Format-Compliant Selective Secret 3D Object Sharing Scheme

Sebastien Beugnon, William Puech, Jean-Pierre Pedeboy

Format-Compliant Selective Secret 3D Object Sharing Scheme

Sébastien Beugnon, *Student Member, IEEE,* William Puech, *Senior Member, IEEE,* and Jean-Pierre Pedeboy

***Abstract*—This paper presents a new method of Secret 3D Object Sharing (S3DOS) which allows sharing of 3D objects whilst preserving its file format by selectively encrypting a 3D object in order to sufficiently protect the visual nature of the content. This scheme, named *Format-Compliant Selective (k, n) Secret 3D Object Sharing* (FCSS3DOS), modifies selected bits of the vertices of a 3D object to protect visual content by inducing geometrical distortions. These distortions are controlled thanks to the application of a degradation level at the beginning of the sharing process. To reconstruct the secret 3D object, at least k shared 3D objects among n have to be combined in order to remove the geometrical distortions and recover the exact original 3D object. Experimental results are presented and evaluated to showcase the feasibility of the proposed scheme.**

***Index Terms*—3D Selective Encryption, Secret Sharing, Information Sharing, 3D Object, Content Protection.**

I. Introduction

With the rise of data exchange and technology evolution, multimedia content takes an important place in world data traffic and in applications. Visual data such as images, videos or 3D objects are transmitted over networks and stored on the cloud. Several previous work have proposed to secure transmissions and storage for this specific type of data [1]–[3]. Concerning 3D objects, they are used in a large number of applications, for example; medical, simulation, video games, animation and special effects for collaborative work. The consumption of 3D objects by large audiences has become a lucrative market which can take the form of 3D object downloading platforms in any format. With this ever faster development, creators and owners, given the cost of production, see their creations as financial assets needing protection from piracy by illegal copying.

The classical method of protection is the use of an encryption algorithm to encrypt content into an unreadable character sequence with the help of a secret key. Only the secret key holders can decrypt the encrypted content to reconstruct the original message. However, the secret key is an essential element in the cryptosystem and if it is lost or even stolen, the security of the secret content is compromised. Furthermore these encryption methods are mainly based on the principle of single-container. This means if the container of the secret is tampered with or lost, the secret will also be lost, even with the correct secret key. Therefore, new mechanisms protecting secrets whilst also allowing redundancy of storage are necessary.

Secret Sharing (SS) schemes have been proposed in 1967 independently by Shamir [4] and Blakley [5] to solve these issues linked to standard encryption methods. More precisely, Shamir [4] and Blakley [5] designed two distinct methods to share a secret among n participants and to reconstruct it with any group of at least k participants, noted (k, n)-threshold scheme as illustrated in Fig. 1.

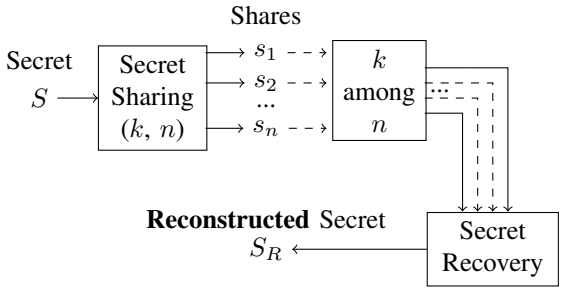


Fig. 1. Overview of a (k, n)-threshold SS scheme.

In the last few decades, many methods of SS have been applied to the field of image processing as Secret Image Sharing (SIS) [6]–[9]. Using this same direction, several papers have been published on Secret 3D Object Sharing (S3DOS) [10]–[14]. Elsheh and Hamza proposed to directly apply SS schemes on 3D objects to protect them using Blakley's SS scheme [5] or Thien and Lin's SIS approach [7]. In 2015, Anbarasi and Mala proposed, a method to share m 3D objects simultaneously with a verification system in order to detect forged shares during their reconstruction [11]. Martín Del Rey proposed a (n, n) multiple secret sharing scheme for 3D objects represented by voxels based on a cellular automata approach [12]. In 2016, Tsai proposed a S3DOS scheme based on reversible 3D data hiding which shares only an approximated geometry of the secret 3D object in cover 3D objects [13]. Recently, Lee *et al.* presented an application case of SS for streaming purposes in order to obtain a group of n 3D objects when at least k of them have been downloaded in high quality [14].

In this paper, we propose a new scheme to share a 3D object called Format-Compliant Selective Secret 3D Object Sharing (FCSS3DOS). This approach selects relevant bits within coordinates of vertices in the 3D object geometry and then generates n parts for each vertex from the selection. These information blocks substitute the selected original bits in the shared 3D objects. Each of these shared 3D objects, can be visualized, for example in a 3D environment, but its content is visually protected by geometrical distortions due to the sharing process. The desired degradation level is defined at the beginning of the sharing process, in order to control how much geometrical distortion is necessary.

The rest of this paper is organized as follows. Section II presents the main schemes of SS, SIS and S3DOS. Our proposed FCSS3DOS scheme is detailed in Section III. Then, Section IV provides experimental results and analysis. Finally, Section V concludes and presents some directions for future work.

II. Related Work

First, in Section II-A we present the first Secret Sharing (SS) schemes proposed by Shamir [4] and Blakley [5]. Then, applications of SS approaches in the field of 2D image processing, known as Secret Image Sharing (SIS) are presented in Section II-B. Section II-C presents schemes designed for Secret 3D Object Sharing (S3DOS) that have been proposed in the last decade. Finally, Section II-D presents current state of the art methods on selective encryption for 3D objects.

A. Secret Sharing Schemes

In this section, we present the most famous SS schemes, namely Shamir's SS scheme [4] and Blakley's SS scheme [5]. Firstly, in Shamir's SS scheme, the secret S is interpreted as an unsigned integer defined on the finite field $\mathbb{F}_P$, where $|\mathbb{F}_P| = P$ [4]. To protect the secret S interpreted as an unsigned integer, Shamir's proposition consists of using polynomials over a finite field where P is a prime number and respects:

$$0 < k \leq n < P, \tag{1}$$

$$S < P, \tag{2}$$

with k the required number of participants to reconstruct the secret S and n the total number of participants.

This (k, n)-threshold scheme distributes shares to each of the n participants and allows the reconstruction of the secret when at least k of n participants combines their shares to solve a polynomial interpolation. Without a group of this size, no information is leaked from shares. During the sharing phase, each participant receives a unique indice value x_j, where $0 < x_j < P$ and $j \in \{0, ..., n-1\}$. A polynomial of degree $(k-1)$ is built, where $(k-1)$ coefficients are generated pseudo-randomly to form $A = \{a_i | i \in \{0, ..., k-1\}\}$ with $a_i < P$ and $a_0 = S$:

$$f(x) = \sum_{i=0}^{k-1} a_i \times x^i \mod P. \tag{3}$$

Thus, $f(0)$, which is equal to a_0, corresponds to the value of the secret S. Each of the n participants then receives the information $(x_j, y_i = f(x_j))$ as a personal share of the secret.

As illustrated in Fig. 2, the reconstruction of the secret S can be achieved by a polynomial interpolation with at least k shares. These shares are interpreted as points which are on the curve of the used polynomial. For example, points $(x_i,\ f(x_i))$ and $(x_j,\ f(x_j))$ in Fig. 2 are the given shares used to interpolate the polynomial of degree 1 illustrated by the red line. Moreover in Fig. 2, the blue and green curves respectively represent a polynomial of degree 2 and 3, which can reconstruct the secret S when at least 3 or 4 shares

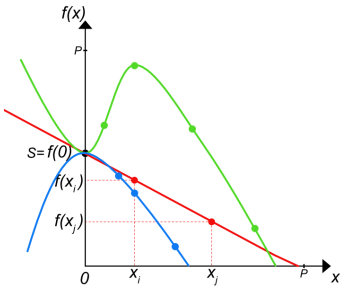


Fig. 2. Example of Shamir's scheme with threshold: in red (2,n), in blue (3, n) and in green (4,n).

