Path planning and pose correction of robot laser cleaning process for specific surfaces of parts

Zhigui Pan
DeFu Liu (liudefu@mail.csu.edu.cn)
Central South Univeristy
Shudan Li
Zixin Deng
Jian Liu
Tao Chen

Research Article

Keywords: Complex surface, Laser cleaning, Stereo lithography model, Region segmentation, Path planning, Robot pose

Posted Date: September 16th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-2049236/v1

Path planning and pose correction of robot laser cleaning process for specific surfaces of parts

Zhigui Pan[a], Defu Liu[a, b, *], Shudan Li[a], Zixin Deng[a], Jian Liu[a], Tao Chen[c]

a *College of Mechanical and Electrical Engineering, Central South University, Changsha, 410083, China*

b *State Key Laboratory of High Performance Complex Manufacturing, Changsha, 410083, China*

c *School of Mechanical Engineering and Rail Transit, Changzhou University, Changzhou, 213164, China*

* Correspondence: liudefu@csu.edu.cn; Tel.: +86-731-88879351

Abstract: The purpose of this paper is to solve a series of problems, such as the difficulty in accurately controlling the specific surfaces, and the interference between the laser beam and the parts, when the laser is used to clean mechanical parts with complex surfaces by using robots. Firstly, a region segmentation algorithm of the stereo lithography (STL) model was designed, and the feature surfaces of the model were obtained by using the dihedral angle as the criterion. Then the path points which could easily cause interference between the laser beam and the part to be cleaned were detected by the method of ray intersection, and the pose of the robot was modified according to the normal vector of two non-interference points that were adjacent to the continuous interference points respectively. Finally, the proposed method was validated by scanning path visualization and robot laser cleaning process experiments. The results showed that the robot can successfully clean the specific surface of the test workpiece without interference by following the laser cleaning path generated by the proposed algorithm, and the transition of the robot pose is smooth, which proves the effectiveness of this study. The method is also useful for the path planning of robots in other processing fields.

Keywords: Complex surface, Laser cleaning, Stereo lithography model, Region segmentation, Path planning, Robot pose

1 Introduction

Laser cleaning is a rapidly developing cleaning technology in recent years. Its working mechanism is that a series of changes in light, heat and force occur in a very short time by laser beam, and the contaminants can be removed under the coupling effect of these changes [1-5]. At present, portable hand-held laser cleaning equipment and three-dimensional (3D) mobile platform laser cleaning equipment are mainly used for cleaning operations in the laser cleaning industry, both of which have their own advantages and disadvantages. Portable

hand-held laser cleaning equipment is flexible and convenient. However, when the surface to be cleaned is constantly changing, stable scanning speed, defocus distance and angle of incidence cannot be guaranteed, so stable and controllable surface cleaning quality cannot be ensured [6]. At the same time, the portable laser cleaning equipment is difficult to be applied to large workpiece cleaning because of low cleaning efficiency [7]. 3D mobile platform laser cleaning equipment can ensure the stable speed, constant defocus distance and angle of incidence, but it is difficult to complete the free cleaning for the parts with complex curved surfaces[8]. With the rapid development of robotics and computer numerical control (CNC) technology, multi-joint robots are widely used in the laser processing fields such as laser cladding, laser deposition and laser welding due to their high degrees of freedom, strong flexibility and easy programming control and other characteristics [9-11].

Path planning for robot machining is the primary requirement to realize the robot automatic machining. The traditional manual teaching programming relies on the experience and operation of technicians to write processing programs, which is time-consuming, inefficient, and difficult to complete complex path programming[12, 13]. Therefore, scholars proposed corresponding robot path planning methods based on the characteristics of different processing methods. Zhang et al. [14] proposed an adaptive slicing algorithm to generate a variable layer thickness based on the local surface geometry and a pre-specific smoothness requirement, which better solved the problem of feature loss caused by the abrupt variety of curvature in the layering direction for parts with complex surfaces. Zheng et al. [15] proposed an interpolation-based cladding path generation method for the intersection points generated by each slice plane, that is, the interpolation operation was determined based on the pose change between two adjacent points, and the obtained path points not only ensured the cladding layer shape but also reduced the scale of processing point. Hu et al. [16] divided the NURBS surface into several simple sub-regions, and then offset the center-of-mass contour line of each sub-region to obtain the processing path covering this region, which alleviated the staircase effect in the laser deposition process. Yang et al. [17] fitted the point cloud data based on the cubic spline function and generated the laser welding seam path with good consistency and smoothness with the actual seam trace. The machining path points of robots are generated by the above path planning methods through the layered slicing of the STL model, the offset of the NURBS surface center-of-mass contour line, and the point cloud data fitting, etc. Among them, the STL model has the characteristics of simple format, easy to read and display, and can exchange data with most computer-aided design (CAD) systems in engineering applications such as SolidWorks, Pro/E, CATIA, etc., which enhances the universality of the algorithm and reduces the constraints

of CAD system categories, so it is widely used as a source file for path planning[18-20].

In the field of laser cleaning, some scholars have also researched the automatic cleaning of parts by combining it with offline programming techniques. Ma et al. [21] used line structured light and camera to obtain 3D point cloud data of the workpiece surface and converted them into the STL model, and then achieved automatic generation of cleaning paths by combining with slicing technology. Jin et al. [22] indirectly obtained the STL model of the workpiece with the help of 3D scanning equipment, and used commercial offline programming software to carry out the laser cleaning path planning, which verified the effectiveness of robot laser cleaning. For simple parts, the conventional robot path planning methods can be used in the path planning of robot laser cleaning, however, in some cases, existing robot path planning techniques cannot meet the demand for precision laser cleaning operations, for the following reasons:

(1) In actual industrial applications, only one or a few specific surfaces of the parts may need to be cleaned, and the contaminants on different surfaces may also require different process parameters to be set. However, it is difficult to accurately obtain the specific surface of the part by 3D scanner, and there are few reports on the method that extracting the specific surface from the STL model for path planning. Therefore, it is of great significance to study the path planning method for specific surfaces of the STL model to improve the flexibility of laser cleaning and increase the universality of its application.

(2) Generally, the center line of the laser beam coincides with the normal vector of the processing point when the robot performs surface laser processing [23, 24], which can not only ensure processing quality but also facilitates control. However, this vertical processing method can easily lead to interference between the workpiece and the laser beam, or the laser is blocked. When studying the damaged area of parts repaired by robot laser cladding, Liu et al. [25] proposed a method of creating a collision region along the beam direction, performing a Boolean operation with the original model for interference detection, and considering the beam radius compensation error for pose correction, but this method was only applicable to regular paths. For laser cleaning, the literature [26, 27] pointed out that the laser slanting on the surface of a part may not degrade the cleaning quality and can even improve the cleaning efficiency. Therefore, it is necessary to propose a more applicable path interference detection and correction method to ensure the safety of the robot laser cleaning process.

In this paper, based on the above two problems, a novel region segmentation method of the STL model is designed to divide the part surface into a series of feature surfaces. A WinForm form application based on .net framework and C# is developed to freely control the feature surfaces of the STL model, and the robot cleaning

path is generated by the slicing algorithm. Then, the ray detection method is used to detect the interference of the cleaning path, and the robot pose is modified to avoid interference. Finally, the laser cleaning of the test workpiece is completed according to the robot program, which proves the proposed laser cleaning path planning method for specific surfaces is reliable and effective.

2 Region segmentation of STL model

2.1 Definition of feature surface

The STL model generated by the CAD system is composed of many discrete and segmented triangular facets [28-30]. To accurately obtain the surface information of the STL model and ensure the accuracy of subsequent path planning, the topology of the STL model should be reconstructed according to the rules given by Yang et al. [31]. These rules are:

(1) Given a triangular facet F, its three edges E_1, E_2, E_3 and three vertices can be obtained.

(2) Given any edge E of a triangular facet, two of its vertices and two adjacent triangular facets F_1 and F_2 can be obtained.

(3) Given a vertex, all the triangular facets in which the vertex is located can be obtained.

All the vertices, edges and facets of the STL model are stored in lists *ListV*, *ListE* and *ListF*, respectively. The indexes of vertices, edges and facets in their respective list are their corresponding *VertexIndex*, *EdgeIndex* and *FaceIndex*. For example, the *FaceIndex* of the first element in the *ListF* is 0 (the subscript starts from 0).

However, the STL model obtained thus only connects all triangular facets but does not establish the relationship between triangular facets and feature surfaces. To perform path planning for a specific surface, it is necessary to first define the feature surface and then add the triangular facets to the different feature surfaces.

