Crypto-compression d'objets 3D
CR 6

Master 2 Informatique, IMAGINE
Université de Montpellier
2021 - 2022
GITHUB:https://github.com/EmeryBV/Crypto-compression_of_3D_objects

Équipe :
- Charles Sayamath
- Emery Bourget-Vecchio

UNIVERSITÉ DE MONTPELLIER

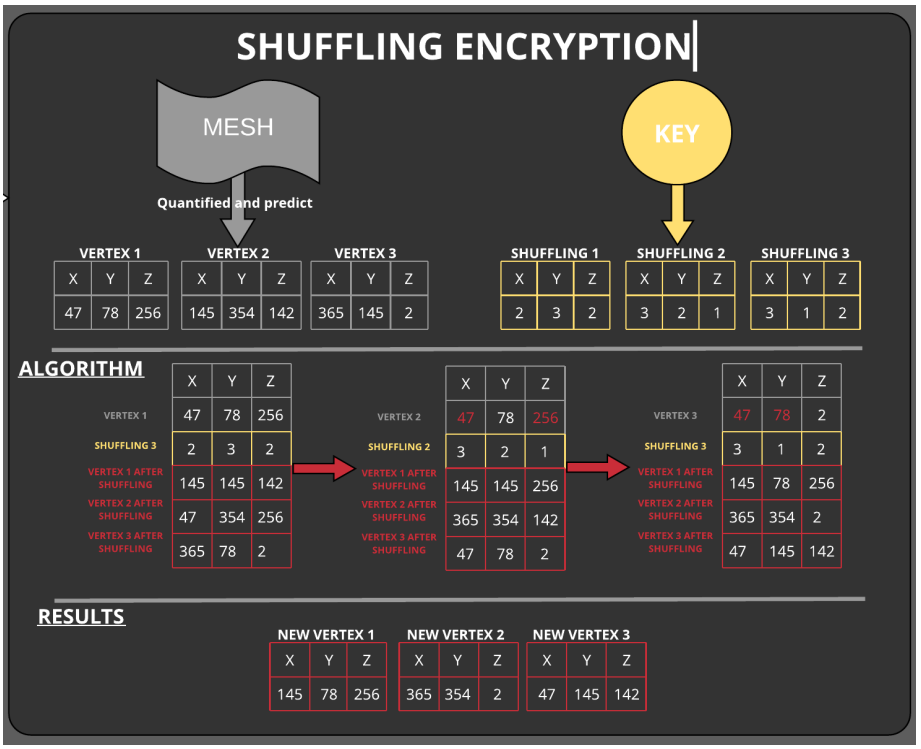
FACULTÉ DES SCIENCES MONTPELLIER

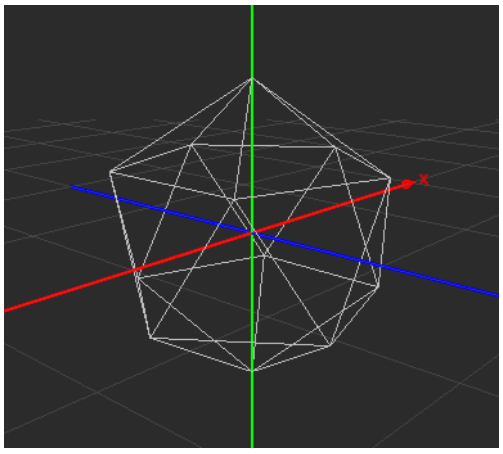
Chiffrement:

Nous avons réussi à implémenter 2 types de chiffrement:

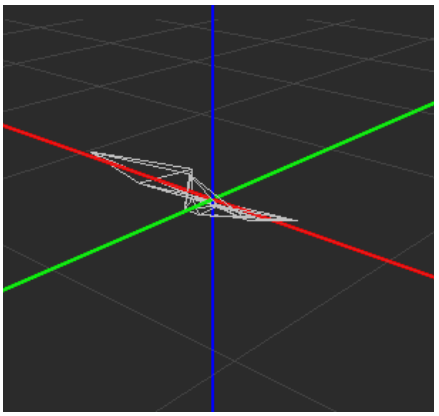
- **Shuffling encryption**

Le chiffrement de type shuffling a pour but d'échanger chacune des coordonnées d'un sommet par les coordonnées d'autres sommets choisis aléatoirement. On obtient ainsi un mélange complet des coordonnées. Le brassage se fait à l'aide d'une clé. Pour retrouver le maillage original, il suffit de réaliser le même procédé dans le sens inverse.
On a appliqué notre algorithme sur les coordonnées prédites quantifiées.





