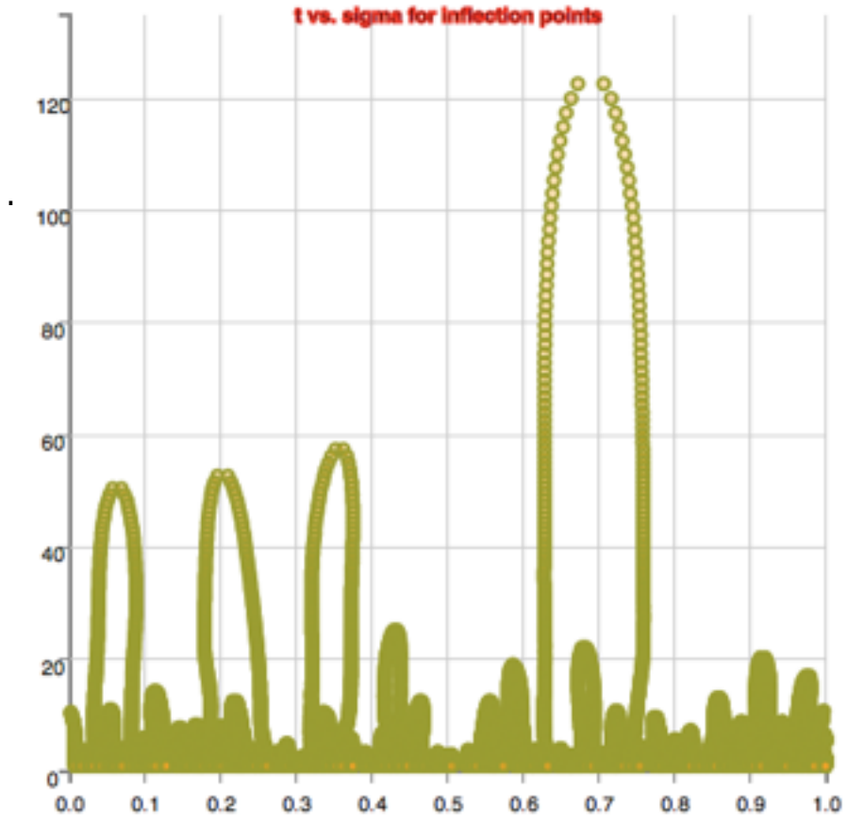
Inflection points for sigma > 0



Scale-Based Description and Recognition of Planar Curves and Two-Dimensional Shapes

FARZIN MOKHTARIAN AND ALAN MACKWORTH

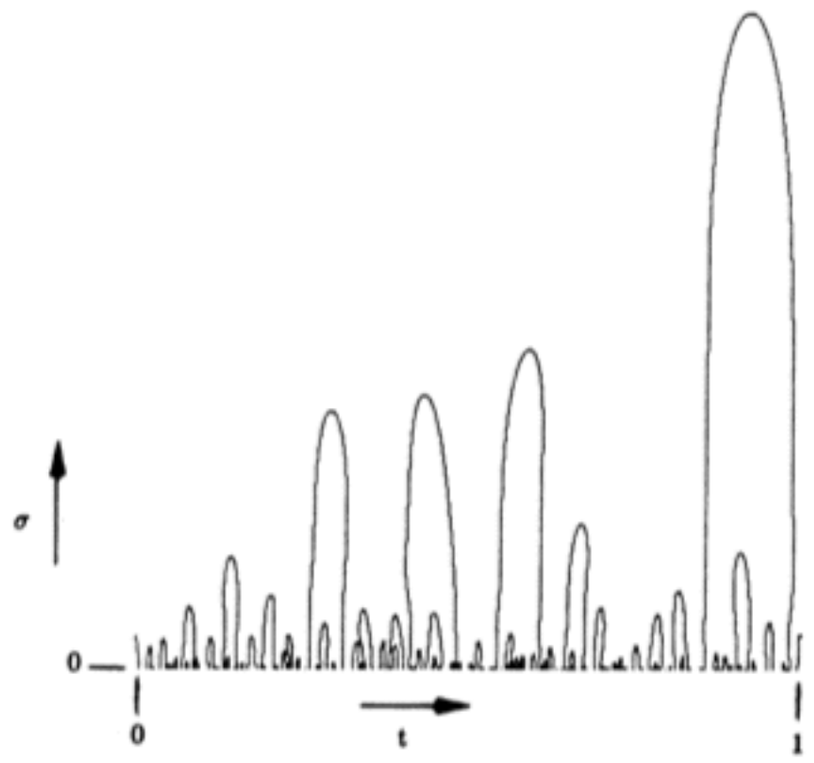
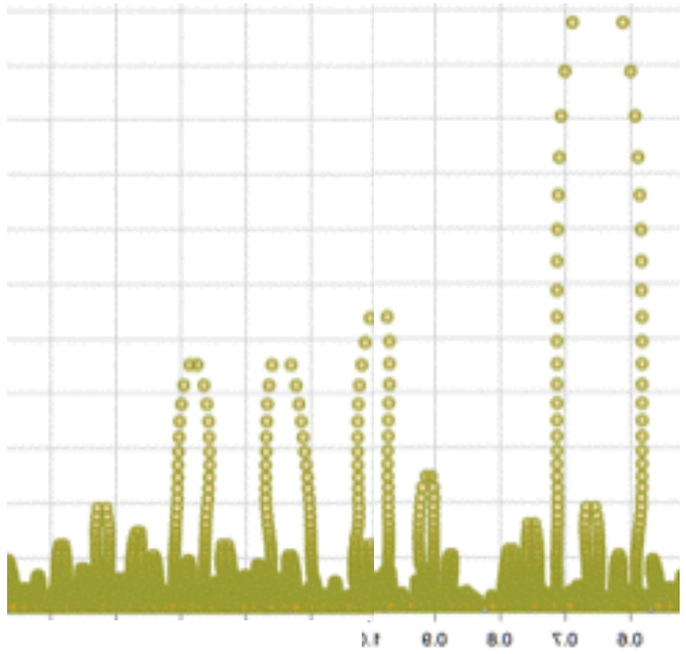


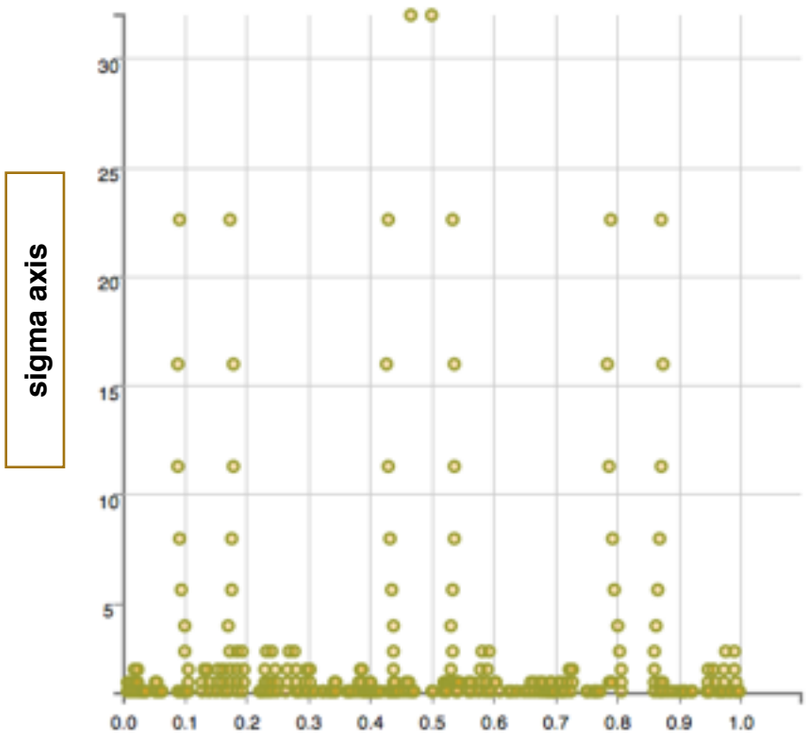
Fig. 3. Generalized scale space image of Africa.

my scale space image agrees with theirs (my implementation follows their paper)

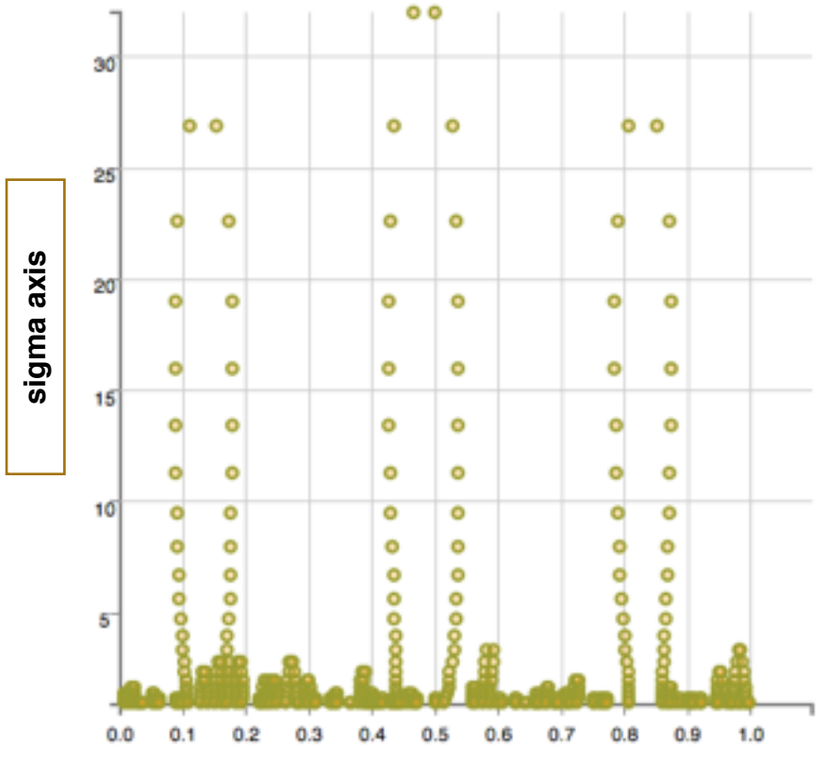


contour finder

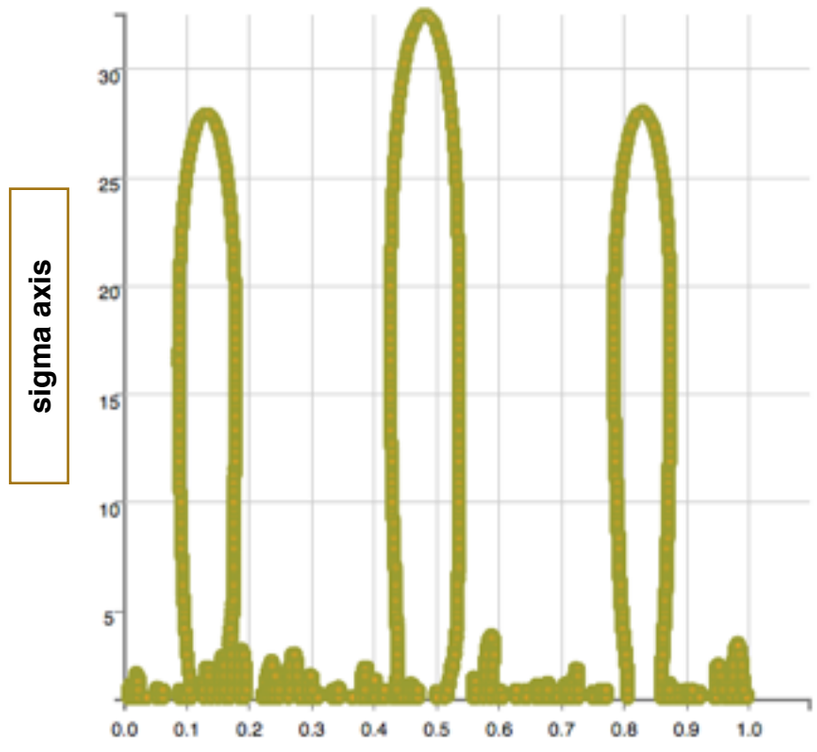
scale space image produced for sigma factors of sqrt(2)



scale space image produced for sigma factors of 2^(1/8)



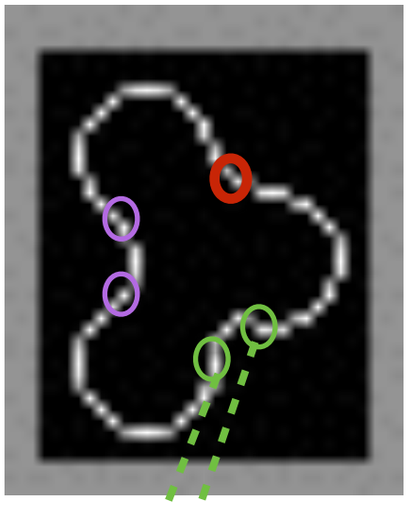
scale space image produced for sigma factors of 2^(1/128)



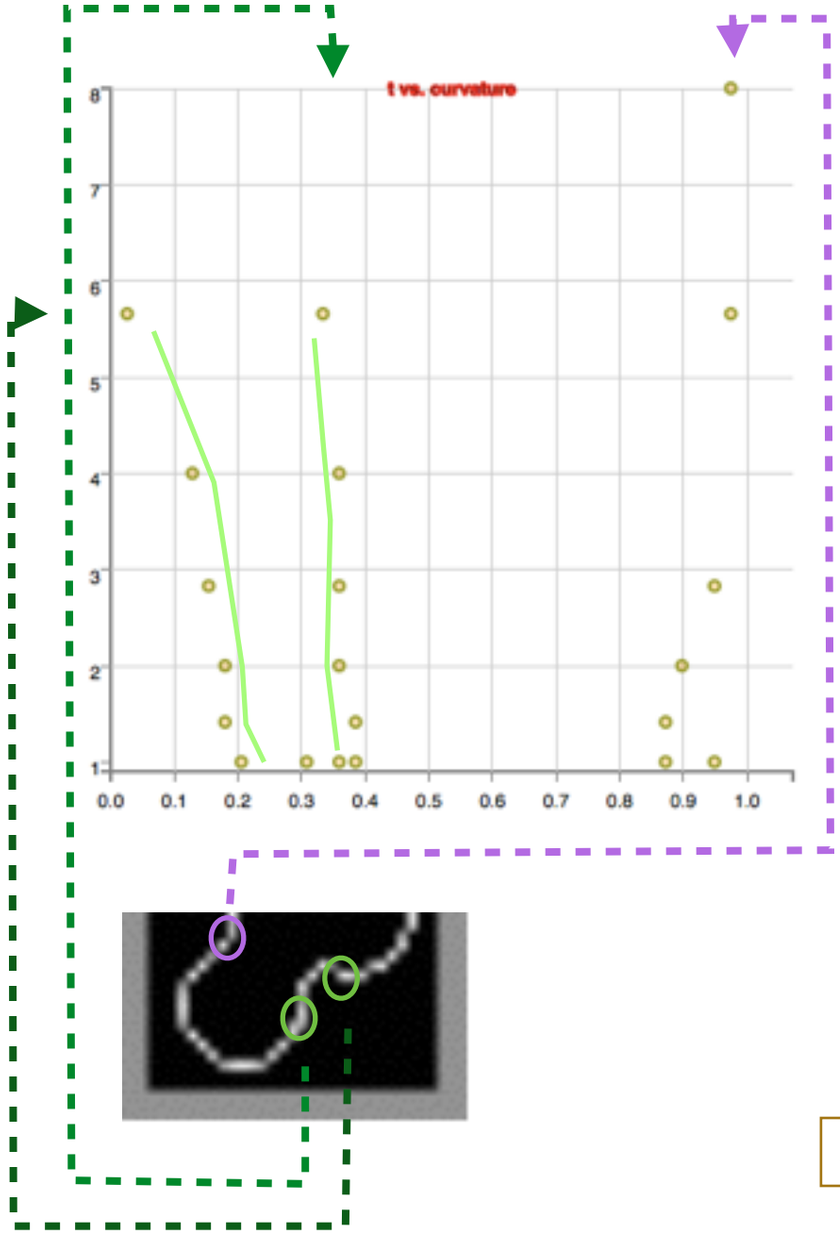
There is an error in estimating the peak of a contour for fastest creation of scale space images (<= sqrt(2)). That error can be reduced overall, by having more contours in the final solution. For an error < 10% in determining a contour's peak height, one should choose a sigma factor of 2^(1/8) or smaller. It takes 2^3 more convolutions if the smaller sigma factor of 2^(1/8) is used instead of 2^(1/2).

contour finder

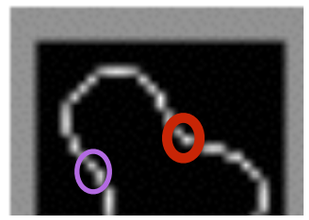
single closed curve's scale space image:

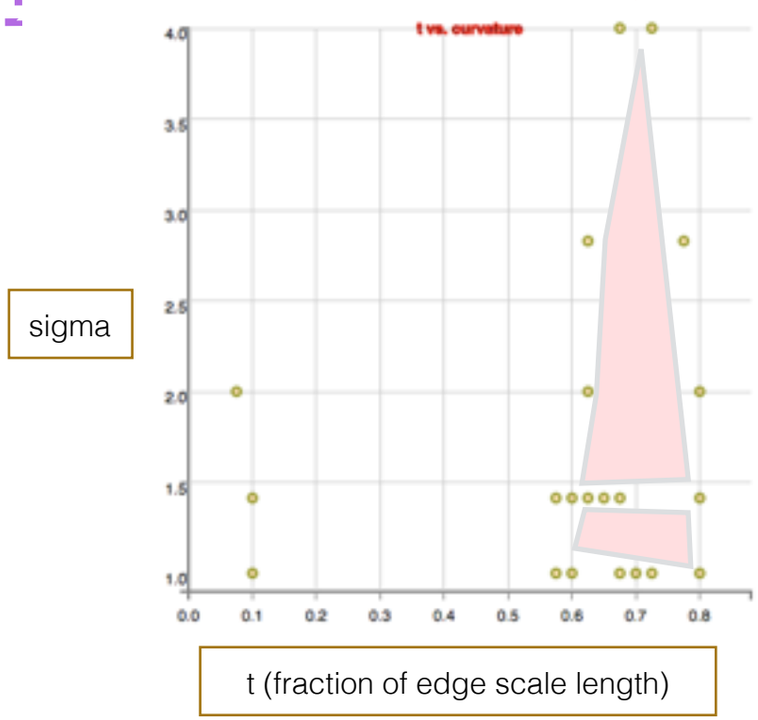


divided into 2 curves manually, then made into scale images (that is, 2 open curves possibly create open contours):

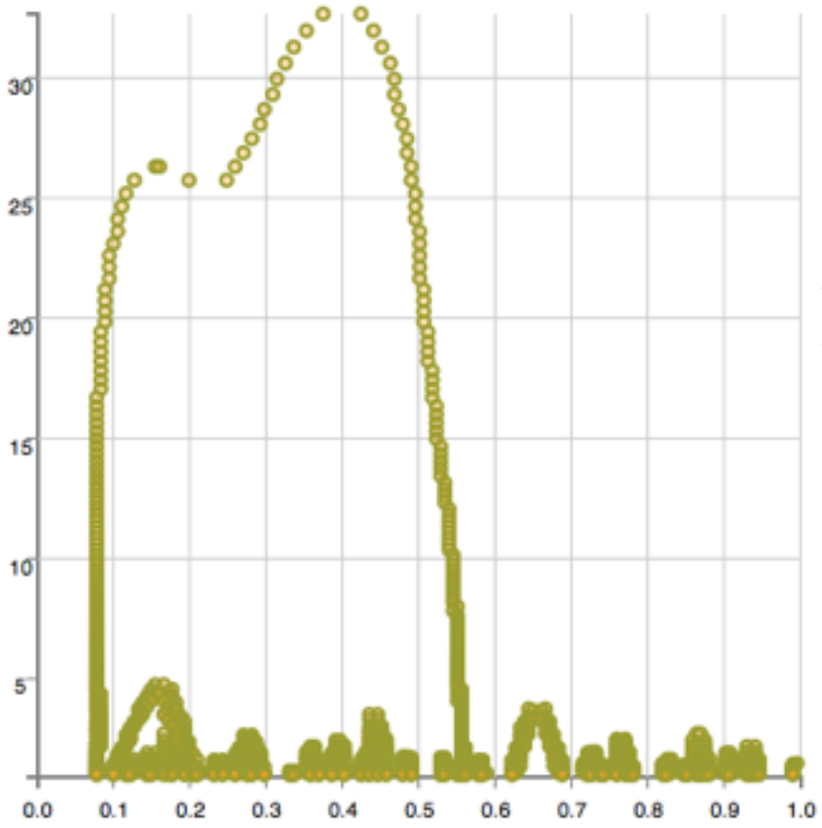


∴ open contours are hard to match in another image's scale images

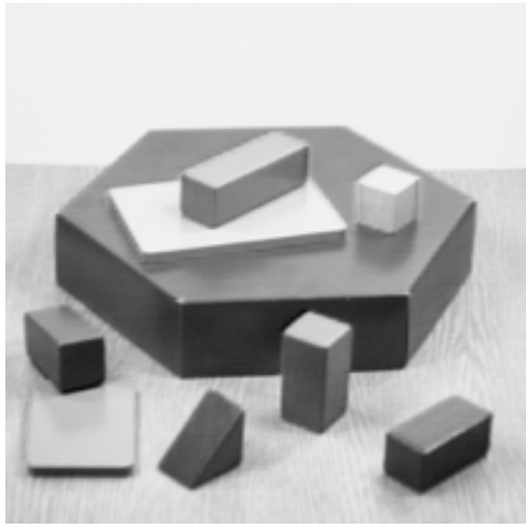




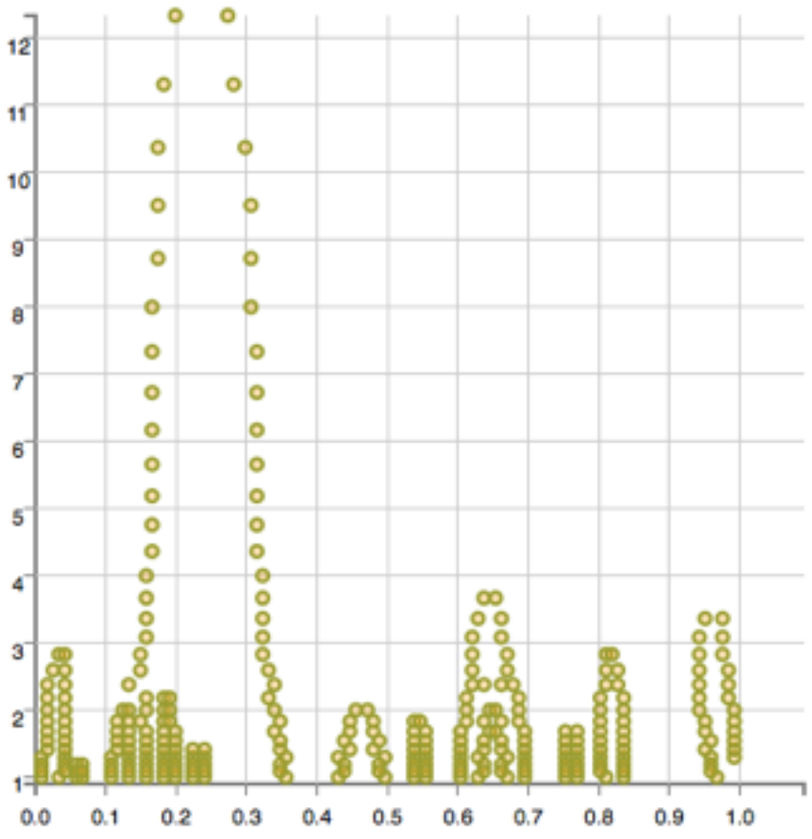
contour finder



Latest results
with improved line
thinner.



The contour finder looks for the peaks at the highest sigma and then follows the left and right branches down, subtracting that contour from the scale space image. each contour is found that way and subtracted to a lower threshold.



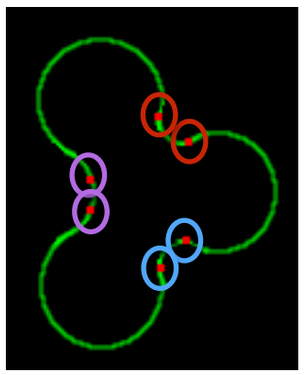
Older results
with older line
thinner.

