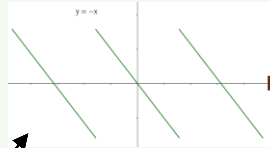
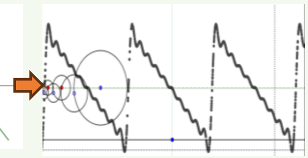
企画番号5

関数アート

佐部利旭　大久保侑也

（目的）円や楕円のように関数で絵を描きたい！
→フーリエ級数展開





フーリエ級数展開

ある周期関数（周期$2L$）を三角関数の和（振動数の異なる波の重ね合わせ）で表現する事が出来る。

$$f(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos\frac{k\pi}{L}t + b_k \sin\frac{k\pi}{L}t$$

$$a_k = \frac{1}{L}\int_{-L}^{L} f(t)\cos\frac{k\pi}{L}t\,dt \quad (k = 0,1,2,3,\dots)$$

$$b_k = \frac{1}{L}\int_{-L}^{L} f(t)\sin\frac{k\pi}{L}t\,dt \quad (k = 1,2,3,\dots)$$

方法

①描きたい画像を読み込む。

②画像(白黒)の輪郭を抽出。

③抽出した輪郭のx座標を基に積分する。（区分求積法）
y座標についても同様に積分する。

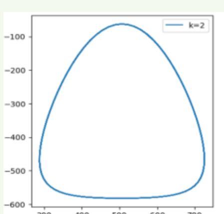
④変数tをパラメータとして、2つのフーリエ級数をそれぞれxy平面に出力する。

成果

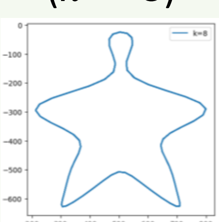
一筆書きでできるような単純な図形であれば、上手く描画できる。
フーリエ級数の項数に関して、kの値が小さい（早い）段階で図形の形が形成され始め、$k = 99$程度まで出力すれば、元画像に近い出力結果を得る。