The feature surface is a set of connected triangular facets with similar characteristics according to some rules. In this paper, the dihedral angle is used as the basis for classifying the feature surface. The dihedral angle is the complement of the angle between the two triangular facets adjacent to the common edge, as shown in Fig. 1, which reflects the smoothness between the two triangular facets of the STL model: the smaller the dihedral angle, the smoother the transition between the two triangular facets; on the contrary, the more abrupt the transition between the two triangular facets. The dihedral angle θ of E_1 can be calculated as shown in Eq. (1):

$$\theta = \cos^{-1}\left(\frac{\boldsymbol{n}_1\boldsymbol{n}_2}{|\boldsymbol{n}_1||\boldsymbol{n}_2|}\right) \tag{1}$$

where $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ are normal vectors corresponding to the two triangular facets F_1 and F_2 adjacent to E_1.

The dihedral angle formed between two adjacent triangular facets within the feature surface is less than the threshold α, while the dihedral angle formed between two triangular facets adjacent to the common edge of the adjacent feature surface is greater than or equal to the threshold α. Yuan et al. [32] conducted a statistical analysis of the dihedral angle of the STL model and found that the corresponding edges of the triangular facet could reflect the characteristics of the model when the dihedral angle was greater than 20°. Therefore, α=20° is used as the threshold value for feature surface classification in this paper.

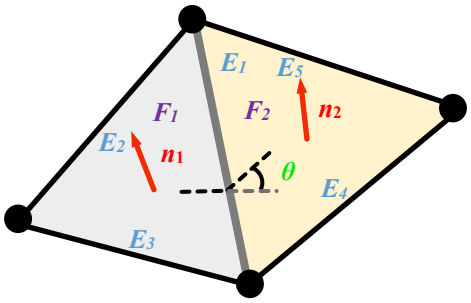

Fig. 1. Schematic diagram of dihedral angle θ.

2.2 Region segmentation algorithm

Since the topological relationship of the STL model is established in advance, all information about the model can be obtained by gradually expanding from a vertex or an edge of any triangular facet. Therefore, the idea of the STL model region segmentation algorithm designed in this paper is to start from an edge of a triangular facet, calculate its dihedral angle, and continuously add adjacent triangular facet with similar characteristics to the feature surface list *ListFF* according to the feature surface partition conditions. Based on the above idea, the edge attributes *E. IsCaculate* and *E. IsFeatureEdge* are declared, and the facet attribute *F. IsCaculate* is declared, all of which have the initial value of *false*. The relationships between these attributes are shown in Eq. (2-4):

$$E.\theta \geq \alpha \rightarrow E.IsFeatureEdge = true, E.IsCaculate = true \quad (2)$$

$$E.\theta < \alpha \rightarrow E.IsCaculate = true \quad (3)$$

$$\begin{cases} E_1.IsCaculate = true \\ E_2.IsCaculate = true \\ E_3.IsCaculate = true \end{cases} \rightarrow F.IsCaculate = true \quad (4)$$

The triangular facets belonging to the current feature surface are stored in *ListFF* according to their calculating order. When all the triangular facets of the feature surface are added, the face separator *SignF* is inserted, and then the triangular facets belonging to the next feature surface are added subsequently. It is important to note that the variables store vertices, edges, and facets are all reference types, which store references to their data (objects), and two variables can refer to the same object [33]. Therefore, an operation

performed on one variable is equally mapped to another variable. To better illustrate the region segmentation algorithm, the pseudocode of this algorithm is given as Algorithm 1.

Algorithm 1

```
initial a ← 0, b ← 0, num ← 0, F ← ListF [0], ListFF ← { }, ListTempF ← { }
function SEGMENTATION (a, b, num ← 0, F, ListT, ListFF, ListFF, ListTempF) // Main fuction
foreach Edge E in F
        if E. IsCaculate = false
            a = F1. FaceIndex, b = F2. FaceIndex
            break
        end if
end foreach
E. θ = cos⁻¹ (n1.n2 / |n1||n2|)
if E. θ > α
    E. IsCaculate = true,E. IsFeatureE = true
    if ListFF ⊅ F
        ListFF ← F ⋃ ListFF, ListTempF ← F ⋃ ListTempF
    end if
    JUDGE(m, a, b, num, ListF, ListFF, ListTempF)
else // E. θ < α
    E.IsCaculate = true
    if ListFF ⊅ ListF [a]
        ListFF ← ListF [a] ⋃ ListFF, ListTempF ← ListF [a] ⋃ ListTempF, m ← a
    end if
    if ListFF ⊅ ListF [b]
        ListFF ← ListF [b] ⋃ ListFF, ListTempF ← ListF [b] ⋃ ListTempF
    end if
    JUDGE (m, a, b, num, ListF, ListFF, ListTempF)
end function
```

Regardless of whether the dihedral angle θ is greater or less than the threshold α, it is needed to determine whether the *IsCaculate* of all the three edges of the current triangular facet in *ListF* is *true*. If all of them are *true*, the *IsCaculate* of the facet is marked as *true* and the current facet is removed from *ListF*, and then the subsequent operations are performed; otherwise, the uncomputed edge of the current triangular facet is obtained to calculate the dihedral angle, and then the subsequent operations are performed. Therefore, the same processing is encapsulated into a calling function to avoid code redundancy and enhance code readability, the relevant related pseudocode is given as Algorithm 2.

Algorithm 2

```
function JUDGE (m, a, b, num, ListF, ListFF, ListTempF)
if IsCaculate of the three edges of ListF [m] is true
    ListF [m]. IsCaculate = true, ListF ← ListF - ListF [m], num ← num + 1.
    foreach F in ListTempF
            if IsCaculate of the three edges of F is true ∧ F.IsCaculate = false
                F.IsCaculate = true, ListTempF ← ListTempF - F
                num ← num + 1
            end if
    end foreach
    if ListF. Count = 0 return ListFF
    if ListFF. Count = num
        ListFF ← ListFF ⋃ SignF, num ← num + 1.
        foreach Facet F in ListF
                if F. IsCaculate = false
```

```
15                SEGMENTATION (a, b, F, ListF, ListFF, ListTempF)
16              end if
17    end foreach
18  else // if ListFF. Count ≠ num
19      foreach Facet F in ListFTempT
20              if  F. IsCaculate = false
21                 SEGMENTATION (a, b, F, ListF, ListFF, ListTempF)
22              end if
23      end foreach
24 else //if IsCaculate is not true for all three edges
25     SEGMENTATION (a, b, F, ListF, ListFF, ListTempF)
26 end function
```

Based on the self-developed region segmentation algorithm, the STL models of several typical parts are divided into some regions, different feature surfaces on the parts are labeled with different colors, and the unsegmented original part models with dark gray color are given as a comparison. The segmentation results are shown in Fig. 2. For example, the end face, the shaft hole surface, and the pin hole surface of the worm gear are divided into different feature surfaces, as shown in Fig. 2(a); The journal surface, shaft tube surface and shaft hole surface connected with the universal joint of the automobile transmission front shaft are also divided into different feature surfaces, as shown in Fig. 2(b). This shows that the STL region segmentation algorithm designed in this paper can correctly divide the part surface into different feature surfaces, and each feature surface corresponds to the surface with different functions on the part. The fundamental reason is that two surfaces with different functions on a mechanical part usually have clear boundaries in shape as well.

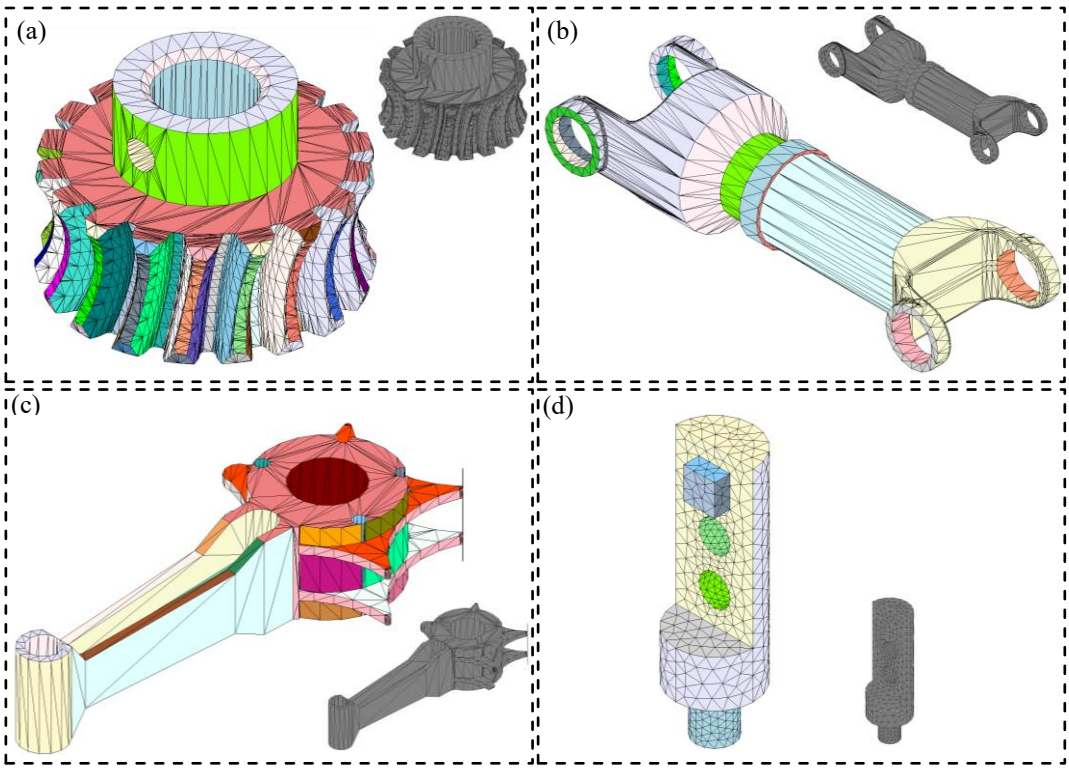

Fig. 2. Example of STL model region segmentation: (a) worm gear, (b) automobile transmission front shaft, (c) engine connecting rod, (d) test workpiece to be cleaned by laser.