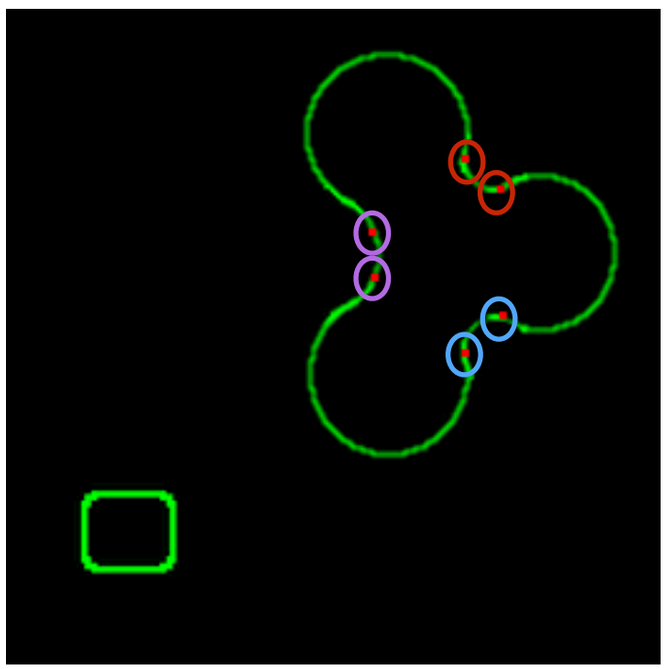
Then contours from one image are matched with the contours of another image (an image having the same content, that is).

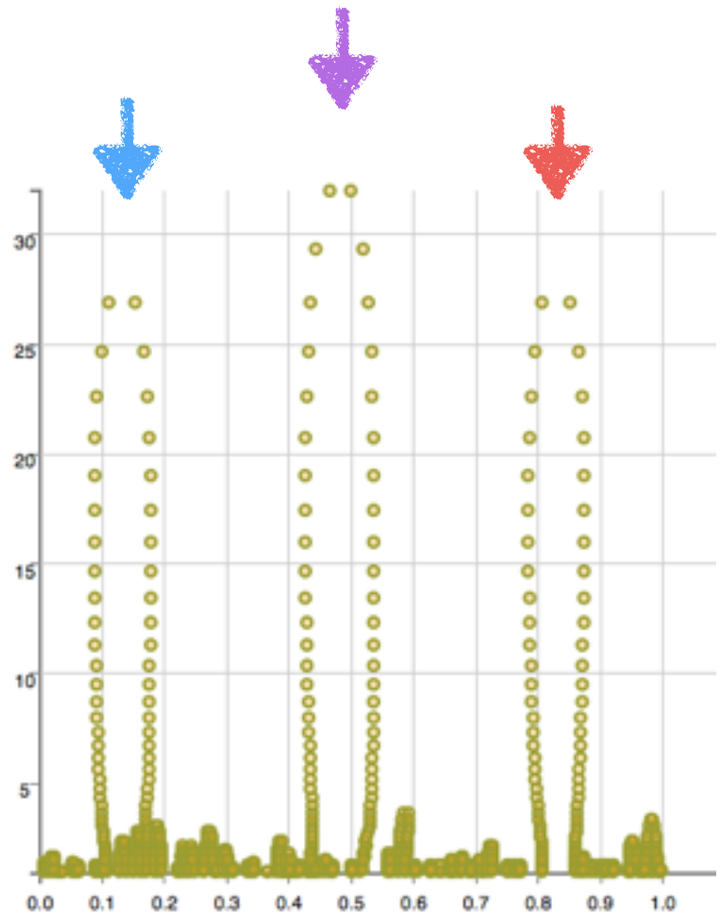
Then euclidean transformation parameters rotation, scale, and translation are calculated from the matched contour peak coordinates.

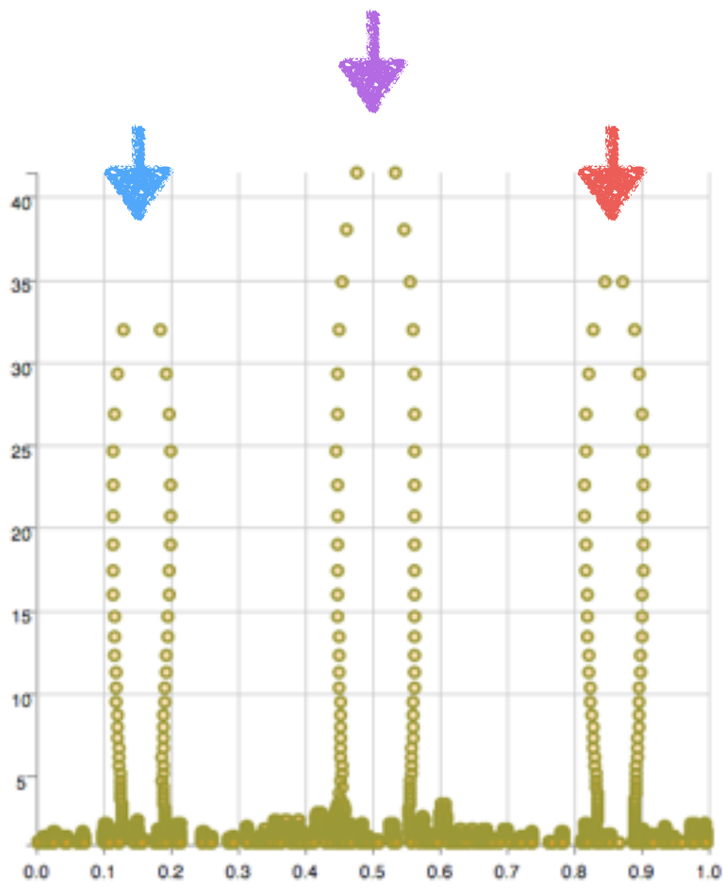
Then, the parameters are refined with small changes and applied to the edges in image 1. The transformed closed curve edges from image 1 are compared to the closest matches in image 2 to find the best fitting transformation parameters.

coordinate transformation, after matching contours





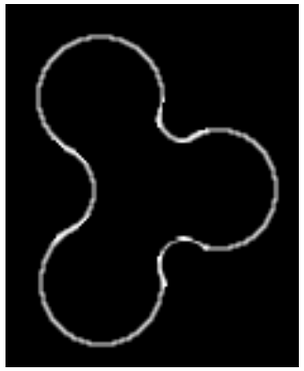


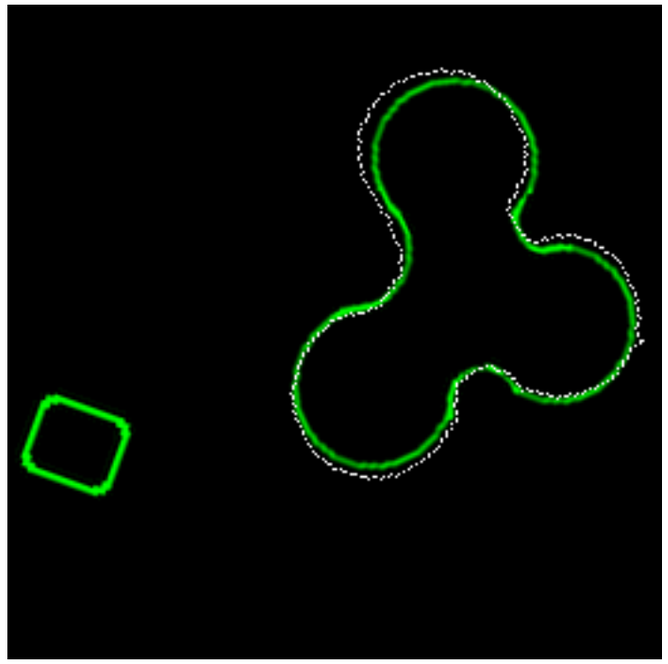


scale should be 1.3
rotation should be 360 - 20

```
Contour matcher solution scale=1.354256510734558
Contour matcher solution shift=-0.1688411384820938
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72)  CONTOUR PEAK2: (43.336529, 0.504396) (157, 108) (159, 101)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99)  CONTOUR PEAK2: (34.148750, 0.157143) (190, 143) (177, 150)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54)  CONTOUR PEAK2: (34.896511, 0.859341) (200, 85) (209, 97)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=1 offsetImgY2=26
rotationInRadians=6.0030236
rotationInDegrees=343.94791799660214
scale=1.3542565
translationX=108.1361
translationY=15.72716
```

apply coordinate transformation

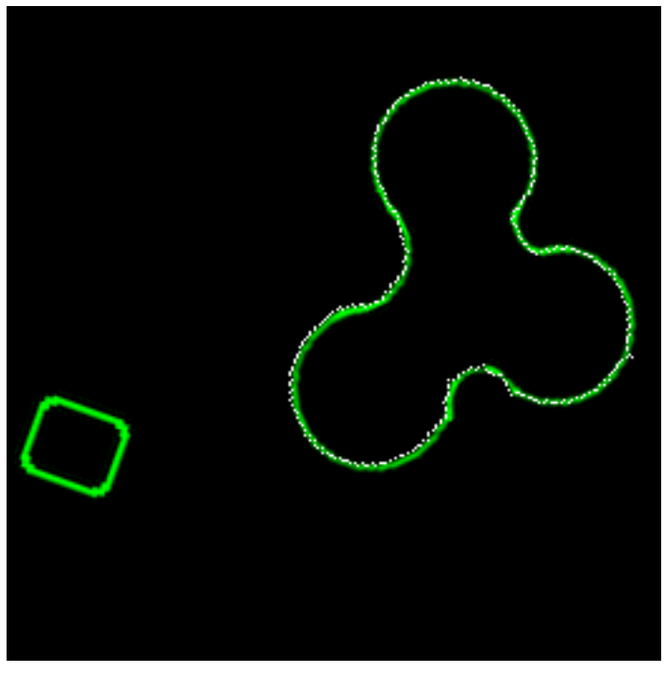




```
rotationInRadians=6.0030236
rotationInDegrees=343.94791799660214
scale=1.3542565
translationX=108.1361
translationY=15.72716
```

scale should be 1.3
rotation should be 360-20

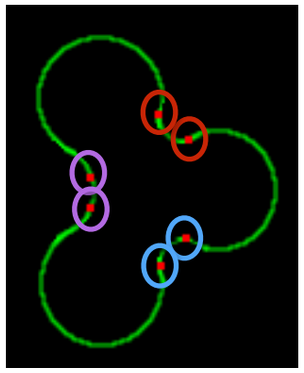
This shows that it's difficult to estimate scale unless some of the inflection points are further from the center of the shape

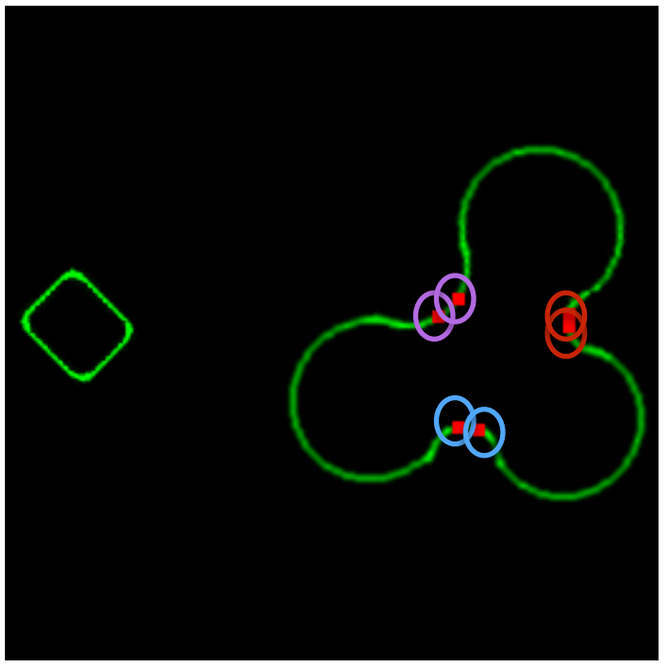


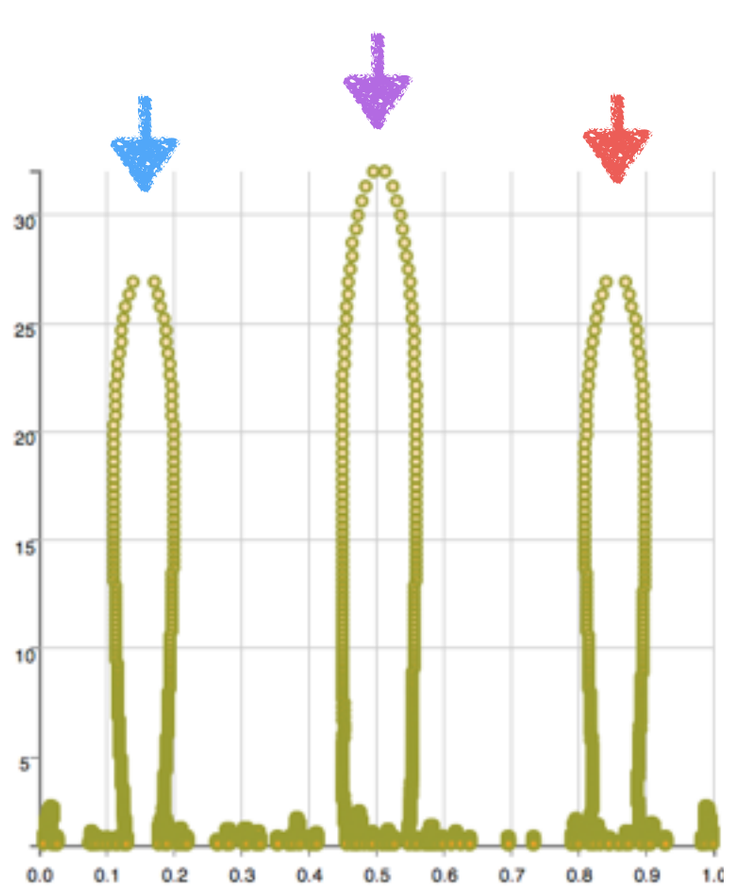
After Refinement

```
rotationInRadians=5.915757
rotationInDegrees=338.94791899582935
scale=1.3042566
translationX=111.0
translationY=20.0
```

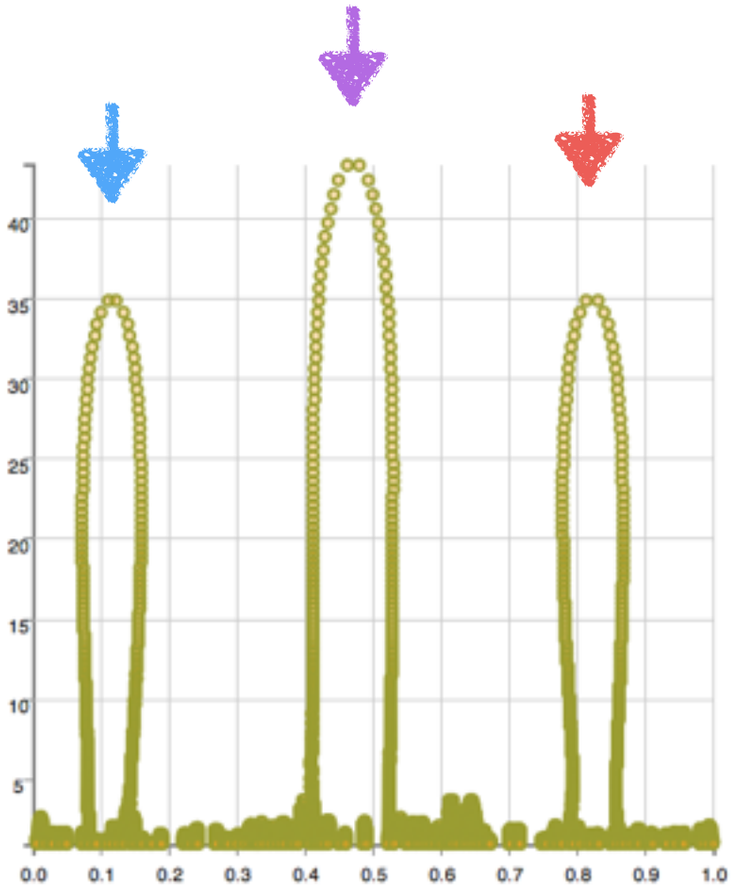
coordinate transformation, after matching contours





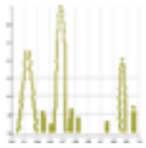


reversed to have CCW ordering



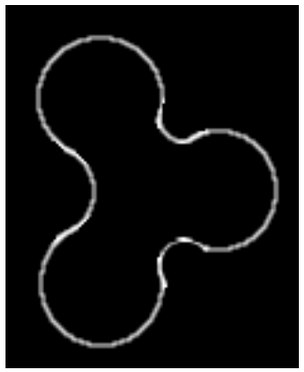
reversed to have CCW ordering

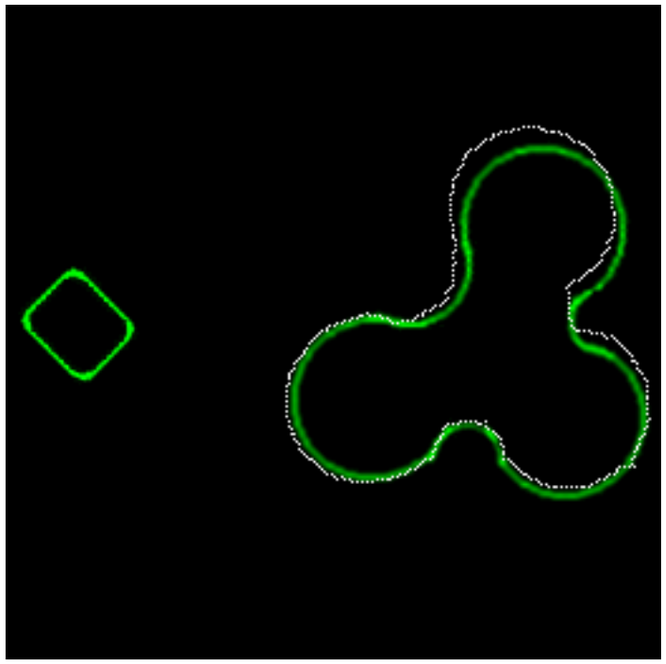
scale should be 1.3
rotation should be 360 - 45



```
Contour matcher solution scale=1.354256510734558
Contour matcher solution shift=-0.1425349861383438
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (43.336529, 0.530702) (172, 123) (180, 116)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (34.896511, 0.179825) (188, 168) (180, 167)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (34.896511, 0.884868) (224, 122) (224, 127)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=3 offsetImgY2=52
rotationInRadians=5.5627766
rotationInDegrees=318.7236195803948
scale=1.3542565
translationX=119.984764
translationY=39.13688
```

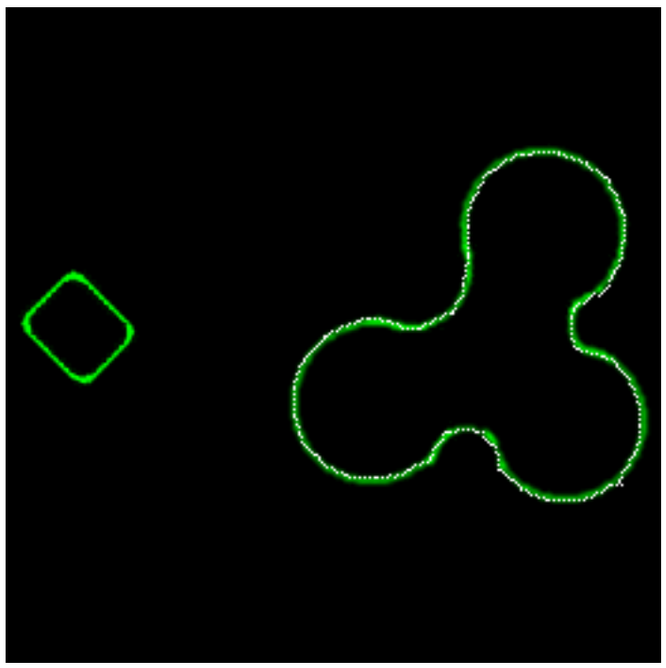
apply coordinate transformation





```
offsetImgX1=10 offsetImgY1=10
offsetImgX2=3 offsetImgY2=52
rotationInRadians=5.5627766
rotationInDegrees=318.7236195803948
scale=1.3542565
translationX=119.984764
translationY=39.13688
```

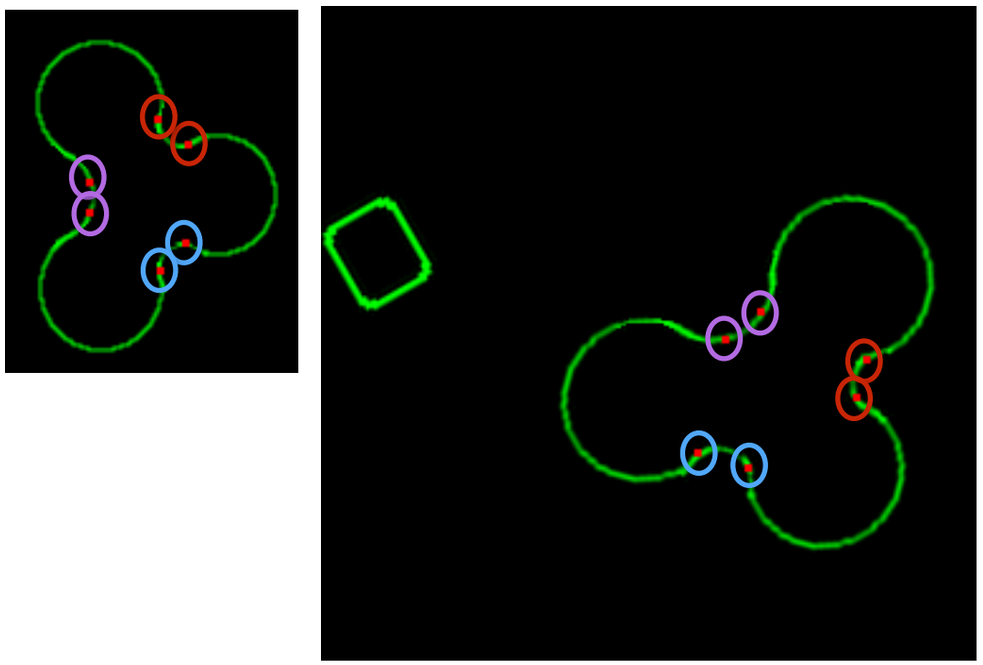
scale should be 1.3
rotation should be 360-45

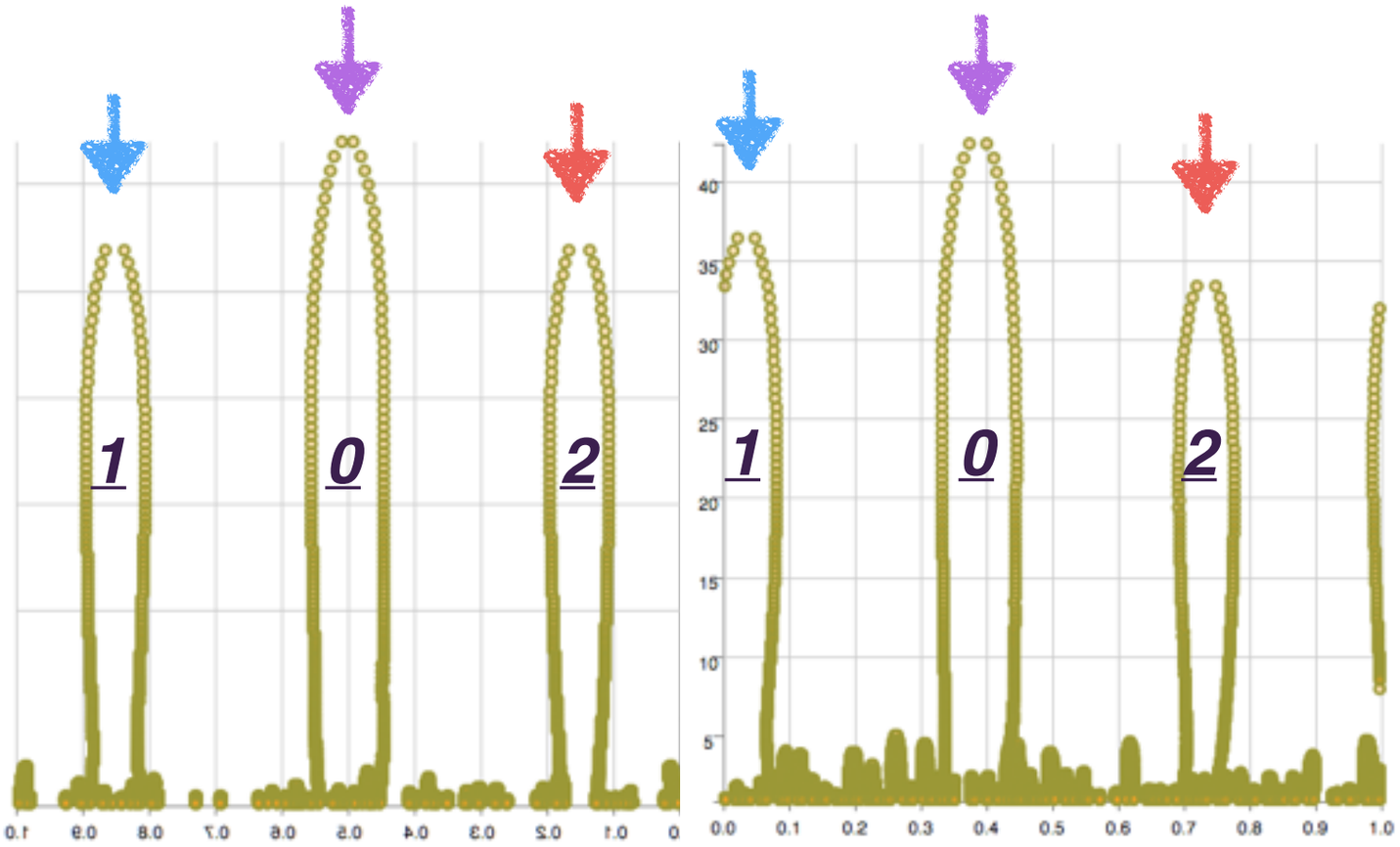


After Refinement

```
rotationInRadians=5.47551
rotationInDegrees=313.72362057962204
scale=1.3042566
translationX=124.0
translationY=47.0
```

