Computers & Graphics 131 (2025) 104337

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on CAD/Graphics 2025

Computational multi-layered wood carving art

Haochen Liu [a], Zhi Li [b], Kang Wu [a,*], Youcheng Cai [a], Xiaoya Zhai [a], Ketian Zhang [c], Ligang Liu [a], Yi Min Xie [b,d], Xiao-Ming Fu [a]

[a] School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, PR China
[b] Centre for Innovative Structures and Materials, School of Engineering, RMIT University, Melbourne, Victoria 3001, Australia
[c] STEREOWOOD Design Ltd., 171-75 Shelton Street, Covent Garden, London, WC2H 9JQ, United Kingdom
[d] College of Future Technologies, Hohai University, Changzhou, Jiangsu 213200, China

ARTICLE INFO

Keywords:
Computational design
Laser cutting
Wooden artworks
Path generation
Image vectorization
Personalized fabrication

ABSTRACT

We present a computational framework for designing Multi-layered Wood Carving Artwork (MWCA), where intricately carved, colored wooden layers are stacked to create visually striking compositions. Each laser-cut layer must preserve global connectivity and minimize fine branching to maintain structural integrity. Traditionally reliant on artisanal trial-and-error, MWCA design has been time-consuming and labor-intensive. Our method is driven by a key observation: during connectivity analysis, the region of an upper layer can be treated as part of the lower layers. Leveraging this insight, we develop an iterative greedy algorithm to jointly determine the color assignment and geometric shape of each layer. To facilitate processing, we perform an image pre-processing step that reduces the input to a limited color palette. Additionally, a post-processing step is incorporated to enhance the structural integrity of the final output. We demonstrate the effectiveness and practical viability of our approach through 18 examples, including 3 fabrication results.

1. Introduction

Multi-layered wood carving artworks (MWCA) refer to an innovative art form characterized by assembling multiple carved wooden sheets layer by layer [1]. This art form has emerged due to its stereo visual effect and remarkably low fabrication costs. Contemporary artists can design cutting paths to produce highly precise and detailed patterns on wooden sheets [2]. After cutting, each wooden sheet is individually sprayed with a single, uniform color to simplify production and enhance layer differentiation. By gluing multiple fabricated sheets together, the artworks showcase sophisticated effects with visual depth and aesthetic richness. As a unique integration of digital fabrication processes with traditional artistic materials, this art form has been widely employed in interior decoration and furniture design (see Fig. 1).

Although the production of MWCA is relatively straightforward, the design process is challenging due to three design constraints. First, each layer must form a contiguous whole without any isolated fragments (connectivity). Second, the colors from the original image must be categorized to ensure that each layer can be uniformly painted with only one color (color uniformity). Third, the design of the layers should avoid overly slender members and stress concentration areas, as this may cause localized fractures of the panels during assembly and movement (structural performance). Current design methods rely heavily on manual processes—each piece requires artists to spend several days drafting the laser-cutting path. Therefore, the existing design process is complicated and time-consuming, as it demands a high level of artistic expertise and a deep understanding of the laser-cutting techniques.

Due to the demand of contemporary consumers for individual expression, personalized fabrication has become a defining trend in contemporary art [3]. However, the customized design process of MWCA relies heavily on labor resources, which significantly improve the cost of an MWCA, thereby limiting the scalability of MWCA. Recent studies have leveraged computational design to fulfill the requirements of personalized fabrication and saving costs. Significant advancements include peeling art design [4], parquetry [5], Chinese paper-cutting [6], overpaint [7], illusion knitting [8], and LEGO toys [9–11]. However, computational design methods specifically tailored to the unique design constraints of MWCA have not yet been thoroughly explored.

In this paper, we propose a computational pipeline that transforms an arbitrary reference image into a manufacturable MWCA design. The pipeline begins with image simplification to minimize color count, then applies a connectivity-aware greedy algorithm to extract a set of wooden sheets. Post-processing merges visually insignificant layers, hollows fully occluded regions, and widens structurally weak members. These steps lower sheet count, machine time, and off-cut waste, thereby

[*] Corresponding author.
E-mail address: kang910042009@gmail.com (K. Wu).

https://doi.org/10.1016/j.cag.2025.104337
Received 26 June 2025; Accepted 20 July 2025
Available online 6 August 2025

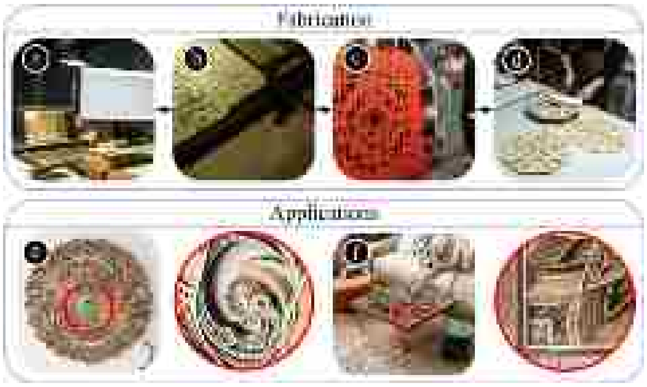

Fig. 1. Production of MWCA provided by STEREOWOOD [1]: (a) Designing laser-cutting paths; (b) Laser cutting; (c) Painting; (d) Gluing; (e) Interior decoration; (f) Furniture design.

reducing fabrication expenses; they also trim overall mass and stack height, cutting packaging volume and freight charges. The resulting sheets can be laser-cut, stacked, and bonded with minimal manual alignment. Evaluated on 18 digital examples, the system consistently produces MWCAs with few layers, low visual error, robust structural performance, and demonstrable savings in both manufacturing and transport costs—highlighting its practicality for scalable, cost-effective mass customization of MWCA pieces.

2. Related work

Personalized fabrication. Personalized fabrication refers to producing mass and highly customized physical objects through flexible design–manufacturing systems [13,14]. Personalized fabrication has become increasingly important in the art market, as contemporary consumers seek uniquely tailored works that reflect their individual tastes and preferences. To satisfy consumers' needs, scalable design methodologies, streamlined production workflows, and automatic fabrication are required [3]. In this context, recent researchers have developed various computational methods. Iseringhausen et al. (2020) develop computational parquetry, a style-transfer framework that selects and arranges natural wood veneer patches under resource constraints to approximate target photographs [5]. Zhu et al. (2024) derive machine-knittable illusion patterns by modeling microgeometry constraints and automating design tasks in an interactive tool [8]. Fukushima et al. (2022) propose a method for automatic multi-layer stencil generation of overpaint [7]. Due to distinct artistic forms, the design constraints of their work and ours differ substantially. Specifically, while both the overpaint method and our MWCA approach generate a sequence of connected layers with color, they fundamentally diverge in color application principles and constraints: Overpaint applies color through stencil holes, focusing solely on hole-layer occlusion orders to achieve image reproduction. In contrast, our method applies color directly onto the solid regions, focusing on both solid-layer and hole-layer occlusion orders to MWCA reconstruction. Moreover, some researchers focus on personalized fabrication of LEGO art, including micro buildings [10], technic assemblies [9], sketches [11], and figurines [15]. These approaches have demonstrated success in both academic research and commercial markets, providing strong motivation to explore personalized fabrication for MWCA.

Path-generation for artworks. The personalized fabrication of MWCA highly depends on advanced path-generation techniques to translate a preferred image into optimized cutting sequences. Dewil et al. (2016) provide a comprehensive review of laser-cutting path planning—covering contour-parallel offsets, task-sequencing formulations (TSP/GTSP), and heuristic/metaheuristic methods under thermal and kinematic constraints [16]. Based on these methods, Liu et al. (2019) propose computational peeling art design, which maps desired 2D shapes onto citrus peels and computes optimal cut paths [4], while Roumen et al. (2021) propose Assembler[3], which converts between 2D laser-cut plans and 3D parametric models to refine complex shapes up to ten times faster [17]. More recently, Wang et al. (2025) developed HarmonyCut, an interactive system for creative Chinese paper-cutting. This tool can generate cutting paths for laser cutters, allowing designers to prototype intricate hollow-work motifs aligned with cultural heritage practices [6].

However, despite these advances, computational methods specifically tailored to address the constraints unique to MWCA remain underexplored. Therefore, this study focuses on the generation of a cutting path for MWCA, aiming to reduce the necessity of manual processes and artistic expertise.