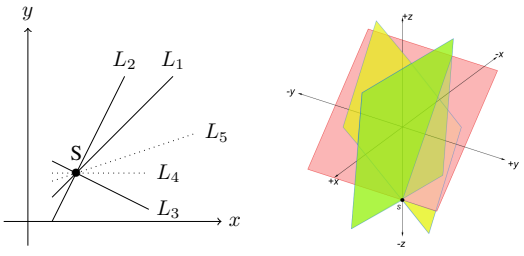
respectively are used as points to interpolate the polynomial. Therefore, the interpolation is used to determine $f(0)$ which is the constant term a_0. To reconstruct the secret S (which is equal to a_0), a group of k or more participants using Lagrange's interpolation, can determine the used polynomial $f(x)$ for sharing and by extension the constant term a_0 of the polynomial corresponding to the secret S:

$$f(x) = \sum_{i=0}^{k-1} y_i \times \prod_{u=0, i \neq u}^{k-1} \frac{x - x_u}{x_i - x_u} \mod P. \tag{4}$$

During the same year in 1979, Blakley proposed a SS scheme based on hyperplane geometry and linear geometry [5]. He defined the secret as a point $S = (x_0, x_1, ..., x_{k-1})$ in k-dimensional space and gave participants k-dimensional hyperplanes where the point S lies. A k-dimensional hyperplane can be defined by a linear equation:

$$b = \sum_{i=0}^{k-1} a_i \times x_i, \tag{5}$$

where, a_i with $i \in \{0, ..., k-1\}$ is the i-th coefficient of the k-dimensional hyperplane defined by the set $A = (a_0, ..., a_{k-1})$ and b the sum result.



(a) (2, n) (b) (3, n)

Fig. 3. Examples of secret point reconstruction for Blakley's scheme with threshold, a) $k = 2$ and b) $k = 3$.

Thanks to hyperplane geometry, the secret point is the intersection point of any groups of k or more hyperplanes. Fig. 3.a and Fig. 3.b illustrate how hyperplanes intersect with each other in only one point for $k = 2$ and for $k = 3$, respectively.

B. Secret Image Sharing Schemes

In this section, we present inspiring SIS schemes like the first popular SIS scheme designed by Thien and Lin [6] and Yang *et al.*'s scheme [9].

In 2002, Thien and Lin proposed to apply Shamir's SS scheme for 2D images by sharing each pixel value of a secret image [6]. The particularity of their scheme is to fully use all coefficients a_i of the polynomial (See Eq. (3)). Instead of just replacing the coefficient a_0 by a pixel value, they pack pixel values in groups of k pixels and replace the coefficients of polynomial by them. Therefore, they transmit the generated images to participants, named *shadow images*, which have $\frac{1}{k}$ of the secret image size. Shadow images look like random noise and reveal nothing of the secret image until k of them are combined to reconstruct the secret image. Since Shamir's SS scheme requires a finite field defined by a prime number, the authors chose $P = 251$. However, since this prime number could not cover all pixel values, they proposed to truncate pixel values above 250 to stay in the finite field definition and provided a lossy scheme. Then, in order to create a lossless scheme, they proposed that when a pixel value is equal to or greater than 250, to store the overflow pixel values, this is the truncated part of pixel values lost at the sharing process. Between 0 and 5, these values are also shared as supplementary pixels of the secret image. So, the size of shadow images can increase if a lot of pixel values of the secret image are above 250.

Yang *et al.* proposed a lossless SIS scheme with steganography and authentication properties to prevent false stego-image and dishonest participants [9]. To avoid the problem known in the SIS scheme of Thien and Lin [6], they decided to use a Galois Field $GF(2^8)$ in the calculation of polynomials. This way, their scheme is lossless and does not require additional pixels.

C. Secret 3D Object Sharing Schemes

In this section, methods of S3DOS are presented. By directly applying SS schemes on 3D objects, the proposed method of Elsheh and Hamza protects the content [10]. All the vertices and the faces of the secret 3D object are shared using Blakley's SS scheme [5] to produce n shares as 3D planes, where $k = 3$ and $n > 3$. The authors proposed to compute shares of the secret 3D object using $Z = a \times X + b \times Y + c$, where variables X, Y, Z are coordinates of a shared vertex or indices of a shared triangles and a and b are coefficients which are pseudo-randomly selected. To secure their scheme, they use a finite field $\mathbb{F}_P$ defined by a prime number P as proposed by Shamir [4]. As a function of the selected prime number P, all values cannot be represented. So, when they share vertex indices of 3D object triangles, they increase the number of vertices by copying the last vertex until they reach P vertices. Therefore, their method does not preserve the size of the secret. The output of their method is strictly binary data without any compliance to the original format of the secret 3D object. Elsheh et Hamza also proposed to use Thien and Lin's SIS scheme's approach [7] to reduce the size of the share in order to ease their transmission over the network. As Thien and Lin, instead of packing pixel values, they pack coordinates from the same vertex, and indices of the same triangle to reduce the size of each shared vertex and triangles by 3 and by extension of the share itself. Furthermore, they proposed to compress the data of the 3D object before, the sharing process, using data compression algorithms like Huffman coding or ZLIB [15], [16] as proposed in [17] to reduce size and redundancy of the data.

In their paper, Anbarasi and Mala propose to share multiple 3D objects [11]. To realize this, they used the same approach as Elsheh and Hamza by compressing 3D data before sharing it, but they replaced the Blakley's SS scheme by Shamir's SS scheme. More exactly, an improved version of the scheme by Yang *et al.* [18] was selected, allowing to share multiple secrets simultaneously. If m, the number of secrets to hide is greater than the desired threshold of participants k, then these secrets are shared using a polynomial of degree $m - 1$, but $m - k$ shares are made public. Anbarasi and Mala's scheme also provides a secure system for share distribution based on RSA and Diffie Hellman approaches proposed in [19].

In the scheme proposed by Tsai [13], vertices of a secret 3D object are encoded into a series of integer values using a space subdivision representation of the secret 3D object. The sequence of integer values associated to a vertex corresponds to its coordinates in the space subdivision structure. This is used to approximately represent the point cloud of the secret 3D object in order to minimize its storage. Then, the S3DOS scheme uses reversible 3D data hiding to share these integer values in cover 3D objects. In order to be able to store the desired quantity of information, the size of cover 3D objects has to be increased using sampling concepts to expand their capacity for 3D data hiding. Shares are generated using Shamir's SS scheme and the series of integer of 3D vertices as input. Finally, these shares are inserted into 3D cover objects. The reconstruction step requires the bounding box of the secret 3D object and the space subdivision used in order to properly recover the point cloud of the secret 3D object. Unlike previous methods, this approach allows the visualization of shares, or in this case of steganographied 3D objects, but not of a degraded version of the secret. It is also more resilient to similarity transformation attacks.

Instead of 3D objects built with polygonal representation, Martín Del Rey proposed to apply secret sharing for voxel-based 3D objects [12]. The proposed (n, n)-threshold scheme allows users to share multiple voxel-based 3D objects using a cellular automaton approach with reversible memory, which the author previously defined [20]. Distributed shares can be visualized in 3D voxel environment.

Through their application, Lee *et al.* presented an efficient method to obtain a group of n 3D objects when at least k of them have been downloaded in high quality [14]. This method's objective is to ease 3D object visualization by letting at the users a render of a simplified version when the original object is downloaded. To do this, they select a set of n 3D objects and then they simplify and compress them using the EdgeBreaker algorithm [21] and LZMA. Compressed data are shared using Reed-Solomon codes [22]. Shares are then inserted using data insertion methods in copies of the original

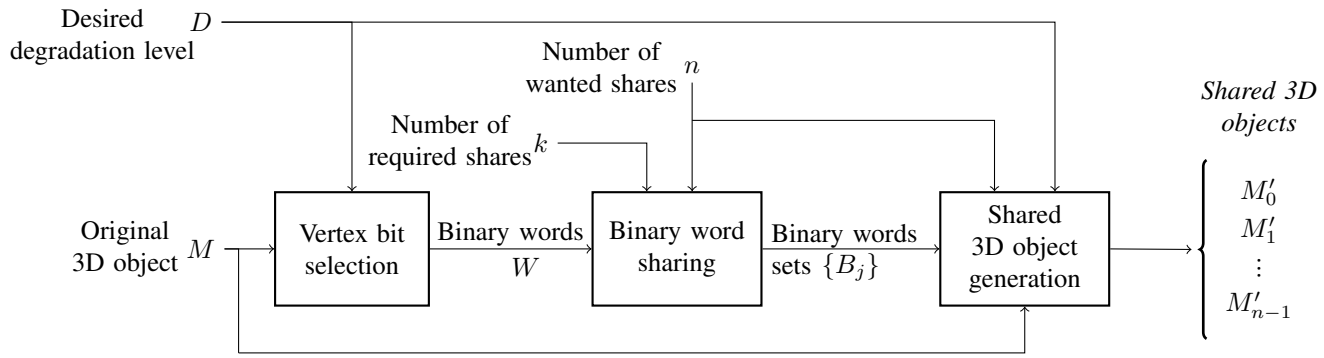


Fig. 4. Overview of the proposed FCSS3DOS scheme.

3D objects [23], [24].

