Watermarking on 3D Model in Spatial and Frequency Domains

Chia-Yen Chen[1], Chi-Fa Chen[2], and Chan-Yuan Hu[2]

[1] Department of Computer Science, The University of Auckland, Auckland, New Zealand
[2] Department of Electrical Engineering, I-Shou University , Kaohsiung, Taiwan
yen@cs.auckland.ac.nz , cfchen@isu.edu.tw

ABSTRACT

Methods of embedding watermarks into 3D models are proposed in this paper. The 3D model is transformed into 2D matrix by first transforming 3D rectangular coordinates of the model into cylindrical coordinates with constant interval in the Z axis and quantizing the radial angles with required angular change. 2D watermarks are respectively embedded into the resulting 2D representation of 3D model in spatial domain and frequency domain. Approaches using JPEG-like compression and reduction in 3D points are adopted as attack processes in testing the robustness of the embedded watermarks. The results show that the watermarks are more robust against attacks from the same domain.

1: INTRODUCTION

Digital contents are becoming increasingly popular with the progresses in multimedia processing, information and network technologies. The right protection of digital contents becomes an important issue with regards to the protection of intellectual properties. Watermarking is an effective approach to the copyright protection of digital contents. Previous researches have shown a wide variety of methods for embedding watermarks into 2D images [1,2,3,4,5,6], but little have been done on the watermarking of 3D data, especially for 3D point cloud models [7,8,9], which are popular 3D shape representations [10,11].

This paper presents several methods of embedding watermarks into 3D point cloud models. A 3D model is generally represented by a number of points having 3D coordinates in, for example, the Cartesian coordinate system. The coordinates of the original 3D data may sparsely span a wide range of space with no regularities between the points. Hence, the 3D data are not typical digital data in the sense that data points are quantized in values of the coordinates and the amplitudes. Computations of such irregular 3D models may be costly. Therefore, digitization of the data is required to facilitate processing of the 3D data. As a continuation of our previous works, we take the inherent advantage of the constant interval in the depth dimension (z) in the 3D data [12]. The digitized 3D model is obtained by first transforming the model's 3D rectangular coordinates into cylindrical coordinates, followed by quantizing the radial angles with the required angular resolution. The digitized 3D model can be represented by a 2D matrix with constant intervals in θ and z coordinates, as discussed in Section 2. In Section 3, we describe the details of embedding 2D watermarks into the respective 2D representations of a 3D model in both the spatial and frequency domains. The extraction of 2D watermarks is essentially the inverse of the embedding process. In Section 4, the robustness of the embedded watermarks is tested, using JPEG-like compression and reduction in 3D points as attack processes. The imperceptibility, robustness, and error rates of embedding the watermarks are also shown and discussed.

2: DIGITIZATION OF 3D MODEL

The proposed algorithms watermark a 3D point cloud model in spatial and frequency domains, respectively, for comparison purposes. Since the data in a 3D point cloud model are generally given in 3D Cartesian coordinates. The x and y coordinates are considered continuous in the sense that they are not limited to a finite set of data. A digitization process is therefore used to quantize the coordinates to desired positions, such that the model's representation becomes regular. Regardless of the algorithm used, coordinates of each voxel in the 3D model need to be digitized for embedding and extraction process later on.

The digitization process starts by transforming rectangular coordinates into cylindrical coordinates with constant z interval. The transformation relationships are shown in Eqns.(1) and (2).

$$r_c = \sqrt{X^2 + Y^2} \quad (1)$$

$$\theta = \tan^{-1}\left(\frac{Y}{X}\right), \quad 0 \le \theta < 2\pi \quad (2)$$

The θ values are quantized with the desired resolution, $2\pi/m$, as the quantization step, where m is the number of points of the 3D model taken along the θ direction. So that the θ coordinate of the j^{th} point is given by Eqn.(3).

$$\theta_j = \frac{2\pi j}{m}, \; j = 0,1,2,\ldots, m-1 \quad (3)$$

The 3D point cloud model is then represented by the matrix ***A***, given in Eqn.(4). Entries in the same row in matrix ***A*** have the same z coordinates, while entries in the same column have the same θ coordinates. For example, the entries on the i^{th} row have the same z_i coordinate, and the entries on the j^{th} column have the same θ_j coordinate. The z and θ intervals between two adjacent points in the digitized 3D model are equal.

$$\boldsymbol{A} = \begin{bmatrix} r_{00} & \cdots & r_{0j} & \cdots & r_{0m} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ r_{i0} & \cdots & r_{ij} & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ r_{n0} & \cdots & \cdots & \cdots & r_{nm} \end{bmatrix} \quad (4)$$

3: WATERMARKING ALGORITHMS

The algorithms for embedding the watermarks are respectively carried out in spatial and frequency domains to compare the effectiveness in different domains. A smaller zone for embedding the watermark in the 3D model is chosen by a sub-sampling process.