Image vectorization. Image vectorization can transform raster images into scalable geometric representations – such as Bézier curves, line segments, and filled polygons – whose explicit contours can be exported directly as paths for laser cutters. According to Tian and Günther [18], existing image vectorization mainly uses mesh-based and curve-based strategies. Both of these strategies start from detecting edges and corner features. Then, mesh-based methods perform patch fitting or triangulation to place patch vertices [19,20], whereas curve-based methods fit parametric or spline curves to the detected features [21,22]. To minimize reconstruction error, the two methods refine the geometry through remeshing or shape-optimization procedures, respectively [23,24]. However, mesh-based approaches often struggle to depict sharp chromatic discontinuities, and poorly chosen patch topologies can leave visible seams that require extensive manual cleanup. Conversely, curve-based techniques rely on solving global PDEs over the entire image domain, which hampers real-time performance and demands dense curve networks to control smooth gradients, leading to large data sizes and limited local editability.

In our method, we use a curve-based method, called layer-wise image vectorization (LIVE) [25]. This method employs a sequential layer-building strategy, iteratively incorporating geometric primitives onto the fabrication canvas. Following each incremental addition, a gradient-based refinement phase optimizes both current and preceding layer parameters to minimize multi-layer interference effects. In this study, we integrate this method into our pre-processing strategy to provide an initial image with a finite number of colors.

3. Method

3.1. Overview

To address the design constraints of MWCA mentioned in Section 1, we first establish an optimization model to describe this design problem. The optimization model is then solved by a novel heuristic algorithm.

Our optimization model takes a reference image as input and outputs a low-cost MWCA that visually resembles the input while adhering to design constraints, including connectivity, manufacturability, and structural stability (Section 3.2).

Our heuristic algorithm has four steps, as shown in Fig. 3. Step 1 (Section 3.3): We propose a pre-processing strategy to reduce the number of colors in the input image. This strategy includes the LIVE method mentioned in Section 2 and the well-known K-means clustering method [26]. It should be noted that the reduced number of colors can decrease the number of wooden sheets, thereby saving the costs of MWCA. Step 2 (Section 3.4): We propose a greedy iterative generation method to create a visually feasible MWCA solution. The solution includes a set of images that represent each layer of MWCA. However, these generated images do not ensure the structural stability of the final product. Step 3 (Section 3.5): We introduce a post-processing strategy

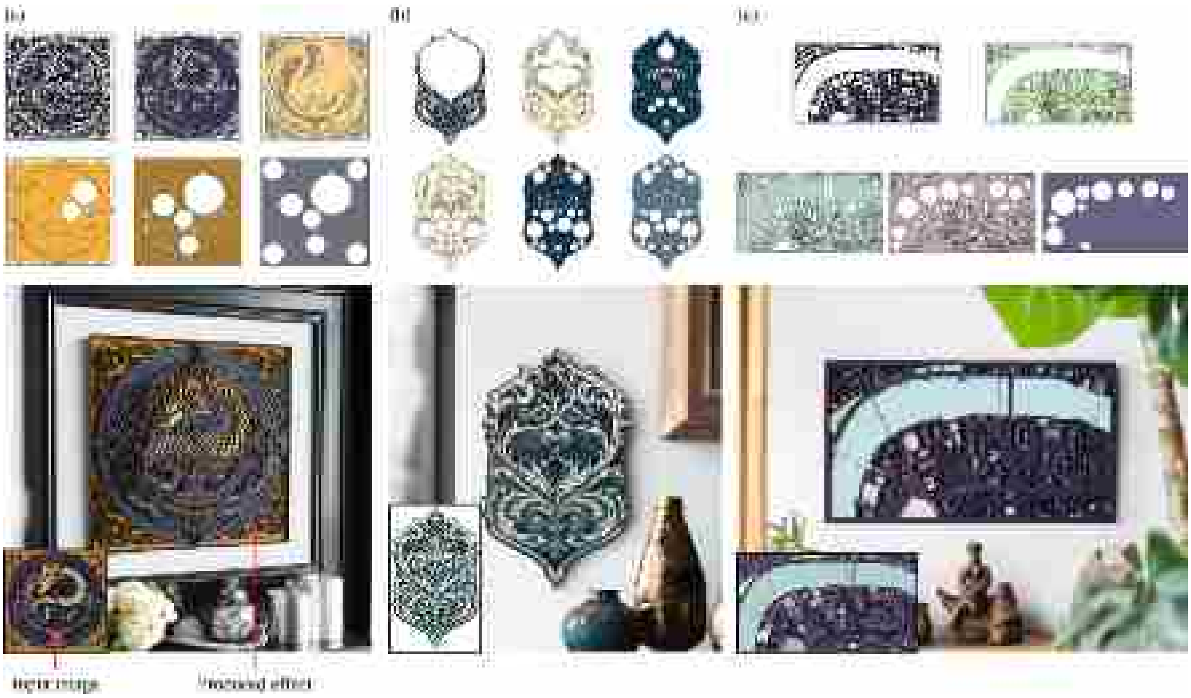

Fig. 2. Top row: All individual wood layers. Bottom row: Three physical examples designed using the proposed method. The input images are created by the well-known generative artificial intelligence, Midjourney [12].

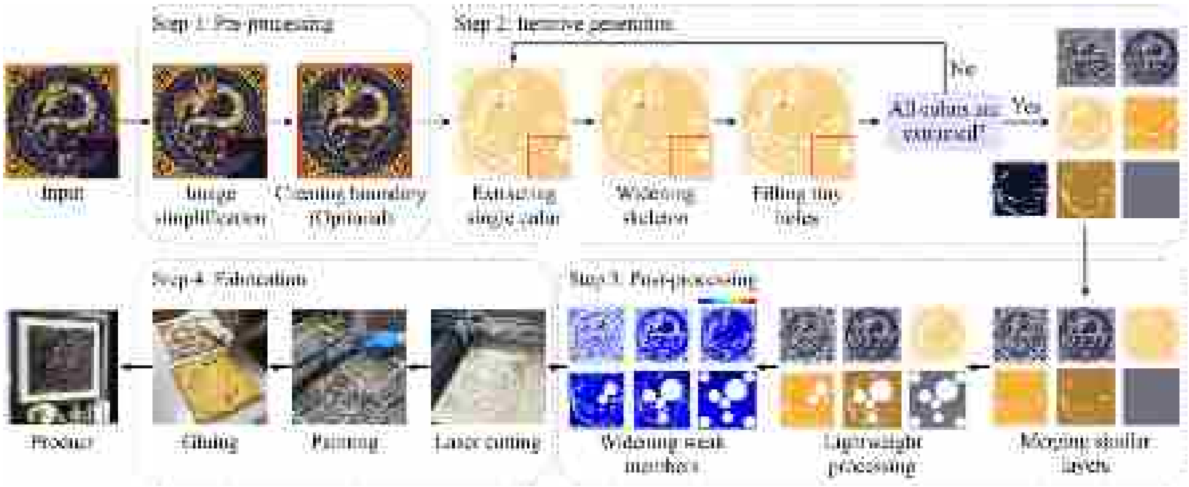

Fig. 3. The pipeline of the proposed heuristic algorithm.

to further reduce production costs and improve structural stability. The strategy includes merging similar layers, lightweight processing, and widening weak members. After post-processing, the obtained result is available to be fabricated. Step 4 (Section 3.6): We use a laser cutter to physically fabricate the generated result to finally obtain a product. More details about these four steps are shown in Sections 3.3– 3.6

3.2. Problem and formulation

Optimization model. The input of our optimization model is a user-specified reference image $\mathcal{I} \in \mathbb{R}^{W\times H\times 3}$, where W (width) and H (height) denote pixel dimensions, and 3 corresponds to RGB color channels. The optimization goal is to generate an MWCA of K layers with identical thickness t, that achieves two objectives: the rendered appearance of the MWCA must closely match the input reference image, and the production and transportation costs must be minimized (see Fig. 4).

We use a set of ordered pair $\{I_i = (C_i, M_i)\}_{i=0}^{K-1}$ to represent the stacked wooden sheets in the MWCA, where $C_i \in \mathbb{R}^3$ is the RGB color of the ith layer and $M_i \in \{0,1\}^{W\times H}$ is binary piercing mask of the ith layer. To clearly introduce our method, the stacking order is defined from front to back (i.e., the ith layer occludes the $i+1$th layer). For a pixel $v \in \mathbb{Z}_+^2$ with coordinates (x, y) in the M_i, $M_i(v) = 0$ and $M_i(v) = 1$ denote the pixel is void and solid, respectively.

