CONFIDENTIAL EXTENDED ABSTRACT.
DO NOT DISTRIBUTE ANYWHERE.

The 22nd Meeting on Image Recognition and Understanding

Blind watermarking for 3d printed objects by modifying layer thickness

Arnaud Delmotte[1,a)] Kenichiro Tanaka[1,b)] Hiroyuki Kubo[1,c)]
Takuya Funatomi[1,d)] Yasuhiro Mukaigawa[1,e)]

Abstract

We present a new blind watermarking algorithm for 3d printed objects that has application for metadata embedding, robotic grasping, counterfeit prevention and crime investigation. Our method can be used on Fused Deposition Modeling (FDM) 3d printers and works by modifying the printed layer thickness on small patches of the surface. These patches can be applied to multiple regions of the object, making it resistant to cropping attacks, local deformations or printing errors. It only requires a single view to extract the watermark, making it faster than most other methods requiring a full scan of the object. In our experiments, we successfully extracted the watermark from flat surfaces using a common 2d paper scanner.

1. Introduction

3D printing became more and more popular and accessible recently, entry-level printers became cheap enough for consumer budget and available in some public libraries, schools, DIY centers, makerspace. But this technology can also help criminals to commit crimes such as counterfeit production, theft by reproducing the keys from pictures, printing TSA master key, or violent crimes by printing untraceable weapons. These objects are often found on the crime scene but difficult to trace [12]. A practical solution would be to insert automatically a watermark containing the printer serial ID and printing date, which requires cooperation from the public printing services and printer manufacturers similarly to what is done for the 2d paper printers[2]. It would not prevent criminals from printing with their own open-source 3d printer to stay untraceable, but it would make their task harder. Additionally, it could be used to identify the owner when a stolen object is found by the police.

Watermark can also be useful for CAD applications to trace the batch ID or retrieve informations about the object. This context has strong constraints about the deformations to preserve the mechanical properties, and crop resistance is also desired to retrieve informations about a part that broke. For robotic grasping, identification and pose estimation of similar looking objects are challenging problems that can be solved by embedding a watermark in the objects. In this context, it is preferable to be able to extract from one face instead of a full scan.

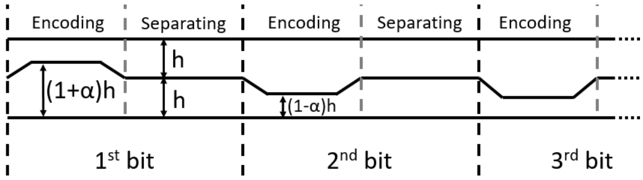

Fig. 1: Encoding layer pattern. It corresponds to 2 layers with variable thickness. We can encode a 1 or 0 bit by respectively increasing or decreasing the thickness of the bottom layer in the encoding region. The top layer thickness will be adjusted such that the sum of the thickness of the 2 layers stay equal to $2h$. The separating region always keep a thickness h.

Developing a good watermark method is challenging because it requires a low surface deformation to be imperceptible to naked eye, but also robustness to printer inaccuracies or other degradations. The previously described contexts also require a large capacity, a fast extraction and the conservation of the mechanical properties. Our method meets these requirements by locally modifying the layer thickness on small patches of the surface, and applying them to multiple regions of the object for redundancy. Even if the object surface is locally degraded or cropped, we can still decode it if some patches are intact. It also provides a low shape distortion and a fast extraction from only one pose instead of a full scan. For flat surfaces, it can be extracted with a standard 2d paper scanner, and does not require any complicated or expensive equipment. Our method works on any Fused Deposition Modeling (FDM) printer that gives access to the motors control. This includes all the printers with 'G-code' support, which is one of the most common file format for 3d printing. We focused on FDM because it is the most used 3d printing technology with a share of 67.7% based on 3dHubs 2018-Q4 trends [1].