2.3 Free control of feature surface

To achieve free control of the feature surfaces to be cleaned by laser, the ListView control of the C# WinForm form is used to manage the feature surfaces. The currently uncleaned surfaces correspond to the items in the list of feature surfaces will be loaded into the list of surfaces to be cleaned when selected, and the corresponding items and feature surfaces are removed automatically from the list of feature surfaces and the STL model. The test workpiece shown in Fig. 2**(**d**)** is operated with free control of the feature surface, the operation steps are shown in Fig. 3(a), and the results corresponding to each step are shown in Fig. 3b.

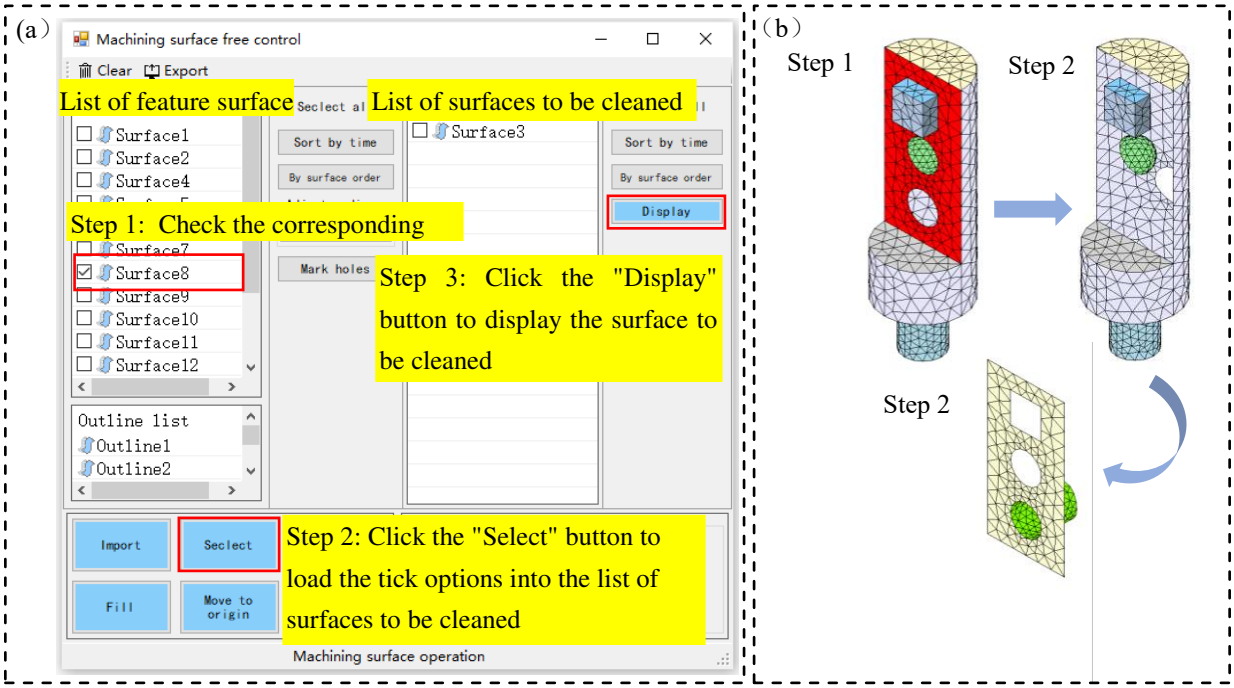

Fig. 3. Example of feature surface manipulation: (a) Feature surface control form, (b) Result of operation.

3 Interference detection and correction of robot laser cleaning path

In this paper, the slicing algorithm based on the topological relation proposed by Li et al. [34] is used to obtain the original points of the robot laser cleaning path, that is, the edges intersect with the current section plane are continuously found among the edges of the triangular facets on the feature surfaces according to the edge topological relation after the section plane is determined, and the intersection points are calculated. After all the intersections are obtained, the transversal lines are obtained by connecting them in series. The transversal line is offset from the original surface by a distance to obtain the robot laser cleaning path. This offset direction is usually related to the robot pose, and the offset distance is defined as the defocus distance. The original robot pose determined by the normal vector of the intersection point may cause interference between the robot laser cleaning path and workpiece, therefore, it is necessary to correct the robot pose to

make the cleaning process safe and effective.

3.1 Interference detection

When the robot is cleaning the surfaces with small curvature variation, the laser beam centerline l can directly coincide with the normal vector $\boldsymbol{\gamma}$ of the cleaning point (pose 1), as shown in Fig. 4. However, when cleaning some areas, such as the "canyon" surface shown in Fig. 4, the coincidence of the laser beam centerline l' with the normal vector $\boldsymbol{\gamma}$' (pose 2) will lead to interference between the laser cleaning path and the workpiece, resulting in damage of the laser head and even serious safety accidents.

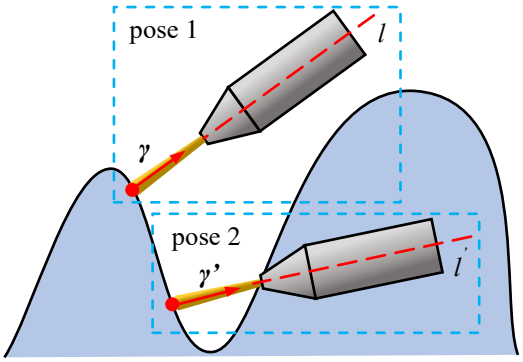

Fig. 4. Two positions of the laser beam on the workpiece.

ListSect $\{v_0, v_1, \ldots, v_i \ldots, v_n\}$ is a transversal line located in the range of possible interference range; *ListInterf* is the list of interference points, and $\boldsymbol{a_i}$ is the normal vector of the point v_i. Therefore, whether the robot interferes at the processing point v_i is essentially that whether the ray r with the v_i as the endpoint and $\boldsymbol{a_i}$ as the direction vector intersects with the triangular facet $F(p_0, p_1, p_2)$. The expression of ray r is shown in Eq. (4):

$$r(t) = v_i + t\boldsymbol{a_i},\ 0 \le t < +\infty \tag{4}$$

where t is the distance that v_i travels along the unit normal vector direction.

According to the Möller-Trumbore algorithm [35], when the ray intersects with a triangular facet, the following equation can be obtained as Eq. (5):

$$v_i + t\boldsymbol{a_i} = (1 - b_1 - b_2)\, p_0 + b_1 p_1 + b_2 p_2 \tag{5}$$

where b_1 and b_2 are the coefficients that make this equation valid when the ray intersects with the triangular facet, and the constraint conditions of b_1 and b_2 are shown in Eq. (6):

$$\begin{cases} 0 \le b_1 \le 1 \\ 0 \le b_2 \le 1 \\ 0 \le b_1 + b_2 \le 1 \end{cases} \tag{6}$$

According to Cramer's law, the following equations can be gotten as Eq. (7):

$$\begin{cases} t = \dfrac{S_2 \cdot P_2}{P_1 \cdot S_1} \\ b_1 = \dfrac{S_1 \cdot S}{P_1 \cdot S_1} \\ b_2 = \dfrac{S_2 \cdot \boldsymbol{a_i}}{P_1 \cdot S_1} \end{cases} \tag{7}$$

where $P_1 = p_1 - p_0$, $P_2 = p_2 - p_0$, $S = v_i - p_0$, $S_1 = \boldsymbol{a_i} \times P_2$, $S_2 = S \times P_1$. By solving Eq. (7) and determining whether the coefficients are within the range of values, it can be determined whether the laser cleaning path will interfere with the workpiece.