The rendered appearance of the MWCA refers to the simulated visual effect of stacked MWCA layers by a rendering model. We use a simplified rendering model that only considers the occlusion relationship since the focus of our problem lies in the occlusion relationship. Formally, the rendered image of an MWCA is:

$$\mathcal{I}_{\text{Render}}(v) = \sum_{k=0}^{K-1} C_k M_k(v) \prod_{i=0}^{k-1} (1 - M_i(v))$$

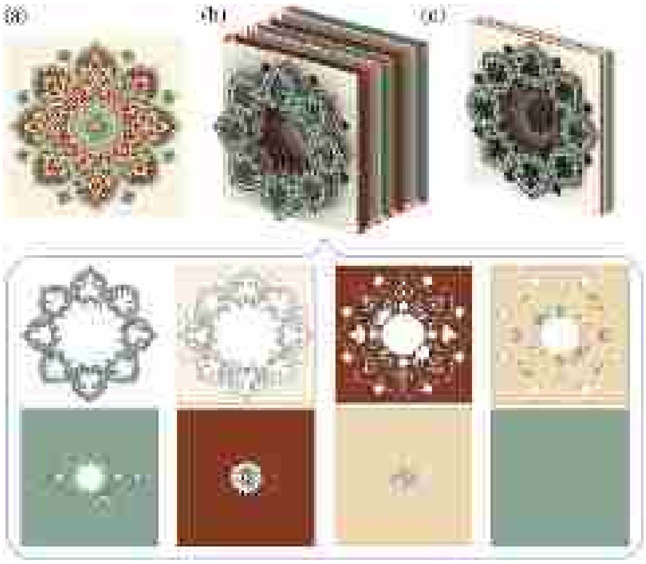

Fig. 4. Problem definition: (a) For an input image (created by GPT-4o [27]), (b) our goal is to generate a set of wooden sheets represented by images I_i, (c) when we stack them together in order, the result is similar to the input image.

The term $\prod_{i=0}^{k-1}(1 - M_i(v))$ describes this occlusion relationship: For a given pixel v_0, if k_0 is the first k such that $M_k(v_0) = 1$, then the rendered image's color at pixel v_0 is C_{i_0} (i.e., $\mathcal{I}_{\text{Render}}(v_0) = C_{i_0}$), since for $k < i_0$, the term $M_k(v_0) = 0$, for $k > i_0$, the term$\prod_{i=0}^{k-1}(1 - M_i(v_0)) = 0$, and only for $k = k_0$,term $M_k(v_0) \prod_{i=0}^{k-1}(1 - M_i(v_0)) = 1$. Since the rendered image serves as a proxy for the MWCA's physical appearance, minimizing the difference between the rendered image and the reference image ensures the physical MWCA closely matches the target visual effect. Therefore, we use a pixel-wise L_2 reconstruction loss $\mathcal{L}_{\text{loss}}$:

$$\mathcal{L}_{\text{loss}} = \sum_{v \in \Omega} \left\| \mathcal{I}(v) - \mathcal{I}_{\text{Render}}(v) \right\|_2$$

to measure the difference between the rendered image and the reference image, where Ω denotes the set of all pixel coordinates in the image.

The production of each layer in MWCA requires an individual disposable wooden sheet. A higher number of layers K can significantly increase the wooden sheet consumption and manual assembly time. To save the production cost of MWCA, K should be reduced as much as possible during the optimization. The transportation cost is positively correlated with the total weight of MWCA. For the ith layer, $\sum_{p \in \Omega} M_i(p)$ is the area of the solid region. When material density and layer thickness are invariant across all wooden sheets, the total weight is linearly proportional to the cumulative solid region area. Thus we use the total weight $\sum_{i=0}^{K-1} \sum_{p \in \Omega} M_i(p)$ to measure the transportation cost. The total cost is defined as

$$C_{\text{total}} = \lambda_1 K + \lambda_2 \sum_{i=0}^{K-1} \sum_{p \in \Omega} M_i(p)$$

where λ_1 and λ_2 are weighting coefficients balancing the relative importance between production and transportation costs.

After determining two optimization targets, we convert the design constraints mentioned in Section 1 into optimization constraints:

1. Connectivity: Each $\{v \mid M_i(v) = 1\}$ should be connected; disconnected components would fragment during laser cutting, greatly increasing the complexity of assembly due to manual repositioning of the disjoint wood pieces.
2. Manufacturability: The void regions $\{v \mid M_i(v) = 0\}$ are the holes in the fabricated wooden sheets. Their size γ should exceed the minimal fabrication size δ. Similarly, for the solid region $\{v \mid M_i(v) = 1\}$, the minimal width of structural members should be larger than δ.
3. Structural stability: The solid regions $\{v \mid M_i(v) = 1\}$ must maintain structural stability under the gravity and fixed boundaries. In detail, the maximum stress should remain below a threshold β, thereby preventing stress concentration-induced fractures during MWCA handling or transportation.

Together, the optimization problem is formulated as follows:

$$\min_{K,\{C_i\},\{M_i\}} \mathcal{L}_{\text{loss}} + \lambda C_{\text{total}}$$

subject to the following manufacturing constraints:

1. Connectivity: Each solid region in M_i must be connected (no disjoint components).
2. Manufacturability: Holes diameter $\geq \gamma$ and solid region width $\geq \delta$ (minimum feature size).
3. Structural stability: The maximum stress $\leq \beta$ under the gravity and fixed boundaries.

where λ is a weight parameter to balance the importance between L_{loss} and C_{total}. The parameters γ (minimum hole diameter) and δ (minimum solid region width) are manufacturing constraints specific to our fabrication equipment, rather than tunable algorithmic hyperparameters. According to our practical limits [1], γ is fixed at 0.5 mm – the minimum hole size our laser cutter can reliably produce – while δ is set to 2 mm for 3 mm thick plates and 1.5 mm for 1.5 mm thick plates to prevent structural fracture during or after fabrication.

This optimization problem is challenging. First, the number of layers K is unknown, which leads to an indefinite number of optimization variables. Moreover, each layer's binary mask introduces a huge number of combinations that standard gradient methods cannot handle. Therefore, a heuristic algorithm is needed to achieve the optimization target.

Key insight. The proposed heuristic algorithm can iteratively generate all layers of an MWCA. The key insight is: If the connected MWCA (despite other constraints) that is relatively approximate to the input image has already been generated, then dealing with other constraints and goals later is relatively simple; the occluded regions can be reused in the following layer to increase the possibility of connection.

3.3. Step 1: Pre-processing

Since each layer is assigned a single color, to make $\mathcal{L}_{\text{loss}} = 0$, the minimum required layer count K equals the number of distinct colors in the input image, but the colors of a natural image are countless. To save material costs, it is necessary to reduce the number of colors to decrease the use of wooden sheets. In addition, our method relies on a high-resolution input image to approximate the input image as much as possible. Therefore, our algorithm applies K-means and LIVE method to transform a low-resolution input $\mathcal{I}$ into a high-resolution image with piece-wise constant colors $\hat{\mathcal{I}} = \{p_i^k\}_{\{i,k\}}$, where p_i^k refers to the ith patch (connected component) under the kth color (e.g. for the 2nd color in $\hat{\mathcal{I}}$, the 3rd patches can be represented as p_3^2).

Specifically, we employ the LIVE method to vectorize the input image, rasterize the vector image into a high-resolution bitmap, and quantize the color space via K-means clustering with an input parameter color count N_c (Discussed in Section 4).

3.4. Step 2: Iterative generation

After obtaining piece-wise constant $\hat{\mathcal{I}} = \{p_i^k\}$, our next goal is to construct a set of connected and solid color images $\{I_i = (C_i, M_i)\}_{i=0}^{K-1}$ that are approximate to $\hat{\mathcal{I}}$ when they stack up in order. The reconstruction

Algorithm 1: Generating $\{I_i\}_{i=0}^{K-1}$

```
Input : the image with piece-wise constant color $\hat{\mathcal{I}}$, the
        Constructing threshold $\omega$, the Widening threshold $\delta$,
        the Filling holes threshold $\gamma$
Output: $\{I_i\}_{i=0}^{n-1}$
$I_{-1} \leftarrow$ Initialization($\hat{\mathcal{I}}$); $j \leftarrow 0$;
while Any area in $\hat{\mathcal{I}}$ is not corresponded in $\{I_i\}_{i=0}^{j-1}$. do
    $I_j \leftarrow$ ContructRaw($\hat{\mathcal{I}}, I_{j-1}, \omega$)
    $I_j \leftarrow$ Widen($I_j, \delta$)
    $I_j \leftarrow$ FillHoles($I_j, \gamma$)
    $j \leftarrow j + 1$;
end
```

loss $\mathcal{L}_{\text{loss}}$ comes from two aspects, Pre-processing, and the following step.

$$\begin{aligned}\mathcal{L}_{loss} &= \sum_{v\in\Omega} \|\mathcal{I}(v) - \mathcal{I}_{\text{Render}}(v)\|_2 \\ &\leq \sum_{v\in\Omega} \left(\|\mathcal{I}(v) - \hat{\mathcal{I}}(v)\|_2 + \|\hat{\mathcal{I}}(v) - \mathcal{I}_{\text{Render}}(v)\|_2\right)\end{aligned}$$

Notice that the subsequent step of our algorithm only is to reduce the term $\|\hat{\mathcal{I}}(v) - \mathcal{I}_{\text{Render}}(v)\|_2$.

