3D Print-Scan Resilient Localized Mesh Watermarking

Yanmei Chen[1]
USTC
cym2016@mail.ustc.edu.cn

Zehua Ma[1]
USTC
mzh045@mail.ustc.edu.cn

Hang Zhou
Simon Fraser University
zhouhang2991@gmail.com

Weiming Zhang[1]†
USTC
zhangwm@ustc.edu.cn

2021 IEEE International Workshop on Information Forensics and Security (WIFS) | 978-1-6654-1717-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/WIFS53200.2021.9648379

***Abstract*—Existing 3D print-scan watermarking schemes usually have some limitations, such as the use of auxiliary materials and expensive high-resolution devices, and low visual quality of watermarked models. Considering these limitations, we propose a novel localized mesh watermarking method, which is resilient to 3D print-scan process and suitable for ordinary consumer-level 3D printing and scanning devices. In our scheme, we use the geodesic distances of the model's surface to determine the location and scope of the localized embedded watermark and construct a special tracking signal for the synchronization of the watermark. When detecting the watermark, we amplify the watermark signal through the residual mesh and achieve blind watermark detection. By evaluating various 3D mesh models, we demonstrate that the proposed localized watermarking method can ensure a high watermark extraction accuracy after the 3D print-scan process while maintaining high visual quality.**

***Index Terms*—watermarking, 3D printing, 3D model, geodesic distances, residual mesh**

I. Introduction

Over the past decade, 3D meshes have emerged in industrial, medical, and entertainment applications, being of large practical significance for 3D mesh information hiding, such as steganography and steganalysis [1] and watermarking. Digital watermarking has been widely used as a technical means for copyright protection. With the widespread application of 3D printing technology and the popularization of 3D printers, stronger demands rise for copyright protection of 3D entities.

Existing 3D watermarking techniques can be divided into two parts: 3D digital watermarking and 3D print-scan watermarking. The former is the traditional method to protect 3D mesh models, of which the generation and dissemination are executed in digital media. Many 3D digital watermarking schemes are based on the mathematical statistics of 3D mesh models, such as distributions of distances from vertices on the 3D object surface to the principal axis of the object [2] and the vertex norm [3], etc. Hou *et al.* [4] proposed a circular shift coding structure for the 3D model to preserve a statistical feature of each disk from the layer dividing process. However, these schemes cannot apply to 3D print-scan occasions because the geometric features based on the topology of the mesh will be lost during the 3D print-scan process.

[1]The authors are with CAS Key Laboratory of Electromagnetic Space Information. † Corresponding author.

For watermarking on 3D entities, the most original and direct method is to print or engrave barcodes on the objects, which are also called 3D printing tags. Adobe [5] obtained a patent for a 3D barcode added to objects during the printing process. Harrison *et al.* [6] suggested inserting an acoustic barcode when printing the objects. HP [7] and Rize [8] suggested using invisible ink to print a QR code that is visible under ultraviolet light. LayerCode [9] proposed by Maia *et al.* used two-color printing, UV ink, or layer thickness modification to print one-dimensional barcodes on objects. Delmotte *et al.* [10] proposed a method to embed 2D labels by partially modifying the thickness of the printed layer. These methods often require auxiliary materials and processing and are concerned with the 3D printing process.

Rather than paying attention to the 3D print-scan process, mining the mesh features which are unchanged before and after the 3D print-scan is an effective method to realize watermarking against the 3D print-scan process. Delmotte *et al.* [11] extended the concept of vertex norm to a continuous surface, and used 3D moments to synchronize watermark signals in a 3D printing occasion. However, its main limitation lies in the low tolerance for center position misalignment.

In addition, the characteristics of the 3D printed model can also be used to design watermarking schemes. Hou *et al.* [12] proposed a blind watermarking method, using spread spectrum watermarking embedded in mesh models by layers, so that the generated models will have unique artifacts. The 3D scanned watermarked model is reoriented to find the print axis by analyzing the layering artifacts, which needs a high-resolution 3D scanner to capture and distinguish thin layering artifacts. The requirement for devices may be difficult for consumers with limited budgets in practice. Also, it requires that the content provider who wants to distribute the 3D printed model should print the watermarked model along the predetermined direction for watermark synchronization.

