

Binary Image Carving for 3D Printing

Jingru Yang^a, Sha He^a, Lin Lu^{a,*}

^aSchool of Computer Science and Technology, Shandong University, Qingdao, China

Abstract

Pierced carving is an artistic form that perforates the given shape by texture patterns or images, widely used in furniture design and decorations. Usually this is manually done by expert designers and carvers, such that the produced carved model is of fine details and structural soundness. The emergence of 3D printing offers substantial opportunities for generalizing the manual efforts to an automatic fabricating manner. However, geometric modeling of the image pierced carving is non-trivial due to two challenging problems: how to adapt the given image to variant, limited 3D printing resolutions with high fidelity; how to ensure the printability and physical soundness of the carved model. In this paper, we take binary images as the input and attempt to tackle the pierced image carving problem from the above two folds. Given an arbitrary binary image and a 3D surface model, we formulate the image carving into image details adaption and structure enhancement problems, iteratively optimize the image and carved shape, and obtain a 3D printable and physically sound model. Experimental results show that our approach produces fabricable and visually satisfactory results.

Keywords: Binary Images, Pierced Carving, 3D Printing.

1. Introduction

Carving is considered as a human activity aimed at producing new shapes by cutting or engraving on initial shapes. Pierced carving is a delicate and intricate ornamental work made from wood, which has been known for centuries as the most effective form of decoration [1]. Similar artistic forms include paper cutting and filigrees, differing in materials like wood, paper, metal, etc. See Fig. 1 for instances. They all share the essence that relationships between positive and negative shapes and between solids and voids are rich in esthetic, cultural and philosophical associations. Owing to the sophisticated and delicate nature, such ornamental works have been popular for decades.



Figure 1: Left: Paper cutting (by artist *Qiuai Pan*); Middle: Filigrees (by *lumecluster*); Right: Wood pierced carving (by artist *Ken Hawtin*).

Usually, these carving works are handmade by skilled artists mastering craftsmanship. The carving patterns or images are carefully chosen or designed in an empirical manner. The base models are of limited shapes. Forming this artistic work via a geometric modeling problem has drawn lots of attention in the computer graphics community [2, 3], in such a way, the model can be automatically generated by computer numerical control (CNC) milling, laser cutting, etc.

With the emergence of 3D printing, the fabrication capabilities for complex geometries have been remarkably improved.

This motivates researchers to consider carving general images on general 3D shapes for numerous applications. However, current efforts mostly focus on synthesizing repetitive patterns on 3D surfaces, using the predesigned pattern images; leaving the pierced carving problem with an arbitrary image nearly unexplored.

Geometric modeling of the image pierced carving problem targeting for 3D printing is non-trivial due to the following reasons. First, the model details or features may be produced in different manners during 3D printing with lower resolutions, making it infeasible for filtering the image in a uniform manner to cure the details. Second, positive and negative features are of the same importance on visual perception, which may cause many conflicts for the detail treatment, as the operation space in the image domain is very limited. Third, the structural soundness requirements cannot be satisfied via pure geometric processing like the connectivity enforcement used in traditional paper cutting.

Despite the efforts to enhance the geometric details and structural soundness of the models for 3D printing [4, 5], we attempt to tackle the above challenges focusing on the image domain, considering the importance of both positive and negative details. Given an arbitrary binary image and a target surface, we aim at pierced carving the image on the surface in a predefined area, to obtain the 3D printable, stable shell model, while preserving the image details as much as possible.

For example, as shown in Fig. 2, given the binary image (a), if we carve the black tone regions on a plate without any modulation and 3D print the model, the model is directly broken as the areas with white tone (solid parts in the printout) are not fully connected (b). Even though we enforce the connectivity (c), the details cannot be fully realized (see the root) or could be violated during the fabrication (see the leaves). With the optimization proposed in the paper, we would achieve the carved image with an optimal compromise between the black and white details (d).

Our main contributions include,

- We examine the artifacts for both solid and void details in manufacturing with mainstream 3D printing techniques.
- We propose an evaluation on the image details with different

*Corresponding author

Email addresses: jingru.jem@gmail.com (Jingru Yang),
hesha.ada@gmail.com (Sha He), lulin.linda@gmail.com (Lin Lu)

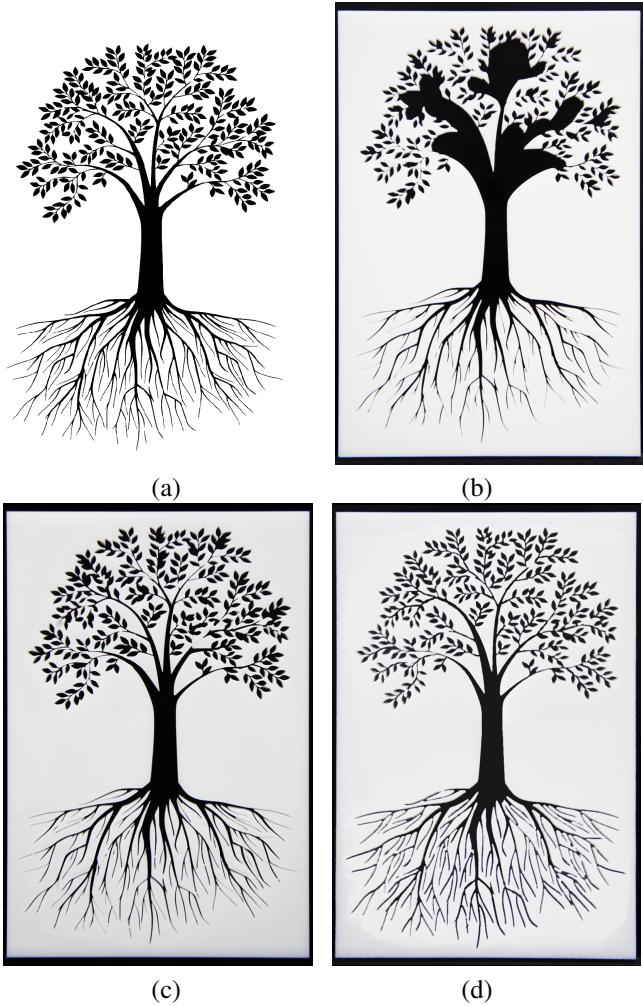


Figure 2: (a) Input image; (b) Physical printout without any treatment; (c) Physical printout only after the connectivity enforcement; (d) Physical printout after the optimization with our method. The size for (b-d) is 8cm×12cm×0.4cm.

characteristics, directly taking fabrication constraints into account, and considering both positive and negative features.

- We develop a framework for optimizing the binary image and carving on a shell-like object with high fidelity, printability and structural soundness.

2. Related Work

Image Binarization. Image binary stylization has been studied a lot in the non-photorealistic rendering (NPR) community. The methods are basically based on either region segmentation [7], or well designed filters [8, 9, 6]. Computer-aided paper cutting design also falls in this category. Comparing with normal image binarization, differences lie in the hard constraints on connectivity, i.e., all foreground pixels are connected. In this context, researchers have paid their attention to stylization for different types of image contents, and mostly enforce the connectivity in the postprocessing step. Xu et al. [3] achieved the binary image via multilayer thresholding operations and enforced connectivity by progressively adding paths between two components. Meng et al. [10] traced the key points on a portrait via facial recognition and solved the image binarization problem for portrait paper-cut.

Liu et al. [11] studied the cyclic and dihedral symmetries of different annuli in paper-cut designs and synthesized new designs with different rotational orders. Liu et al. [12] further developed a system for paper cutting fabrication, including a computational toolkit for pattern generation and light direction devices to provide fabricating instructions. Lin et al. [13] presented a computational framework for automatically generating 2D black-and-white abstractions from 3D man-made shapes at multiple scales. In contrast with the existing approaches, we take binary images as the input and aim at reproducing them on 3D surfaces with high fidelity, considering the fabrication constraints.

Textures/Images on Surfaces. For years, much research work focuses on synthesizing an exemplar texture to cover a target surface [14, 15]. These techniques are further extended by involving geometric textures [16, 17]. Mostly, textures are synthesized with the guidance of a vector or tensor field [18]. Bas-relief generation from images is also a well-studied area, commonly seen in architectural decorations or portraits on coins. The key issue is to infer depth values for objects in the image [19, 20, 21, 22]. Our work has a different target, and only deals with binary tones currently.