3.2 Robot pose correction

It is challenging to correct the robot pose of the interference points. Firstly, due to the complexity of the part surface, it is difficult to directly establish the rules of the incident direction of the laser; secondly, if the robot pose changes greatly between adjacent interference points after correction, not only will the robot joints rotate rapidly, but also the laser scanning path and the planned path will not coincide when the robot moves between two points. Therefore, the idea of the robot pose correction algorithm in this paper is to perform interference detection on each transversal line by point to generate a list of interference points *ListInterf*, and obtain two non-interference points adjacent to the first and last interference points, then the robot pose of the interference point is corrected according to the robot pose of the two non-interference points.

As shown in Fig. 5, the points in *ListInterf* are all interference points except p_0 and p_n, so it is necessary to correct the robot pose of p_1, p_{n-1} and the points between them. First, calculate the intersection point O of the rays of p_0 and p_n along the normal vector direction, so when the two rays intersect, Eq. (8) is satisfied:

$$p_0 + k_0 \boldsymbol{a}_0 = p_n + k_n \boldsymbol{a_n} \tag{8}$$

where k_0 and k_n are respectively the moving distances of p_0 and p_n along the normal direction when the two rays intersect. The result of k_0 is shown in Eq. (9):

$$k_0 = \frac{\left[\left(p_n - p_0\right) \times \boldsymbol{a_n}\right] \cdot \left(\boldsymbol{a_0} \times \boldsymbol{a_n}\right)}{\left\|\boldsymbol{a_0} \times \boldsymbol{a_n}\right\|^2} \tag{9}$$

The intersection point O can be obtained by substituting k_0 into Eq. (8). The vector of O point pointing to the point p_i ($1 \le i \le n - 1$) in *ListInterf* is taken as the direction vector of the incident laser light at p_i. Therefore, any point between p_1 and p_n must not cause interference.

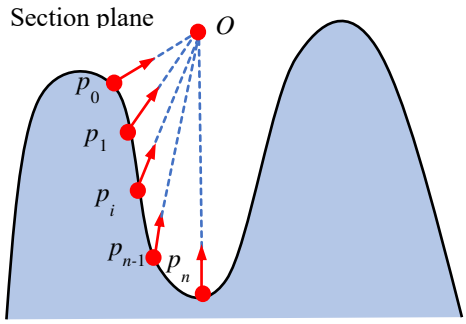

Fig. 5. Robot pose correction.

The two rays of p_0 and p_n along the normal vector direction may not intersect. Therefore, before judging whether the intersection point will cause interference between the cleaning path and the workpiece, the normal vector $\boldsymbol{a_i}$ of the intersection points within *ListSect* is projected onto the section plane, as shown in Fig. 6. Let the normal vector of the section plane be $\boldsymbol{n}$. Then the projection of the normal vector $\boldsymbol{a_i}$ is shown in Eq. (10):

$$\boldsymbol{a_i'} = \frac{\boldsymbol{a_i} \cdot \boldsymbol{n}}{|\boldsymbol{n}|^2} \tag{10}$$

In this way, the robot pose correction problem in three dimensions is transformed into the two-dimensional plane, which greatly reduces the complexity of the problem.

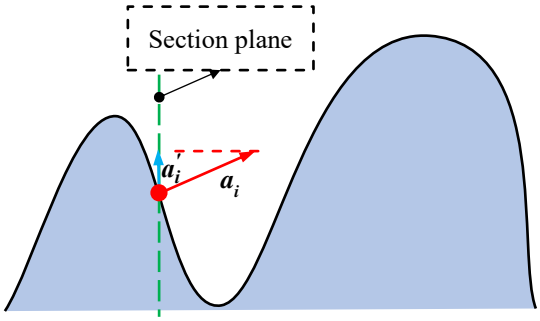

Fig. 6. Projection of the intersection normal vector into the intercepted plane

To further illustrate the robot pose correction algorithm, the pseudocode of this algorithm is given in Algorithm 3. The Boolean variable *IsLastInterf* is used to record whether the previous intersection point is an interference point. The Boolean variable *IsCurInterf* indicates whether the current intersection point is an interference point. The initial values are all *false*, indicating that the intersection point is a non-interference point.

Algorithm 3

```
1 function ADJUST (ListSect {v0, v1, … vi … , vn} )
2 initial ListInterf ← { }, IsLastInsec = false，IsCurInsec = false
3 for j ← 0 to n do
4    Project the normal vector of vj onto the section plane
5    Interference detection for vj
6    if vj is the interference point then
7       IsCurInsec = true.
8       if IsLastInsec = false then ListInterf ← ListInterf ⋃ v i - 1, k ← 1 end if
9       ListInterf ← ListInterf ⋃ vi, k ← 1
```

```
    else if IsCurInsec = false ∧ IsLastInsec = true then
        ListInterf ← ListInterf ∪ v_i, k ← 0
        Robot pose correction for points in ListInterf
        Clear the ListInterf
end for
end function
```

The STL models shown in Fig. 2(a) and (d) are corrected for robot pose respectively. The model shown in Fig. 7 is a worm gear, and after intersecting the intercepted plane with the worm gear teeth, the normal vectors of the intersection points of the gear teeth root points to the opposite gear teeth root, shown in Fig. 7(a). If the laser incident direction and the normal vector remain coincident at this point, the laser beam will be blocked by the gear teeth or the laser head will interfere with the gear teeth. Fig. 7(b) shows the interference between the laser scanning path and the worm gear teeth at a defocus distance of 3 mm. After the robot pose is corrected, the laser cleaning path no longer interferes with the worm gear teeth, as shown in Fig. 7(c). Fig. 8 directly shows the change of the laser cleaning path before and after the robot pose correction on the worm gear surface from the perspective of the cross-sectional plane.

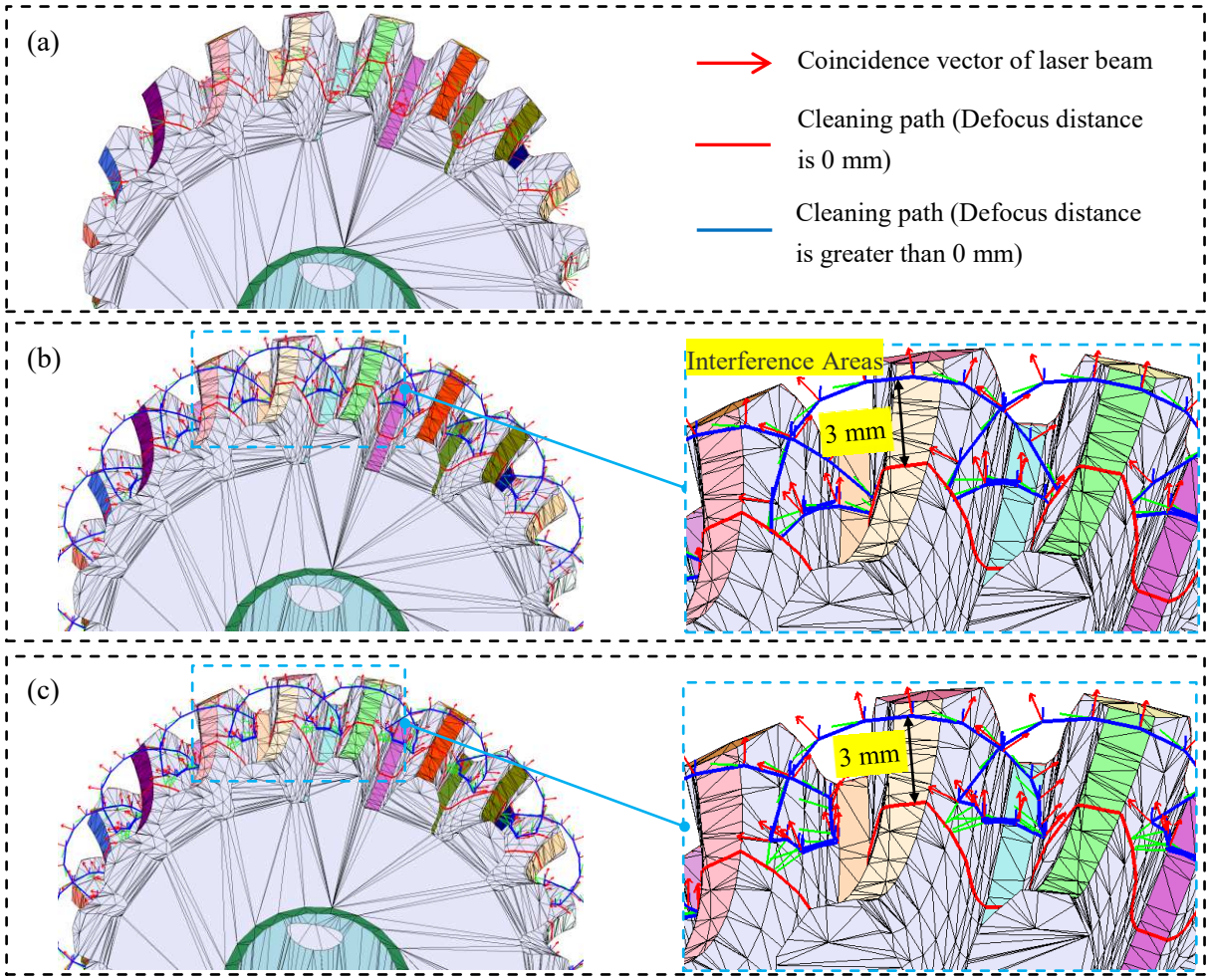

Fig. 7. Laser cleaning path on the worm gear surface, robot pose: (a) before correction(Defocus distance is 0 mm), (b) before correction (Defocus distance is 3 mm), (c) after correction.

