Secret Sharing for Object Structure Data

Wen-Pinn Fang

Yuanpei University, HsinChu, Taiwan

Summary
This paper proposed a secure sharing method for object structure data. Based on the secret sharing scheme, a safe, fault tolerance method for transmission and store object structure data is achieved. Different from traditional transmission method, the method of this paper is not only easy to manage critical data but also have all advantages.

Key words:
Secret Sharing; Polynomial; fault-tolerance; 3d model.

1. Introduction

Secret sharing[1] is a perfect fault-tolerance method to transmit messages via many channels. It makes sure the key of cryptography is safe when the key is transmitted. A well-known method of secret sharing is based on that there are infinite numbers in a curve which generated from a polynomial function. Because secret sharing is an elegant method, there are a lot of researches discuss with it. One of the topics which is relate with secret sharing is secret image sharing . Secret image sharing is proposed by Thein and Lin [2] first. They plug pixel value into the parametric. The safety of their method based on the number of pixel is large enough. There are a lot of discussion following their paper[2-8].

However, there are no paper apply secret sharing engine into object-structure data as author's survey. The object-structure data is adopted in many situations today. For example, 3-d model data is a good case. In this paper, the most important topic is how to share object structure data with fault-tolerance and safety property.

The rest of this paper is organized as the description of secret sharing is in Section II. The proposed method is in Section III, experimental results are in Section IV, and the discussion and future work is in Section V.

2. Secret sharing

Shamir[1] presented a secret sharing method. Secret sharing involves transmitting different shares via different channels. Nobody can see the entire secret message with a single share. The only way to obtain a secret is to collect more shares than a predefined threshold. Thus, secret sharing is "fault-tolerance" because if a channel of share is not available, then any other channel can be adopted instead. Figure 1 shows the algorithm.

Initialization Phase
1. D choose w distinct, non-zero elements of Z_p, denoted $x_i, 1 \le i \le n$ (this is where we require $p \ge n+1$). For $1 \le i \le n$, D gives the value x_i to P_i. The values x_i are public.

Share Distribution
2. Suppose D wants to share a key $K \in Z_p$. D secretly chooses (indepently at random) t-1 elements of Z_p, $a_1, \cdots a_{t-1}$.
3. For $1 \le i \le n$, D computes $y_i = a(x_i)$, where
$$a(x) = K + \sum_{j=1}^{r-1} a_j x^j \mod p.$$
4. For $1 \le i \le n$, D gives the share x_i to P_i

Figure 1. The Shamir (r,n)-threshold scheme in Zp

3. Proposed method

This section presents the method to transmission and store an object-structure data. In section 3.1, a popular object structure data format is introduced. There are two phase of proposed method: share generation and original data recovery. The detail is shown as below:

3.1 smf file format

The most famous object structure data is three dimension geometry model file. For example, simple model format (smf) file format [9]. This format is designed for the description of 3D geometric models. It describes a single object. An SMF file is a text file consisting of a series of lines. Each line is interpreted independently and in sequence. A line may have one of the following three forms:

(1) Entirely whitespace. These lines are completely ignored.

(2) A comment line, beginning with the character '#' . These lines are also ignored.

(3) A command line of the form: <op> <arg>*

Manuscript received November 5, 2010
Manuscript revised November 20, 2010

The first token on the line is interpreted as a command name. The remaining tokens are arguments to the command; their interpretation is command-dependent. Tokens are whitespace-separated character sequences.

The operator definition is shown as table 1.

Table 1. the description of smf

operators	***description***
v <x> <y> <z>	Defines a new vertex with coordinates [<x> <y> <z>]
f <v1> <v2> <v3>	Defines a new triangular face whose corners are the vertices identified by the three numbers <v1> <v2> <v3>.
c <r> <g> <b>	Defines an RGB color.
n <a> <b> <c>	Defines a normal vector.
r <s> <t>	Defines a texture coordinate.
tex <filename>	Specify a file to load the texture from
trans <dx> <dy> <dz>	Defines translate operator
rot [x|y|z] <theta>	Defines rotate operator
scale <sx> <sy> <sz>	Defines scale operator

Below is a simple example to describe a unit cube.