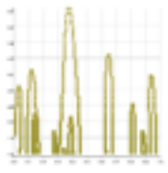
coordinate transformation, after matching contours





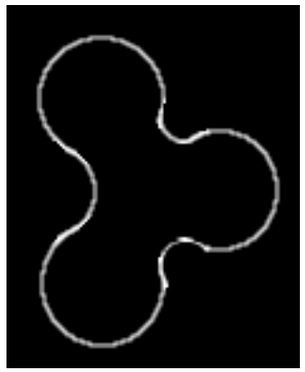
reversed to have CCW ordering

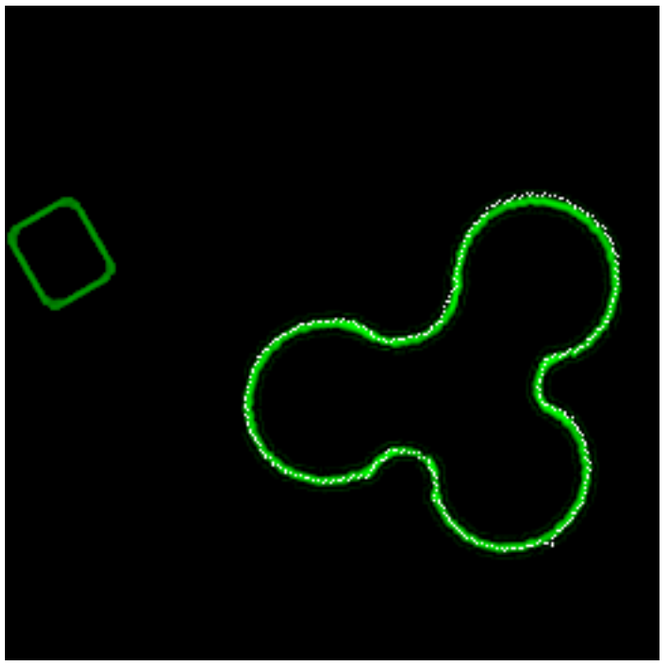
scale should be 1.3
rotation should be 360 - 60



```
Contour matcher solution scale=1.325237512588501
Contour matcher solution shift=-0.2715021073818207
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (42.407913, 0.387309) (161, 133) (172, 126)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (36.441517, 0.035011) (168, 182) (157, 176)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (33.417011, 0.734136) (215, 143) (213, 156)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=0 offsetImgY2=71
rotationInRadians=5.2398615
rotationInDegrees=300.2219485151509
scale=1.3252375
translationX=107.96625
translationY=59.87517
```

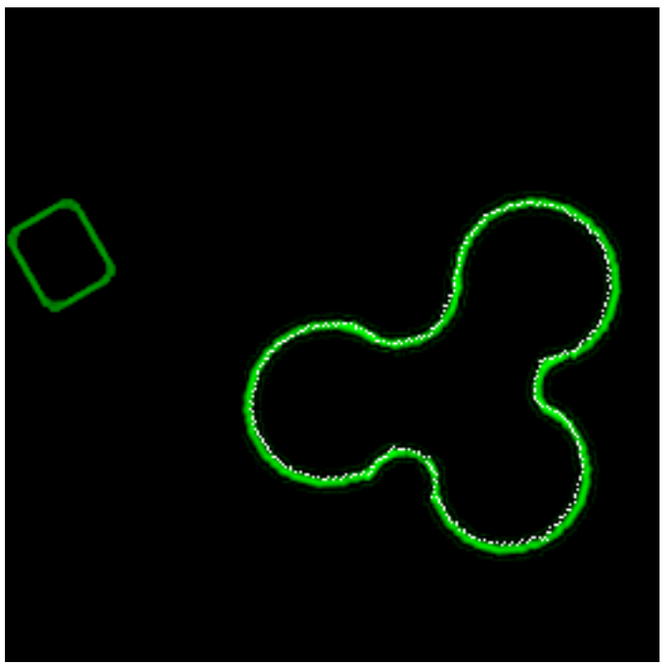
apply coordinate transformation





```
Contour matcher solution scale=1.325237512588501
Contour matcher solution shift=-0.2715021073818207
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72)
  CONTOUR PEAK2: (42.407913, 0.387309) (161, 133) (172, 126)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99)
  CONTOUR PEAK2: (36.441517, 0.035011) (168, 182) (157, 176)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54)
  CONTOUR PEAK2: (33.417011, 0.734136) (215, 143) (213, 156)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=0 offsetImgY2=71
rotationInRadians=5.2398615
rotationInDegrees=300.2219485151509
scale=1.3252375
translationX=107.96625
translationY=59.87517
```

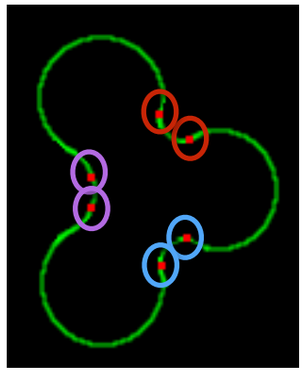
scale should be 1.3
rotation should be 360 - 60

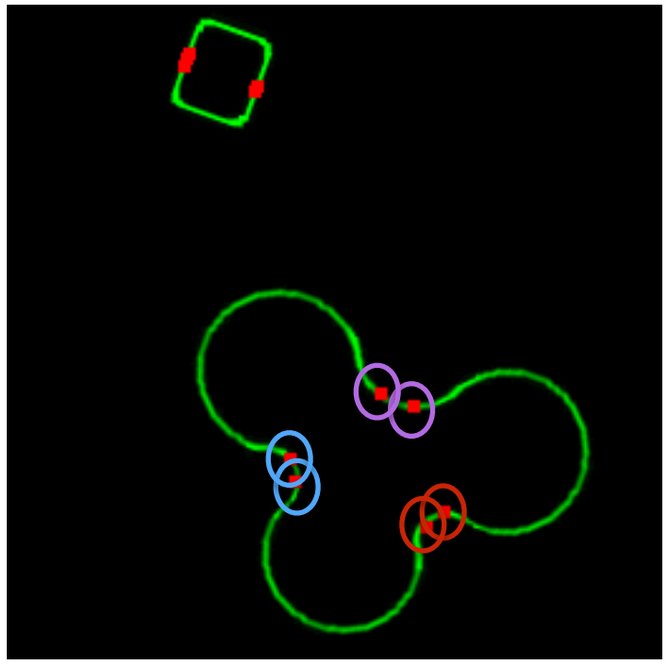


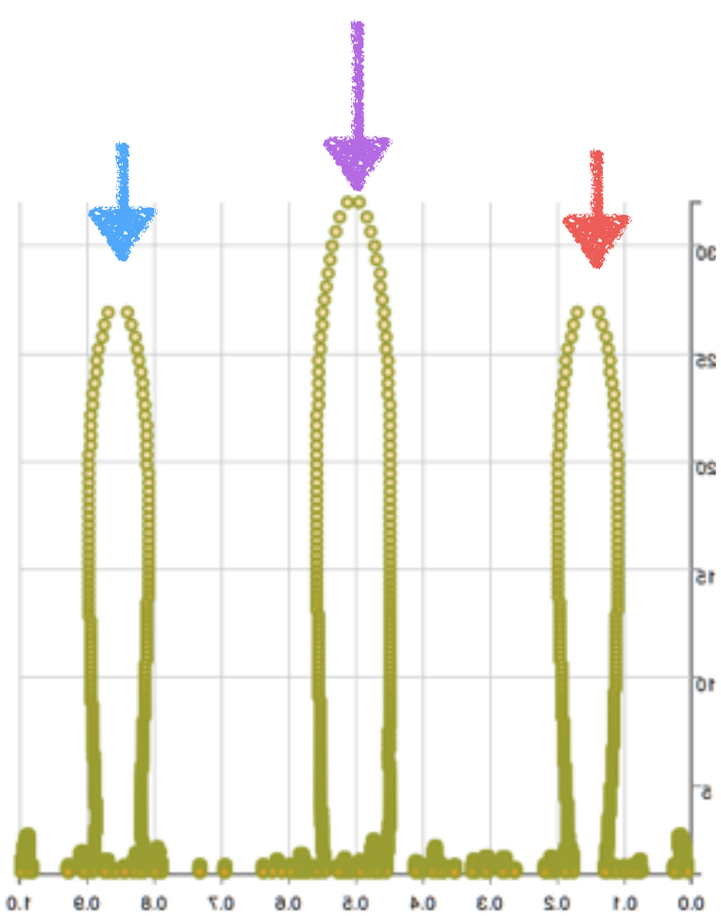
After Refinement

```
rotationInRadians=5.2398615
rotationInDegrees=300.2219485151509
scale=1.2752376
translationX=110.0
translationY=63.0
```

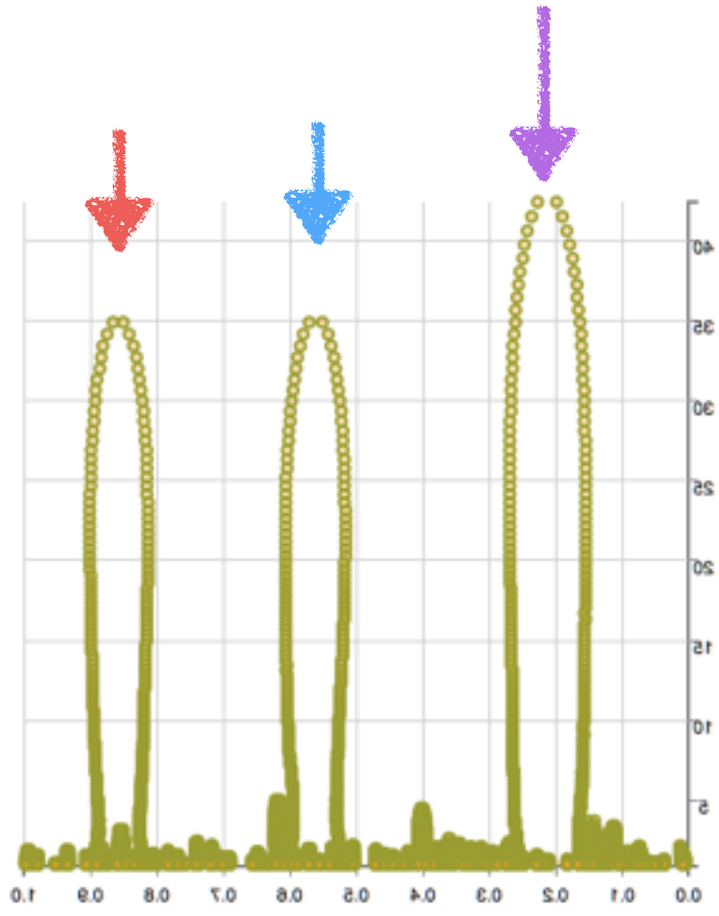
coordinate transformation, after matching contours





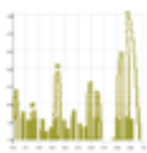


reversed to have CCW ordering



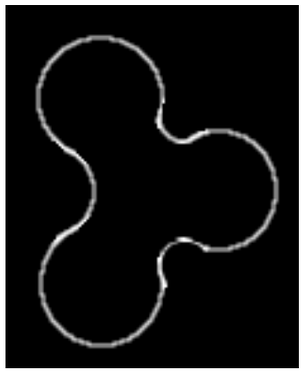
reversed to have CCW ordering

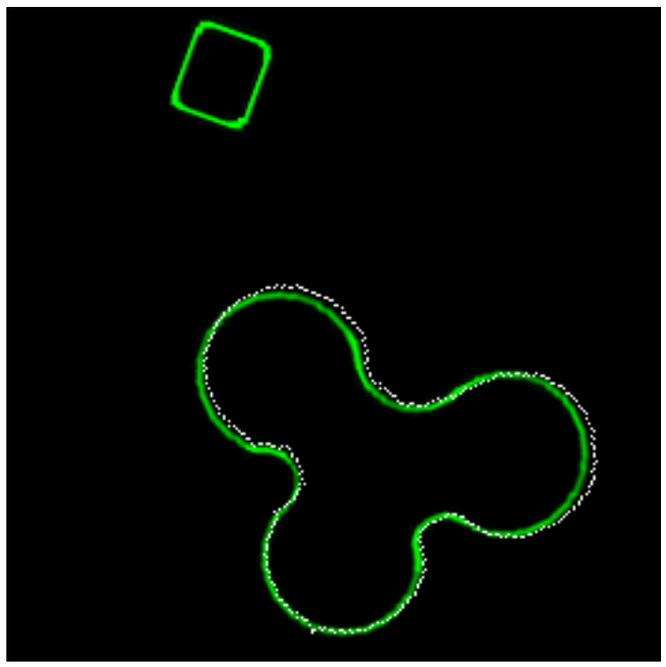
scale should be 1.3
rotation should be 360 -110



```
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (42.407913, 0.787445) (148, 154) (161, 159)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (34.896511, 0.439427) (114, 189) (112, 180)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (34.896511, 0.142070) (173, 201) (166, 207)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=62 offsetImgY2=1
rotationInRadians=4.336784
rotationInDegrees=248.47941332561248
scale=1.3252375
translationX=70.99721
translationY=91.054344
```

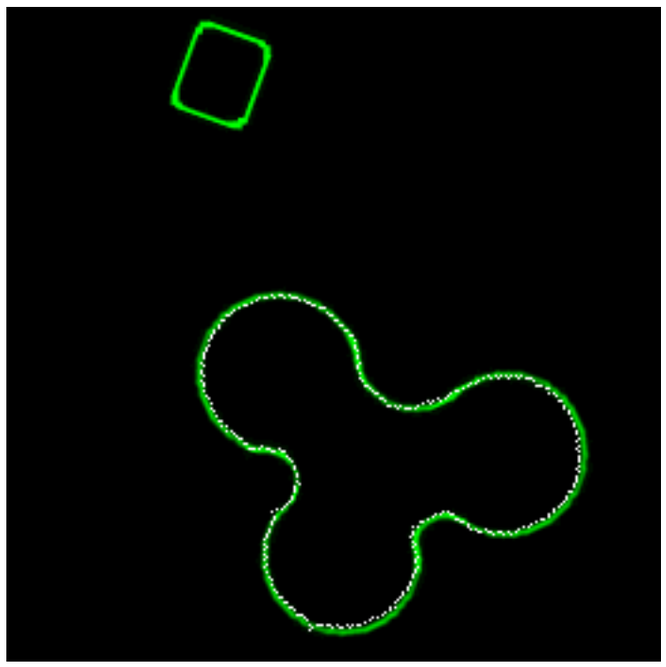
apply coordinate transformation





```
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72)
CONTOUR PEAK2: (42.407913, 0.787445) (148, 154) (161, 159)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99)
CONTOUR PEAK2: (34.896511, 0.439427) (114, 189) (112, 180)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54)
CONTOUR PEAK2: (34.896511, 0.142070) (173, 201) (166, 207)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=62 offsetImgY2=1
rotationInRadians=4.336784
rotationInDegrees=248.47941332561248
scale=1.3252375
translationX=70.99721
translationY=91.054344
```

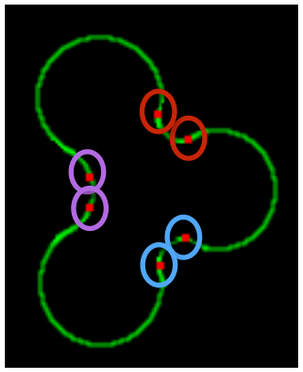
scale should be 1
rotation should be 360 - 110 (250)

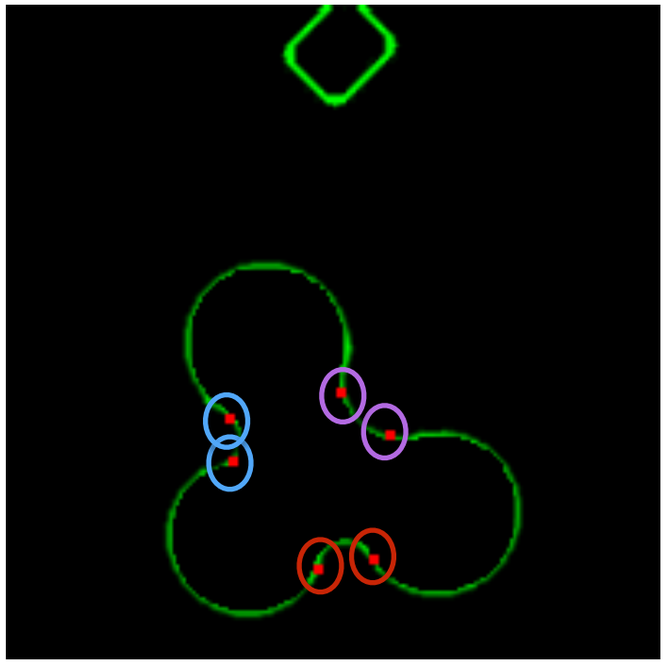


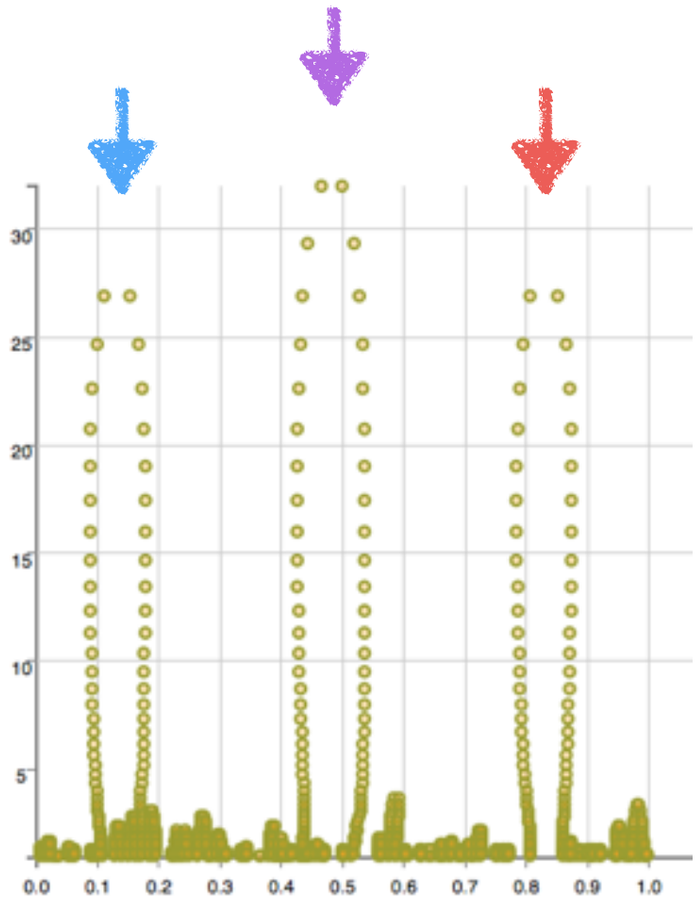
After Refinement

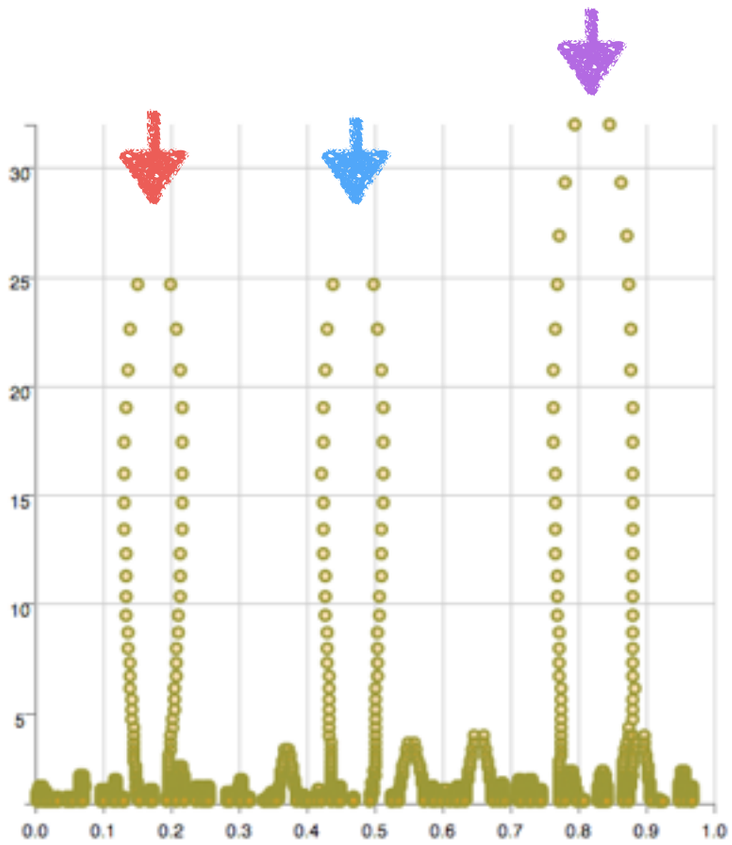
```
rotationInRadians=4.354237
rotationInDegrees=249.4794076616157
scale=1.2752376
translationX=71.0
translationY=95.0
```

coordinate transformation, after matching contours





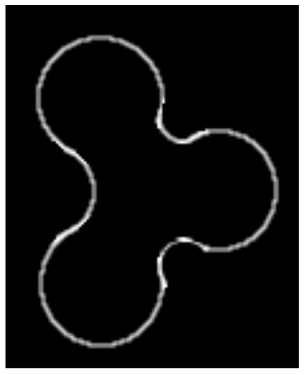


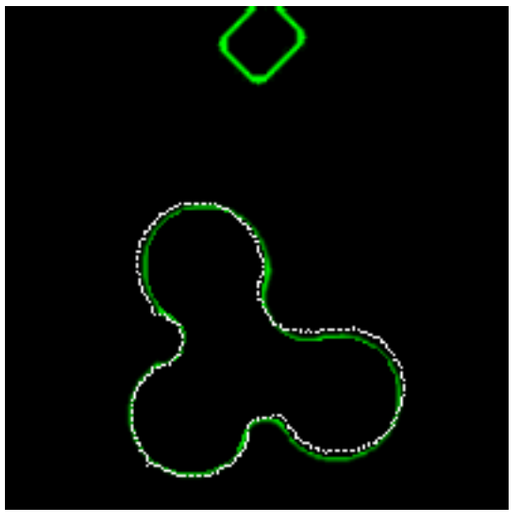


scale should be 1.0
rotation should be 360 - 135

```
Contour matcher solution scale=1.0218971967697144
Contour matcher solution shift=0.3104316294193268
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (32.700951, 0.818444) (103, 120) (115, 132)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (25.768024, 0.468300) (69, 136) (69, 127)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (25.768024, 0.175793) (105, 164) (97, 166)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=46 offsetImgY2=0
rotationInRadians=3.9818497
rotationInDegrees=228.14318077005984
scale=1.0218972
translationX=32.712353
translationY=70.68979
```