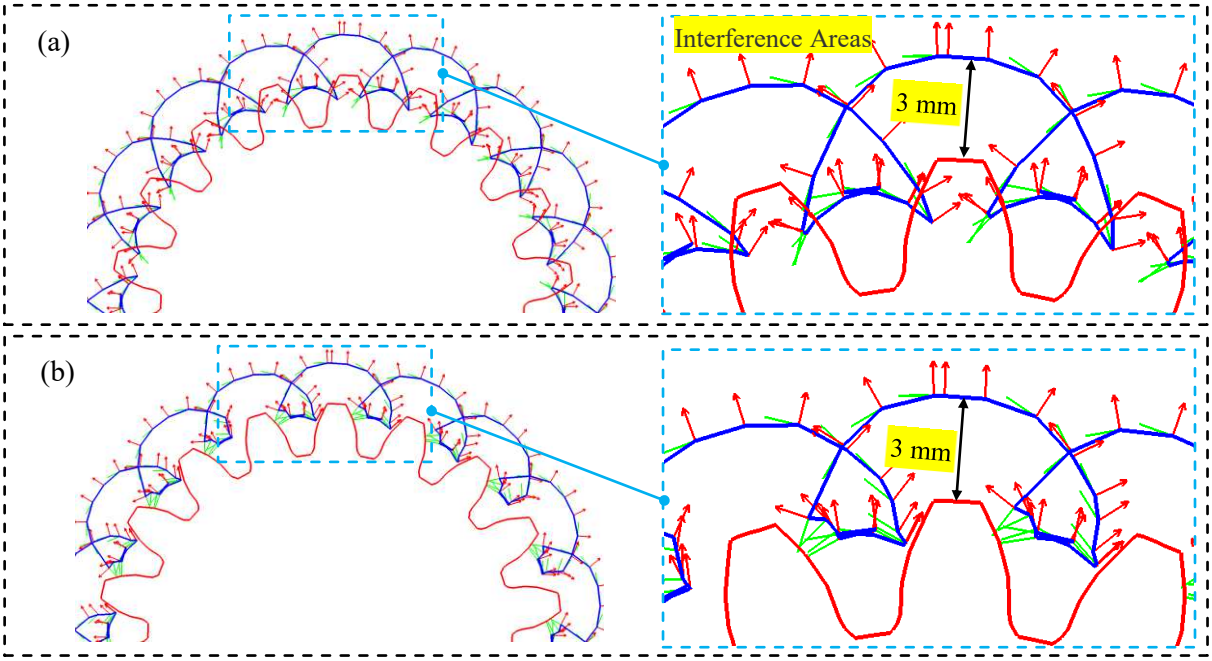

Fig. 8. Laser cleaning path of worm gear surface from cross-plane view, robot pose: (a) before correction, (b) after correction.

Fig. 9 shows the comparison of the laser cleaning path for the test workpiece before and after the correction of the robot pose. The laser scanning path in the crater area interferes with the square convex plate when the defocus distance is 20 mm before the pose correction. So, the smaller defocus distance will lead to the collision between the laser head and the workpiece; while the larger defocus distance will cause the laser to be blocked by the test workpiece. After pose correction, the cleaning path no longer interferes with the workpiece.

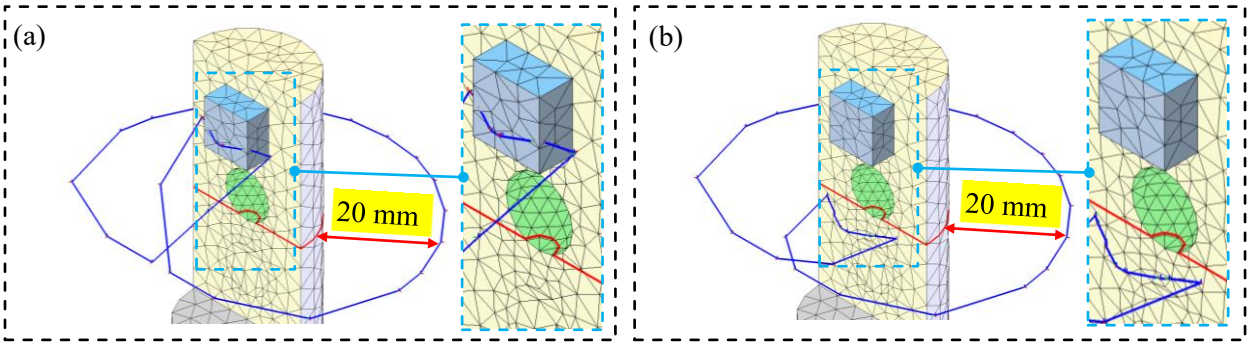

Fig. 9. Surface laser cleaning path of the test workpiece, robot pose: (a) before correction, (b) after correction.

4 Experimental validation

To verify the validity of the proposed robot laser cleaning path planning and pose correction algorithm for the specific surface of the part, the robot laser cleaning experiments were conducted on the test workpiece. The robot laser cleaning system adopted is shown in Fig. 10, which is mainly composed of four parts: laser system, control system, robot motion system and auxiliary system. The laser system (Jiangsu Zhongke Sixiang

Laser Technology Co., Ltd. Jiangsu, China) includes an RFL-C500 fiber laser and ZF-CH002A laser cleaning head; the control system is a computer equipped with robot laser cleaning path planning and off-line programming software developed by ourselves; the robot motion system (NACHI-FUJIKOSHI(CHINA)CO., LTD.) contains NACHI MZ04 robot and control cabinet; the auxiliary system includes argon gas protection device and water-cooling device. Among them, the robot processing mode adopts the mode that the laser head is fixed and the robot clamps the workpiece to move relative to the laser head. This mode maximizes the processing range and improves the processing efficiency while avoiding the installation of a seventh axis; the direction of laser emission is always vertical and downward to ensure the safety of the cleaning process.

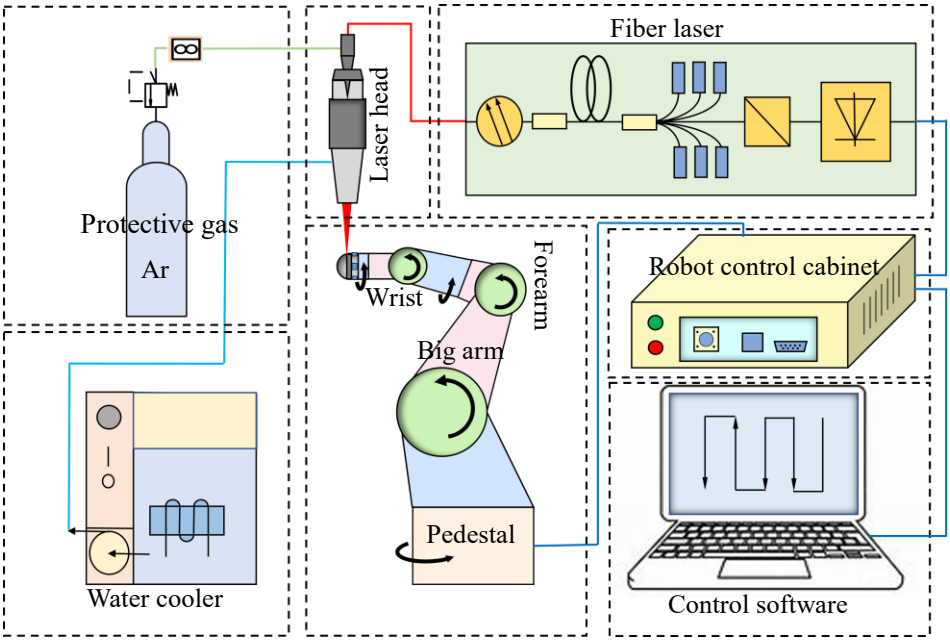

Fig. 10. Robotic laser cleaning system.