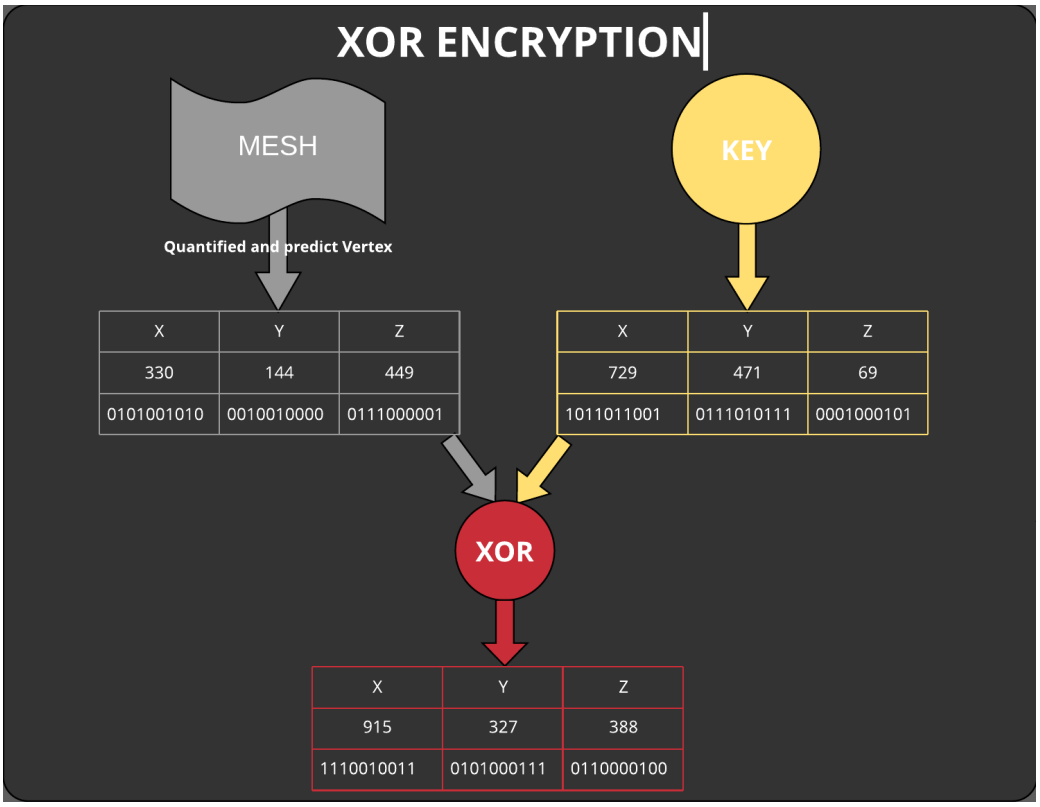
Maillage original

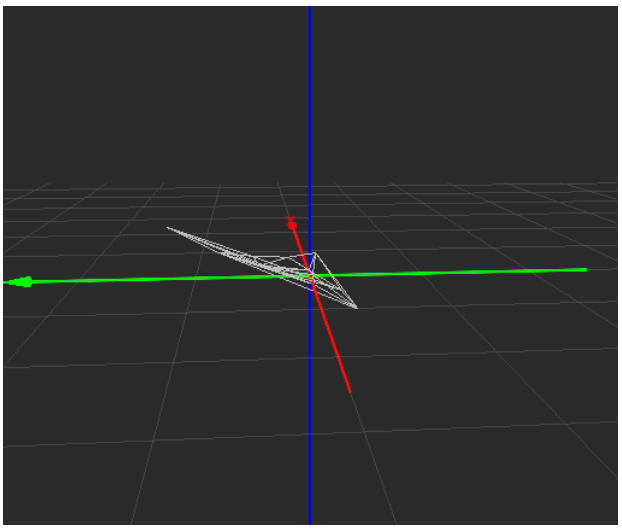


Maillage encrypté avec shuffling

- **XOR encryption**

Le chiffrement XOR sur des maillages 3D repose sur les mêmes principes que le chiffrement XOR sur des images.
On va dans un premier temps générer une clé binaire aléatoire avec comme taille 3 fois le nombre de sommets du maillage (pour avoir X;Y;Z).
Pour chaque coordonné, on va ainsi appliquer un chiffrement XOR.
Le déchiffrage se fait en réappliquant un XOR avec la même clé sur les sommets chiffrés.