D. 3D selective encryption

Protection against unauthorized access has recently received some attention [25], [26]. Full encryption does not take into account the content of encrypted data. Media content is processed as binary files, destroying the internal structure of files during the encryption process. Selective encryption is a category of encryption algorithm proposing to only encrypt a subset of the data of a media in order to maintain its format compliance. This approach also aims to reduce the quantity of information to encrypt while maintaining a sufficient level of security. 3D selective encryption regroups methods of selective encryption approaches applied on 3D objects like Gschwandtner and Uhl who used progressive mesh representation with layers of refinement [25]. Usually employed to transmit meshes 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. This way, an user can decrypt the layer of refinement as a function of the user's access rights.

Watermarking approaches for 3D objects have been mainly studied [27], [28]. These approaches propose to embed secret messages into 3D objects in the spatial domain or in the spectral domain in order to be more resilient against attacks. Recently, Jian *et al.* [28] proposed a reversible data hiding method embedding a secret message into a 3D object which has been encrypted. The novelty of their approach is that they can reverse the modifications induced in the 3D encrypted object or the 3D decrypted object thanks to a second secret key used at the data hiding step.

Later, Éluard *et al.* presented geometry-preserving encryption methods (GPE) [26]. These methods preserve properties like the bounding box or the convex hull in order to minimize impact on rendering time. The authors present *Coordinate Shuffling* which is a permutation of coordinates using a secret key. Geometrical information is preserved, but the approach ensures a full confidentiality of the visual content. Éluard *et al.* also describe an approach where a pseudo-random noise is applied over vertices of the 3D object. This method, called *Dithering*, proposes to control the amplitude of the noise α applied on all vertices with a wrap around property in order to maintain the bounding box. The quality of the encrypted 3D object could vary as a function of the amplitude of the noise letting the content be visually recognizable or totally hidden. Three levels of selective encryption are defined in order to answer specific needs for 3D object protection:

- **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.

III. Format-Compliant Selective Secret 3D Object Sharing Scheme

In this section, the proposed FCSS3DOS scheme is presented. The proposed method takes control over geometrical distortions induced by sharing selected bits from vertices of a 3D object as a function of a desired degradation level. The selected bits are substituted in the shared 3D objects by binary words built during the sharing step according to the chosen approach (Shamir or Blakley). As a reminder, a 3D object is represented by a 3D vertex cloud and edges connecting vertices between them and building faces. Therefore, the 3D object is defined as $M = (V, F)$, where V is the set of vertices and F is the set of faces. The number of vertices is noted $|V|$ and the number of faces $|F|$.

As illustrated in Fig. 4, in addition to the 3D object to share M, the method requires three parameters as inputs: k the number of required participants to reconstruct the secret 3D object, n the total number of wanted shared 3D objects and the desired degradation level D. The method is composed of three main tasks: vertex bit selection, binary word sharing and shared 3D object generation. First of all, in Section III-A, we introduce the selection of vertex bits to share and to substitute as a function of the desired degradation level D. In Section III-B, the sharing process is presented with the generation of binary words according to the used SS scheme (Shamir or Blakley). Then, in Section III-C, selected bits of 3D

object geometry are substituted by computing binary words at the sharing step in order to generate n shared 3D objects as output. Finally, in Section III-D, the reconstruction method is presented.

A. Vertex bit selection

In order to share a 3D object that is format compliant, our proposed method only applies a sharing scheme on the coordinates of the 3D object's vertices. The desired degradation level D determines which range of bits is selected for the sharing and the substitution step. 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 [29]. 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.

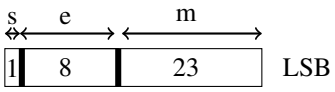


Fig. 5. Representation of a floating value by the *IEEE 754* Norm [29].

As illustrated in Fig. 5, each of the three types of information constituting a floating value has a specific quantity of bits: the bit on the far left 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.

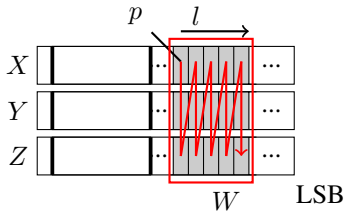


Fig. 6. Selected bits of coordinates used for sharing a vertex as a function of the desired degradation level $D =< p, l >$ and synchronisation strategy to extract bits.

As illustrated in Fig. 6, the degradation level D can vary as a function of two parameters:

- $p \in \{0, ..., 22\}$, the position of the first selected bit;
- $l \in \{1, ..., p+1\}$, the length of the range to share.

The selected sequence of bits is recovered as a binary word noted W for each vertex of the 3D object with $|W| = 3 \times l$. The selection starts with the most significant bit of the coordinate X in the range defined by the degradation level D to end with the least significant bit of the coordinate Z passing through the bits of Y. To extract this binary word from vertex coordinates, bits are read by interleaving the coordinate bits as illustrated in Fig. 6. This approach gathers bits of the same weight from three coordinates in most significant bits of W. Formally, if b_t^X, b_t^Y and b_t^Z are respectively the t-th bits of coordinates X, Y and Z where $t \in \{p-(l-1), ..., p\}$, then:

$$W = (b_p^X, b_p^Y, b_p^Z, ..., b_{p-(l-1)}^X, b_{p-(l-1)}^Y, b_{p-(l-1)}^Z). \quad (6)$$

B. Binary word sharing

During this step, from the binary word W, for each vertex the method generates n binary words B_j, where $j \in \{0, ..., n-1\}$, as $|B_j| = |W|$. This process's objective is to create n binary words to substitute the selected bits on the secret 3D object in the n shared 3D objects. By grouping these binary words, the cess is able to rebuild the binary word W and to reconstruct missing bits of coordinates. In Section III-B1 we present how our proposed scheme works with Shamir's SS scheme and with Blakley's SS scheme in Section III-B2.

1) Using Shamir's scheme: with the SIS approach of Yang *et al.*'s [9], Shamir's SS scheme can be used with Galois field noted $GF(P^m)$ such as $GF(2^m)$ where m is the size in bits of the secret S to share. Therefore, we operate on the Galois field $GF(2^m)$ where $m = 3 \times l$. This way allows us to increase the number of participants and the security of the method as a function of the desired degradation level D. The greater parameter l from degradation level D, the safer the scheme. We note x_j, where $j \in \{0, ..., n-1\}$, the value which is assigned to the j-th participant and its respective shared 3D object:

$$\begin{cases} x_j \in GF(2^m), \\ x_j \neq 0, \\ \forall u, w \in \{0, ..., n-1\},\ u \neq w \Leftrightarrow x_u \neq x_w. \end{cases} \quad (7)$$

Value of x_j is used for all the vertices of the 3D object and has to be transmitted along the shared 3D objects on a private channel in order to reconstruct the secret 3D object. A coefficient set $A = \{a_i | a_i \in GF(2^m) \text{ and } i \in \{1, ..., k-1\}\}$ is pseudo-randomly generated for each vertex. Coefficient a_0 is assigned with the value of W. Then, the method builds n results of polynomial noted $B_j = f(x_j)$ as Eq. (3) using the set A for the coefficient set of the polynomial for the vertex v:

$$B_j = f(x_j) = \sum_{i=0}^{k-1} a_i \times x_j^i. \quad (8)$$

Thanks to Shamir's approach, it is then possible to have a flexible threshold for k as $2 \leq k \leq n$.

2) Using Blakley's scheme: unlike Shamir's method, which is in two dimensions, Blakley's approach is itself based on k dimensions, where k is the required number of participants to reconstruct the secret 3D object. The method considers the secret as a k-dimensional point for each vertex of the 3D object. For this, the binary word W is uniformly split in blocks x_i, where $i \in \{0, ..., k-1\}$, to form the secret k-dimensional point S as:

$$\begin{cases} S = (x_0, x_1, ..., x_{k-1}), \\ W = \bigcup_{i=0}^{k-1} x_i, \\ |W| = \sum_{i=0}^{k-1} |x_i| \end{cases} . \quad (9)$$

Our method then generates n k-dimensional hyperplanes, such as the point S which represents the intersection of hyperplanes. A pseudo-random set of coefficients $A = \{a_i | a_i \in \mathbb{N}^* \text{ and } i \in \{0, ..., k-1\}\}$ is generated for each hyperplane with the following constraints:

- The constraint of **resolution** forces the n k-dimensional hyperplanes generated for each vertex to be distinct and allows the resolution of linear system for $\binom{n}{k}$ combinations of possible hyperplanes to reconstruct the secret 3D object. The verification of such combinations is heavily time-consuming. We force three conditions for this constraint:
  - $a_{k-1} = 1$, this way the n equations of k-dimensional hyperplanes are not linear combinations between them. This makes it possible to write the hyperplane equation as:

$$x_{k-1} = -b + \sum_{i=0}^{k-2} a_j \times x_i; \quad (10)$$

  - $\forall u, v \in \{0, ..., n\}\ u \neq v \Rightarrow h_u \neq h_v$, hyperplane equations h_u and h_v are **unique** (in relation to their coefficients a_j);
  - $\forall i \in \{0, ..., k-1\}, a_i \neq 0$.