2. Related works

Most 2d paper printers include watermark with informations such as printer ID and print time. The most famous

[1] Nara Institute of Science and Technology
a) arnaud.delmotte.zr3@is.naist.jp
b) ktanaka@is.naist.jp
c) hkubo@is.naist.jp
d) funatomi@is.naist.jp
e) mukaigawa@is.naist.jp

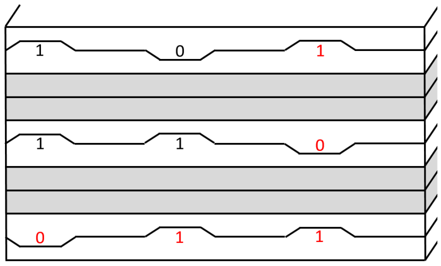

Fig. 2: Example of the watermark pattern Encoding the value "1011" + the parity bits. The digits correspond to the encoded bits, those in red are parity bits. The layers with white background are encoding layers, those in gray are separating layers.

technique consists of inserting a grid of yellow dots[4][6], which are barely visible to naked eye but easily retrievable from a scan or by illuminating with blue light. The principal usage is crime prevention by *e.g.* tracking leaked documents[2]. Our method can be seen as a 3d print version of these methods because it shares a lot of similarities in the properties, usage and applications. Alpha-dot[*1], a UK company, developed microdots containing an ID. These dots are 1mm in diameter and can be glued to valuables. It is officially approved by UK police, which can identify the owner when they find stolen valuables. Our method could be used for the same scenario without needing to buy anything, but our patches are bigger. Hou *et al.* [9] published a blind watermarking method using spread-spectrum watermarking, and analysing the layer artifacts to retrieve the print axis. It can resist to reprint if the print axis is conserved, it is robust to printing errors and has resistance to small crop, but requires a full 3d scan and is low capacity (1 bit). Adobe [10] made a patent about a visible 3d barcode that can be added during the printing. We developed a blind watermarking method [5] that is retrievable from a full 3d scan and resistant to reprint, but has stronger deformations and is not resistant to crop or non uniform deformations. The application scenario is different from this paper, our previous method supposed that we try to extract from a 3d scan made by someone else, whereas this paper supposes that the user will follow our procedure, *i.e.*, the person doing the scan is the person trying to extract the watermark. Li *et al.* [11] developed a method to embed tags below the surface, and retrieve them with a camera, projector and polarizer. Their method has similar usage and applications than ours, it has higher resistance to surface degradation, but put constraints on the inner stucture which may modify the mechanical properties, takes more time to extract, and is limited to gently curved surface. Li *et al.* [12] developed a method that allows to extract a fingerprint from a 3d printer by analysing the layers bonding differences due to the mechanical component tolerances. It works on any printer and does not require to insert a watermark, but it is more difficult to scale to a large number of printers because it requires to print an object and extract the fingerprint for each printer. Additionally, it does not allow to know which user printed the model on a shared printer. Our method requires the printer to include the watermark, similarly to the yellow dots for 2d printers, but can be easily scaled to a large number of printers. It is more suitable for public printer services, while printracker is more suitable to provide evidence after analysing the printer of a suspect. Both methods can be used together to provide stronger evidence.

[*1] `https://www.alpha-dot.co.uk/`

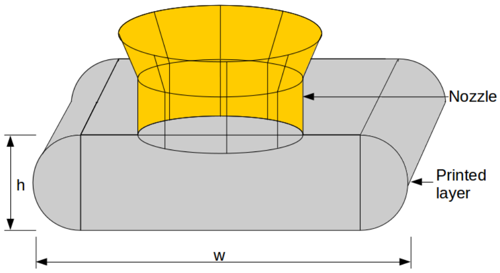

Fig. 3: Cross-section of the nozzle and printed layer

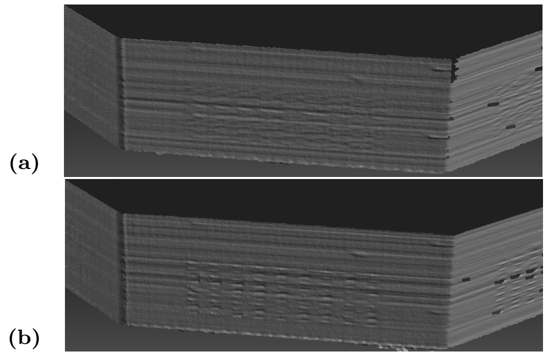

Fig. 4: 64 bits watermarked object scanned by 'HP 3D structured light scanner pro S3' (a) with adjustment of the plasting extrusion to the layer thickness (b) without adjusment

3. Watermark embedding

To embed the watermark, we chose to locally modify the layer thickness because it allows to conserve the global curvature of the surface while embedding a pattern at high frequency. The modifications are done along the tangeant of the surface instead of the normal, and thus cause less deformations to the shape. The layer thickness is a feature of the print that is typically constant and has low noise, which allows us to get a high signal to noise ratio while maintaining a low visibility. Fig. 1 illustrates these layer thickness modifications.

