



A non-supporting printing algorithm for fused filament fabrication of multi-branch structure

Pengfei Tang¹ · Xianfeng Zhao¹ · Hongyan Shi¹

Received: 19 July 2022 / Accepted: 13 March 2023

© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

3D printing technology overcomes the geometric constraints of parts manufacturing through directional layering, which has attracted significant attention from academia and industry. When printing an overhanging structure in the traditional printing process, it needs to create additional supporting structures. The supporting structures will increase additional printing time and material consumption and reduce the surface finish of the structure. In order to solve these problems, this paper proposed a five-axis non-supporting printing method based on point cloud data. This method segmented the multi-branch model to generate components by the concavity and convexity of adjacent voxels. Segmented overhanging of components by using the topological information that described by the skeleton. Based on the plane used for segmentation, the components were sliced at a suitable angle. Our algorithm was verified on a five-axis printer. The results show that this method saved the printing time by 22–36% and reduced material consumption by 20–30%. Simultaneously, it also reduced the structure surface roughness to a certain extent.

Keywords Five-axis printing · Fused filament fabrication · Non-supporting printing · Path planning

1 Introduction

Proposed first in the 1980s, 3D printing technology is also called layered manufacturing technology or rapid prototyping technology on some occasions. It is widely applied in electronics [1], energy equipment [2], medicine [3], biotechnology [4], optics [5], construction [6], aerospace [7], and so on due to its ability to manufacture complex structures. At present, the mainstream 3D printing technology includes stereo lithography appearance (SLA), digital light processing (DLP), fused filament fabrication (FFF), selective laser sintering (SLS), direct laser deposition (DLD), and wire arc additive manufacturing (WAAM).

As one of the mainstream 3D printing technologies, the key process of FFF can be introduced in the following three steps. First, a 3D model is modeled by a computer-aided design or 3D scanner. Slice the 3D model to generate a series of cross-sections. Then, it scans and fills the inside of the acquired slicing contour (i.e., path planning, commonly used

Z-shaped contour, and offset contour scanning) [8]. Convert the path to a GCODE that can be recognized by the machine. Finally, the printer reads the GCODE information of the input file and uses hot melt filaments to print out these slicing sections layer by layer. When it comes to overhanging structures, this printing method requires the creation of additional supporting structures to prevent the filaments from collapsing. Unfortunately, the supporting structures for printing will increase additional printing time and material consumption, causing difficulty in post-processing [9].

Nowadays, many non-supporting printing strategies have been proposed. Sundaram et al. [10] proposed a five-axis slicing algorithm based on the CAD model. This method segmented the overhanging structure by analyzing the angle between the normal vector of the model surface and the construction direction vector. However, this method is limited to specific CAD environments. In addition, this simple comparison method may not be applicable to some convex structures without overhanging. Lee et al. [11] proposed a five-axis laser direct deposition slicing method based on model topology. This method is simple and easy to implement, but it is only applicable to overhanging structures without mutation. Wang et al. [12] proposed a five-axis dynamic slicing algorithm to achieve non-supporting printing of structures

✉ Xianfeng Zhao
zxf5111@126.com

¹ College of Mechanical Engineering, Guizhou University, Guiyang 550025, China

with holes or grooves inside. However, due to the uneven slicing layer thickness, the surface of the structure has concave features. In order to solve the problem of Wang et al., Liu et al. [13] proposed a wire arc additive manufacturing method for a non-supporting structure; the model was sliced by equidistant polylines on the basis of Wang et al.'s method.

The collapse of hot melt filament is the key factor limiting non-supporting printing. Zhao et al. [14] explored the printable angles and influencing factors of overhanging structures. The printable overhanging angle depended on the slicing tilt angle and layer height. They proposed an inclined layer printing without assisted supporting structures in the traditional FFF printer. This method took the overhanging angle as an important parameter of the objective function, which optimized the slicing angle and realized the inclined layer printing. Wüthrich et al. [15] printed a test object with overhanging angles of 30–85°, where inaccurate edges could be found around 45°, accompanied by a decrease in the surface quality of the object. They proposed a four-axis FFF printing process with the printheads configured with a 45-degree tilt, allowing structures to be printed in inclined layers over a range of ± 45 degrees. Rotation axis C enabled the printer to print overhanging structures in a range of 360 degrees. 3D printing company Blackbelt 3D had unveiled a non-supporting, large-scale 3D printer with a moving conveyor belt as a build platform that could theoretically print structures of infinite length [16]. The printer could rotate around the Y-axis to print overhanging structures at 45-degree angles, but these overhanging structures could only be fabricated in one build orientation.

Non-supporting printing of multi-branch structures is the current research focus of non-supporting printing. Hu et al. [9] proposed to divide the branches of the model into multiple sub-objects by performing a K-means clustering algorithm so that multi-branch structure could be constructed in different directions. This method can only roughly divide the structure, but not accurately divide the structure from the concave. Wu et al. [17] proposed a volume segmentation printing algorithm, which segmented a model into multiple sub-objects and generated a printing path based on collision-free constraints of the system. This algorithm also does not accurately divide multi-branch structures. Dai et al. [18] proposed a voxel decomposition method, which has received much attention. This method decomposed the model into a series of voxels (the voxel size was equal to the nozzle diameter); the printing order of the voxels was determined by a scalar field algorithm. However, the structure will show artifacts due to factors such as hardware positioning errors, non-uniform layer thicknesses, and gaps between toolpaths.

Although these methods effectively solve the non-supporting printing of overhanging structures, they also have some problems, such as the inability to accurately locate the concave surface of the structure, uneven layer

thickness, and inability to print in large sizes. Unlike existing non-supporting printing methods, our method achieves precise segmentation of structural branches by positioning on concave surfaces and enables large-scale non-supporting printing through uniform inclining layer slicing. We proposed a printing method for multi-branch overhanging structures. In this method, the multi-branch of the model is segmented by the concavity and convexity of adjacent voxels to generate components. The skeleton is used to sub-segment the overhanging of components. Based on the plane used for segmentation, the components were sliced at a suitable angle. The proposed method can be used for non-supporting printing of complex multi-branch structures, which improves the accuracy and efficiency of the overhanging structures and reduces the consumption of printing materials.

2 Non-supporting printing strategy

2.1 Overview of non-supporting printing

When printing overhanging structures in the traditional 3-axis printing process, additional supporting structures need to be created to prevent the filaments from collapsing, as shown in Fig. 1. The supporting structure will increase the printing time and material consumption. In addition, the supporting structure needs to be removed after completion, which makes post-processing more difficult (such as peeling the supporting structure and cleaning). Most important of all, the surface finish of the structures is degraded due to the residue of supporting materials during the process of peeling the supporting structures. The non-supporting printing of overhanging structures can be realized by adjusting the printing direction during the printing process by multi-axis motion, so the supporting structures are not necessarily required to prevent the filaments from collapsing.

In order to achieve non-supporting printing of complex multi-branched structures, we proposed a five-axis non-supporting printing method. We use point cloud data as the data model because it is convenient to extract the skeleton of the model and avoids the transformation between models. The algorithm flow chart is stated in Fig. 2. The method we use will be shown as follows: First, the multi-branch model is segmented into different components by concavity and convexity of adjacent voxels. Then, the L1 median method is used to extract the skeleton of the pair of components; at the same time, the overhanging structures are segmented by the skeleton analysis method. For an already segmented model, we slice it by using a tangent plane paralleled to the segmentation plane and extract its contours by using the alpha shape algorithm. Finally, the

Fig. 1 Traditional printing method: (a) overhanging structure and (b) traditional printing path

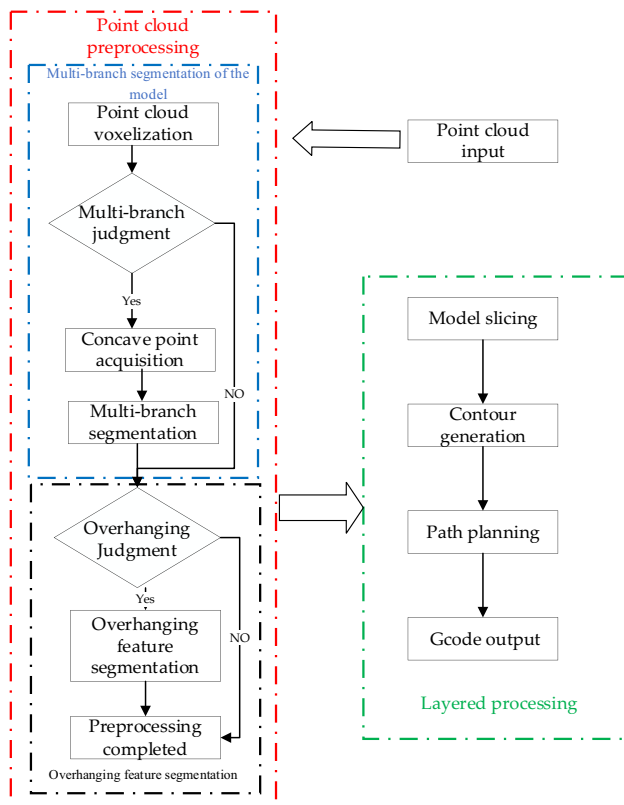
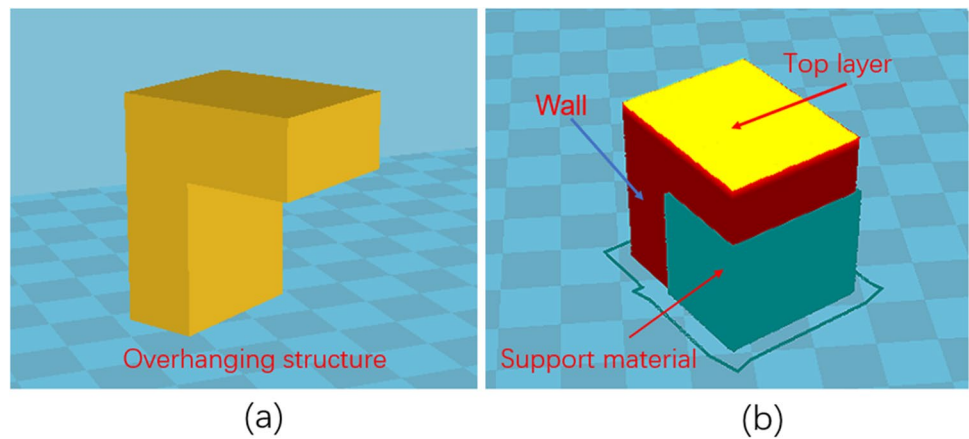


