

Epicycles研究笔记3

使用离散点和离散傅里叶变换画出路径

方式一：2D图形绘制

```
In[*]:= g[pts_] := DynamicModule[{if, p, len}, if = InverseFourier[Complex @@@ pts];  
      |动态模块 |离散傅立叶逆变换 |复数  
      len = Length@pts;  
      |长度  
      Manipulate[p = Sum[if[[Mod[n + len, len] + 1]] Exp[I t n], {n, -m, m}];  
      |交互式操作 |求和 |模余 |... |虚数单位  
      Graphics[{Line@ReIm@Table[p, {t, 0, 2  $\pi$ , step}]}], {m, 100, 500, 1},  
      |图形 |线段 |实... |表格  
      {step, {0.5, 0.3, 0.1, 0.05, 0.03, 0.02, 0.01}, ControlType -> Setter}]]  
      |控件类型 |设置按钮
```

方式二：参数方程绘制

```
In[*]:= pp[pts_] := DynamicModule[{if, cs}, if = InverseFourier[Complex @@@ pts];  
      |动态模块 |离散傅立叶逆变换 |复数  
      Manipulate[cs = Join[if[[-m ;;]], if[;; m + 1]];  
      |交互式操作 |连接  
      ParametricPlot[ReIm@Sum[cs[[n]] Exp[(n - m - 1) I t], {n, 2 m + 1}], {t, 0, 2  $\pi$ },  
      |绘制参数图 |实... |求和 |指数形式 |虚数单位  
      {m, 100, 500, 1}]]
```

实例

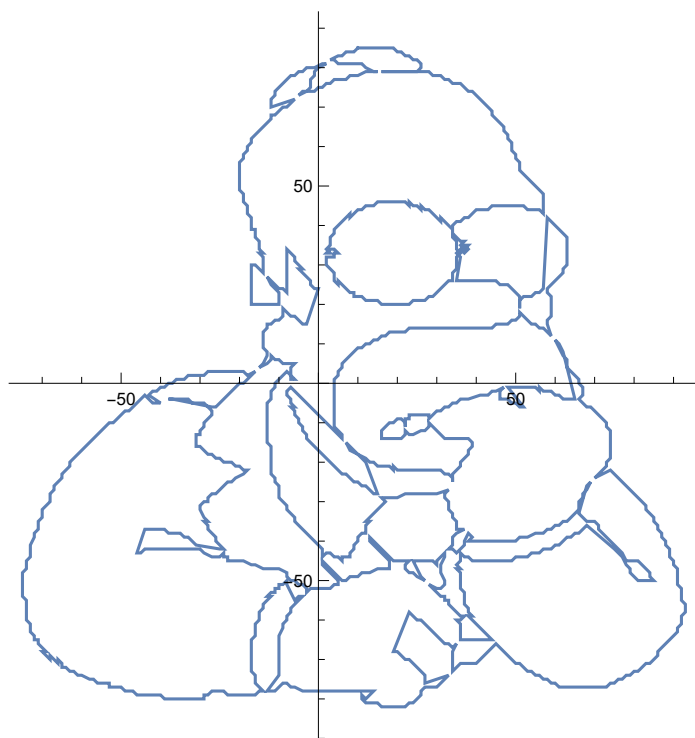
使用来自网络的参照图像

```

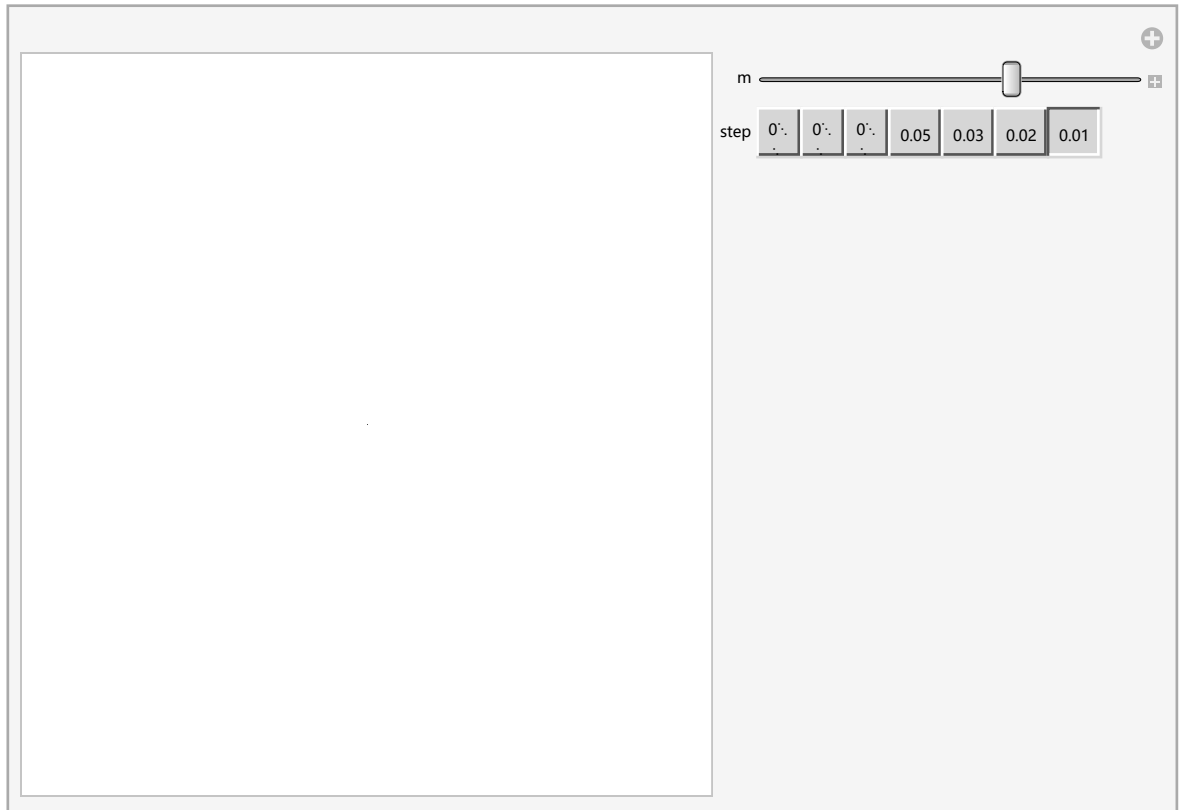
In[ ]:= img = Import["http://www.qedcat.com/misc/homer1_clean_pixel.png"];
           |导入
pts = PixelValuePositions[img, Black, 0.05];
           |像素位置 |黑色
pts = Map[{-90, -90} + # &, pts];
           |映射
elephantPlot = ListPlot[pts, AspectRatio → Automatic];
               |绘制点集 |宽高比 |自动
shortest = Last@FindShortestTour@pts;
           |最... |遍历各点的最短距离
pts = pts[[shortest]];
ListLinePlot[pts, AspectRatio → Automatic]
           |绘制点集的线条 |宽高比 |自动