This model is the surface of the unit cube. Each of the six faces of the cube is represented by two triangles.

```
# Vertices on the bottom of the cube (z=0)
v 0 0 0
v 1 0 0
v 0 1 0
v 1 1 0
# Vertices on the top of the cube (z=1)
v 0 0 1
v 1 0 1
v 0 1 1
v 1 1 1
# Triangles on the bottom
f 1 4 2
f 1 3 4
# Triangles on the top
 f 5 6 8
 f 5 8 7
# All the remaining sides
f 1 2 6
f 1 6 5
f 2 4 8
f 2 8 6
f 4 3 7
f 4 7 8
f 3 1 5
f 3 5 7
```

3-2 shares generation phase

When generate shares, there are three parts, (1) split data into face part and other parts (2) generate face-part shares by sharing engine (3) combine face share part and the other part.

<u>(*n*,*r*)-- generate share algotithm</u>

n: number of shares

r: threshold to recover original data

S_j^k: the k[th] share, in sector j

V_i: the vertices of i[th] Face

p: the nearest prime number of max vertice index

Input: object structure data D, share number *n* and threshold *r*

Output: geometry shares $S_1, S_2 \ldots S_n$

Open smf file

While not end of file

  Read smf file line by line

  i$\leftarrow$0

  If prefix is f then

    Store V_i (v_{ix},v_{iy} and v_{iz})

    i$\leftarrow$i+1

  else

    store in string list D'

  end if

end while

For j=1 to i

  k=pseudo-random number by seed k

  Swap v_j and v_k

end for

max=0

For j=1 to i/r

  For k=1 to n

$$S_j^k = \sum_{m=0}^{r} v_m k^m \quad \text{MOD p}$$

```
      if max< Sj^k then
            max= Sj^k
    end if
   end for
 end for
 combine S and D
 For j=i+1 to max
   random generate vj
 end for
```

---- end of algorithm

3-3 recovery phase

In the recovery phase, after user collects enough shares first and then reads the face data, get the original face data by Lagrange Interpolate. Remove redundant vertices. Combine the other part with face part.

3.4 simple example

Let the original data is shown as section 3.1. To make the example easier to read, here skip the permutation step.

The face data of the 3-d model are

{(1, 4, 2), (1, 3, 4),(5, 6, 8),(5, 8 ,7),(1, 2, 6),(1, 6, 5),(2, 4, 8),(2, 8, 6),(4, 3, 7),(4, 7, 8),(3, 1, 5),(3, 5, 7)},

assume that n=3, r=2,

the max id of vertex is 8, the nearest prime number p is equal to 11.

Group faces 4-by-3

{(1, 4, 2), (1, 3, 4),(5, 6, 8)}

{(5, 8 ,7),(1, 2, 6),(1, 6, 5)}

{(2, 4, 8),(2, 8, 6),(4, 3, 7)}

{(4, 7, 8),(3, 1, 5),(3, 5, 7)},

Let the polynomial equation is shown as Equation.1

$$F(x)=v_1+v_2x+v_3x^2 \tag{1}$$

After plug in the vertex indices into the equation (1),

$F_1(1)=1+4+2=7$

$F_1(2)=1+4\times2+4\times2^2 \bmod 11$

$=3$

$F_1(3)= 1+4\times3+4\times3^2 \bmod 11$

$=5$

$F_1(4)= 1+4\times4+4\times4^2 \bmod 11$

$=4$

for the second sector

$F_2(1)=1+3+4=8$

$F_2(2)=1+3\times2+4\times2^2 \bmod 11$

$=1$

$F_2(3)= 1+3\times3+4\times3^2 \bmod 11$

$=2$

$F_2(4)= 1+3\times4+4\times4^2 \bmod 11$

$=0$

for the third sector

$F_3(1)=5+6+8 \bmod 11$

$=7$

$F_3(2)=5+6\times2+8\times2^2 \bmod 11$

$=5$

$F_3(3)= 5+6\times3+8\times3^2 \bmod 11$

$=7$

$F_3(4)= 5+6\times4+8\times4^2 \bmod 11$

$=3$

the first face data of shares are shown as below

Share 1: (7,8,7)

Share 2: (3,1,5)

Share 3: (5,2,7)

Share 4: (4,0,3)

In recover phase, if user collect share 1 ,2 and share 3.

$$\begin{cases} v_1+v_2+v_3=7 \\ v_1+2v_2+4v_3=3 \\ v_1+3v_2+9v_3=5 \end{cases} \tag{2}$$

the first original face will be recover

v_1=1, v_2=4, v_3=2

By the same way, whole original data will be recovered.