Fabrication-Aware Carving. Recently, the growing popularity of personalized fabrication has motivated researchers to develop algorithms to stylize 3D shapes via creative or artistic design [23]. Zhou et al. [24] generated single connected ornaments along curves via considering 1D pattern synthesis while constraining the topology. Dumas et al. [25] synthesized fabricable patterns along surfaces by combining structural properties optimization with appearance optimization. Martinez et al. [26] targeted the synthesis of flat shapes that are manufactured with laser cutting. Subsequently, Chen et al. [27] synthesized filigree-like structures along surfaces. They relaxed the packing problem by allowing appearance-preserving overlaps between elements. Their method produces appealing patterns, but cannot guarantee the base shape is preserved. Chen et al. [28] further proposed a method for generating packed patterns on a base surface with a set of flat tiles. Unlike our goal, they focused on patterns synthesis and only took account the solid parts in the optimization. Zehnder et al. [29] explored the interactive design of curve networks onto surfaces, and simulated the curves as elastic rods, such that a tight packing is achieved via minimized deformations. The produced airy curve networks can be fabricated on high-end printers. These work share the same scenario that taking the predefined ornamental pattern as an input and performing the structure synthesis.

There are some creative researches on designing or stylizing 3D shapes to generate target images. The shadow art work [30] computes a 3D sculpture that casts several 2D shadows close to the given binary images via geometric optimizations. The recent wire art work [31] creates objects with multiple interpretable line drawings at different viewpoints. Generating 3D printed, non-photorealistic line drawing styles from an input image has drawn research attention as well [32, 33]. The perforated lampshades [34] generate continuous projected images on the wall by carving optimized distributed tiny tubes on the shell-like lampshades. As far as we know, we are the first that tackle the image pierced carving problem in the context of 3D printing, targeting for high fidelity of the images, considering the fabrication constraints as physical resolutions and stability.

Optimization for 3D Printing. Optimizing the models for 3D printing in terms of printability, perceptual quality, etc. has gained much attention recently. Besides the interior structures design and modeling to optimize on the strength-to-weight ratio [35, 36], efforts have been made to repair and filter the

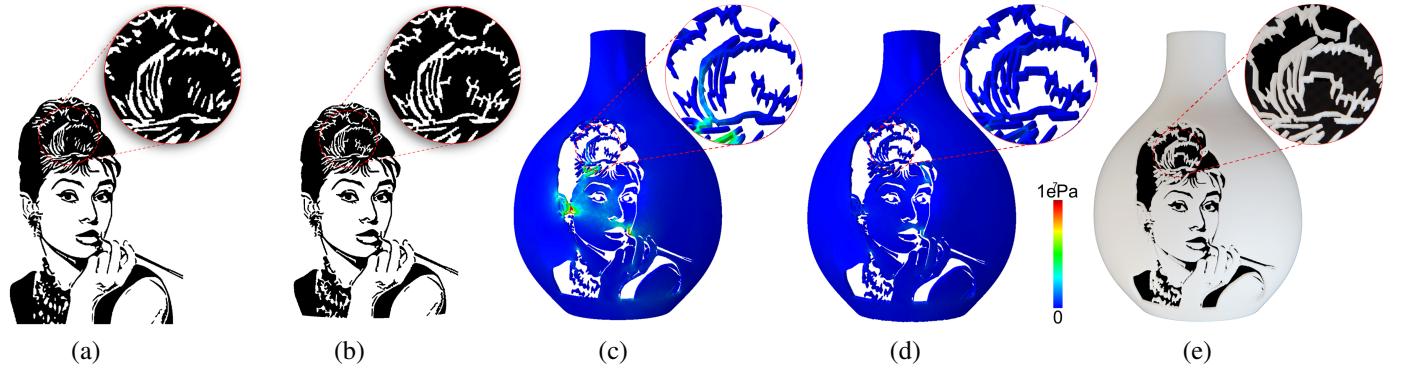


Figure 3: Given a binary image (a) and a target surface mesh, we first generate connections between isolated components, then further adapt the image details considering the fabrication constraints (b) and generate the 3D carved model (c). Finally, structural analysis and optimization is performed on the carved model, such that the fabricated model is stable (d). The image resolution is 1000×1000 , the parameterized image on target surface mesh is $12.4\text{cm} \times 12.4\text{cm}$, the physical printout (e) is $32.6\text{cm} \times 21.1\text{cm} \times 16.2\text{cm}$, with thickness 0.2 cm. Input image courtesy of [6].

unprintable models to adapt them to the fabrication requirements [37, 38]. Pintus et al. [4] modified the geometry of the shape to counter the effects of powder-based 3D printing. Herholz et al. [5] developed an unsharp masking approach to improve the visual appearance for 3D printing, with the hypothesis that light exaggeration of features helps to compensate for the deficiencies of 3D printed artifacts. The slicing is also critical to the quality of the printouts for FDM and SLA printing techniques. Methods have been proposed to optimize the slicing thickness regarding to the “staircase” effect [39, 40]. Zhang et al. [41] reduced the artifact caused by support structures and used perceptual models to minimize the artifacts during fabrication. Different from these works, we focus on the image carving problem and consider the equal importance of both positive and negative image details.

3. Overview

Given a binary image and a target shape, our goal is to carve the image on the surface on a predefined region in a pierced manner. Meanwhile, both the positive and negative features are preserved as much as possible under the given fabrication resolutions, and the fabricated model is of structural soundness. For easy exposition in this paper, we choose the black tone in the image as the negative part to be pierced and the white tone as the positive part to be printed in solid.

To accomplish this target, we propose an optimization pipeline with two stages, optimizing the details in the parameterized image domain and enhancing the structural stability in the shape domain.

First, we project the input image (Fig. 3a) on the surface with no self-occlusions, and compute the parameterized image by inverse perspective transformation. Objectives of the optimization say that the image details can be reproduced as much as possible under the given limited resolutions. Image optimization includes three steps, connectivity enforcement, compensation filtering, and details adaption. The full connectivity of the solid parts is achieved by generating minimal connections between isolated white components. Then we perform an image filtering to compensate the black tone due to extrusion artifacts. We propose an evaluation on the image, classify the details in terms of their reproducibilities and label the confliction areas, such that a fabrication-aware optimization on the image details can be performed. Black details, white details and conflict regions are optimized individually in an iterative manner. After the image optimi-

mizations, the parameterized result image can be directly carved on the surface (Fig. 3b).

The second stage is structural optimization, to guarantee that the carved model is stable and soundness. We perform structural analysis via finite element methods (FEM) to obtain the stress map and displacement map of the carved model under given forces (Fig. 3c). Then we iteratively search the largest displacement region and add connectors between the weak part and its closest neighboring part, until the whole piece is sustainable to the forces (Fig. 3d).

Finally, we obtain a fabricable and structurally sound model carved by the input image with high fidelity (Fig. 3e).

4. Image Optimization

In this section, we describe the optimization algorithm on the parameterized binary image I considering the fabrication constraints and features.

We first compute the parametrization of the image on the surface, and then all the following processes proceed in the parameterized image domain. Note that we only consider the shape with no self-occlusions, such that the carved image can be fully produced.

The algorithm is described in Alg. 1 as below. The fabrication feature tolerance φ is the input parameter. However, we will derive the actual minimal feature size for realizing solid and void parts based on certain fabrication constraints. Each main step will be elaborated in the following sections.

Algorithm 1 Image Optimization

Input: A binary image I , the fabrication tolerance φ ;

Output: The image that can be fully fabricated;

- 1: *Connectivity Enforcement* (I);
 - 2: *Determine the fabrication parameters* $\varphi_e, \varphi_s, \varphi_v$;
 - 3: *Compensation Filtering* (I, φ_e);
 - 4: *Details Adaption* (I, φ_s, φ_v);
 - 5: **return** the optimized image.
-

4.1. Connectivity Enforcement

Assume that we carve the black tone of the image out of the surface, we need to guarantee that the white parts form a single connected component. This coincides with the requirement for

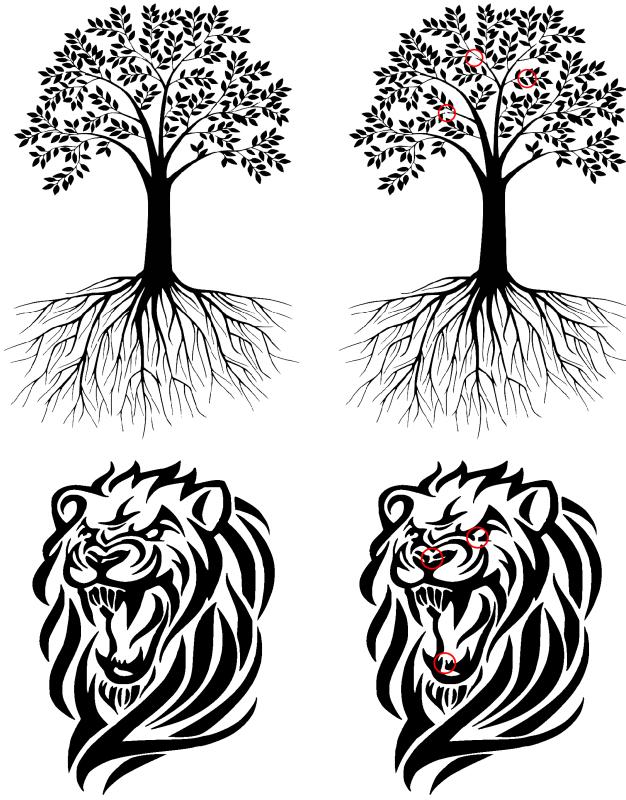


Figure 4: Examples on connectivity enforcement. Left: initial images; Right: fully-connected images for the white tone. Some revisions are highlighted in red circles.

paper cutting. Thus, we employ the method in [3] to accomplish it. The basic idea is to add the shortest path between isolated components. We first compute all white components in the image, and then iteratively pick out the smallest component and connect it with its nearest neighbor, with the minimal amount of pixels, until all white components are connected. The detailed algorithm is shown in Alg. 2. We show two examples of connectivity enforcement in Fig. 4.