  With these three conditions, the method is sure to be able to achieve the expected result for all combinations of hyperplanes.
- Another constraint is that of space, this last one limits the available range of values for each coefficient a_i pseudo-randomly chosen. The space which is used to store the coefficient of hyperplane is limited to the maximal value of 69 bits by the vertex with the mantissa bits and by the desired degradation level D. It is necessary to constrain the bit size of the coefficients a_i and b:

$$|b| + \sum_{i=0}^{k-2} |a_i| = |W|, \text{ with } |W| = 3 \times l. \quad (11)$$

As the coefficients a_i and x_i are considered as integers, it is necessary that the coefficient b can represent the sum defined in Eq. (5). Once the coefficients a_i are pseudo-randomly generated according to the previous constraints, the method solves the hyperplane equation to determine the value of coefficient b and repeats this operation n times in order to generate the set of hyperplanes for each vertex. Then, when coefficients a_i and b are fixed, the method concatenates coefficients to form binary words B_j.

C. Shared 3D object generation

Binary words B_j generated by one of the two approaches presented in Section III-B respects the condition $|B_j| = |W|$: Shamir by the use of Galois field $GF(2^{|W|})$, and Blakley with Eq. (11).

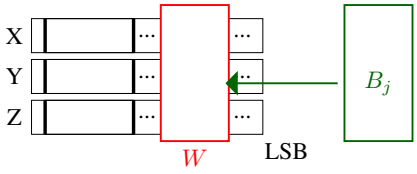


Fig. 7. Bit substitution of W by B_j (where $j \in \{0, ..., n-1\}$).

As illustrated in Fig. 7, our method reuses the same synchronization strategy employed for the selection, presented in Section III-B to substitute W by B_j in coordinates of each vertex of the secret 3D object in order to generate n new 3D objects called, *shared 3D objects* and noted M'_j. Coordinates generated by the substitution of W by binary words B_j become different from those of the original 3D object by the appearance of geometric distortions in the shared 3D objects.

D. Secret 3D object reconstruction

As illustrated in Fig. 8, the reconstruction process of the secret 3D object is quite similar to the one used for sharing.

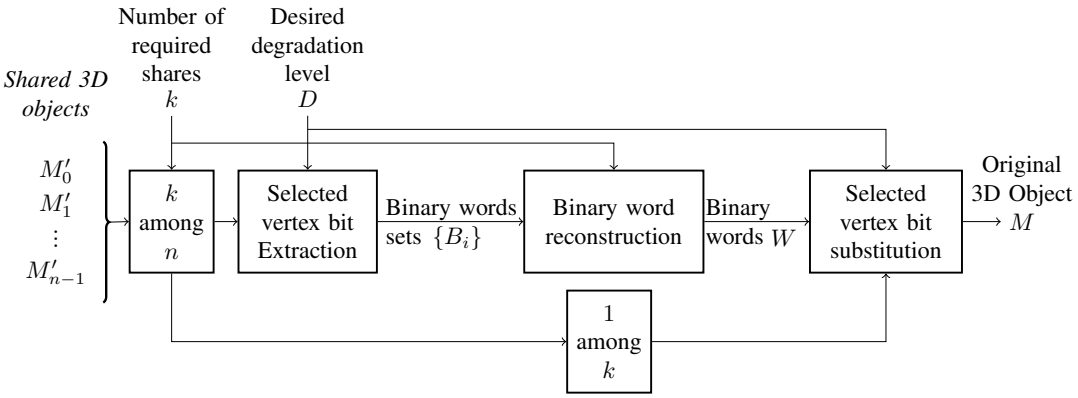


Fig. 8. Reconstruction process overview.

As illustrated in Fig. 8, when k shared 3D objects from the authorized set are grouped together, the proposed method extracts a set of binary words $\{B_i\}$, where B_i is the binary word for the current processed vertex of the shared 3D object of the i-th participant among the k of the reconstruction group. Then, this set is transferred to the reconstruction method of vertex data sharing:

- For Shamir's approach, Lagrange's interpolation (see Eq. (4)) is used on each vertex with the point set $\{(x_i, B_i)\}$. For each vertex, when the number of participants reaches the threshold k, the interpolation returns W, which is exactly the same as the one extracted from the original 3D object.
- For Blakley's approach, the reconstruction method recovers a set of coefficients for a k-dimensional hyperplane for each vertex. The method gathers hyperplanes associated to the same vertex in a set and tries to solve the linear system formed by these hyperplanes to recover the secret point S for each vertex. If the hyperplanes are belonging to the authorized set previously generated, then coordinate bits of S are concatenated to create the binary word W.

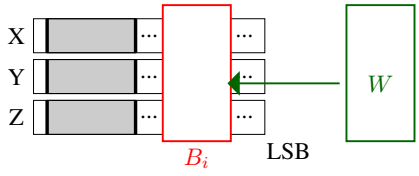
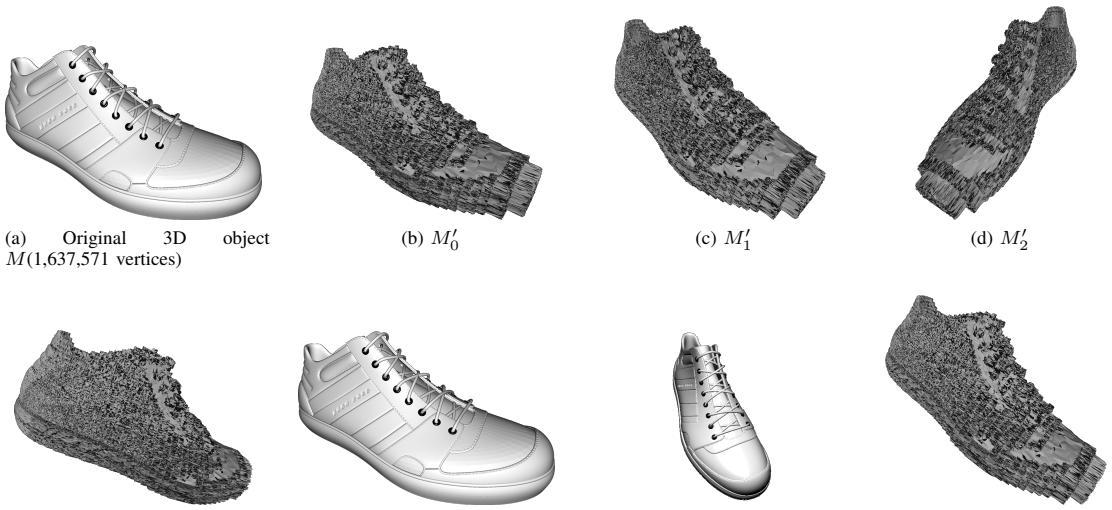


Fig. 9. Bit substitution of B_i by W of 3D object M'_i.

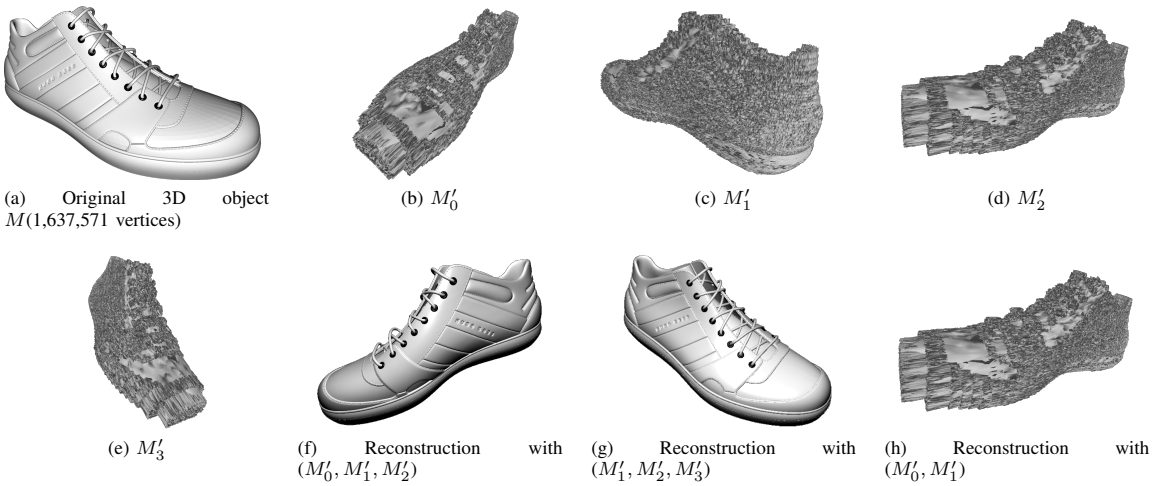
As shown in Fig. 9, our method recovers a shared 3D object M'_i as a host object and substitutes selected bits from the current vertex by those of the reconstructed binary word W. This operation is reiterated for each vertex of the 3D object. Finally, the reconstructed 3D object is identical to the secret one without loss in case of binary serialization.