The cleaning parameters shown in Table 1 were used in the robot laser cleaning validation experiments. The material of the test workpiece was 45 steel, and the surface of the test workpiece had a bright metallic luster before the application of paint, as shown in Fig. 11(a) and (b). Before the laser cleaning experiment, a layer of black alkyd paint was applied to the surface of the test workpiece and left to dry at room temperature for 12 hours. Fig. 11(c) and (d) show the front and back of the painted test workpiece, respectively.

Table 1. Laser cleaning process parameters

Laser power	Defocus distance	Scanning speed	Scanning spacing	Spot diameter
500 W	20 mm	3 mm/s	0.5 mm	1 mm

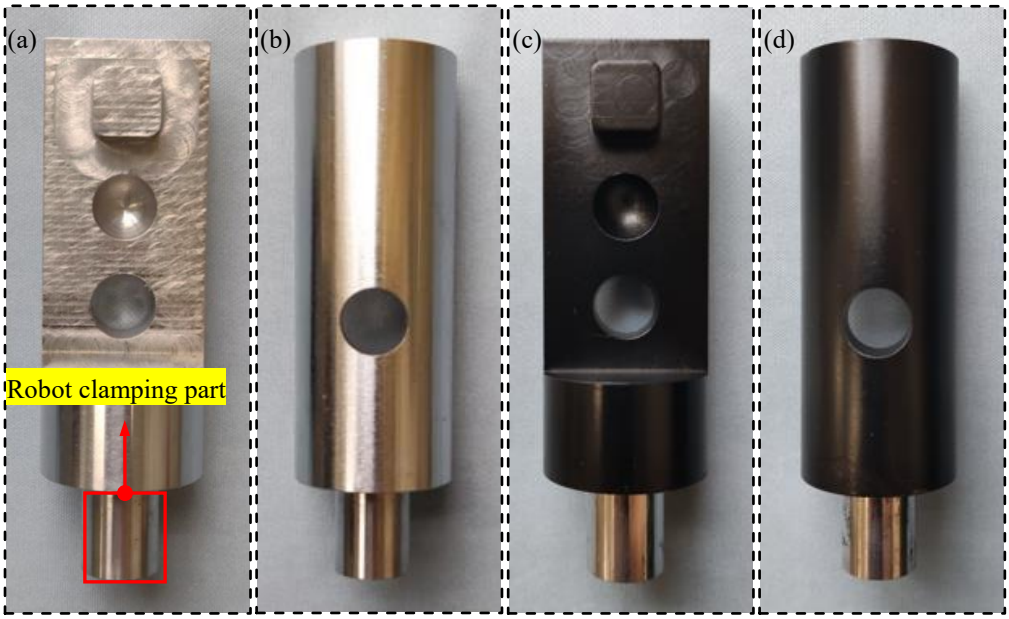

Fig. 11. Before applying paint to the test workpiece: (a) front view; (b) rear view; (c) painted test workpiece (front view); (d) painted test workpiece (back view).

4.1 Path planning and cleaning of global surface on a test workpiece

Firstly, the whole surface of the painted test workpiece was cleaned to verify the rationality of the global path planning. Fig. 12(a) shows the global path planning result on the surface of the test workpiece, and the layered direction is the *Z* direction. It can be seen that the generated paths cover the surface of the test workpiece evenly. As can be seen in Fig. 12(b) and (c), the paint on the surface of the test workpiece was removed by laser cleaning process according to the planned path, exposing the original metallic luster.

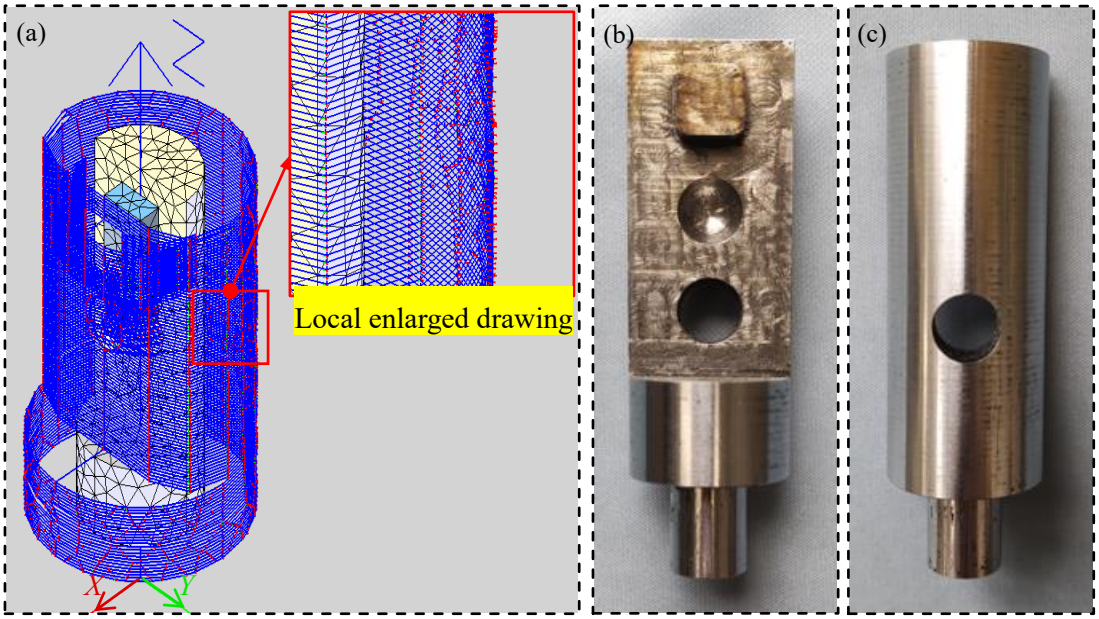

Fig. 12. Laser cleaning of global surface on the test workpiece: (a) global path planning result of the test workpiece; (b) test workpiece after cleaning (front view); (c) test workpiece after cleaning (back view).

4.2 Path planning and cleaning of specific surface on a test workpiece

Firstly, the path planning and cleaning verification of a specific single surface are carried out by cleaning the top surface on the lug boss marked by the red box in Fig. 13(a). After selecting the feature surface corresponding to the lug boss in the feature surface control form, a series of cross-sectional planes are used to find the intersection points with the edges of the triangular facets of the surface to be cleaned, and the intersection points are the original points of cleaning path points on the specific surface. According to the set of the corresponding layered parameters, the path planning results are shown in Fig. 13(b). The right side of Fig. 13b shows the planned cleaning path of the surface to be cleaned, and the cleaning path is uniformly covered above the lug boss; the left side is the local amplification of the cleaning path, and a target coordinate system is fixed on each path point, Where $\boldsymbol{x}$ (red) is the normal vector at the surface of the part corresponding to the path point, $\boldsymbol{y}$ (green) is the tangent vector of the path point located in the section plane, and $\boldsymbol{y} = \boldsymbol{x} \times \boldsymbol{n}$, $\boldsymbol{z}$ (blue) is obtained by $\boldsymbol{x}$ and $\boldsymbol{y}$ according to the right-handed rule. The robot pose is determined by $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$. A blue line is used to connect the path points in the series. The top surface of the lug boss was cleaned by the robot according to the planned cleaning path. As shown in Fig. 13(c), the paint on the lug boss surface has been accurately removed, while the paint on the non-cleaned surface is still intact.

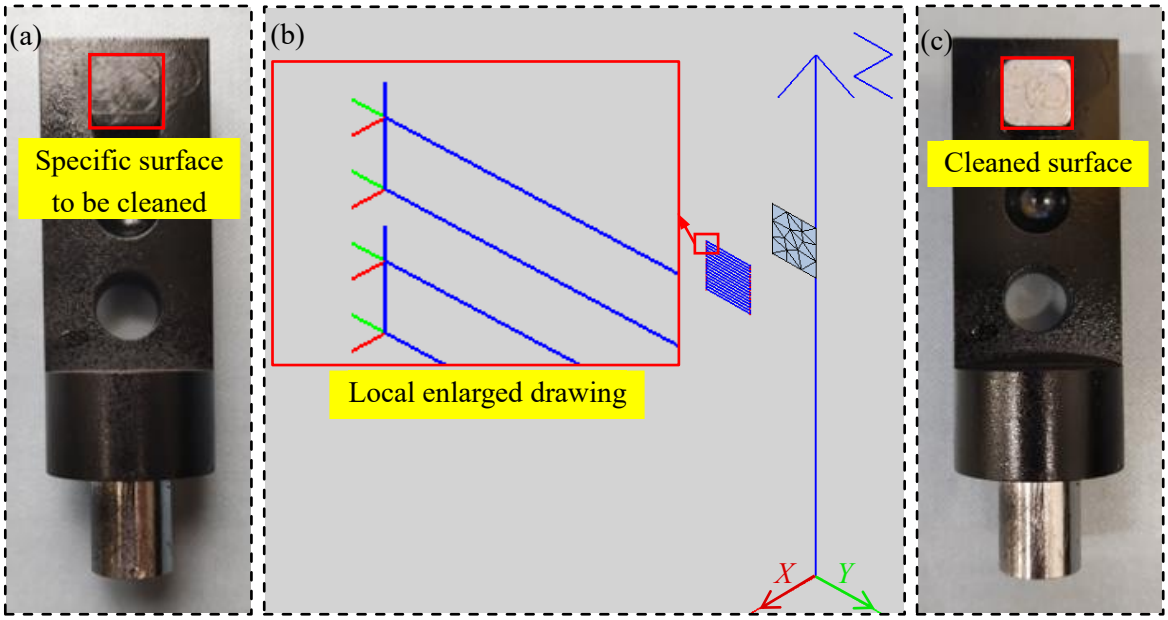

Fig. 13. Local surface laser cleaning of the test workpiece: (a) painted test workpiece (front view); (b) local enlargement of the machining path; (c) the result of path planning for the specific surface; (d) the specific surface after cleaning (front view).