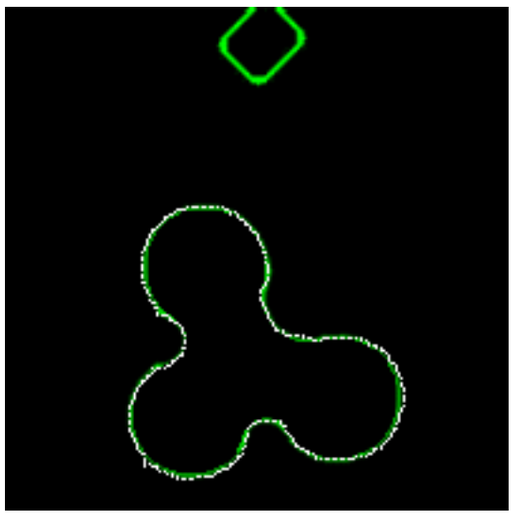
apply coordinate transformation





```
offsetImgX1=10 offsetImgY1=10
offsetImgX2=46 offsetImgY2=0
rotationInRadians=3.9818497
rotationInDegrees=228.14318077005984
scale=1.0218972
translationX=32.712353
translationY=70.68979
```

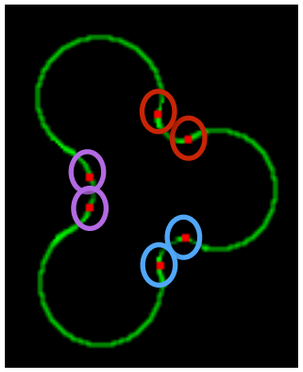
scale should be 1
rotation should be 360 - 135

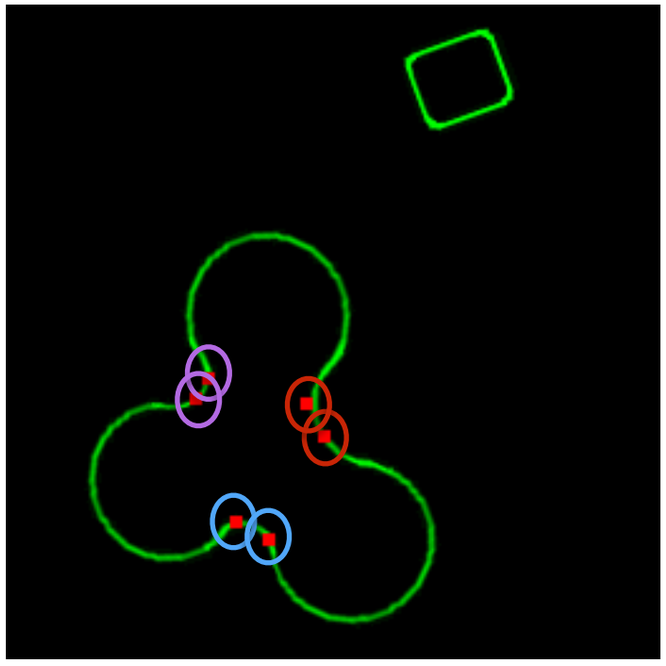


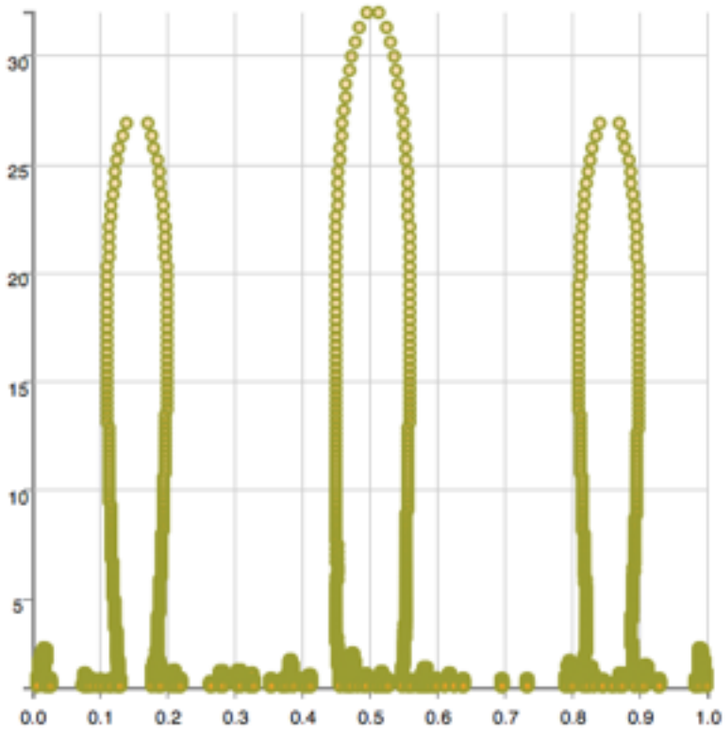
After Refinement

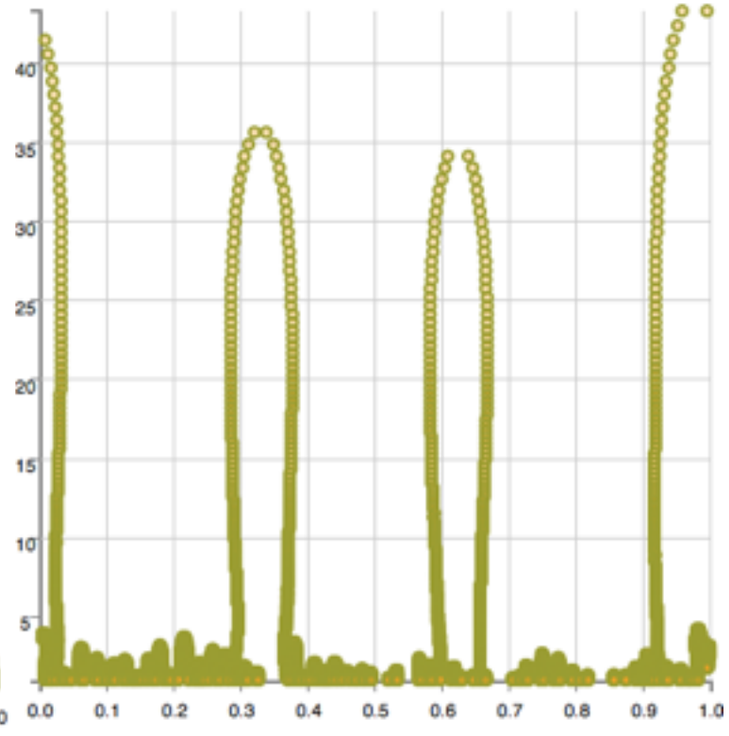
```
rotationInRadians=3.9406004
rotationInDegrees=225.77977139269535
scale=1.0218972
translationX=33.0
translationY=72.0
```

coordinate transformation, after matching contours







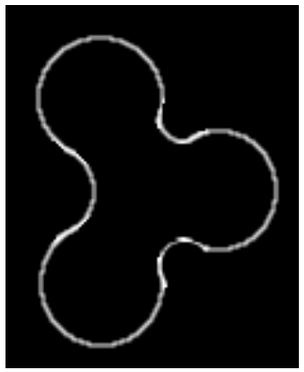


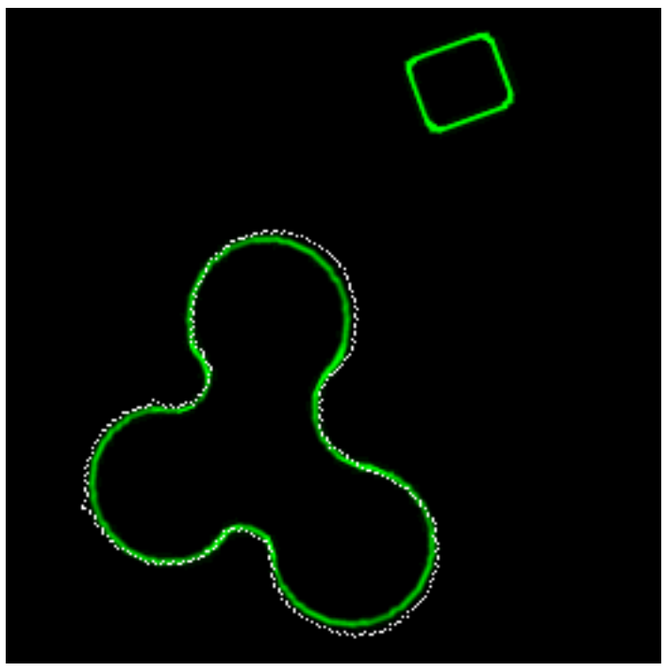
reversed to have CCW ordering

scale should be 1.3
rotation should be 360 - 160

```
Contour matcher solution shift=-0.648127555847168
Contour matcher solution cost=3.0
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (43.336529, 0.025109) (119, 158) (126, 171)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (35.660648, 0.672489) (75, 156) (80, 148)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (34.148750, 0.377729) (104, 212) (91, 205)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=29 offsetImgY2=6
rotationInRadians=3.4556763
rotationInDegrees=197.995668339729
scale=1.3542565
translationX=17.566353
translationY=81.42969
```

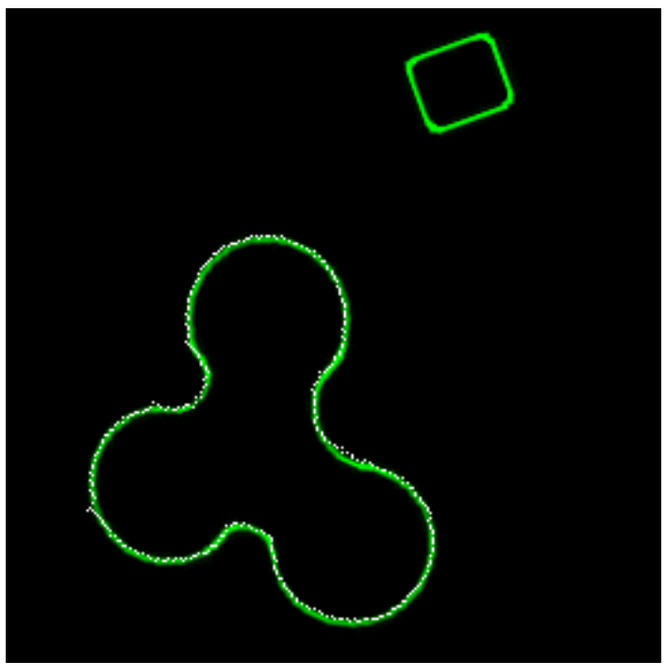
apply coordinate transformation





```
rotationInRadians=3.4556763
rotationInDegrees=197.995668339729
scale=1.3542565
translationX=17.566353
translationY=81.42969
```

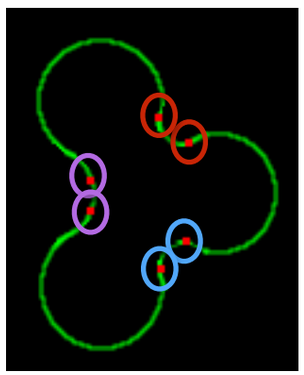
scale should be 1
rotation should be 360 - 160

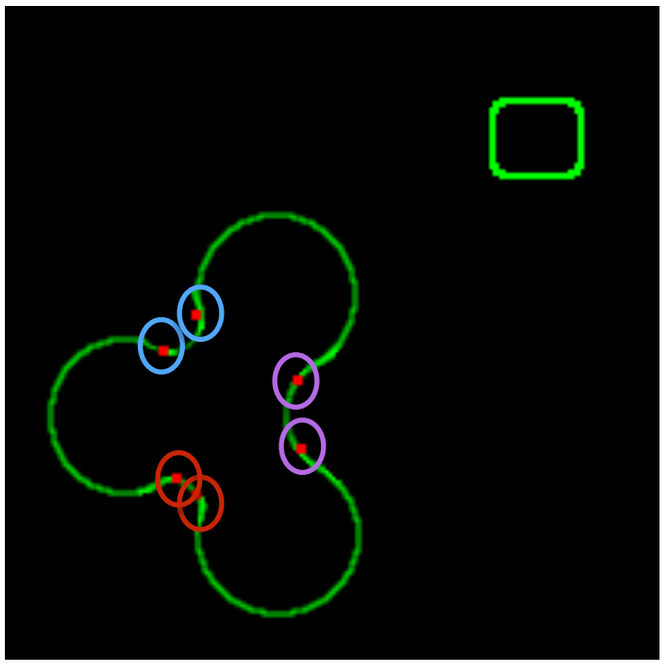


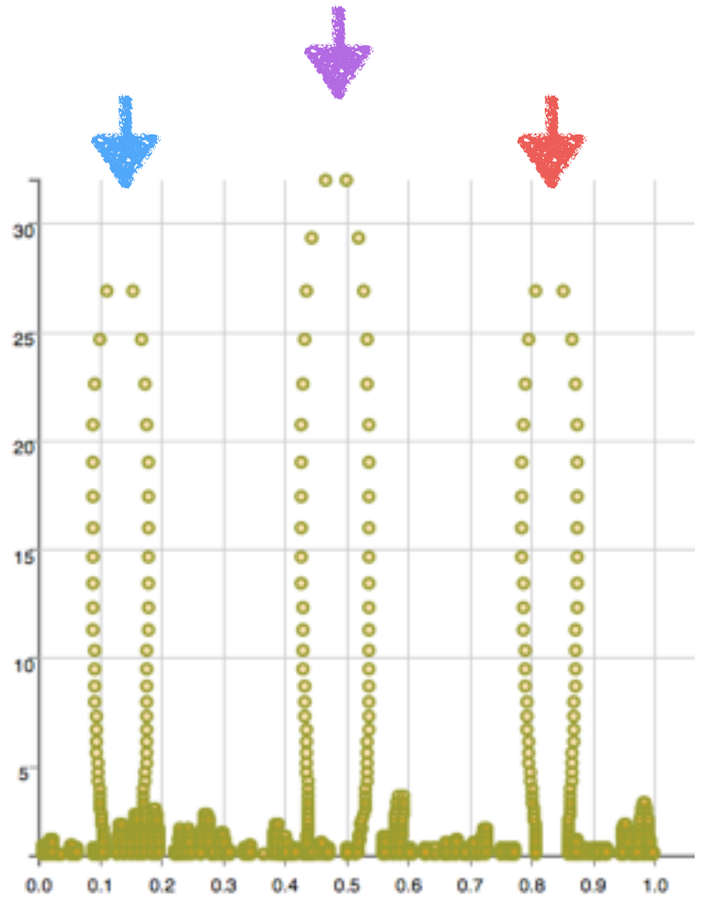
After Refinement

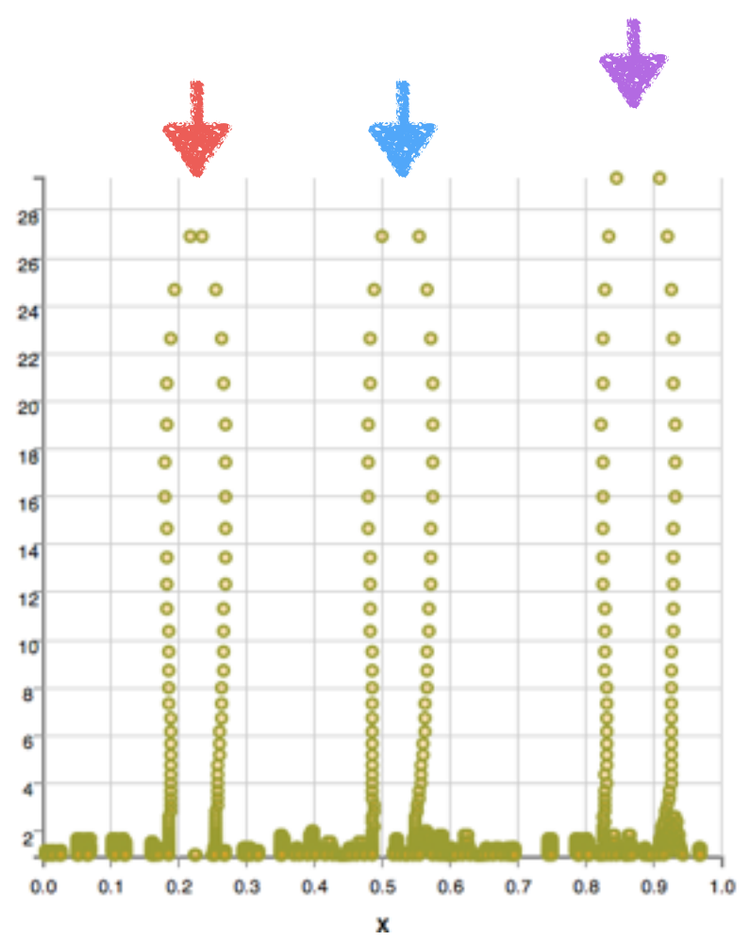
```
rotationInRadians=3.4993095
rotationInDegrees=200.49566784011543
scale=1.3042566
translationX=20.0
translationY=84.0
```

coordinate transformation, after matching contours







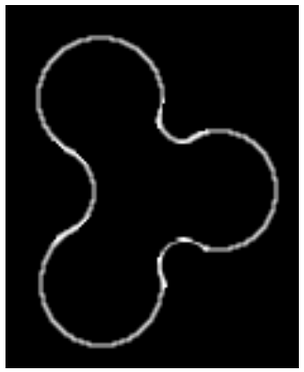


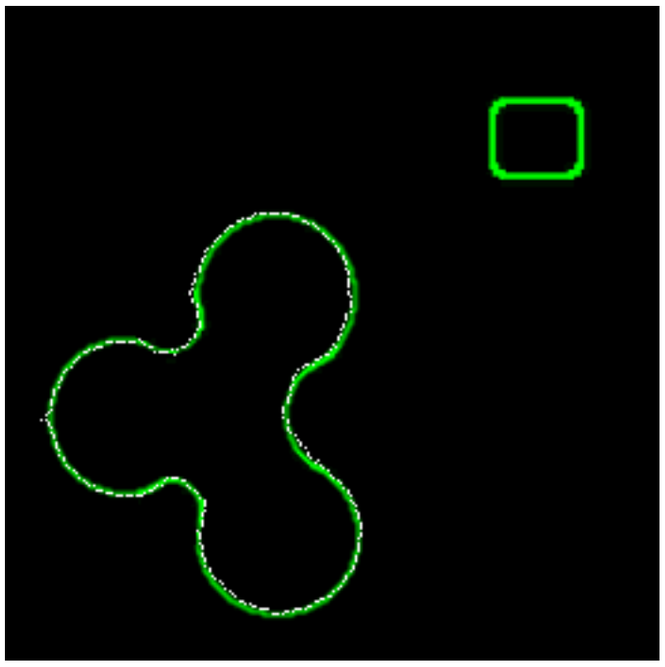
reversed to have CCW ordering

scale should be 1.3
rotation should be 180

```
Contour matcher solution scale=1.2968404293060303
Contour matcher solution shift=0.22837476432323456
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (41.499199, 0.873068) (111, 158) (112, 167)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (35.660648, 0.522075) (67, 137) (77, 127)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (34.896511, 0.222958) (75, 194) (69, 188)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=14 offsetImgY2=33
rotationInRadians=3.1657186
rotationInDegrees=181.38231235356184
scale=1.2968404
translationX=5.891382
translationY=70.504585
```

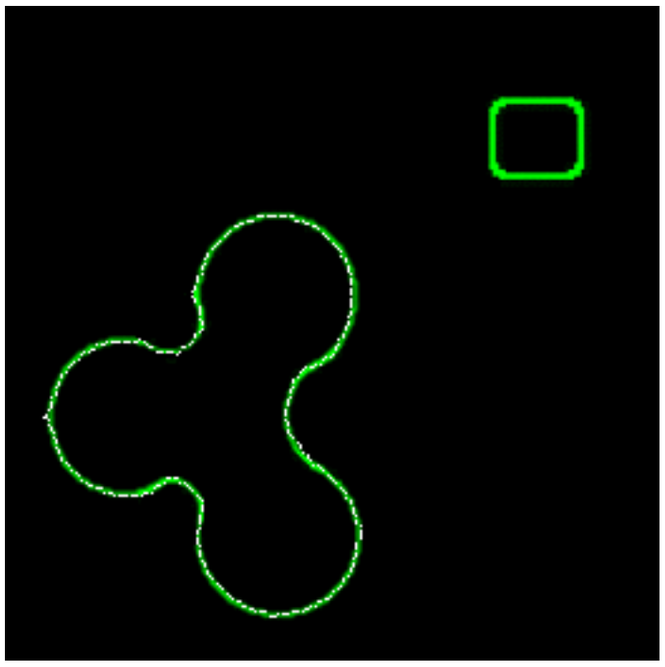
apply coordinate transformation





```
offsetImgX1=10 offsetImgY1=10
offsetImgX2=14 offsetImgY2=33
rotationInRadians=3.1657186
rotationInDegrees=181.38231235356184
scale=1.2968404
translationX=5.891382
translationY=70.504585
```

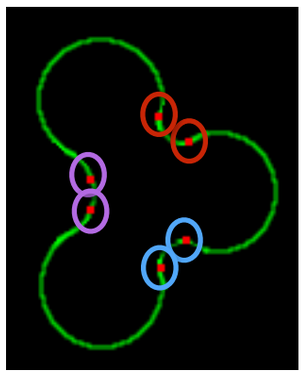
scale should be 1.3
rotation should be 180

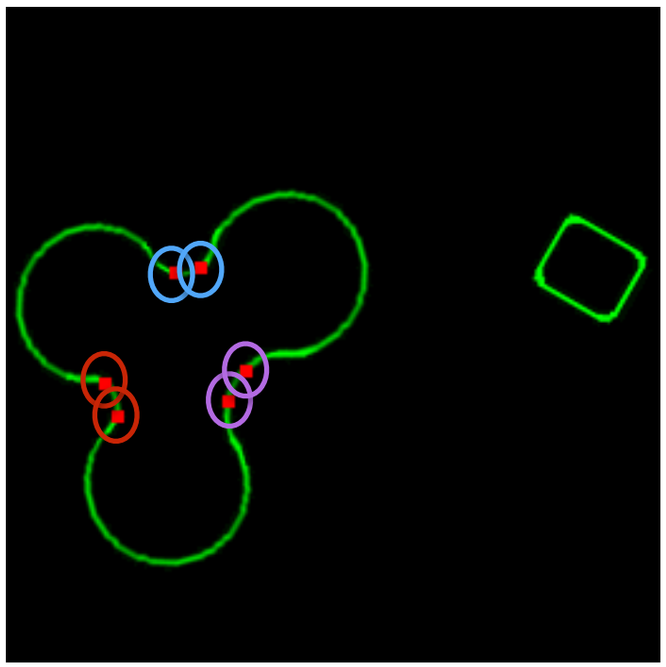


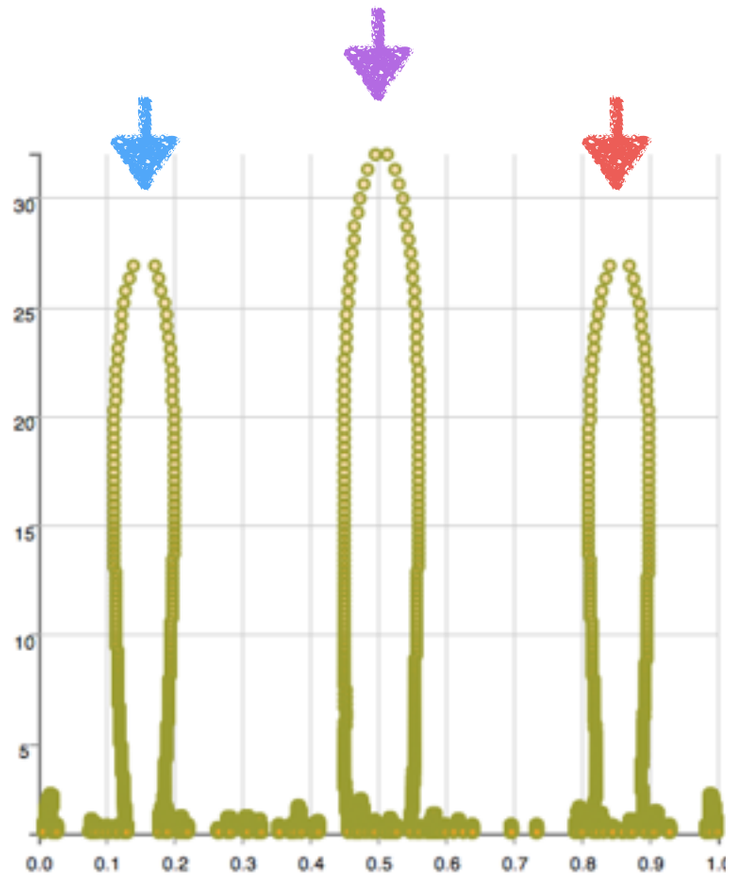
After Refinement

```
rotationInRadians=3.1482654
rotationInDegrees=180.38231801755862
scale=1.2968404
translationX=6.0
translationY=71.0
```

