

Scrum Update V (YANNIX)

Tags	Scrum Update
Create	@March 29, 2022 9:53 AM
Nickname	
Last edited time	@March 29, 2022 7:22 PM
Site	
Student	
URL	
Scrum Update	[Last week] - ศึกษาและ Implement Region growing segmentation - ออกรอบแบบโปรแกรมและ ตั้ง Main feature ของโปรแกรม - ศึกษาและนำ Pillow มาใช้ใน Region growing segmentation [This week] - เตรียมเรื่องการพิมพ์โค้ดเพื่อ Implement โปรแกรม - ศึกษาเรื่องการทำ segmentation ด้วย Deep learning (เนื้อหาเพิ่มเติม) [Blocks] - สอนวิชา HRI - ทำ presentation วิชา deep learning
Week	
Scores	
Status	

Resource:

Pillow

Source1: <https://youtu.be/6Qs3wObeWwc>

<https://youtu.be/6Qs3wObeWwc>

Source2: <https://youtu.be/oYuM7IjiQZM>

<https://youtu.be/oYuM7IjiQZM>

Source3: <https://youtu.be/2QR2A8EOMIM>

<https://youtu.be/2QR2A8EOMIM>

Source4: <https://youtu.be/dkp4wUhCwR4>

<https://youtu.be/dkp4wUhCwR4>

Region growing segmentation

Source1: <https://youtu.be/ZAXjl9CFvDU>

<https://youtu.be/ZAXjl9CFvDU>

Source2: <https://youtu.be/VaR21S8ewCQ>

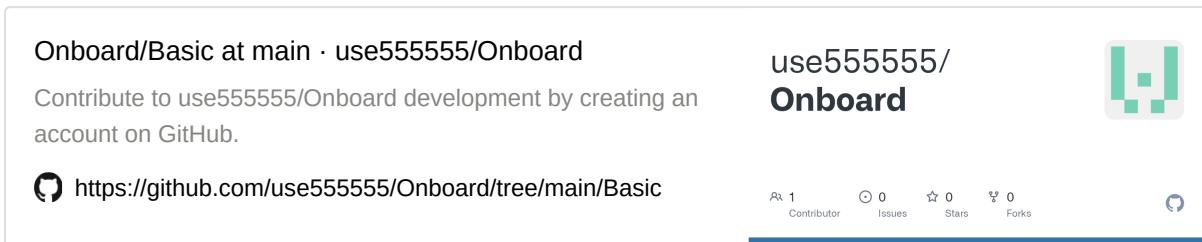
<https://youtu.be/VaR21S8ewCQ>

Code source: [https://developpaper.com/simple-implementation-of-region-growing-in-python/#:~:text=Region growing is an image,are met%, region growth stops.](https://developpaper.com/simple-implementation-of-region-growing-in-python/#:~:text=Region%20growing%20is%20an%20image,are%20met%2C%20region%20growth%20stops.)

My Note:

ในอาทิตย์นี้จะเน้นไปในการศึกษาและเรียน Code อยู่ซึ่งจะเป็นเรื่องของ การรวม Implement region growing segmentation และ การใช้งานของ Library Pillow ซึ่ง code จะอยู่ใน Github folder Pillow และ Segmentation

Code source: <https://github.com/use555555/Onboard/tree/main/Basic>



Pillow

เป็นโปรแกรมที่มีไว้สำหรับการแก้ไขปรับแต่งรูปซึ่งสามารถทำได้หลายอย่างมาก ซึ่งสามารถถูกได้จาก Source4 ของ Pillow และผู้ใช้สามารถนำเสนองานที่ผบกติดว่าจะต้องใช้และ ได้ลองจากหลังที่ได้ดู ก็จะมี Source ซึ่งก็จะมี

การปรับขนาดโดยที่ยังจะเก็บ Aspect ratio ของรูปภาพ ซึ่งจะเป็นการทำการลดขนาดรูปได้อย่างเดียวซึ่งที่ผบกติดว่าจะต้องใช้และ ได้ลองมีขนาดดังนี้

```
size_300 = ( 300, 300 )
size_700 = ( 700, 700 )
```

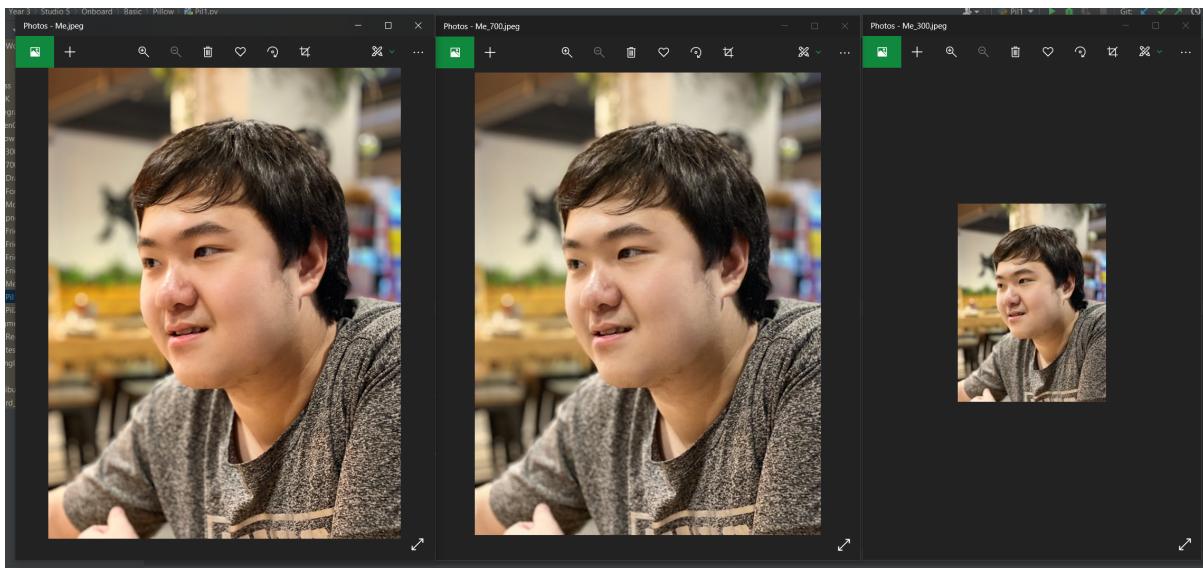
ซึ่งใน code ผบกติดว่าจะต้องการ save หลาย file และผบกติดว่าจะต้องใช้รูปของผบกติดว่าจะต้องการแสดง ซึ่ง Code จะเป็นดังนี้

```
# Get all file and save with resize
for f in os.listdir( '.' ):
    if f.endswith( '.jpeg' ):
        i = Image.open( f )
        fn, fext = os.path.splitext( f )

        i.thumbnail(size_700)
        i.save('700/{}_700{}'.format(fn, fext))

        i.thumbnail( size_300 )
        i.save( '300/{}_300{}'.format( fn, fext ) )
```