```

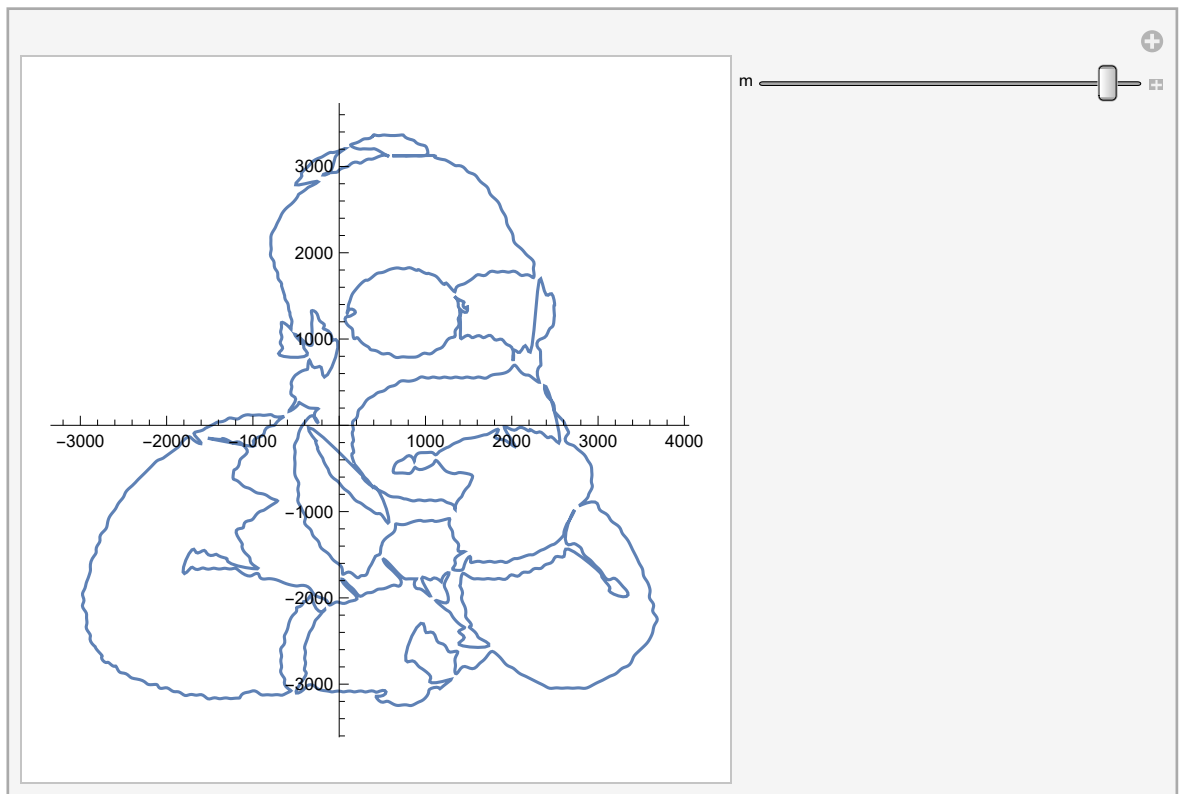
Out[]:=



In[*]:= **g[pts]**
 Out[*]=



In[*]:= **pp[pts]**
 Out[*]=



先将变换结果整理成 $\{-m, m\}$ 的形式，再转变成索引值

```
In[ ]:= DynamicModule[{if, cs, len, k}, len = Length@pts;
  |动态模块 |长度
  k = Ceiling[len / 2];
  |向上取整
  if = InverseFourier[Complex@@@pts];
  |离散傅立叶逆变换 |复数
  cs = Join[if[[k - len ;;]], if[[;; k]];
  |连接
  Manipulate[ParametricPlot[ReIm@Sum[cs[[n + Floor[len / 2] + 1]] Exp[n I t], {n, -m, m}],
  |交互式操作 |绘制参数图 |实... |求和 |向下取整 |指... |虚数单位