(a) Original 3D object M(1,637,571 vertices) (b) M'_0 (c) M'_1 (d) M'_2

(e) M'_3 (f) Reconstruction with (M'_0, M'_1, M'_2) (g) Reconstruction with (M'_1, M_2, M'_3) (h) Reconstruction with (M'_1, M'_2)

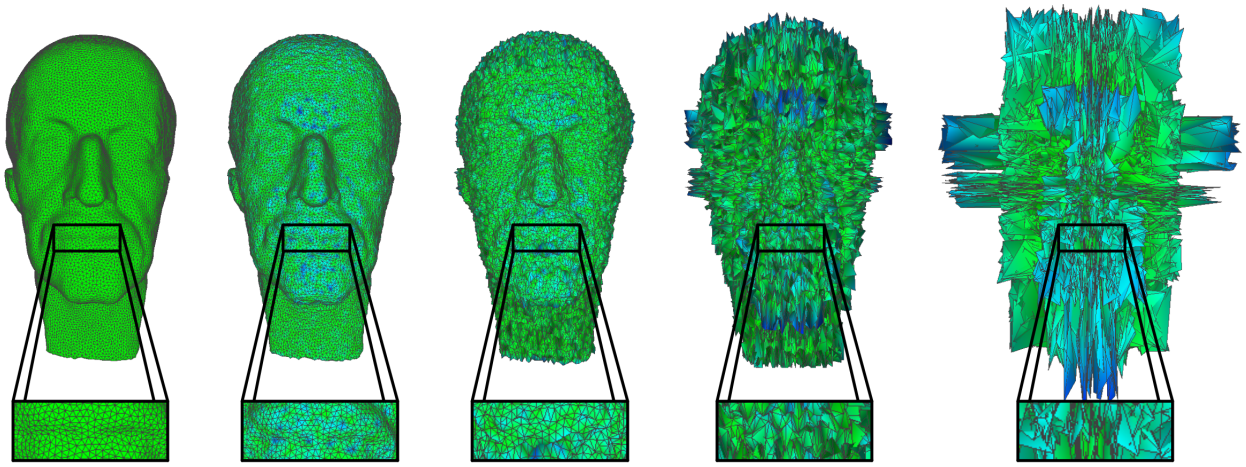
Fig. 10. Sharing of 3D object M with parameters $k = 3$, $n = 4$ and $D =< 18, 19 >$ using Shamir's scheme.



(a) Original 3D object M(1,637,571 vertices) (b) M'_0 (c) M'_1 (d) M'_2

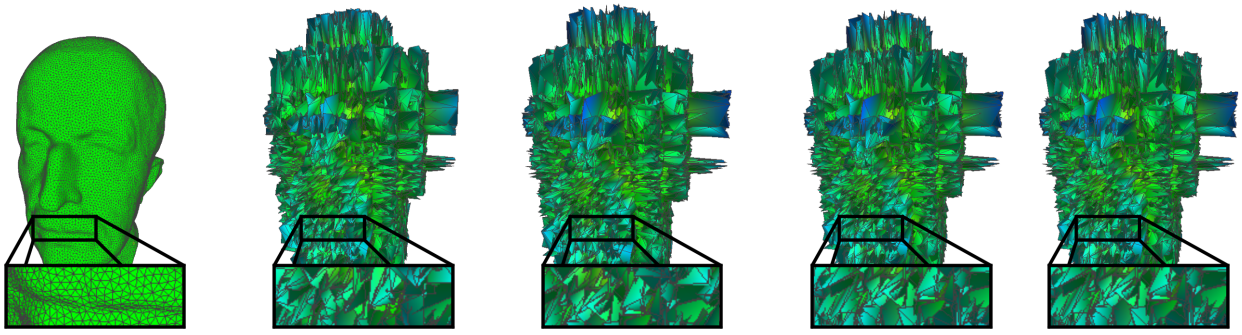
(e) M'_3 (f) Reconstruction with (M'_0, M'_1, M'_2) (g) Reconstruction with (M'_1, M'_2, M'_3) (h) Reconstruction with (M'_0, M'_1)

Fig. 11. Sharing of 3D object M with parameters $k = 3$, $n = 4$ and $D =< 18, 19 >$ using Blakley's scheme.



(a) Original 3D object (b) $D =< 16, 17 >$ (c) $D =< 18, 19 >$ (d) $D =< 20, 21 >$ (e) $D =< 22, 23 >$

Fig. 12. Visual degradation of shared 3D objects as a function of desired degradation level D and distance maps between shared 3D objects and secret 3D objects using RMSE (with Shamir's SS scheme).



(a) Original 3D object (b) $l = 1$ (c) $l = 8$ (d) $l = 15$ (e) $l = 22$

Fig. 13. Visual degradation of shared 3D objects as a function of desired degradation level D with fixed parameter $p = 21$ and varying parameter l and distance maps between shared 3D objects and secret 3D objects using RMSE (with Shamir's SS scheme).

IV. Experimental results

In this section, we show experimental results of our FCSS3DOS method. Firstly, in Section IV-A we present applications of our method with both secret sharing schemes. Then, in Section IV-B we analyze the shared 3D objects with different metrics and study the effects of our degradation level. In Section IV-C, we show the maximum amount of shared 3D objects that can be generated as a function of the degradation level and the chosen SS scheme. Afterwards in Section IV-D, we evaluate the security of our scheme and its robustness against content-recovery attacks. Finally, in Section IV-E we compare our method to the previously proposed methods current state of the art methods.

A. Application

Two experiments are presented with the following parameters: $k = 3$, $n = 4$, $D =< 18, 19 >$ with Shamir's SS scheme and Blakley's SS scheme described in Section II-A.

Fig. 10 and Fig. 11 illustrate the application of our method with a 3D object of a shoe[1], noted M, with 1.6 million vertices and 3 million triangles (Fig. 10.a). The 3D objects in Fig. 10.b-e are 4 generated shared 3D objects produced by our proposed sharing process. These 3D objects have the same number of vertices as the original 3D object, their geometry is degraded, but they remains still usable for their integration into 3D scene environments. Fig. 10.f and Fig. 10.g show that from any group of at least 3 shared 3D objects, it is possible to exactly reconstruct the original 3D object. Fig. 10.h represents a reconstructed 3D object using only 2 shared 3D objects instead of 3, this example remains as degraded as the used shared 3D objects.

When we compare results using Shamir's scheme with those using Blakley's scheme, illustrated Fig. 11, even if the degradation are not exactly the same, we find that both approaches visually offer similar results.

[1]Provided by STRATEGIES (https://www.romans-cad.com/)

B. Measure of the visual degradation

We compared the shared 3D objects generated by our method with the original 3D object using the following metrics:

- The Hausdorff Distance (HD) [30], which corresponds to the maximum between the greatest distance of a vertex from a 3D object M_a to the surface of a 3D object M_b and its symmetrical one with vertices of M_b and the surface of M_a;
- The Root Mean Square Error (RMSE) which is proposed to compute the mean distance between paired vertices of two 3D objects [31].

Fig. 12 illustrates with color the distances of vertices from the original 3D object M to their corresponding vertices on shared 3D objects generated at different degradation levels to the surface. The more the vertex is blue, the further the vertex is from their corresponding vertex on the original 3D object. We note that the position of the first selected bits in the mantissa greatly alters the geometrical distortions as shown in Fig. 12. Table I resumes the HD and the RMSE results for each experimented degradation level. Both reveals an increase of the distance of sampled vertices of shared 3D objects to the original 3D object (for $D =< 16, 17 > HD = 0.008$ and $RMSE = 0.003$ and for $D =< 22, 23 > HD = 0.614$ and $RMSE = 0.200$).

TABLE I
RESULTS FOR THE HD AND THE RMSE METRICS BETWEEN THE SECRET 3D OBJECT AND THE SHARED 3D OBJECTS WITH DIFFERENT DEGRADATION LEVELS.