Algorithm 2 Connectivity Enforcement

Input: The binary image I ;

Output: The image with white parts fully connected;

- 1: $N \leftarrow$ number of white components in I ;
 - 2: **while** $N > 1$ **do**
 - 3: *Connect the smallest white component with its closest neighboring component;*
 - 4: *Update N;*
 - 5: **end while**
 - 6: **return** the connected image.
-

4.2. Compensation Filtering

Fabrication Defects. During 3D printing the carved images, we observe that defects exist in the printouts, i.e., the solid part is wider than expected, and the void part is narrower than expected. Specifically, we design a reference board and print it on an FDM (fused deposition modeling) based 3D printer (Fig. 5). Then we physically measure all solid and void features, statistically compare with the designed size and compute the average error. For

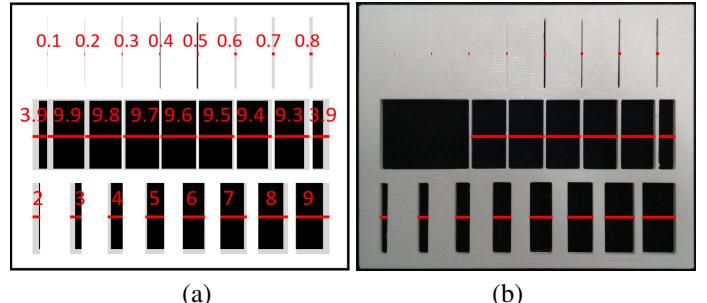


Figure 5: Fabrication defects exist for solid and void features, i.e., solid expands and void shrinks. (a) The digital reference board; (b) The FDM-printed model on a *Hori Z500* 3D printer. The sizes in mm for the void features are labelled on the board. Four void bars with width smaller than 0.5mm in the first row are not printed out. Two solid bars with width smaller than 0.3mm in the middle row are not printed out.

the printout in Fig. 5, the average error is 0.1mm. The minimal feature size in the specification is 0.4mm. However, from the printed reference board, the solid and void parts are fabricated in different minimal sizes. I.e., the minimal successfully printed solid feature is 0.3mm, the minimal realized void feature is 0.5mm. Actually, we tested on different 3D printing techniques including SLA (stereolithography) and SLS (selective laser sintering), and surprisingly found out that such defects appear as well.

To discover and overcome the potential defects, we print the reference board on the given 3D printer and introduce three parameters $\varphi_e, \varphi_s, \varphi_v$, to depict the actual printing feature size. The parameter φ_e represents the average error between solid and void features. The actual minimal feature size for solid and void parts is denoted by φ_s and φ_v , respectively. Suppose that the fabrication feature size in the specification is φ , we basically have $\varphi_s = \varphi - \varphi_e$, and $\varphi_v = \varphi + \varphi_e$. Regarding to the 3D printer in Fig. 5, $\varphi_e = 0.1\text{mm}$, $\varphi_s = 0.3\text{mm}$, $\varphi_v = 0.5\text{mm}$.

Compensation Filtering. To overcome this defect, we perform an erosion operator on the white tone for compensating the expansion of solid parts, as below.

$$I' = I \ominus S_C \quad (1)$$

Here, I' is the resulting image, S_C is the square structuring element for the compensation. The scale of S_C is determined by the resolution of the image and the physical printing parameters.

Suppose that the resolution of the image is $v \times v$, the physical size of the carved region is $\mu \times \mu$ and the erosion scale in the compensation is ω_c . Then, $\omega_c = \lceil \frac{\varphi_e v}{2\mu} \rceil + 2$. We restrict the ceiling operation here to odd numbers to feed the structuring element. The minimal number for ω_c is 3, to ensure the validity of the structuring element S_C .

The compensation filtering effectively improves the printout's appearance fidelity, as shown in Fig. 6. However, there still exists error comparing with the input image (see Fig. 6f). The main reason for the remaining error is that there are fine features whose size is smaller than the printing feature size. Such fine features may exist in white or black tones, which will be analyzed in the next section.

4.3. Details Evaluation

To deal with the mismatch of the digital image resolution and physical printing resolution and further reduce the error in the fabricated model, we need to identify the fine details whose size

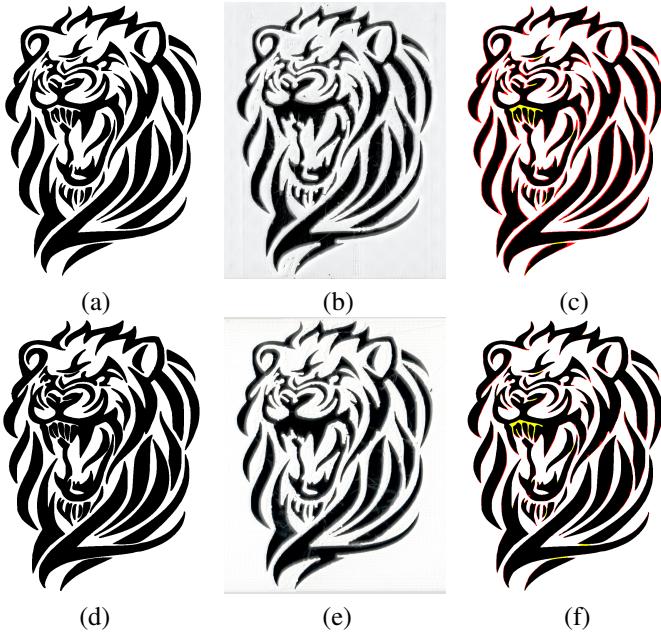


Figure 6: Demonstration on image compensation. (a) The image with white tone connected. (b) Scanning result of the printout from (a). (d) The image after compensation filtering. (e) Scanning result of the printout from (d). (c,f) The error map images, in which the red color indicates the missing part and the yellow color indicates the excess part between the physical printouts and input images. Physical printing size for both lion models is 7cm in height.

are below the physical realization size and treat them accordingly.

Details Detection. We denote the white and black details of the image by D_W and D_B , respectively. They can be extracted via morphological transformations from the image I and its inversion I^{-1} . We first perform opening operator on the binary image, which is defined as an erosion followed by a dilation using the same structuring element for both operations. Then the details are the difference between the resulting image and the original one.

$$D_W = |I - I \circ S_W| = |I - (I \ominus S_W) \oplus S_W| \quad (2)$$

$$D_B = |I^{-1} - I^{-1} \circ S_B| = |I^{-1} - (I^{-1} \ominus S_B) \oplus S_B| \quad (3)$$

In the two equations, S_W and S_B are the square structuring element for detecting details. Let ω_w and ω_b denote the scale for S_W and S_B , respectively. Similar to Eqn. 1, ω_w and ω_b is determined by the image resolution and fabrication parameters, i.e., $\omega_w = \lceil \frac{\varphi v}{\mu} \rceil$ and $\omega_b = \lceil \frac{\varphi v}{\mu} \rceil$. Similar to the computation of ω_c , the ceiling operations here are restricted to odd numbers, to feed the structuring element. The detected details are shown in Fig. 7a, highlighted in blue, for both tones.

Details Classification. Based on the shape and structures, we classify the details into three types. If its boundary in the other tone has more than one connected parts, such detail tends to be a “critical” region that distinguishes the two tones. We label it with *type A* detail. *Type B* detail component has only one neighboring boundary but in a thin and sharp shape, which implies an important feature in its tone. Otherwise, the detail component belongs to *type C*. These three types of details are shown in yellow, magenta and green, respectively in Fig. 7(b-c).