และได้ Output ดังนี้ (Output เป็น Original, 700px และ 300px)

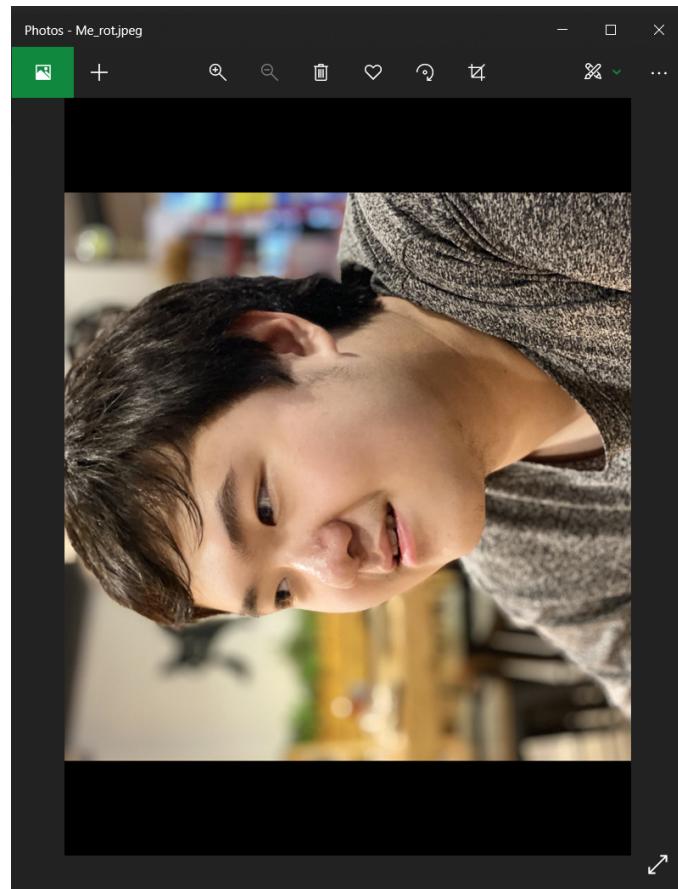


จากที่ Photo ของ window จะแสดงขนาดที่เก่ากันใน 2 รูปแรกแต่จริงแล้วรูปที่สองจะมีความละเอียดน้อยกว่ารูปแรกเนื่องจากมีการปรับความละเอียดลดลง

การหมุนรูปภาพมี code ดังนี้

```
# Rotate picture
Me = Image.open( 'Me.jpeg' )
Me.rotate( 90 ).save( 'Mod/Me_rot.jpeg' )
```

ซึ่งจะมี Output ได้ดังนี้



จะเห็นได้ว่ามีขอบเบื้องจาก code นี้เป็นการหมุนข้อมูลแต่ละ pixel แต่ไม่มีการเปลี่ยนขนาดของรูปภาพจังท่าให้ส่วนที่ไม่มีข้อมูลสีในแบบที่สั้นกว่าก็จะเลือกสีดำ

การปรับโคนสี ซึ่งผมใช้เป็น Gray scale มี code ดังนี้

```
# Convert to black and white
Me = Image.open( 'Me.jpeg' )
Me.convert( mode = 'L' ).save( 'Mod/Me_gray.jpeg' )
```