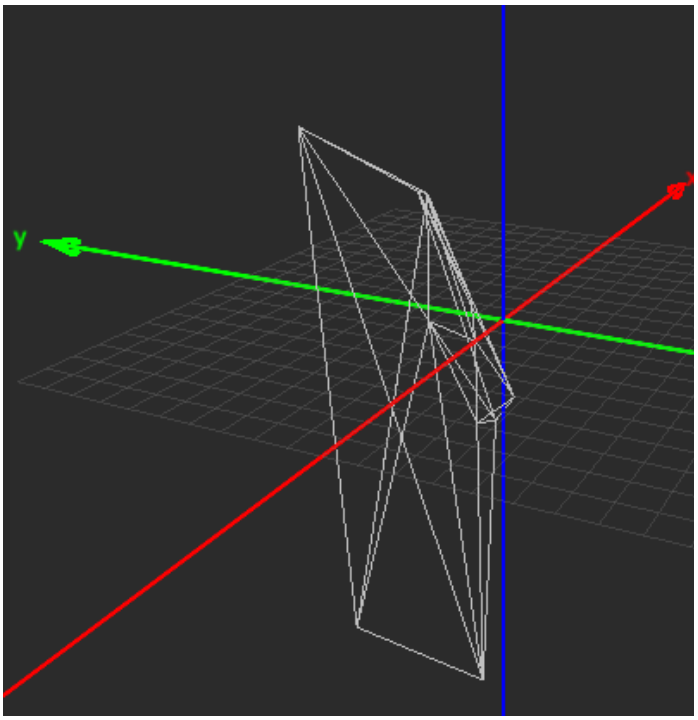




Chiffrement XOR

- **Conclusion sur le chiffrement**

Comme nous l'avons vu, nous appliquons 2 chiffrements différents sur les sommets quantifiés prédits. Cela nous permet d'avoir un très bon niveau de sécurité et d'avoir un maillage complètement méconnaissable de sa version originale.
En appliquant les deux chiffrements, nous obtenons ce résultat:



Compression:

Nous progressons encore mais quelques problèmes persistent ce qui nous empêche de pouvoir appliquer notre algorithme sur des maillages complexes.

Analyse du taux de compression:

Pour étudier la différence entre le maillage original et le maillage reconstruit, nous utilisons la distance de Hausdorff. Elle compare une à une la distance entre les sommets du maillage de base et les sommets du maillage reconstruit. La distance de la paire de sommets avec le plus de différence est retournée.

Soient S et T deux ensembles de points, la distance de Hausdorff est définie par:

$$D_H(S,T) = \max \{f_d(S,T), f_d(T,S)\},$$

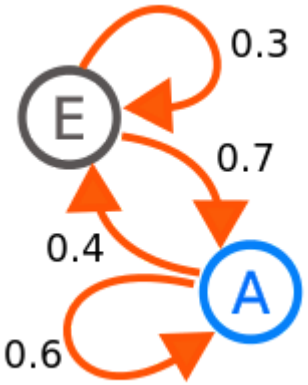
Pour nos maillages les plus simples,avec une quantification de 1024, la distance est de 0.0008088328283425843. C'est un résultat très bon car cela veut dire qu'on n'a presque aucune différence entre nos deux maillages.

Compression entropique:Huffman et chaînes de Markov

Une fois notre maillage compressé avec notre algorithme de “driven-valence”, notre objectif est de supprimer les redondances de notre fichier. En effet, ce dernier va contenir en grande majorité une suite d’instruction “add” avec la plupart du temps le nombre 6 qui va suivre (un sommet étant dans la grande majorité des objets 3D reliés à 6 autres sommets).

Nous appliquons donc dans un premier une compression avec le codage de Huffman, ce qui nous permet de construire un arbre binaire de telle sorte que les feuilles correspondantes aux éléments les plus fréquents seront situées près de la racine, alors que les éléments très peu fréquents nécessitent un parcours complexe depuis la racine pour y accéder.
Par conséquent, les éléments les plus fréquents auront un codage très court.

Le codage de Huffman se combine extrêmement bien avec les chaînes de Markov.
Ce dernier permet à partir d’un symbole d’avoir la probabilité du prochain symbole.



Ces compressions nous permettent de passer de 1.9 kb à 351 octets. On a donc divisé la taille du fichier par plus de 5.