Therefore, we propose the localized mesh watermarking scheme, which selects robust locations in the mesh model to embed the watermark to resist the 3D print-scan process. Our scheme can be realized using ordinary consumer-level FDM printers and does not require high-resolution 3D scanners. Compared to [12], we allow users to freely set the printing direction when printing the models, which is suitable for more occasions and users' needs in practice.

In this paper, we propose the watermarking scheme based on geodesic distances, the invariant against 3D print-scan process. The contributions of this paper are listed as follows:

- We use geodetic distance, a robust invariant for 3D print-scan process, to uniquely identify the model, determine and locate the watermark embedding fields, and locally embed the watermark.
- We put forward the idea of using the residual model to amplify the watermark signal in our watermarking scheme to detect the watermark on the surface of the model after the 3D print-scan process.
- We add a special tracking signal to help us synchronize the watermark and improve its robustness.

The rest of this paper is organized as follows. Sec. II illustrates the motivation of our work. In Sec. III we present the framework of our watermarking scheme which realizes local embedding and watermark synchronization. Results of experiments on kinds of models are elaborated in Sec. IV to demonstrate the effectiveness of the proposed watermarking scheme. Conclusion and future work are given in Sec. V.

II. Motivation

Generally speaking, 3D watermarking protection for 3D entities is vital for people. However, the existing methods still have some limitations. As mentioned before, Hou *et al.* [12] proposed a blind watermarking scheme for 3D printed models, but emphasized the fixed printing axis for models and the importance of high-resolution 3D scanners. However, the printing direction should be able to adapt to users' actual demands. For example, people may rotate the model to find a better angle to save filaments for supports. In this case, Hou's method is not suitable for people since the printing axis in the watermarked model is not aligned with the actual printing direction of 3D printing. Besides, high-resolution 3D scanners are always expensive and less friendly to ordinary consumers.

Considering the problems encountered by existing works, we hope to propose a more flexible and applicable 3D print-scan resilient watermarking scheme, which can print models freely setting the printing direction and successfully detect watermark using consumer-level 3D printers and scanners. Therefore, we try to find the invariants from the model itself that can uniquely characterize the model before and after the 3D print-scan to achieve model calibration and positioning, and then realize blind detection. Geodesic distances, as a fine description of the model's surface, can achieve this function well. Based on these analyses, we proposed a novel framework of watermarking and illustrate it in detail in the next section.

III. Proposed method

A. Notations

Our description of the 3D mesh model is as follows.

A 3D mesh model $\mathcal{M}$ containing V vertices and F faces can be formed as a set $\mathcal{M} = \{\mathcal{V}, \mathcal{F}\}$, where $\mathcal{V}$ is the vertex set $\mathcal{V} = \{v_i\}_{i=1,2,\ldots,V}$ and $\mathcal{F}$ is the face set. The 3D position $\mathbf{p_i}$ is connected to each vertex $v_i \in \mathcal{V}$ [13]:

$$\mathbf{p}_i := \mathbf{p}(v_i) = [x(v_i), y(v_i), z(v_i)]^{\mathrm{T}} \in \mathbb{R}^3. \tag{1}$$

B. Watermark embedding

Fig. 1 shows the workflow of the proposed watermark embedding process. It can be roughly divided into three main modules: watermark signal generation, embedding fields determination, and local watermark embedding. The implementation details are illustrated as follows.

1) Watermark signal generation: Firstly, a bipolar watermark sequence $\mathbf{w}$ with length l_w is generated using a private key. Then, the payload signal $p(x)$ can be represented as:

$$p(x) = \sum_{i=1}^{l_w} w_i \cdot \cos(2\pi x(i + f_s)), \tag{2}$$

where $x \in [0, 1]$ and f_s is minimum frequency band. According to Fourier theorem, the watermark sequence $\mathbf{w}$ can be extracted in the frequency band $[f_s + 1, f_s + l_w]$ of $p(x)$.