To further verify the effectiveness of cleaning path planning on the specific surfaces, laser cleaning was performed on the surface marked by the red box in Fig. 14(a), including crater, lug boss, and planes with holes.

The planning paths of the specific surfaces to be cleaned are shown in Fig. 14(b). The robot completed the laser cleaning of the specific surfaces according to the planned path, as shown in Fig. 14(c,) and the robot did not touch the rest of the surfaces when cleaning the specific surfaces , as sown in Fig. 14(d), which proves that path planning for multiple specific surfaces is still effective.

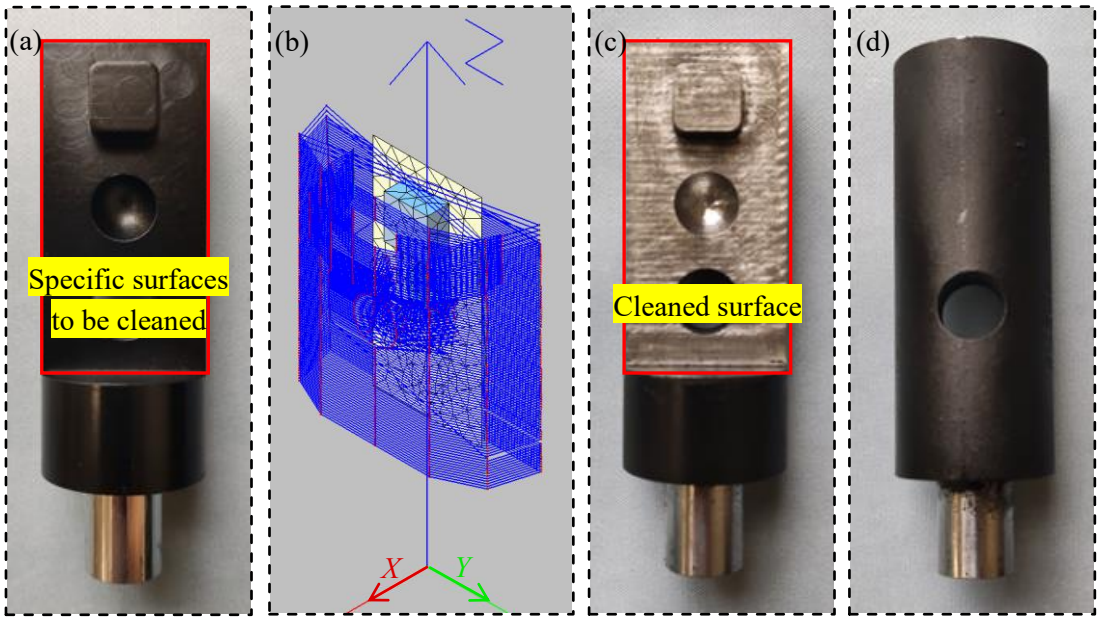

Fig. 14. Laser cleaning of specific surfaces on the test workpiece: (a) painted test workpiece (front view); (b) painted test workpiece (back view); (c) path planning result of the specific surfaces; (d) specific surfaces after cleaning (front view); (e) specific surfaces after cleaning (back view).

Due to the interference points detection and robot pose correction, theoretically, the generated robot cleaning path does not lead to interference with the test workpiece. As shown in Fig. 15(a), the cleaning path of the crater area surface is smooth and no abrupt change occurred. In the laser cleaning process of the robot, the robot can safely pass through the crater where interference is easy to occur, and the laser is also successfully irradiated to the surface of the test workpiece. The robot pose is stable, and there is no rapid swing or rotation. Fig. 15(b), (c) and (d) show the cleaning effectiveness of the crater area surface from various views respectively. It can be seen that the crater has been cleaned. Although there are some small paint spots, the problem can be solved through subsequent adjustment of process parameters. Therefore, the robot pose correction method is still effective.

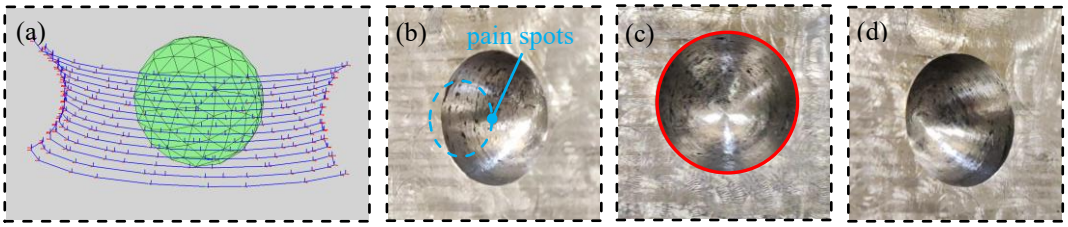

Fig. 15. Laser cleaning of crater area: (a) crater area path planning; (b) right 45° view; (c) front view; (d) left 45° view.

5 Conclusions

This paper mainly studied the path planning strategy of robot laser cleaning for specific surfaces and the easy interference area of the part. To solve the shortcomings of the application of robots in the field of laser cleaning, the region segmentation technology of the STL model was introduced into the field of robot path planning, and a robot pose correction method was proposed for the interference phenomenon in the robot cleaning process, so the free cleaning control of the part surface was realized by combining the path planning algorithm self-developed. The main conclusions are as follows：

(1) The region segmentation algorithm of the STL model based on topological relationship is designed, and it is organically combined with the WinForm form application developed based on C # and .net framework so that users can freely set the surfaces to be cleaned.

(2) The interference points are searched by the ray method, and the robot pose of two non-interference points adjacent to the first and the last interference points in the interference point list is used as the basis for correcting the robot pose of the interference points, which means the interference detection and pose correction of the robot laser cleaning path are realized.

(3) Laser cleaning experiments are carried out on test workpieces with multi-feature surfaces based on the robot laser cleaning system. The robot drives the test workpiece to complete the laser cleaning of the global surface and the specific surfaces safely and accurately according to the planned path with no rapid swing or sudden change in a pose during the cleaning process. It is proved that the method proposed in this paper can effectively realize the robot path planning of the specific surfaces and the robot pose correction of the interference points.

Data Availability Statement:

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgments:

This work was supported by the following funds: (1) the National Natural Science Foundation of China (Grant 51775559); (2) the Natural Science Foundation of Jiangsu Province (BK20221382); and (3) the Fundamental Research Funds for the Central Universities of Central South University (Grant 160171022).

Conflicts of Interest/Competing interests:

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

Code availability:

Not applicable.

Author contributions:

Zhigui Pan: Writing – Original draft preparation, Methodology, Completion of experiment, Visualization, Investigation. Defu Liu: Conceptualization, Methodology, Supervision, Writing – Reviewing and editing, Project 35 administration. Shudan Li: Completion of experiment, Writing – Reviewing and editing. Zixin Deng: Manuscript modification and language proofing. Jian Liu: Completion of experiment. Tao Chen: Completion of experiment.

Ethics approval:

Not applicable.

References

1 Guangxing Zhang, Xueming Hua, Ye Huang, et al. Investigation on mechanism of oxide removal and plasma behavior during laser cleaning on aluminum alloy. Applied Surface Science. 506 (2020) 144666. DOI: 10.1016/j.apsusc.2019.144666.

2 Zhichao Li, Donghe Zhang, Xuan Su, et al. Removal mechanism of surface cleaning on TA15 titanium alloy using nanosecond pulsed laser. Optics & Laser Technology. 139 (2021) 106998. DOI: 10.1016/j.optlastec.2021.106998.