Fig. 2 Flowchart of the algorithm described above

contour offset algorithm is used to plan the path inside the contour and thus converts the path into GCODE. In Fig. 2, multi-branch judgment depends on whether there is a series of concave points. If there are concave points, the segmentation plane is generated by fitting the concave points to segment the multi branches of the model. The overhanging structures are judged by the angle between the connection vector of adjacent skeleton points and the construction direction. If the angle exceeds the setting threshold, the overhanging feature is considered to exist.

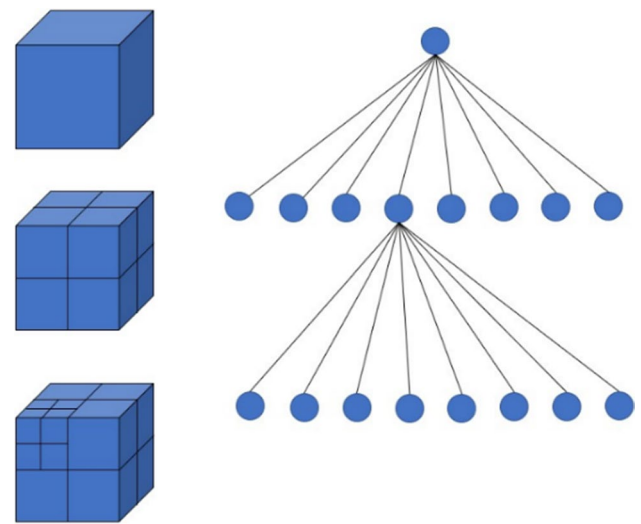


Fig. 3 Point cloud voxelization based on an octree structure

2.2 Segmentation of multi-branch structure

2.2.1 Point cloud voxel processing

Based on the octree structure, we split the point cloud to generate voxels. The point cloud voxel method can reduce computational costs and improve the efficiency of segmentation. As shown in Fig. 3, we first create a cube to contain the entire point cloud data and use it as the root node of the octree. The root cube will be divided into eight sub-cubes. If the sub-cube contains point cloud data, it will be retained; otherwise, it will be discarded. These sub-cubes are further divided according to the voxel resolution. If the resolution of the point cloud voxels is less than the limit of the user-defined threshold, it is further divided into eight sub-cubes; otherwise, it is stored as a terminal node.

To further improve computational efficiency, we perform super-voxels clustering on the point cloud. Super-voxels are unsupervised over-segmentation into regions of perceptually

similar voxels. We use a local region growing to generate super-voxel $\bar{V}_i = (\bar{x}_i, \bar{n}_i, N_i)$, where \bar{p}_i is the centroid, \bar{n}_i is the normal vector, and N_i is the adjacent super-voxel. Based on the voxel cloud connectivity segmentation (VCCS), we cluster the voxels by introducing their spatial distance D_s , color distance D_c , and normal distance D_n (calculated by using the histogram intersection kernel) as a basic parameter to generate super-voxels. To control the distance between adjacent voxels, we should normalize the spatial parameter. Equation (1) is used to calculate the normalized distance [19] as follows:

$$D = \sqrt{w_c D_c^2 + \frac{w_s D_s^2}{3R_{\text{seed}}^2} + w_n D_n^2} \quad (1)$$

where w_c , w_s , and w_n are the weights of color distance, space distance, and normal distance, respectively. R_{seed} is the seed resolution.

In these works, we do not consider color weight (i.e., $w_c = 0$), using only spatial distance weight ($w_s = 1$) and normal distance weight ($w_n = 4$). Then, we set the voxel resolution to 0.08 and the seed resolution to 0.32 for clustering. The super-voxel clustering result of the Γ -shaped structure is shown in Fig. 4. The corresponding CAD model is shown in Fig. 1a.

2.2.2 Obtaining the concave points of the model

When there are branches in the model, the model will have concave surfaces. Once we obtain the super-voxels adjacency map, we use the local concave and convex criteria to describe the concave and convex relations of adjacent super-voxels [20], as shown in Fig. 5. Define the unit vector \vec{d} parallel to $\vec{P}_1 - \vec{P}_2$, then,

$$\vec{d} = \frac{\vec{P}_1 - \vec{P}_2}{\|\vec{P}_1 - \vec{P}_2\|} \quad (2)$$

For concave adjacency voxels, there is always

$$\alpha_1 - \alpha_2 > 0 \quad (3)$$

Fig. 4 Voxel clustering result for Γ -shaped structure

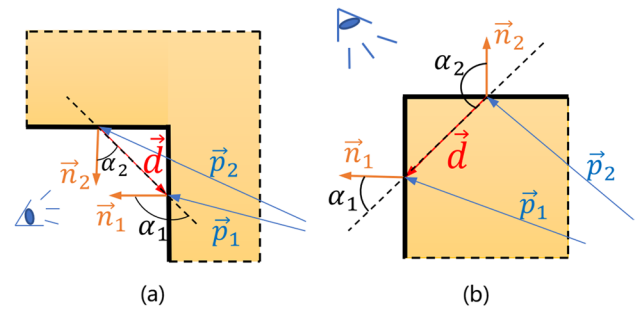
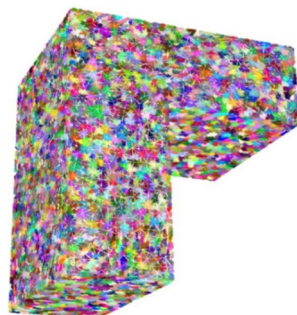


Fig. 5 Voxel local concave and convex criteria: (a) concave adjacency of voxels and (b) convex adjacency of voxels

or

$$\cos \alpha_1 - \cos \alpha_2 < 0 \quad (4)$$

For $\cos \alpha_1$ and $\cos \alpha_2$, the following formula

$$\begin{aligned} \cos \alpha_1 &= \frac{\vec{d}_1 \cdot \vec{n}_1}{\|\vec{d}_1\| \|\vec{n}_1\|} \\ \cos \alpha_2 &= \frac{\vec{d}_2 \cdot \vec{n}_2}{\|\vec{d}_2\| \|\vec{n}_2\|} \end{aligned} \quad (5)$$

If two voxels are concavity adjacency, then

$$\vec{n}_1 \cdot \vec{d} - \vec{n}_2 \cdot \vec{d} < 0 \quad (6)$$

Similarly, if two voxels are convex adjacency, the following formula can be obtained

$$\vec{n}_1 \cdot \vec{d} - \vec{n}_2 \cdot \vec{d} \geq 0 \quad (7)$$

To segment the branches of the model, we need to get the concave points of the model. Similar to the CPC algorithm [21], we convert the adjacency edges of voxels to point clouds, using points to represent the adjacency edges of voxels. The point coordinates are set to the average of two adjacency super-voxels centroid (i.e., $(\vec{P}_1 + \vec{P}_2)/2$), the black point cloud in Fig. 6a). In addition, we need to measure the relationship between the adjacent edge unit vector \vec{d} and the angle between the two super-voxels normal (\vec{P}_1, \vec{P}_2) . We use formula 6 to describe the concave connection and formula 7 to describe the convex connection.

Since some super-voxels could span the concavities of the model, two concave point sets are extracted (see Fig. 6a). In order to avoid the interference of noisy data on the subsequent segmentation plane construction, we use the RANSAC algorithm [22] to extract/fit the concave points (the red point cloud in Fig. 6a). The extracted concave points are shown in Fig. 6b,c. In Fig. 6, the blue points represent the original point cloud; the red points represent the extracted concave points.