To avoid positioning error, a tracking signal $t(x)$ is added into the payload signal $p(x)$ to generated the watermark signal. Note that $p(x)$ is symmetrical, we choose to add a tracking signal in its symmetry axis. Fig. 1(a) shows the watermark signal $f(x)$ added with $t(x) = -\sin(x/10)$ (colored orange) as its tracking signal. Adding a centrosymmetric tracking signal affects little to $f(x)$'s symmetry, but can improve the synchronization accuracy. We will illustrate it in Sec. III-C.

2) Embedding fields determination: In this paper, we take advantage of the invariance of geodesic distances during the 3D print-scan process to implement the watermarking scheme. Inspired by Crane *et al.* [14], we can use the heat source diffusion model to quickly calculate the geodesic distances of the model surface. Firstly, we calculate the centroid of the host 3D mesh and obtain distances between all vertices and the centroid. The farthest vertex is found as the starting point for the geodesic distance algorithm. We can retrieve an approximation of the distance function ϕ from the level set of implicit heat diffusion step μ [14]. In this way, we can quickly get the results, in which ϕ can be approximately regarded as the geodetic distance from the starting point to the rest points. By computing iteratively with a farthest-point sampling, we can get a set of positioning points as shown in Fig. 1(b), which are evenly distributed on the surface of the model.

The proposed watermark is embedded locally in the surrounding fields of these positioning points. Thus, the embedding fields size should be determined. Considering the requirements of partially embedding, there should be no intersection among these fields. As mentioned before, the positioning points are sufficiently scattered. An obvious idea is that we choose the nearest pairs in the positioning points set and set the size as half of the distance. As shown in Fig. 1(c), we obtain embedding fields. For each positioning point, the field's size is adaptive to the surface of the mesh, which can be recovered on the 3D print-scanned mesh.

3) Local watermark embedding: For better robustness, the watermark signal $f(x)$ is embedded repeatedly around each positioning point. As shown in Fig. 1(c), taking the left bunny ear as example, defining vertices in that embedding field as

