Project Idea description:

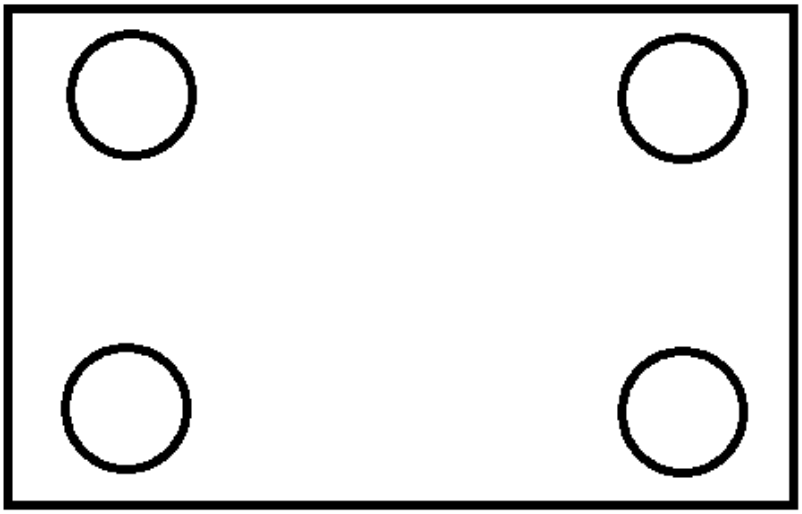
Desktop application for drawing shapes in AutoCAD.

The application takes image contains shapes like circles and rectangles as an input, process this image then outputs the same shapes into AutoCAD project.

The project running through three steps:-

1- preprocessing and detection:

Our algorithm first read image like this one



It contains a rectangel and four circles inside it ,

We detect the rectangle be detecting every line of it by using openCV's function HoughLinesP and detect the four cirlces by hough circle function.

We also can detect half circles (arcs) as we will show examples of it in the video.

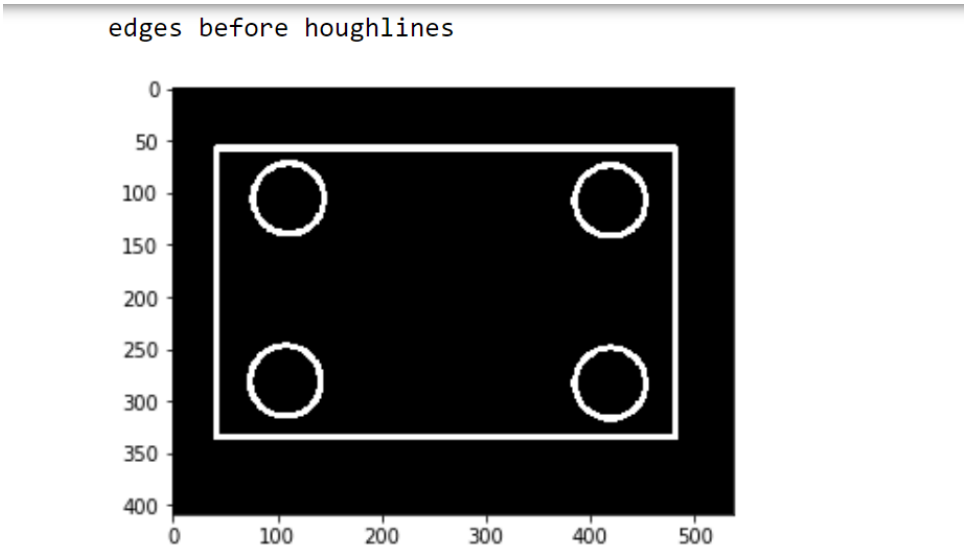
First we convert the original image to gray scale using this opencv's function

```
gray = cv2.cvtColor(test, cv2.COLOR_BGR2GRAY)
```

Then we move to the detect lines step , to prepare the image to HoughLinesP function we apply canny edge detecion to it and convert it to binary image,we also find out that using closing technique will make results better.

