IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 9, 2019, accepted April 25, 2019, date of publication May 1, 2019, date of current version May 15, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2914248

Efficient Free-Form Contour Packing Based on Code Matching Strategy

BAOSU GUO[1,2], YULONG JI[1], JINGWEN HU[1], FENGHE WU[1,2], AND QINGJIN PENG[3]

[1]College of Mechanical Engineering, Yanshan University, Qinhuangdao 066004, China
[2]Heavy-Duty Intelligent Manufacturing Equipment Innovation Center of Hebei Province, Qinhuangdao 066004, China
[3]Department of Mechanical Engineering, University of Manitoba, Winnipeg, MB R3T 5V6, Canada

Corresponding author: Qingjin Peng (Qingjin.Peng@umanitoba.ca)

This work was supported in part by the National Natural Science Foundation of China under Grant 51890881 and Grant 51605422, in part by the Natural Science Foundation of Hebei Province under Grant E2017203156 and Grant E2017203372, in part by the Science and Technology Projects of Universities in Hebei Province under Grant QN2017152, and in part by the Postdoctoral Science Foundation of Hebei Province, Jiangsu Key Laboratory of Precision and Micro-Manufacturing Technology, under Grant B2016003021.

ABSTRACT Freeform surfaces exist widely in the stock cutting process of clinical prosthesis preparation, aviation, ship, and other manufacturing industries. The free-form contours of surfaces need to be packed before they are machined from raw materials. The existing methods search a contour position by rotating the contour and translating it to connect other contours for packing. The relative position between two contours will be changed after the rotation as the contour description is lack of geometric invariance. These methods easily miss the best layout position resulting in interspaces in the raw material. Moreover, this result seriously reduces the performance and efficiency of an automatic packing system. Therefore, a new packing algorithm is proposed in this paper by combining the geometric invariant description and coding matching for contours to solve the contour rotating and position connecting problems. The optimal position of a contour can be found directly and then connected by the extracted similar complement features of the contour. The experimental results show that the proposed method can greatly improve quality and efficiency of the layout, especially in the material utilization.

INDEX TERMS Computer graphics, code matching, shape, free-form contour packing, freeman chain code, geometric invariant description, layout.

I. INTRODUCTION

The problem of 2D packing is widespread in industrial applications. According to whether the shape of a layout contour is regular or not, there are regular and irregular packing problems. The regular packing is relatively easy to locate regular contours such as rectangles and triangles [1]–[3]. As irregular contours are increasingly applied in fields of the aviation, automobile, clothing, and furniture, research on the layout of 2D irregular contours has been increasing [4]. For improving utilization of raw materials or capacity planning, a slight increase in the contour layout efficiency can generate great economic and environmental benefits [5], [6], such as reducing negative impacts of manufacturing on environments for green manufacturing [7]. The free-form surfaces are usually machined through computer numerical control (CNC) machining center. The machining boundaries of free-form surfaces can be represented by complex irregular polygons. Free-form contour packing problem can be regard as complex irregular packing problem.

The layout positioning strategy determines the packing contour position and final outcomes of material requirement planning [8]. Irregular contour positioning strategies can be divided into transformation method and trial solution algorithm (TSA). The former method transforms complex irregular contours into regular or simple irregular contours before contours are positioned. An example of this method is the envelope polygon packing algorithm that accomplishes the transformation using the smallest envelope polygon of irregular contours [9]. If the results are regular contours such as the smallest envelope rectangle, the regular contour positioning strategy will be adopted. Otherwise, the TSA will be used for the layout. This method only simplifies the problem of irregular packing since it does not solve the problem of positioning

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaogang Jin.

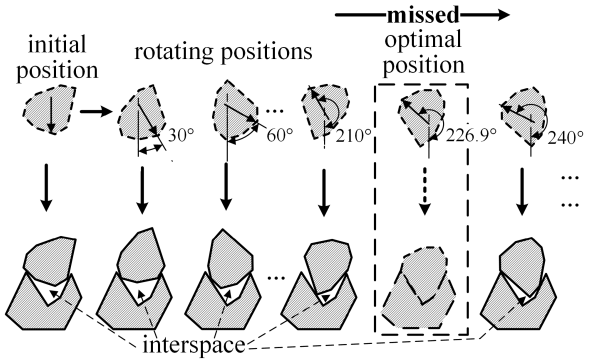

FIGURE 1. Illustration of finding the optimal position by TSA.

of irregular nesting [10]. In addition, such method has a limited ability to solve complex contour packing problems. TSA searches a series of layout contour positions by rotating each contour. An optimal position of the contours is then selected from these positions based on the principle of the lowest center of contour or the lowest height of packing. Examples of these methods include algorithms of no-fit polygons (NFP) [11] and discrete nesting and description of image pixels [12]. As the layout result of this method is affected by the rotating angle, the reduced angle is only considered to improve the material utilization, which results in the low efficiency of nesting. An optimal contour position is easily missed because of the contour initial position and improper angle used in this method. As shown in Fig. 1, the contour is rotated 12 times at 30° to find the optimal position using the TSA. However, the optimal position is found at 226.9°, a general angle that cannot be found by rotating the contour evenly at a smaller interval angle. Choosing the smaller interval angle will result in more searching time. Thus, most researchers have to choose a position based on some measures such as the contour center lowest principle. Under these methods, the cumulative interspace among contours causes a poor layout solution.

Although irregular packing problems can be solved effectively by the transformation method and TSA, it is challenging for the free-form contour packing problem. In this paper, a new packing algorithm is proposed based on the geometric invariant description and coding matching. The optimal position of a contour can be found directly and then connected by the extracted similar complement features of the contour. The proposed method can search an optimal contour position more quickly and efficiently than TSA. Experimental results show that the proposed method can greatly improve quality and efficiency of the layout, especially the average 15% improvement of material utilization is achieved. Related work of this research is reviewed in the next section. The proposed algorithm is expounded in Section 3. Experimental results of the proposed method and comparisons with the existing algorithm, TSA, are presented in Section 4. The conclusion and outlook of our method are discussed in Section 5.

II. RELATED RESEARCH

Different algorithms have been proposed for positioning problems but none of them can provide an optimal layout solution. Guo *et al.* [13] proposed a contour packing algorithm for free-form contour machining, which discretizes the packing area of materials based on the bottom-left and the lowest gravity center stray. This algorithm finds the position by collided the contour after rotating it, thus it belongs to the trial solution method (TSA) and takes a lot of time to obtain a satisfied result. Cherri *et al.* [14] and Leao *et al.* [15] used a mixed-integer linear programming (MIP) model to solve the complexity of geometry handling algorithms for the piece non-overlapping constraints. It improves robustness of modeling method but the complex process is difficult to apply in practice. Elkeran [16] proposed a method that hybridizes cuckoo search and guided local search optimization techniques, which clustered the group congruent polygons together to discover matched features among them. This algorithm can reduce the complexity of the nesting problem but it has to calculate the NFP to conduct intersection tests between pairs of polygons to confirm the connecting position. Thus the algorithm efficiency is lower. Wang *et al.* [17] proposed a customized branch-and-bound approach to address the nesting problem to guaranteed optimality of the layout, but the algorithm is very complex and difficult to apply in factories. In summary, the above algorithms have limitations in solving nesting problems because of defects of TSA and ignoring contour features.