3.1: SPATIAL DOMAIN ALGORITHM

In this algorithm, 2D binary images of dimensions 64 pixels by 64 pixels, as shown in Fig. 1, are used as the watermarks. The sizes of the images can be varied as required.

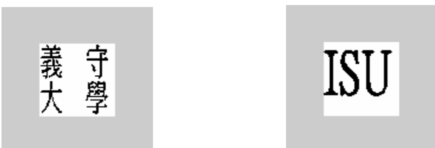


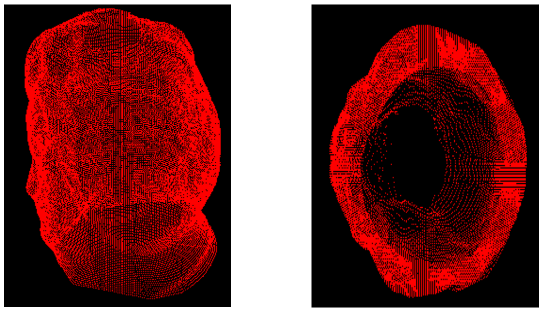
Fig. 1. The 64 by 64 watermarks used in the spatial domain algorithm.

The watermark is added directly to the chosen area in the digitized 3D model representation, i.e. to the sub-matrix ***B*** of the matrix ***A***, to produce the watermark embedded matrix ***A'***, where ***B*** has the same size as the selected watermark. Matrix ***B*** will be saved as a key for the extraction of the watermark. Matrix ***A'*** is then inversely transformed to the 3D point cloud representation in rectangular coordinate system. The 3D Beethoven head model in 3D point cloud format is used in this experiment for demonstration. Figure 2 shows the side view (a) and bottom view (b) of the used model. Figure 3 shows the side view (a) and bottom view (b) of the watermark embedded model. Some translations of the points can be observed by carefully comparing the points in Fig. 3 with those in Fig. 2. However, the differences are not significant.

The total error between points in the 3D models before and after embedding is given by the differences in Euclidean distances between corresponding points as shown in Eqn.(5).

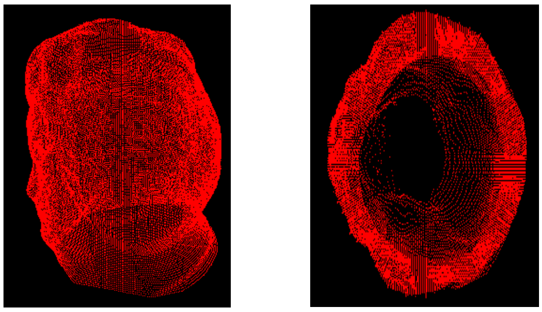
$$E = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(x(i)-x'(i))^2+(y(i)-y'(i))^2} \quad (5)$$

where E is the measured error,
N is the total number of points in the model,
$x(i)$, $y(i)$ are the coordinates before embedding,
and
$x'(i)$, $y'(i)$ are the coordinates after embedding.



(a) (b)

Fig. 2. 3D Beethoven head model.



(a) (b)

Fig. 3. Watermark embedded 3D Beethoven model (spatial domain algorithm).

The overall error of the watermark embedded model show in Fig.3 has the value of 0.0410. The extraction of the watermark is readily carried out using matrix ***B*** as the key in the reverse procedure.

3.2: FREQUENCY DOMAIN ALGORITHM

This algorithm uses DCT to convert the point data in matrix ***A*** into frequency domain. The watermark used in this experiment has been reduced to 32 pixels by 32 pixels to match the limited number of coefficients in the chosen frequency band of the transformed matrix ***A***.

Fig. 4. The 32 by 32 watermarks used in frequency domain algorithm.