3 Guodong Zhu, Zhenhai Xu, Yang Jin, et al. Mechanism and application of laser cleaning: A review. Optics and Lasers in Engineering. 157 (2022) 107130. DOI: 10.1016/j.optlaseng.2022.107130.

4 Kunyan Ding, Kainan Zhou, Guoying Feng, et al. Mechanism and conditions for laser cleaning of micro and nanoparticles on the surface of transparent substrate. Vacuum. 200 (2022) 110987. DOI: 10.1016/j.vacuum.2022.110987.

5 Haichao Zhao, Yulin Qiao, Xian Du, et al. Laser cleaning performance and mechanism in stripping of Polyacrylate resin paint. Applied Physics A. 126 (5) (2020). DOI: 10.1007/s00339-020-03551-0.

6 Angel Rodríguez, Ana J. López, Javier Lamas, et al. Development of a laser cleaning robot system for the processing of 3D surfaces.: SPIE, 2021:117850V.

7 Qingzeng Ma, Yuan Ren, Shuo Jin, Leiguang Nan, Hailong Ge, Wei Cheng Design of Laser Cleaning System for Large-scale Plane Workpiece. Journal of physics. Conference series. 1748 (6) (2021) 62034. DOI: 10.1088/1742-6596/1748/6/062034.

8 Guodong Zhu, Zhenhai Xu, Yang Jin, et al. Mechanism and application of laser cleaning: A review. Optics and Lasers in Engineering. 157 (2022) 107130. DOI: 10.1016/j.optlaseng.2022.107130.

9 Habiba Zahir Imam, Yufan Zheng, Rafiq Ahmad An efficient tool-path planning approach for repair of cylindrical components via laser cladding. Journal of Remanufacturing. 11 (2) (2021) 137-46. DOI: 10.1007/s13243-020-00096-6.

10 Xuemei Liu, Chengrong Qiu, Qingfei Zeng, Aiping Li Kinematics Analysis and Trajectory Planning of collaborative welding robot with multiple manipulators. Procedia Cirp. 81 (2019) 1034-9. DOI: 10.1016/j.procir.2019.03.247.

11 Martin Pollák, Jozef Dobránsky Structural Design and Material Cutting Using a Laser End Effector on a Robot Arm. TEM Journal. DOI: 10.18421/TEM94.

12 Heping Chen, Thomas Fuhlbrigge, Xiongzi Li A review of CAD‐based robot path planning for spray painting. Industrial Robot: An International Journal. 36 (1) (2009) 45-50. DOI: 10.1108/01439910910924666.

13 Li He-xi, Shi Yong-hua, Wang Guo-rong, Zheng Xiao-xi Automatic Teaching of Welding Robot for 3-Dimensional Seam Based on Ant Colony Optimization Algorithm.: IEEE, 2009:398-402.

14 Ke Zhang, Danyang Li, Hao Gui, Zhuguo Li An adaptive slicing algorithm for laser cladding remanufacturing of complex components. The International Journal of Advanced Manufacturing Technology. 101 (9-12) (2019) 2873-87. DOI: 10.1007/s00170-018-3107-2.

15 Huadong Zheng, Ming Cong, Dong Liu, Hang Dong, Yi Liu Optimization of STL model and layer shape for laser cladding forming. The International Journal of Advanced Manufacturing Technology. 100 (1-4) (2019) 599-608. DOI: 10.1007/s00170-018-2714-2.

16 Zeqi Hu, Lin Hua, Xunpeng Qin, Mao Ni, Zhimin Liu, Congming Liang Region-based path planning method with all horizontal welding position for robotic curved layer wire and arc additive manufacturing. Robotics and Computer-Integrated Manufacturing. 74 (2022) 102286. DOI: 10.1016/j.rcim.2021.102286.

17 Lei Yang, Yanhong Liu, Jinzhu Peng, Zize Liang A novel system for off-line 3D seam extraction and path planning based on point cloud segmentation for arc welding robot. Robotics and Computer-Integrated Manufacturing. 64 (2020) 101929. DOI: 10.1016/j.rcim.2019.101929.

18 P. Olivieri, L. Birglen, X. Maldague, I. Mantegh Coverage path planning for eddy current inspection on complex

aeronautical parts. Robotics and Computer-Integrated Manufacturing. 30 (3) (2014) 305-14. DOI: 10.1016/j.rcim.2013.10.002.

19 Huadong Zheng, Ming Cong, Hang Dong, Yi Liu, Dong Liu CAD-based automatic path generation and optimization for laser cladding robot in additive manufacturing. The International Journal of Advanced Manufacturing Technology. 92 (9-12) (2017) 3605-14. DOI: 10.1007/s00170-017-0384-0.

20 Hao Wu, Yudi Wang, Xiaoxu Wei, Dahu Zhu Spatial Path Planning for Robotic Milling of Automotive Casting Components Based on Optimal Machining Posture. Metals. 12 (8) (2022) 1271. DOI: 10.3390/met12081271.

21 Qingzeng Ma, Dongbin Zhang, Shuo Jin, Yuan Ren, Wei Cheng, Yanlei Li On Path Generation Method for Laser Cleaning Robot Based on Line Structured Light.: Technical Committee on Control Theory, Chinese Association of Automation, 2020:5953-7.

22 Jin Shuo, Ren Yuan, Ma Qingzeng, Ge Hailong, Li Wenlong, Cheng Wei Off-line programming of robot on laser cleaning for large complex components. Journal of physics. Conference series. 1748 (2) (2021) 22027. DOI: 10.1088/1742-6596/1748/2/022027.

23 Xinlong Wang, Wenlei Sun, Ying Chen, Jianjie Zhang, Yong Huang, Haibo Huang Research on trajectory planning of complex curved surface parts by laser cladding remanufacturing. The International Journal of Advanced Manufacturing Technology. 96 (5-8) (2018) 2397-406. DOI: 10.1007/s00170-018-1737-z.

24 Gangxian Zhu, Shihong Shi, Geyan Fu, et al. The influence of the substrate-inclined angle on the section size of laser cladding layers based on robot with the inside-beam powder feeding. The International Journal of Advanced Manufacturing Technology. 88 (5-8) (2017) 2163-8. DOI: 10.1007/s00170-016-8950-4.

25 Jinduo Liu, Wenlei Sun, Yong Huang, Zhiyong Ba Interference detection and correction algorithm for laser cladding trajectory of complex components. Computer Integrated Manufacturing Systems. 27 (01) (2021) 102-8. DOI: 10.13196/j.cims.2021.01.008.

26 C. Curran, KG Watkins, JM Lee Effect of wavelength and incident angle in the laser removal of particles from silicon wafers. ICALEO® 2001: Proceedings of the Laser Materials Processing Conference and Laser Microfabrication Conference.

27 G. Vereecke, E. Röhr, M. M. Heyns Influence of beam incidence angle on dry laser cleaning of surface particles. Applied Surface Science. 157 (1) (2000) 67-73. DOI: https://doi.org/10.1016/S0169-4332(99)00521-8.

28 Yang Min, Zhu Chengjun, Li Xiaojing, Wang Di The Influence of Data Process on Forming Precision of SLA Rapid Prototyping.: IEEE, 2012:486-9.

29 Hai Ru Yang, Guo Zhi Chen, Hui Zhen Wang, Liang Li 3d Geometric Model of Transparent Tower Equipment Display in OpenGL. Applied Mechanics and Materials. 722 (2014) 281-4. DOI: 10.4028/www.scientific.net/AMM.722.281.

30 Yingzhong Zhang, Zhenyan Yang, Guangzhi He, Yun Qin, Hong-chao Zhang Remanufacturing-Oriented Geometric Modelling for the Damaged Region of Components. Procedia Cirp. 29 (2015) 798-803. DOI: 10.1016/j.procir.2015.02.164.

31 Guang Yang, Wei Jun Liu, Wei Wang, Lan Yun Qin Research on the Rapid Slicing Algorithm Based on STL Topology Construction. Advanced Materials Research. 97-101 (2010) 3397-402. DOI: 10.4028/www.scientific.net/AMR.97-101.3397.

32 Hao Yuan Research on Sketch-Based Product Model Retrieval: University, Southeast, 2011.

33 Anders Hejlsberg, Mads Torgersen, Scott Wiltamuth, Pater Golde The C# Programming Language, 4th Edition. The MIT Press: Addison-Wesley Professional, 2010.

34 Yuanjin Li, Tao Chen, Defu Liu Path Planning for Laser Cladding Robot on Artificial Joint Surface Based on Topology Reconstruction. Algorithms. 13 (4) (2020) 93. DOI: 10.3390/a13040093.

35 N. Altin, E. Yazgan RCS prediction using fast ray tracing in Plücker coordinates.: IEEE, 2013:284-8.