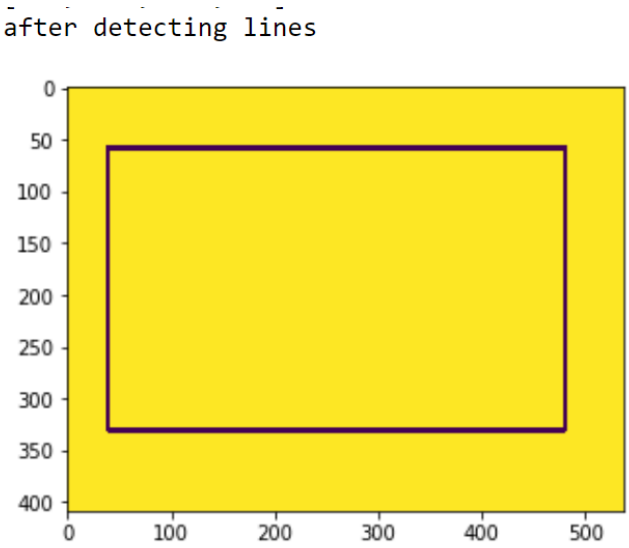
```
    edges = cv2.Canny(gray, 50, 100)                                # get the edges

    #convert to binary image
    ret,edges = cv2.threshold(edges,40,255,cv2.THRESH_BINARY)



    kernel=cv2.getStructuringElement(cv2.MORPH_RECT,(7,7))
    edges = cv2.morphologyEx(edges,cv2.MORPH_CLOSE,kernel)
HoughLinesP
    plt.imshow(edges, cmap='gray')
    plt.show()
    lines = cv2.HoughLinesP(edges, 1, np.pi/180, 80, minLineLength=0, maxLineGap=0)
```

The image after preparing it to enter HoughLinesP function :



HoughLinesP function will output many identical lines so we write code to loop for this lines and filter out only unique lines and true lines of the image .

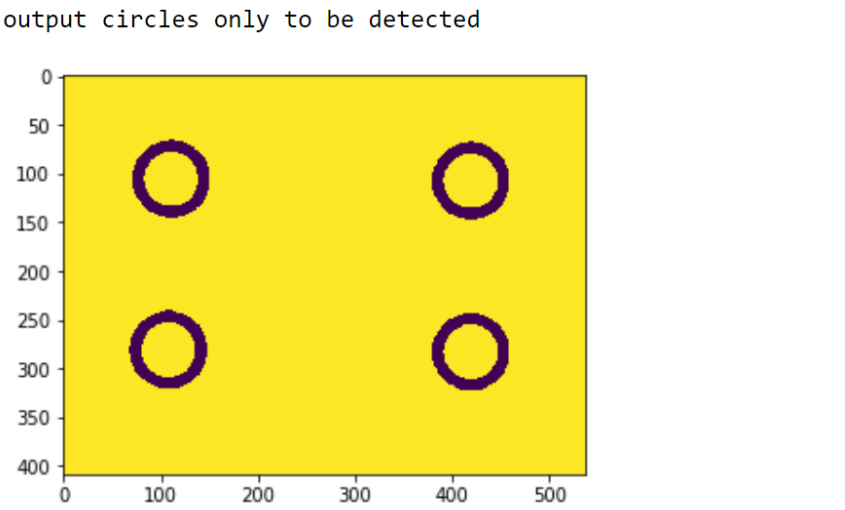
This is image of drawing lines after filtering on a blank image :



Then we loop over all lines and divide them into Vertical lines, Horizontal lines and general lines and put every gener in a list to prepare them to be drawn on Autocad later.

After that we deal with circuits and arcs ,we prepare the image to enter the HoughCircles function by deleting lines from the image and leaving only cirlces and arcs, we do that by subtracting the image with lines only from the original image

This will be the output of substracting and it will be ready for HoughCircles function:



After extracting the cirlces we put it in a list to be ready to be drawn on Autocad.

Before using dxfwrite library we loop over circles list to divide them into full circles , upper Arc ,lower Arc, Right Arc and left Arc .

2- linking with Autocad

Now we will illustrate how th dxfwrite library work.

```
from dxfwrite import DXFEngine as dxf
from dxfwrite.dimlines import dimstyles, LinearDimension ,RadialDimension
```

Class shapes:

This class links python with AutoCAD. And Generating dxf file.

It contains many functions to draw lines, circles and arcs without dimensions and also with appropriate dimensions.

It supports also writing text if you want on your drawing.