Type A details can be detected by its neighborhood in the other tone. While type B and C share the same neighboring structure and thus need a further check. Intuitively, the distinction between type B and C depends on the shape of the feature, in other words,

the visual significance. To measure the detail’s shape, we compute its *medial axis* and evaluate the length of the medial axis. If the length is larger than a predefined threshold $\frac{\varepsilon v}{\mu}$, the detail is classified to type B. Otherwise, it is regarded as type C. Here, the threshold parameter ε indicates the physical length of the thin feature. Only if the printed feature is longer than ε , we treat it; otherwise, it is ignored in the treatment. In our experiments, $\varepsilon = 4mm$.

From Fig. 7(b-c), we can see that type C details basically lie on the tip of the shape with small occupancy. For such kind of details, extra treatment would violate the appearance.

In the following, we would process type A and B details of the two tones, such that their features can be preserved as much as possible. For the sake of simplicity, we keep using the symbols D_W and D_B , while type C details are excluded (Fig. 7d).

4.4. Details Adaption

To preserve the details as much as possible with the given fabrication parameters, we need to modulate them to match the resolutions. Such modulation needs extra space. If there is enough operation space, we simply enlarge the small details such that they can be fabricated.

Conflict Areas. If details of two tones are adjacent, there might be no enough space. In other words, expansion of one part will violate the other part. We define such part as a conflict region, as seen in Fig. 7e, highlighted in red.

We denote the set of conflict pixels by Ω , which can be identified by,

$$\Omega = \{p | p \in D_W, \exists N_{\omega_w}(p) \in D_B \text{ or } p \in D_B, \exists N_{\omega_b}(p) \in D_W\}, \quad (4)$$

where $N_{\omega_w}(p)$ denotes the neighborhood of pixel p within ω_w length, and the same for $N_{\omega_b}(p)$.

There is no trivial solution to eliminate conflict pixels without damaging both tone details. Therefore, we judge the ownership of the conflict regions according to the area. Larger area in certain tone indicates the dominance of the tone.

Thus, the tone for each pixel in conflict region Ω_i is determined by the area of the white tone A_W and area of black tone A_B :

$$P_{\Omega_i} = \begin{cases} 1 & \text{if } A_W > A_B \\ 0 & \text{if } A_W < A_B \\ \text{random}(0, 1) & \text{if } A_W = A_B \end{cases} \quad (5)$$

Based on this strategy, if black is the majority tone in the conflict region, we will carve it as a black region, and vice versa. If both tones share equal proportion, we just randomly pick carving or printing.

With all the above analysis, we propose the image details adaption algorithm in Alg. 3. Since the adaption may cause a new disconnection of white tones, we iteratively perform the connectivity enforcement and details adaption till all details are treated. We note that both D_W and D_B details are treated with the same strategies, regardless of type A and B. However, along with the iterations, type A details either grow into conflicts or vanish. Type B details either change into conflicts or evolve into type C, as long as $\varepsilon > 0$. In such a way, the detail adaption converges. Fig. 8 shows the iterations of the treatment.

5. Structural Optimization

After the detail optimization in the image domain, we map the image on the surface and generate the pierced carving mesh

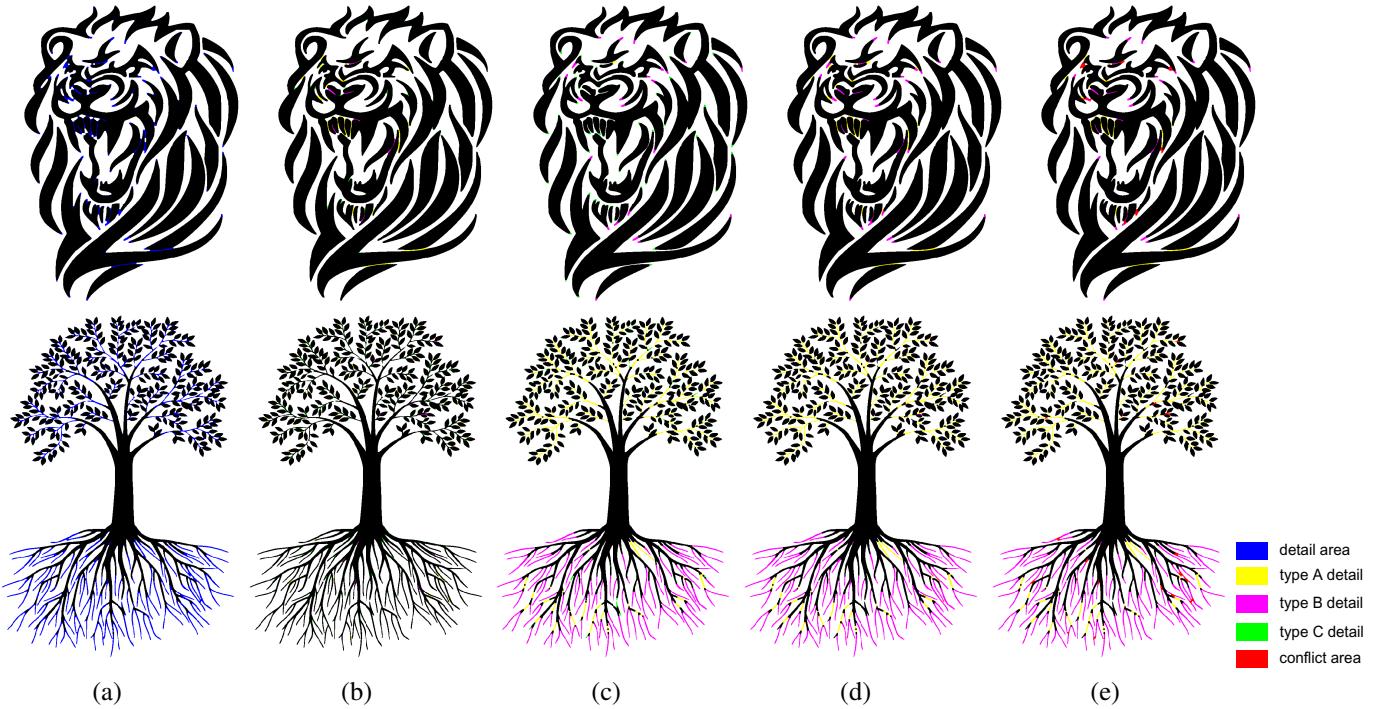


Figure 7: Details detection and analysis on the connected images, lion (upper row) and tree (lower row). (a) Detailed areas ($D_W \cup D_B$) highlighted in blue (lion: $\omega_w=5$, $\omega_b=7$, tree: $\omega_w=3$, $\omega_b=5$); (b) Three types of the white details D_W ; (c) Three types of the black details D_B ; (d) Details (type A and B) need to be treated for both tones; (e) Conflict areas are highlighted in red. The legend in the right bottom corner works for both cases.

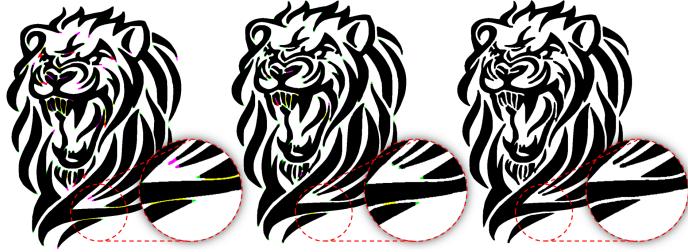


Figure 8: Process of the details adaption. It takes 5 iterations for the lion image to handle all the details including the conflicts. The rightmost image is the final result. The color legend is the same as that in Fig. 7.

model via computing the intersections between the image contour and the surface mesh, and the constrained Delaunay triangulation (Fig. 9).

To enhance the stability of the carved model, structural optimization is performed. First, we carry out the structural analysis via finite element methods (FEM) on the shell model. We fix the image boundaries as the boundary condition and apply uniform pressure on each face along the inverse of its normal direction. Both the stress map and displacement map of the carved model are obtained. Then we iteratively search the largest displacement region and add connectors between the weak part and its closest neighboring part, until the whole piece is stable. I.e., the maximal stress value is below the yielding value (10MPa for the PLA plastic in our experiments). Larger external forces are employed, more connectors would be added. See Fig. 10 for the demonstration, in which the external forces (totally 15KN) are uniformly applied to the image model.

Algorithm 3 Details Adaption

Input: A binary image I ; the feature tolerance φ_s, φ_v ;

Output: The image that can be fully fabricated;

- 1: $N \leftarrow$ number of white components in I ;
- 2: **repeat**
- 3: **Connectivity Enforcement** (I);
- 4: **Compute** D_W and D_B in I (Sec. 4.3);
- 5: **Detect** conflict pixel set Ω (Eqn. 4) and group connected conflict pixels;
- 6: **Treat** each conflict region Ω_i (Eqn. 5);
- 7: **Dilate** $D_W - \Omega$ and $D_B - \Omega$ with the minimal scale;
- 8: **Update** N, D_W, D_B ;
- 9: **until** $N = 1$ and $D_W = D_B = \emptyset$
- 10: **return** the current image.