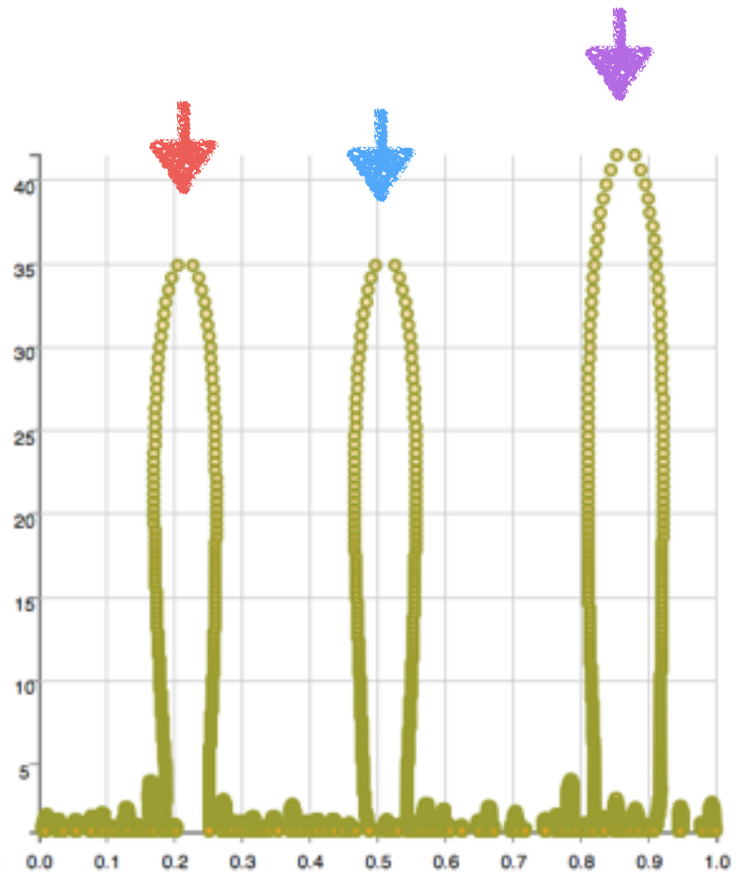
coordinate transformation, after matching contours







reversed to have CCW ordering

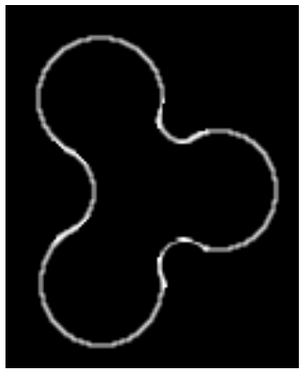


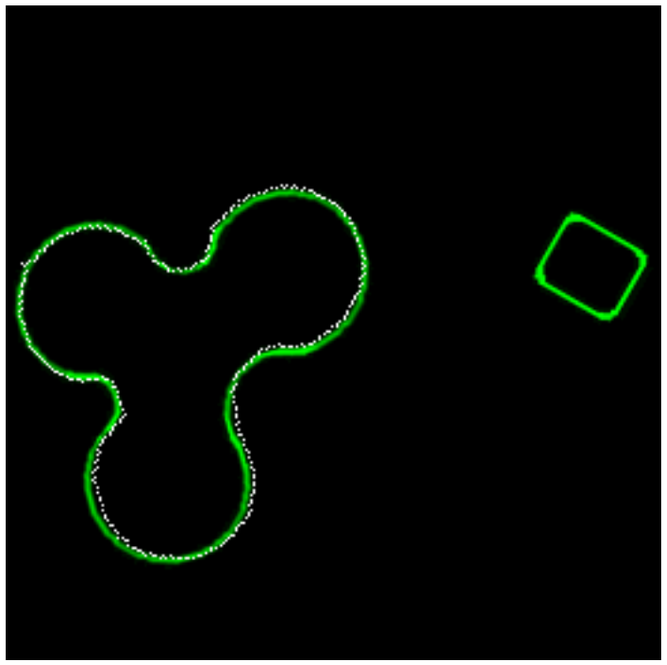
reversed to have CCW ordering

scale should be 1.3
rotation should be 360 - 210

```
Contour matcher solution scale=1.2968404293060303
Contour matcher solution shift=-0.5097379088401794
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (41.499199, 0.134956) (95, 144) (88, 156)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (34.896511, 0.785398) (67, 105) (77, 103)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (34.896511, 0.490044) (44, 162) (39, 149)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=1 offsetImgY2=70
rotationInRadians=2.6591716
rotationInDegrees=152.3593086078036
scale=1.2968404
translationX=-10.876109
translationY=41.450577
```

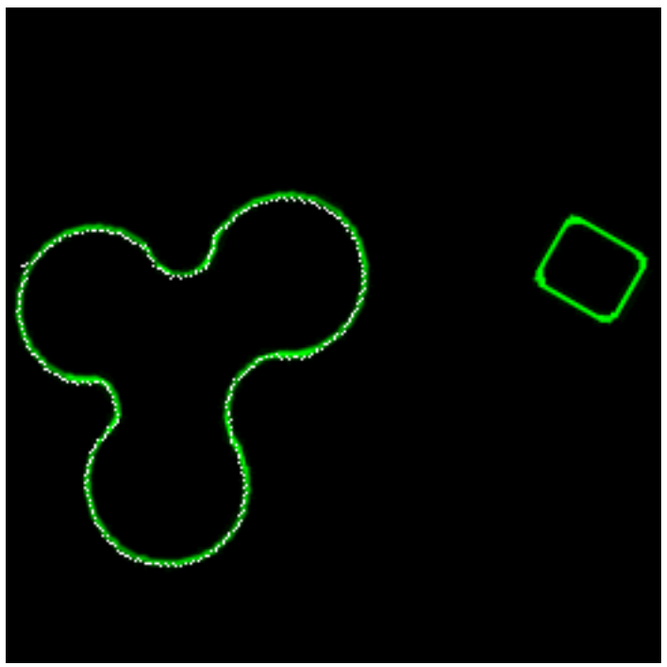
apply coordinate transformation





```
offsetImgX1=10 offsetImgY1=10
offsetImgX2=1 offsetImgY2=70
rotationInRadians=2.6591716
rotationInDegrees=152.3593086078036
scale=1.2968404
translationX=-10.876109
translationY=41.450577
```

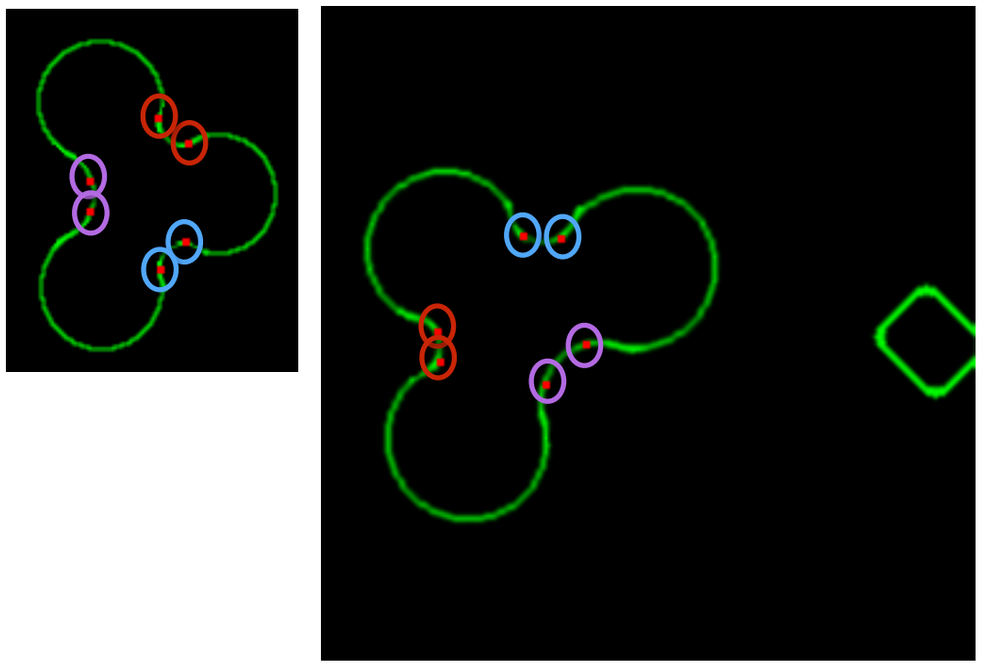
scale should be 1.3
rotation should be 360 - 210

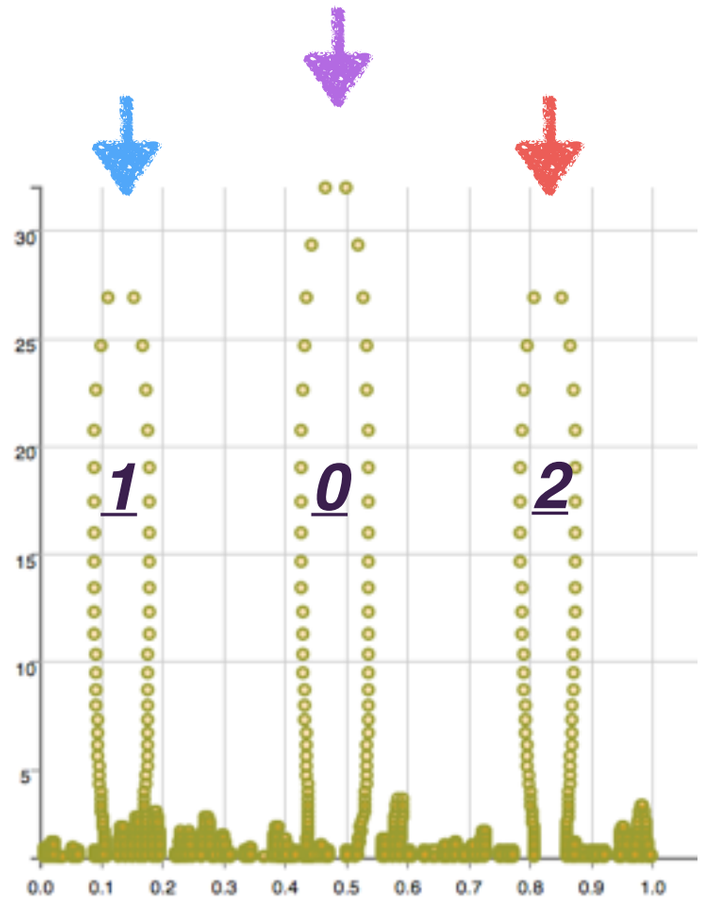


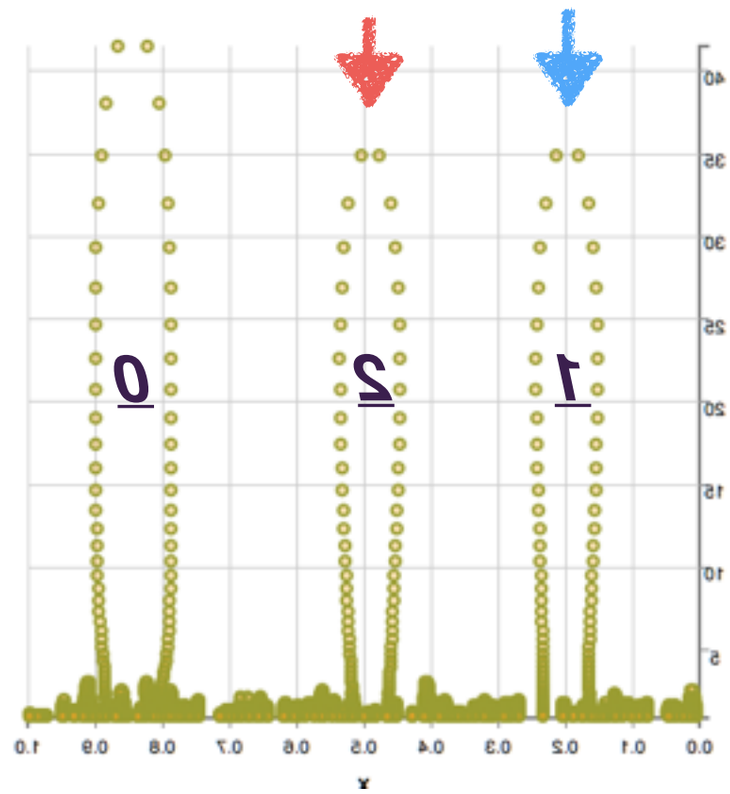
After Refinement

```
rotationInRadians=2.6155384
rotationInDegrees=149.85930910741718
scale=1.2968404
translationX=-12.0
translationY=43.0
```

coordinate transformation, after matching contours





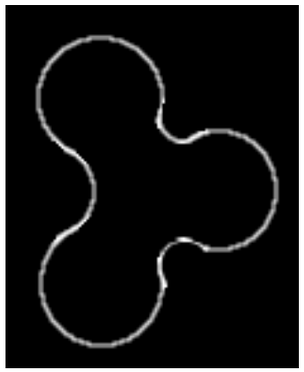


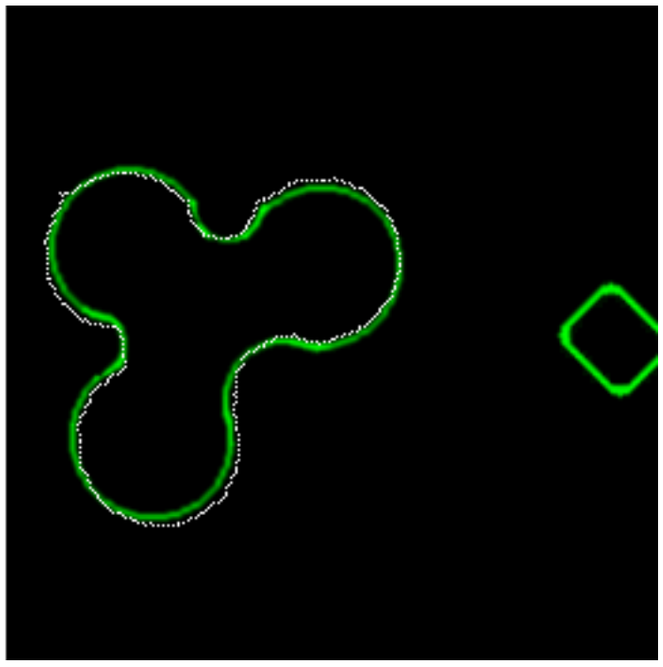
reversed to have CCW ordering

scale should be 1.3
rotation should be 360 - 225

```
Contour matcher solution scale=1.325237512588501
Contour matcher solution shift=-0.5053889751434326
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (42.407913, 0.153422) (98, 136) (91, 143)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (34.896511, 0.802428) (81, 91) (92, 92)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (34.896511, 0.508830) (46, 140) (45, 129)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=14 offsetImgY2=61
rotationInRadians=2.4194849
rotationInDegrees=138.6262707153875
scale=1.3252375
translationX=-5.8268623
translationY=25.13414
```

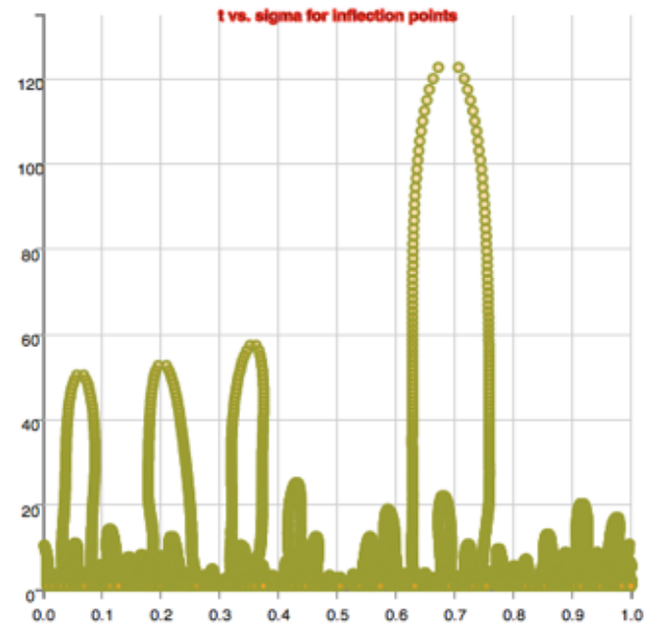
apply coordinate transformation





```
offsetImgX1=10 offsetImgY1=10
offsetImgX2=14 offsetImgY2=61
rotationInRadians=2.4194849
rotationInDegrees=138.6262707153875
scale=1.3252375
translationX=-5.8268623
translationY=25.13414
```

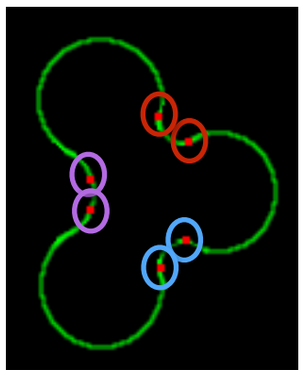
scale should be 1.3
rotation should be 360 - 225

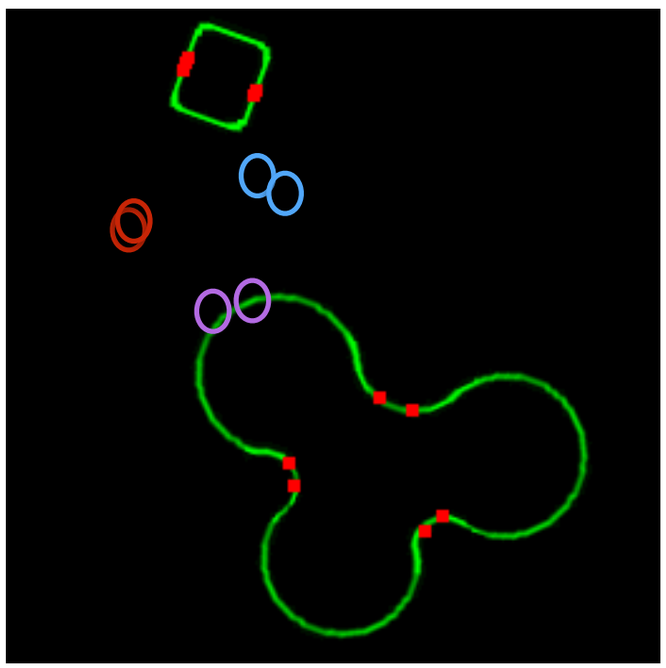


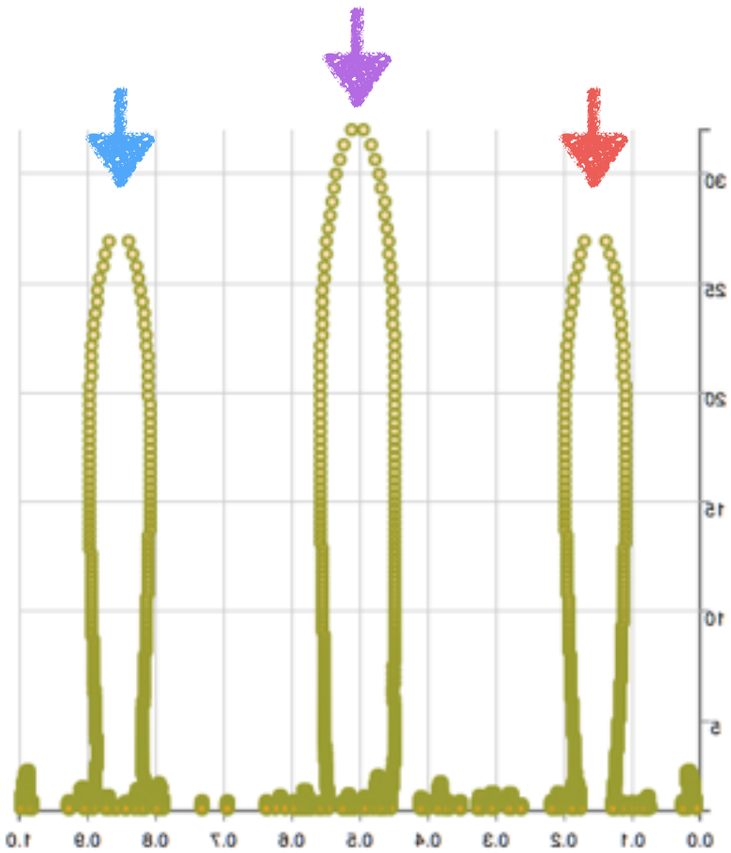
After Refinement

```
rotationInRadians=2.3322184
rotationInDegrees=133.6262717146147
scale=1.2752376
translationX=-3.0
translationY=27.0
```

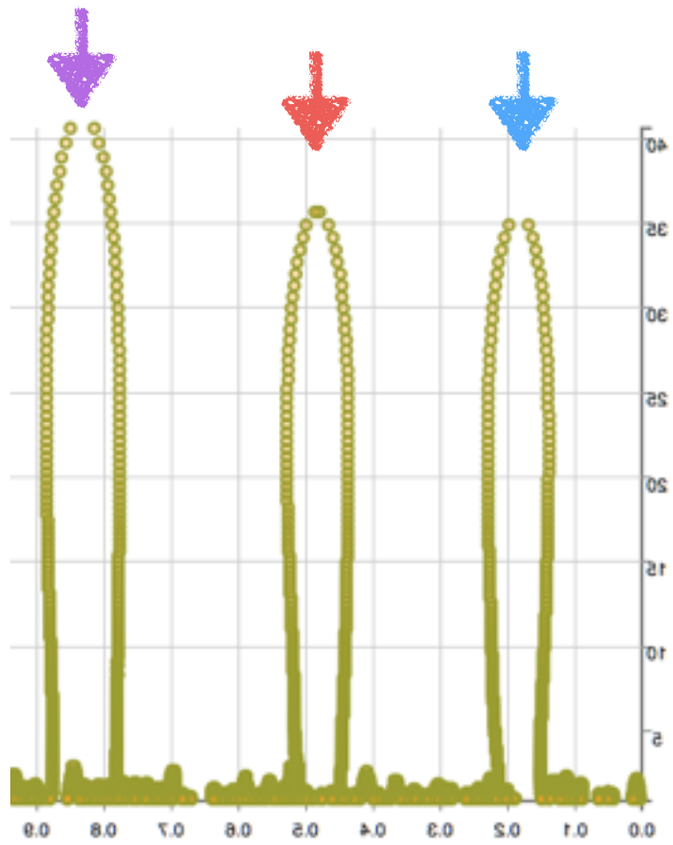
coordinate transformation, after matching contours







reversed to have CCW ordering

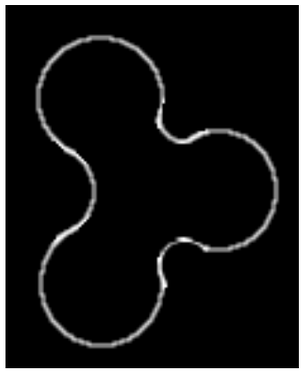


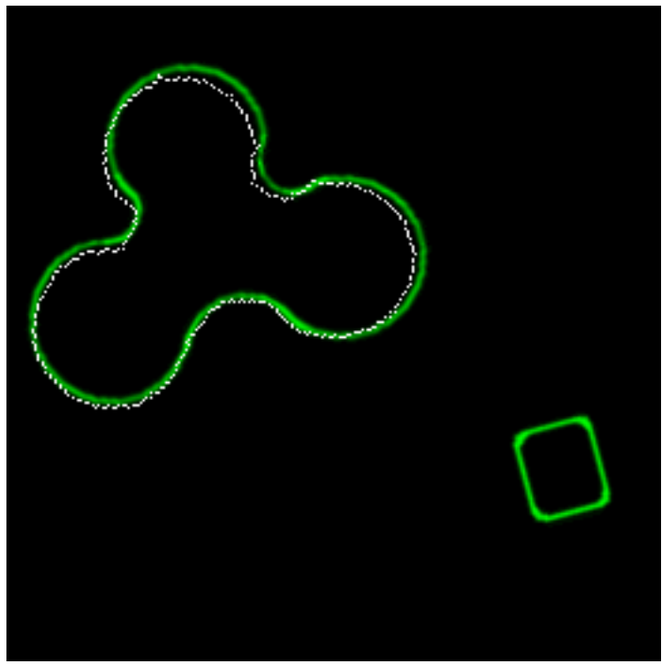
reversed to have CCW ordering

scale should be 1.3
rotation should be 360 - 255

```
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (40.609955, 0.168514) (98, 115) (82, 118)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (34.896511, 0.817073) (101, 65) (113, 73)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (35.660648, 0.517738) (49, 89) (51, 86)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=5 offsetImgY2=20
rotationInRadians=1.8595577
rotationInDegrees=106.54481071213507
scale=1.2690517
translationX=3.6613295
translationY=-1.6037707
```