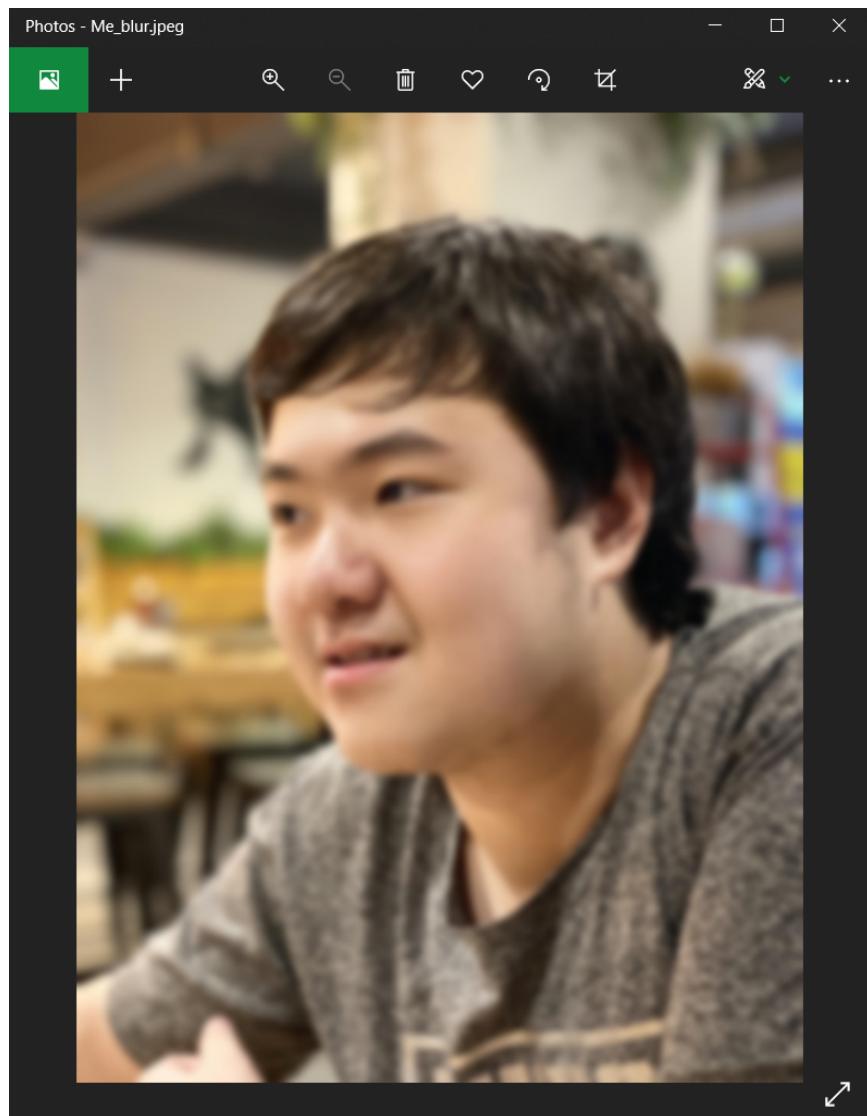
มี Output ดังนี้



การกำเบลอ ชื่อพมໄດ້ລອງໃຊ້ Gaussian blur ມີ code ດັ່ງນີ້

```
# Blur image
Me = Image.open( 'Me.jpeg' )
Me.filter( ImageFilter.GaussianBlur( 15 ) ).save( 'Mod/Me_blur.jpeg' )
```

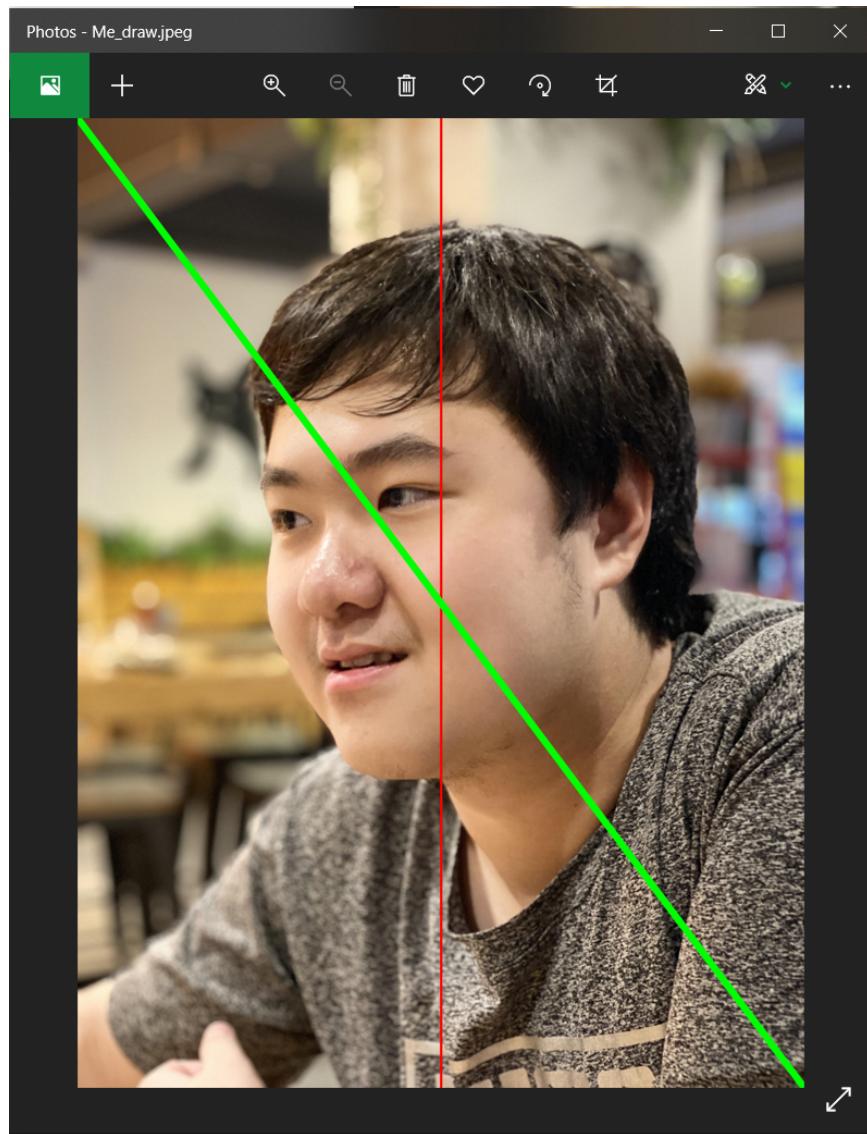
ໃນ code ສາມາຄປຮັບວ່າໃຫ້ເບລອຂນາດໄກນ ชຶ່ງກີ່ພມລອງໃຊ້ໄປເປັນ 15 ຊົ່ງນີ້ Output ດັ່ງນີ້



การวาดเส้นบนรูปภาพมี code ดังนี้

```
# Draw Line on picture
draw = ImageDraw.Draw( Me )
draw.line( ( width/2, 0, width/2, height ), fill = ( 255, 0, 0 ), width = 10 )
draw.line( ( 0, 0, width, height ), fill = ( 0, 255, 0 ), width = 30 )
# XXX.line( start and end coordinate, linecolor, line width )
```