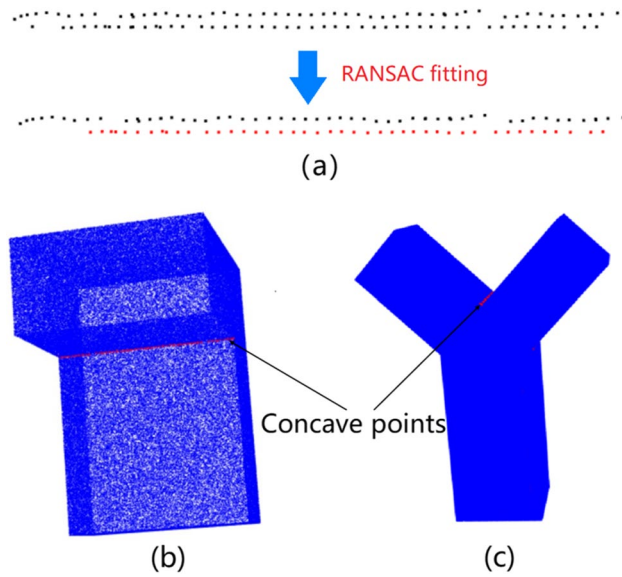


Fig. 6 Concave points extraction for the model: (a) RANSAC fitting of the converted point cloud; (b) concave points extracted from Γ -shaped part; and (c) the concave point extracted by the Y-shaped part

2.2.3 Multi-branch segmentation of structures

After obtaining the concave points of the model, we need to solve for the segmentation plane of the structure. The least square method is used to fit the concave points to obtain the segmentation plane. For an arbitrary plane, its normal formula is

$$x \cos \alpha + y \cos \beta + z \cos \gamma - d = 0 \quad (8)$$

where $\{\cos \alpha; \cos \beta; \cos \gamma\}$ is the unit normal vector of the plane, which satisfies $\sqrt{\cos^2 \alpha + \cos^2 \beta + \cos^2 \gamma} = 1$. The distance from the origin to the plane is expressed as d . The following assumption is

$$a_1 = -\cos \alpha / \cos \gamma; a_2 = -\cos \beta / \cos \gamma; a_3 = d / \cos \gamma \quad (9)$$

For n -dimensional discrete points (x_i, y_i, z_i) , the squared error should be minimized as follows:

$$S = \arg \min \sum_{i=0}^n (a_1 x_i + a_2 y_i + a_3 - z_i)^2 \quad (10)$$

or

$$\frac{\partial S}{\partial a_k} = 0, k = 1, 2, 3 \quad (11)$$

The equation matrix is.

$$\begin{pmatrix} \sum_{i=0}^n x_i^2 & \sum_{i=0}^n x_i y_i & \sum_{i=0}^n x_i \\ \sum_{i=0}^n x_i y_i & \sum_{i=0}^n y_i^2 & \sum_{i=0}^n y_i \\ \sum_{i=0}^n x_i & \sum_{i=0}^n y_i & n \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^n x_i z_i \\ \sum_{i=0}^n y_i z_i \\ \sum_{i=0}^n z_i \end{pmatrix} \quad (12)$$

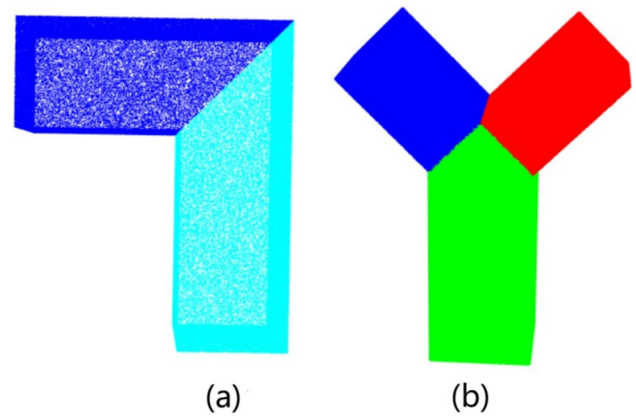


Fig. 7 Segmenting the branches of the structure: (a) single-branch model segmentation and (b) multi-branch model segmentation

By solving formula 9 and formula 12, we can obtain the segmentation plane of the structure. Due to the uncertainty of some segmentation planes, there may exist multiple sets of different roots. The segmentation angle can be set according to the constraints of the printer, as shown in Fig. 7a (in this paper, the angle γ is set to 45 degrees).

Considering that some models may have multiple sets of concave points, the K-means clustering method is used to classify the concave points. K-means clustering divides the sample set into k subsets according to the distance between the cluster centroids and the sample points. The essence of clustering is to minimize the sum of the distances between the cluster centroids and the sample points. The calculation formula is

$$SSE = \arg \min \sum_{i=1}^k \sum_{p \in p_i} \|p - c_i\|^2 \quad (13)$$

where k is the number of clusters, p_i is the i -th cluster, and c_i is the i -th cluster centroid.

The K-means clustering effectively classifies the concave points. In our algorithm, the RANSAC algorithm is used to fit the points transformed by adjacent edges. These concave points that have been clustered will be fitted by the least square method. The segmentation result with multiple groups of concave points is shown in Fig. 7b.

2.3 Overhanging segmentation

2.3.1 Research on the collapse angle of hot melt filament

Be aimed at better comparing the effect of the overhanging angle on the surface quality of the structure and exploring the critical angle of the collapse of the hot melt filament, the contour of the overhanging surface of the structure was extracted. Specifically, the printed overhanging structure

(see Fig. 8a) was photographed by an MT9T001 CMOS industrial camera. The contour was extracted by the graphics processing software developed by our research group. The structure of the experiment is shown in Fig. 8b. The fitting effect of the contour is shown in Fig. 8c.

In this experiment, the Cura slicer is used for slicing. The 5-axis printer we designed is used to print structures with different overhanging angles. The slicing parameters of the overhanging structure are shown in Table 1. To make sure these results clearer, we extracted the overhanging contours with different process parameters and plotted the curves described in Fig. 9. In the coordinate axis of Fig. 9, the horizontal axis represents the length of the structure contour; the vertical axis represents the degree of surface deformation of the structure contour.

The overhanging contour at overhanging angles of 20°, 30°, 40°, 50°, 60° and 70° are plotted in Fig. 9a,b, respectively. The folds/curves in the figure represent the surface shapes of the structures. Table 2 shows the maximum deformation of the structure under different overhanging angles. We can sequentially obtain the maximum deformation deviation of the surface under the overhanging angles of 20°, 30°, 40°, 50°, 60°, and 70°, which are 91 μm , 104 μm , 120 μm , 160 μm , 234 μm , and 314 μm , respectively. According to Fig. 9, when the overhanging angle of the structure exceeds 40 degrees, the maximum peak surface deformation of the overhanging structure increases sharply. When the overhanging angle exceeds 40°, the surface roughness decreases sharply, which is mainly due to the collapse of the hot melt filament. In our study, combined with the geometric analysis of the

Table 1 Printing parameters for overhanging structures

Printing speed, mm/s	Layer thickness, mm	Wall thickness, mm	Nozzle diameter, mm	Printing temperature, °C	Fan speed, RPM
20	0.2	0.3	0.3	200	255

collapse of hot melt filaments in literature [14, 15], we set the critical collapse angle of the hot melt filament to 45°.

2.3.2 Model skeleton extraction

In the “Segmentation of multi-branch structure” section, we only extract concave points to segment the multi-branch

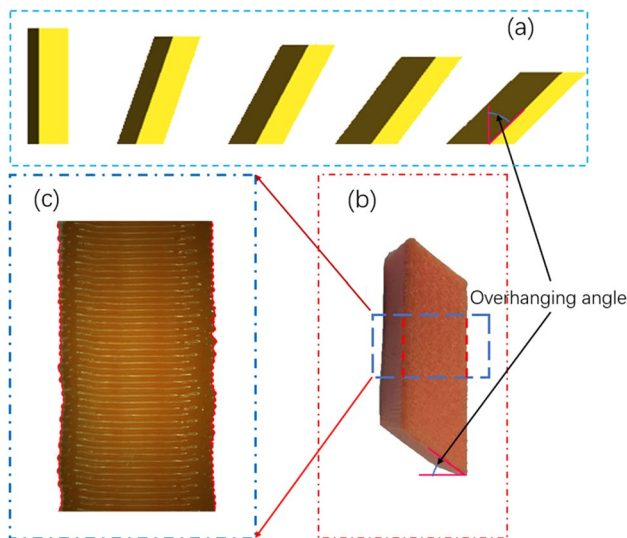
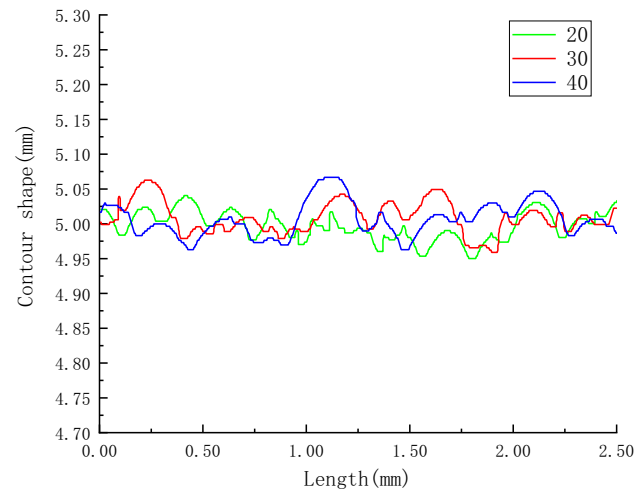
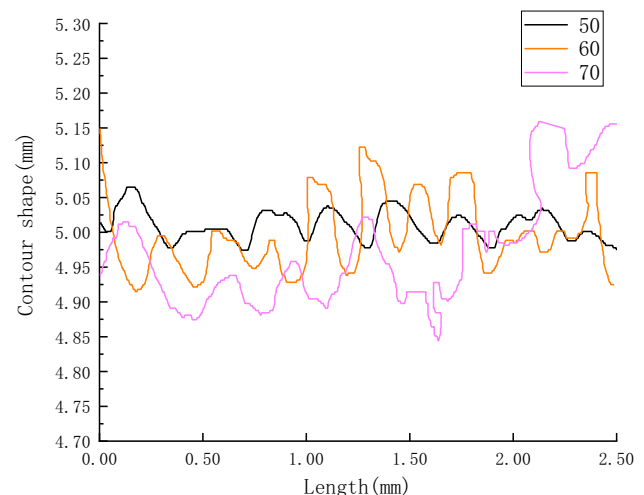