（元画像）

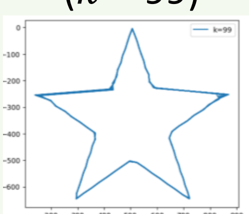
（出力結果$k = 2$）



$(k = 8)$



$(k = 99)$



（フーリエ級数）　$(k = 30, L = 1382.352)$

$$\begin{cases}
f_x(t) \\
= 1012.992 + 2.302\cos\frac{\pi}{L}t - 216.683\sin t\frac{\pi}{L} + 0.323\cos\frac{2\pi}{L}t - 14.955\sin\frac{2\pi}{L}t + 0.555\cos\frac{3\pi}{L}t - 17.675\sin\frac{3\pi}{L}t \\
+2.244\cos\frac{4\pi}{L}t - 52.664\sin\frac{4\pi}{L}t - 1.802\cos\frac{5\pi}{L}t + 33.871\sin\frac{5\pi}{L}t - 1.021\cos\frac{6\pi}{L}t + 16.018\sin\frac{6\pi}{L}t \\
+1.786\cos\frac{7\pi}{L}t - 23.937\sin\frac{7\pi}{L}t - 1.785\cos\frac{8\pi}{L}t + 20.98\sin\frac{8\pi}{L}t - 0.148\cos\frac{9\pi}{L}t + 1.557\sin\frac{9\pi}{L}t \\
+1.474\cos\frac{10\pi}{L}t - 13.728\sin\frac{10\pi}{L}t - 1.347\cos\frac{11\pi}{L}t + 11.43\sin\frac{11\pi}{L}t + 1.202\cos\frac{12\pi}{L}t - 9.34\sin\frac{12\pi}{L}t \\
+0.502\cos\frac{13\pi}{L}t - 3.584\sin\frac{13\pi}{L}t - 1.813\cos\frac{14\pi}{L}t + 12.07\sin\frac{14\pi}{L}t + 1.61\cos\frac{15\pi}{L}t - 10.051\sin\frac{15\pi}{L}t \\
-0.406\cos\frac{16\pi}{L}t + 2.397\sin\frac{16\pi}{L}t - 1.528\cos\frac{17\pi}{L}t + 8.321\sin\frac{17\pi}{L}t + 1.774\cos\frac{18\pi}{L}t - 9.184\sin\frac{18\pi}{L}t \\
-0.857\cos\frac{19\pi}{L}t + 4.175\sin\frac{19\pi}{L}t - 0.321\cos\frac{20\pi}{L}t + 1.495\sin\frac{20\pi}{L}t + 1.657\cos\frac{21\pi}{L}t - 7.325\sin\frac{21\pi}{L}t \\
-1.606\cos\frac{22\pi}{L}t + 6.749\sin\frac{22\pi}{L}t + 0.52\cos\frac{23\pi}{L}t - 2.085\sin\frac{23\pi}{L}t + 1.044\cos\frac{24\pi}{L}t - 3.983\sin\frac{24\pi}{L}t \\
-2.032\cos\frac{25\pi}{L}t + 7.459\sin\frac{25\pi}{L}t + 1.098\cos\frac{26\pi}{L}t - 3.854\sin\frac{26\pi}{L}t + 0.365\cos\frac{27\pi}{L}t - 1.224\sin\frac{27\pi}{L}t \\
-1.506\cos\frac{28\pi}{L}t + 4.931\sin\frac{28\pi}{L}t + 1.698\cos\frac{29\pi}{L}t - 5.344\sin\frac{29\pi}{L}t - 0.539\cos\frac{30\pi}{L}t + 1.648\sin\frac{30\pi}{L}t \\
f_y(t) \\
= -756.3 - 259.739\cos\frac{\pi}{L}t - 2.771\sin\frac{\pi}{L}t + 55.582\cos\frac{2\pi}{L}t + 1.169\sin\frac{2\pi}{L}t + 8.817\cos\frac{3\pi}{L}t + 0.28\sin\frac{3\pi}{L}t \\
+42.762\cos\frac{4\pi}{L}t + 1.827\sin\frac{4\pi}{L}t + 10.094\cos\frac{5\pi}{L}t + 0.531\sin\frac{5\pi}{L}t + 12.441\cos\frac{6\pi}{L}t + 0.789\sin\frac{6\pi}{L}t \\
-0.524\cos\frac{7\pi}{L}t - 0.035\sin\frac{7\pi}{L}t + 1.623\cos\frac{8\pi}{L}t + 0.141\sin\frac{8\pi}{L}t - 4.306\cos\frac{9\pi}{L}t - 0.416\sin\frac{9\pi}{L}t \\
+0.337\cos\frac{10\pi}{L}t + 0.03\sin\frac{10\pi}{L}t - 1.492\cos\frac{11\pi}{L}t - 0.173\sin\frac{11\pi}{L}t + 3.448\cos\frac{12\pi}{L}t + 0.452\sin\frac{12\pi}{L}t \\
+1.417\cos\frac{13\pi}{L}t + 0.194\sin\frac{13\pi}{L}t + 2.946\cos\frac{14\pi}{L}t + 0.434\sin\frac{14\pi}{L}t + 0.836\cos\frac{15\pi}{L}t + 0.15\sin\frac{15\pi}{L}t \\
+1.343\cos\frac{16\pi}{L}t + 0.247\sin\frac{16\pi}{L}t - 0.881\cos\frac{17\pi}{L}t - 0.164\sin\frac{17\pi}{L}t + 0.066\cos\frac{18\pi}{L}t + 0.004\sin\frac{18\pi}{L}t \\
-0.877\cos\frac{19\pi}{L}t - 0.185\sin\frac{19\pi}{L}t + 0.487\cos\frac{20\pi}{L}t + 0.101\sin\frac{20\pi}{L}t + 0.132\cos\frac{21\pi}{L}t + 0.032\sin\frac{21\pi}{L}t \\
+0.951\cos\frac{22\pi}{L}t + 0.223\sin\frac{22\pi}{L}t + 0.582\cos\frac{23\pi}{L}t + 0.14\sin\frac{23\pi}{L}t + 1.04\cos\frac{24\pi}{L}t + 0.273\sin\frac{24\pi}{L}t \\
-0.045\cos\frac{25\pi}{L}t - 0.01\sin\frac{25\pi}{L}t + 0.225\cos\frac{26\pi}{L}t + 0.06\sin\frac{26\pi}{L}t - 0.441 cos\frac{27\pi}{L}t - 0.131\sin\frac{27\pi}{L}t \\
+0.023\cos\frac{28\pi}{L}t + 0.005\sin\frac{28\pi}{L}t - 0.22\cos\frac{29\pi}{L}t - 0.065\sin\frac{29\pi}{L}t + 0.293\cos\frac{30\pi}{L}t + 0.099\sin\frac{30\pi}{L}t
\end{cases}$$