Our algorithm is based on a key observation: regions occluded by preceding layers $\cup_{i=0}^{j-1} M_i$ (The operator ∪ means pixel-wise logical OR between binary masks) can be strategically reused in subsequent layer region M_j to bridge otherwise disconnected same color patches and C_j is the color of those patches. Thus, beginning with the top layer, in each iteration, our algorithm consists of the following steps:

1. Constructing a connected M_j having the most neighbors with a color C_j from I_{j-1} as the raw I_j (Extracting single color).
2. Widening M_j to satisfy the manufacturability (Widening skeleton).
3. Filling the holes whose areas are too small to be fabricated in M_j to maintain manufacturability (Filling tiny holes).

When constructing the first layer $I_0 = (M_0, C_0)$, we set $I_{-1} = (M_{-1}, C_{-1})$ to perform the above method, where C_{-1} is not important and $M_{-1} = \emptyset$ or be a optional photo frame. A photo frame implies that two spatially disconnected regions, both adjacent to the image boundary, can be computationally unified as a single connected component. We terminate the iteration until all pixels (or the pixels in the foreground part of $\hat{\mathcal{I}}$) of M_j is 1, since in this case, each part of the image $\hat{\mathcal{I}}$ will correspond to the exposed part of a certain layer. After the iteration is terminated, the number of layers K is determined. In the post-processing part, we will further reduce the number of layers by merging similar layers. Alg. 1 shows our workflow. The following sections detail the three-stage implementation procedure sequentially.

Extracting single color. Assuming $p = \cup_{i=0}^{j-1} M_i$ is the region occluded by preceding layers and $p^{\complement} = 1 - p$ is the exposed region. We need to select several patches with the same color $\{p_i^k\}_i$ from the exposed region $p^{\complement}$ to construct a connected region as raw $I_j = (M_j, C_j)$ by reusing p. For a given kth color, some patches are the neighbors of p, which means those patches can be connected by p, and some patches are not. We only need to handle those unconnected patches. There are two options: deferral of them to subsequent iterations, or connecting them with p by some lines. The deferral operator will increase the number of layers K, but the connecting operator will increase the reconstruction loss. To handle the balance between the cost and loss, we define the distance between two regions p_1 and p_2 as the L_2-norm of the shortest connecting line segment, and we perform the following steps to construct I_j:

1. For each color k, we greedily connect $\{p_i^k\}_i$ to p. Starting with $p_0' = p$, let $p_{i_0}^k$ be the color k patch closest to p_0', then connect p_0' and $p_{i_0}^k$ to form an updated region p_1'. In the subsequent iteration, the nearest color-k patch $p_{i_1}^k$ to p_1' is selected and connected to p_1', yielding p_2'. This process repeats iterates until the cumulative connection distance of p_{m+1}' exceeds a threshold ω ($\omega = 3\%$ of the diagonal, in our experiments), at which point the final region p_m' is designated as $p^{(k)}$.
2. Compute the number of neighbor patches for all $p^{(k)}$, and choose the color l that $p^{(l)}$ has the largest number of neighbor patches for minimizing the number of layers K, then we set $M_j = p^{(l)}$ and C_j is the lth color to construct raw $I_j = (M_j, C_j)$.

An example of the iteration process is shown in Fig. 5.

Widening skeleton. After getting the raw I_j, our algorithm applies the Zhang-Suen Thinning algorithm to generate the skeleton of M_j. And then we use morphological dilation to dilate the skeleton to a threshold δ ($\delta = 0.2\%$ of the diagonal, in our experiments) to satisfy the manufacturability constraint. Specifically, we take the union of the dilated skeleton structure with M_j (see Fig. 6).

Filling tiny holes. The widened I_i will still have some small holes that cannot be manufactured. So our method detects the holes of M_j whose areas are below γ ($\gamma = 0.01\%$ of the image area, in our experiments) and fills them to make sure our result is manufacturable. The detection method is to calculate all the connected components of $\{v | M_j(v) = 0\}$ and assign a value of 1 to those holes whose area is less than the threshold γ (see Fig. 7).

3.5. Step 3: Post-processing

In the third step, we merge similar layers to save production costs further. Then, the total weight of the MWCA is reduced by lightweight processing. Finally, we conduct a structural analysis and widen the weak members on each layer to achieve a structurally stable MWCA.

Merging similar layers. Merging those layers whose visible region is small will only slightly increase the reconstruction loss, but will reduce the number of layers and lower the weights. Therefore, we merge the layers whose visible region is below a threshold ϵ ($\epsilon = 1.0\%$ of the image area, in our experiments) into their upper layer to achieve a good trade-off between reconstruction loss and the total cost.

Lightweight process. Our algorithm considers the region occupied by the upper layer $p = \cup_{i=0}^{j-1} M_i$ as part of the jth layer ($p \subset M_j$). This leads to our results being too heavy since the region occupied by p in the jth layer is not visible, and some parts of it can be removed as long as the connectivity constraint is not violated. Therefore, our algorithm removes this redundant material to further reduce the weight C_{trans} of our result (see Fig. 8).

This algorithm for reducing materials is very simple. First, calculate the invisible region, which is generally the upper layer region $p = \cup_{i=0}^{j-1} M_i$. Then find the point farthest from the edge in this region, and record this point as v and the distance as d. If d is greater than a certain value (7% of the diagonal, in our experiments), we will make a hole in the area with $d - r$ as the radius with this point as the center. The r is a protective distance(1% of the diagonal, in our experiments), which is to maintain connectivity and stability of the structure.

Widening weak members. For the structural stability constraint, we calculate the stress distribution of the solid region of $\{M_i\}_{i=0}^{n-1}$ under the gravity and fixed boundaries. The material parameters are set with a density of 500 kg/m^3, Young's modulus is 7.2 GPa, and Poisson's ratio is 0.3. We then widen the areas where stress is concentrated until the stress levels are uniformly below a specified threshold β ($\beta = 20$ MPa, in our experiments). Fig. 9 shows the stress distribution before and after our widening operation.

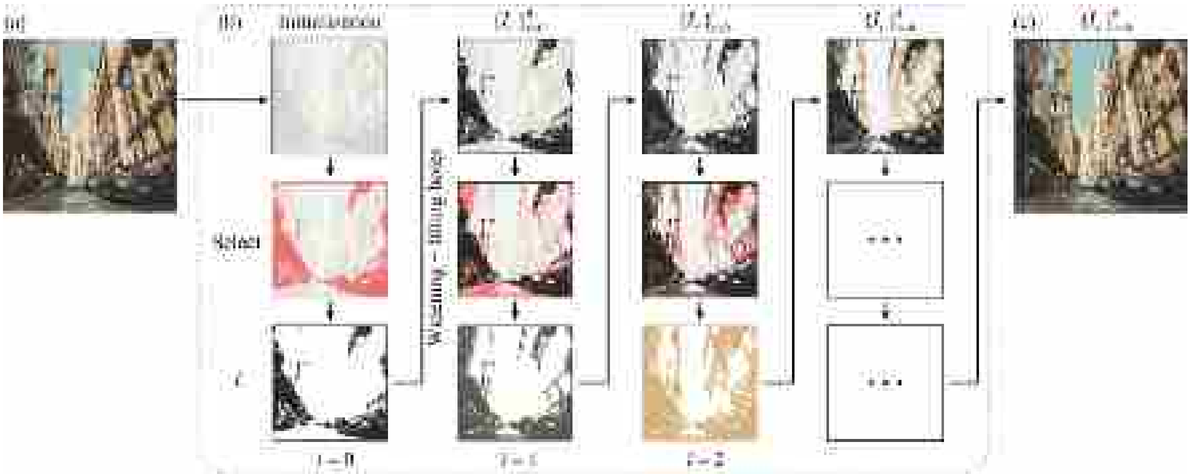

Fig. 5. Iteratively contructing raw $\{I_j\}$: (a) Given an image with piece-wise constant colors $\hat{\mathcal{I}}$ (taken by the authors); (b) Row 1: the previous layer is overlaid on the $\hat{\mathcal{I}}$; row 2: the selected area connected to the upper layer by our method; row 3: The current layer; (c) The images $\{I_j\}_{i=0}^{9}$ stacking up in order.