6. Results and discussion

We implemented our proposed approach in C++, using a Windows 10 system with an Intel® Core™ i7-7700K CPU @ 3.6GHz and 16GB RAM. We use OpenCV library [42] for image processing, libigl [43] and CGAL [44] for mesh generation in the implementation. The OOFEM [45] library is utilized for structural analysis. We also employ the commercial software ABAQUS for validation and visualization on the results.

We take the widely used, average-quality 3D printers for fabricating the models. The Lion models in Fig. 12a are printed by FDM (*Hori Z500*). All the other models in this paper are printed by SLA. For the tree model, we printed it on a *UnionTech Lite600* 3D printer with $\varphi_e = 0.1mm$, $\varphi_s = 0.25mm$, $\varphi_v = 0.45mm$. The other models are printed on a *Formlabs Form 1+* printer with $\varphi_e = 0.1mm$, $\varphi_s = 0.3mm$, $\varphi_v = 0.5mm$.

Statistics of the results including image resolution, printing

Table 1: Statistics of the results.

Image	Resolution	Model	Size (cm)	$\omega_c, \omega_w, \omega_b$	ImageOpt. Time (s) & #Iter.	StructuralOpt. Time (s) & #Iter.
Tree1 (Fig. 2&Fig. 7)	800x1200	Plane	8x12x0.4	3, 3, 5	215 & 3	117 & 0
Tree2 (Fig. 12b)	800x1200	Plane	6.7x10x0.4	3, 3, 5	215 & 3	108 & 0
Tree3 (Fig. 12b)	800x1200	Plane	5.6x8.5x0.4	3, 5, 7	360 & 5	114 & 0
Lion1 (Fig. 7&Fig. 12a)	1000x1000	Plane	7x7x0.4	3, 5, 7	383 & 5	95 & 0
Lion2 (Fig. 12a)	1000x1000	Plane	4x4x0.4	3, 7, 13	574 & 7	93 & 0
Scenery1 (Fig. 12c)	2400x2400	Plane	12x12x0.4	3, 7, 11	1333 & 5	1203 & 8
Scenery2 (Fig. 12c)	2400x2400	Plane	8.5x8.5x0.4	5, 9, 15	851 & 5	870 & 7
Scenery3 (Fig. 13)	2400x2400	Plane	12x12x0.4	3, 7, 11	1287 & 6	607 & 4
Hepburn (Fig. 3)	1000x1000	Vase	32.6x21.1x16.2	3, 3, 5	251 & 5	528 & 4
Plum-flower (Fig. 19)	1500x1344	Bunny	22.5x21.6x17.3	3, 3, 5	692 & 4	384 & 3
Monroe (Fig. 19)	1000x700	Cylinder	7.1x7.4x3.3	3, 5, 7	236 & 3	154 & 1

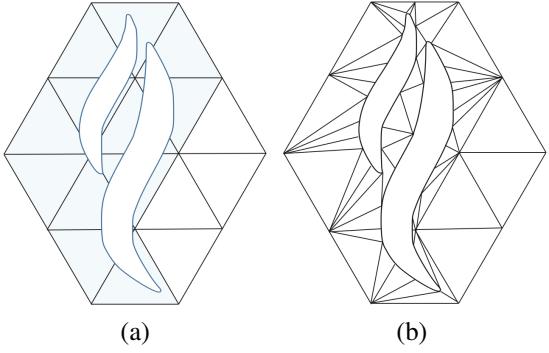


Figure 9: (a) The intersection between the mesh and image contours. (b) Constrained Delaunay triangulations of the remaining region.

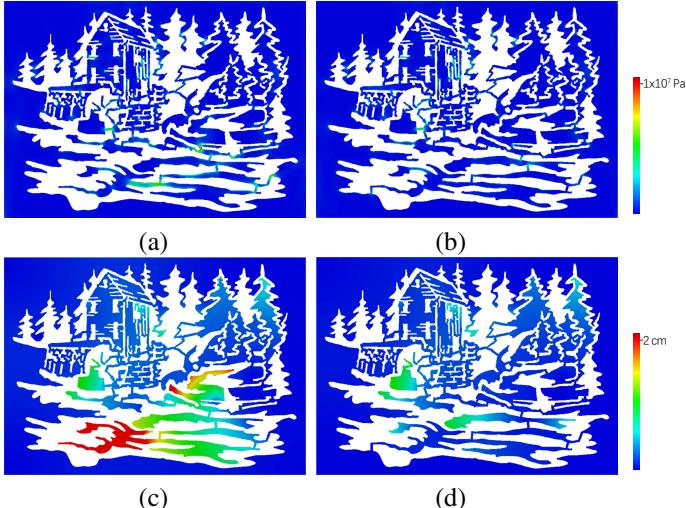


Figure 10: The stress map without optimization (a) and with optimization (b). The displacement map without optimization (c) and with optimization (d). (Size: 10cmx7.2cmx1cm. Force: 15KN)

size, running time and the number of iterations are listed in Tab. 1. The parameters ω_c , ω_w , ω_b derived from the physical sizes and fabrication constraints are also included. Five models have zero number of iterations in structural optimization as they are stable in the first FEM simulation.

To illustrate the effectiveness of our algorithm for retaining the details considering the fabrication constraints, we enforce the connectivity of all the input images and print the connected carving results for comparisons. Fig. 12 shows examples for the 3D printed results in two different sizes. For the lion image (Fig. 12a), to fully realize its details on a plate, the physical size needs at least 15cm×15cm. If printing in smaller sizes like 7cm×7cm and 4cm×4cm, the details cannot be all preserved. Comparing to its original version with connectivity enforcement only, we can see the optimized results preserve more details and offer more fidelity. For the tree image (Fig. 12b), it requires at least a 12cm×18cm plate to fully realize the details without adaptation. For the scenery image (Fig. 12c), the minimal printing size would be 36cm×36cm to preserve the image details.

The error maps for the four lion models (Fig. 12a) are shown in Fig. 11, which refer to the difference between the physical printouts and the input image. We computed all the precision rates, i.e., the ratio of correct pixels to the total pixels, on the printed results for evaluation, shown in Fig. 14.

Since our algorithm considers both positive and negative features, it is simple to carve the white tone, i.e., just reversing the input image. Fig. 13 shows these dual results on the scenery image. Fig. 16 gives an example optimized on a low precision FDM printer. We also show printed results carving on the surface in Fig. 19.

We also carried out a user study to subjectively evaluate the results. We listed photographs of both printouts with and without optimization for eight models, including the Tree image in three scales, Lion image in two scales, and Scenery image in two scales and the inverted one. Questions are designed from two aspects, the similarity with the input images and the aesthetics of the printouts. We built an online questionnaire including 16 questions. For each model, we show the two photographs of the printouts before and after optimization in scrambled order and design two questions. One is asking the user to pick the photograph that is closer to the input image, the other is asking the user to pick the photograph that is of more aesthetics. The 100 volunteers include 80 college students from different majors in different universities and 20 persons with little background in computer science. The college students age from 20 to 30 and the others are from 40 to 55. All volunteers finished the ques-

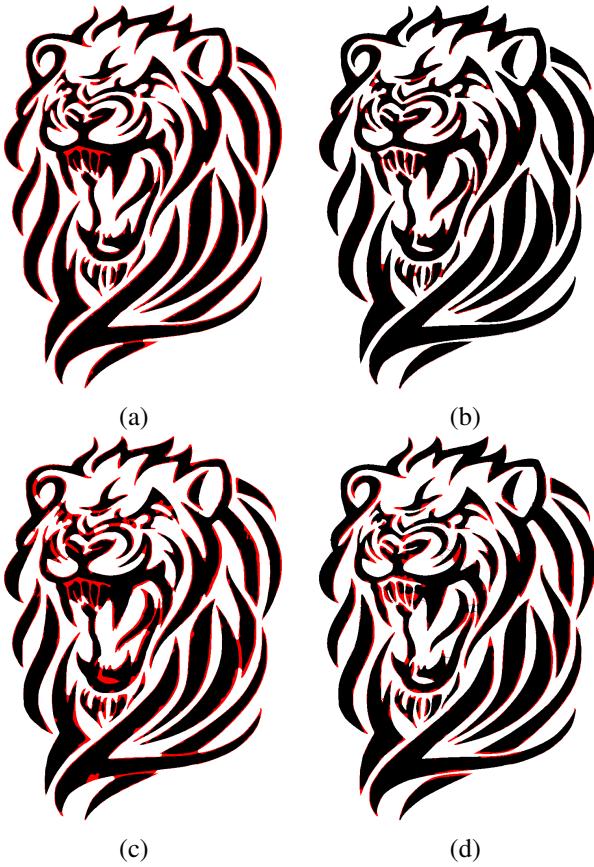


Figure 11: Error maps for the four models in Fig. 12a. The red color indicates the error comparing to the initial image. Lion model in 7cm×7cm without optimization has the error rate 4.914% (a), while after optimization the error rate reduces to 1.538% (b). Lion model in 4cm×4cm without optimization has the error rate 5.394% (c), while after adaption the error rate is 3.302% (d).

tions without pre-training or interruptions. It took each volunteer 2–5 minutes on average to finish the test depending on their background. The results are summarized in Fig. 15, in which the vertical axis is the percentage that the user picked.