D	$< 16, 17 >$	$< 18, 19 >$	$< 20, 21 >$	$< 22, 23 >$
HD	0.00789055	0.0350072	0.126543	0.613757
RMSE	0.00298951	0.0119936	0.048057	0.199993

Fig. 13 shows shared 3D objects with the proposed method using Shamir's SS scheme where the value of parameter p is fixed to 21 and the value of parameter l is set to 1, 8, 15 and 22. We observe that the shared 3D objects are visually similar for the human visual system. The value of parameter l of degradation level has only a limited impact on the visual geometrical distortions.

TABLE II
RESULTS FOR THE THE HD AND THE RMSE METRICS BETWEEN THE SECRET 3D OBJECT AND THE SHARED 3D OBJECTS WITH $p = 21$ AND DIFFERENT VALUES FOR THE DEGRADATION LEVEL PARAMETER l.

D	$< 21, 1 >$	$< 21, 8 >$	$< 21, 15 >$	$< 21, 22 >$
HD	0.262084	0.29119	0.290968	0.290968
RMSE	0.095659	0.09903	0.099027	0.099027

Table II resumes the HD and the RMSE metrics for the experiment presented in Fig. 13. We observe that metric results are close between the shared 3D objects with $l \in \{8, 15, 22\}$ ($HD \simeq 0.291$ and $RMSE \simeq 0.099$). Only the shared 3D object with $l = 1$ has a smaller value for the HD ($HD = 0.262$) and the RMSE ($RMSE = 0.096$) because our proposed scheme shares only 3 bits of coordinates. Therefore, the metric results for $l = 1$ correspond to the minimal geometrical distortion induced in the shared 3D objects for the desired degradation level.

Even if the value of the parameter p is the most important because it controls the geometrical distortions, value of the parameter l also plays a role to improve the safety of our scheme. Indeed, this last parameter controls the number of bits used to store sharing data and by extension the number of bits protected by our proposed method as presented in Section IV-D. We can note that perceptual-based metrics could be used to measure the visual degradation of our proposed approach [32]–[34].

C. Maximum number of shared 3D objects

In this section, we present how many participants can receive a shared 3D object. The maximum number of shared 3D objects n_{max} varies as a function of the chosen SS scheme and the desired degradation level D. For Shamir's SS scheme, n_{max} depends on the number of elements constituting the used Galois field. This one is computed as a function of the desired degradation level D and more precisely on the length l of the selected range:

$$n_{max} = |GF(2^m)| - 1 = |GF(2^{3\times l})| - 1 = 2^{(3\times l)} - 1. \quad (12)$$

For example, if $D =< p, l = 3 >$, then the Galois field $GF(2^{(3\times l)})$ has $2^9 = 512$ elements and the method can generate up to 511 shared 3D objects.

Unlike Shamir's SS scheme, where n_{max} depends on the range l associated to degradation level D and the required number of participants k, Blakley's SS scheme requires storing coefficients set $\{\{a_i | i \in \{0, ..., k-2\}\}, b\}$ to represent a k-dimensional hyperplane. It is then possible to estimate the maximum number of participants:

$$\forall i \in \{0, ..., k-2\}, |a_i| = [\frac{3 \times l}{2 \times (k+i)}], \quad (13)$$

$$n_{max} = \prod_{i=0}^{k-2} C_1^{2^{|a_i|}} = \prod_{i=0}^{k-2} 2^{|a_i|}, \quad (14)$$

where $[x]$ is the nearest integer of x.

For example, if $D =< p, l = 3 >$, then 9 bits of the vertex are used to store the k-dimensional hyperplane associated to the vertex. Since the value of k also defines the dimension of the hyperplane, it determines how selected bits are uniformly distributed for variables x_i, coefficients a_i and b. The size in bits of coefficients a_i can be estimated using Eq. (13).

Table III resumes the size in bits of different coefficients and variables as a function of k for the fixed degradation level

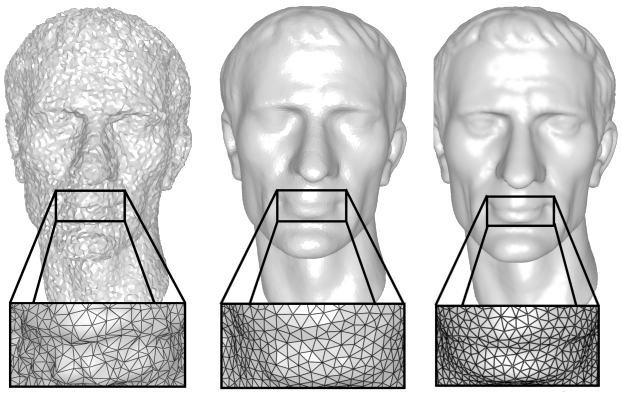
TABLE III
MAXIMUM AMOUNT OF SHARED 3D OBJECTS USING BLAKLEY'S SS SCHEME FOR DEGRADATION LEVEL $D =< p, l = 3 >$ AS A FUNCTION OF k.

k	2	3	4	5	6
$(\|x_0\|, ..., \|x_{k-1}\|)$	(5, 4)	(3, 3, 3)	(3, 2, 2, 2)	(2, 2, 2, 2, 1)	(2, 2, 2, 1, 1, 1)
$(\|a_0\|, ..., \|a_{k-2}\|)$	(2)	(2, 1)	(1, 1, 1)	(1, 1, 1, 1)	(1, 1, 1, 1, 0)
$\|b\|$	7	6	6	5	5
n_{max}	4	8	8	16	16

$D =< p, l = 3 >$. When the size of a coefficient a_i reaches 0, it means that its value is set to 1. We also remark that coefficient b requires a lot of bits in order to represent the sum result of Eq. (5). Using a larger sliding window over the coordinates of vertices by increasing the value of parameter l, allows more bits to be used for coefficients a_i and b. So, in this case, the proposed scheme can produce more shared 3D objects.

D. Attack sensitivity analysis

The degradation level D directly induces the security of the secret 3D object by defining the number of selected bits in each vertex of the secret 3D object during the sharing process. However, as explained in [35], methods partially encrypting data are sensitive to attacks looking to recover the content rather than to use a secret key. An adversary is able to use clear information to build proper attacks in order to sufficiently reconstruct the content. Since our proposed method selects bits from vertex coordinates, then it preserves a part of the original 3D object vertices. 3D object processing techniques, such as smoothing methods [36], [37] or even reconstruction approaches like the marching cubes algorithm [38], can help to slightly improve the quality of a shared 3D object. Nonetheless, even if a smoothed 3D object reveals more of the secret content, the high quality features of the 3D object are still protected. As illustrated in Fig. 14, the preservation of some properties of the 3D object can allow adversaries to design attacks to reconstruct the most significant encrypted bits of the 3D object, but the high quality 3D object cannot be reconstructed. Indeed, with $D =< 18, 1 >$, $HD = 0.019$ and $RMSE = 0.004$.

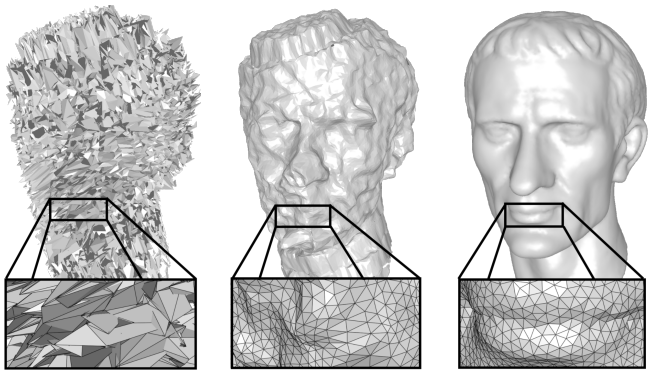


(a) Shared 3D object (b) Smoothed shared 3D object (c) Original 3D object

Fig. 14. Laplacian smoothing attack on a shared 3D object generated with degradation level $D =< 18, 1 >$.

Furthermore, as shown in Fig. 15 a higher value of degradation level D is able to produce a more secure shared 3D object against 3D object processing attacks. So, in this case (with $D =< 21, 1 >$) the low and high quality content of the secret 3D object, in addition to the high quality version, is equally protected with $HD = 0.191$ and $RMSE = 0.049$.

Table IV resumes the metric results of the HD and the RMSE for the shared 3D object and the smoothed one illustrated in Fig. 14 and Fig. 15. We can observe that the smoothed



(a) Shared 3D object (b) Smoothed shared 3D object (c) Original 3D object

Fig. 15. Laplacian smoothing attack on a shared 3D object generated with degradation level $D =< 21, 1 >$.

TABLE IV
RESULTS FOR THE HD AND THE RMSE METRICS BETWEEN THE SHARED 3D OBJECTS AND THE SMOOTHED SHARED 3D OBJECTS WITH THE SECRET 3D OBJECT OF FIG. 14 AND FIG. 15.