There are some other methods that use heuristic and genetic algorithms to solve the irregular nesting problem. Elkeran [16] proposed the cuckoo search for a population meta-heuristic to group congruent polygons together in pairs and then place them. Del Valle *et al.* [9] proposed a GRASP based heuristic to solve the cutting stock problem with items of the irregular shape by combining with a column generation algorithm. Burke *et al.* [18] proposed a genetic programming contributing to a growing research area that represents a paradigm shift in search methodologies. Beyaz *et al.* [19] proposed an algorithm that introduces new crossover and mutation operators for the selection of the heuristics to obtain a high percentage of the optimal solutions. Hong *et al.* [20] proposed a mixed bin packing algorithm combined with a heuristic packing and the best fit algorithm as a hybrid heuristic algorithm based on iterative simulated annealing. The binary search was then developed to further improve results of their backtracking algorithm. Pinheiro *et al.* [21] presented a random-key genetic algorithm (RKGA) for the nesting problem combined with the well-known position rules. Therefore, increasingly sophisticated and complex heuristic approaches have been developed to address these problems. The role of heuristic approaches is globe searching. In other words, the heuristic approach is used to search an optimal order among all contours, it cannot locate the contour position. It has to combine with the traditional position principle to solve the nesting problem. They do not involve in improving the basic algorithm for contour positioning.

In summary, the exiting layout positioning methods rotate the contour to find the final layout position. As the contour description method does not possess geometric rotating invariance, each contour generates a new layout position with every rotation as shown in Fig. 1. These methods can generate confused and flexible data of the contour positions resulting in the low efficiency and poor material utilization [22]. If a method can be found to describe the contour with the invariance, there is no need to rotate contours. The optimal position of a contour can be found directly by the geometric invariant description and then connected by the extracted similar complement features of the contour.

III. PROPOSED METHODS

The process of the proposed algorithm is summarized as follows. A contour is first discretized into a polygon with some equaling units such as pixels. Relationships of adjacent pixels are established using the theory of geometric invariability of curves [23]. The contour then be encoded using Freeman Chain Code (FCC) based on the established relationships among adjacent pixels to extract contour shape features. After then, the longest common subsequence (LCS) is obtained by matching the code sequence to find the similar complement features of two contours. Positioning of the contour can be decided using the proposed hub-and-spoke algorithm. The outer boundary of contours nested in the raw material contour is treated as a new contour to match the next contour until all contours are nested in the raw material contour. There is no need to rotate the contour to find the optimal contour position. Contours can contact more closely because the contact mode becomes the line contact from the point contact using contour shape features. Experimental results show that the proposed algorithm can be used to improve the automatic nesting system in CNC Machining systems to obtain an optimal material requirement plan.

A. GEOMETRIC INVARIANT DESCRIPTION OF CONTOUS

The description of a contour has to be remained when it is transformed for positioning and connecting to others. As features extracted by the description are used to search optimal positions that can directly contact to other contours closely, the geometric invariance is an essential requirement for the description of contours. According to the theory of geometric invariability of curves, the description of curve vector parameterization can meet requirements of the geometric invariance.

1) VECTOR PARAMETERIZATION OF CURVE FOR THE GEOMETRIC INVARANCE

The geometric invariant description represents a shape with limited information such as a parabola with 3 points. If the relative relationship of these points is determined, the shape and its description will not be changed when it is rotated.

A parabola of $y = -0.5x^2 + x$ is shown in Fig. 2. After it is processed using the vector parameterization, a vector quadratic polynomial curve equation can be obtained to describe the shape as follows.

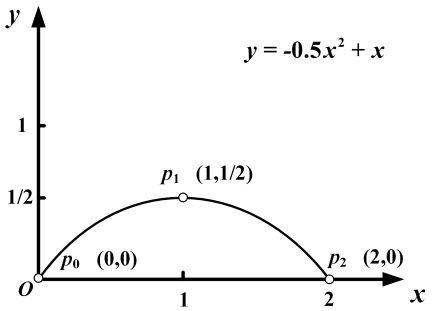

FIGURE 2. Illustration of a parabola of $y = -0.5x^2 + x$.

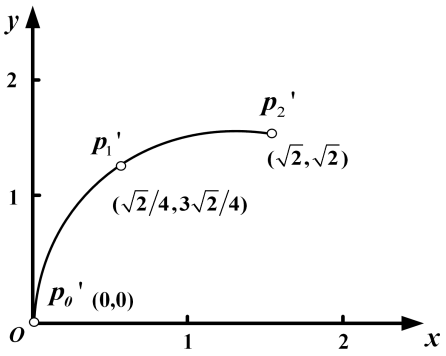

FIGURE 3. Rotated anticlockwise at 45°.

$$\begin{aligned} \boldsymbol{p}(u) &= \sum_{i=0}^{n} \boldsymbol{a}_i \varphi_i = \sum_{i=0}^{3} \boldsymbol{p}_i \varphi_i(u) \\ &= 2(u-0.5)(u-1)\,\boldsymbol{p}_0 - 4u(u-1)\,\boldsymbol{p}_1 \\ &\quad + 2u(u-0.5)\,\boldsymbol{p}_2 \end{aligned} \tag{1}$$

where $\boldsymbol{p}_0$, $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ indicate three position vectors that are also referred to as absolute vectors, φ_i expresses a basis function with parameter u. Inter-relationships among points of the curve have been determined by the basic function.

When the parabola is rotated anticlockwise at 45° as shown in Fig. 3, position vectors turn into $\boldsymbol{p}'_0$, $\boldsymbol{p}'_1$ and $\boldsymbol{p}'_2$. Multiplied with the rotation matrix based on (1), an equation is formed as follows.

$$\begin{aligned} \boldsymbol{p}(u) &= \sum_{i=0}^{n} \boldsymbol{a}_i \varphi_i = \sum_{i=0}^{3} \boldsymbol{p}_i{}' \varphi_i(u) \\ &= 2(u-0.5)(u-1)\boldsymbol{p}_0{}' - 4u(u-1)\boldsymbol{p}_1{}' \\ &\quad + 2u(u-0.5)\boldsymbol{p}_2{}' \end{aligned} \tag{2}$$

The shape has not been changed because the selected basic function possess the geometric invariance. In conclusion, the description method can express a shape with unchanged information when the shape position is changed. However, the shape described by a basis function that only possesses the full normativeness or partial normativeness has the geometric invariance. Thus, the basic function has three categories as follows.

1) Full normalized basic function
The basis function satisfies Cauchy Condition: $\sum_{i}^{n}\varphi_i \equiv 1$ such as the linear interpolation: $\boldsymbol{p}(u) = (1-u)\boldsymbol{p}_0 + u\boldsymbol{p}_1$.
2) Partial normalized basic function
The basis function satisfies the condition of $\sum_{i}^{k}\varphi_i \equiv 1, 0 \le k < n$, such as $\boldsymbol{p}(u) = \boldsymbol{a}_0 + \boldsymbol{a}_1 u$.
3) Basic function without normativeness
Other cases except the two cases mentioned above, such as $\boldsymbol{p}(u) = (1-u)^2\boldsymbol{p}_0 + u^2\boldsymbol{p}_1$.