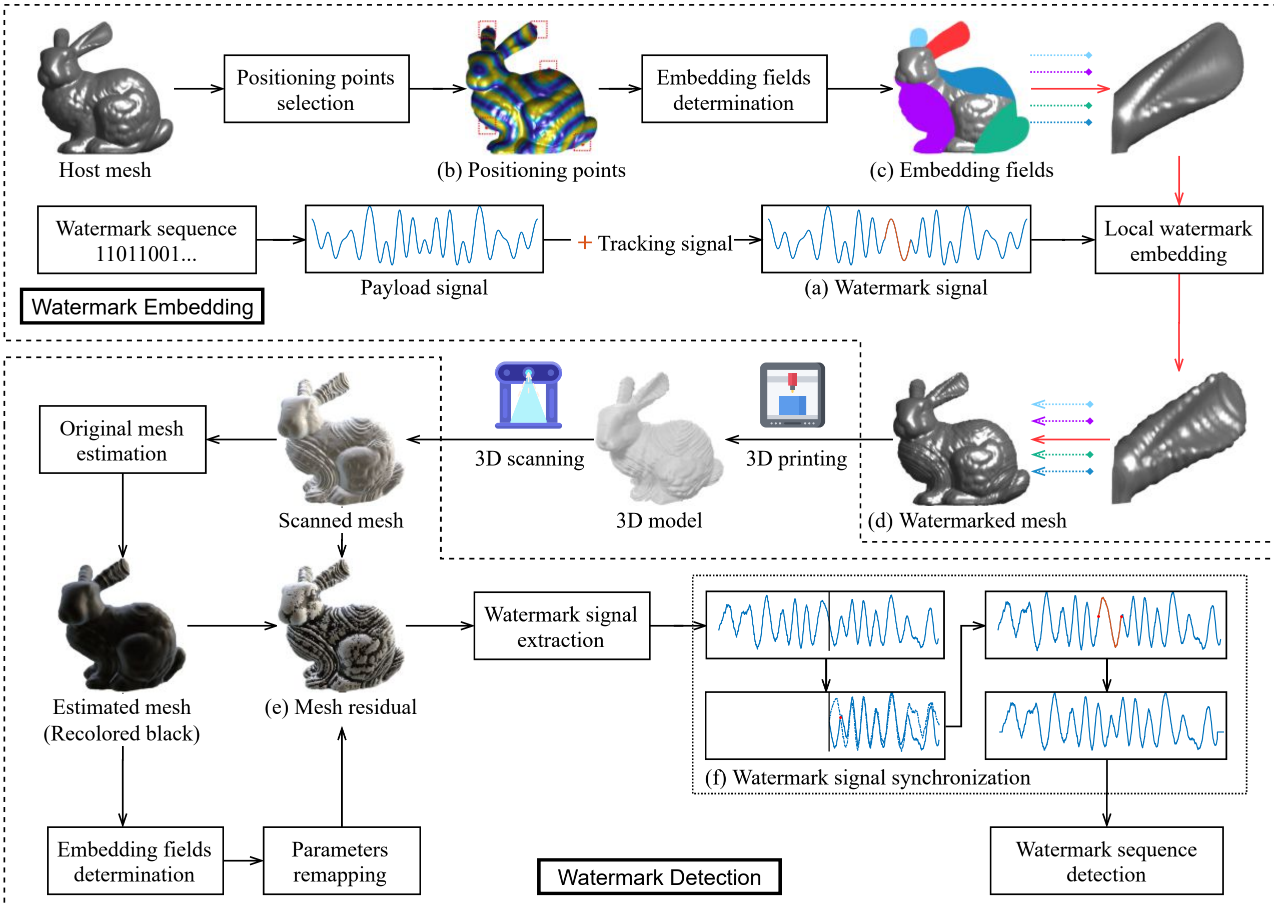


Fig. 1: The workflow of the proposed watermark embedding and detection.

$\mathcal{V}$, $\mathcal{V} = \{v_i\}_{i=1,2,...,N}$, the watermark embedding operation modifies the position $\mathbf{p}_i$ in the normal direction:

$$\mathbf{p}'_i = \mathbf{p}_i + \beta \cdot f(\tilde{\phi}(v_i)) \cdot \mathbf{n}(v_i), \quad (3)$$

where $\mathbf{n}(v_i)$ refers to the normal vector of v_i, $f(\cdot)$ is watermark signal function, and β is the watermarking strength. $\tilde{\phi}(\cdot)$ is the normalized ϕ in $\mathcal{V}$. As shown in Fig. 1(d), for other positioning points, the same embedding processes are implemented to generate the watermarked mesh.

After the embedding in all fields of the positioning points, centroid compensation is executed. By comparing the difference of centroids between the original mesh model and the watermarked mesh model, we slightly modify the non-embedding vertices to ensure the centroid is unchanged.

C. Watermark detection

Fig. 1 shows the proposed watermark detection process, including the watermark signal extraction, synchronization, and watermark sequence detection.

1) Watermark signal extraction: To detect the watermark hidden in a printed 3D model, it is firstly scanned by a consumer-level 3D scanner to get the digital mesh $\mathcal{M}$. To get position parameters as close to the original mesh as possible, we use the smoothed $\mathcal{M}$ as the original mesh's estimation $\mathcal{M}'$. In this paper, we use mesh simplification [15] and Laplacian smoothing [16] to process the scanned mesh models. Similar to Sec. III-B2, for each positioning point, we calculate the embedding field $\mathcal{V}'$, the geodesic distance $\tilde{\phi}'$, and the vertices' normal vectors $\mathbf{n}'$ of $\mathcal{M}'$. By calculating the average parameters of the nearest several points to v_i, we can remapping $\mathcal{V}'$, $\tilde{\phi}'$, and $\mathbf{n}'$ to $\mathcal{M}$. Thus, we obtain the embedding field $\mathcal{V} = \{v_i\}_{i=1,2,...,N}$ in $\mathcal{M}$. And for each v_i, its estimated original position $\mathbf{p}'_i$, original geodesic distance $\phi(v'_i)$, and original normal vector $\mathbf{n}(v'_i)$ are obtained. The remapping operation achieves lower computational cost because most parameters are calculated in the simplified mesh $\mathcal{M}'$. It also concludes more accurate embedding fields because the smoothed mesh $\mathcal{M}'$ is closer to the host mesh.