And here we will discuss these functions with details:

This function uses to write any text at any position and also you can choose one of the seven color.

```
def text(self,text,x,y,color = Foshia):
    text = dxf.text(text, (x, y), height=0.3, rotation=0)
    text['layer'] = 'TEXT'
    text['color'] = color
    self.drawing.add(text)
    self.drawing.save()
```

This function uses to draw a line with any slope with no dimension

```
def line_with_NO_dims(self,x1, y1, x2, y2,color = White):
    line = dxf.line((x1, y1), (x2, y2))
    line['color'] = color
    self.drawing.add(line)
    self.drawing.save()
```

We use this to draw horizontal line with under dimension

```
def Horizontal_line_with_under_dims(self,x1, y1, x2, y2,length,color = White):
```

We use this to draw horizontal line with upper dimension

```
def Horizontal_line_with_upper_dims(self,x1, y1, x2, y2,length,color = White):
```

And this function for vertical lines only with left dimension

```
def vertical_line_with_left_dims(self,x1, y1, x2, y2,length,color = White):
```

This is the same, but with Right dimension

```
        self.drawing.save()

    def vertical_line_with_right_dims(self,x1, y1, x2, y2,length,color = White):
```

This function use to draw a circle without dimensions
r = radius of the circle

(x,y) = coordinates of the center

Color = used to determine color of circle

```
def circle_with_NO_dims(self, r, x, y,color = White):
```

The same as before but with dimensions

```
def circle_with_dims(self,r, x, y,color = White):
```

This used to draw an arc with no dimensions

```
def arc(self,r,x,y,Theta1,Theta2,color = White):
```

This used to draw dashed line

```
def dashed_line(self,x1,y1,x2,y2,color = Red):
```

Dimensions :

A bit more details about how to control the Dimensions :

First we use dxfwrite.dimlines to draw Dimensions

1) Linear Dimensions :

```
points = [(x1, y1), (x2, y1)]
```

we determine the x-position of the Dim line above. And the y-position determine the height of blue line.

```
dimstyles.new("arrow",
              tick="DIMTICK_ARROW",
              scale=1,
              height = 2,
              tick2x=True,
              dimlineext=0.,
              tickfactor=length/5)
```

These line of code create new dim line and determine some properties :

1st and 2nd parameter: choose the dim style.

3rd parameter: choose any scale you want (m , cm , mm , …..)

4th parameter: determine the font of dimension written above the line .

5th parameter: it says true to be two arrows in the dimension.

6th parameter: determine the extension of line out of arrows.

7th parameter: determine the size of arrow itself.

```
self.drawing.add(
    LinearDimension((10, y1 - length/10), points, dimstyle='arrow', angle=Theta))
```

this line add this linear dim to autocad file.

1st parameter: determines y-position of the dim line.

2nd parameter: is the points above.

3rd parameter: is the style of dim line (here it's arrow).

4th parameter: is the angle of rotation of this dim line.

We create under and upper dimensions for horizontal lines, also we create right and left dimensions for vertical lines.

We choose the appropriate parameters for each one.