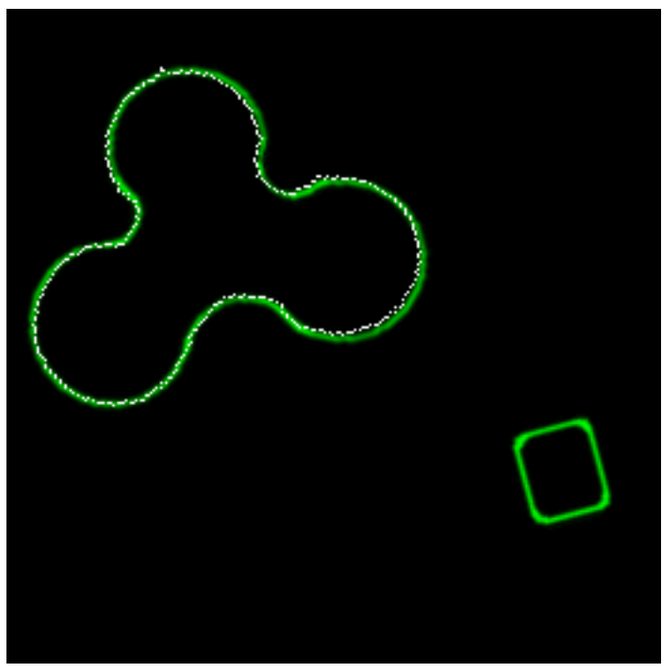
apply coordinate transformation





```
rotationInRadians=1.8595577
rotationInDegrees=106.54481071213507
scale=1.2690517
translationX=3.6613295
translationY=-1.6037707
```

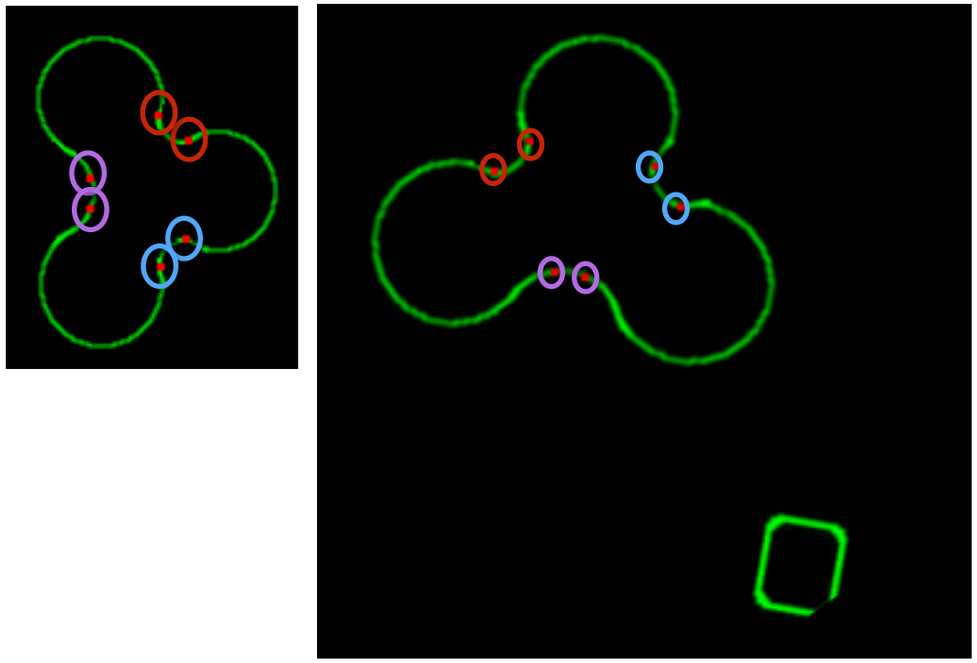
scale should be 1.3
rotation should be 360 - 255

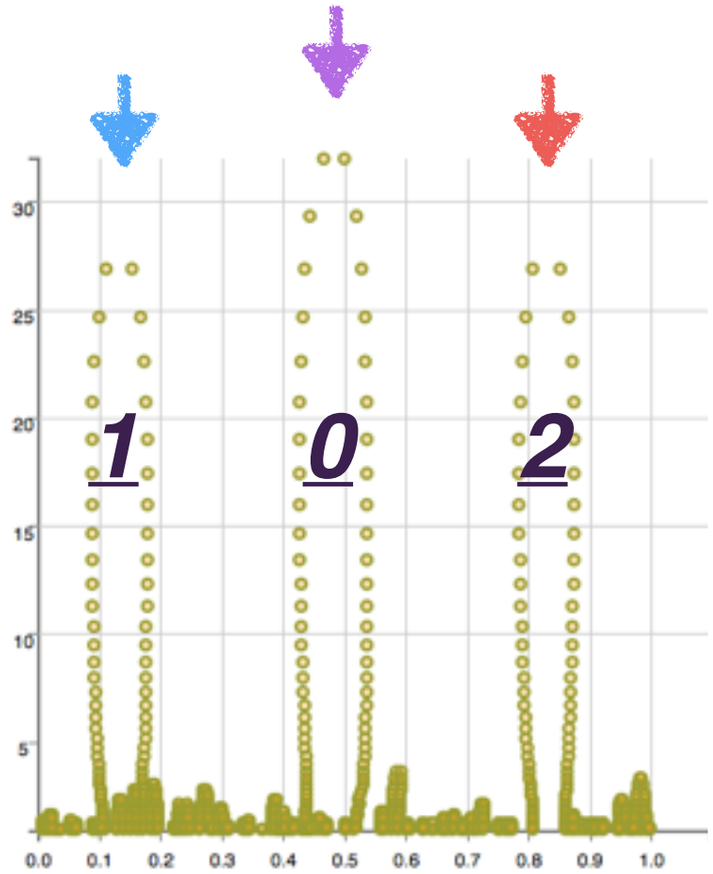


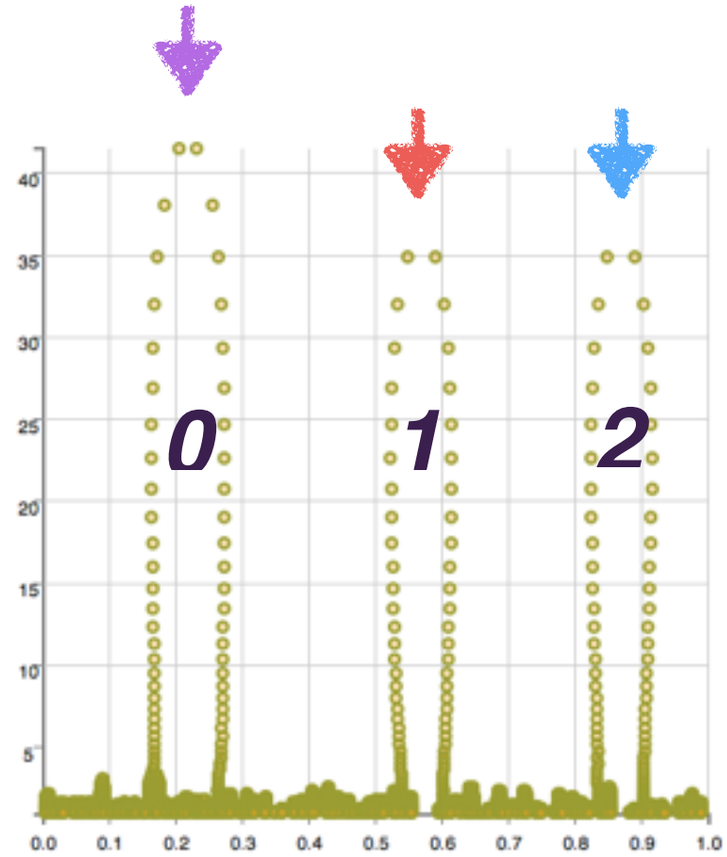
After Refinement

```
rotationInRadians=1.8558925
rotationInDegrees=106.33480971591386
scale=1.2845517
translationX=4.0
translationY=-6.0
```

coordinate transformation, after matching contours



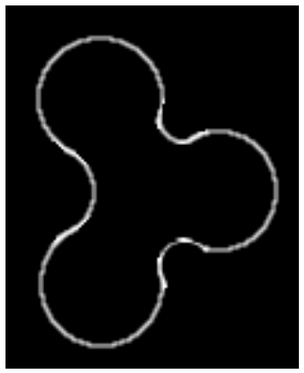


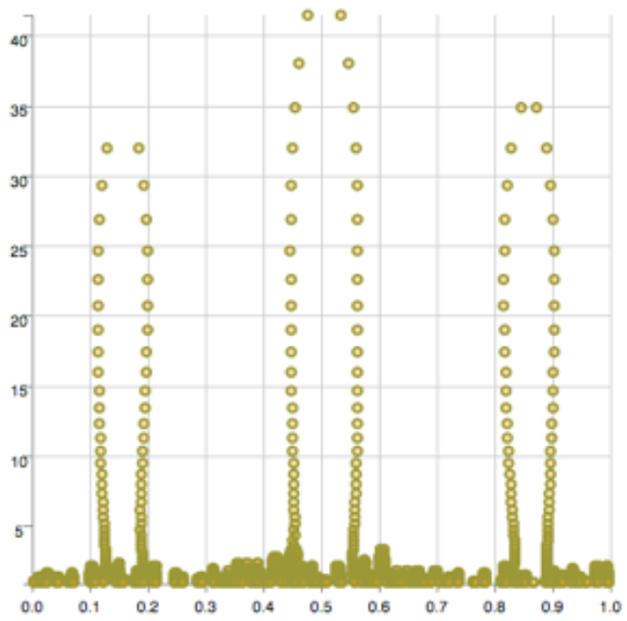


scale should be 1.3
rotation should be 360 - 280

```
Contour matcher solution scale=1.2968404293060303
Contour matcher solution shift=-0.4310373365879059
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (41.499199, 0.213656) (106, 108) (96, 106)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (35.660648, 0.865639) (134, 70) (137, 76)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (35.660648, 0.566079) (74, 67) (82, 60)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=19 offsetImgY2=9
rotationInRadians=1.4486057
rotationInDegrees=82.9989903033234
scale=1.2968404
translationX=28.18638
translationY=-16.518988
```

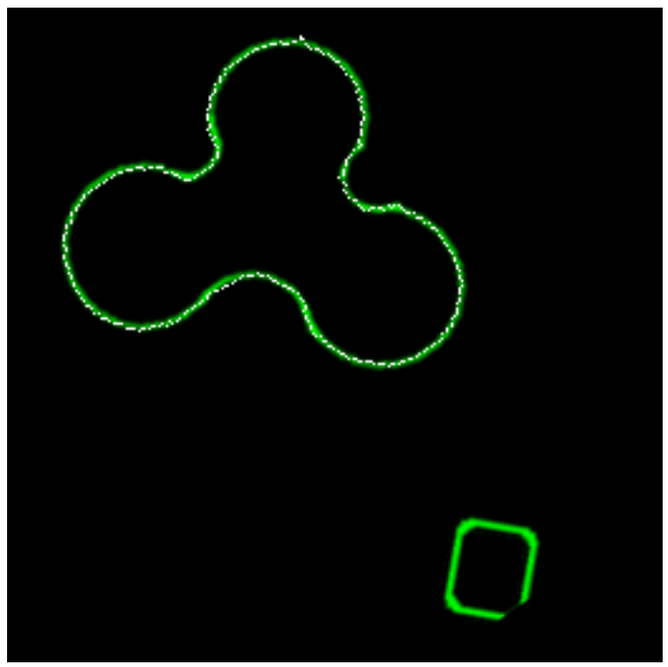
apply coordinate transformation






```
offsetImgX1=10 offsetImgY1=10
offsetImgX2=19 offsetImgY2=9
rotationInRadians=1.4486057
rotationInDegrees=82.9989903033234
scale=1.2968404
translationX=28.18638
translationY=-16.518988
```

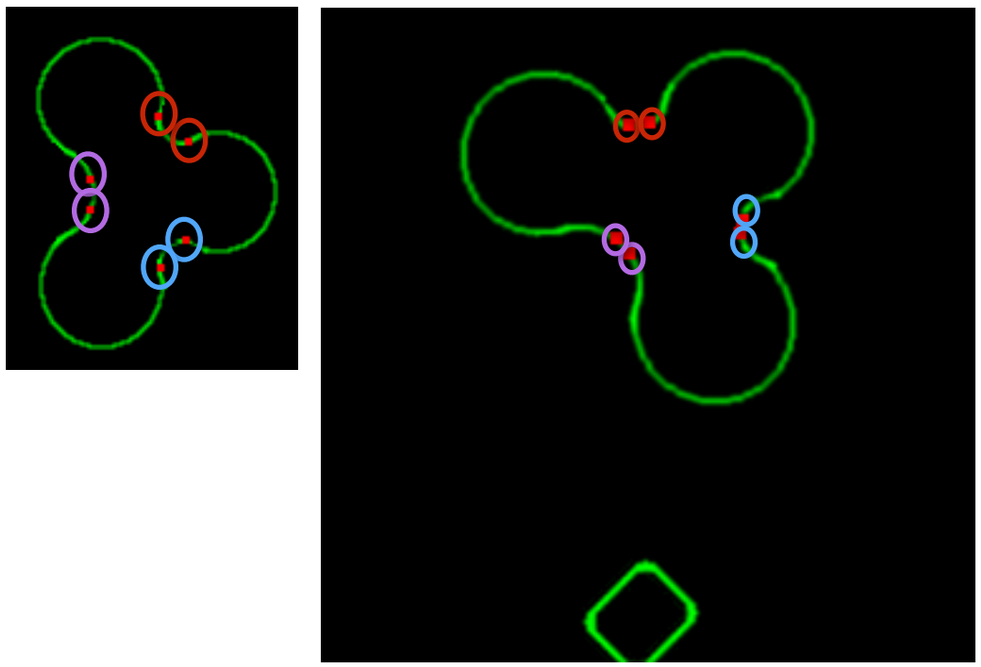
scale should be 1.3
rotation should be 360 - 280

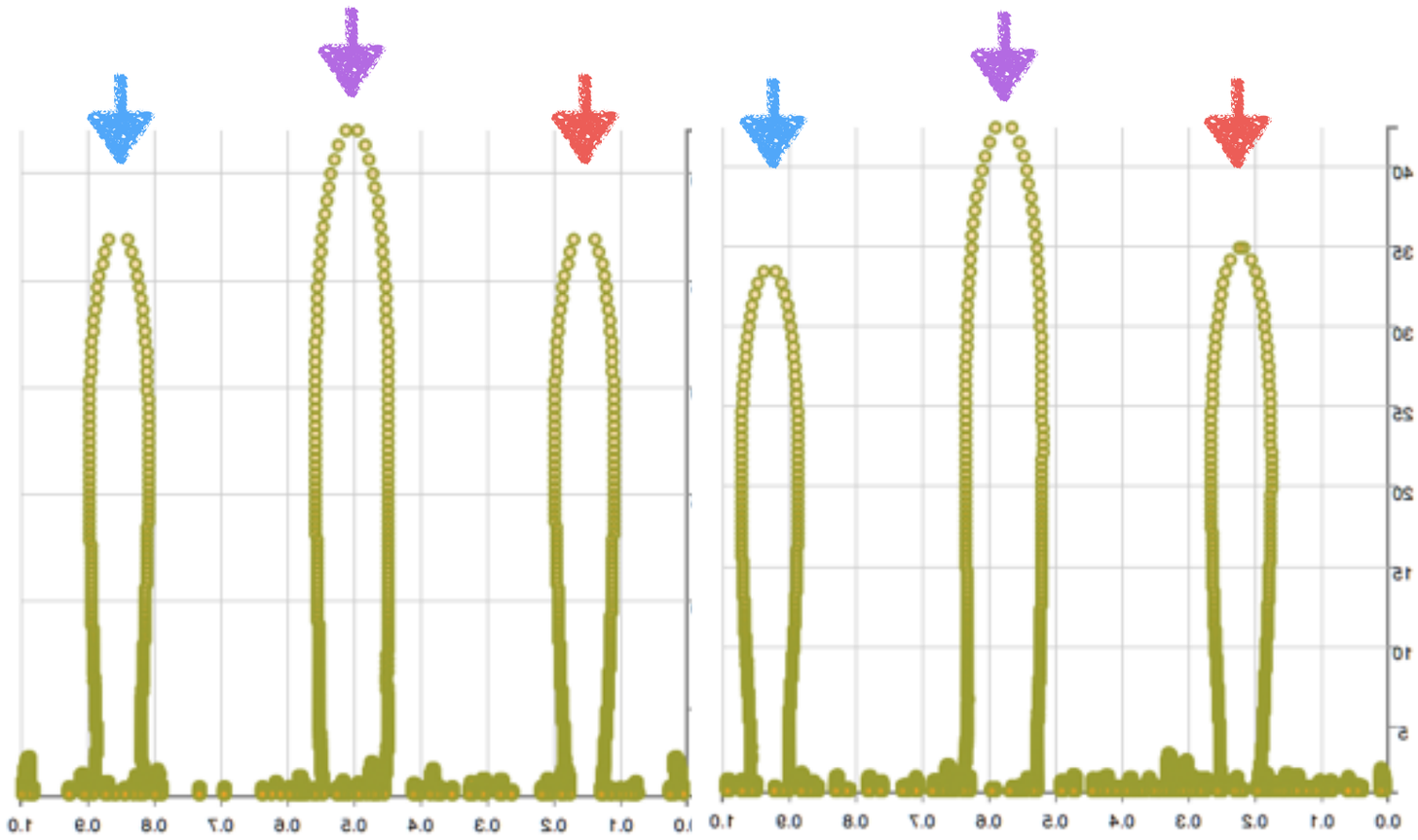


After Refinement

```
rotationInRadians=1.4049724
rotationInDegrees=80.49899080293699
scale=1.2968404
translationX=29.0
translationY=-17.0
```

coordinate transformation, after matching contours





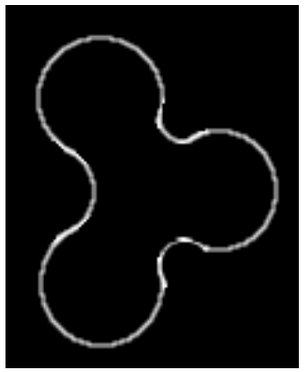
reversed to have CCW ordering

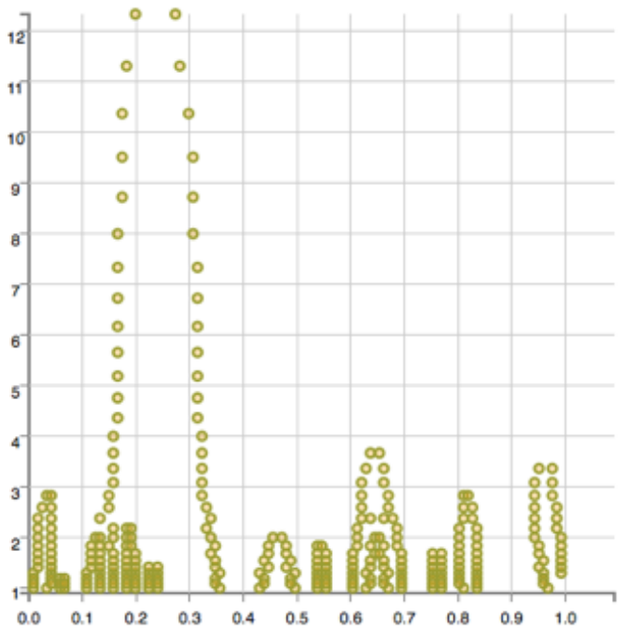
reversed to have CCW ordering

scale should be 1.3
rotation should be 360 - 315

```
Contour matcher solution scale=1.325237512588501
Contour matcher solution shift=-0.2311936765909195
Contour matcher solution cost=89.0
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (42.407913, 0.427617) (125, 102) (112, 89)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (33.417011, 0.074610) (169, 80) (168, 93)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (34.148750, 0.783964) (120, 45) (132, 45)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=52 offsetImgY2=14
rotationInRadians=0.7365556
rotationInDegrees=42.2015259002177
scale=1.3252375
translationX=60.744064
translationY=-25.366236
```

apply coordinate transformation






```
offsetImgX1=10 offsetImgY1=10
offsetImgX2=52 offsetImgY2=14
rotationInRadians=0.7365556
rotationInDegrees=42.2015259002177
scale=1.3252375
translationX=60.744064
translationY=-25.366236
```

scale should be 1.3
rotation should be 360 - 315

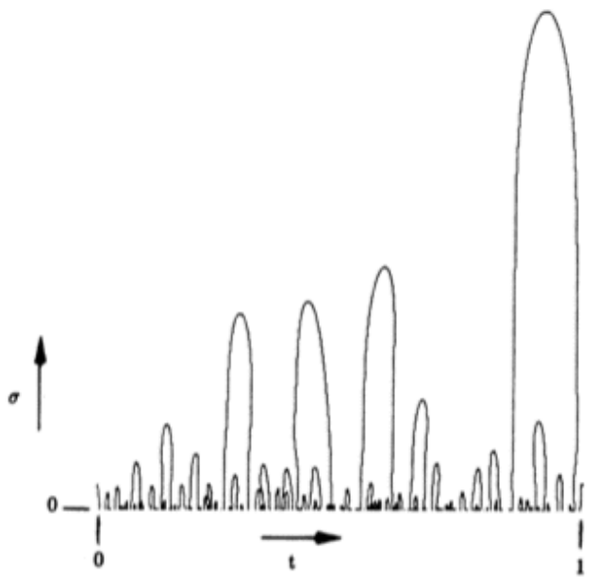
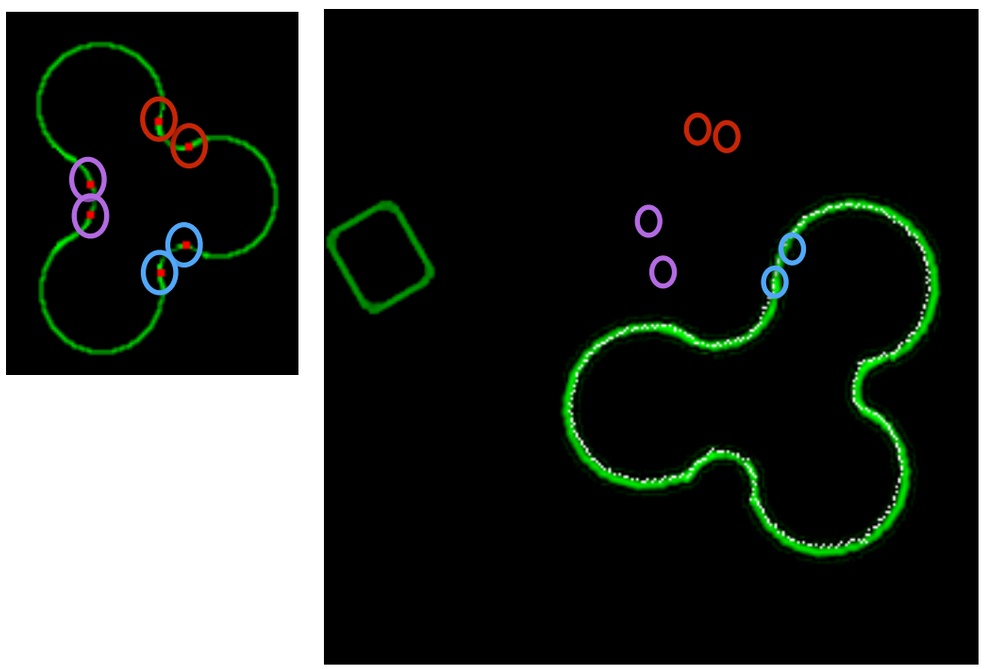


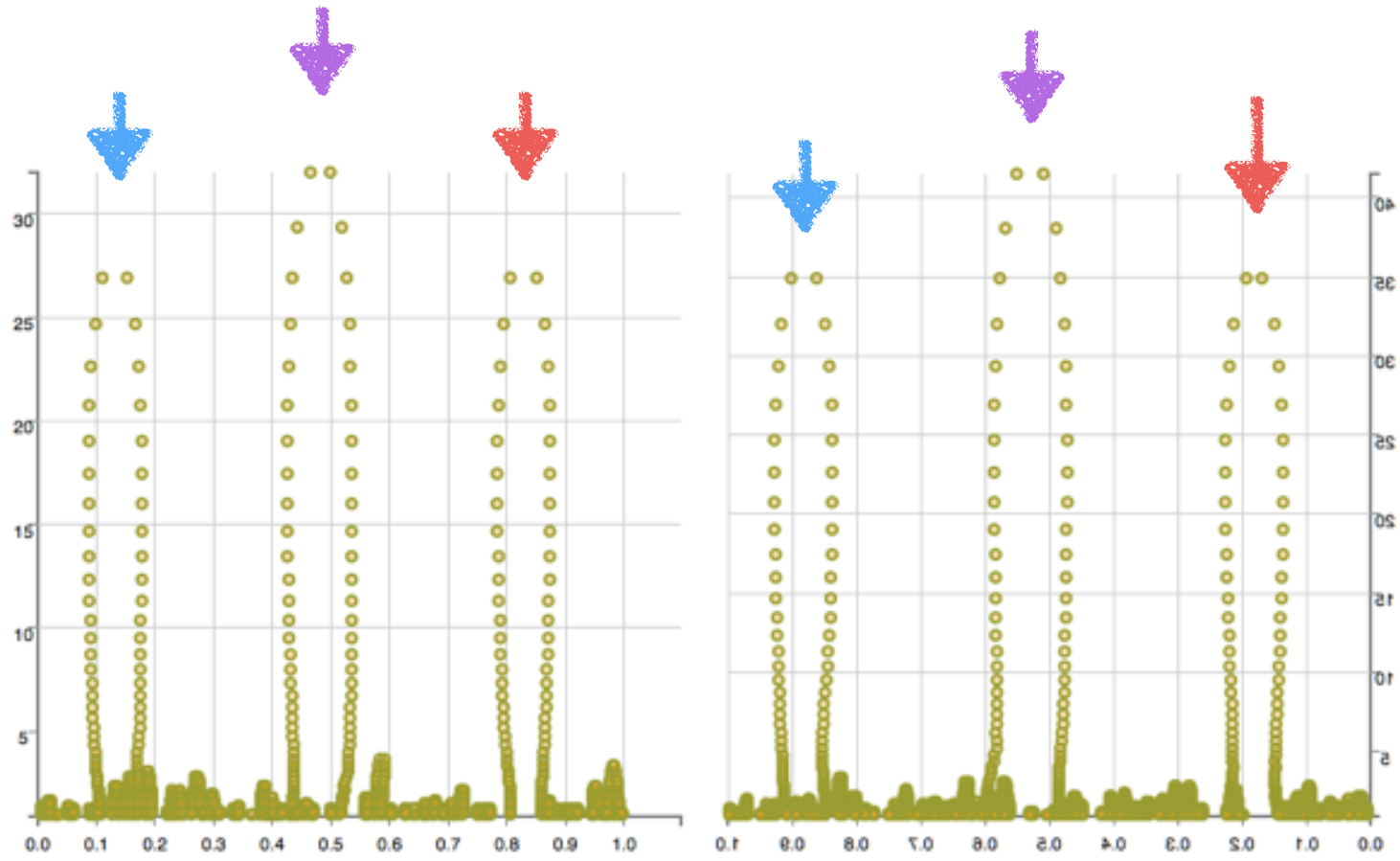
Fig. 3. Generalized scale space image of Africa.