For each vertex v_i in V, the mesh residual Δ between $\mathcal{M}'$ and $\mathcal{M}$ can be calculated as:

$$\Delta(v_i) = (\mathbf{p}'_i - \mathbf{p}_i) \cdot \mathbf{n}(v'_i). \quad (4)$$

Fig. 1(e) visually emphasizes the mesh residual. By resampling Δ according to ϕ', the residual could be converted to the watermark signal $g(x)$.

2) Watermark signal synchronization: However, considering the distortions occurring in the 3D print-scan process and mesh estimation, we hardly obtain the same embedding fields as the one in the embedding process. Thus, the extracted watermark signal $g(x)$ in the previous subsection is usually out of synchronization. Experiments show that the losses usually occur at the beginning and end of the watermark signal, corresponding to the regions nearest and farthest from the position point. Thus, the extracted watermark signal $g(x)$ is actually cropped and scaled.

To synchronize $g(x)$, we first calculate its symmetrical axis with auto-convolution function proposed in [17], [18]. The first image in Fig. 1(f) shows $g(x)$ and its symmetrical axis (a black vertical line). Noted that we embedded a centrosymmetric tracking signal, by flipping $g(x)$ around its symmetrical axis, the tracking signal would intersect with itself. By calculating that intersection, we could obtain the tracking signal's location and length. Then, according to the predefined length relationship between tracking signal and payload signal, the watermark signal is synchronized and the payload signal is extracted. Fig. 1(f) is an overview of the proposed synchronization process.

3) Watermark sequence detection: For notational simplicity, we reuse $p(x)$ to represent the extracted payload signal. For each $p(x)$ extracted in different positioning points, we transform it to Fourier domain, whose value in $[f_s+1, f_s+l_w]$ is the extracted watermark sequence $\widehat{\mathbf{w}}$. Then, the correlation $corr$ between $\widehat{\mathbf{w}}$ and the embedded one $\mathbf{w}$ is calculated to illustrate the detection results in different positioning points. Additionally, several payload signals with top $corr$ values are accumulated in spatial to get a enhanced payload signal. Following the same steps, that signal's correlation $\bar{corr}$ is calculated as the final detection result.

IV. Experiment

In this section, we demonstrate experimental results to show the proposed watermarking scheme is robust against the 3D print-scan process and discuss the pros and cons of the scheme.

In our experiment, the geometrical distortion of a mesh model including $\mathcal{V}$ vertices after embedding the watermark can be measured by the signal-to-noise ratio (SNR) [19]. We set the SNR between the watermarked mesh model and the original one at 50 ± 0.5 to maintain good visual effects and make the embedded watermark imperceptible.

A. Determining the parameters

To verify the feasibility of our method, we set the watermark parameters as follows during the experiment. The number of positioning points on each mesh model is set as 5, and the surrounding field of each point is embedded with a 16-bit watermark with an adaptive intensity which maintaining the SNR of 3D mesh models at 50 ± 0.5 as mentioned before. The minimum frequency band f_s is set as 3 to separate the watermark signal from the DC signal.

Before printing the watermarked model, it is necessary to slice the model according to the settings of the printer. We use Ultimaker Cura [20] to slice the mesh models, setting the layer height to $0.2mm$ to weigh the printing speed and printed models' resolution, and adding support appropriately. Since removing supports of the model will leave extra dots on the surface, which is also an inevitable distortion in the printing process, we regard it as an attack on our watermarking scheme. The experimental results show that our watermarking scheme is also robust to this kind of distortion.

Fig. 2: Watermarked bunny mesh printed with different materials and colors.

Fig. 3: Models used in experiments.

TABLE I: THE HARDWARES AND FILAMENTS WE USED IN OUR EXPERIMENT.

