ where N is the number of vertices, just to retrieve at least one bit of each encrypted coordinate.

By repeating l times the search for the right combination for one bit by coordinate, an adversary can slowly decrypt a 3D object. Each time, the attacker conserves, for the next iteration, the *closest* 3D object, meaning the one which reveals coherent information on the 3D object. Realizing such attack would obviously require a significant investment in time and power, which would really be prohibitive to be successful as the number of vertices in the encrypted 3D object increases.

3.3.4. Mesh processing attacks

Another possible attack is to try to directly process the 3D encrypted object. As our method generates encrypted 3D objects allowing us to recognize the content of the 3D object as transparent encryption, it is normal that our approach is sensitive to suck attacks, as smoothing or reconstruction. So, for some values of degradation level D, for example a Laplacian smoothing [13], with a deformation factor $\lambda = 0.3$ and 100 iterations, can very closely recover a 3D object to its original state.

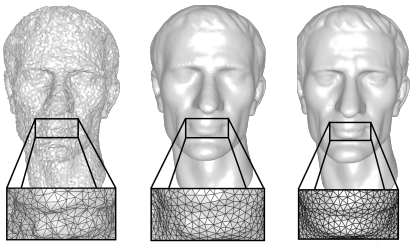


Fig. 9: Laplacian smoothing attack: a) Encrypted 3D object at $D =< 18, 1 >$, b) Result after smoothing ($\lambda = 0.3$ and NbIterations = 100), c) Original 3D object.

As illustrated in Fig. 9, Laplacian smoothing transforms the encrypted object into a slightly similar one to the original. However, we observe losses of information in very distinct areas of the 3D objects, particularly in low frequency ones. Using *HD*, the smoothed 3D object has a value for the metric around 1.550×10^{-2}, whereas the encrypted one is about 1.689×10^{-2}. Even if visually, it gets closer to the original, the 3D object is still different. Furthermore, this kind of attack becomes ineffective until a degradation level is reached. For example, Fig. 10 represents a smoothing attack where the

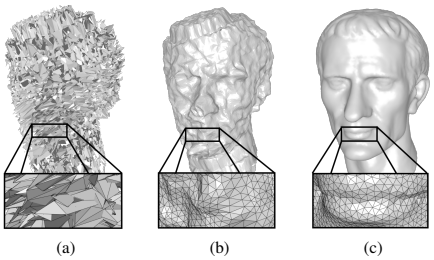


Fig. 10: Laplacian smoothing attack: a) Encrypted 3D object at $D =< 21, 1 >$, b) Result after smoothing ($\lambda = 0.3$ and NbIterations = 100), c) Original 3D object.

given result is, for the human visual system, very different to the original 3D object. The value of *HD* for the smoothed 3D object is about 8.416×10^{-2}, meanwhile the encrypted one has a value of 1.689×10^{-2} for the same metric. As expected, the smoothed 3D object strays away from the original 3D object and the increase in *HD* reveals this too. As shown in Fig. 11, the choice of the degradation level D allows us to select the desired confidentiality from visual confidentiality to transparent encryption through sufficient encryption. However, in order to properly select the right values for the degradation level, we have seen that objective metrics cannot help us to determine such thresholds for our categories. This is why we require a more subjective metric to make the selection more easier.

3.4. Comparison with other methods

Table 1 resumes comparison points of our proposed method with state-of-the-art 3D encryption methods. Our proposed method encrypts a range of bits of the coordinates of all vertices. This allows us to properly select the desired confidentiality as a function of the needs of the user. Alone the connectivity of a 3D object is not enough to recognize a 3D object, only the genus at least. So, we aimed to protect the geometry rather than the connectivity because as we previously explained in Section 3.3, the connectivity of the secret 3D object can be reconstructed. Thus, the method of Cho *et al.* which encrypts only indices of faces [5] can have its content recovered. Unlike Eluard *et al.* [7] whose methods preserve the bounding box of the secret 3D object, we hide it within the bounds of the relative error of floating point values. This is not perceptible for low degradation level. However, if a visual confidentiality is requested to protect the 3D content then, the bounding box of the encrypted 3D object grows, revealing nothing of the dimension of the secret 3D object.

4. CONCLUSION

In this paper, we proposed a format compliant encryption method, which selectively protects a 3D object by partially encrypting the floating representation of coordinates of vertices using a secret key K. Our method allows us to choose a degradation level which gives us control over geometrical artifacts brought to the 3D object. Encrypted 3D objects can be displayed in 3D scenes, because the internal structure of the