Fig. 8 Contour fitted by using graphics processing technology: (a) tested overhanging model; (b) printed overhanging structure; and (c) fitted contour



(a)



(b)

Fig. 9 Relationship between overhanging angle and surface deformation: (a) contour shape with the overhanging angle of 20–40 and (b) contour shape with the overhanging angle of 50–70

Table 2 The relationship between the maximum deformation and the overhanging angle

Overhanging angle (°)	20	30	40	50	60	70
Maximum deformation (μm)	91	104	120	160	234	314

of the model without segmenting the overhanging features of the multi-branch model. L1 median method [23] is used to extract the skeleton of components and then sub-segment the overhanging feature by analyzing the skeleton. For a given point set $Q = \{q_j\}, j \in J$, we need to find a sample point x^* to minimize the distance from the sample point to the point set. The L1 median value is

$$x^* = \sum_{j \in J} \frac{q_j w_j}{\sum_{j \in J} w_j} \quad (14)$$

$$w_j = \frac{1}{\|x^* - q_j\|}$$

The skeleton cannot be obtained directly by using the data of the whole point cloud. It is necessary to down-sample the point cloud data. The L1 median value of the neighborhood centered on the sampling point is calculated. The L1 median of the neighborhood is sampled by continuous iteration; at the same time, the iterative L1 median is taken as a new sampling point. After a certain number of iterations, the L1 median falls into the local minimum when the iteration is stopped. Given a sequence of sample points $X = \{x_i\}, i \in I$, the optimization formula [23] is used as follows:

$$\arg \min_X \sum_{i \in I} \sum_{j \in J} \|x_i - q_j\| \theta(\|x_i - q_j\|) + R(x) \quad (15)$$

In order to avoid the sampling function falling too fast, the Gaussian weight function is used to make the whole energy system more stable. The Gaussian weight function is

$$\theta(\|x_i - q_j\|) = \exp[-4(\|x_i - q_j\|/h)^2] \quad (16)$$

where h is the sampling radius. In order to form a repulsion between the sampling points and avoid the distance between the skeleton points being too close, a repulsion term needs to be added. The repulsion term is

$$R(x) = \sum_{i \in I} \gamma_i \sum_{i' \in I \setminus \{i\}} \frac{\theta(\|x_i - \bar{x}_{i'}\|)}{\sigma_i \|x_i - \bar{x}_{i'}\|} \quad (17)$$

where γ_i is the parameter to balance attraction and repulsion, $\bar{x}_{i'}$ is the average of the sampling points, and σ_i is the distribution metric value of the point. σ_i is calculated as follows:

$$\sigma_i = \sigma(x_i) = \frac{\lambda_i^2}{\lambda_i^0 + \lambda_i^1 + \lambda_i^2} \quad (18)$$

where λ_i^0 , λ_i^1 , and λ_i^2 are the eigenvalues of the sampling point covariance matrix, which satisfies the relation: $\lambda_i^0 < \lambda_i^1 < \lambda_i^2$. The sampling point covariance matrix is as follows:

$$C_i = \sum_{i' \in I \setminus \{i\}} \theta(\|x_i - \bar{x}_{i'}\|) (x_i - \bar{x}_{i'})^\top (x_i - \bar{x}_{i'}) \quad (19)$$

Figure 10 is the extraction process of the L1 median method, in which the black, blue, and red points represent the original, sampling, and skeleton points, respectively. First, the point cloud is down-sampled (see Fig. 10a). The spherical neighborhood with radius h is constructed with the sampling point as the center. The distribution metric value σ_i of each sampling point is calculated. Then, the L1 median of the neighborhood will be calculated. The generated L1 median is used as a new sampling point to construct a new neighborhood. At the same time, the L1 median value of the new neighborhood is continuously calculated (see Fig. 10b). After reaching a certain number of iterations, the generated sampling points fall into a local minimum, thus stopping iteration (see Fig. 10c).

2.3.3 Overhanging segmentation of structure

In order to be able to segment the overhanging features of the model, adjacent sampling skeleton points are used to analyze the topological orientation of the model (see Fig. 11). Given the i -th skeleton point $\bar{Q}_i = (x_i, y_i, z_i)$, we calculate the unit connection vector between the current skeleton point and the previous skeleton point. The unit vector \bar{n}_i is

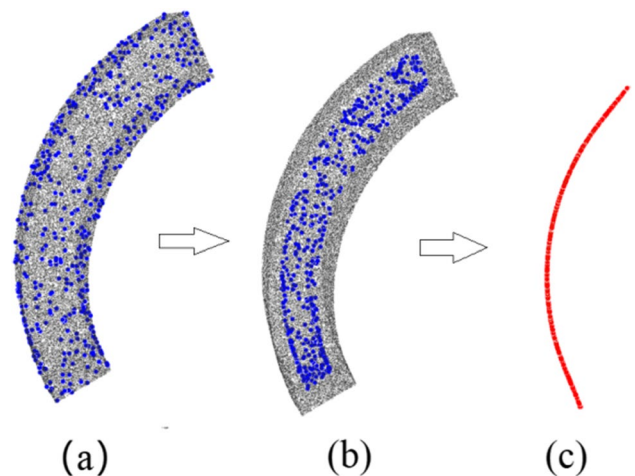


Fig. 10 L1 median skeleton extraction process: (a) point cloud down-sampling; (b) L1 median iteration; and (c) generated model skeleton

$$\vec{n}_i = \frac{\vec{Q}_i - \vec{Q}_{i-1}}{\|\vec{Q}_i - \vec{Q}_{i-1}\|} \quad (20)$$

By defining a vector that represents the construction direction \vec{n}_c , we can solve for the segmentation angle θ of the overhanging feature of the structure. The angle θ is

$$\theta = \cos^{-1} \frac{\vec{n}_c \cdot \vec{n}_i}{\|\vec{n}_c\| \|\vec{n}_i\|} \quad (21)$$

When the overhanging angle is greater than 45 degrees, the structure will be re-segmented, and the skeleton connection vector is used as the normal vector of the segmentation plane, as shown in Fig. 11. Given the slicing plane formula,

$$Ax + By + Cz + D = 0 \quad (22)$$

The parameters in formula 22 are

$$\begin{aligned} A &= x_i - x_{i-1} \\ B &= y_i - y_{i-1} \\ C &= z_i - z_{i-1} \\ D &= -Ax_{i-1} - By_{i-1} - Cz_{i-1} \end{aligned} \quad (23)$$

Fig. 11 Constructing the segmentation plane of overhanging

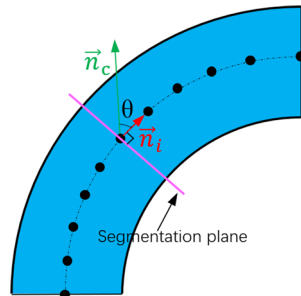
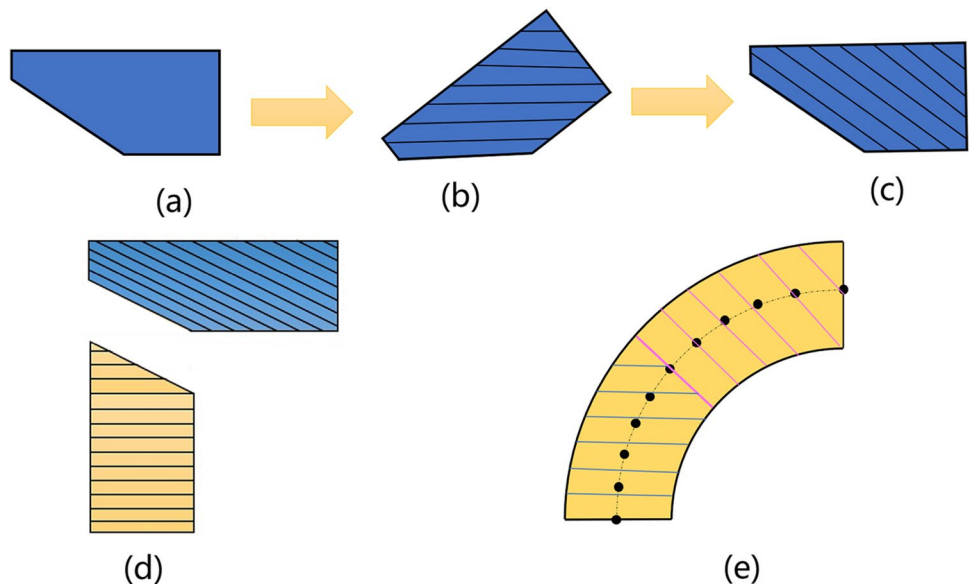


Fig. 12 Model slicing: (a) segmented structure; (b) rotated structure; (c) original state; (d) Γ -shaped overhanging structure slicing; and (e) arc-shaped overhanging structure slicing



2.4 Model slicing and contour generation

After segmenting the overhanging of the structure, it is necessary to divide the segmented models into different groups according to the overhanging angle range. Each set of overhanging models will be sliced at an appropriate slicing angle. In order to ensure the stability of the connection during the construction of different overhanging components, a plane parallel to the segmentation plane is used to slice the model. Rotate the segmented structure (see Fig. 12a) to an appropriate angle along the segmentation plane (see Fig. 12b); then, slice the rotated structure with a plane. Finally, rotate it to the original state (see Fig. 12c). Figure 12b is a slicing view of a Γ -shaped overhanging structure. Figure 12c is a slicing view of an arc-shaped overhanging structure.