2) EXPRESSION OF THE LAYOUT CONTOUR USING PARTIAL NORMALIZED BASIC FUNCTION

For a layout contour with the closed irregular polygon, the contour is expressed by partial normalized basic functions in this paper. The general formula of a vector function can be rewritten as follows.

$$\boldsymbol{p}(u) = \sum_{i=0}^{n}\boldsymbol{a}_i\varphi_i = \sum_{i=0}^{k}\boldsymbol{a}_i\varphi_i + \sum_{i=k+1}^{n}\boldsymbol{a}_i\varphi_i \tag{3}$$

where $(\boldsymbol{a_0}, \boldsymbol{a_1}, \boldsymbol{a_2}, \cdots\cdots, \boldsymbol{a_k})$ represents absolute vectors, $(\boldsymbol{a}_{k+1}, \boldsymbol{a}_{k+2}, \boldsymbol{a}_{k+3}, \cdots\cdots, \boldsymbol{a_n})$ are relative vectors, $\sum_{i}^{k}\varphi_i \equiv 1$. Let the rotation matrix be '$\boldsymbol{M}$' and translation vector be '$\boldsymbol{c}$'. Using (3), an equation after the rotation and translation is formed as follows.

$$\begin{aligned}
\boldsymbol{p}^*(u) &= \boldsymbol{p}(u)\boldsymbol{M} + \boldsymbol{c} \\
&= (\sum_{i=0}^{k}\boldsymbol{a}_i\varphi_i + \sum_{i=k+1}^{n}\boldsymbol{a}_i\varphi_i)\boldsymbol{M} + \boldsymbol{c} \\
&= \sum_{i=0}^{k}\boldsymbol{a}_i\varphi_i\boldsymbol{M} + \sum_{i=k+1}^{n}\boldsymbol{a}_i\varphi_i\boldsymbol{M} + \boldsymbol{c} \\
&= \sum_{i=0}^{k}\boldsymbol{a}_i\varphi_i\boldsymbol{M} + \sum_{i=0}^{k}\varphi_i\boldsymbol{c} + \sum_{i=k+1}^{n}\boldsymbol{a}_i\varphi_i\boldsymbol{M} \\
&= \sum_{i=0}^{k}(\boldsymbol{a}_i\boldsymbol{M} + \boldsymbol{c})\varphi_i + \sum_{i=k+1}^{n}(\boldsymbol{a}_i\boldsymbol{M})\varphi_i \\
&= \sum_{i=0}^{n}\boldsymbol{a}_i'^*\varphi_i
\end{aligned} \tag{4}$$

where $\boldsymbol{a}_i' = \boldsymbol{a}_i\boldsymbol{M} + \boldsymbol{c}$, $(i = 1, 2, 3, \cdots, k)$ and $\boldsymbol{a}_i' = \boldsymbol{a}_i\boldsymbol{M}$, $(i = k+1, k+2, \cdots, n)$. They are vectors rotated and translated. However, basis functions are as same as before. Therefore, the expression of a contour is not changed.

As shown in Fig. 4, a contour is expressed with the vector parameterization as follows.

$$\boldsymbol{p}(u) = \sum_{i=0}^{7}\boldsymbol{a}_i\varphi_i = \boldsymbol{a}_0\boldsymbol{\varphi}_0 + \boldsymbol{a}_1\boldsymbol{\varphi}_1 + \ldots + \boldsymbol{a}_7\boldsymbol{\varphi}_7 \tag{5}$$

Absolute vector $\boldsymbol{a}_0$ positions the contour in the coordinate system, relative vectors are a collection of

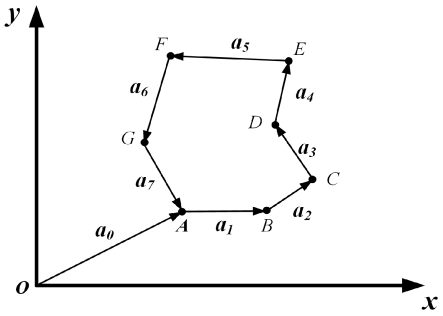

FIGURE 4. Schematic diagram of the geometric invariance of layout contour.

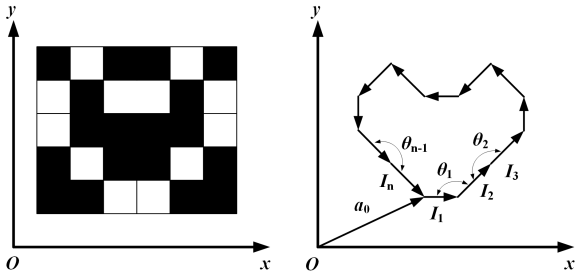

FIGURE 5. (a) Contour discretized into pixels. (b) Contour expressed by vectors.

$(\boldsymbol{a_1}, \boldsymbol{a}_2, \boldsymbol{a}_3, \boldsymbol{a}_4, \boldsymbol{a}_5, \boldsymbol{a}_6, \boldsymbol{a}_7)$ representing each edge of the contour. $(\boldsymbol{\varphi}_0, \boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \ldots, \boldsymbol{\varphi}_7)$ is a collection of basis functions for each corresponding vector.

Let $\boldsymbol{\varphi}_0$ equal to 1, (5) is expressed using partial normalized basic functions no matter what the other basis functions will be taken. But the other basis functions should be used to describe the contour shape. The next step is to establish the relationship between relative vectors as the basic function.

3) SELECTION AND FORMING OF THE BASIS FUNCTION

As the contour is complex, there is no exact mathematical equation to express it. The relationship of the adjacent vector has to be considered to form the contour geometric invariance. For an unequal-length edge of the contour, there are two parameters, length and relative angle of the adjacent relative vectors. Therefore, the entire contour is firstly discretized into polygons with some equilong units such as pixels to reduce one parameter, the length. As all pixels are same, the contour is expressed by the collection of relative vectors between the adjacent pixels as follows.

$$\boldsymbol{p}(u) = \sum_{i=0}^{7}\boldsymbol{a}_i\varphi_i = \boldsymbol{a}_0 + \boldsymbol{I}_1\boldsymbol{\varphi}_1 + \boldsymbol{I}_2\boldsymbol{\varphi}_2 + \boldsymbol{I}_3\boldsymbol{\varphi}_3 + \ldots + \boldsymbol{I}_n\boldsymbol{\varphi}_n \tag{6}$$

where $(\boldsymbol{I}_1, \boldsymbol{I}_2, \boldsymbol{I}_3, \boldsymbol{I}_4, \ldots, \boldsymbol{I}_n)$ and $(\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \ldots, \boldsymbol{\varphi}_n)$ are collections of adjacent relative vectors and basis functions, respectively, as shown in Fig. 5.