After Refinement

```
rotationInRadians=0.7801888
rotationInDegrees=44.70152540060411
scale=1.2752376
translationX=65.0
translationY=-21.0
```

coordinate transformation, after matching contours

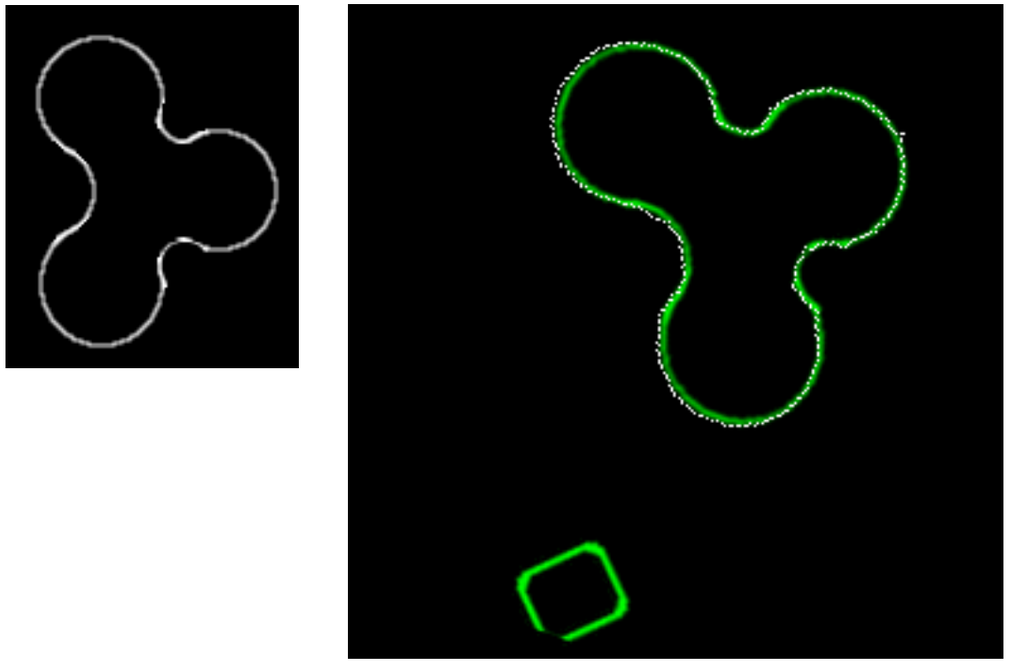




scale should be 1.3
rotation should be 360 - 335

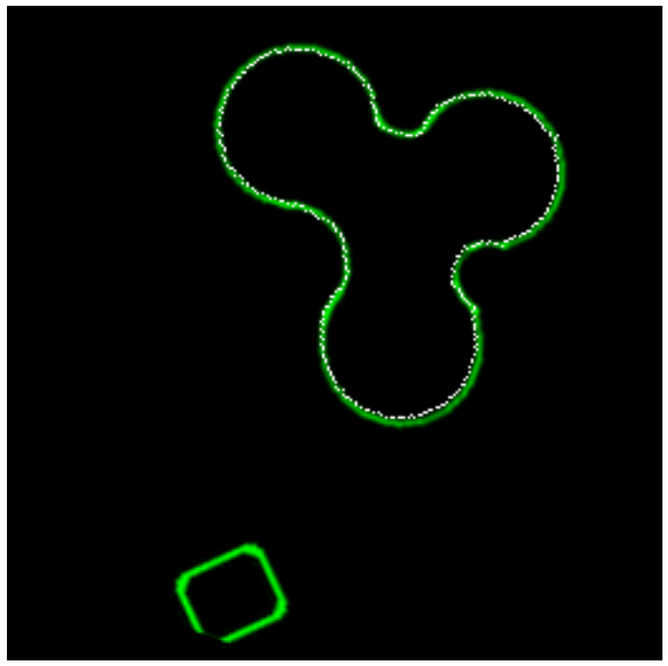
```
Contour matcher solution scale=1.325237512588501
Contour matcher solution shift=-0.19089026749134064
CONTOUR PEAK1: (32.000237, 0.497126) (34, 78) (35, 72) CONTOUR PEAK2: (42.407913, 0.467920) (134, 99) (131, 90)
CONTOUR PEAK1: (26.908875, 0.146552) (70, 93) (61, 99) CONTOUR PEAK2: (34.896511, 0.115044) (182, 98) (178, 106)
CONTOUR PEAK1: (26.908875, 0.846264) (60, 45) (69, 54) CONTOUR PEAK2: (34.896511, 0.818584) (150, 48) (158, 51)
offsetImgX1=10 offsetImgY1=10
offsetImgX2=63 offsetImgY2=12
rotationInRadians=0.43239865
rotationInDegrees=24.77461754427387
scale=1.3252375
translationX=81.613556
translationY=-16.98058
```

apply coordinate transformation



```
offsetImgX1=10 offsetImgY1=10
offsetImgX2=63 offsetImgY2=12
rotationInRadians=0.43239865
rotationInDegrees=24.77461754427387
scale=1.3252375
translationX=81.613556
translationY=-16.98058
```

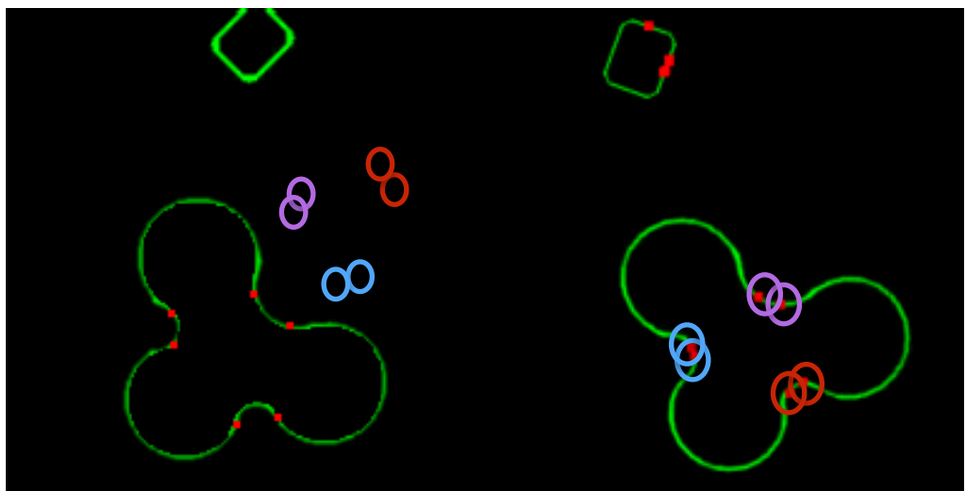
scale should be 1.3
rotation should be 360 - 335



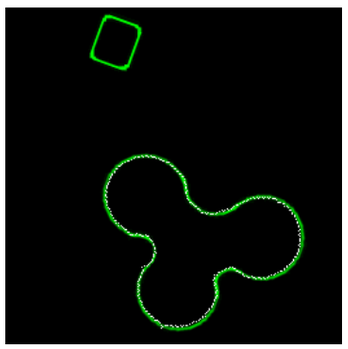
After Refinement

```
rotationInRadians=0.43239865
rotationInDegrees=24.77461754427387
scale=1.2752376
translationX=85.0
translationY=-14.0
```

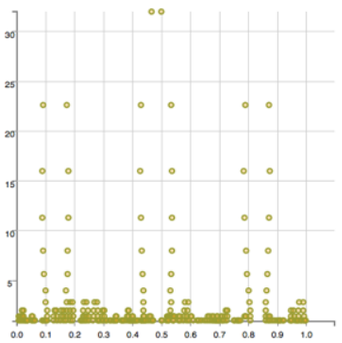
coordinate transformation, after matching contours

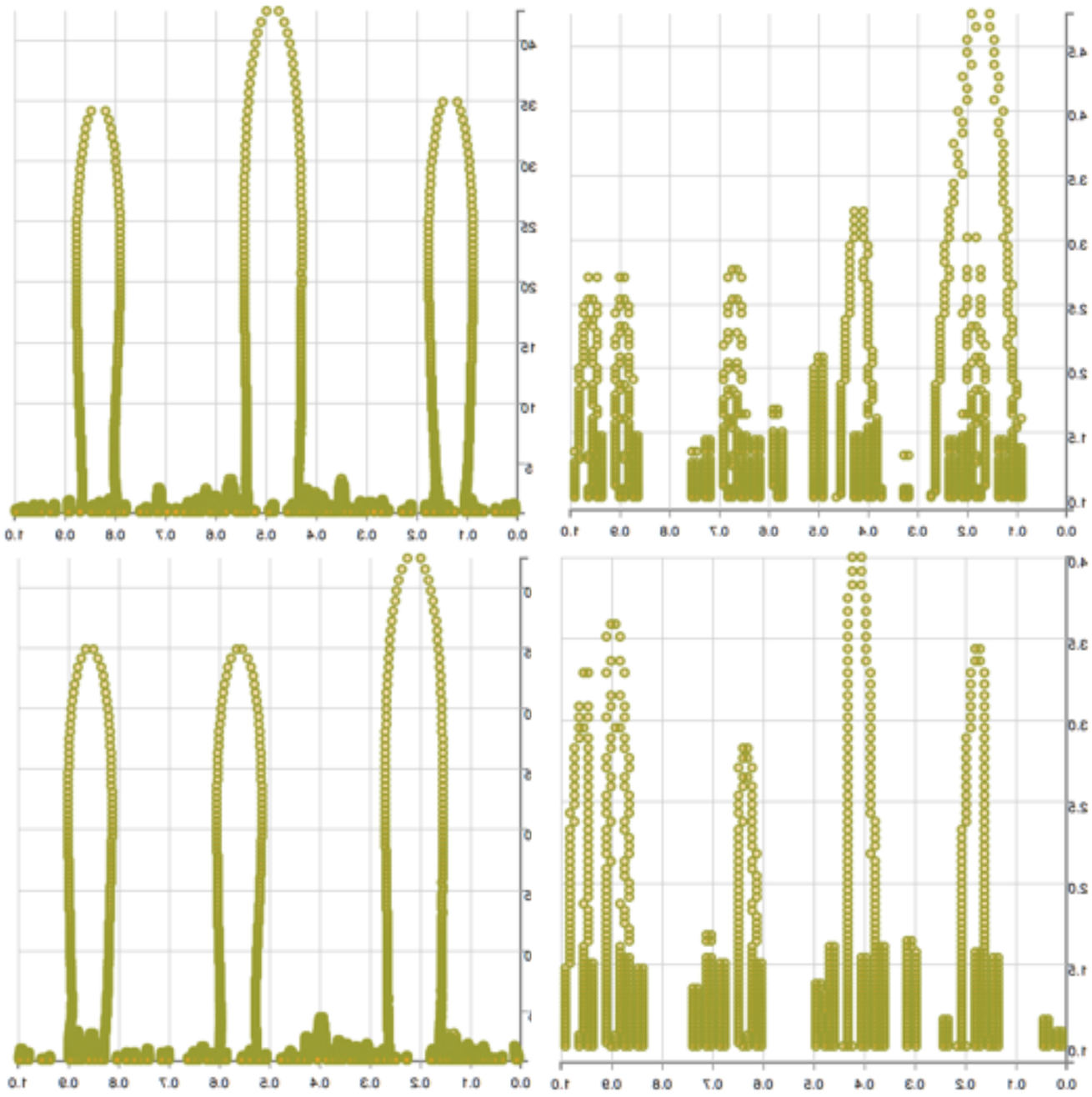


rotationInRadians=4.7316465
rotationInDegrees
=271.1033767625259
scale=1.0
translationX=0.5612701
translationY=0.34837633



rotationInRadians=4.7141933
rotationInDegrees
=270.10338242652267
scale=1.0
translationX=-0.19677997
translationY=0.0

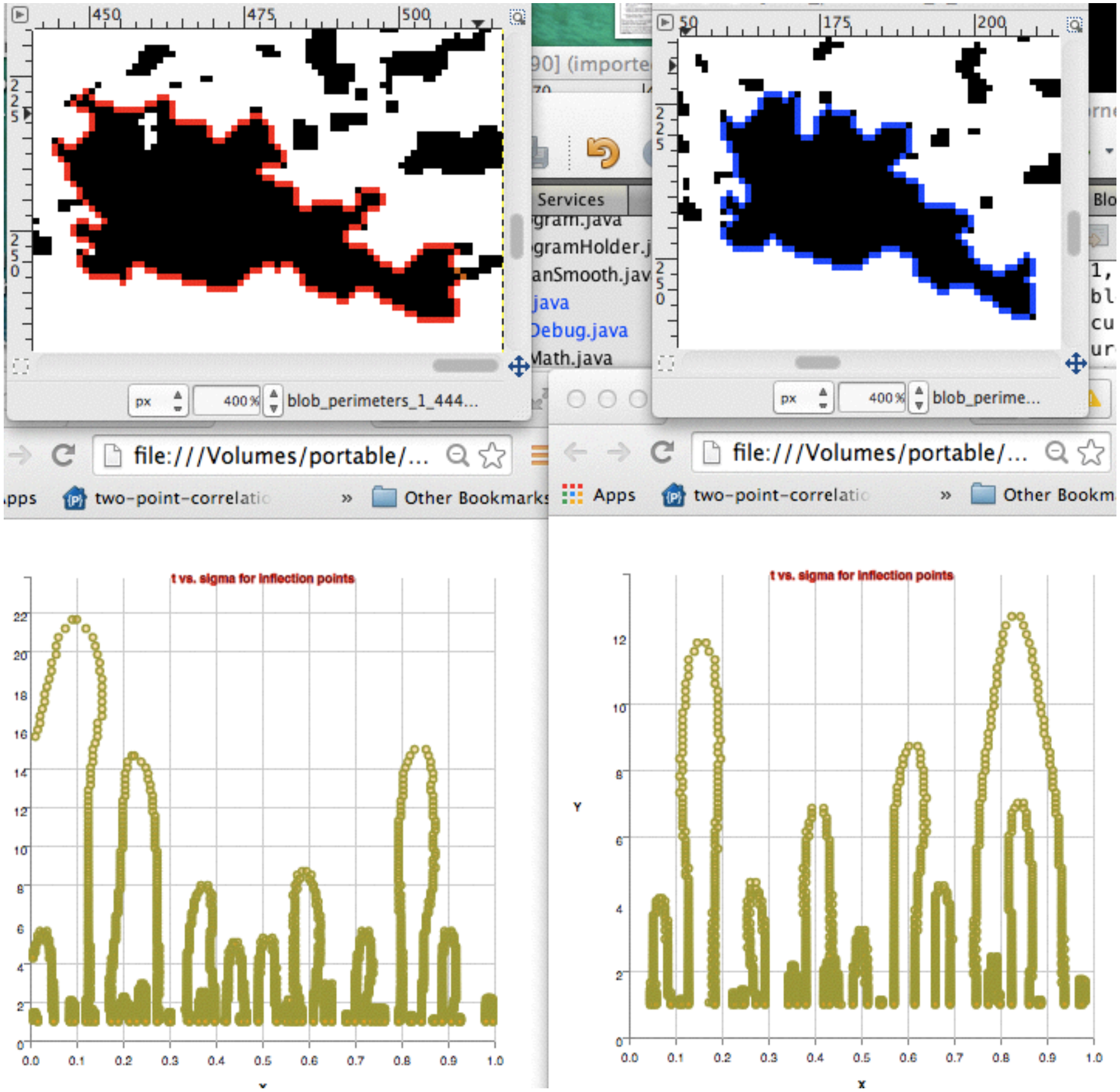




Contour matcher solution scale=1.0
Contour matcher solution shift=0.27483445405960083
Contour matcher solution cost=14.0
CONTOUR PEAK1: (42.407913, 0.513245) (156, 110) (159, 99) CONTOUR PEAK2: (42.407913, 0.788080) (149, 155) (161, 159)
CONTOUR PEAK1: (34.896511, 0.866446) (201, 83) (208, 96) CONTOUR PEAK2: (34.896511, 0.142384) (173, 201) (166, 207)
CONTOUR PEAK1: (4.756843, 0.845455) (33, 160) (33, 160) CONTOUR PEAK2: (3.589427, 0.103604) (99, 33) (98, 34)
CONTOUR PEAK1: (34.148750, 0.166667) (191, 144) (178, 149) CONTOUR PEAK2: (34.896511, 0.440397) (115, 187) (113, 182)
CONTOUR PEAK1: (3.220988, 0.581818) (9, 170) (10, 168) CONTOUR PEAK2: (3.437247, 0.828829) (90, 9) (90, 9)
CONTOUR PEAK1: (3.018335, 0.809091) (28, 158) (30, 159) CONTOUR PEAK2: (3.291518, 0.049550) (101, 27) (101, 28)

very important to have similar curves.

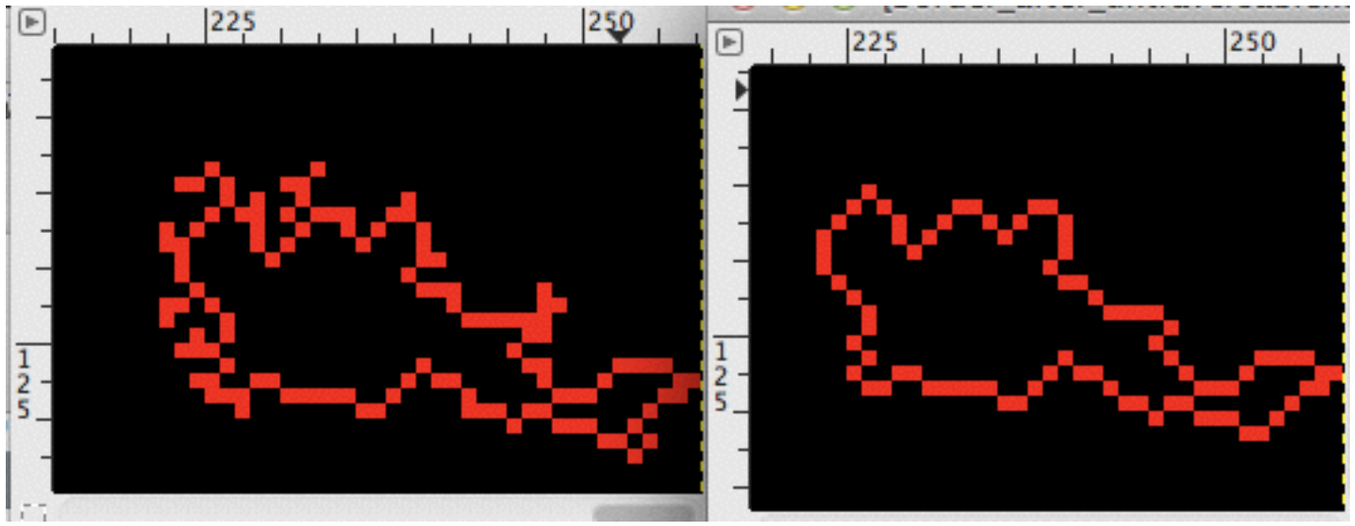
The same blob in different images with a couple of large bumps introduced from the image processing and then perimeter extraction instead of edges, shows a scale space map on left dominated by the difference. The contour matcher prefers the strongest peaks so does not give as good of a score for the true match (seen from 0.2 to 0.9)