    {t, 0, 2  $\pi$ }], {m, 100, k, 1}]]
```

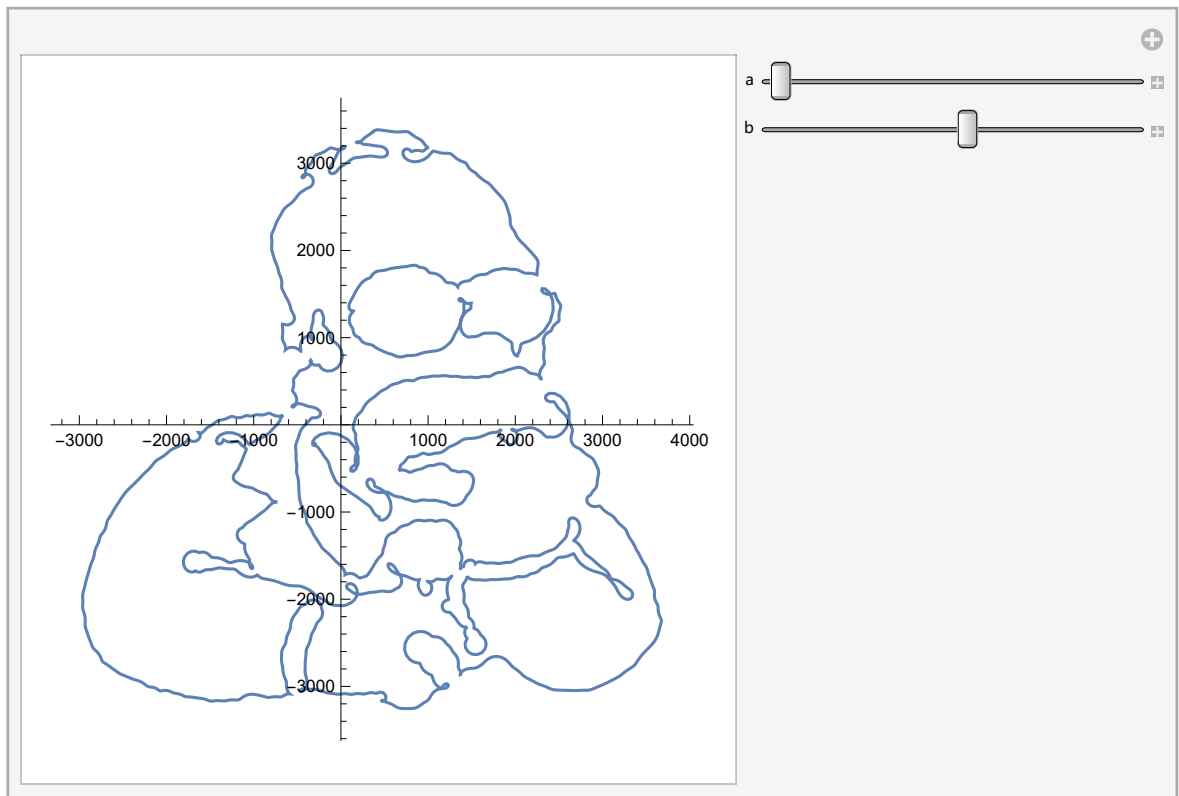
Out[]:=



直接使用周期数与索引的关系

```
In[ ]:= DynamicModule[{if, cs, len, k}, len = Length@pts;
  |动态模块 |长度
  if = Fourier[Complex@@@pts]; Manipulate[
  |傅立叶 |复数 |交互式操作
    ParametricPlot[ReIm@Sum[if[[Mod[n + len, len] + 1]] Exp[n I t], {n, -a, b}], {t, 0, 2  $\pi$ },
    |绘制参数图 |实... |求和 |模余 |指... |虚数单位
    {a, 100, Ceiling[len / 2] - 1, 1}, {b, 0, Ceiling[len / 2] - 1, 1}]]
  |向上取整 |向上取整
```

In[]:=



意外效果