Team Number:	2020230010136
Problem Chosen:	A

2020 APMCM summary sheet

With laser, the hatch tool of laser marking machine can be used to realize the work of making two-dimensional graphics. However, the selection and layout of hatch methods and paths greatly affect the difficulty and efficiency of incubation. The hatch problem of the laser marking machine on the two-dimensional closed graph can be equivalent to the problem of filling the two-dimensional closed curvilinear polygon with the curves connected by scattered points. Here, we provide two Python-based filling algorithms. They respectively correspond the zigzag hatch and the contour parallel hatch. Both of these two algorithms utilize vectors to the greatest extent on the basis of inward shrinking of the initial graphics to process the graphics lattice data, and use the distance between two points and the argument of vector as the basis for determining. Importantly, the algorithm automatically filters the lattice data at the sharp points of the figure after rational simplification, optimizes the shrinking of the arc with a very small radius of curvature. It also can ensure that the graphics formed after filtering can meet the requirements. This method breaks the limitation of the traditional algorithm that can only handle simple geometric figures. It provides the possibility to deal with complex curve polygons, and optimize the laser marking machine's refined processing of complex graphics. Meanwhile, we avoid solving a large number of linear equations in the algorithm and improve its efficiency. However, because of the filtering of the lattice, the graphics we get may have some unexpected deformation. Therefore, we still need some extra program to adjust it.

Keywords: Python filling algorithm vector zigzag hatch contour parallel hatch

Contents

I. Introduction

The sharpest knife, laser, has been widely used to mark specified 2D-compound curve graph. Efficiency is an important indicator for this craft. To maximize the efficiency, we need to make the hatch curves generated automatically and quickly by laser marking machine keep parallel to the boundary lines of the figures, and distribute evenly.[7]

1.1 Problem Restatement

The curve used by laser etching has a certain width. In order to simplify the calculation, we approximate it to a one-dimensional curve, and set the spacing of the curve to achieve the filling effect. Taking into account the moving efficiency of the robotic arm in actual industrial production, the curve used to fill the graph must be as continuous as possible. Based on the above discussion, we can equate the problem as the optimal solution for filling a two-dimensional curve polygon with a continuous curve.

1.2 Previous Work

In previous work, we have learned that there has been a work of depicting a vector diagram by calculating basic data through DSP. In the processing of bitmaps, due to the large storage space occupied by bitmap files, a small amount of pixel data is selected according to the set resolution during the processing of the marking software. Such processing greatly reduces the amount of marking data of the bitmap file and improves its marking speed. Increasing the resolution of the picture can improve the fineness of the graphic marking, but due to the limitation of the scanning speed, the marking time increases accordingly.

1.3 Problem Solving Strategies

For the two hatching methods mentioned in the question, we originally proposed two different general algorithm. They can solve such problems universally. For the zigzag hatching, we first indent the graphics formed by the original lattice. And by finding the extreme points of the graphics, a dividing line parallel to the x-axis is made according to the traversal direction to intersect the graphics at one point. Therefore, we divide the entire graphics into a number of areas. Finally, we do zigzag hatching from top to bottom in each partition. For the contour parallel hatch, we will iterate the steps of

indentation. In the process of iteration, we will remove points that will cause overlap in the coverage area. When the last curve generated by the iteration is about to cross, stop the iteration and partition the area formed by the last curve. Finally, we continue to iterate until the filling accuracy requirement is reached.

II. The Description of the algorithms

2.1 How do we use algorithms to implement the whole process?

-
-
-

2.2 The Reasonableness of approximating

1) :

2) From the perspective of the :

3) Compromise:

2.3 The local optimization and the overall optimization

-
-
- Virtually:

2.4 Algorithmic Complexity

III. Models

3.1 Basic Model

3.1.1 *Terms, Definitions and Symbols*

The signs and definitions are mostly generated from queuing theory.