In what follows, we explain the detail of our system. The selection of regions to watermark is explained in sec. 3.1, the pattern is explained in sec. 3.2 and the modification of the printer controls is explained in sec. 3.3

3.1 Watermark region selection

To embed a watermark on the surface of the object, we first generate the list of printer motors commands, *i.e.*, the 'G-code' or similar format, using the printer software and recommanded parameters. Then, we detect the printed traces corresponding to the object surface and search for regions composed of enough layers following a similar path.

The easiest case is when the trace paths are aligned on top of each other or with a constant offset between each layer, but it is possible to support more complex shape if the trace paths are similar enough.

3.2 Watermark pattern

To embed a N bits watermark, we first reshape the signal into a H-by-W matrix, with $H * W = N$, and add a column and row of parity bits for error detection, giving a $(H+1)$-by-$(W+1)$ matrix.

Fig. 1 illustrates the encoding of one row of the matrix by locally modifying the thickness of the 2 encoding layers. The layers are divided into equally sized encoding and separating regions for each bit. In the encoding regions of each bit, we multiply the thickness of the bottom layer by $(1+\alpha)$ or $(1-\alpha)$ to encode a 1 or 0 bit, respectively. The top layer thickness is adjusted to keep the sum of the 2 layers thickness constant.

Fig. 2 illustrates a 2-by-2 watermark, in which the encoding layers are separated by M separating layers. Both separating regions and separating layers are used to simplify the detection and correlation at extraction time, and reduce the deformations by having smoother transitions. The required number of layers is $(H+1)*2$ encoding layers and $H*M$ separating layers. The width of the patern is the width per bit bit_{width} multiplied by $(W+1)$.

In practice, we used $\alpha = 0.4$, $M = 2$ and $bit_{width} = 2.78$ or 5mm. H and W must be even numbers to allow the parity check to detect if all the bits have been inverted, which happens when the pattern is rotated by 180°.

3.3 Printer control

Modifying the layer thickness requires to adjust the extruded plastic volume to keep the layer width constant. High precision models have been developed [13] [14] but are complex to use, we instead use the simplified model proposed by [8][7]. The cross-section of a layer is approximated by a rounded rectangle as illustrated in Fig.3. The area can be calculated by :

$$A = h(w-h) + \pi\frac{h^2}{4} \tag{1}$$

with A, the cross section area, w, the layer width, and h, the layer height. We obtain the volume of plastic by multiplying the cross section area by the length of the layer. Adjusting the plastic extrusion is important to reduce the deformation on the surface as shown in Fig.4. We still have some small deformations because of the approximations in our model and because the plastic flow can not change instantaneously due to the non-linearities in the liquifier as explained in [3].

4. Watermark extraction

The extraction can be done in multiple ways. All we need to do is to segment the different layers and correlate it with the encoding pattern to extract the data. The borders of each layer can be detected thanks to their rounded shape that reflects the light non uniformly.

For flat surfaces, we can simply use a low cost 2d paper scanner to extract the watermark. We need to adjust gamma, brightness and contrast to obtain a picture where the layer separations are visible as shown in Fig. 5a, these parameters will depends on the plastic color and printer model.

After obtaining the scanned image, the image is aligned so that the layers become horizontal using Fourier transform, *i.e.*, detecting the peak magnitude value, computing the corresponding angle, and reorienting the image. This process is similar to the frequency analysis section from [9]. If the layer thickness is known, it is possible to restrict the search range for the peak magnitude and be more resistant to potential errors. Otherwise, the thickness is calculated based on the point with the highest magnitude, and if the printer thickness parameter list is available (*e.g.*, increments of 0.05mm), it can be rounded to the closest value from the list.

Then we extract the highlight lines that separate each layers using 1D Non-Maximum Suppression on the columns of the image, with a neighborhood of half layer thickness. For a 0.2mm layer at 1200 dpi, it gives 0.1mm * 47.244 pixel/mm = 4.72 pixels. Fig. 5b shows the result of the Non-Maximum Suppression applied on Fig. 5a. We use the regularity of the pattern to compute the maximum correlation and find the watermark location.