ซึ่งจะมี output ดังนี้

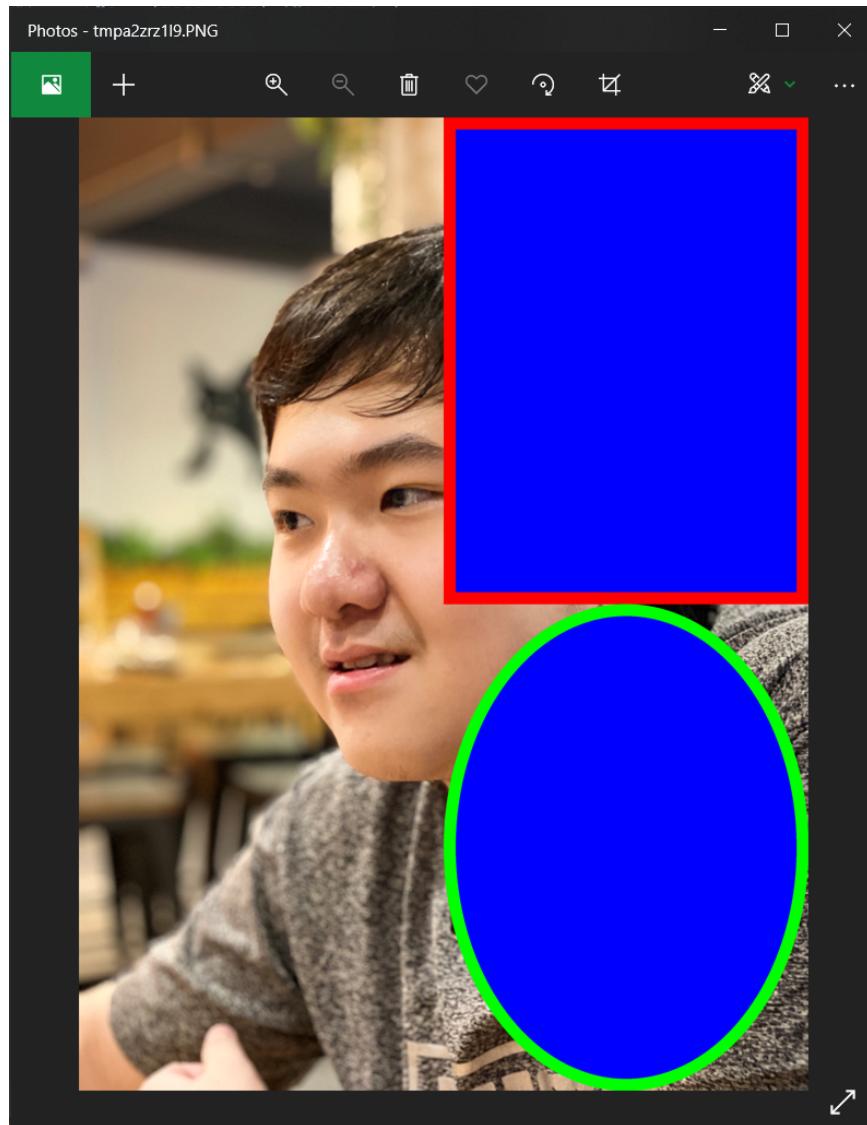


การวาดรูปทรงบนรูปซึ่งจากที่ได้ไปลองก็มี สีเหลี่ยม และ วงกลม/วงรี ซึ่งมี code ดังนี้

```
# Draw shape in
draw = ImageDraw.Draw( Me )
draw.rectangle( ( width/2, 0, width, height/2 ), fill = ( 0, 0, 255 ), width = 50, outline = ( 255, 0, 0 ) )
# XXX.rectangle( top left and bottom right coordinate, fill color, border width, outline color )

draw.ellipse( ( width/2, height/2, width, height ), fill = ( 0, 0, 255 ), width = 50, outline = ( 0, 255, 0 ) )
# XXX.ellipse( top left and bottom right coordinate of bounding box, fill color, border width, outline color )
```

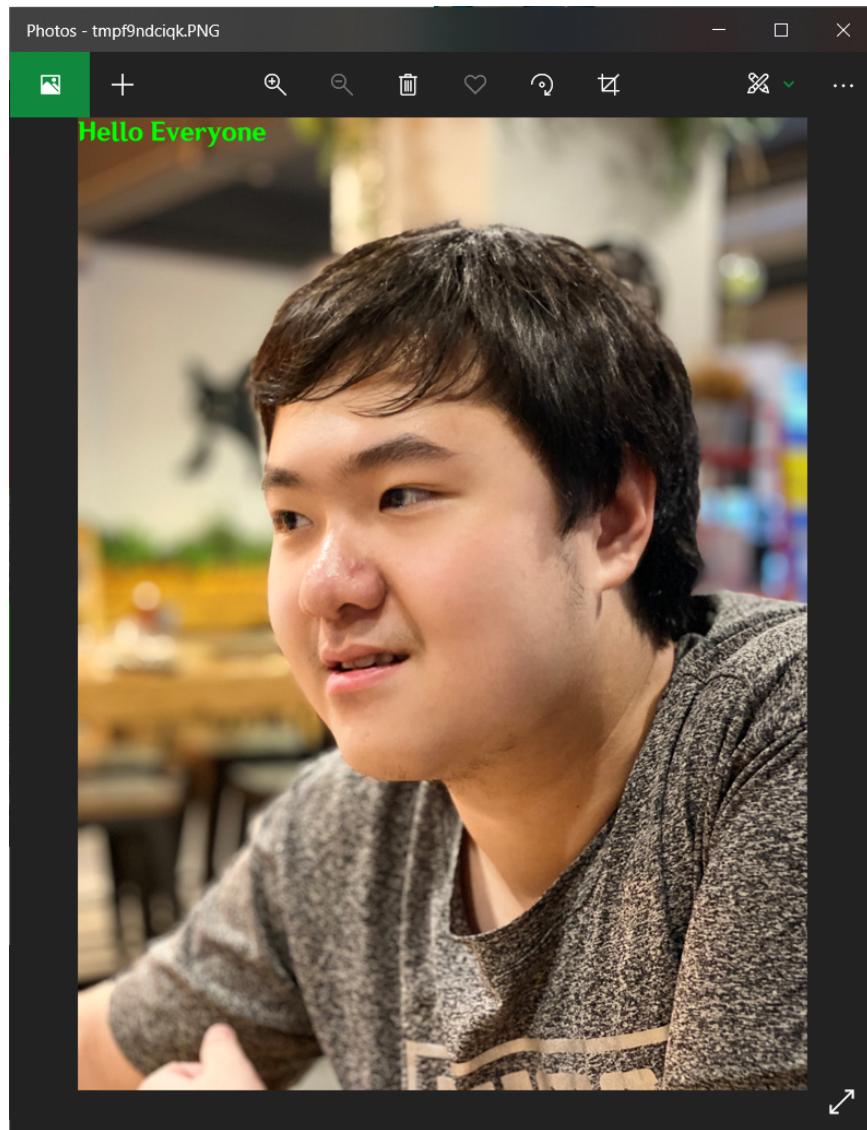
ซึ่งจากที่ได้ลองไปก็มี output ดังนี้



การเขียน text บนรูปซึ่งจะมีการโหลด font รวมด้วยซึ่งจะมี code ดังนี้

```
# Draw text on picture
# Font
font = ImageFont.truetype( 'Font/zh-cn.ttf', 100 )
draw = ImageDraw.Draw( Me )
draw.text( ( 0, 0 ), 'Hello Everyone', fill = ( 0, 255, 0 ), font = font)
# XXX.text( top left coordinate, text, text color, font )
```