After slicing the point cloud into 2.5D layers, we project the data points between adjacent layers onto the slicing plane to form the projected point band. We use the alpha shape algorithm [24] to extract the contour of the projection point band. The key of the alpha shape algorithm is to judge which points are on the contour line. In point set P, two points $p_1(x_1, y_1)$ and $p_2(x_2, y_2)$ are randomly selected to construct a circle with radius R. If there are no other points in the circle, p_1 and p_2 are considered to be points on the contour and the line-segment p_1p_2 constitutes the contour. Given the center of the circle is $P_0(x_0, y_0)$, there is the following formula:

$$\begin{cases} x_0 = x_1 + \frac{1}{2}(x_2 - x_1) + H(y_2 - y_1) \\ y_0 = y_1 + \frac{1}{2}(y_2 - y_1) + H(x_2 - x_1) \end{cases} \quad (24)$$

In formula 24,

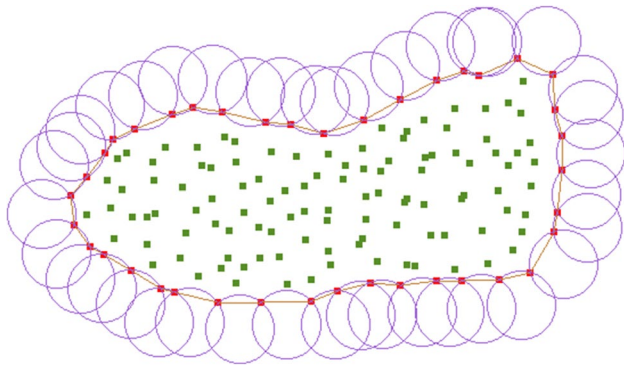


Fig. 13 Alpha shape algorithm trajectory

$$H = \sqrt{\frac{R^2}{S_{p_1 p_2}^2} - \frac{1}{4}} \quad (25)$$

$$S_{p_1 p_2}^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2 \quad (26)$$

As shown in Fig. 13, the contour line is extracted by the alpha shape algorithm. Its essence is that the circle with radius R rolls outside the point cloud band. The trace of its rolling is the boundary line of this point cloud band. When the value of R is small, an arbitrary point in the point cloud may be included in the circle. When R is infinite, the obtained contour is a convex polygon. After contour generation, the interpolation efficiency and printing speed of the 3D printer will be affected due to a large number of contour points. Therefore, redundant points on the same line need to be deleted in order to simplify the contour points while only retaining the initial point and end point.

2.5 Contour filling

After obtaining the contour layer, we use the contour offsetting algorithm to fill the inside of the contour. The contour

Fig. 14 Contour offset

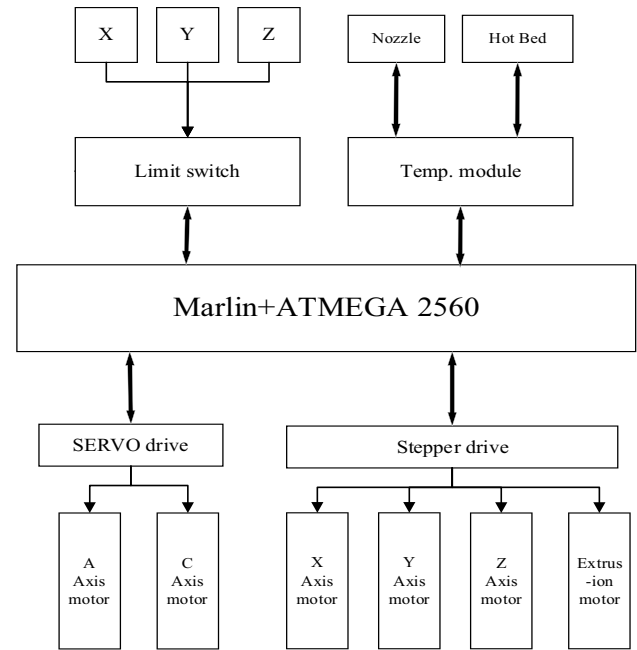
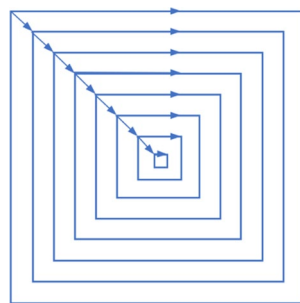


Fig. 15 Printer control system structure

offsetting algorithm offsets the contour to the direction of the entity with a scanning vector, scanning layer by layer from inside to outside or from outside to inside, as shown in Fig. 14.

Given an arbitrary polygon, its centroid is $P(X_0, Y_0, Z_0)$. The distance from the centroid to the front edge of the offsetting is defined as $\|\overrightarrow{PP_n}\|$. The distance from the centroid to the edge after offsetting is defined as $\|\overrightarrow{PP_n}\|$. For its scaling along the center point P , there is the formula as follows:

$$k\|\overrightarrow{PP_n}\| = \|\overrightarrow{PP_n}\| \quad (27)$$

In the formula 27, k is the scaling factor. Given the offset distance L as follows

$$\|\overrightarrow{PP_n}\| = L + \|\overrightarrow{PP_n}\| \quad (28)$$

Therefore, the scaling factor of each side can be obtained as follows:

$$k = \frac{L + \|\overrightarrow{PP_n}\|}{\|\overrightarrow{PP_n}\|} \quad (29)$$

The centroid of the polygon is translated to the origin of the coordinate system. After scaling the polygon, it is translated to its original position. The normalized homogeneous coordinate matrix after coordinate point offsetting is

$$T = T_1 \cdot T_2 \cdot T_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -X_0 & -Y_0 & -Z_0 & 1 \end{bmatrix} \begin{bmatrix} k & 0 & 0 & 0 \\ 0 & k & 0 & 0 \\ 0 & 0 & k & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ X_0 & Y_0 & Z_0 & 1 \end{bmatrix} \quad (30)$$

2.6 GCODE generation

The GCODE consists of the start coordinate, end point coordinate, and auxiliary code of each printing line segment. Auxiliary code includes temperature module code and extrusion module code. During the printing process, the extruder extrudes the hot filaments in different lengths and extrusion diameters. Therefore, it is necessary to calculate the extrusion length. We define the extruded filaments diameter of the printer as d_1 , the diameter of the printing consumable as d_2 , the length of the extruded hot filament as L_1 , and the length of the consumable as L_2 . The corresponding formula is as follows:

$$\frac{L_1 \pi d_1^2}{4} = \frac{L_2 \pi d_2^2}{4} \quad (31)$$

In the process of printing, the diameter of the extruded filaments may be inconsistent. If d_i is the diameter of the i -th line, it satisfies the following formula:

$$d_i = \frac{\sqrt{\|P_i P\|^2 - \left(\frac{\overline{P_i P} \cdot \overline{P_i P_{i+1}}}{\|P_i P_{i+1}\|}\right)^2}}{k} \quad (32)$$

The length of the i -th consumable material is

$$L_i = L_2 \left(\frac{d_2}{d_i}\right)^2 \quad (33)$$

Considering that the parameters of the slicing layer need to be converted into the parameters of the A-axis and C-axis, it is necessary to establish a rotation matrix around the A-axis and C-axis. The rotation matrix is

$$T = T_A \cdot T_C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\delta & -\sin\delta & 0 \\ 0 & \sin\delta & \cos\delta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 & 0 \\ \sin\varphi & \cos\varphi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (34)$$

2.7 Design of five-axis printing system

As shown in Fig. 15 is our five-axis printing system. The printer control system can execute the GCODE sent by the host system and send the print status report to the host system through USB communication. The printer control

system includes control module, limit module, temperature module, step module, and servo module. The control module includes Marlin GCODE firmware and ATMEGA 2560. After completing the GCODE conversion by Marlin, ATMEGA will send a command message to each sub-module so that each module can complete the relative function task. The limit module determines the printing range and initial printing point of the machine. The temperature module controls the temperature of the nozzle and the hot bed. The stepping module controls the movement of the X, Y, Z, and extruder motor. The servo module controls the rotational movement of the A and C axis of the printing platform.

Figure 16 shows the mechanical structure of the five-axis printer. The printer adds 2 degrees of freedom on the basis of traditional printers to achieve five-axis printing (see Fig. 16a). The mechanical structure is mainly composed of a feeding mechanism and a rotary printing platform (see Fig. 16b). The movement of the feeding mechanism is realized by the Core XY mechanism. The screw pair is used to realize the movement of the platform in the Z-axis direction. The rotation around the X-axis and the Z-axis is realized by the Servo. The upper computer adopts the secondary developed Pronterface software, which loads the G code into the software and communicates with the printer through the G code. In Marlin firmware, parsing instruction is executed through the switch case. ATMEGA analyzes buffer instructions through the switch case and assigns the decoded information to the corresponding function to control the movement of the five-axis printer.