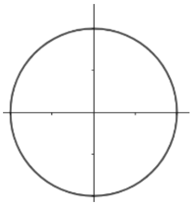
課題と考察

描く図形の中が塗りつぶされていない画像では、二重線で描画される箇所があり、上手く描画できない。
図形の明暗の差を輪郭として認識する為、図形が塗りつぶされていないと座標が余分に抽出されてしまう事が原因だと考えられる。

（補足資料）

パラメータ表示

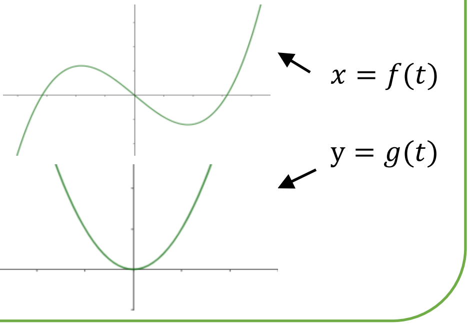
例えば単位円は以下のようにパラメータtを用いてxとyの関係を間接的に表現できる。



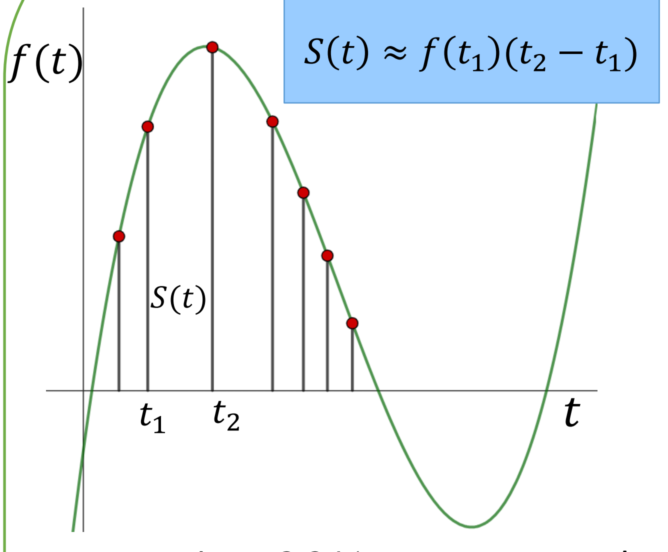
$$\begin{cases} x = \cos t \\ y = \sin t \end{cases}$$