The watermark is embedded in the median frequency band of the transformed coefficient matrix, and indices of the embedded positions are saved for the extraction process. Figure 5 shows an example of the indices selected in the frequency domain. The matrix on the left shows the ordering of the DCT coefficients, and the matrix on the right shows the selected indices.

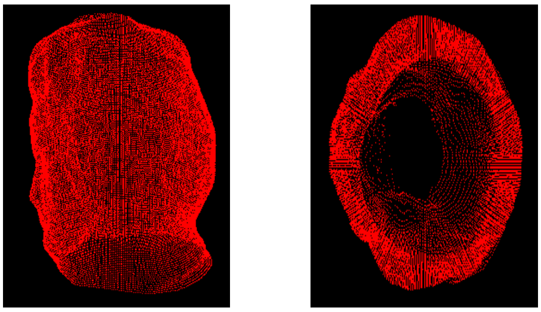
10	11	12	13	**14**	**15**	**16**	17
18	19	20	21	**22**	23	24	25
26	27	28	**29**	**30**	31	32	33
34	35	**36**	**37**	**38**	39	40	41
42	**43**	**44**	**45**	46	47	48	49
50	**51**	**52**	53	54	55	56	57
58	59	60	61	62	63	64	65
66	67	68	69	70	71	72	73

→

14	15	16	22
29	30	36	37
38	42	43	44
45	50	51	52

Fig. 5. The indices of the chosen frequency band.

Figure 6 shows the side view (a) and bottom view (b) of the 3D model with watermark embedded using the frequency domain algorithm. Point translations in Fig. 6 are less noticeable than those in Fig. 3. However, the total error between Euclidean distances of the points has a greater value of 0.1213.



(a) (b)

Fig. 6. Watermark embedded 3D Beethoven model (frequency domain algorithm).

4: ATTACK TO WATERMARKS

DCT compression similar to that which is used in JPEG images and point reduction of the 3D model are used to test the resilience of the watermarks embedded in 3D models. The mean absolute error (MAE) shown in Eqn.(6) is used to judge the difference between the original and extracted watermarks.

$$MAE = \frac{1}{M * N} \sum_{i=1}^{M} \sum_{j=1}^{N} \left| a(i,j) - b(i,j) \right| \qquad (6)$$

where $a(i,j)$ is the pixel value of the original watermark at the coordinate (i,j), and $b(i,j)$ is the pixel value of the extracted watermark at the coordinate (i,j).

There are two kinds of attacks used to test these two watermarking algorithms, giving a total of four different combined cases.

4.1: RESULTS FOR POINT REDUCTION ATTACK

A. Spatial domain algorithm

The total number of points in the original 3D model is 41866. The test reduces the number of points and calculates the MAE for each extracted result. Table 1 shows the results for the two watermarks using spatial domain algorithm. The corresponding extracted watermarks are shown in Figs. 7(a)-(h). The watermarks become invisible when the points are reduced to about half as shown in Figs. 7(d) and (h).

Point number	MAE	
	義守 大學	ISU
35000	0.0386(a)	0.0470(e)
30000	0.0662(b)	0.0789(f)
25000	0.0996(c)	0.1138(g)
20000	0.1375(d)	0.1548(h)

Table 1. MAE of watermark extracted using spatial domain algorithm with point reduction.

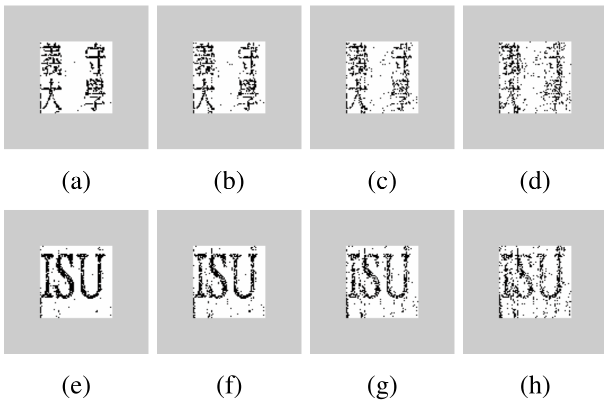


Fig. 7. Watermarks extracted using spatial domain algorithm with point reduction.