3 Results and analysis

3.1 Experimental scheme and algorithm verification

In the experimental section, based on our proposed non-supporting printing algorithm and traditional slicing method, 3 types of overhanging structures are printed. In order to compare the printing time and material consumption of the two algorithms, we set travel speed, the inner/outer layer speed, and the top/bottom layer speed to 30 mm/s. The process parameters used for both printing methods in our experiments are shown in Table 3. The material consumption and printing time of the structures are recorded, respectively. The effects of the two methods on the surface quality of the structures are compared.

Fig. 16 Five-axis printer: (a) motion freedom of printer and (b) components of printer

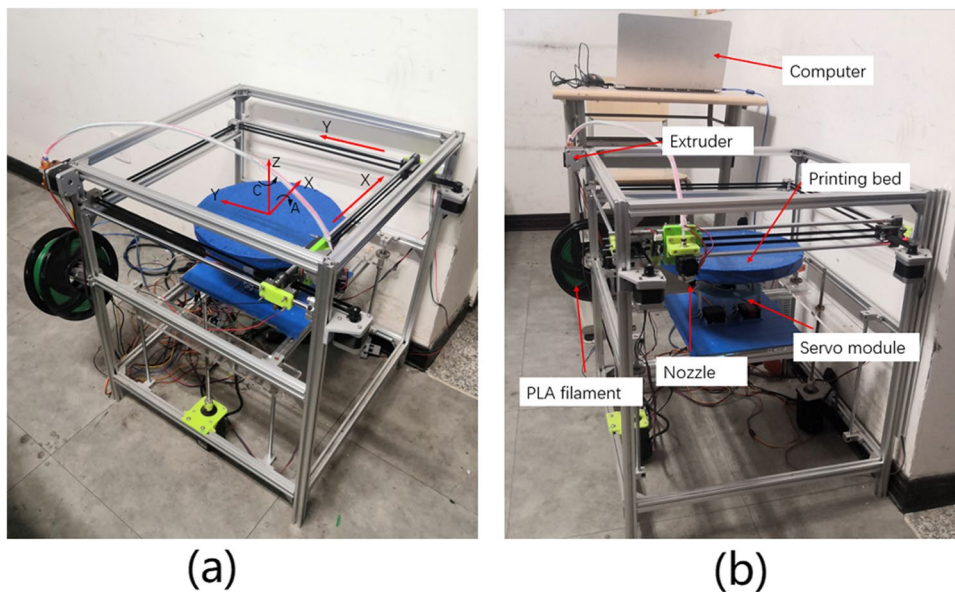


Table 3 Printing parameters

Layer thickness, mm	Printing speed, mm/s	Temperature, °C	Fan speed, RPM	Wire diameter, mm	Nozzle diameter, mm
0.2	30	210	255	1.75	0.3

The printed structures by the non-supporting printing strategy and the traditional printing strategy are all carried out on the five-axis printer we designed. Figure 17 shows the structure printed by two different methods. Figures a1, b1, and c1 show our 5-axis non-supporting printing structures. Figures a2, b2, and c2 show the structures by using

Fig. 17 Comparison of two printing methods: (a1–c1) five-axis non-supporting printing structure and (a2–c2) structure printed by traditional printing method

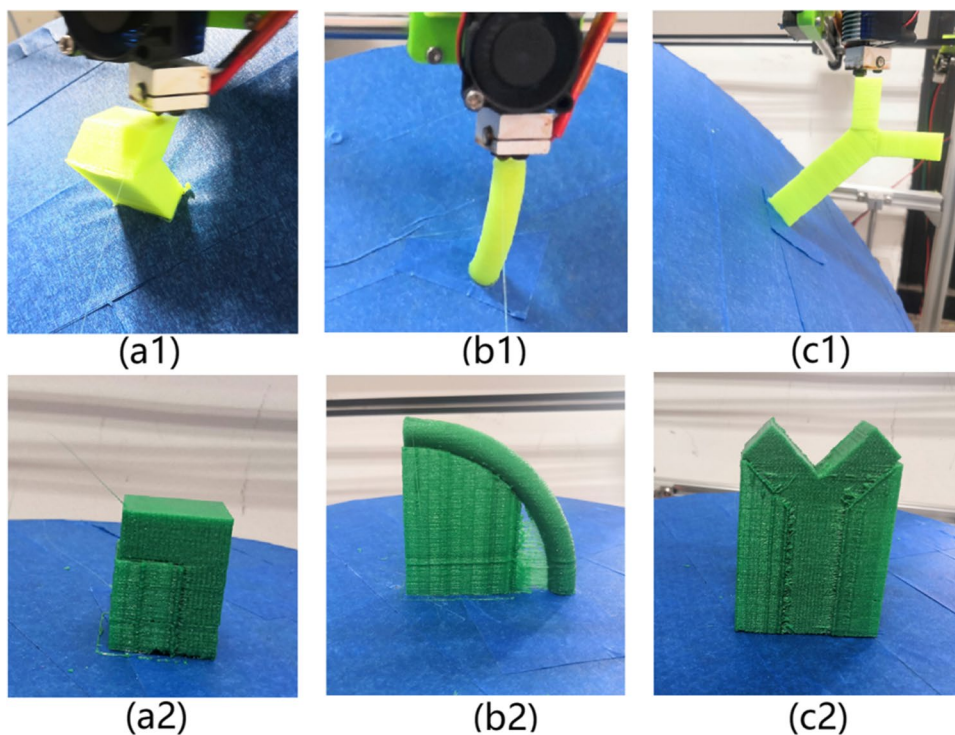
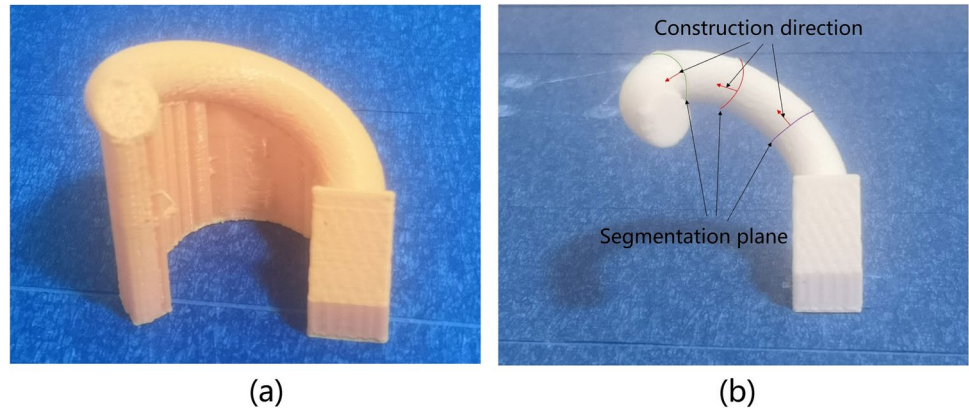


Fig. 18 Spiral structure: (a) traditional printing and (b) schematic diagram of structure segmentation



the traditional printing method. Figures a1 and a2 show the Γ -shaped overhanging structure. Figures b1 and b2 show the

arc-shaped overhanging structure. Figures c1 and c2 show the Y-shaped overhanging structure.

Fig. 19 Comparison of two printing strategies: (a) non-supporting printing Γ -shaped structure; (b) non-supporting printing arc-shaped structure; (c) 3-axis printing Γ -shaped structure; (d) 3-axis printing arc-shaped structure; and (e) printing structure contour shape