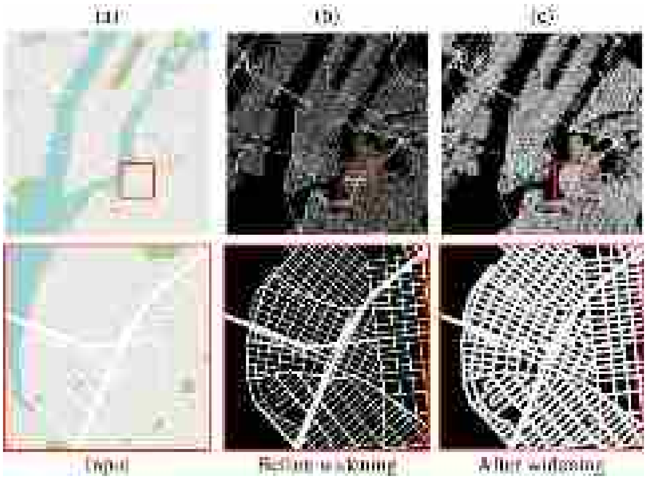

Fig. 6. An example of the widening method: (a) Input image; (b) Raw I_j (the black background is used to show the widening effect clearly); (c) Widened I_j.

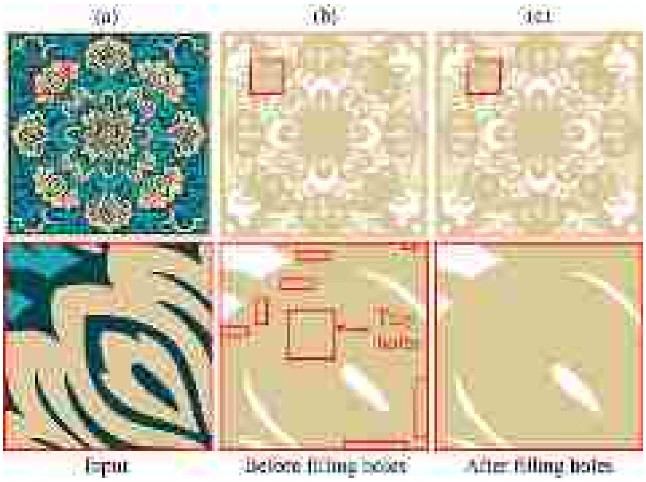

Fig. 7. An example of the filling holes method: (a) Input image; (b) The raw I_j has many tiny holes; (c) The result after applying the filling holes method.

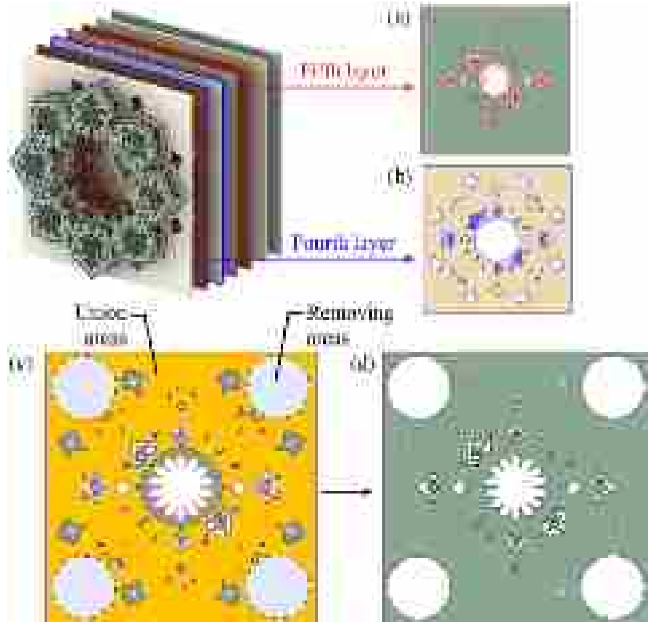

Fig. 8. An example of the lightweight process: (a) Base layer (red) to be lightweighted; (b) Upper layer (blue) to find removable areas; (c) Generate the circular removable areas on the union areas (orange) between the base and its upper layers. (d) Lightweighted base layer.

3.6. Step 4: Fabrication

After obtaining $\{I_i\}_{i=0}^{n-1}$, we perform a contour extraction algorithm [28] to extract the boundaries of each layer's solid region and vectorize [29] them to generate the smooth cut path for the laser cutting machine. Each layer is generated from a 400 × 400 × 3 mm wooden sheet by a CO_2 laser tube with 100 watt power. We then spray-painted these processed sheets to color them. Since each layer contains the frame of the input image, it is easy to align them. We then use glue to fix the layers together.

4. Experiments and evaluations

We conducted 18 digital simulation trials and selected three representative cases for physical fabrication to validate the proposed method. The investigation focuses on the generation effect of changing

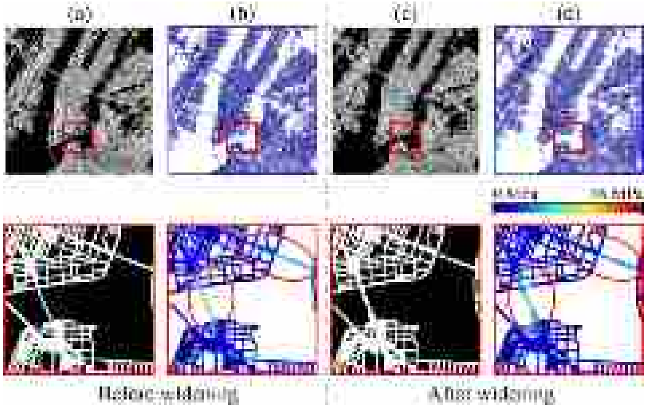

Fig. 9. Stress analysis: (a) One layer we construct after our iterative greedy method (black areas are the background); (b) The stress analysis result of this layer shows that some areas are subjected to large stress; (c) The result after we widen the areas subjected to large stress; (d) The stress analysis result after widening.

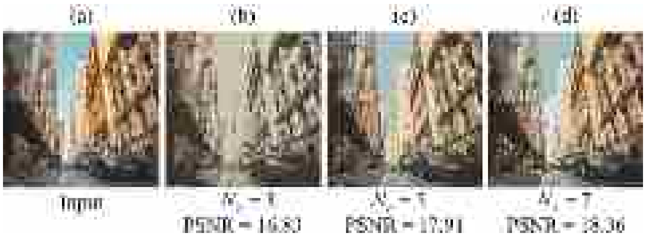

Fig. 10. Comparative experiment with different numbers of colors: (a) Input image; (b) Result with 3 colors; (c) Result with 5 colors; (d) Result with 7 colors.

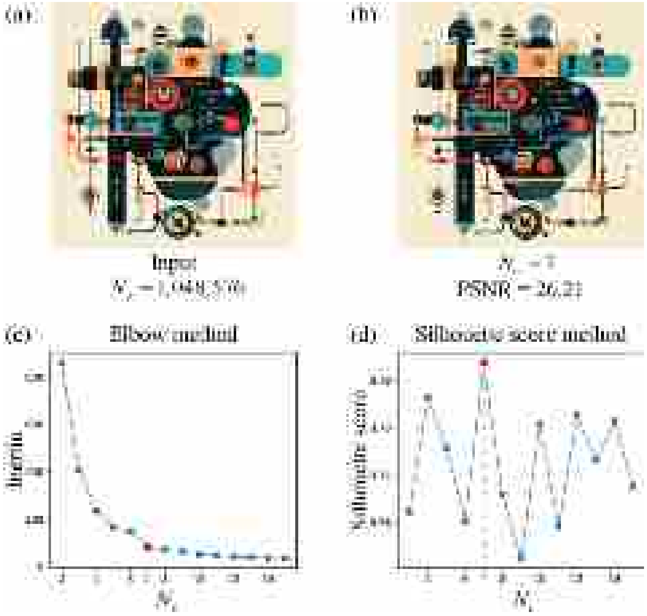

Fig. 11. Determination of the optimal N_c using the elbow method and the silhouette score method: (a) Input image with 1,048,576 colors; (b) Optimal result with 7 colors; (c) The elbow method; (d) The silhouette score method.

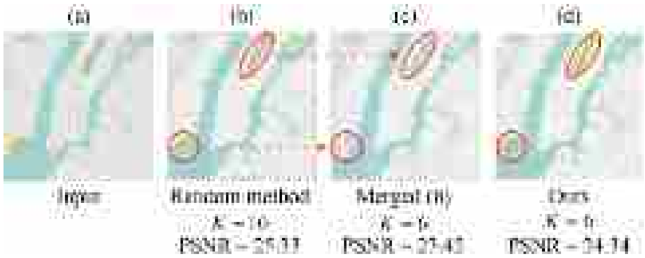

Fig. 12. Comparative experiment with a random method: (a) Input image (created by Snazzy Maps [30]); (b) Result of the random method; (c) Merged result of the random method; (d) Result of our method.

input parameters, including the number of colors N_c, the order of layers, image resolution, and image styles. To evaluate the generation effect, we use the peak signal-to-noise ratio (PSNR), which can quantify the reconstructed image's fidelity to the input (higher means better). Moreover, the running time R_T of each example is also provided to validate the efficiency of our method.