Degradation level	Metric	Shared 3D object	Smoothed 3D object
$D =< 18, 1 >$	*HD*	0.0316368	0.018657
	RMSE	0.0116293	0.003460
$D =< 21, 1 >$	*HD*	0.262084	0.191365
	RMSE	0.095659	0.049094

3D object has lower metric results than the shared 3D object of Fig. 14. Meanwhile for the smoothed shared 3D object when the degradation level was $D =< 21, 1 >$, we remark that the metric results are lower than the ones for the shared 3D object, but not as low it was when $D =< 18, 1 >$. The geometrical distortions are more efficient with this degradation level, this prevents the recovery of a much higher quality 3D object.

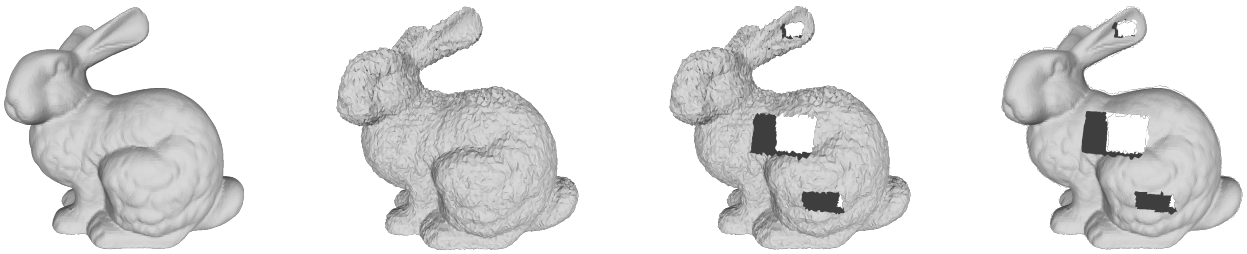
An adversary can also employ a brute force attack by trying to reconstruct the secret 3D object. Naively, because such an attack requires the right set of bits for each vertex of the 3D object which corresponds to finding the right combination among: $2^{3\times l\times |V|}$. However, an adversary can decide to only look for the most significant selected bits of each coordinate for all vertices in order to reduce the degradation level of the shared 3D object. Nonetheless, in order to establish if the modifications brought to the most significant bits are relevant, an adversary needs to find a way to evaluate if the modifications reveal the secret content. In examples illustrated in Fig. 14 and Fig. 15, we use the HD and the RMSE to evaluate it, but these metrics are with full reference, which means we need the original 3D object to make a comparison, unlike an adversary who does not have access to the original 3D object. Depending on the degradation level D applied during the sharing step, an adversary could only compare their results with a smoothed version of the shared 3D object which, as previously stated, has no high quality features.

E. Comparison with previous methods

In this section, we compare our proposed method with recent state-of-the-art S3DOS schemes: methods by Elsheh and Hamza [10], Anbarasi and Mala [11], Tsai [13] and Lee *et al.* [14]. Table V presents several properties we want to

TABLE V
COMPARISON OF OUR PROPOSED METHOD WITH PREVIOUS ONE OF 3D OBJECT SHARING SCHEMES [10], [11], [13], [14].

Properties	Elsheh and Hamza [10]	Anbarasi and Mala [11]	Tsai [13]	Lee *et al.* [14]	Our scheme
Secret Sharing Scheme	(a) Blakley (b) Thien & Lin	Shamir	Shamir	Reed-Solomon Codes	(a) Shamir (b) Blakley
Shared data	Geometry/Connectivity	Geometry/Connectivity	Geometry	Decimated 3D objects	Geometry
Shared data compression (Before sharing)	Lossless (Huffman coding + ZLIB)	Lossless (Huffman coding + ZLIB)	Space subdivision	Decimation, EdgeBreaker, LZMA	None
Threshold k	3	$\{2, ..., n\}$	$\{2, ..., n\}$	$\{2, ..., n\}$	(a) $\{2, ..., n\}$ (b) f(n, D)
Threshold n	P : prime	P : prime	255	Number of 3D objects in host group	(a) $2^{3\times l}-1$ (b) $\prod_{j=0}^{k-2} 2^{\lvert a_j \rvert}$
Multiple	No	Yes	No	Yes	No
Meaningful	No	No	Yes	Yes	**Yes**
Format-Compliant	No	No	Yes	Yes	**Yes**
Selective	No	No	No	No	**Yes**
Size of $shares$	a) Same b) Smaller ($\frac{1}{k}$)	Same	Bigger	Bigger	**Same**
Output	n binary files	n binary files	n steganographied 3D objects from host group	n steganographied 3D objects from host group	n geometrically-distorted 3D objects as a function of degradation level D



(a) Original 3D object (b) Shared 3D object (c) Shared 3D object with holes (d) Reconstructed 3D object

Fig. 16. Cropping attack over vertices (Same vertices were removed in each shared 3D objects).

compare. Concerning **shared data**, our proposed scheme, like the method of [13], shares geometry by protecting all vertices since we want to protect relevant geometrical information of the secret 3D object. Moreover since our shared 3D objects contain exactly the same connectivity as the secret 3D object, we are then able to reconstruct the secret 3D object without sharing connectivity like in [10], [11], [14]. The property **multiple** defines the ability to share simultaneously multiple secret 3D objects, only [11], [14] propose this property. The property **meaningful** means that shares are not just considered as random noise data like in [10], [11]. Our proposed method generates shared 3D objects representing the secret 3D object in low quality version as a function of the desired degradation level D, meanwhile [13], [14] embed sharing data in host 3D objects. The property **format-compliant** is attributed to schemes returning the same type of file in output as it was presented in input. Our method has this property, like in [13], [14], by generating shared 3D objects. This way, the share of a participant can still be visualized as a 3D object. Concerning the **selective** property, our proposed method is the first S3DOS scheme which proposes to generate distorted shared 3D objects with a desired degradation level D. Using this property, participants can receive a low quality preview of the secret 3D object. With our approach it is possible to encrypt a secret 3D object in a transparent, sufficient or confidential way. For the **size of shares**, unlike [13], [14], since we choose to generate geometrically distorted shared 3D objects, our method preserves the number of vertices and the size of the original secret 3D object. While other schemes apply sampling methods to correctly use the finite field they have selected [10], [11] or to embed their host 3D objects inside [13], [14], we choose to apply the approach of Yang *et al.* [9] to keep the size of the shared 3D objects equal to the secret 3D object. Note that in addition to having the same size as the secret 3D object, these shared 3D objects all have the same level of degradation. Consequently, since vertices are shared independently, if the shared 3D objects have the same vertices cropped from their geometry, our proposed scheme can still reconstruct the remaining secret content as shown in Fig. 16. In conclusion, we can note that our method is the only one that is meaningful, format-compliant, selective and where the share 3D objects preserve the original size of the secret 3D object.

V. CONCLUSION

In this paper, a FCSS3DOS scheme is proposed. By selecting, sharing and substituting bits of the secret 3D object, our method protects and shares 3D objects. To do this, our

method distributes to n participants, the shared 3D objects in which bits of vertices of the secret 3D object are selected and substituted by those generated by the chosen SS scheme. Vertices are shared independently and then the shared 3D objects keep the same size as the original 3D object. The shared 3D objects generated by the sharing process can be visualized inside 3D scenes to allow collaborative work. As a function of the desired degradation level, all shared 3D objects have the same level of geometrical distortions induced in them. Furthermore, the degradation level can increase the number of shared 3D objects which can be generated and the computational complexity according to the used SS scheme (Shamir or Blakley) to share selected bits of all vertices from the 3D object. We presented experimental results and analyzed our method. The proposed algorithm was motivated by the intention of letting users hold shared 3D objects which can be visualized. It allows content producers to select the degradation level according to their needs in terms of visual confidentiality for their shared 3D objects. This way, they can choose values which bring a total visual confidentiality or a transparent encryption passing through sufficient encryption for their shared 3D objects. Several applications, including in simulation, video games, animation and special effects are interested in this type of approach to manipulate objects while preserving copyright. We also compared our method with current state of the art S3DOS schemes and show the advantages of our methods in terms of efficiency. Through the use of Shamir's or Blakley's SS schemes, our proposed method opens up a new way to integrate new properties and features according to the improvements brought to both of them during the last few decades from SS and SIS work.

Future work will concentrate on new functionalities for FCSS3DOS schemes. But also, with the increased usage of 3D protection techniques by 3D selective encryption or S3DOS, a new issue is rising about the evaluation of protected 3D objects. Objective metrics can help to evaluate how much a protected 3D object is different from the secret one from a statistical point of view. However, this does not allow us to determine the thresholds from which our method starts to protect the shape and the content of the secret 3D object. We will work on the perceptual evaluation of 3D objects correlated to the human visual system in order to determine the human threshold of 3D object quality for 3D protection purposes.