2) Radial Dimensions

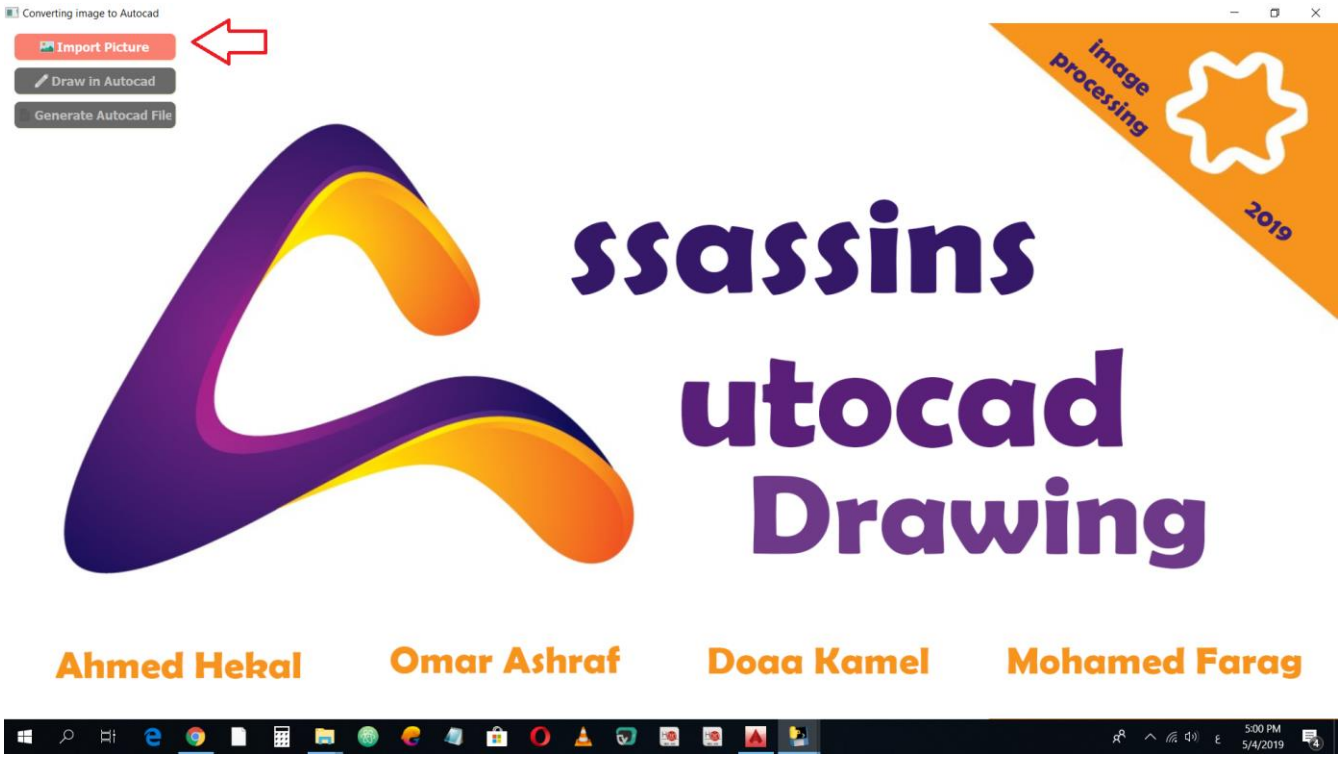
We use radial dimensions with circles

```
dimstyles.new("radius",scale=1, height=r/4, prefix='R=',tickfactor=r/1.5)
```

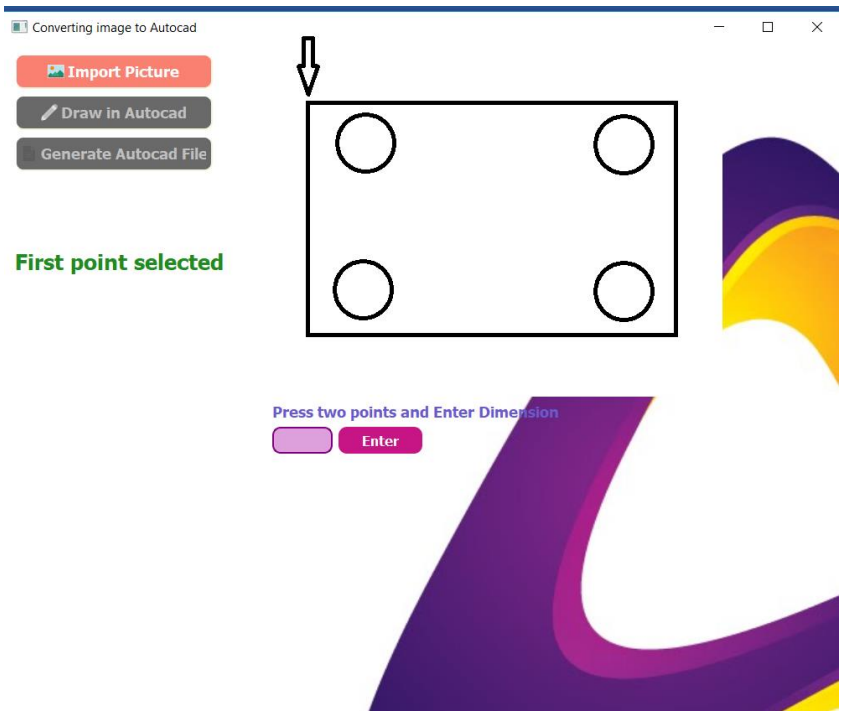
And we use the same parameters except for prefix which used to write "R=" before the radius.