To describe the relationship, the rotation angle of adjacent vectors is chosen as the basis function parameter.

$$\begin{aligned} &\boldsymbol{I}_2\boldsymbol{\varphi}_2 = \boldsymbol{I}_1{}^*\boldsymbol{f}(\theta_1); \boldsymbol{I}_3\boldsymbol{\varphi}_3 = \boldsymbol{I}_2{}^*\boldsymbol{f}(\theta_2); \ldots; \\ &\boldsymbol{I}_n\boldsymbol{\varphi}_{\mathrm{n}} = \boldsymbol{I}_{\mathrm{n}-1}{}^*\boldsymbol{f}(\theta_{\mathrm{n}-1}) \end{aligned} \tag{7}$$

Bring (7) to the right of (6), a derivation process of the adjacent vector relationship is as follows.

$$\begin{aligned} &\left(\boldsymbol{I}_2\boldsymbol{\varphi}_2 + \boldsymbol{I}_3\boldsymbol{\varphi}_3 + , \ldots, + \boldsymbol{I}_{\mathrm{n}}\boldsymbol{\varphi}_{\mathrm{n}}\right) \\ &= (\boldsymbol{I}_1{}^*\boldsymbol{f}(\theta_1) + \boldsymbol{I}_2{}^*\boldsymbol{f}(\theta_2) + , \ldots, + \boldsymbol{I}_{\mathrm{n}-1}{}^*\boldsymbol{f}(\theta_{\mathrm{n}-1})) \\ &= (\boldsymbol{I}_1{}^*\boldsymbol{f}(\theta_1) + \boldsymbol{I}_1{}^*\boldsymbol{f}(\theta_1)\boldsymbol{f}(\theta_2) + , \ldots, \\ &\quad + , \boldsymbol{I}_1{}^*\boldsymbol{f}(\theta_1)\boldsymbol{f}(\theta_2)\ldots\boldsymbol{f}(\theta_{\mathrm{n}-1})) \\ &= \boldsymbol{I}_1{}^*(\boldsymbol{f}(\theta_1) + \boldsymbol{f}(\theta_1)\boldsymbol{f}(\theta_2) + , \ldots, + \boldsymbol{f}(\theta_1)\boldsymbol{f}(\theta_2)\ldots\boldsymbol{f}(\theta_{\mathrm{n}-1})) \\ &= \boldsymbol{I}_1{}^*\boldsymbol{F}(\theta_1, \theta_2, \ldots, \theta_{\mathrm{n}-1}) \end{aligned} \tag{8}$$

where $\boldsymbol{I}_1$ represents the original relative vector, the collection of $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \ldots, \boldsymbol{\theta}_{\mathrm{n}-1})$ indicates the rotation angle between adjacent vectors, $F(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \ldots, \boldsymbol{\theta}_{\mathrm{n}-1})$ shows the relationship between adjacent pixels.

According to (8), the contour can be expressed by the original relative vector and established relationships based on angles. When a contour is rotated, the original relative vector is multiplied with the rotation matrix. But the description of adjacent vector relationships will not be changed.

In summary, $(\boldsymbol{a}_0, \boldsymbol{I}_1, \boldsymbol{I}_1^* F(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \ldots, \boldsymbol{\theta}_{n-1}))$ is used to describe a contour. This method has the geometric invariance expressed by absolute vectors, relative vectors and basis functions.

B. EXTRACTION OF CONTOUR SHAPE FEATURES FOR CONTOURS' CONNECTION AND POSITION

After forming the geometric invariance of the contour, the next step is extracting contour shape features by encoding the contour with the geometric invariance for the contour connection more closely and quickly to reduce interspaces among contours. The features are extracted by encoding contours. An extracting method is developed for the geometric invariant description.

There are many methods that can encode a contour such as Wavelet Transform and Fourier Transform [24]. But results from these two methods do not possess the geometric invariance. Chain Code is a common method for contour coding. Compared to the above two methods, the Chain Code can extract the feature information with the geometric invariance. Freeman Chain Code (FCC) is a sequence to express indexical relations between adjacent units after a contour is discretized into pixels. The pixels and their relationship are treated as equilong units and basis functions, respectively. After taking the difference of FCC, a new generated sequence (DFCC) can represent the contour with shape information of the geometric invariance.

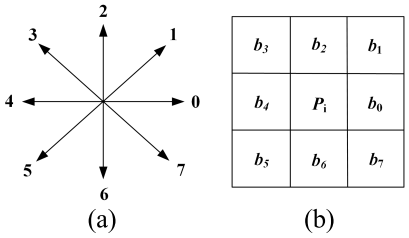

FIGURE 6. (a) Value of the chain code. (b) Direction of the chain code.

1) DIFFERENCE FREEMAN CHAIN CODE (DFCC) FOR THE CONTOUR REPRESENTATION

The direction that a pixel points to the next adjacent pixel is called a chain. There are 8 pixels around the central pixel as shown in Fig. 6 (b), In other words, there are 8 kinds of chains. Values of them are shown in Fig. 6 (a). FCC is generated by tracking the contour expressed by the pixel. A flow diagram of generating the normalized DFCC is shown in Fig.7.

As shown in Fig. 7, a contour is tracked. FCC is obtained to describe shape features of the contour but there is no geometric invariance. The general expression of the FCC is as follows.

$$\boldsymbol{M} = \boldsymbol{S} \overset{n}{\underset{i=1}{\boldsymbol{C}}} \boldsymbol{a}_i = \boldsymbol{S}a_1\boldsymbol{a}_2\boldsymbol{a}_3\cdots\boldsymbol{a}_n, \boldsymbol{a}_i \in \boldsymbol{Z}[0, 7] \tag{9}$$

where S indicates the coordinate of an original pixel, which is equivalent to the absolute vector. $\boldsymbol{a}_{\mathbf{i}}$ is a chain that expresses the direction between adjacent pixels, $\overset{n}{\underset{i=1}{\boldsymbol{C}}} \boldsymbol{a}_i$ expresses the chain sequence.

However, the relationship between adjacent pixels has not been established because the chain value is independent, in other words, every chain relies on the absolute coordinate system. The relationship is built by taking the difference as follows:

$$\begin{cases} \boldsymbol{a}'_1 = (\boldsymbol{a}_1 - \boldsymbol{a}_n) \bmod \boldsymbol{N}, \boldsymbol{i} = 2, 3, \ldots \boldsymbol{n} \\ \boldsymbol{a}'_i = (\boldsymbol{a}_i - \boldsymbol{a}_{i-1}) \bmod \boldsymbol{N}, \boldsymbol{i} = 2, 3, \ldots \boldsymbol{n} \\ \boldsymbol{M}' = \boldsymbol{S}' \overset{n}{\underset{i=1}{\boldsymbol{C}}} \boldsymbol{a}'_i, \quad \boldsymbol{i} = 2, 3, \ldots \boldsymbol{n} \end{cases} \tag{10}$$

where 'mod' represents the modular arithmetic and the value of $\boldsymbol{N}$ represents number of the chain kind. $\boldsymbol{S}'$ equals to $\boldsymbol{S}$. After the operation, DFCC is represented by $\boldsymbol{M}'$.

For the only representation of a contour with the sequence, the sequence is processed with the normalization. Above all, the DFCC sequence is treated as a positive integer with n digits. The sequence is then searched corresponding to the minimum positive integer [25].

After the differential and normalized processing, a sequence is obtained with feature information of the contour. In addition, this sequence satisfies the requirement of the geometric invariant description as the relationship between the adjacent units has been established.