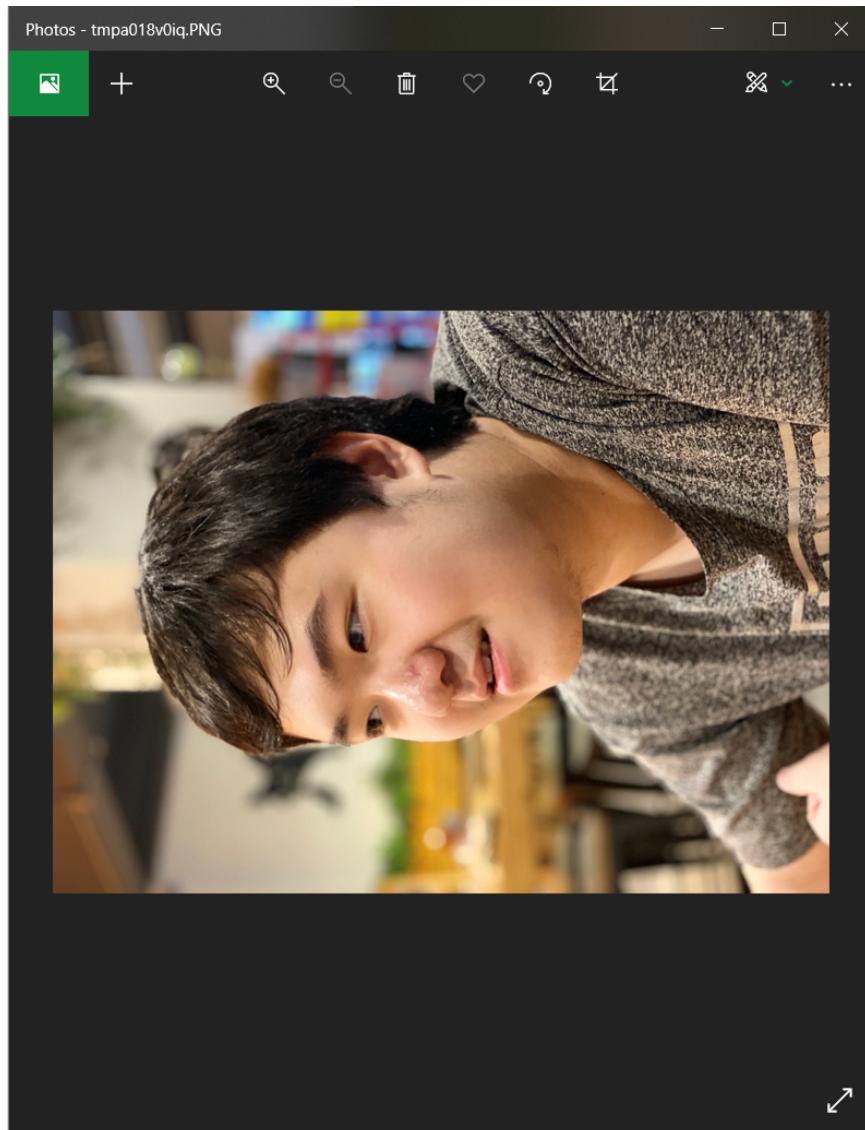
มี output ดังนี้



อย่างสุดท้ายก็ได้ไปลองคือการเปลี่ยน orientation ของรูปซึ่งมี code ดังนี้

```
# Changing orientation using transpose  
Rot = Me.transpose( Image.ROTATE_90 )
```

ซึ่งมี output ดังนี้



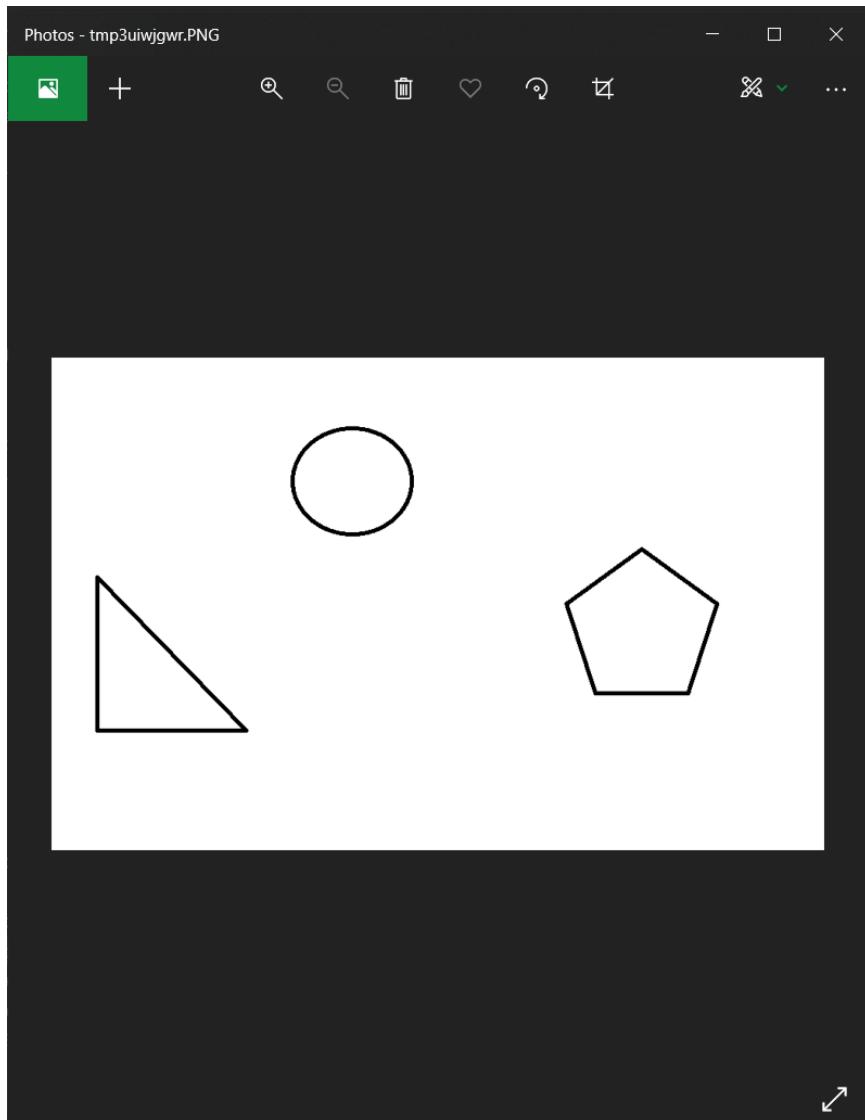
จะเห็นได้ว่าไม่มีขอบดำเหมือนตอนทำการ rotate เพราะคราวนี้จะเป็นการทำการเปลี่ยน orientation ของกั้งตัว matrix ข้อมูลไม่ใช่การหมุนเพียงแค่ข้อมูล ซึ่งในตัวนี้ไม่ได้มีเพียงแค่ rotate แต่เมื่อเรื่องของการ Flip รูปได้ด้วย