Devices	Brand	Type
Printer	JR Maker Z-603S	FDM
	Ultimaker S3	FDM
Scanner	Wiiboox-Reeyee	Structured light
Filament	Esum	PLA
	Polymaker	PLA/ABS

In the experiment, we used different models of printers and different filaments (PLA and ABS) in different colors like red, green, and blue to print watermarked models, as shown in Fig. 2 and Table I. The 3D scanner's dot pitch equals $0.2mm$. The obtained experimental data is based on the average result of the combination of these devices.

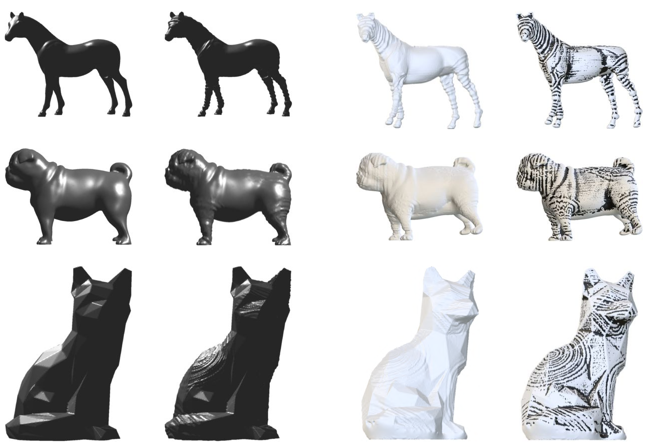
B. Experimental results

We have conducted experiments on different models to verify the feasibility and applicability of our scheme. The watermark detection tests for 3D printed watermarked models were conducted using different devices, filaments, and parameters. Fig. 3 shows physical samples of the 3D printed watermarked models, and Fig. 4 illustrates some models' mesh in embedding and detection process. Table II shows the performance of our watermarking scheme. We calculated the

TABLE II: THE PERFORMANCE ON DIFFERENCE MODELS OF OUR WATERMARKING SCHEME.*

Model	$corr$ in five delimited fields					$corr_m$	$\bar{corr}$
Bunny	0.934	0.570	0.640	0.930	0.699	0.934	0.952
Bulbasaur	0.931	0.887	0.603	0.508	0.859	0.931	0.943
Cat	0.757	0.504	0.541	0.416	0.515	0.757	0.797
Fox	0.451	0.723	0.633	0.506	0.833	0.833	0.920
Hand	0.595	0.532	0.675	0.408	0.468	0.675	0.724
Rabbit	0.569	0.652	0.557	0.554	0.360	0.652	0.776
Charmander	0.810	0.376	0.762	0.669	0.600	0.810	0.902
Totodile	0.493	0.790	0.620	0.522	0.749	0.790	0.894
Pug	0.600	0.752	0.475	0.656	0.533	0.752	0.805
Horse	0.764	0.890	0.829	0.794	0.852	0.890	0.869
Cow	0.466	0.797	0.568	0.695	0.614	0.797	0.901

* The experiment sets the watermark parameters that l_w equals 16 and f_s equals 3, maintaining the SNR at 50 ± 0.5.



(a) (b) (c) (d)

Fig. 4: Mesh examples in the embedding process and detection process. (a) are host meshes. (b) are the watermarked meshes. (c) are print-scan meshes. (d) show the visually enhanced mesh residual.

(a) (b) (c)

Fig. 5: Watermarked bunnies using two schemes under the same set of low-resolution devices. (a) and (c) are embedded using Hou's [12] and (b) is embedded using our method.

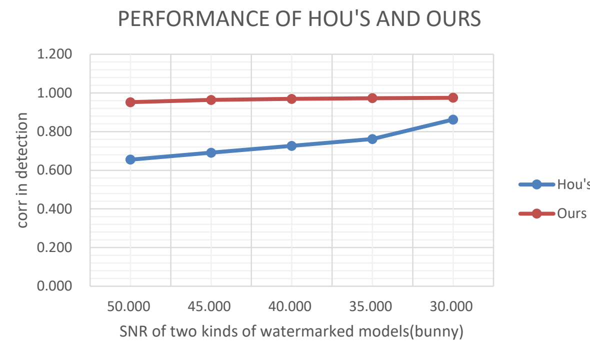


Fig. 6: The detection performance of watermarking scheme in Hou's [12] and ours.