We finally compute the value of each bit as shown in Fig. 5c, and verify the parity bits. If the majority of parity bits have the wrong value, it means that the image is probably wrongly oriented by 180°and needs to be rotated. Instead of rotating the image and restart the extraction, it is possible to rotate by 180°the obtained watermark matrix, and apply a 'NOT' operator on all the bits.

5. Experiments

We printed a CAD part shown in Fig. 6 with white PLA using an 'Original Prusa i3 MK2S with Multi Material Upgrade V1', with the G-code generated using 'Slic3r Prusa Edition' and modified to embed a watermark in the vertical surfaces that were large enough to contain it.

We did experiments with 16, 32 and 64 bits watermark, corresponding respectively to 5×5, 9×5 and 9×9 watermark matrix including the parity bits. The surface used to encode the watermark was $3.6 \times 25mm$ for 16 bits, and $6.8 \times 25mm$ for 32 and 64 bits, and the bit_{width} was $5mm$ for 16 and 32 bits, and $2.77mm$ for 64 bits. For the scan, we used a 'Canon PIXUS MG3630', controlled it with the opensource software 'XSane' on linux which gives us more control than the default windows software, and we used the following parameters : color: gray, resolution: 1200dpi, gamma: 0.30, brightness: -30.0, contrast: 100.0.

Fig. 7 show the mean number of error we got after extracting the watermark from the 3 faces that we could scan. These results were obtained without applying error correction and without using redundancy. In practice, the parity bits allow to correct 1 error bit per watermark matrix, and is

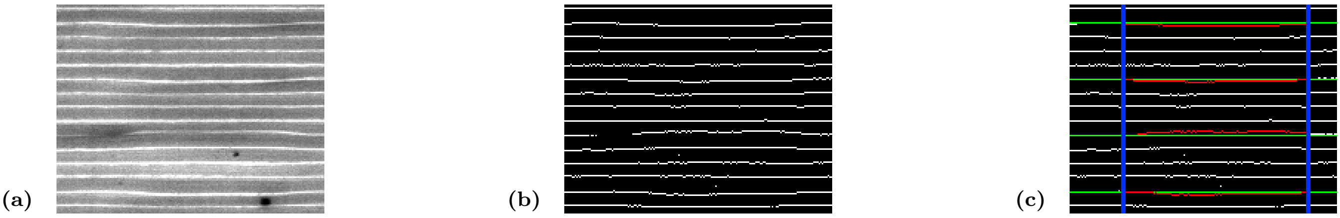

Fig. 5: (a) Image of the layers obtained from the 2d paper scanner. The highlights allow to segment each layer. (b) Edges extracted from image a (c) Edges with annotation, the blue vertical lines are the start and end of the encoding region, the red horizontal curved lines are the encoding region, the green horizontal lines are the middle between the 2 other edges. We decode a 1 or 0 if the red curve if above or below the green line, respectively. From top to below, the extracted bits are 0,0,1,0.

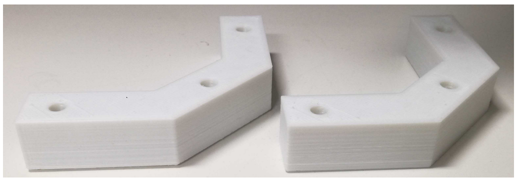

Fig. 6: left : non watermarked object, right : watermarked object (64 bits)

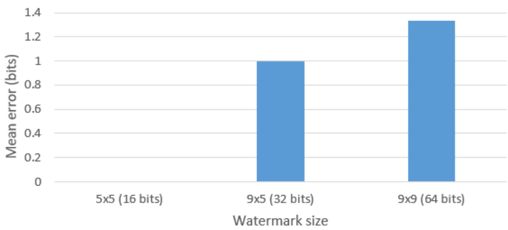

Fig. 7: Mean number of error bits during the extraction of the watermark

guaranteed to detect errors of up to 3 bits. Using redudancy, *i.e.*, comparing results from multiple markers from same or different faces, provides much higher resistance against errors.