REFERENCES

[1] S. Çiftçi, A. O. Akyüz, and T. Ebrahimi, "A reliable and reversible image privacy protection based on false colors," *IEEE Transactions on Multimedia*, vol. 20, no. 1, pp. 68–81, Jan 2018.

[2] Z. Shahid and W. Puech, "Visual protection of hevc video by selective encryption of cabac binstrings," *IEEE Transactions on Multimedia*, vol. 16, no. 1, pp. 24–36, Jan 2014.

[3] Y. Zheng, X. Yuan, X. Wang, J. Jiang, C. Wang, and X. Gui, "Toward encrypted cloud media center with secure deduplication," *IEEE Transactions on Multimedia*, vol. 19, no. 2, pp. 251–265, Feb 2017.

[4] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[5] G. R. Blakley, "Safeguarding cryptographic keys," *Proceedings of the National Computer Conference*, vol. 48, pp. 313–317, 1979.

[6] C. Thien and J. Lin, "Secret image sharing," *Computers & Graphics*, vol. 26, no. 5, pp. 765–770, 2002.

[7] ——, "An image-sharing method with user-friendly shadow images," *IEEE Transactions on circuits and systems for video technology*, vol. 13, no. 12, pp. 1161–1169, 2003.

[8] C. Lin and W. Tsai, "Secret image sharing with steganography and authentication," *Journal of Systems and software*, vol. 73, no. 3, pp. 405–414, 2004.

[9] C.-N. Yang, T.-S. Chen, K. H. Yu, and C.-C. Wang, "Improvements of image sharing with steganography and authentication," *Journal of Systems and software*, vol. 80, no. 7, pp. 1070–1076, 2007.

[10] E. Elsheh and A. B. Hamza, "Secret sharing approaches for 3d object encryption," *Expert Systems with Applications*, vol. 38, no. 11, pp. 13 906–13 911, 2011.

[11] L. J. Anbarasi and G. A. Mala, "Verifiable multi secret sharing scheme for 3d models." *International Arab Journal of Information Technology*, vol. 12, no. 6, pp. 708–713, 2015.

[12] A. Martín del Rey, "A multi-secret sharing scheme for 3d solid objects," *Expert Systems with Applications*, vol. 42, no. 4, pp. 2114 – 2120, 2015.

[13] Y.-Y. Tsai, "A secret 3d model sharing scheme with reversible data hiding based on space subdivision," *3D Research*, vol. 7, no. 1, p. 1, 2016.

[14] S.-S. Lee, Y.-J. Huang, and J.-C. Lin, "Protection of 3d models using cross recovery," *Multimedia Tools and Applications*, vol. 76, no. 1, pp. 243–264, 2017.

[15] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, 1952.

[16] P. Deutsch and J.-L. Gailly, "Zlib compressed data format specification version 3.3," Tech. Rep., 1996.

[17] R.-Z. Wang and C.-H. Su, "Secret image sharing with smaller shadow images," *Pattern Recognition Letters*, vol. 27, no. 6, pp. 551–555, 2006.

[18] C.-C. Yang, T.-Y. Chang, and M.-S. Hwang, "A (t,n) multi-secret sharing scheme," vol. 151, no. 2, pp. 483–490.

[19] J. Zhao, J. Zhang, and R. Zhao, "A practical verifiable multi-secret sharing scheme," vol. 29, no. 1, pp. 138–141, 2007.

[20] A. M. Del Rey, J. P. Mateus, and G. R. Sánchez, "A secret sharing scheme based on cellular automata," *Applied mathematics and computation*, vol. 170, no. 2, pp. 1356–1364, 2005.

[21] J. Rossignac, "Edgebreaker: connectivity compression for triangle meshes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 1, pp. 47–61, 1999.

[22] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *Journal of the society for industrial and applied mathematics*, vol. 8, no. 2, pp. 300–304, 1960.

[23] C.-H. Lin, M.-W. Chao, J.-Y. Chen, C.-W. Yu, and W.-Y. Hsu, "A high-capacity distortion-free information hiding algorithm for 3d polygon models," *International Journal of Innovative Computing, Information & Control*, vol. 9, no. 3, pp. 1321–1335, 2013.

[24] A. Bogomjakov, C. Gotsman, and M. Isenburg, "Distortion-free steganography for polygonal meshes," in *Computer graphics forum*, vol. 27. Wiley Online Library, 2008, pp. 637–642.

[25] M. Gschwandtner and A. Uhl, *Protected Progressive Meshes*. Springer, 2009.

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

[27] K. Wang, G. Lavoue, F. Denis, and A. Baskurt, "A comprehensive survey on three-dimensional mesh watermarking," *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1513–1527, Dec 2008.

[28] R. Jiang, H. Zhou, W. Zhang, and N. Yu, "Reversible data hiding in encrypted three-dimensional mesh models," *IEEE Transactions on Multimedia*, vol. 20, no. 1, pp. 55–67, Jan 2018.

[29] IEEE, "Ieee standard for floating-point arithmetic," *IEEE Std 754-2008*, pp. 1–70, Aug 2008.

[30] 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)*. IEEE, 2002.

[31] P. Cignoni, C. Rocchini, and R. Scopigno, "Metro: Measuring error on simplified surfaces," in *Computer Graphics Forum*, vol. 17, no. 2. Wiley Online Library, 1998, pp. 167–174.

[32] G. Lavoue and M. Corsini, "A comparison of perceptually-based metrics for objective evaluation of geometry processing," *IEEE Transactions on Multimedia*, vol. 12, no. 7, pp. 636–649, Nov 2010.

[33] S.-W. Jeong and J.-Y. Sim, "Saliency detection for 3d surface geometry using semi-regular meshes," *IEEE Transactions on Multimedia*, vol. 19, no. 12, pp. 2692–2705, Dec 2017.

[34] L. Dong, Y. Fang, W. Lin, and H. S. Seah, "Perceptual quality assessment for 3d triangle mesh based on curvature," *IEEE Transactions on Multimedia*, vol. 17, no. 12, pp. 2174–2184, Dec 2015.

[35] A. Said, "Measuring the strength of partial encryption schemes," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, 2005.
[36] L. Herrmann, "Laplacian-isoparametric grid generation scheme," *Journal of the Engineering Mechanics Division*, vol. 102, no. 5, pp. 749–907, 1976.
[37] G. Taubin, "Curve and surface smoothing without shrinkage," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*. IEEE, 1995, pp. 852–857.
[38] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," *SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 163–169, Aug. 1987.

Sébastien Beugnon received his M.D. in Computer Sciences from the Univ. Montpellier, France (2016). Since then, he prepares his Ph.D. Degree in applied computer sciences, through a thesis between the LIRMM Laboratory and STRATEGIES company. His current interests are mainly in the areas of Computer Graphics, Multimedia Security, 3D Data Hiding and 3D Printed Objects.

William Puech received the diploma of Electrical Engineering from the Univ. Montpellier, France (1991) and a Ph.D. Degree in Signal-Image-Speech from the Polytechnic National Institute of Grenoble, France (1997) with research activities in image processing and computer vision. In 1995 he served as a Visiting Research Associate to the University of Thessaloniki, Greece. From 1997 to 2000, he has been an Assistant Professor at the University of Toulon, France, with research interests including methods of active contours applied to medical images sequences. Between 2000 and 2008, he has been an Associate Professor at the Univ. Montpellier, France. Since 2009, he is full Professor in image processing at the Univ. Montpellier, France. His current interests are in the areas of image forensics and security for safe transfer, storage and visualization by combining data hiding, compression and cryptography. He is the head of the ICAR team (Image & Interaction) in the LIRMM, has published more than 40 journal papers and 120 conference papers and is associate editor for 5 journals (JASP, SPIC, SP, JVCIR and IEEE TDSC) in the areas of image forensics and security. He is reviewer for more than 15 journals (IEEE Trans. on Image Processing, IEEE Trans. on Multimedia, IEEE Trans. on Circuits and Systems for Video Technology, IEEE Trans. on Information Forensic and Security, Signal Processing: Image Communication, Multimedia Tools and Applications ...) and for more than 10 conferences (IEEE ICME, IEEE ICIP, IEEE ICASSP, IEEE MMSP, IEEE WIFS, EUSIPCO, ...). Since 2017, he is the general chair of the IEEE Signal Processing French Chapter and since 2018, he is a member of the IEEE Information Forensics and Security TC.

Jean-Pierre Pedeboy is the head of STRATEGIES company since 30 years. He is an engineer and he is very well known in 3D modeling of manufactured objects.