以下のようにパラメータ表示すれば、xとyについてそれぞれフーリエ級数展開すればよい。

$$\begin{cases} x = f(t) \\ y = g(t) \end{cases}$$



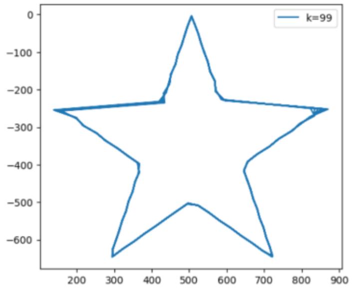
区分求積法



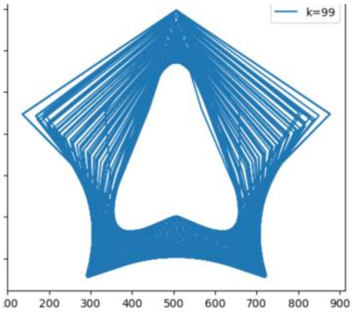
取得した輪郭の座標が上記の説明における$f(t)$や$g(t)$から得られたものと考えると、$f(t)$や$g(t)$の概形は分からずとも、区分求積法による積分が可能になる。

今回はtの間隔として、取得した座標間の距離の総和つまり描く図形の周囲の長さを(座標の数-1)で割ったものを採用しており、星形図形の場合その値は、およそ2.341である。
この値に設定したときが最もよい結果を得られた。

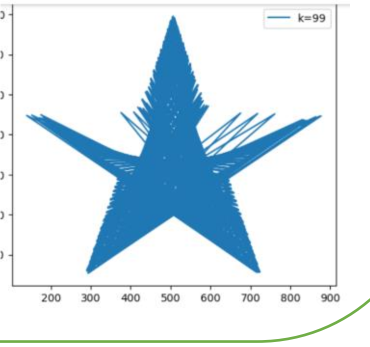
$\Delta t = 2.341$



$\Delta t = 0.234$



$\Delta t = 0.023$

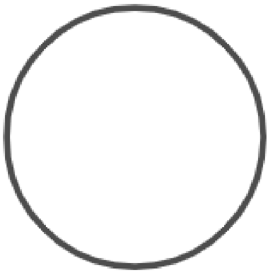


（補足資料）

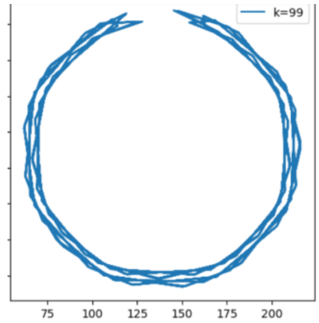
輪郭抽出の課題

N: 積分に用いた座標の数

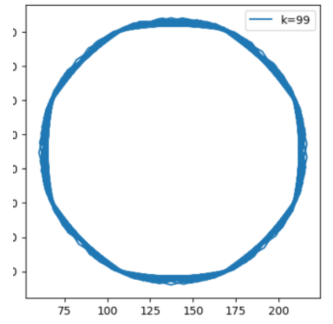
（元画像）



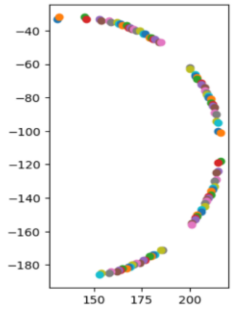
（修正前）$N = 400$



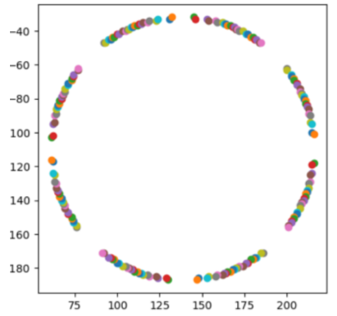
（修正後）$N = 200$



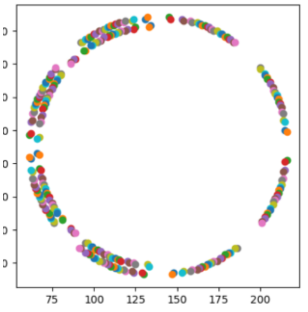
(0〜100)



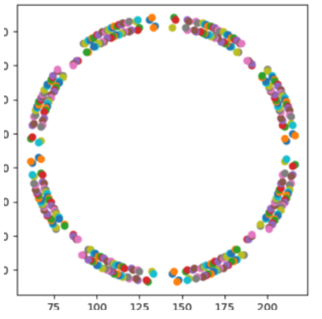
(0〜200)