The proposed method is implemented in Python, and all experiments are conducted on a laptop equipped with a 13th Gen Intel® Core™ i7-13700H processor (2.40 GHz) and 16 GB RAM.

The number of colors. In our algorithm, the number of colors N_c should be determined in the pre-processing. In this experiment, we discussed the selection of the number of colors to make the $\sum_{v\in\Omega} \|\mathcal{I}(v) - \hat{\mathcal{I}}(v)\|$ as low as possible and N_c as few as possible. Fig. 10 shows the influence of the number of colors on our results. With the increase in the number of colors, the Peak Signal-to-Noise Ratio (PSNR) between our result I^s and the input image $\mathcal{I}$ also increases. Meanwhile, a larger number of colors results in more layers. In our experiment, the determination of color quantity is guided by a dual-criteria analytical framework. We first convert the input image from RGB space into CIELAB space to align with human visual perception characteristics. Then we executed K-means clustering across a systematically defined candidate range (3 to 15 clusters in our experiments), iteratively computing both the within-cluster sum of squared errors (WCSS) for elbow analysis and the global silhouette score for cluster validity assessment. After that, we identify the critical inflection point of the elbow detection, where the WCSS reduction rate transitioned from quadratic to linear decay patterns, by using piecewise linear regression (see Fig. 11(c)). The silhouette validation enforced a minimum threshold of 0.65 to guarantee inter-cluster discriminability (see Fig. 11(d)). The final color count is then selected as the minimal k-value satisfying both the elbow position constraint and the silhouette quality requirement. Experimental results demonstrate that our color count determination framework achieves a balance between loss $\sum_{v\in\Omega} \|\mathcal{I}(v) - \hat{\mathcal{I}}(v)\|$ and colors count N_c.

The order of each colored layer. We propose an iterative greedy method to determine the color of each layer to decrease the number of layers in our final results. To prove the effectiveness of our method, we perform a comparative experiment with a method that randomly selects each layer color. For the sake of fairness, only the color selection part is different; the other parts of the algorithms remain the same. As shown in Fig. 12, although the result of the random selecting color method has a larger PSNR, its number of layers is also larger than ours. If we perform the merging layers step to decrease its number of layers to the same as ours, its PSNR becomes smaller than ours. In general, our method can obtain a better visual effect with fewer layers.

Different image resolution. We use the images with three different resolutions as inputs for our algorithm. Fig. 13 shows the results. Increasing the resolution of images we process in our algorithm can improve the quality of our results. But the time required for our algorithm's operation also increases exponentially Generally speaking, we suggest that users use images with a resolution of 2000 × 2000 if they want to get a high-quality result and images with a resolution of 1000 × 1000 if they prefer the efficiency.

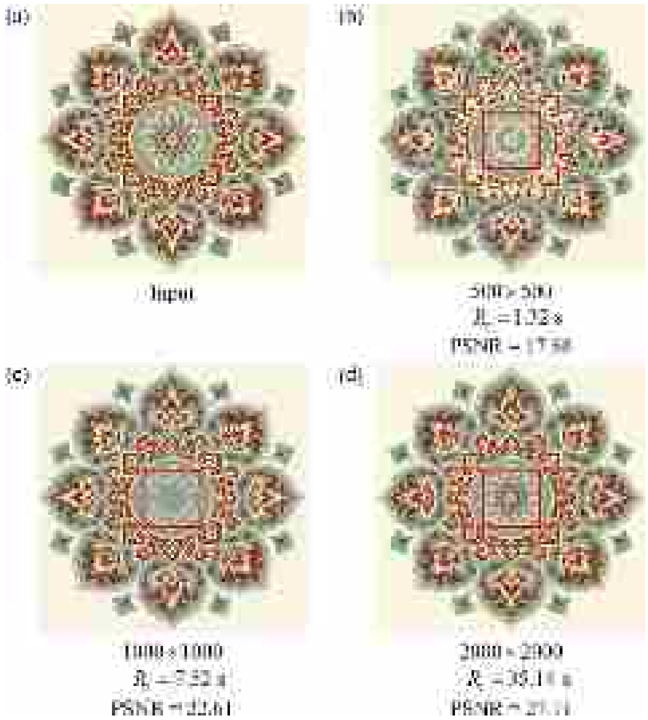

Fig. 13. Comparative experiment with different resolutions: (a) Input image (created by GPT-4o [27]); (b) 500 × 500; (c) 1000 × 1000; (d) 2000 × 2000.

Different image style. Our algorithm can be combined with image style transfer algorithms to enhance the diversity of our results. Fig. 14 shows the results with different image styles. We apply the generative AI models (GPT-4o/Midjourney) to transfer the original image to different image styles, such as illustration, oil painting, and cubism, while keeping the same content as the original images. With image style transfer, our algorithm can better display the content of the input image, which will improve the generality of our algorithm.

Manual–computational design comparison. Our computational approach differs fundamentally from traditional craftsmanship, where artists freely design layered artworks through experiential drawing without strict fidelity to references, inherently embedding layering decisions during creation. By contrast, our algorithm decomposes arbitrary input images into manufacturable layers. Given this methodological divergence, direct comparisons are invalid. Instead, we evaluated decomposition performance using identical hand-drawn artworks (created by master craftsmen with layer planning) as input for both approaches. Results demonstrate our method achieves layer decomposition in 30 s–60× faster than manual processing (30 min)–while reducing total assembly weight by 20.6% through optimized material distribution, maintaining full structural integrity (see Fig. 15).

Gallery. Our algorithm is applied to 18 different images. All of them achieve a favorable trade-off between visual errors and the number of layers. Fig. 16 shows 10 of them. Noticed that Fig. 16(j) is a borderless image. To handle this kind of image, we expand the image's boundary to a certain width to serve as the border so that our algorithm can deal with it. To demonstrate the practicality and feasibility of our algorithm, we also use a laser cutter to fabricate a result, which is shown in Fig. 2.

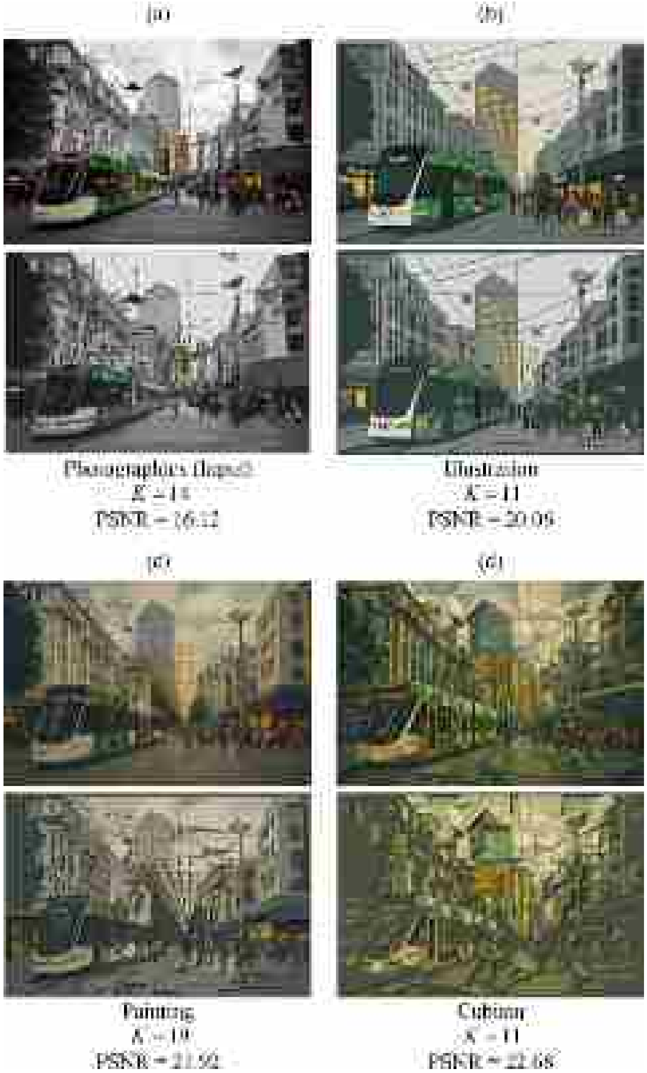

Fig. 14. Comparative experiment with different image styles. The above is an input image, and the below is the output MWCA. (a) Photograph, which serves as the input image for the style transformation of Figures (b–d) through GPT-4o; (b) Illustration; (c) Painting; (d) Cubism.