As shown in Fig. 8, diagrams (a) and (b) show the original and rotated at 90° anticlockwise contours expressed by the chain code, respectively. It is found that the normalized DFCC of Figs 8. (a) and (b) are the same in Table 1. Thus the

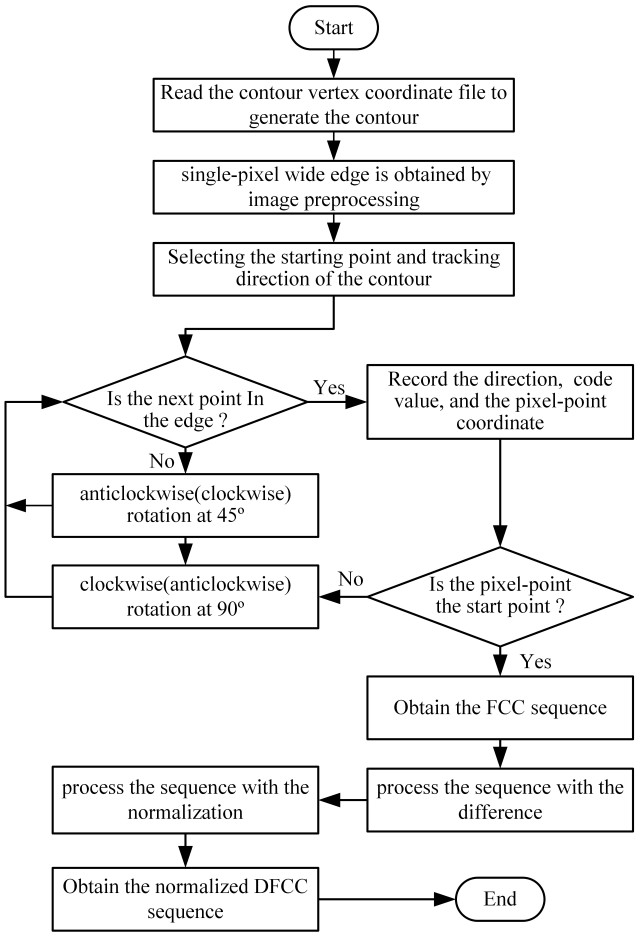

FIGURE 7. Flow diagram of generating normalized DFCC.

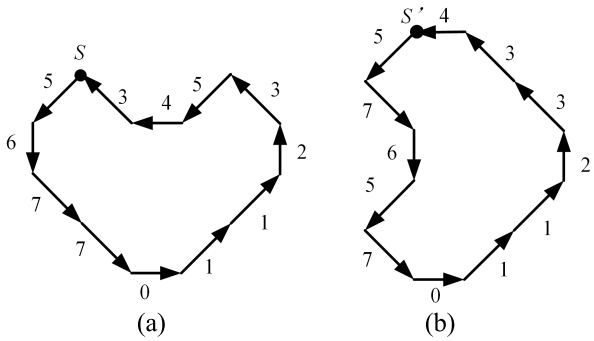

FIGURE 8. (a) Original chain code. (b) The transformational contour.

relationship established between adjacent pixels by the normalized DFCC possess the geometric invariance.

2) TWO CONTOURS CONNECTION

As a contour is represented by DFCC, the connection problem has been transformed into the sequence alignment problem to search the longest common subsequence (LCS) of two sequences that are respectively obtained by encoding two contours according to clockwise and anticlockwise. LCS reflects the matching position with the combination of shape features of tow contours. Each contour will be finally connected by matching parameters calculated by LCS [26], [27].

TABLE 1. Comparison of three chain codes.

	Chain code	Difference chain code	Normalized difference chain code
Fig.8(a)	(3)567701123543	211011011277	011011277211
Fig.8(b)	(4)576570112334	127721101101	011011277211

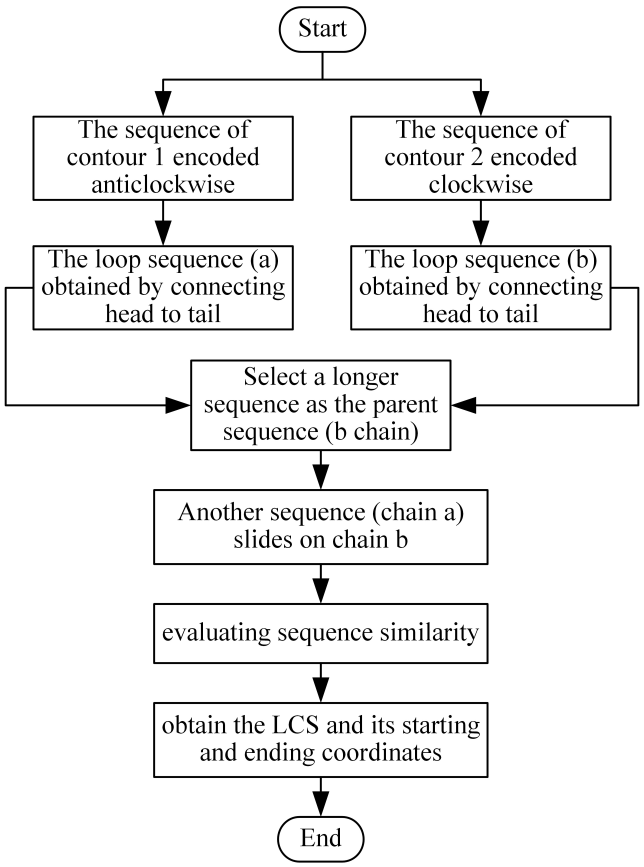

FIGURE 9. Flowchart of the sequence alignment.

Because the number of chains is large, a cycle search algorithm is developed for the sequence alignment. A flowchart of the algorithm is shown in Fig. 9.

From the sequence alignment, it is found that the most similar part of two contours includes the origin-termination coordinate of the LCS.

The two pairs of coordinates are regarded as two vectors: $\boldsymbol{a}$ and $\boldsymbol{b}$, as shown in Fig. 10. Transformation parameters are then calculated according to vectors as shown in Fig. 11.

Where the rotation parameter θ, and translation parameters t_x and t_y are calculated as follows.

$$\begin{cases} t_x = x_A - x'_A \\ t_y = y_B - y'_B \end{cases} \tag{11}$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} * \begin{pmatrix} x - t_x \\ y - t_y \end{pmatrix} \tag{12}$$

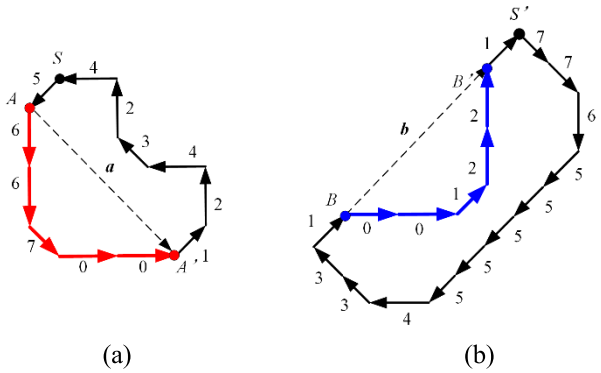

FIGURE 10. (a) LCS of two contours encoded in anticlockwise. (b) LCS of two contours encoded in clockwise.

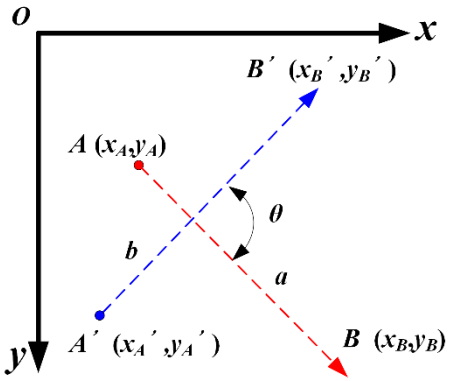

FIGURE 11. The angle between the vectors.