4. Experiment result

As show in Fig.2, it is the result of (3, 2) sharing. The shares are noise- like, as Fig 2(a)-(c). If people collect any two shares, the dealer can recover the original geometry mode as show in Fig.2 (d).

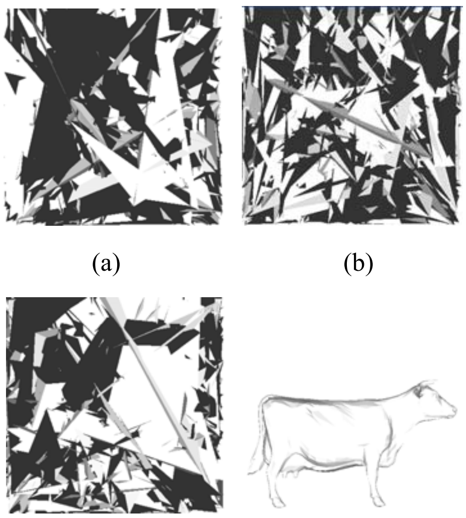

Fig. 2 Experiment result (a)-(c) are shares and (d) is the recovery image.

5. Conclusion

This paper proposed a novel method to transmitted and store object-structure file. A user can not see the original 3-D model without collecting enough shares. All shares are noisy like shapes. However, if he collects enough number of shares in any order, he can recover original 3-D model. The method can be adopted in multimedia application to make sure the 3-D models are stored in secure.

References

[1] Shamir, “How to share a secret,” Communication of the ACM, Vol. 22, no. 11, 1979, pp. 612-613,.

[2] C.C. Thein and J.C. Lin,” Secret Image Sharing”, Computers & Graphics, Vol.26, 2002, pp. 765-770.

[3] W. P. Fang, “Secret Image Sharing Safety”, Proceeding on IEEE International Conference on the 14th Asia-Pacific Conference on Communications (APCC2008), Akihabara, Tokyo, Japan, 2008, 10, 14－2008, 10, 16.

[4] C.C. Thein and J. C. Lin, “An Image-Sharing Method with User-Friendly Shadow Images, “IEEE- Transaction on Circuit and Systems, Video Technology, Vol. 13, no.12, 2003, pp.1161-1169.

[5] W.P. Fang and J.C. Lin, “Universal Share for the Sharing of Multiple Images”, Journal of the Chinese Institute of Engineers , Vol. 30, no. 4, 2007, pp. 753-757.

[6] S.K. Chen and J.C. Lin, “Fault-tolerant and progressive transmission of images,” Pattern Recognition, Vol. 38, 2005, pp. 2466-2471.

[7] W. P. Fang, “Quality Controllable Progressive Secret Image Sharing – Discrete Cosine Transform Approach, ”International Journal of Education and Information Technology, Vol.1, 2007, pp. 43-47.

[8] W.P. Fang and S.J. Lin, “Fast Secret Image Sharing Scheme in HPC,”Proceeding on the 10th International Conference on High-Performance Computing in Asia-Pacific Region(HPC ASIA 2009) joint WorkShop on PC-Grid, Grand Hi-Lai Hotel, Kaohsiung, Taiwan, 2009, 3, 2－2009, 3, 5.

[9] http://www.csit.fsu.edu/~burkardt/data/smf/smf.txt

Wen-Pinn Fang received his BS degree in mechanical engineering in 1994 from National Sun-Yet-Sen University and his MS degree in mechanical engineering in 1998 from National Chiao Tung University, where he get his PhD degree in Computer Science in 2006 from National Chiao Tung University. His recent research interests include image sharing, pattern recognition,, image processing and e-learning.