Statistics show that over 80% users think our optimized results are more similar to the initial images. The number of users that like our optimized results from the aesthetics aspect is slightly smaller than from the similarity aspect, but still with the majority. This happens especially for elder users with no computer science background. Users take more time to choose the answer for models in larger sizes.

Limitations. There is no specific request on the input binary images, but the aesthetics of the resulting models do depend on the input. If the isolated white features are distributed too sparse, the connectors added in the connectivity enforcement stage might violate the overall aesthetics, as shown in Fig. 17. If the features are too dense to be fully realized, i.e., the details in two tones are interleaving and heavily conflicting, the conflict treatment would drive the result image into a hole or a solid region (Fig. 18). Moreover, since only the details below the printing tolerances are treated, we lose the stroke variations in the details adaption.

7. Conclusion and future work

In this paper, we discuss the problem of carving a binary image on the target surface in the context of 3D printing. Regarding

the equivalent importance of positive and negative features, we examine the defects on both solid and void parts in fabrication. We present a framework for optimizing the binary image to preserve more details in the printouts and the carved model for the structural soundness. The result models satisfy the fabrication constraints and can be produced with high fidelity.

Our current image optimization is only based on local geometric details. We would like to take higher level details from the knowledge domain, e.g., the objects layouts and attributes into consideration in further research. The importance map from visual perception may also help. Moreover, the functionality of the carved object would be considered to integrate in the whole framework.

Acknowledgements

This work was supported by grants from National Natural Science Foundation of China (61572291) and Young Scholars Program of Shandong University (YSPSDU).

References

- [1] P. N. Hasluck, Manual of Traditional Wood Carving, Dover Publications, 1977.
- [2] A. Pasko, V. Savchenko, A. Sourin, Synthetic carving using implicit surface primitives, Computer-Aided Design 33 (5) (2001) 379 – 388. doi:[http://doi.org/10.1016/S0010-4485\(00\)00129-9](http://doi.org/10.1016/S0010-4485(00)00129-9). URL <http://www.sciencedirect.com/science/article/pii/S0010448500001299>
- [3] J. Xu, C. S. Kaplan, X. Mi, Computer-generated papercutting, in: Computer Graphics and Applications, 2007. PG '07. 15th Pacific Conference on, 2007, pp. 343–350. doi:[10.1109/PG.2007.10](https://doi.org/10.1109/PG.2007.10).
- [4] R. Pintus, E. Gobbetti, P. Cignoni, R. Scopigno, Shape enhancement for rapid prototyping, The Visual Computer 26 (6) (2010) 831–840. doi:[10.1007/s00371-010-0488-0](https://doi.org/10.1007/s00371-010-0488-0) URL <https://doi.org/10.1007/s00371-010-0488-0>
- [5] P. Herholz, S. Koch, T. Boubekeur, M. Alexa, Unsharp masking geometry improves 3d prints, Computers & Graphics 66 (2017) 135 – 142, shape Modeling International 2017. doi:<https://doi.org/10.1016/j.cag.2017.05.018>. URL <http://www.sciencedirect.com/science/article/pii/S0097849317300675>
- [6] H. Li, D. Mould, Contrast-enhanced black and white images, Comput. Graph. Forum 34 (7) (2015) 319–328. doi:[10.1111/cgf.12770](https://doi.org/10.1111/cgf.12770). URL <http://dx.doi.org/10.1111/cgf.12770>
- [7] J. Xu, C. S. Kaplan, Artistic thresholding, in: Proceedings of the 6th International Symposium on Non-photorealistic Animation and Rendering, NPAR '08, ACM, New York, NY, USA, 2008, pp. 39–47. doi:[10.1145/1377980.1377990](https://doi.org/10.1145/1377980.1377990). URL <http://doi.acm.org/10.1145/1377980.1377990>
- [8] H. Winnemöller, J. E. Kyprianidis, S. C. Olsen, Xdog: An extended difference-of-gaussians compendium including advanced image stylization, Computers & Graphics 36 (6) (2012) 740 – 753, 2011 Joint Symposium on Computational Aesthetics (CAe), Non-Photorealistic Animation and Rendering (NPAR), and Sketch-Based Interfaces and Modeling (SBIM). doi:[http://doi.org/10.1016/j.cag.2012.03.004](https://doi.org/10.1016/j.cag.2012.03.004). URL <http://www.sciencedirect.com/science/article/pii/S009784931200043X>
- [9] P. L. Rosin, Y.-K. Lai, Artistic minimal rendering with lines and blocks, Graphical Models 75 (4) (2013) 208 – 229. doi:[http://doi.org/10.1016/j.gmod.2013.03.004](https://doi.org/10.1016/j.gmod.2013.03.004). URL <http://www.sciencedirect.com/science/article/pii/S1524070313000143>
- [10] M. Meng, M. Zhao, S.-C. Zhu, Artistic paper-cut of human portraits, in: Proceedings of the 18th ACM International Conference on Multimedia, MM '10, ACM, New York, NY, USA, 2010, pp. 931–934. doi:[10.1145/1873951.1874116](https://doi.org/10.1145/1873951.1874116). URL <http://doi.acm.org/10.1145/1873951.1874116>