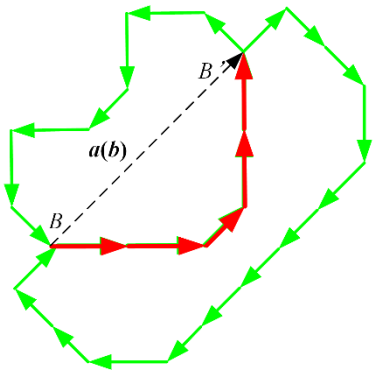

FIGURE 12. Schematic diagram of finding LCS.

After the parameters are calculated, the two contours are connected based on the coordinate transformation matrix using Eqs. (11) and (11). Where (x', y') are new coordinates of the contour, (x, y) indicates the original coordinate of the contour. The result of connection is shown in Fig. 12.

If connected contours overlap in the matching part, contour 1 will be translated in pixels according to the normal direction of vector ($\boldsymbol{b}$) until connected contours are next to each other. Finally, the connected contour is treated as a new standard contour. The outer boundary of the contour is extracted to match other contours to complete the layout of all contours.

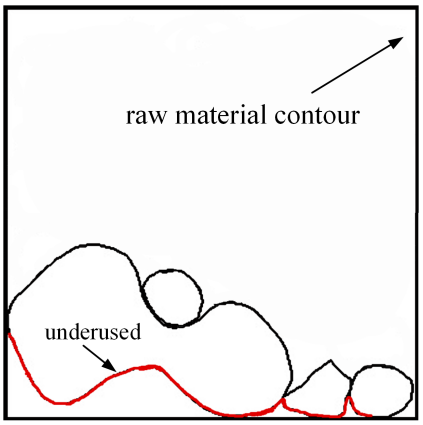

FIGURE 13. Result of traditional positioning algorithm.

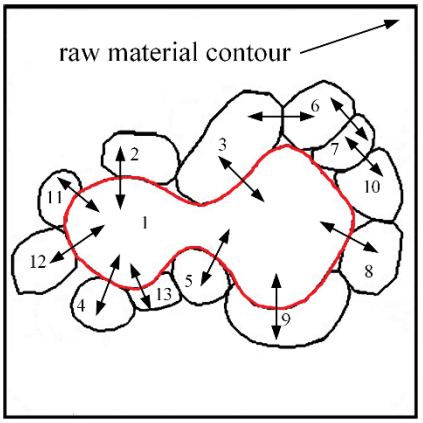

FIGURE 14. Result of the hub-and-spoke algorithm.

3) POSITIONING OF THE NESTED CONTOUR BY HUB-AND-SPOKE ALGORITHM

Although two contours have been connected each other, they have not been positioned in the raw material contour. The hub-and-spoke algorithm is proposed for positioning.

The traditional positioning algorithms, such as those using the Bottom-left principle [28]–[30] and the lowest center principle, place the contour with the contour boundary of raw material, which results in some parts of the contour cannot be used as shown in Fig. 13. The contour marked in red is underused.

The proposed algorithm places the contour using the only contour itself. The boundary of the raw material contour is regarded as a constraint to ensure the connected contour is in the raw material contour as shown in Fig. 14. Numbers in the figure express the nesting order. Arrows indicate the positioning of contours. It can be found that contour number 1 is fully used and the space between contours is reduced observably using the hub-and-spoke algorithm by following steps.

Step 1: A largest and most complex contour is chosen as the original contour. Its centroid coordinates are calculated. The contour is translated to the center of the raw material contour.

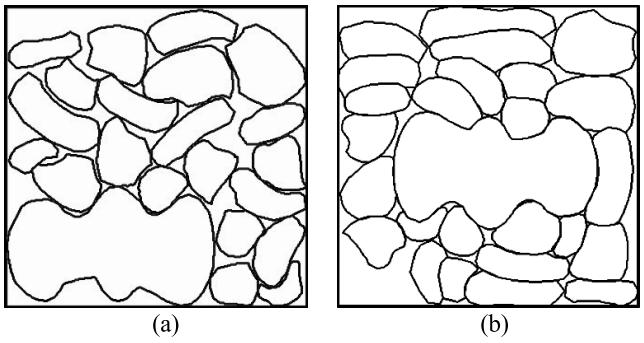

FIGURE 15. Comparison results of sample 1. (a) Nesting result of TSA (15°). (b) Result of the proposal method.

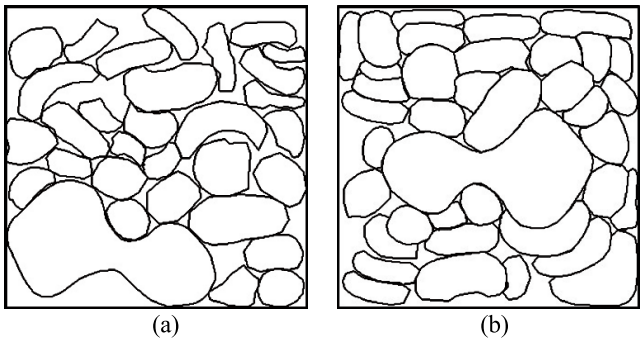

FIGURE 16. Comparison results of sample 2. (a) Nesting result of TSA (15°). (b) Result of the proposal method.

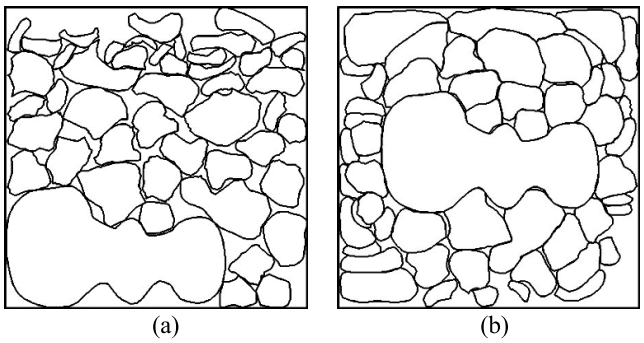

FIGURE 17. Comparison results of sample 3. (a) Nesting result of TSA (15°). (b) Result of the proposal method.

Step 2: Other contours are connected around the original contour one after another in the order through the proposed algorithm. After a contour is connected with the original contour, the outer boundary will be extracted from connected contours to match other contours until all of them has been nested in the raw material contour.

Step 3: When two contours are connected each other, their outer boundaries are checked for whether beyond the raw material contour. If yes, the contours are rotated as a whole around the center. Because of the geometric invariance of the contour, the expression information of the contour will remain.

The above steps are repeated to position all of the contours. Features of each contour are used for contours connections closely to reduce interspaces among contours.