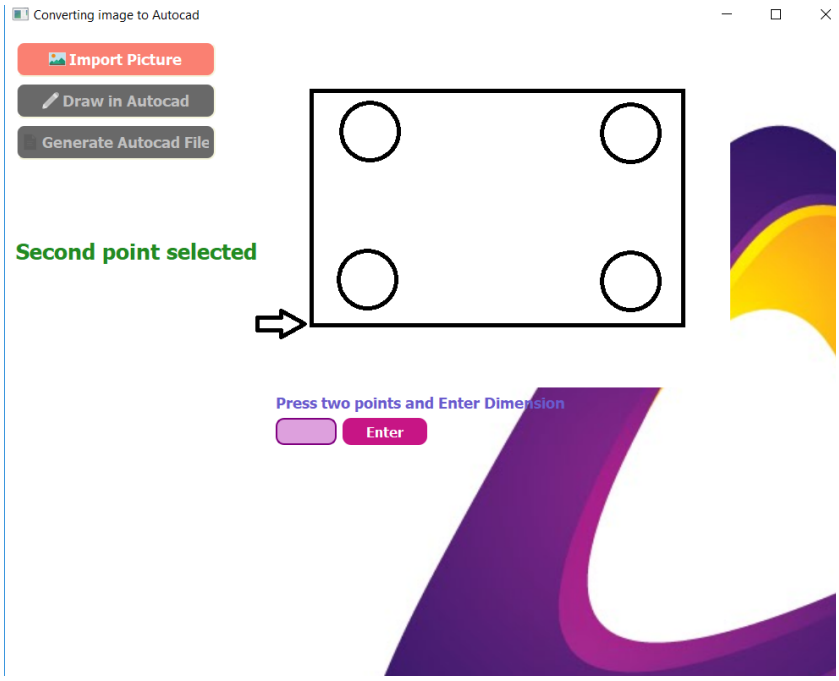
3- merging the code with user interface (Gui):

First user will import the image by clicking on “import picture” button and choose an image



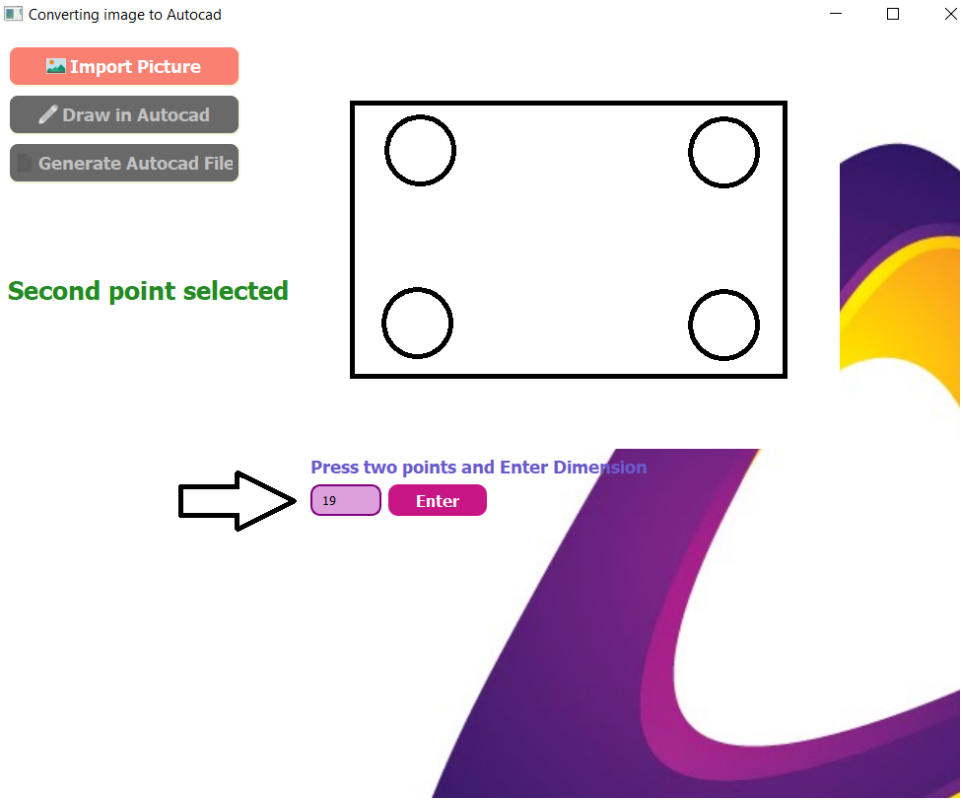
then he should click on two points of any line like this :





Then user should enter the length of that line with cm

Then press “enter” button



Finally if the user has autocad in his computer he can press “Draw in Autocad” to open an autocad project and draw on it, other wise he can press “generate Autocad file” to only generate dxf file containig the drawing.

The output in Autocad :