3.1.2 *Assumptions*

3.1.3 *The Foundation of Model*

1) The utility function

- The cost of :
- The loss of :
- The weight of each aspect:
- Compromise:

Figure 1 关注我们公众号，学习更多知识

3) The overall optimization and the local optimization

- The overall optimization:
- The local optimization:
- The optimal number of :

3.1.4 *Solution and Result*

1) The solution of the integer programming: 2) Results:

3.1.5 *Analysis of the Result*

- Local optimization and overall optimization:
- Sensitivity: The result is quite sensitive to the change of the three parameters
-
- Trend:
- Comparison:

3.1.6 *Strength and Weakness*

Strength: The Improved Model aims to make up for the neglect of . The result seems to declare that this model is more reasonable than the Basic Model and much more effective than the existing design.

Weakness: Thus the model is still an approximate on a large scale. This has doomed to limit the applications of it.

IV. Conclusions

4.1 Conclusions of the problem

-
-
-
-

4.2 Methods used in our models

-
-
-
-

4.3 Applications of our models

-
-
-
-

V. Future Work

5.1 Another model

5.1.1 *The limitations of queuing theory*

5.1.2

5.1.3

5.1.4

VI. References

[1] Author, Title, Place of Publication: Press, Year of publication.

[2] author, paper name, magazine name, volume number: starting and ending page number, year of publication.

[3] author, resource title, web site, visit time (year, month, day).

[4] LaTeX资源和技巧学习 https://www.latexstudio.net

[5] LaTeX问题交流网站 https://wenda.latexstudio.net

[6] 模板库维护 https://github.com/latexstudio/APMCMThesis

[7] Kaposi, M. "IDIOPATHISCHES MULTIPLES PIGMENTSARKOM DER HAUT." Wiener Klinische Wochenschrift, vol. 4, no. 2, 1872, pp. 265273.

VII. Appendix

Listing 1: The matlab Source code of Algorithm

```
kk=2;[mdd,ndd]=size(dd);
while ~isempty(V)
[tmpd,j]=min(W(i,V));tmpj=V(j);
for k=2:ndd
[tmp1,jj]=min(dd(1,k)+W(dd(2,k),V));
tmp2=V(jj);tt(k-1,:)=[tmp1,tmp2,jj];
end
tmp=[tmpd,tmpj,j;tt];[tmp3,tmp4]=min(tmp(:,1));
if tmp3==tmpd, ss(1:2,kk)=[i;tmp(tmp4,2)];
else,tmp5=find(ss(:,tmp4)~=0);tmp6=length(tmp5);
if dd(2,tmp4)==ss(tmp6,tmp4)
ss(1:tmp6+1,kk)=[ss(tmp5,tmp4);tmp(tmp4,2)];
else, ss(1:3,kk)=[i;dd(2,tmp4);tmp(tmp4,2)];
end;end
dd=[dd,[tmp3;tmp(tmp4,2)]];V(tmp(tmp4,3))=[];
[mdd,ndd]=size(dd);kk=kk+1;
end; S=ss; D=dd(1,:);
```

Listing 2: The lingo source code

```
kk=2;
[mdd,ndd]=size(dd);
while ~isempty(V)
    [tmpd,j]=min(W(i,V));tmpj=V(j);
for k=2:ndd
    [tmp1,jj]=min(dd(1,k)+W(dd(2,k),V));
    tmp2=V(jj);tt(k-1,:)=[tmp1,tmp2,jj];
end
    tmp=[tmpd,tmpj,j;tt];[tmp3,tmp4]=min(tmp(:,1));
if tmp3==tmpd, ss(1:2,kk)=[i;tmp(tmp4,2)];
else,tmp5=find(ss(:,tmp4)~=0);tmp6=length(tmp5);
if dd(2,tmp4)==ss(tmp6,tmp4)
    ss(1:tmp6+1,kk)=[ss(tmp5,tmp4);tmp(tmp4,2)];
else, ss(1:3,kk)=[i;dd(2,tmp4);tmp(tmp4,2)];
```

```
end;
end
    dd=[dd,[tmp3;tmp(tmp4,2)]];V(tmp(tmp4,3))=[];
    [mdd,ndd]=size(dd);
    kk=kk+1;
end;
S=ss;
D=dd(1,:);
```