ซึ่ง Pillow เองมี function มากกว่านี้แต่ส่วนที่ผมมาลองจะเป็นส่วนที่ผมคาดว่าอาจจะใช้ใน Project

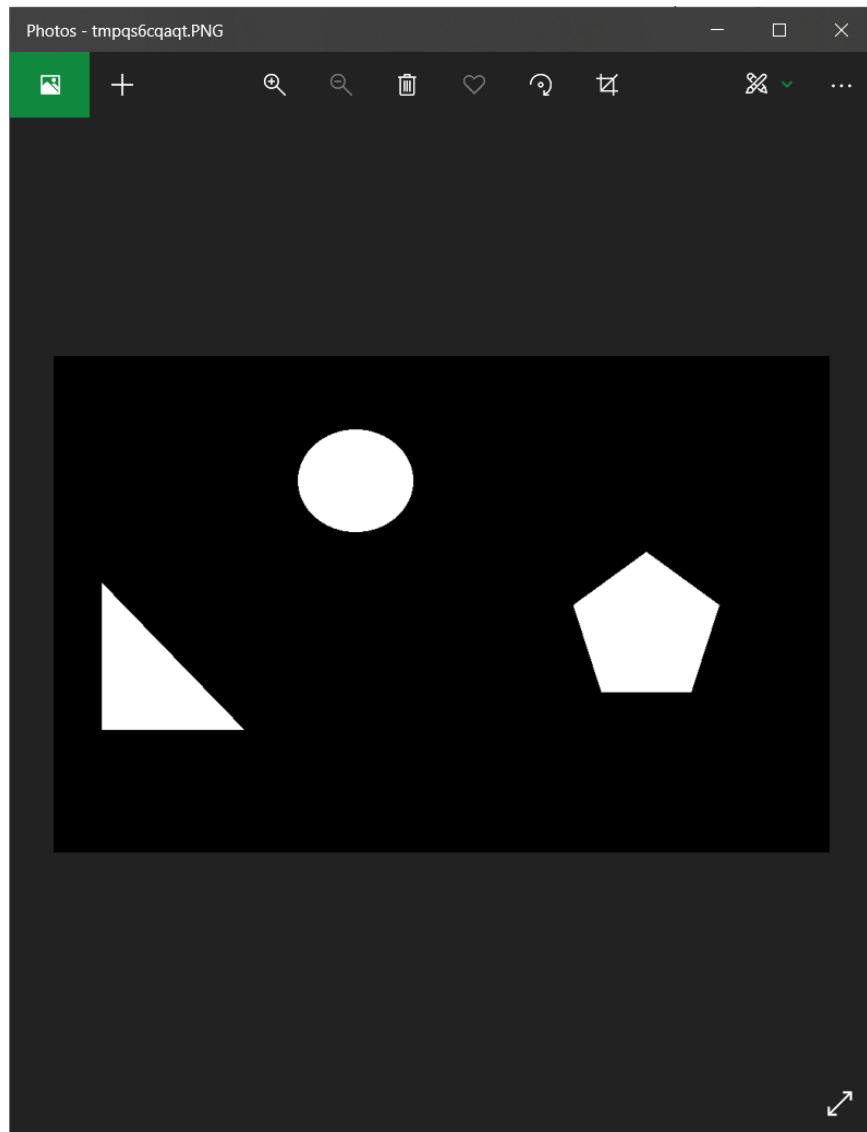
Region growing segmentation

ในส่วนนี้ผมจะเป็นการทำการ Note และนำ code จาก source มาทำความเข้าใจซึ่งผมได้ทำการปรับ code ให้มาใช้ Pillow เนื่องจาก Code ตอนแรกมีการใช้เป็น OpenCV ที่ Advisor ไม่ได้แนะนำให้ใช้ ซึ่ง Idea หลักๆ ของการทำ Region growing segmentation คือการที่เราหย่อนจุดเริ่บต้น (seed) ของ Region ไปแล้วให้โปรแกรมทำการเช็คความแตกต่าง เช่น สี หรือความเข้มใน pixel ที่อยู่รอบตัว seed และเช็คว่าเกิน threshold กี่ตั้งไว้หรือไม่แล้ว ถ้าไม่เกินก็จะให้ pixel ที่

เช็คบัน្តเป็น seed ของ iteration ต่อไป กำหนดไปเรื่องๆ จนกว่าจะไม่มี seed เหลือแล้วทำการเก็บข้อมูล ซึ่งจากที่ผมได้ไปลองและใช้ Input ดังนี้