5. Discussion

Stress optimization. Although our algorithm runs a stress analysis step in post-processing to improve the structural stability performance of our results, we just simply widen the areas where the stress level is above β. In some extreme cases, this trivial approach may lead to local overwidth, which produces a result with high visual error. One possible idea is to add connection structures at other locations based on stress analysis results, which is a worthwhile direction for future research.

Lightweighting strategy. The current lightweighting strategy is deliberately straightforward: circular holes are drilled exclusively in regions that are hidden beneath the preceding layer—areas that remain invisible in the final assembly. A promising avenue for future work would be to apply topology optimization methods [31] to remove additional redundant material while safeguarding structural performance. Nonetheless, preliminary assessments indicate that such techniques would yield only marginal extra weight savings for the actual product

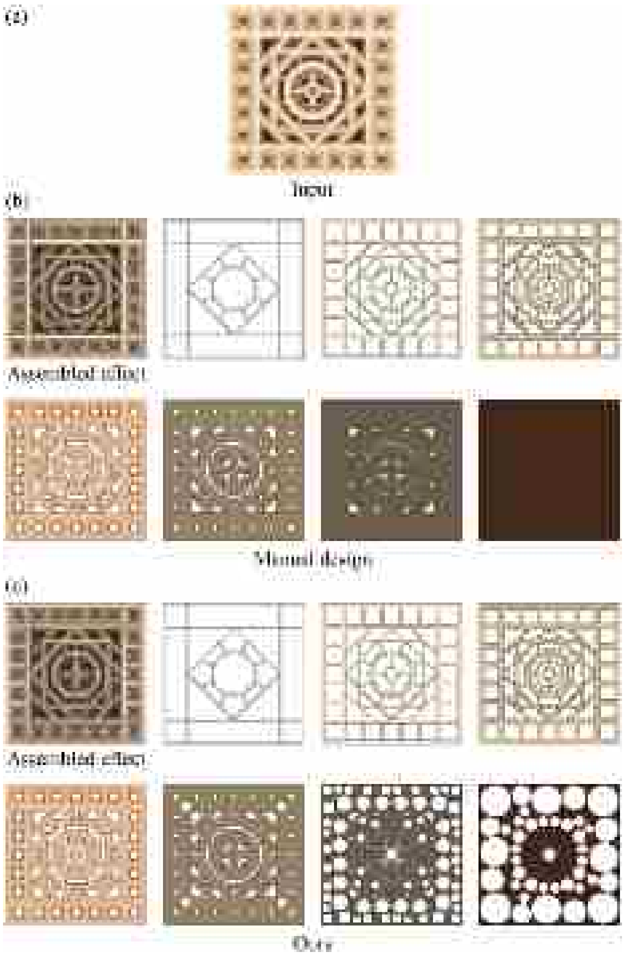

Fig. 15. Comparative experiment with manual design and algorithm design. (a) The input image, which is a manually designed image created by the artist [1]; (b) Manual design with all layers; (c) Ours.

while incurring a markedly higher computational cost. Consequently, topology optimization was not adopted in this study.

The way to connect isolated patches. In order to decrease the number of layers, we need to connect some isolated patches into a connected area. Our algorithm uses the shortest line between two patches to connect them. Although our algorithm avoids connecting the patches that are too far apart, using lines to connect the patches still reduces users' viewing experience. In the future, how to utilize human perception and psychology to generate the connecting paths between patches that do not disrupt the integrity of the content of the input is a worthwhile problem.

Stereo effect. The MWCA is a 2.5D artwork, which means it can showcase 3D effects to a certain extent. However, our algorithm does not take this point as the goal. Indeed, enhancing the 3D effects of the final result will no doubt increase the number of layers. But proposing a method to achieve a good trade-off between them is still a worthy idea.

Effect of different artistic styles. Our outcomes are inevitably influenced by the artistic style of the input image. Illustrations and maps – whose visuals are dominated by large, uniform color patches – tend to yield more satisfactory carvings (e.g. Figs. 16(a–g)). By contrast, photographic pictures rich in color gradients and fine details often fall short of expectations (Figs. 16(i) and (j)). Image vectorization struggles to convert gradient colors into the desired discrete uniform color patches, as this can introduce noticeable distortion. Therefore, it can be understood that the poor expression of gradient colors is an inherent limitation of MWCA. A promising remedy is to first employ an artificial intelligence-driven style-transfer step that converts complex photographs into an illustration before feeding them into our pipeline.

Potential interactive design. The proposed workflow lends itself to a highly marketable software solution centered on interactive design. During preprocessing, user-friendly tools – such as a paint-bucket tool, brushes, and even artificial intelligence-based image-generation modules – could be introduced to let customers rapidly refine their input images to match personal tastes. In post-processing, intuitive controls allow them to choose whether to include a frame and to adjust lightweighting parameters – e.g., the minimum hole size and protective offset distance – to further reduce the artwork's overall mass and production cost. Moreover, the unpainted, natural-wood variant is also visually compelling (see Fig. 16). Our future interactive design will allow users to decide whether to keep the raw version of MWCA. Taken together, these features enable true mass customization, empowering users with no artistic background to transform any desired picture into a physical MWCA.

Human-centered quality evaluation. While our framework successfully automates MWCA fabrication, validating its experiential impact requires deeper investigation. We plan to conduct systematic user studies exploring whether the MWCA we designed conforms to the subjective aesthetic standards of the general public. By collecting preference ratings from artists, craftsmen, and general audiences across digital prototypes and physical installations, we will establish quantitative correlations between computational choices and human aesthetic judgment.

Holistic performance metrics. Additionally, extending our evaluation beyond PSNR and fabrication efficiency is crucial. Future work will develop multi-dimensional metrics integrating material consumption ratios (relative to solid baselines), structural safety factors (via parametric FEA), and durability benchmarks under environmental stressors (humidity/load cycles). This will create a comprehensive assessment framework balancing technical constraints, economic viability, and experiential quality for personalized fabrication systems.

6. Conclusion

This work delivers the first complete, image-to-fabrication pipeline for multi-layer wood-carving artwork (MWCA). By color-palette compression, a connectivity-aware greedy layer extractor, and a suite of structural and economic post-processors (layer merging, strategic hollowing, and stress-guided widening), the method simultaneously optimizes visual fidelity, structural safety, fabrication effort, and downstream logistics costs. Its modular design, fully automatic operation, and direct compatibility with standard laser cutters make it a practical foundation for truly mass-customized wooden reliefs.

Experimental validation across 18 digital examples and 3 physical prototypes demonstrates the system's ability to produce structurally viable artworks while preserving visual fidelity to reference inputs. The automated pipeline significantly reduces reliance on manual expertise, enabling efficient production workflows for personalized fabrication. Practical outcomes highlight reduced material waste, simplified assembly through boundary alignment, and improved structural durability compared to trial-and-error manual designs.

Future work will explore integrating AI-driven style transfer for gradient handling, advanced topology optimization for lightweight, and interactive tools for user-customizable adjustments. The methodology establishes a foundational framework for computational wood carving art, bridging digital design precision with traditional craftsmanship constraints.