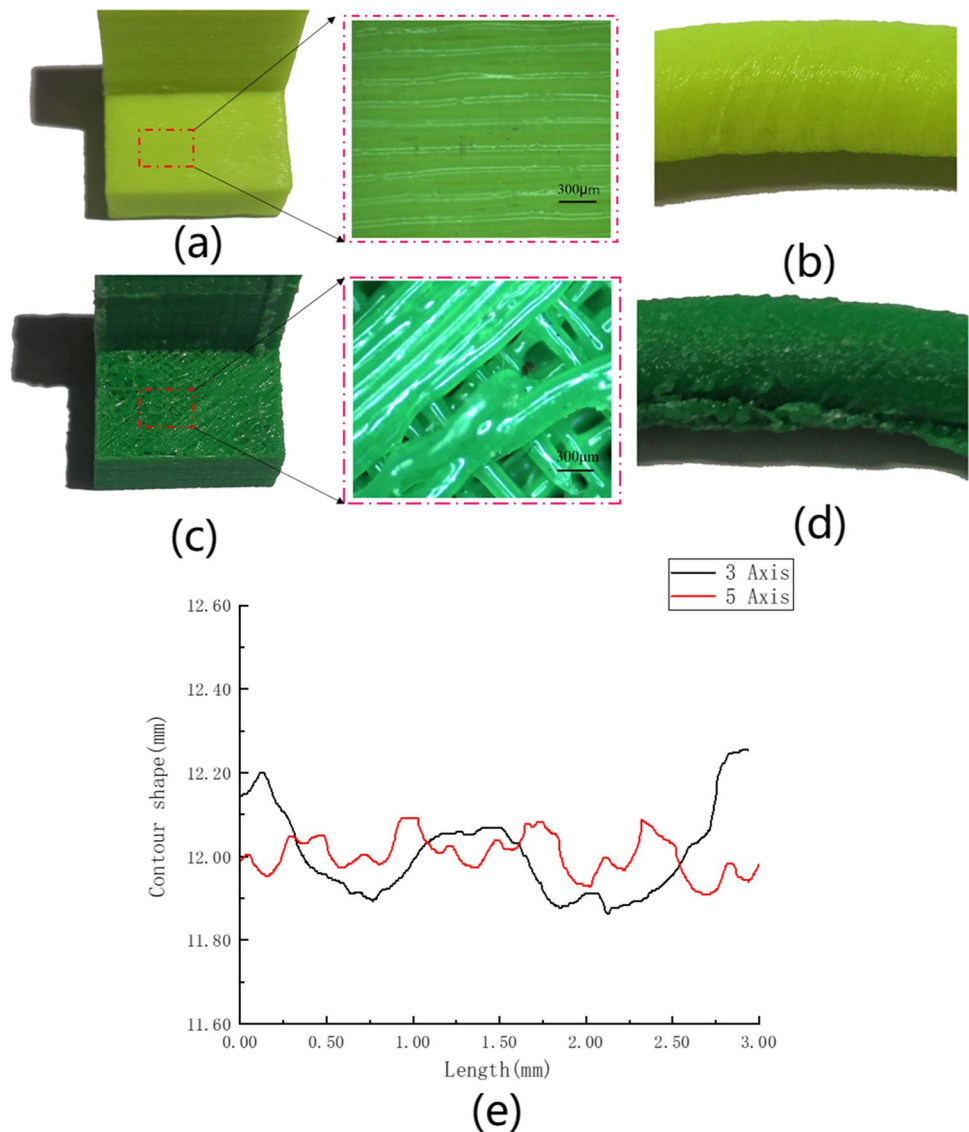


Table 4 Structural printing materials and time statistics

	Γ-shaped structure		Arc-shaped structures		Y-shaped structure	
	3-axis	5-axis	3-axis	5-axis	3-axis	5-axis
Time (h)	2.15	1.68	1.84	1.2	6.27	4.01
Material (g)	15.2	12.1	10.62	8.01	41.1	28.8
Time saving ratio	22%		35%		36%	
Material saving ratio	20%		25%		30%	

As shown in Fig. 18, which is a spiral structure for printing, Fig. 18a shows a spiral structure printed by a traditional method; Fig. 18b shows the segmentation diagram of the spiral structure using our non-supporting algorithm. First, we segmented the multi-branch of the structure and extracted the skeleton of the branch. Then, the overhanging of the branch was segmented by the angle between the connection vector between adjacent skeleton points and the construction direction; the segmentation plane was taken as the new construction plane. Finally, the overhanging was continuously segmented by measuring the angle between the construction direction and the skeleton point connection vector.

3.2 Comparison of printed structure

Figure 19 shows the contoured surfaces of the printing overhanging structure by two strategies. Figure 19a,b shows the surface effect of printing the overhanging structure by using the five-axis non-supporting method. Figure 19c,d shows printing using traditional methods. Compared to the overhanging surface of the structure of Fig. 19b, the filaments of the structure of Fig. 19a are arranged more neatly. Although supporting structures are added to prevent the filaments from collapsing in the printing method of Fig. 19b, the printed structures have recessed features after removing the supporting structures. The main cause of this recessed feature is the direct contact of the structure with the supporting material.

That is, some of the filaments of the structure are removed directly together during the dismantling of the supporting structure.

In order to better compare the deformation degree of these structures, we extracted the overhanging contour of the structure (see Fig. 19e). As we can see in Fig. 19e, the surface deformation of traditional printing overhanging is significantly higher than that of non-supporting printing. The main reason for the deterioration of the surface quality is the addition of supporting structures. In the traditional three-axis printing process, the structure is always fabricated layer by layer from bottom to top along a fixed Z-axis, which makes the overhanging surface contact with the supporting structures, resulting in excessive surface roughness of the structure.

As shown in Table 4, we measured the printing materials and printing time. Compared with traditional 3D printing, our strategy can avoid the use of supporting materials, thus shortening the printing time and significantly improving the printing efficiency. We calculated the percentage of material savings and percentage of time saving for our non-supporting strategy. Compared with traditional 3D printing, non-supporting printing can save 20–30% of the material and reduce time by 22–36%.

As we all know, 3D printing technology has always been considered a green manufacturing technology. 3D printing maximizes the use of resources and minimizes the impact on the environment from the entire product life cycle (including

Fig. 20 Staircase effect: (a) inclining layer slicing strategy and (b) staircase effect after layering

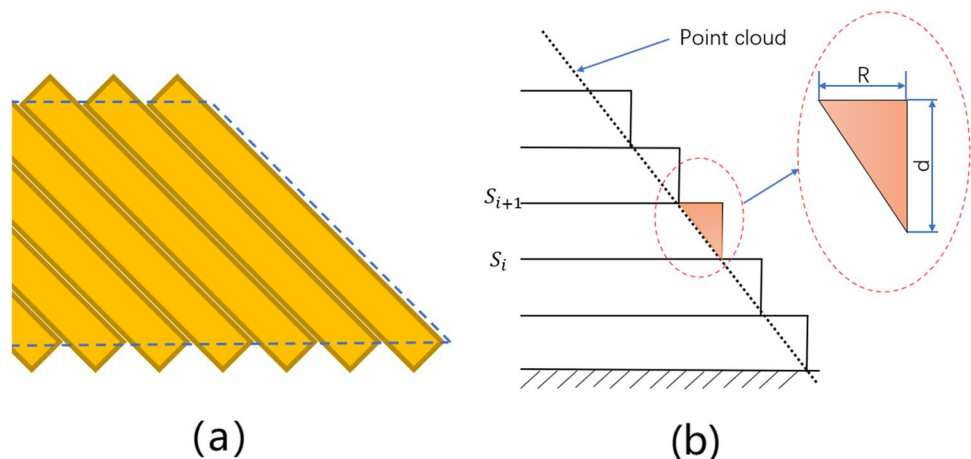
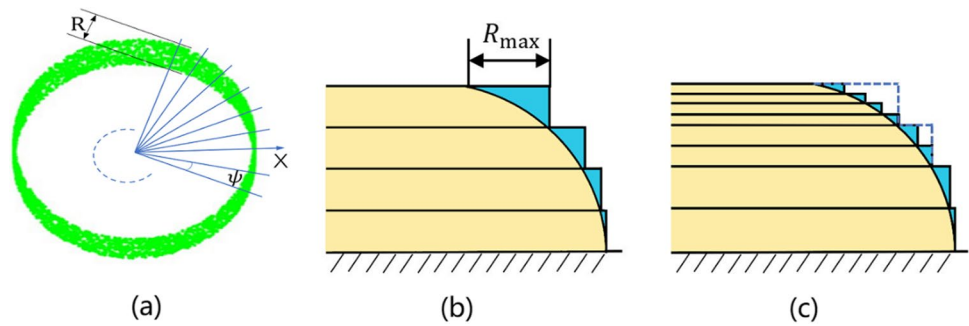


Fig. 21 Adaptive slicing illustration: (a) the radial width of the point cloud band; (b) slice at maximum slicing thickness; and (c) slice by maximum radial width



the process of design, manufacturing, transportation, use, and end-of-life disposal). However, most traditional 3D printing does not achieve the maximum utilization of resources. Especially when printing overhanging structure, additional printing supporting structure is required. Therefore, it can be considered that only non-supporting 3D printing can realize the maximum utilization of resources and the manufacturing advantage of additive manufacturing.

3.3 Algorithm improvement

In order to enable large-scale printing and non-supporting printing, our algorithm does not make the construction direction coincide with the normal of the model surface (see Fig. 20a). The staircase effect will occur when there is an angle between the normal of the model surface and the construction direction of the structure (see Fig. 20b). The generation of the staircase effect will affect the dimensional accuracy and surface roughness of the fabricated structure.

Considering that there may be a staircase effect when printing some structures with our algorithm, adaptive slicing is applied to our algorithm. In the adaptive slicing algorithm, we control the slicing thickness with the maximum radial width as an error. When the step size R of the adjacent slicing layer exceeds our defined threshold, the adaptive slicing algorithm works to suppress the step effect. The method projects the point cloud between the slices plane S_i and S_{i+1} of the two layers to the slicing plane S_i to form a point cloud band (see Fig. 20b). After obtaining the point cloud band, the radial width R of the point cloud band needs to be calculated. The function of R is that it can more accurately reflect the maximum gap between the two layers of contour, so as to effectively control the shape error. To obtain the radical width R , do the following steps: First, use the plane passing through the centroid and parallel to the X plane as the original plane to rotate and cut the point cloud band, as shown in Fig. 21a. The included angle between the cutting planes is ψ . R is calculated by projecting the point

between two cutting planes to the next cutting plane. After calculating all the radial width R of this layer, find out the maximum radial width R_{\max} of this layer.

In order to improve the printing accuracy of the structure, the adaptive slicing strategy is adopted. As shown in Fig. 21b, the point cloud is first sliced and stored with the maximum allowed slicing layer thickness H . Next, the maximum radial width R_{\max} of each slicing layer is calculated. Finally, it is determined whether the slicing layer needs to be subdivided according to the maximum radial width R_{\max} , see in Fig. 21c. Repeat the above steps until the maximum radial width is less than the set threshold.