(0〜300)



(0〜400)



輪郭の座標が二重に抽出されてしまう。

作成したプログラム

1

```
import numpy as np
import matplotlib.pyplot as plt
import math

#画像ファイルのアップロード
from google.colab import files
uploaded_file = files.upload()

#ファイル名の取得
uploaded_file_name = next(iter(uploaded_file))
print(uploaded_file_name)

from google.colab.patches import cv2_imshow
#関数imreadで画像を読み込む
cv_img = cv2.imread('circle.png')

#cv2_imshowで表示
cv2_imshow(cv_img)
#グレー画像に変換
img_dst0 = cv2.cvtColor(cv_img, cv2.COLOR_BGR2GRAY)
#Gaussianオペレータ
img_dst1 = cv2.GaussianBlur(img_dst0, (11,11), 1)
#2値化
thresh = 180
rest, img_dst2 = cv2.threshold(img_dst1, thresh, 255, 0)
cv2_imshow(img_dst2)

# 輪郭を抽出する.
contours, hierarchy = cv2.findContours(
    img_dst2, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE
)
x_list = []
y_list = []
for i in range(1, len(contours)):
        if len(contours[i]) > 0:
            buf_np = contours[i].flatten() # numpyの多重配列になっているため、一旦展開する.
            #print(buf_np)
            for i, elem in enumerate(buf_np):
                if i%2==0:
                    x_list.append(elem)
                else:
                    y_list.append(-1*elem)
N = len(x_list)
print(N)
```

2

```
#積分
a1 = [0]*100
b1 = [0]*100
pi = np.pi
T = 0
for i in range(0,N-1):
    L = (x_list[i]-x_list[i+1])*(x_list[i]-x_list[i+1]) + (y_list[i]-y_list[i+1])*(y_list[i]-y_list[i+1])
    L = math.sqrt(L)
    T += L
T = T/2
a = -T
b = T

#0からスタートする連番（100回繰り返す）
for n in range(100):
  s = 0
  t = a
  for i in range(0,N-1):
    s += (x_list[i])*np.cos(n*pi*t/T)*(b-a)/(N-1)
    t += (b-a)/(N-1)
  a1[n] = s / T
```

3

```
#0からスタートする連番（100回繰り返す）
for n in range(100):
  t = a
  s = 0
  for i in range(0,N-1):
    s += (x_list[i])*np.sin(n*pi*t/T)*(b-a)/(N-1)
    t += (b-a)/(N-1)
  b1[n] = s / T
s1 = a1[0] / 2

a2 = [0]*100
b2 = [0]*100

#0からスタートする連番（100回繰り返す）
for n in range(100):
  s = 0
  t = a
  for i in range(0,N-1):
    s += y_list[i]*np.cos(n*pi*t/T)*(b-a)/(N-1)
    t += (b-a)/(N-1)
  a2[n] = s / T
```

4

```
#0からスタートする連番（100回繰り返す）
for n in range(100):
  t = a
  s = 0
  for i in range(0,N-1):
    s += y_list[i]*np.sin(n*pi*t/T)*(b-a)/(N-1)
    t += (b-a)/(N-1)
  b2[n] = s / T
s2 = a2[0] / 2
x = np.linspace(-10000, 10000, 500)
for k in range(1, 100):
  s1 += a1[k]*np.cos(k*pi*x/T) + b1[k]*np.sin(k*pi*x/T)
  s2 += a2[k]*np.cos(k*pi*x/T) + b2[k]*np.sin(k*pi*x/T)
  if k == 2 or k == 4 or k == 8 or k == 15 or k == 50 or k == 75 or k == 99:
    plt.plot(s1, s2, label="k={}".format(k))
    plt.legend(prop={"size": 10})
    plt.gca().set_aspect('equal', adjustable='box')
    plt.show()
```