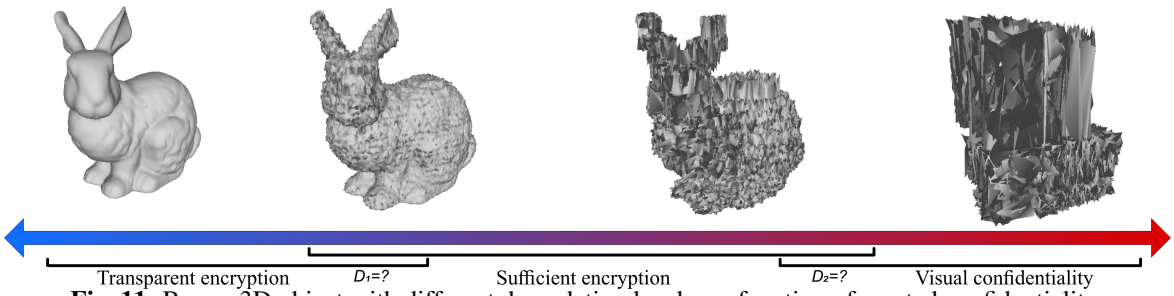


Fig. 11: Bunny 3D object with different degradation level as a function of wanted confidentiality.

Methods	[5]	[6]	*Coordinate Shuffling* [7]	*Dithering* [7]	*Fragment Scaling* [7]	**Proposed**
Protected content	Connectivity	Both	Geometry	Geometry	Geometry	**Geometry**
Bounding box	Preserved	Hidden	Preserved	Preserved	Preserved	**Hidden**
Computational Complexity	Moderate	High	Low	Low	Moderate	**Low**
Transparent encryption	Yes	Yes	No	Yes	Yes	**Yes**
Sufficient encryption	No	No	No	Yes	Yes	**Yes**
Visual confidentiality	No	No	Yes	Yes	No	**Yes**

Table 1: Comparison of our 3D selective encryption method with previous work.

3D object files has been preserved. To select the bits to encrypt, a strategy to build a mask was proposed. D-SW mask, as we call it, is using a sliding window to choose the position of the first bit of the mask and the number of bits to encrypt in each coordinate. We also present a special case, D-LSB Mask, which encrypts the l least significant bits of coordinates. By comparing results, we observe that encrypted 3D objects are identical to the human visual system, but each of them brings different useful properties, like a function of the desired application. Indeed, D-LSB mask has a higher computational security to brute force attacks, whereas, D-SW mask can produce the same visual result faster by encrypting at least one bit per coordinate, so 3.125% of the 3D object geometry. Since the method is based on the standard norm of floating values, it can be adapted for any precision of the standard used in any binary 3D object formats. We also present the computational security level of the method. In particular, the cryptographic primitive can be easily replaced by another. Future work will concentrate on perceptual evaluation of encrypted 3D objects in order to establish a subjective measurement to define the human visual system thresholds between visual confidentiality, transparent encryption and sufficient encryption.

5. REFERENCES

[1] T. Harte and A. G. Bors, "Watermarking 3d models," in *IEEE International Conference on Image Processing (ICIP)*. 2002, IEEE.

[2] K. Wang, G. Lavoué, F. Denis, and A. Baskurt, "Robust and blind mesh watermarking based on volume moments," *Computers & Graphics*, 2011.

[3] V. Itier, W. Puech, and J.-P. Pedeboy, "High capacity data-hiding for 3d meshes based on static arithmetic coding," in *IEEE International Conference on Image Processing (ICIP)*. 2015, IEEE.

[4] D. Koller, M. Turitzin, M. Levoy, M. Tarini, G. Croccia, P. Cignoni, and R. Scopigno, "Protected interactive 3d graphics via remote rendering," in *ACM SIGGRAPH 2004 Papers*. 2004, SIGGRAPH '04, ACM.

[5] M. Cho, S. Kim, M. Sung, and G. On, "3d fingerprinting and encryption principle for collaboration," in *International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution (AXMEDIS)*. 2006, IEEE.

[6] M. Gschwandtner and A. Uhl, *Protected Progressive Meshes*," *in Advances in Visual Computing* 2009.

[7] M. Éluard, Y. Maetz, and G. Doërr, "Impact of geometry-preserving encryption on rendering time," in *IEEE International Conference on Image Processing (ICIP)*. 2014, IEEE.

[8] IEEE, "Ieee standard for floating-point arithmetic," *IEEE Std 754-2008*, 2008, IEEE.

[9] X. Chen, A. Golovinskiy, and T. Funkhouser, "A benchmark for 3D mesh segmentation," *ACM Transactions on Graphics (TOG)*, 2009, ACM.

[10] W. Rucklidge, *Efficient visual recognition using the Hausdorff distance*, 1996, Springer.

[11] N. Aspert, D. Santa-Cruz, and T. Ebrahimi, "Mesh: Measuring errors between surfaces using the hausdorff distance," in *IEEE International Conference on Multimedia and Expo (ICME)*. 2002, IEEE.

[12] A. Said, "Measuring the strength of partial encryption schemes," in *IEEE International Conference on Image Processing (ICIP)*. 2005, IEEE.

[13] L. Herrmann, "Laplacian-isoparametric grid generation scheme," *Journal of the Engineering Mechanics Division*, 1976.