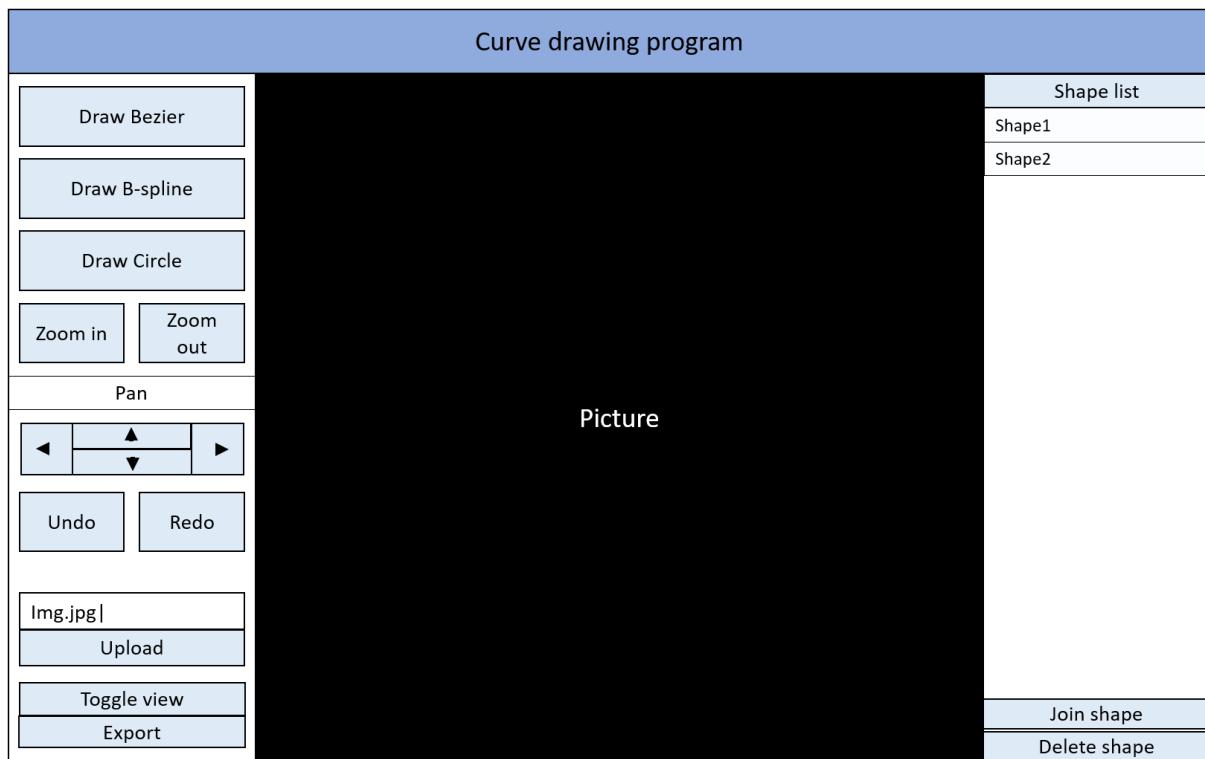
แล้วทำการหย่อน seed ไปในพื้นที่ของรูปทรง และได้ outcome มาเป็น



ซึ่งจะเห็นว่า output ก็ได้เป็น พื้นที่ที่อยู่ภายในของรูปทรงก่ออยู่ในรูป input

Program design

ในส่วนนี้จะเป็นการอธิบายแบบโปรแกรมใหม่หลังจากที่ได้ไปเห็นโปรแกรมการทำการเขียนรูปทรงต่างๆในปัจจุบัน ซึ่ง UI ที่ออกแบบมาที่ดังนี้