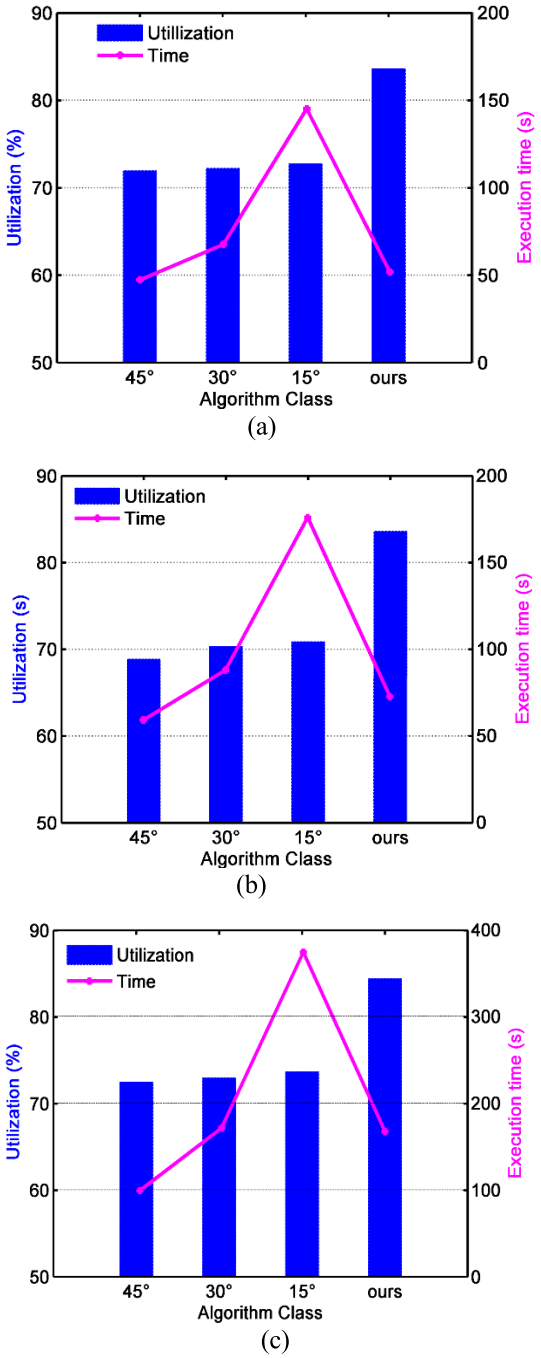

FIGURE 18. (a) Statistical nesting results using two algorithms for sample 1, (b) sample 2, and (c) sample 3.

However, because of neglecting the shape of the raw material contour, this algorithm is not applicable in a free-shape contour of raw material. This problem is to be solved in the future work.

IV. EXPERIMENTAL RESULTS

In order to verify the feasibility and effectiveness of the proposed algorithm, the algorithm is implemented using Matlab and tested by three instances with different amounts and shapes of contours chosen from freeform surfaces of plane

TABLE 2. Computational results of two algorithms.

Instances (total number)	Algorithm Class	Angle (°)	Capacity	Utilization (%)	Execution time(s)
Sample 1 (26)	TSA	45	22	71.95	47.26
		30	22	72.23	67.51
		15	22	72.74	144.94
	ours	-	26	83.62	51.69
Sample 2 (38)	TSA	45	30	68.88	58.95
		30	31	70.36	87.72
		15	31	70.87	175.67
	ours	-	38	83.61	72.44
Sample 3 (53)	TSA	45	47	72.49	99.4
		30	47	72.95	171.31
		15	48	73.69	374.04
	ours	-	53	84.42	167.36

projections. As testing for improvement of the layout positioning strategy is the purpose of the experiments, contours' order is equally treated. A comparison is conducted to find the optimal position of each contour using the proposed algorithm and trial solution algorithm (TSA). The TSA represents a class of methods that find an optimal position by the contour rotation based on some principles such as the lowest center principle. These methods include NFP, discrete nesting, and image pixel algorithms. For the comparison, each contour of the sample is rotated at interval angles, 15°, 30° and 45°, respectively.

Figs. 15-17 show layout results of three instances. Computational and comparison results of material utilizations are shown in Table 2. Fig. 18 is statistical results of the material utilization and execution time of all instances using the two methods.

From experimental results shown in Figs. 15-17, it is obvious that our algorithm achieves a complementary of contour shape features. Spaces between contours are reduced observably. In addition, the average 15% improvement of material utilization is achieved based on data in Table 1. Fig. 18 shows that our algorithm has the better performance on nesting than that of TSA.

From Fig. 18, it can be found that the relation of decreased rotating angles and increased time is nonlinear. As the angle becomes smaller, the processing time increases faster and the material utilization increases little. In sample 1, the angle is changed from 45° to 30°, the time is increased by 43% according to the data in Table 2. When the angle is decreased by 15°, the processing time is increased by 115% based on data in Table 2. By comparing results of the three samples, we also find that the TSA has less ability to increase the material utilization by reducing the angle. Furthermore, the similar statistical results of three samples prove the stability of our algorithm. In conclusion, the algorithm proposed in this paper has achieved the expected goal for increasing the material utilization and reducing the execution time.

V. CONCLUSION AND OUTLOOK

Existing nesting algorithms rotate contours for positioning, which cannot easily find the optimal positions of contours. There are many interspaces left among the contours. As shown in Fig. 15 (a). The proposed algorithm can position contours contact closely in the line contact using the contour shape feature with the geometric invariant description for the complementary combination of concave and convex features.

The optimal position of each contour can be found quickly and accurately by shape features using the nesting algorithm proposed based on invariance of the contour description. There is no need to rotate the contour to find the optimal contour position. Contours can contact more closely in the line contact using contour shape features. Therefore, the proposed method improves the strategy of nesting positioning in the execution efficiency to promote automation of the irregular nesting system. The similarity of contours will be assessed further to improve the versatility and practicality of the nesting system in the future work.

REFERENCES

[1] Y. Wang and L. Chen, "Two-dimensional residual-space-maximized packing," *Expert Syst. Appl.*, vol. 42, no. 7, pp. 3297–3305, 2015.

[2] A. Fernández, C. Gil, R. Baños, and M. G. Montoya, "A parallel multi-objective algorithm for two-dimensional bin packing with rotations and load balancing," *Expert Syst. Appl.*, vol. 40, no. 13, pp. 5169–5180, 2013.

[3] L. R. Mundim, M. Andretta, and T. A. de Queiroz, "A biased random key genetic algorithm for open dimension nesting problems using no-fit raster," *Expert Syst. Appl.*, vol. 81, pp. 358–371, Sep. 2017.

[4] T. C. Martins and M. de Sales Guerra Tsuzuki, "Simulated annealing applied to the irregular rotational placement of shapes over containers with fixed dimensions," *Expert Syst. Appl.*, vol. 37, no. 3, pp. 1955–1972, 2010.

[5] E. López-Camacho, H. Terashima-Marin, P. Ross, and G. Ochoa, "A unified hyper-heuristic framework for solving bin packing problems," *Expert Syst. Appl.*, vol. 41, no. 15, pp. 6876–6889, 2014.

[6] A. Bouganis and M. Shanahan, "A vision-based intelligent system for packing 2-D irregular shapes," *IEEE Trans. Autom. Sci. Eng.*, vol. 4, no. 3, pp. 382–394, Jul. 2007.

[7] M. Vanzela, G. M. Melega, S. Rangel, and S. A. de Araujo, "The integrated lot sizing and cutting stock problem with saw cycle constraints applied to furniture production," *Comput. Oper. Res.*, vol. 79, pp. 148–160, Mar. 2017.