TABLE III: THE DETECTION RESULTS ON DIFFERENT PRINTING DIRECTIONS.

Model	$corr$ in five delimited fields					$corr_m$	$\bar{corr}$
Original	0.934	0.570	0.640	0.930	0.699	0.934	0.952
Rotated	0.965	0.912	0.660	0.922	0.634	0.965	0.954

average of the signals in the spatial domain and transformed it to the frequency domain, and then the correlation $\bar{corr}$ in the average sense is obtained. We set the threshold for detection to 0.8 and most models can be successfully detected watermark.

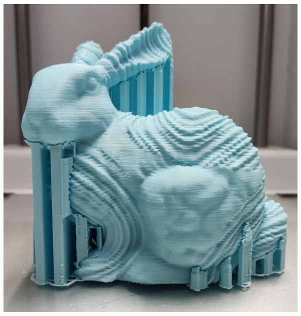
We compare the watermarking performance of Hou's [12] and ours under the same set of low-resolution devices in our experiments. Fig. 5 shows printed bunnies watermarked using our method and Hou's [12]. The middle one (SNR=53.035, $\bar{corr}$=0.952) is embedded using our scheme, the left (SNR=31.075, $\bar{corr}$=0.862 and right (SNR=52.346, $\bar{corr}$=0.655) ones are embedded both using Hou's scheme [12]. The three bunny models are all successfully detected the watermark. It can be seen that when the SNR of the models is close, which is regarded as the objective indicator of the visual quality of the model, the detection performance of our method is better. When the detection performance is similar, the visual quality of the models has a clear difference. That's to say, our scheme has a better detection performance than Hou's when the visual quality stays in an acceptable range as shown in Fig. 6. This verifies the high tolerance of our solution to low-resolution 3D devices.

We have also done the experiment on different printing directions of the model. Fig. 7 shows the same watermarked model printed in different directions. Table III is the detection results of these two models. The experiment results prove the proposed method's robustness under different printing directions. Compared to Hou's method [12], the method proposed in this paper provides more print-setting freedom.

In the whole process, although we do mesh simplification and Laplacian smooth on the 3D scanned mesh models, which are strong attacks in mesh processing, the watermark can still be detected. This proves that our method is robust and suitable for many occasions, and is more tolerant to the performance requirements of 3D printers and scanners. It is obvious that our watermarking scheme performed well on kinds of models.

C. Limitations

From the experimental results, we can see that not all of the five delimited fields of the model can successfully detect



(a) Original setting.

(b) Rotated 30°.

Fig. 7: The same watermarked mesh is printed with different settings.

the watermark signal. We found that the model will not be completely scanned due to the scanning angle and accuracy, which means the loss of information. We try to repair the models before watermark detection by such as closing holes and mesh homogenization. However, these operations still cannot completely recover the impact of lost information, and they are also a kind of attack on the watermarked model to a certain degree. In our experiments, we repeatedly embedded the same watermark signal at multiple fields to guarantee the detectability of the watermark. It means we have increased redundancy and sacrificed watermark capacity to obtain sufficient robustness. We will further our work to solve the problem of information loss in the future. Once different messages can be embedded in multiple positions, the watermark capacity will be significantly increased.

Since our solution needs to consider the performance of low-resolution 3D printing and scanning devices, compared with Hous' work [12], our visual quality is still acceptable whatever from an objective or a subjective perspective. Especially, we can embed the watermark in fewer fields, sacrificing watermark capacity to further improve the visual quality of the model and the imperceptibility of the watermark.

V. Conclusion and Future Work

In our work, we have successfully implemented a robust 3D watermarking scheme against 3D print-scan process based on the geodetic distances, which is robust to 3D print-scan process. By dividing the model and locally embedding, it can detect the watermark against such strong distortion and maintain models' high visual quality. The experiment results verified that the proposed watermarking scheme does work.

Meanwhile, our method has many issues for improvement. For example, the holes in the scanned models need more adaptive methods to process to make up for the loss of the watermark information. The capacity of the watermark can be further enlarged once we improve the process at watermark detection and this is the direction of our future work.