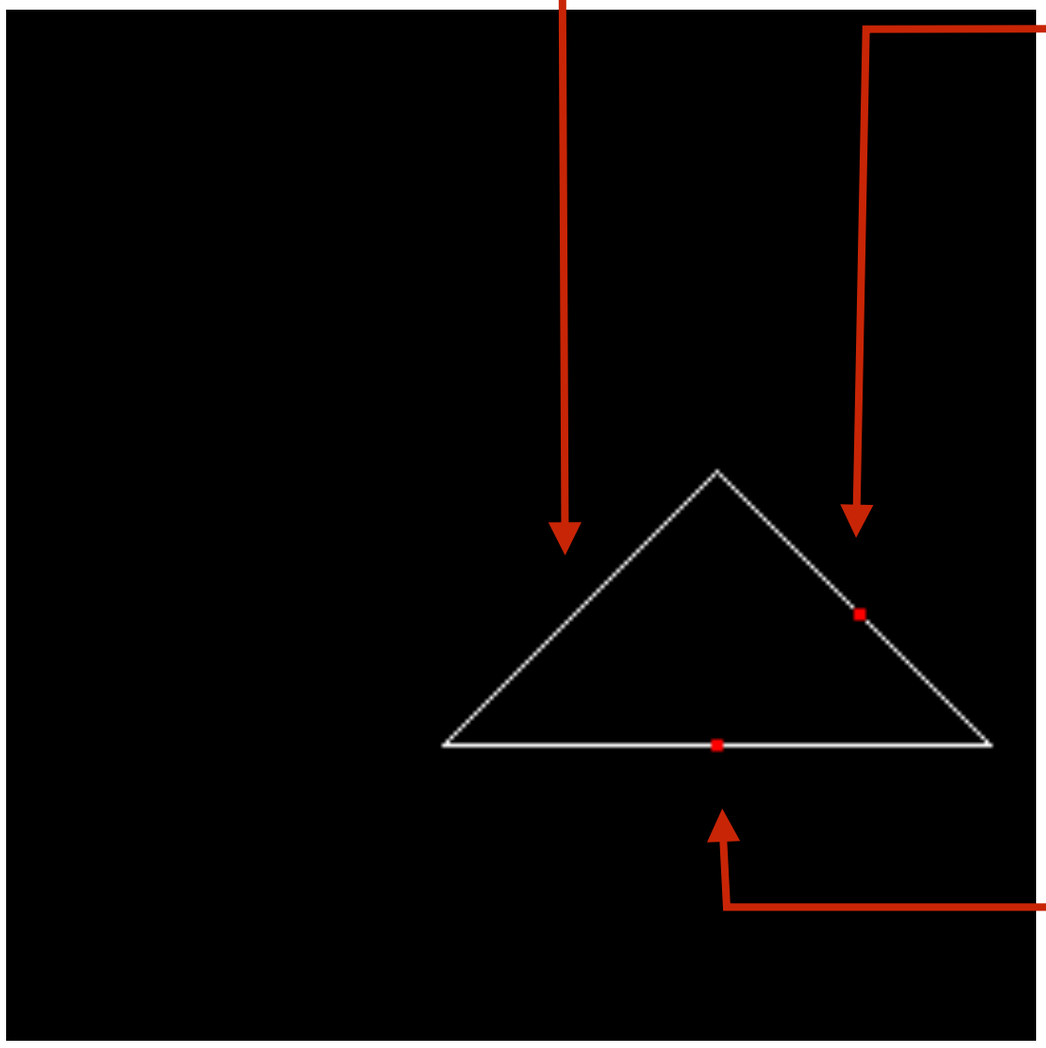


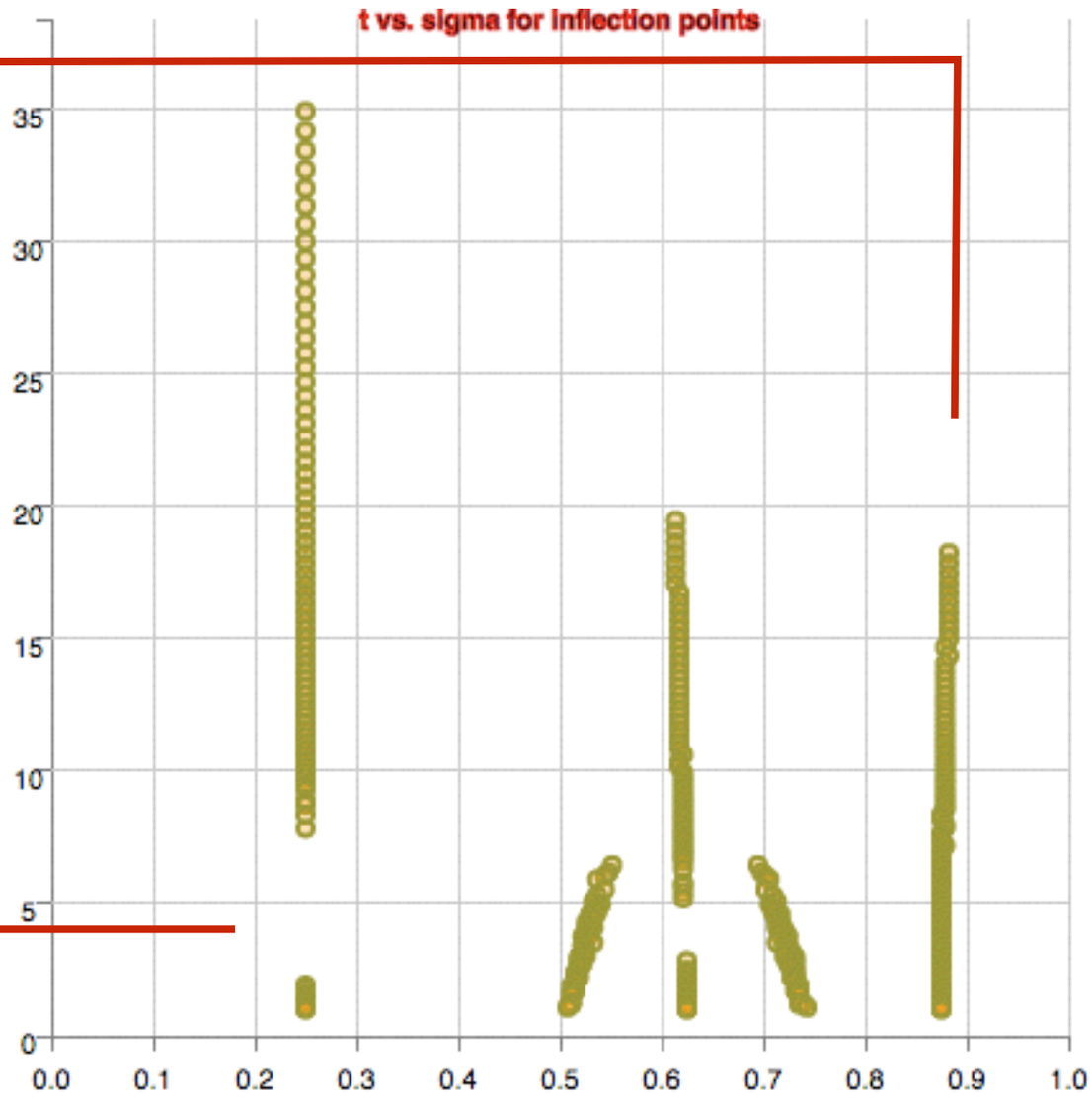
Note that the blob perimeters were not constructed from differences of gaussians, but instead from the borders of contiguous pixels in segmented images.

So a lot of processing of the perimeter is needed to make a usable contour.

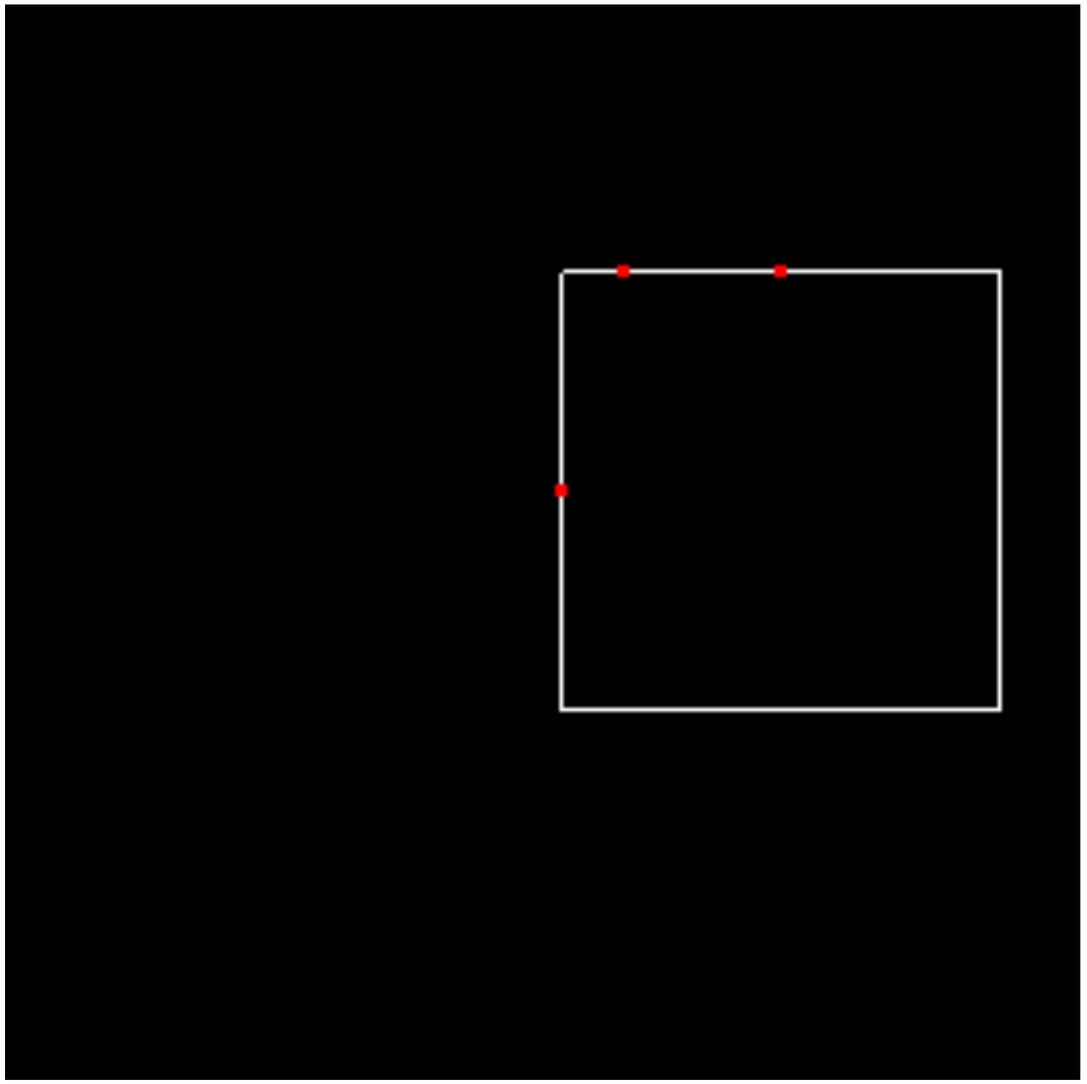
Here’s an example of the blob from a binned segmented image before and after additional line processing.

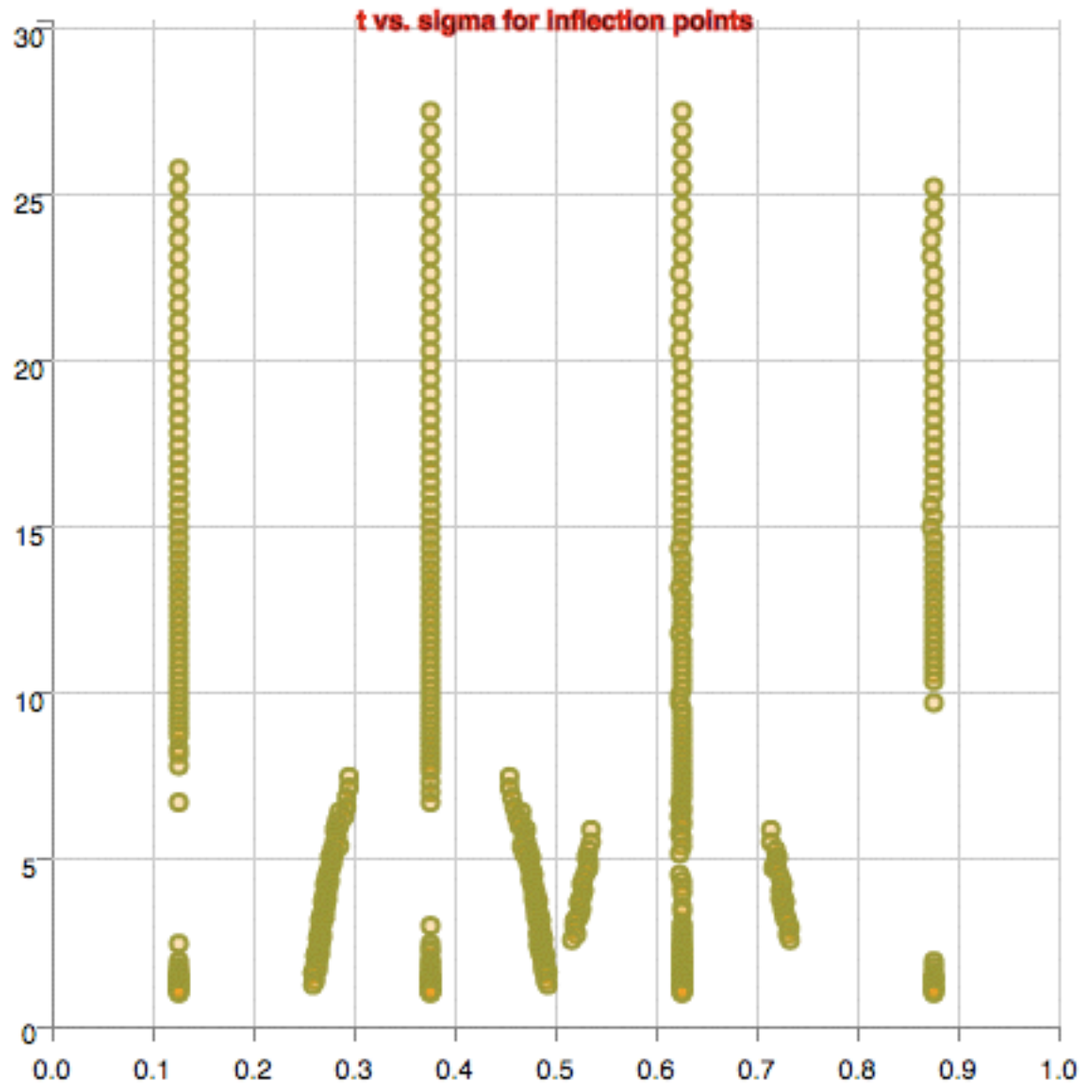




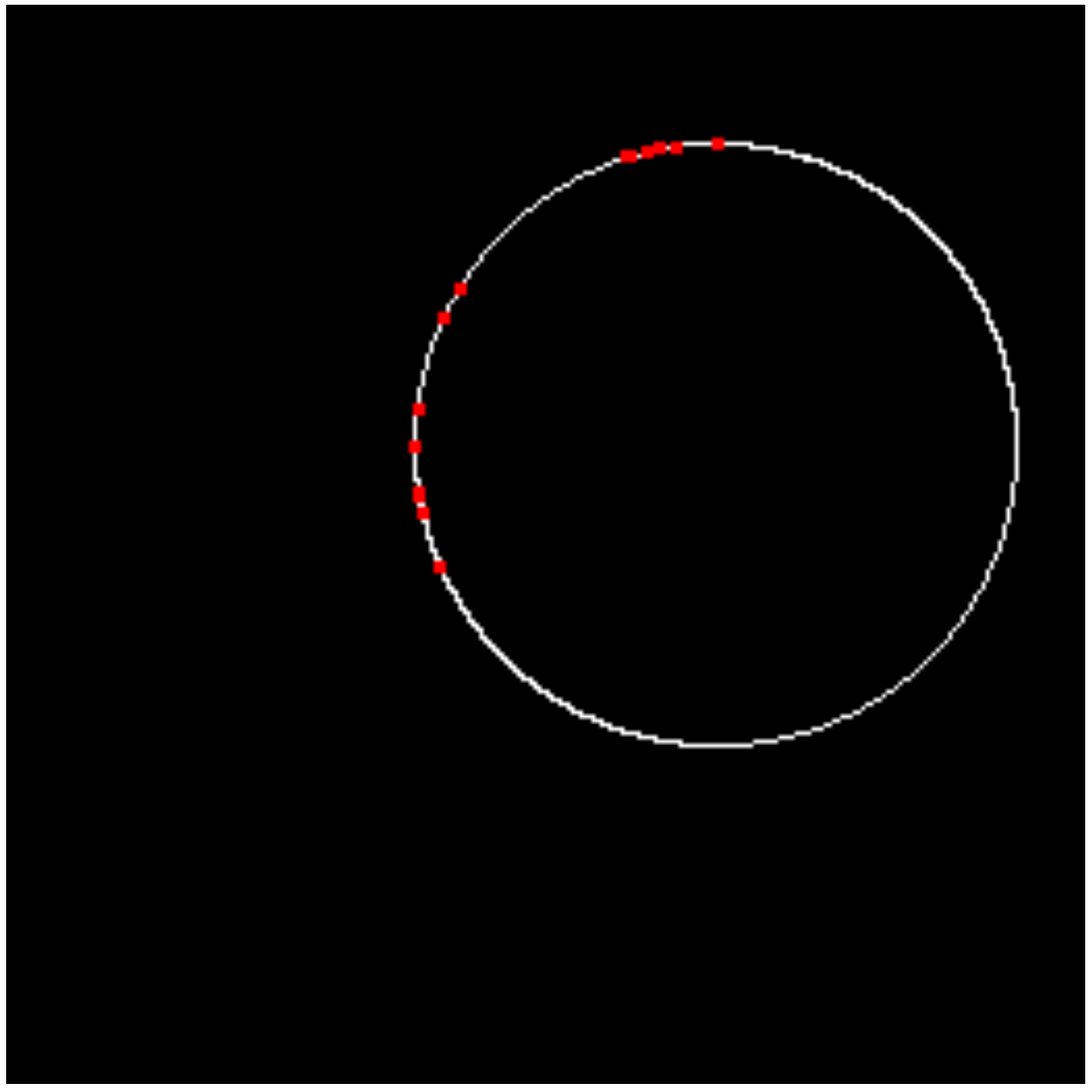


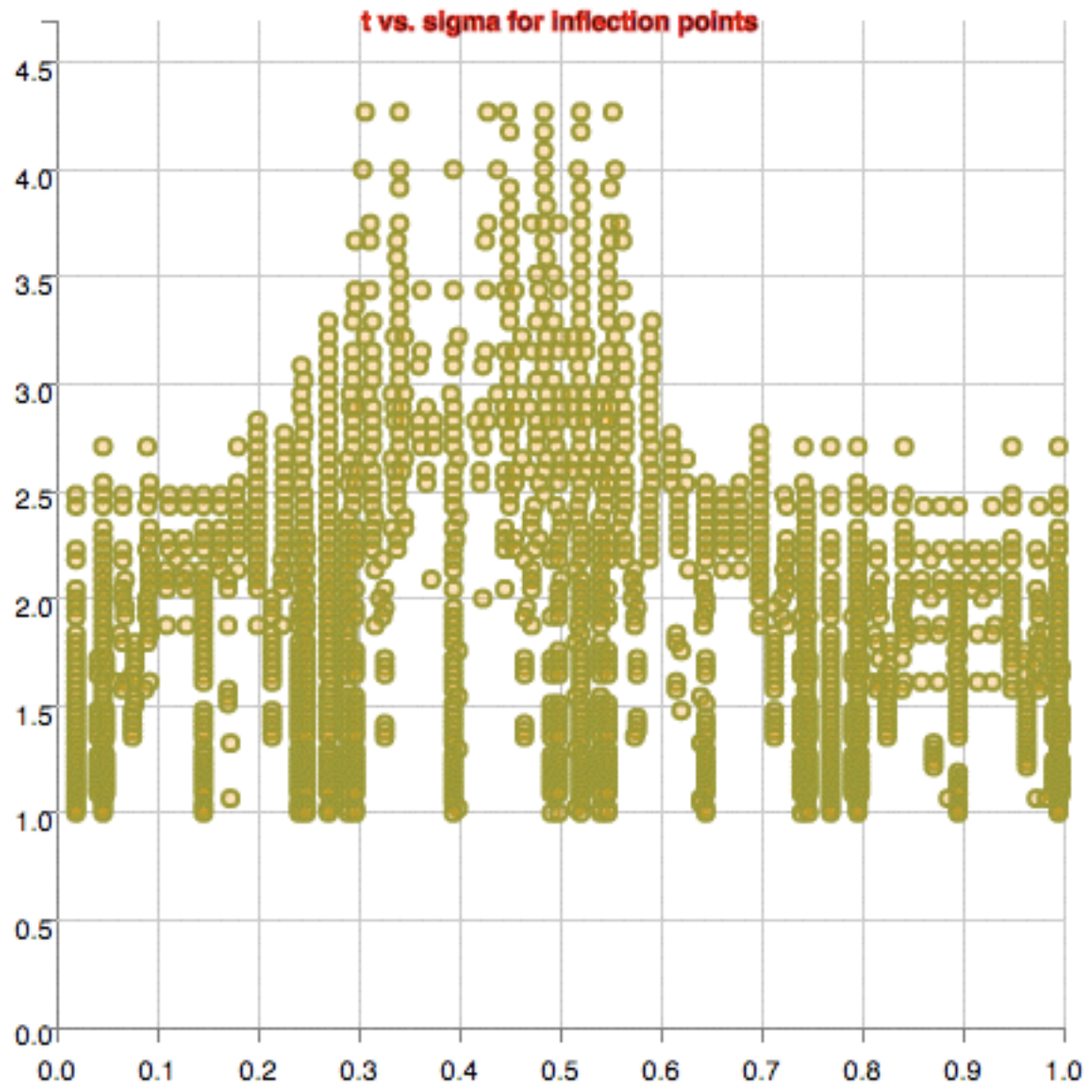
The scale space images are dependent upon defining the zero crossings of curvature, and need to be improved. Finding wide minima combined with zero-crossings has improved the results.





The scale space images are dependent upon defining the zero crossings of curvature, and need to be improved. Finding wide minima combined with zero-crossings has improved the results.





The scale space images are dependent upon defining the zero crossings of curvature, and need to be improved. Finding wide minima combined with zero-crossings has improved the results.

Considering a hybrid approach to use corners around an object, but the runtime might be larger than the contour approach (not considering corner extraction vs scale space image construction and moving feature comparison into the first search):

(1) find the closed curve surrounding an object.

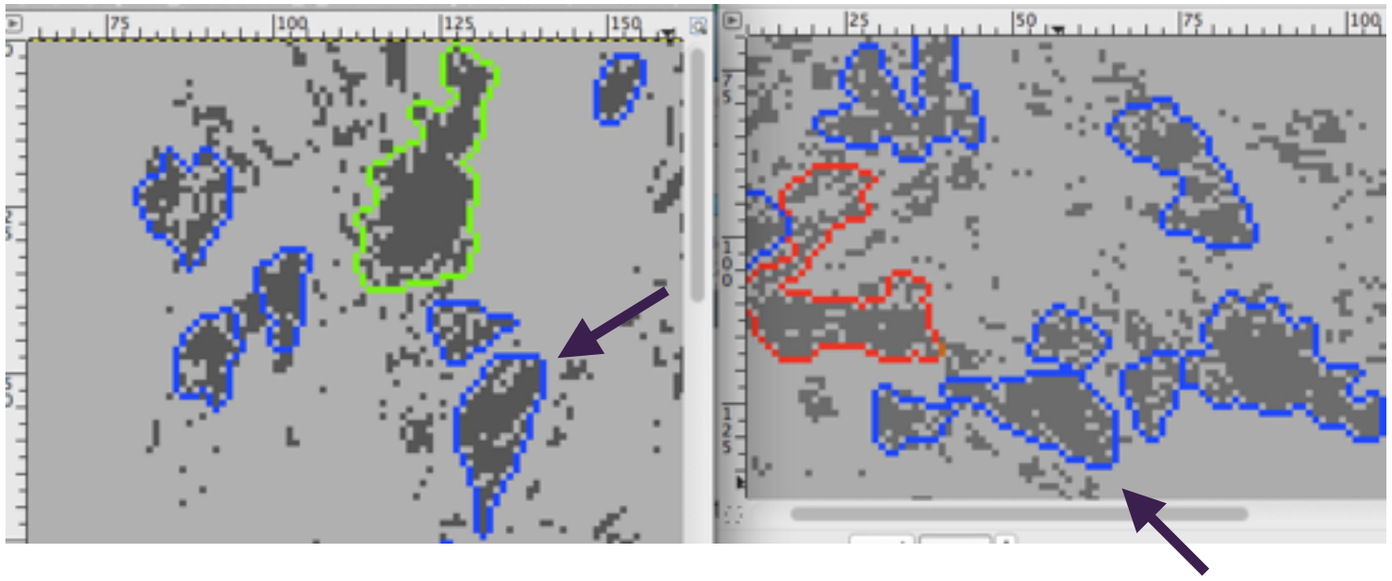
(2) determine corners on the curve.

(3) match pairs of points in one curve in an image with pairs of points in a curve in another image using location and feature descriptors. Like the contour matching, one could make an assumption of strongest characteristic to create a reference point used to determine a reference for the scores and penalties of matching. For a non-scale-space-image approach, the smallest error point descriptor along the first curve could be treated as the reference for all of the first curve's matches. Then one determines the best match for it along the second curve (O(N)). That single point match needs a second correct match to make a euclidean transformation estimate. The second single point to match along the first curve could be found as the one with the next smallest feature error at a reasonable distance from the first point. It's match along the second curve would be found in O(N). From those two matching points, one could estimate the euclidean transformation and then apply it one by one to the remaining points in the first curve and use a scoring an penalty system adapted from the contour matching (SSD instead of sigma, but additive not multiplicative scoring and penalty).

The first few steps of that match are stored in a min cost heap as the cost of the start of a solution. Then the next solution is started w/ the next smallest error point not already in a solution, etc until each point is a starter in a solution.

Then, just as in the contour matching algorithm, the best solution is applied to the remaining points and the best solution (min cost) is found.

Next is an example of blobs found in binary segmented images. The arrowed blob does not have strong inflection points (as currently implemented), but does have somewhat strong corners. (note the greyscale image is used for feature descriptor rather than segmented image shown).



note that for the contour matching, errors in scale due to curve length differences because of deviations in segmentation more strongly affect the scale estimate than a method based on point pair matches.

test images shown have scale =1.

nCurvePoints1	nCurvePoints2
97	78
60	60

So, will implement the new hybrid method of point pair matching around blob contours to estimate scale.

Note that if the only goal is scale, the euclidean transformation for scale could be considered only in scale space, that is in number of points along the curve, but has the same vulnerability to the curve length. That method is safe for textures if the textures are all of the same size in an image, that is corner location predicted from starter solution is relative to location on the curve (does not predict location in image x,y space, only curve point index space). This approach would be better for an image with large projection differences across it too.

If one could be certain that the contours were extremely well formed and deviations such as a large bump present in only one were not possible, no occlusion, etc, the scale space image contour method would be preferred. The hybrid method seems necessary.