[8] A. K. Sato, T. C. de Martins, and M. S. G. de Tsuzuki, "A pairwise exact placement algorithm for the irregular nesting problem," *Int. J. Comput. Integr. Manuf.*, vol. 29, no. 11, pp. 1177–1189, Nov. 2016.
[9] A. M. Del Valle, T. A. de Queiroz, F. K. Miyazawa, and E. C. Xavier, "Heuristics for two-dimensional knapsack and cutting stock problems with items of irregular shape," *Expert Syst. Appl.*, vol. 39, no. 16, pp. 12589–12598, 2012.
[10] E. K. Burke, R. S. R. Hellier, G. Kendall, and G. Whitwell, "Complete and robust no-fit polygon generation for the irregular stock cutting problem," *Eur. J. Oper. Res.*, vol. 179, no. 1, pp. 27–49, 2007.
[11] D. Domovic, T. Rolich, D. Grundler, and S. Bogovic, "Algorithms for 2D nesting problem based on the no-fit polygon," in *Proc. 37th Int. Conv. Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, May 2014, pp. 1094–1099.
[12] W. K. Wong, X. X. Wang, P. Y. Mok, S. Y.-S. Leung, and C. K. Kwong, "Solving the two-dimensional irregular objects allocation problems by using a two-stage packing approach," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 3489–3496, 2009.
[13] B. Guo, Q. Peng, X. Cheng, and N. Dai, "Free-form contour packing based on material grid approximation and lowest-gravity-center methods," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 1864–1871, 2015.
[14] L. H. Cherri, L. R. Mundim, M. Andretta, F. M. B. Toledo, J. F. Oliveira, and M. A. Carravilla, "Robust mixed-integer linear programming models for the irregular strip packing problem," *Eur. J. Oper. Res.*, vol. 253, no. 3, pp. 570–583, 2016.
[15] A. A. S. Leao, F. M. B. Toledo, J. F. Oliveira, and M. A. Carravilla, "A semi-continuous MIP model for the irregular strip packing problem," *Int. J. Prod. Res.*, vol. 54, no. 3, pp. 712–721, Feb. 2016.
[16] A. Elkeran, "A new approach for sheet nesting problem using guided cuckoo search and pairwise clustering," *Eur. J. Oper. Res.*, vol. 231, no. 3, pp. 757–769, Dec. 2013.
[17] A. Wang, C. L. Hanselman, and C. E. Gounaris, "A customized branch-and-bound approach for irregular shape nesting," *J. Global Optim.*, vol. 71, no. 4, pp. 935–955, 2018.
[18] E. K. Burke, M. Hyde, G. Kendall, and J. Woodward, "A genetic programming hyper-heuristic approach for evolving 2-D strip packing heuristics," *IEEE Trans. Evol. Comput.*, vol. 14, no. 6, pp. 942–958, Dec. 2010.
[19] M. Beyaz, T. Dokeroglu, and A. Cosar, "Robust hyper-heuristic algorithms for the offline oriented/non-oriented 2D bin packing problems," *Appl. Soft Comput.*, vol. 36, pp. 236–245, Nov. 2015.
[20] S. Hong, D. Zhang, H. C. Lau, X. Zeng, and Y.-W. Si, "A hybrid heuristic algorithm for the 2D variable-sized bin packing problem," *Eur. J. Oper. Res.*, vol. 238, no. 1, pp. 95–103, 2014.
[21] P. R. Pinheiro, B. Amaro, Jr., and R. D. Saraiva, "A random-key genetic algorithm for solving the nesting problem," *Int. J. Comput. Integr. Manuf.*, vol. 29, no. 11, pp. 1159–1165, Nov. 2016.
[22] A. Martinez-Sykora, R. Alvarez-Valdés, J. A. Bennell, R. Ruiz, and J. M. Tamarit, "Matheuristics for the irregular bin packing problem with free rotations," *Eur. J. Oper. Res.*, vol. 258, no. 2, pp. 440–455, 2017.
[23] W. Hu, Z. Chen, H. Pan, Y. Yu, E. Grinspun, and W. Wang, "Surface mosaic synthesis with irregular tiles," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 3, pp. 1302–1313, Mar. 2016.
[24] Z. Dong-Bao, H. E. Tian, and Z. Ka, "Shape contour description and matching method based on complex moments," *J. Sichuan Univ., Eng. Sci. Ed.*, vol. 43, pp. 109–115, 2011.
[25] F. Janan and M. Brady, "Shape description and matching using integral invariants on eccentricity transformed images," *Int. J. Comput. Vis.*, vol. 113, no. 2, pp. 92–112, 2015.
[26] S. W. Bae and I. Lee, "On finding a longest common palindromic subsequence," *Theor. Comput. Sci.*, vol. 710, pp. 29–34, Feb. 2018.
[27] S. Kuo and G. R. Cross, "An improved algorithm to find the length of the longest common subsequence of two strings," *ACM SIGIR Forum*, vol. 23, nos. 3–4, pp. 89–99, 1989.
[28] B. S. Baker, E. G. Coffman, Jr., and R. L. Rivest, "Orthogonal packings in two dimensions," *SIAM J. Comput.*, vol. 9, no. 4, pp. 846–855, 1980.
[29] D. Liu and H. Teng, "An improved BL-algorithm for genetic algorithm of the orthogonal packing of rectangles," *Eur. J. Oper. Res.*, vol. 112, no. 2, pp. 413–420, 1999.
[30] B. Chazelle, "The bottomn-left bin-packing heuristic: An efficient implementation," *IEEE Trans. Comput.*, vol. C-32, no. 8, pp. 697–707, Aug. 1983.

BAOSU GUO received the B.S. degree in mechanical engineering from Naval Aeronautical and Astronautical University, Qingdao, in 2009, and the Ph.D. degree in aerospace manufacturing engineering from the Nanjing University of Aeronautical and Astronautics, Nanjing, China, in 2015.

He is currently a Lecturer with the School of Mechanical Engineering, Yanshan University. His research interests include intelligent manufacturing, big data, and image processing.

YULONG JI received the B.S. degree in mechanical engineering from the Taiyuan University of Science and Technology, Taiyuan, in 2016. He is currently pursuing the master's degree in mechanical engineering with Yanshan University, Qinhuangdao, China. His research interests include intelligent manufacturing and intelligent irregular nesting systems.

JINGWEN HU received the B.S. degree in mechanical engineering from the Wuhan University of Bioengineering, Wuhan, in 2016. He is currently pursuing the M.S. degree in mechanical engineering with Yanshan University, Qinhuangdao, China. His research interests include machine learning, deep learning, computer vision, and intelligent irregular nesting systems.

FENGHE WU received the B.S. and M.S. degrees in mechanical engineering from Yanshan University, Qinhuangdao, in 1991 and 1998, respectively, and the Ph.D. degree in aerospace manufacturing engineering from Beihang University, Beijing, China, in 2006.

He is currently a Professor with the School of Mechanical Engineering, Yanshan University. His research interests include digital manufacturing, intelligent manufacturing, bionic design, and optimized design.

QINGJIN PENG received the B.S. and M.S. degrees in mechanical and manufacturing engineering from Xi'an Jiaotong University, Xi'an, China, in 1982 and 1988, respectively, and the Ph.D. degree in mechanical and manufacturing engineering from Birmingham University, Birmingham, U.K., in 1998.

He is currently a Professor with the Department of Mechanical Engineering, University of Manitoba, Canada. His research interests include virtual manufacturing, product design, and image-based methods in reverse engineering.