Acknowledgment

This work was supported in part by the Natural Science Foundation of China under Grant 62072421, 62002334, 62121002 and U20B2047, Anhui Science Foundation of China under Grant 2008085QF296, and by Exploration Fund Project of University of Science and Technology of China under Grant YD3480002001.

References

[1] H. Zhou, W. Zhang, K. Chen, W. Li, and N. Yu, "Three-dimensional mesh steganography and steganalysis: A review," *IEEE Transactions on Visualization and Computer Graphics*, 2021.

[2] S. Zafeiriou, A. Tefas, and I. Pitas, "Blind robust watermarking schemes for copyright protection of 3D mesh objects," *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 5, pp. 596–607, 2005.

[3] J.-W. Cho, R. Prost, and H.-Y. Jung, "An oblivious watermarking for 3-d polygonal meshes using distribution of vertex norms," *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 142–155, 2006.

[4] J.-U. Hou, D.-G. Kim, S. Choi, and H.-K. Lee, "3d print-scan resilient watermarking using a histogram-based circular shift coding structure," in *Proceedings of the 3rd ACM workshop on information hiding and multimedia security*, 2015, pp. 115–121.

[5] A. Kumar, N. P. Goel, and M. Hemani, "Method and apparatus for storing and retrieving data embedded into the surface of a 3D printed object," Jul. 26 2016, uS Patent 9,400,910.

[6] C. Harrison, R. Xiao, and S. Hudson, "Acoustic barcodes: passive, durable and inexpensive notched identification tags," in *Proceedings of the 25th annual ACM symposium on User interface software and technology*, 2012, pp. 563–568.

[7] M.Molitch-Hou, "The future of hp's multi jet fusion 3D printing," [EB/OL], https://www.engineering.com/3DPrinting/3DPrintingArticles/ArticleID/12298/The-Future-of-HPs-Multi-Jet-Fusion-3D-Printing.aspx Accessed May 21, 2019.

[8] B.Yusuf, "Digitally augmented additive manufacturing parts from rize," [EB/OL], https://all3dp.com/digitally-augmented-additive-manufacturing-parts-from-rize Accessed April 24, 2019.

[9] H. T. Maia, D. Li, Y. Yang, and C. Zheng, "Layercode: optical barcodes for 3D printed shapes," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–14, 2019.

[10] A. Delmotte, K. Tanaka, H. Kubo, T. Funatomi, and Y. Mukaigawa, "Blind watermarking for 3-d printed objects by locally modifying layer thickness," *IEEE Transactions on Multimedia*, vol. 22, no. 11, pp. 2780–2791, 2019.

[11] ——, "Blind 3d-printing watermarking using moment alignment and surface norm distribution," *IEEE Transactions on Multimedia*, 2020.

[12] J.-U. Hou, D.-G. Kim, and H.-K. Lee, "Blind 3D mesh watermarking for 3D printed model by analyzing layering artifact," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2712–2725, 2017.

[13] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez, and B. Lévy, *Polygon mesh processing*. CRC press, 2010.

[14] K. Crane, C. Weischedel, and M. Wardetzky, "Geodesics in heat: A new approach to computing distance based on heat flow," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 5, pp. 1–11, 2013.

[15] M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 1997, pp. 209–216.

[16] O. Sorkine, "Laplacian mesh processing," in *Eurographics (State of the Art Reports)*. Citeseer, 2005, pp. 53–70.

[17] H. Fang, W. Zhang, Z. Ma, H. Zhou, S. Sun, H. Cui, and N. Yu, "A camera shooting resilient watermarking scheme for underpainting documents," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 4075–4089, 2020.

[18] Z. Ma, W. Zhang, H. Fang, X. Dong, L. Geng, and N. Yu, "Local geometric distortions resilient watermarking scheme based on symmetry," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2021.

[19] M. F. Deering, "Geometry compression," pp. 13–20, 1995.

[20] "Ultimaker Cura," https://github.com/Ultimaker/Cura/, accessed July 20, 2021.