B. Frequency domain algorithm

In this experiment, the watermark used is 32 by 32 pixels. Therefore, 32 by 32 entries in the median frequency band of the DCT transformed ***A*** matrix are chosen to embed the 32 by 32 pixel values of the watermark. The number of points is reduced and the MAE is calculated for each extracted result. Table 2 shows the results for the two watermarks using frequency domain algorithm. The corresponding extracted watermarks are shown in Figs. 8(a)-(f). Similarly, the watermarks become invisible when the number of points is reduced to about half as shown in Figs. 8(c) and (f). The extracted watermark shown in

Fig. 8(c) and (f) appear to be worse than that in Figs. 7(d) and (h).

Point number	MAE	
	義守大學	ISU
35000	0.0322(a)	0.0176(d)
30000	0.0752(b)	0.0654(e)
25000	0.1572(c)	0.1338(f)

Table 2. MAE of watermarks extracted using frequency domain algorithm with point reduction.

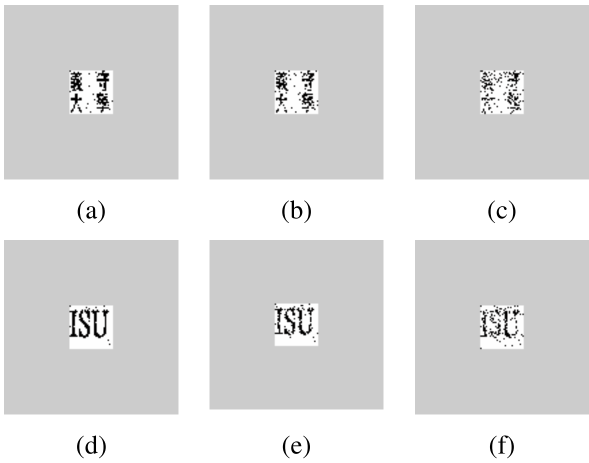


Fig. 8. Watermarks extracted using spatial domain algorithm with point reduction.

4.2: RESULTS FOR COMPRESSION ATTACK

A. Spatial domain algorithm

Different compression rates are applied to the watermark embedded matrix A. The MAE of watermarks in the decompressed 3D models are calculated. Table 3 shows the results for the spatial domain algorithm. The corresponding extracted watermarks are shown in Figs. 9(a)-(h). The watermarks are still visible even when the compression rate is about 26 as shown in Figs. 9(d) and (h).

Compression rate	MAE	
	義守大學	ISU
14.6	0.0872(a)	0.0417(e)
17.5	0.1066(b)	0.0596(f)
21.5	0.1111(c)	0.0696(g)
26	0.1257(d)	0.0947(h)

Table 3. MAE of watermark extracted using spatial domain algorithm with compression.

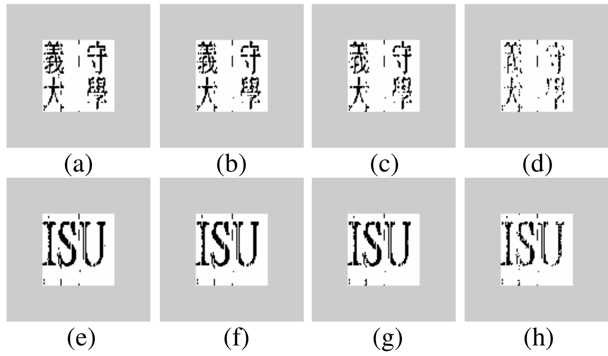


Fig. 9. Watermarks extracted using spatial domain algorithm with compression.

B. Frequency domain algorithm

The different compression rates are applied to the DCT matrix A, with watermarks embedded in frequency domain. The MAE of the extracted watermarks are calculated and shown in Table 4. Corresponding watermarks extracted are shown in Figs. 10(a)-(h). The watermarks are still visible even when the compression rate is about 75 as shown in Figs. 10(d) and (h). The extracted watermarks shown in Figs. 10(d) and (h) appear to be better than that in Figs. 9(d) and (h).

Compression rate	MAE	
	義守大學	ISU
24.8	0.0537(a)	0.0586(e)
40.9	0.0654(b)	0.0664(f)
54.6	0.0791(c)	0.0745(g)
74.5	0.1211(d)	0.1055(h)

Table 4. MAE of watermark extracted using frequency domain algorithm with compression.