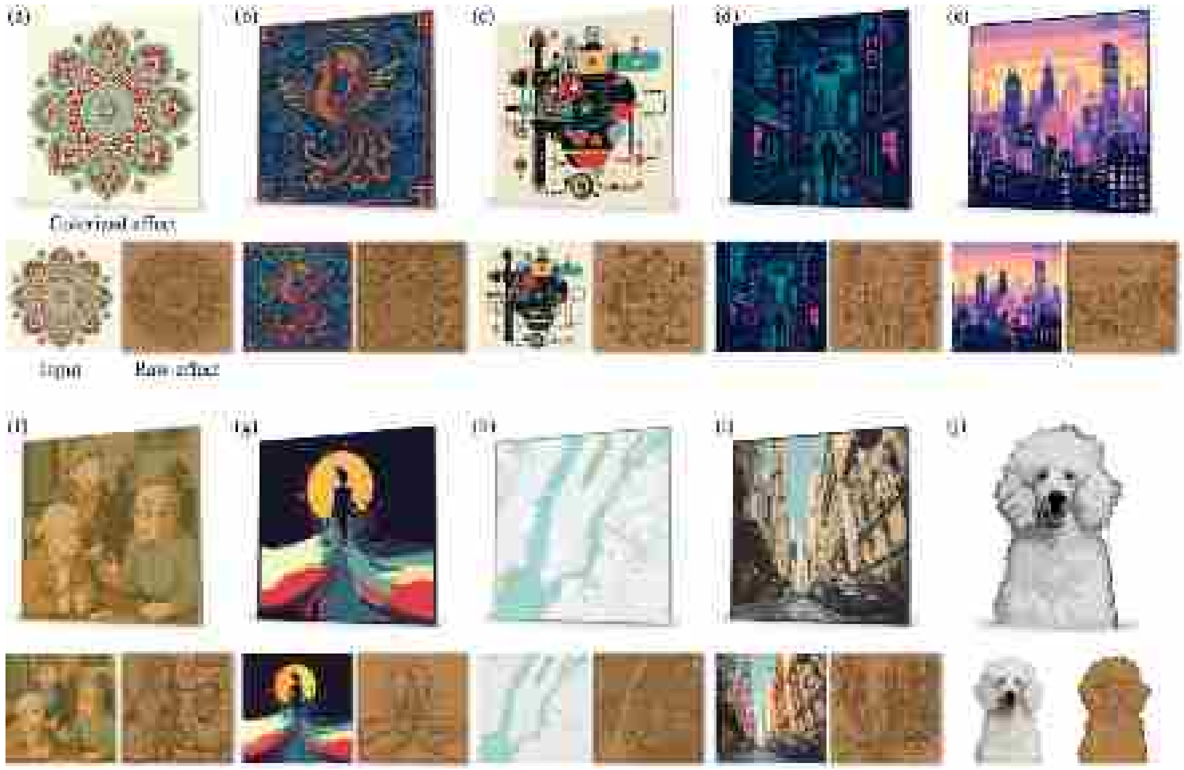

Fig. 16. A gallery of MWCA created using the proposed method, including various art themes: (a) Traditional Chinese pattern (created by GPT-4o [27]); (b) Phoenix pattern (created by GPT-4o [27]); (c) Mechanical style (created by Midjourney [12]); (d) Cyber Punk style (created by Midjourney [12]); (e) Illustration style (created by Midjourney [12]); (f) Ghibli style (created by GPT-4o [27]); (g) Poster (created by Midjourney [12]); (h) Map (created by Snazzy Maps [30]); (i) Photograph (taken by the authors); (j) Pet portraits (taken by the authors).

CRediT authorship contribution statement

Haochen Liu: Formal analysis, Investigation, Methodology, Validation, Visualization, Writing – original draft, Writing – review & editing, Conceptualization, Data curation. **Zhi Li:** Conceptualization, Formal analysis, Methodology, Supervision, Visualization, Writing – original draft, Writing – review & editing. **Kang Wu:** Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Supervision, Writing – original draft, Writing – review & editing. **Youcheng Cai:** Conceptualization, Formal analysis, Methodology. **Xiaoya Zhai:** Conceptualization, Formal analysis, Methodology. **Ketian Zhang:** Conceptualization, Formal analysis, Methodology. **Ligang Liu:** Funding acquisition, Supervision, Writing – review & editing. **Yi Min Xie:** Supervision, Writing – review & editing. **Xiao-Ming Fu:** Conceptualization, Formal analysis, Methodology, Project administration, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive suggestions and comments. This work is supported by the National Natural Science Foundation of China (62272429, 62025207) and the Australian Research Council (FL190100014).

Data availability

Data will be made available on request.

References

[1] STEREOWOOD. Stereo wood art. 2025, URL https://stereowoodart.com/.

[2] Umapathi U, Chen H-T, Mueller S, Wall L, Seufert A, Baudisch P. LaserStacker: Fabricating 3D objects by laser cutting and welding. In: Proceedings of the 28th annual ACM symposium on user interface software & technology. 2015, p. 575–82.

[3] Da Silveira G, Borenstein D, Fogliatto S. Mass customization: Literature review and research directions. Int J Prod Econ 2001;72(1):1–13. http://dx.doi.org/10.1016/S0925-5273(00)00079-7.

[4] Liu H, Zhang X-T, Fu X-M, Dong Z-C, Liu L. Computational peeling art design. ACM Trans Graph 2019;38(4):1–12.

[5] Iseringhausen J, Weinmann M, Huang W, Hullin MB. Computational parquetry: Fabricated style transfer with wood pixels. ACM Trans Graph 2020;39(2):1–14.

[6] Wang H, Qiu T, Li J, Lu Z, Ma Y. HarmonyCut: Supporting creative Chinese paper-cutting design with form and connotation harmony. 2025, arXiv preprint arXiv:2502.07628.

[7] Fukushima Y, Qi A, Shen I-C, Igarashi T. OVERPAINT: Automatic multi-layer stencil generation without bridges. In: SIGGRAPH Asia 2022 technical communications. New York, NY, USA: Association for Computing Machinery; 2022, http://dx.doi.org/10.1145/3550340.3564217.

[8] Zhu A, Mei Y, Jones B, Tatlock Z, Schulz A. Computational illusion knitting. ACM Trans Graph 2024;43(4):1–13.

[9] Xu H, Hui K-H, Fu C-W, Zhang H. Computational LEGO technic design. ACM Trans Graph 2021;38(6). http://dx.doi.org/10.1145/3355089.3356504.

[10] Ge J, Zhou M, Fu C-W. Learn to create simple LEGO micro buildings. ACM Trans Graph 2024;43(6):1–13.

[11] Zhou M, Ge J, Xu H, Fu C-W. Computational design of LEGO® sketch art. ACM Trans Graph 2023;42(6):1–15.

[12] Midjourney I. Midjourney image-generation platform. 2022, Open beta launched July 12, 2022, URL https://www.midjourney.com. [Accessed 6 May 2025].

[13] Tseng M, Jiao R, Wang C. Design for mass personalization. CIRP Ann 2010;59(1):175–8.

[14] Baudisch P, Mueller S, et al. Personal fabrication. Found Trends® Human–Comput Interact 2017;10(3–4):165–293.

[15] Ge J, Zhou M, Bao W, Xu H, Fu C-W. Creating LEGO figurines from single images. ACM Trans Graph 2024;43(4):1–16.

[16] Dewil R, Vansteenwegen P, Cattrysse D. A review of cutting path algorithms for laser cutters. Int J Adv Manuf Technol 2016;87:1865–84.

[17] Roumen T, Kommana Y, Apel I, Lempert C, Brand M, Brendel E, et al. Assembler3: 3d reconstruction of laser-cut models. In: Proceedings of the 2021 CHI conference on human factors in computing systems. 2021, p. 1–11.

[18] Tian X, Günther T. A survey of smooth vector graphics: Recent advances in repr esentation, creation, rasterization, and image vectorization. IEEE Trans Vis Comput Graphics 2022;30(3):1652–71.

[19] Price B, Barrett W. Object-based vectorization for interactive image editing. Vis Comput 2006;22:661–70.

[20] Battiato S, Barbera G, Di Blasi G, Gallo G, Messina G. Advanced SVG triangulation/polygonalization of digital images. In: Internet imaging VI, vol. 5670, SPIE; 2005, p. 1–11.

[21] Orzan A, Bousseau A, Winnemöller H, Barla P, Thollot J, Salesin D. Diffusion curves: a vector representation for smooth-shaded images. ACM Trans Graph 2008;27(3):1–8.

[22] Jeschke S. Generalized diffusion curves: An improved vector representation for smooth-shaded images. In: Computer graphics forum, vol. 35, Wiley Online Library; 2016, p. 71–9.

[23] Liao Z, Hoppe H, Forsyth D, Yu Y. A subdivision-based representation for vector image editing. IEEE Trans Vis Comput Graphics 2012;18(11):1858–67.

[24] Zhou H, Zheng J, Wei L. Representing images using curvilinear feature driven subdivision surfaces. IEEE Trans Image Process 2014;23(8):3268–80.

[25] Ma X, Zhou Y, Xu X, Sun B, Filev V, Orlov N, et al. Towards layer-wise image vectorization. In: 2022 IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 16293–302. http://dx.doi.org/10.1109/CVPR52688.2022.01583.

[26] Macqueen J. Some methods for classification and analysis of multivariate observations. In: Proc. symp. math. statist. and probability, 5th, vol. 1, 1967.

[27] OpenAI. GPT-4o (“Omni”) multimodal large language model. 2024, Official announcement published 13 May 2024. URL https://openai.com/index/hello-gpt-4o. [Accessed 6 May 2025].

[28] Suzuki S, Abe K. Topological structural analysis of digitized binary images by border following. Comput Vis Graph Image Process 1985;30(1):32–46. http://dx.doi.org/10.1016/0734-189X(85)90016-7, URL https://www.sciencedirect.com/science/article/pii/0734189X85900167.

[29] Douglas DH, Peucker TK. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. Cartographica 1973;10(2):112–22. http://dx.doi.org/10.3138/FM57-6770-U75U-7727.

[30] Krogh A. Snazzy maps: Free styles for google maps. 2013, URL https://snazzymaps.com. [Accessed 6 May 2025].

[31] Huang X, Xie YM. Evolutionary topology optimization of continuum structures: methods and applications. John Wiley & Sons; 2010.