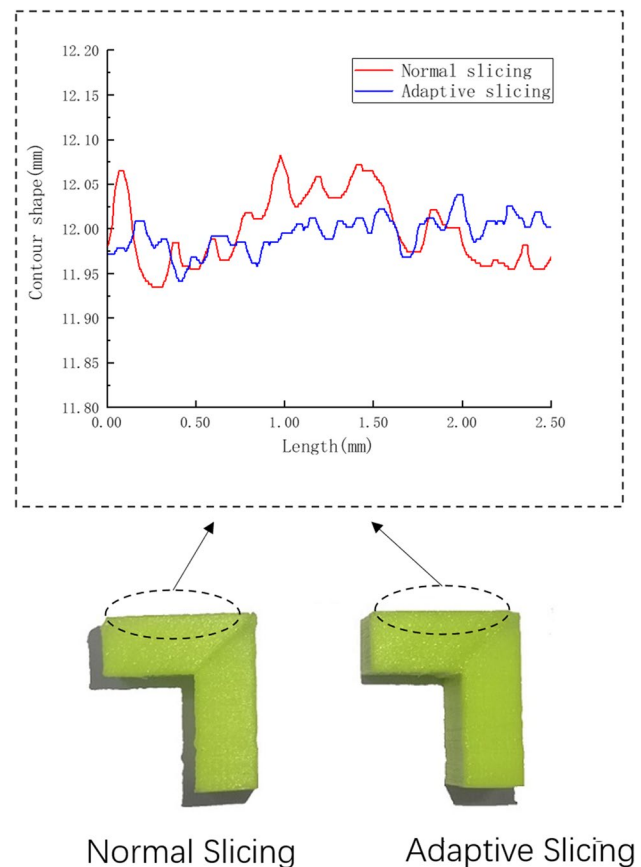


Fig. 22 Comparison of normal slicing and adaptive slicing

In this paper, set parameters $\psi = 0.5$ degree and $R_{\max} = 0.05$ mm. First, the point cloud model with a layer thickness of 0.2 is sliced. Then, the maximum radial width R_{\max} of each layer is calculated. If it is a layer whose maximum radial width is greater than the setting threshold, the slicing will continue. Figure 22 compares the surface quality of the structure with and without adaptive slicing. We extracted the contour shape and plotted the shape of the contour. From Fig. 22, we can see that the peak and trough of the contour using adaptive slicing are flatter, and the surface finish of the structure is improved.

4 Conclusions

In this paper, we proposed a five-axis non-supporting printing method based on point cloud data. The method first segmented the multi-branch of the model through the concavity and convexity of adjacency voxels to generate the components. Then L1 median method was used to extract the skeleton of components. The overhanging feature of the structure was segmented by the angle between the skeleton and the structural direction. Finally, four types of overhanging structures were used to verify the algorithm. Compared to traditional printing, this method saved printing time by 22–36% and reduced material consumption by 20–30%. In addition, the surface roughness of the structure was also reduced to a certain extent. There are also some problems with our method, which may produce a staircase effect when printing some structures. In order to reduce the influence of the problem, we used an adaptive slicing algorithm to suppress the generation of the step effect, which effectively improved the surface finish of the structure.

Large-scale non-supporting printing and stepless printing restrict each other. Some researchers have proposed using slender nozzles to solve this problem, which means that some research on the nozzle needs to be done again. For example, explore the impact of nozzle shape on material extrusion performance, structural performance, path planning, etc. Our next work adopts the STL model as the slicing model. Multi-branch segmentation is achieved by extracting the concave points of the STL model. The free-form surface printing of overhanging structures is achieved through the layering of curved surfaces.

Author contribution All authors read and approved the published version of the manuscript. Pengfei Tang and Xianfeng Zhao came up with the idea for the study; Pengfei Tang built the experimental platform; Hongyan Shi helped draft the manuscript; all authors revised the paper.

Funding The authors acknowledge funding support from the National Natural Science Foundation of China (grant No. 51765009).

Availability of data and material All data generated or analyzed during this study are included in this manuscript.

Code availability The code for this paper cannot currently be shared as it is also part of ongoing research. We will share it on GitHub when the research is done.

Declarations

Ethics approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

Conflict of interest The authors declare no competing interests.

References

- Lewis JA, Ahn BY (2015) Three-dimensional printed electronics. *Nature* 518(7537):42–43. <https://doi.org/10.1038/518042a>
- Sun K, Wei TS, Ahn BY, Seo JY, Dillon SJ, Lewis JA (2013) 3D printing of interdigitated Li-ion microbattery architectures. *Adv Mater* 25(33):4539–4543. <https://doi.org/10.1002/adma.201301036>
- Goyanes A, Fina F, Martorana A, Sedough D, Gaisford S, Basit AW (2017) Development of modified release 3D printed tablets (printlets) with pharmaceutical excipients using additive manufacturing. *Int J Pharm* 527(1–2):21–30. <https://doi.org/10.1016/j.ijpharm.2017.05.021>
- Mannoor MS, Jiang Z, James T, Kong YL, Malatesta KA (2013) 3D printed bionic ears. *Nano letters* 13(6):2634–2639. <https://doi.org/10.1021/nl4007744>
- Jonušauskas L, Gailevičius D, Mikoliūnaitė L, Sakalauskas D (2017) Optically clear and resilient free-form μ -optics 3D-printed via ultrafast laser lithography. *Materials* 10(1):12. <https://doi.org/10.3390/ma10010012>
- Salet TA, Ahmed ZY, Bos FP, Laagland HL (2018) Design of a 3D printed concrete bridge by testing. *Virtual Phys Prototyp* 13(3):222–236. <https://doi.org/10.3390/ma10010012>
- Huang WC, Chang KP, Wu PH, Wu CH (2016) 3D printing optical engine for controlling material microstructure. *Phys Procedia* 83:847–853. <https://doi.org/10.1016/j.phpro.2016.08.088>
- Jiang J, Xu X, Stringer J (2018) Support structures for additive manufacturing: a review. *J Manuf Mater Process* 2(4):64. <https://doi.org/10.3390/jmmp2040064>
- Hu Q, Feng D, Zhang H, Yao Y, Aburaia M, Lammer H (2020) Oriented to multi-branched structure unsupported 3D printing method research. *Materials* 13(9):2023. <https://doi.org/10.3390/ma13092023>
- Sundaram R, Choi J (2004) A slicing procedure for 5-axis laser aided DMD process. *J Manuf Sci Eng* 126(3):632–636. <https://doi.org/10.1115/1.1763180>
- Lee K, Jee H (2015) Slicing algorithms for multi-axis 3-D metal printing of overhangs. *J Mech Sci Technol* 29(12):5139–5144. <https://doi.org/10.1007/s12206-015-1113-y>
- Wang M, Zhang H, Hu Q, Liu D, Lammer H (2019) Research and implementation of a non-supporting 3D printing method based on 5-axis dynamic slice algorithm. *Rob Comput-Integr Manuf* 57:496–505. <https://doi.org/10.1016/j.rcim.2019.01.007>

13. Liu B, Shen H, Deng R, Zhou Z, Fu J (2020) Research on GMAW based non-supporting thin-wall structure manufacturing. *Rapid Prototyping Journal* 27:333–345. <https://doi.org/10.1108/RPJ-05-2020-0102>
14. Zhao HM, He Y, Fu JZ, Qiu JJ (2018) Inclined layer printing for fused deposition modeling without assisted supporting structure. *Rob Comput-Integr Manuf* 51:1–13. <https://doi.org/10.1016/j.rcim.2017.11.011>
15. Wüthrich M, Elspass WJ, Bos P, Holdener S (2021) Novel 4-axis 3D printing process to print overhangs without support material. *Int Conf Add Manuf Products Appl* 130–145
16. Blackbelt (2017). <https://blackbelt-3d.com>
17. Wu C, Dai C, Fang G (2017) RoboFDM: a robotic system for support-free fabrication using FDM. In *Proceedings of the International Conference on Robotics and Automation (ICRA)* 1175–1180. <https://doi.org/10.1109/ICRA.2017.7989140>
18. Dai C, Wang C, Wu C (2018) Support-free volume printing by multi-axis motion. *ACM Trans Graph* 37:1–14. <https://doi.org/10.1145/3197517.3201342>
19. Papon J, Abramov A, Schoeler M (2013) Voxel cloud connectivity segmentation-supervoxels for point clouds. *Proceedings of the IEEE Conference On Computer Vision And Pattern Recognition* 2027–2034. <https://doi.org/10.1109/CVPR.2013.264>
20. Christoph Stein S, Schoeler M, Papon J (2014) Object partitioning using local convexity. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 304–311. <https://doi.org/10.1109/2FCVPR.2014.46>
21. Schoeler M, Papon J, Worgotter F (2015) Constrained planar cuts-object partitioning for point clouds. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 5207–5215. <https://doi.ieeecomputersociety.org/10.1109/CVPR.2015.7299157>
22. Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Comm ACM* 24(6):381–395. <https://doi.org/10.1016/B978-0-08-051581-6.50070-2>
23. Huang H, Wu S, Cohen-Or D, Gong M, Zhang H, Li G, Chen B (2013) L1-medial skeleton of point cloud. *ACM Trans. Graph* 32(4):65–1. <https://doi.org/10.1145/2461912.2461913>
24. Edelsbrunner H, Kirkpatrick D, Seidel R (1983) On the shape of a set of points in the plane. *IEEE Trans Inform Theory* 29(4):551–559. <https://doi.org/10.1109/TIT.1983.1056714>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.