We also tried the same model with gray PLA and 64 bits watermark, changed gamma to 1.0 and brightness to 30.0, and got 1 bit of error in 2 watermark patches, which could be recovered thanks to the parity bits error correction, and no error in the 3rd patch. The gray PLA is more shiny than white PLA, it makes the layers more easy to distinguish on the scanned image, but also increases the visibility of the artifacts caused by the watermark patches.

6. Conclusion

Our method has numerous advantages due to its low shape deformation and low visibility while providing a high data density, allowing high redundancy and resistance to attacks such as cropping and degradation. The extraction error rate is low, and the method is robust if we use redundancy, *i.e.*, embedding multiple patches in the same surface and comparing the decoded bits to find and correct the errors. For objects containing flat surfaces, which are common for CAD, our method can be applied with low cost equipment. Our method can be extended to objects without flat surfaces, the encoding method already works, but we still need to develop an extraction method for curved surfaces. Concretely, we need to identify and segment the printed layers, and follow the printed path to parametrise from 3d to 2d. In future work, we will try to develop new extraction methods working for any surface using high resolution 3d scanner, photometric stereo, or a combination of both for the layer detection. We will also try to modelise more precisely the printing process to compensate the remaining artifacts on the watermark patches.

References

[1] 3dHubs: Online Manufacturing Trends Q4-2018, [Online] Available: `https://www.3dhubs.com/trends` (2018). Accessed: 2019-05-27.

[2] Baraniuk, C.: Why printers add secret tracking dots, [Online] Available: `http://www.bbc.com/future/story/20170607-why-printers-add-secret-tracking-dots` (2017). Accessed: 2019-03-17.

[3] Bellini, A., Guceri, S. and Bertoldi, M.: Liquefier dynamics in fused deposition, *Journal of Manufacturing Science and Engineering*, Vol. 126, No. 2, pp. 237–246 (2004).

[4] Chiang, P.-J., Khanna, N., Mikkilineni, A. K., Segovia, M. V. O., Suh, S., Allebach, J. P., Chiu, G. T.-C. and Delp, E. J.: Printer and scanner forensics, *IEEE Signal Processing Magazine*, Vol. 26, No. 2, pp. 72–83 (2009).

[5] Delmotte, A., Tanaka, K., Kubo, H., Funatomi, T. and Mukaigawa, Y.: Blind Watermarking for 3-D Printed Objects using Surface Norm Distribution, *2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, IEEE, pp. 282–288 (2018).

[6] EFF: Investigating machine identification code technology in color laser printers, [Online] Available: `http://www.eff.org/wp/investigating-machine-identification-code-technology-color-laser-printers` (2005). Accessed: 2019-03-17.

[7] Greeff, G. P. and Schilling, M.: Closed loop control of slippage during filament transport in molten material extrusion, *Additive Manufacturing*, Vol. 14, pp. 31–38 (2017).

[8] Hodgson, G., Ranellucci, A. and Moe, J.: Slic3r Manual: Flow Math, [Online] `http://manual.slic3r.org/advanced/flow-math` (2011).

[9] Hou, J.-U., Kim, D.-G. and Lee, H.-K.: 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).

[10] Kumar, A., Goel, N. P. and Hemani, M.: Method and apparatus for storing and retrieving data embedded into the surface of a 3D printed object (2016). US Patent 9,400,910.

[11] Li, D., Nair, A. S., Nayar, S. K. and Zheng, C.: Aircode: unobtrusive physical tags for digital fabrication, *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, ACM, pp. 449–460 (2017).

[12] Li, Z., Rathore, A. S., Song, C., Wei, S., Wang, Y. and Xu, W.: PrinTracker: Fingerprinting 3D Printers using Commodity Scanners, *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ACM, pp. 1306–1323 (2018).

[13] Pandey, A. and Pradhan, S. K.: Investigations into Complete Liquefier Dynamics and Optimization of Process Parameters for Fused Deposition Modeling, *Materials Today: Proceedings*, Vol. 5, No. 5, pp. 12940–12955 (2018).

[14] Xia, H., Lu, J., Dabiri, S. and Tryggvason, G.: Fully resolved numerical simulations of fused deposition modeling. Part I: fluid flow, *Rapid Prototyping Journal*, Vol. 24, No. 2, pp. 463–476 (2018).