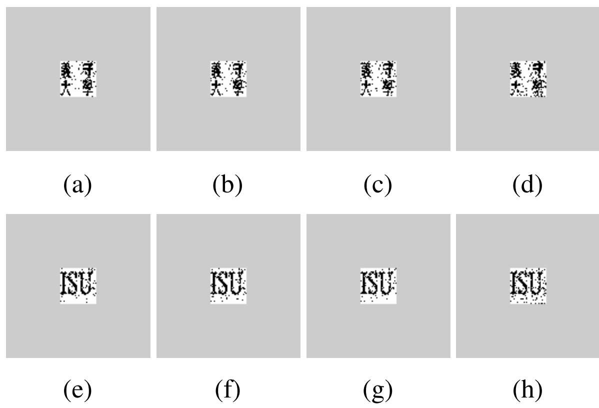


Fig. 10. Watermarks extracted using frequency domain algorithm with compression.

5: CONCLUSION AND DISCUSSION

The paper has proposed several methods of embedding watermark into a 3D point cloud model. Methods in both spatial and frequency domains have been considered for comparison. Reductions of 3D data points and lossy compression have been used attacks to test the resilience of the embedded watermarks. Simulation results have shown that the watermarks are more robust against attacks within the same domain, that is, watermarks embedded in spatial domain have stronger resilience to attack processes carried out in the spatial domain, and watermarks embedded in frequency domain have stronger resilience to attack processes carried out in the frequency domain. In future works, we consider the use of multiple watermarks, orthogonal watermarks, and selection of watermarks subjected to the properties of the 3D model.

Acknowledgments. This research was partly funded by the project ISU95-01-02.

REFERENCES

[1] M. Bami, F. Bartolini, V. Cappellini, A. Piva and F. Salucco, “The based geometric normalization for robust watermarking of digital maps,” International Conference on Image Processing, Vol. 1, pp.1082-1085, 2001.

[2] C. T. Hsu and J. L. Wu, “Hidden digital watermarks in image,” IEEE Transaction on Image Processing, Vol. 8, pp.58-68, Jan. 1999.

[3] Wen-Nung Lie, Guo-Shiang Lin, Chih-Liang Wu, and Ta-Chun Wang , “Robust Image Watermarking On the DCT Domain,” IEEE International Symposium on Circuits and Systems, Geneva, Switzerland, May 28-31, 2000, vol.1, Page(s):228-231.

[4] H.Kii, J. Onishi and S. Ozawa “The digital watermarking method by using both patchwork and DCT,” IEEE Multimedia Computing and System, pp.895-899, Jun 1999.

[5] C.F Wu and W.S. Hsieh, “Digital watermarking using zero tree of DCT,” IEEE Transactions on Consumer Electronics, Vol.46, No.1, pp.8-94, 2000.

[6] Wen-Nung Lie, Guo-Shiang and Ta-Chun Wang, “DIGITAL WATERMARKING FOR OBJECT-BASED COMPRESSED VIDEO,” IEEE International Symposium on Volume 2, 6-9 May 2001, pp.49-52 VOL.2, 2001.

[7] R. Ohbuchi, A. Mukaiyama, S. Takahashi, “Watermarking a 3D shape model defined as a point set,” Proceedings of the 2004 International Conference on Cyberworlds, 18-20 Nov. 2004, Page(s):392 – 399.

[8] A. G. Bors, “Blind watermarking of 3D shapes using localized constraints,” Proceedings of 2nd International Symposium on 3DPVT 2004, 6-9 Sept. 2004, Page(s):242 – 249.

[9] M. Ashourian, R.Enteshari, J. Jeon, “Digital watermarking of three-dimensional polygonal models in the spherical coordinate system,” Proceedings of Computer Graphics International, 2004, Page(s):590 – 593.

[10] Chia-Yen Chen, Reinhard Klette and Chi-Fa Chen, "Shape from Photometric Stereo and Contours," Proc. of CAIP 2003, August 25-27, The Netherlands, 2003.

[11] Chia-Yen Chen, Reinhard Klette and Chi-Fa Chen, "3D Reconstruction Using Shape from Photometric Stereo and Contours," Proc. of IVCNZ 2003, November 26-28, New Zealand, 2003 pp. 251-255.

[12] Chi-Fa Chen and Chia-Yen Chen, “Compression of 3D Point Data Using Discrete Consine Transform,”Proc. of Image and Vision Computing New Zealand 2005, pp.279-284.