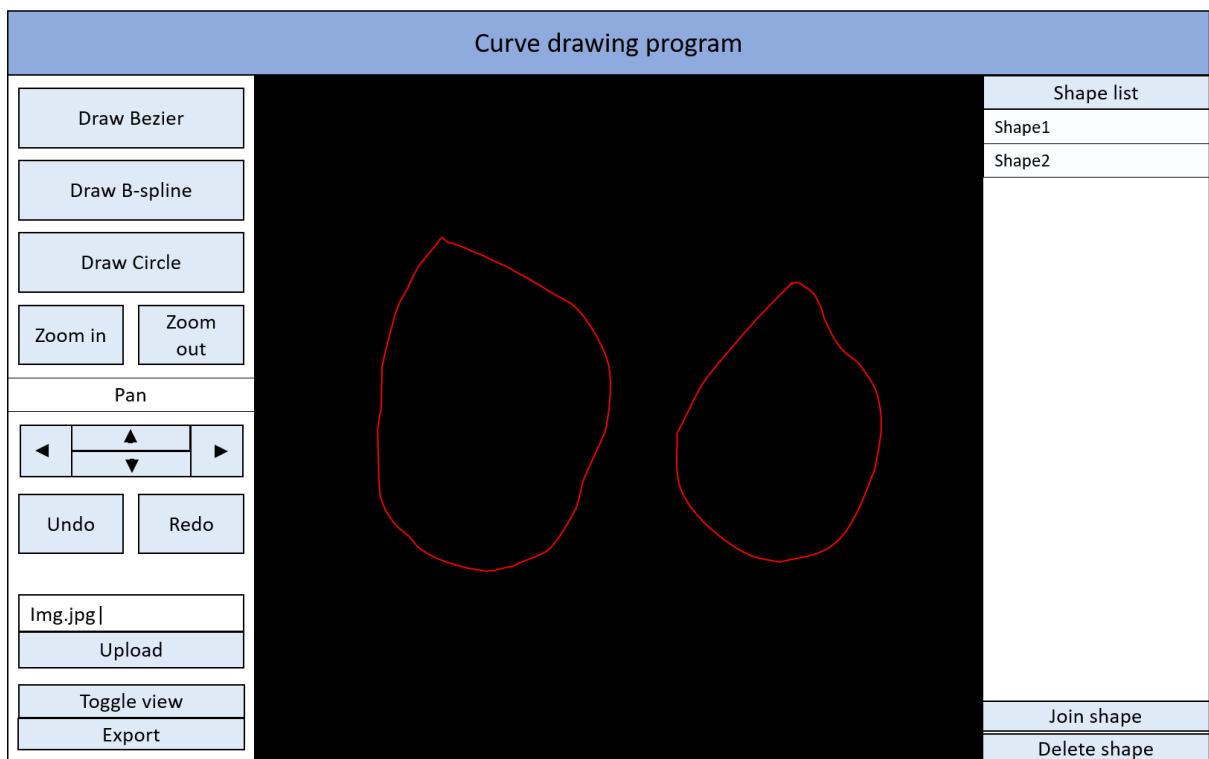


แล้วจะมี main feature ดังนี้

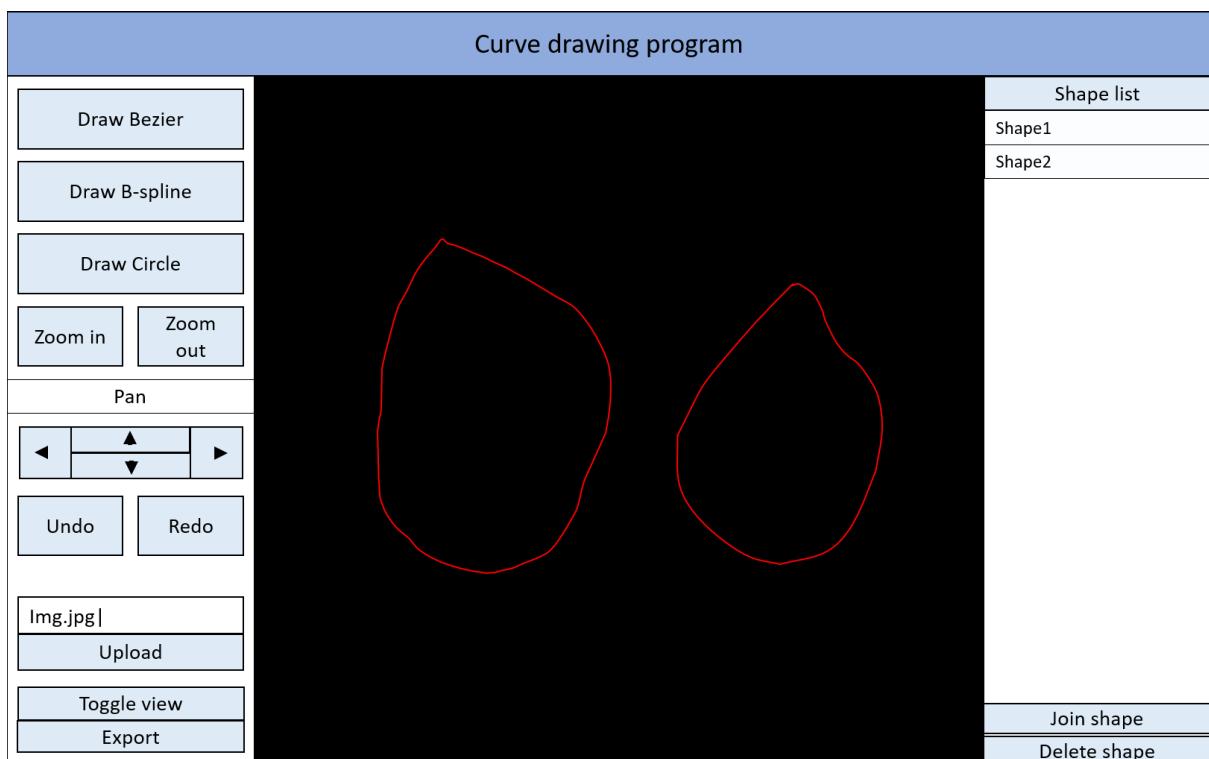
- สามารถวาด Bezier curve, B-spline curve และ circle ได้
- โปรแกรมต้องสามารถ zoom เข้าออก และสามารถ pan หน้าจอได้
- โปรแกรมสามารถโหลดรูปได้ทั้งรูป
- เวลาวาดในโปรแกรมแต่ละครั้งถ้าไม่มีการเลือก shape ใน list จะเป็นการวาด shape ใหม่
- ถ้ามีการเลือก shape จะเป็นการวาดต่อใน shape นั้นๆ
- บุ้ม undo จะเป็นการย้อนจุดล่าสุดใน curve นั้นๆ
- โปรแกรมสามารถ toggle ดูรูปแบบ output ที่จะออกมาได้ และสามารถ output ให้เป็น binary mask ของ file csv ได้

ซึ่ง feature ต่างๆ นั้นก็ได้ออกแบบว่ามีการทำงานดังนี้

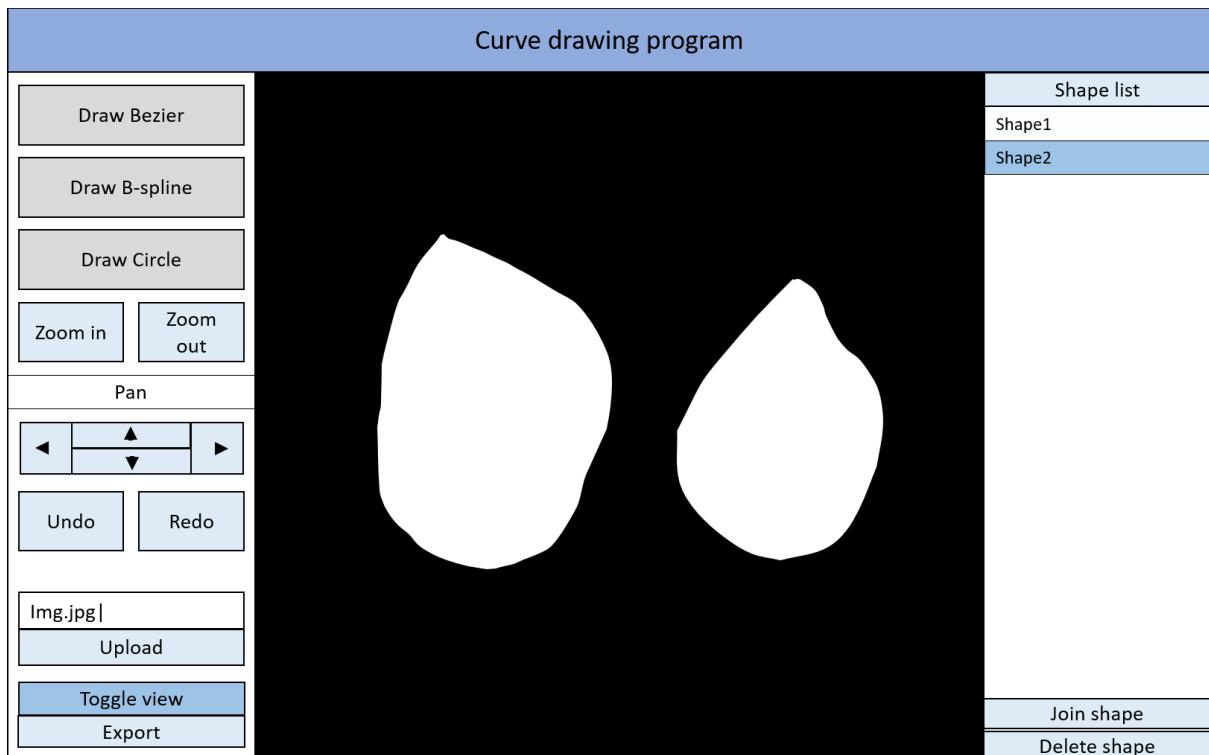
การแสดงผลตอนไม่มีการเลือก shape ได้



การแสดงผลหลังจากเลือก Shape



การแสดงผลเวลา toggle ค่ segmentation



กิจ join shape