- [11] Y. Liu, J. Hays, Y.-Q. Xu, H.-Y. Shum, Digital papercutting, in: ACM SIGGRAPH 2005 Sketches, SIGGRAPH '05, ACM, New York, NY, USA, 2005. doi:10.1145/1187112.1187231.
- [12] L. Liu, Y. Chen, P. Wang, Y. Liu, C. Zhang, X. Li, C. Yao, F. Ying, Papercut: Digital fabrication and design for paper cutting, in: Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems, CHI EA '18, ACM, New York, NY, USA, 2018, pp. LBW078:1–LBW078:6. doi:10.1145/3170427.3188522. URL <http://doi.acm.org/10.1145/3170427.3188522>
- [13] Y.-E. Lin, Y.-L. Yang, H.-K. Chu, Scale-aware black-and-white abstraction of 3d shapes, ACM Trans. Graph. 37 (4) (2018) 117:1–117:11. doi:10.1145/3197517.3201372. URL <http://doi.acm.org/10.1145/3197517.3201372>
- [14] G. Turk, Texture synthesis on surfaces, in: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '01, ACM, New York, NY, USA, 2001, pp. 347–354. doi:10.1145/383259.383297. URL <http://doi.acm.org/10.1145/383259.383297>
- [15] S. Lefebvre, H. Hoppe, Appearance-space texture synthesis, ACM Trans. Graph. 25 (3) (2006) 541–548. doi:10.1145/1141911.1141921. URL <http://doi.acm.org/10.1145/1141911.1141921>
- [16] K. Zhou, X. Huang, X. Wang, Y. Tong, M. Desbrun, B. Guo, H.-Y. Shum, Mesh quilting for geometric texture synthesis, ACM Trans. Graph. 25 (3) (2006) 690–697. doi:10.1145/1141911.1141942. URL <http://doi.acm.org/10.1145/1141911.1141942>
- [17] Y. Li, F. Bao, E. Zhang, Y. Kobayashi, P. Wonka, Geometry synthesis on surfaces using field-guided shape grammars, IEEE Transactions on Visualization and Computer Graphics 17 (2) (2011) 231–243. doi:10.1109/TVCG.2010.36. URL <http://dx.doi.org/10.1109/TVCG.2010.36>
- [18] V. Amir, C. Marcel, D. Olga, P. Daniele, B. David, H. Klaus, B.-C. Mirela, Directional field synthesis, design, and processing, Computer Graphics Forum 35 (2) (2016) 545–572. arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12864>, doi:10.1111/cgf.12864. URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12864>
- [19] M. Alexa, W. Matusik, Reliefs as images, ACM Trans. Graph. 29 (4) (2010) 60:1–60:7. doi:10.1145/1778765.1778797. URL <http://doi.acm.org/10.1145/1778765.1778797>
- [20] J. Wu, R. R. Martin, P. L. Rosin, X. F. Sun, F. C. Langbein, Y. K. Lai, A. D. Marshall, Y. H. Liu, Making bas-reliefs from photographs of human faces, Comput. Aided Des. 45 (3) (2013) 671–682. doi:10.1016/j.cad.2012.11.002. URL <http://dx.doi.org/10.1016/j.cad.2012.11.002>
- [21] J. Wu, R. Martin, P. Rosin, X.-F. Sun, Y.-K. Lai, Y.-H. Liu, C. Wallraven, Use of non-photorealistic rendering and photometric stereo in making bas-reliefs from photographs, Graphical Models 76 (4) (2014) 202 – 213. doi:<https://doi.org/10.1016/j.gmod.2014.02.002>. URL <http://www.sciencedirect.com/science/article/pii/S152407031400006X>
- [22] Q. Zeng, R. R. Martin, L. Wang, J. A. Quinn, Y. Sun, C. Tu, Region-based bas-relief generation from a single image, Graphical Models 76 (3) (2014) 140 – 151, computational Visual Media Conference 2013Second Computational Visual Media Conference (CVM). doi:<http://doi.org/10.1016/j.gmod.2013.10.001>. URL <http://www.sciencedirect.com/science/article/pii/S1524070313000283>
- [23] B. Bernd, C. Paolo, M. Luigi, P. Nico, State of the art on stylized fabrication, Computer Graphics Forum 0 (0). arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13327>, doi:10.1111/cgf.13327. URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13327>
- [24] S. Zhou, C. Jiang, S. Lefebvre, Topology-constrained synthesis of vector patterns, ACM Trans. Graph. 33 (6) (2014) 215:1–215:11. doi:10.1145/2661229.2661238. URL <http://doi.acm.org/10.1145/2661229.2661238>
- [25] J. Dumas, A. Lu, S. Lefebvre, J. Wu, C. Dick, By-example synthesis of structurally sound patterns, ACM Trans. Graph. 34 (4) (2015) 137:1–137:12. doi:10.1145/2766984. URL <http://doi.acm.org/10.1145/2766984>
- [26] J. Martínez, J. Dumas, S. Lefebvre, L.-Y. Wei, Structure and appearance optimization for controllable shape design, ACM Trans. Graph. 34 (6) (2015) 229:1–229:11. doi:10.1145/2816795.2818101.
- [27] W. Chen, X. Zhang, S. Xin, Y. Xia, S. Lefebvre, W. Wang, Synthesis of filigrees for digital fabrication, ACM Trans. Graph. 35 (4) (2016) 98:1–98:13. doi:10.1145/2897824.2925911. URL <http://doi.acm.org/10.1145/2897824.2925911>
- [28] W. Chen, Y. Ma, S. Lefebvre, S. Xin, J. Martínez, w. wang, Fabricable tile decors, ACM Trans. Graph. 36 (6) (2017) 175:1–175:15. doi:10.1145/3130800.3130817. URL <http://doi.acm.org/10.1145/3130800.3130817>
- [29] J. Zehnder, S. Coros, B. Thomaszewski, Designing structurally-sound ornamental curve networks, ACM Trans. Graph. 35 (4) (2016) 99:1–99:10. doi:10.1145/2897824.2925888. URL <http://doi.acm.org/10.1145/2897824.2925888>
- [30] N. J. Mitra, M. Pauly, Shadow art, ACM Trans. Graph. 28 (5) (2009) 156:1–156:7. doi:10.1145/1618452.1618502. URL <http://doi.acm.org/10.1145/1618452.1618502>
- [31] K.-W. Hsiao, J.-B. Huang, H.-K. Chu, Multi-view wire art, ACM Trans. Graph. 37 (6) (2018) 242:1–242:11. doi:10.1145/3272127.3275070. URL <http://doi.acm.org/10.1145/3272127.3275070>
- [32] Z. Chen, Z. Shen, J. Guo, J. Cao, X. Zeng, Line drawing for 3d printing, Computers & Graphics 66 (2017) 85 – 92. doi:<https://doi.org/10.1016/j.cag.2017.05.019>. URL <http://www.sciencedirect.com/science/article/pii/S0097849317300687>
- [33] Z. Chen, W. Chen, J. Guo, J. Cao, Y. J. Zhang, Orientation field guided line abstraction for 3d printing, Computer Aided Geometric Design 62 (2018) 253 – 262. doi:<https://doi.org/10.1016/j.cagd.2018.03.014>. URL <http://www.sciencedirect.com/science/article/pii/S0167839618300281>
- [34] H. Zhao, L. Lu, Y. Wei, D. Lischinski, A. Sharf, D. Cohen-Or, B. Chen, Printed perforated lampshades for continuous projective images, ACM Trans. Graph. 35 (5) (2016) 154:1–154:11. doi:10.1145/2907049. URL <http://doi.acm.org/10.1145/2907049>
- [35] L. Lu, A. Sharf, H. Zhao, Y. Wei, Q. Fan, X. Chen, Y. Savoye, C. Tu, D. Cohen-Or, B. Chen, Build-to-last: Strength to weight 3D printed objects, ACM Trans. Graph. 33 (4) (2014) 97:1–97:10. doi:10.1145/2601097.2601168. URL <http://doi.acm.org/10.1145/2601097.2601168>
- [36] Y. Mao, L. Wu, D.-M. Yan, J. Guo, C. W. Chen, B. Chen, Generating hybrid interior structure for 3d printing, Computer Aided Geometric Design 62 (2018) 63 – 72. doi:<https://doi.org/10.1016/j.cagd.2018.03.015>. URL <http://www.sciencedirect.com/science/article/pii/S0167839618300293>
- [37] Z. Bo, L. Lu, A. Sharf, Y. Xia, O. Deussen, B. Chen, Printable 3d trees, Computer Graphics Forum 36 (7) (2017) 29–40. arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13269>, doi:10.1111/cgf.13269. URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13269>
- [38] M. Attene, As-exact-as-possible repair of unprintable stl files, Rapid Prototyping Journal 24 (5) (2018) 855–864. arXiv:<https://doi.org/10.1108/RPJ-11-2016-0185>. URL <https://doi.org/10.1108/RPJ-11-2016-0185>
- [39] W. Wang, H. Chao, J. Tong, Z. Yang, X. Tong, H. Li, X. Liu, L. Liu, Saliency-preserving slicing optimization for effective 3d printing, Computer Graphics Forum 34 (6) (2015) 148–160. arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12527>, doi:10.1111/cgf.12527. URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12527>
- [40] M. Alexa, K. Hildebrand, S. Lefebvre, Optimal discrete slicing, ACM Trans. Graph. 36 (1). doi:10.1145/2999536. URL <http://doi.acm.org/10.1145/2999536>
- [41] X. Zhang, X. Le, A. Panotopoulou, E. Whiting, C. C. L. Wang, Perceptual models of preference in 3d printing direction, ACM Trans. Graph. 34 (6) (2015) 215:1–215:12. doi:10.1145/2816795.2818121. URL <http://doi.acm.org/10.1145/2816795.2818121>
- [42] Open source computer vision library, <https://github.com/opencv> (2015).
- [43] A. Jacobson, D. Panozzo, Libigl: Prototyping geometry processing research in c++, in: SIGGRAPH Asia 2017 Courses, SA '17, ACM, New York, NY, USA, 2017, pp. 11:1–11:172. doi:10.1145/3134472.3134497.

- [44] The CGAL Project, CGAL User and Reference Manual, 4.12 Edition, CGAL Editorial Board, 2018.
URL <https://doc.cgal.org/4.12/Manual/packages.html>
- [45] B. Patzák, OOFEM - an object-oriented simulation tool for advanced modeling of materials and structures, *Acta Polytechnica* 52 (6) (2012) 59–66.



(a)



(b)



(c)

Figure 12: 3D printed results for the models in two different sizes without optimization (left) and with optimization (right). The information is listed in Tab. 1.

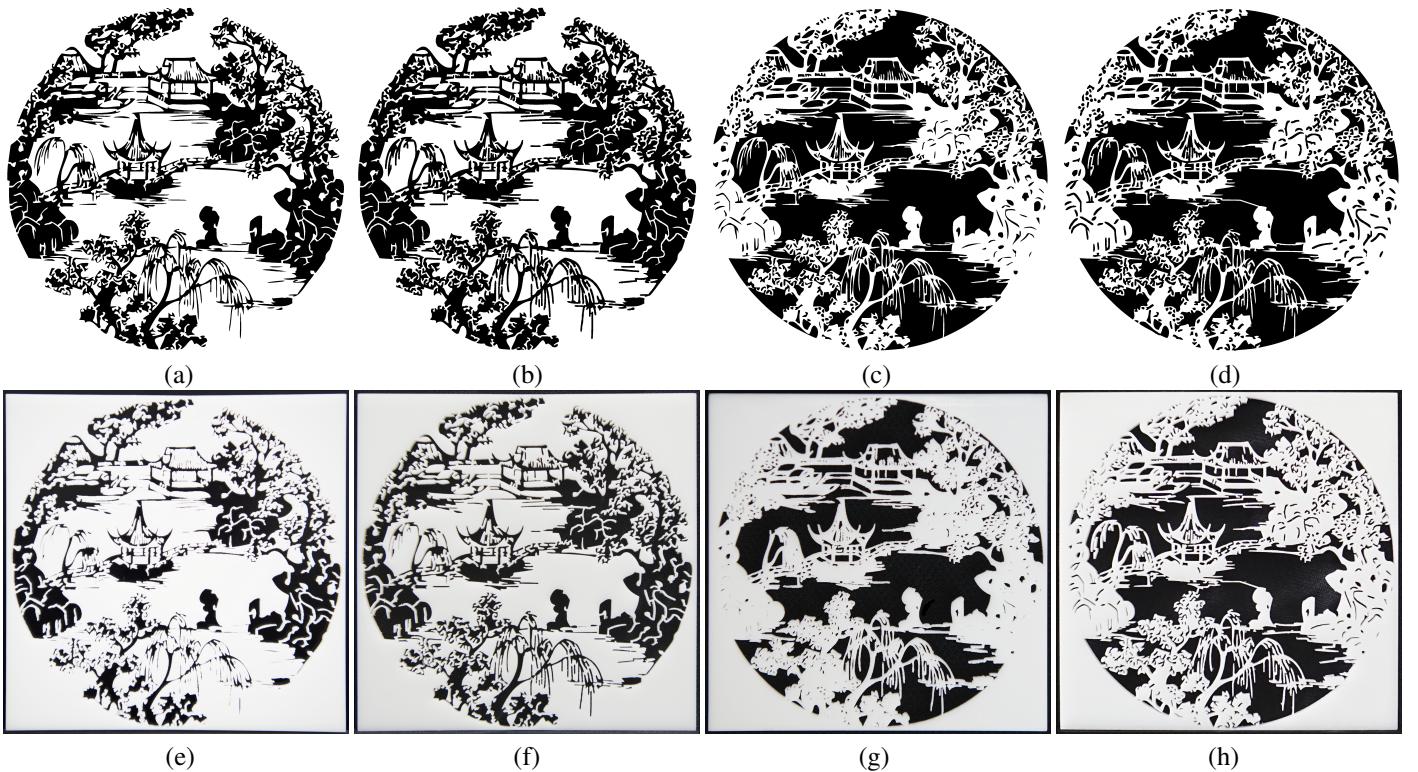


Figure 13: For given scenery image (a), we can perform the algorithm on its inversion image (c). The images in (b) and (d) are the results after optimization. The printouts are shown in the second row, corresponding to its image in the first row. The size is 12cm×12cm×0.4cm.

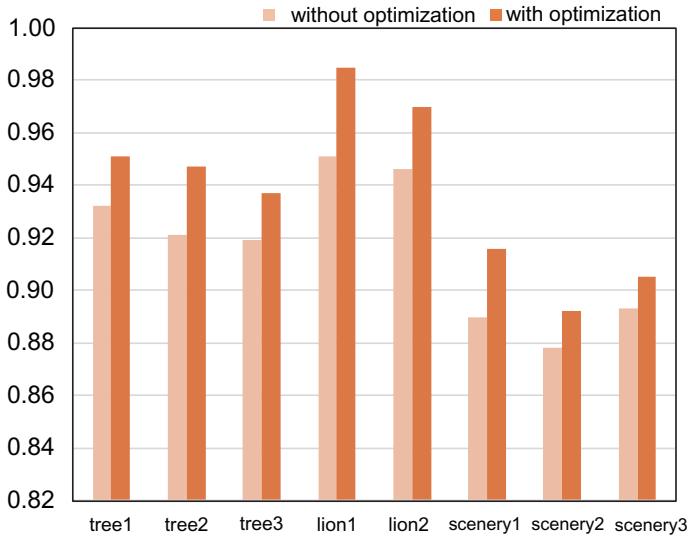


Figure 14: The precision rates for eight printed models comparing to the input images.

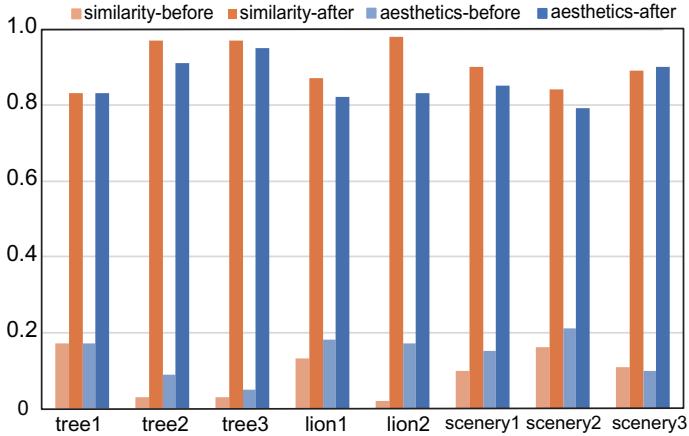


Figure 15: Statistics of the user study. The vertical axis is the percentage of the users who picked the certain models.

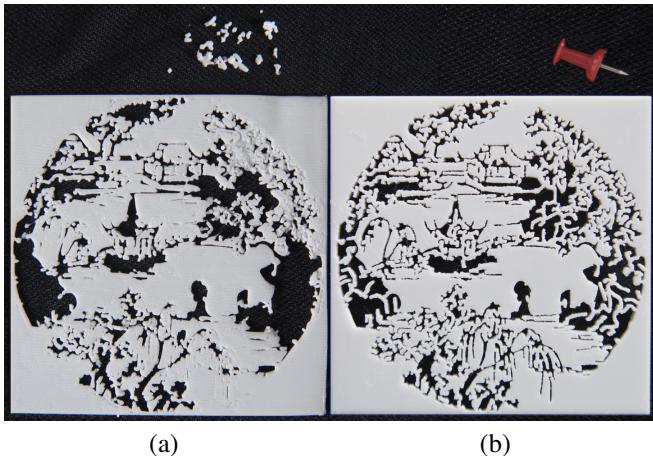


Figure 16: 3D printed results in 8.5×8.5×0.4cm without (a) and with optimization (b).

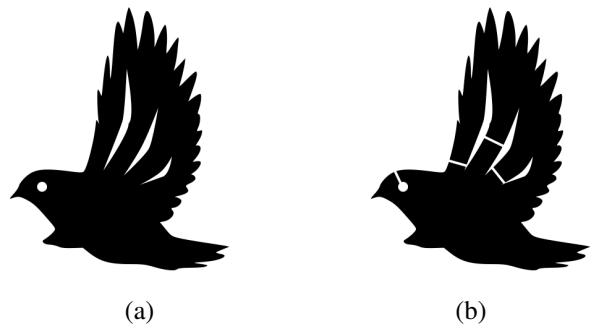


Figure 17: Connectivity enforcement produces visible connectors (b) in the input image with sparse and isolated white details (a).

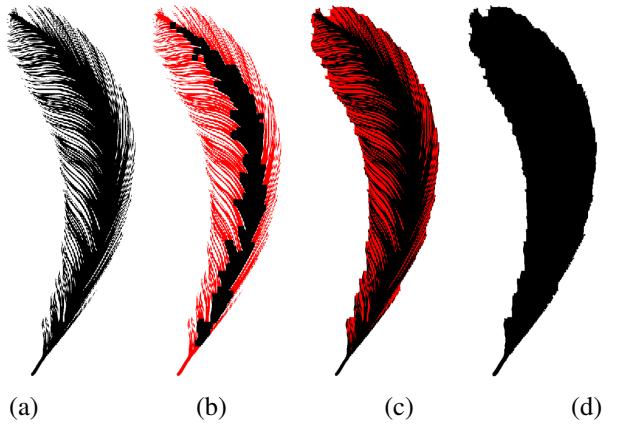


Figure 18: (a) Input image. D_B and D_W is highlighted as red in (b) and (c), respectively. (d) Details adaption result.



Figure 19: More printed results. Left: Plum flower on the bunny model, 22.5cm×21.6cm×17.3cm. Right: Monreо on the cylinder model, 7.1cm×7.4cm×3.3cm. The thickness for both models is 0.2 cm.