Assignment #1

Automatic pieces detection and quantification in thermographic image

- **Libraries used**
  1. OpenCV Python Library
  2. NumPy Python Library
  3. MatplotLib Python plotting Library
- **Algorithm**
  1. **Resize the original image**
     The original input image is resized while maintaining aspect ratio so that parameters of Hough circles function are the same
  2. **Convert to grayscale**
     convert resized image to grayscale
  3. **Apply GaussianBlur**
     Apply GaussianBlur with 5*5 kernel to gray image to reduce noise and improve contour detection
  4. **Apply HoughCircles to detect circles**
     Apply Houghcircles function to blurred image to obtain all circles even partial circles since votes parameter is not returned
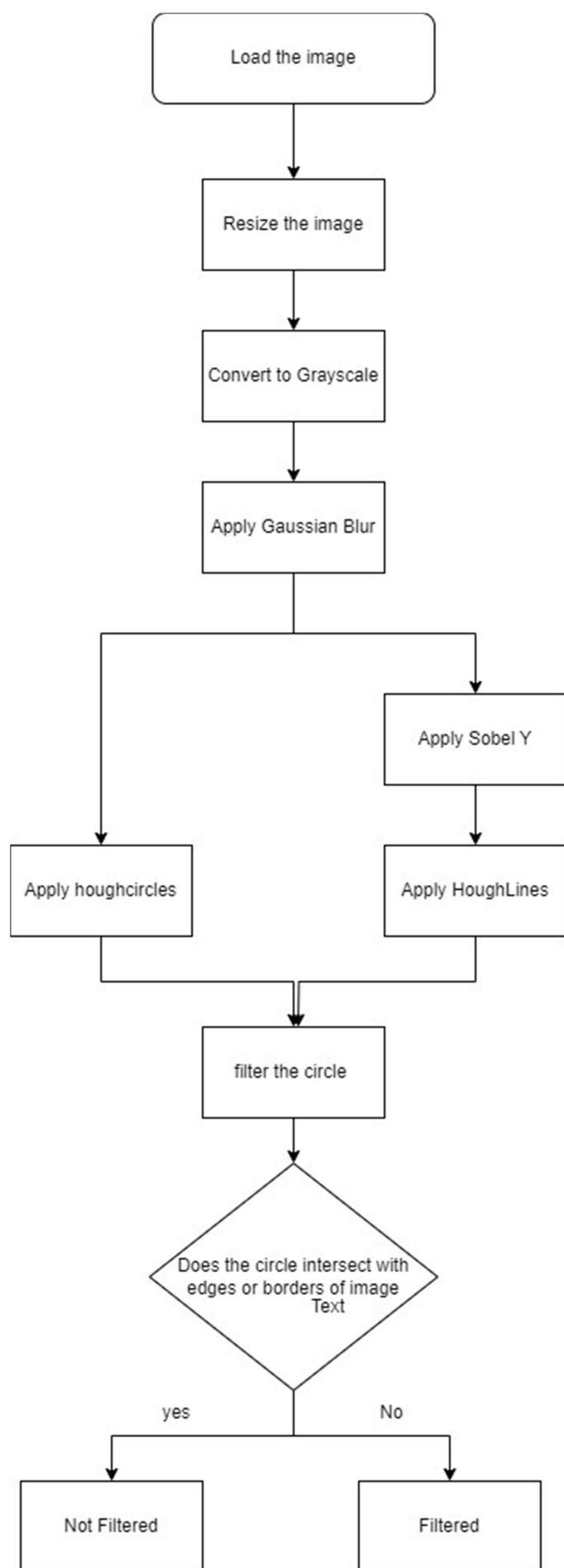 5. **Apply Sobel Y filter**
     Apply Sobel Y filter to blurred image to detect horizontal edges used later to detect intersection of edges with circles since main edge is horizontal
  6. **Apply Hough Lines to Sobel Y filter**
     Apply HoughLinesP to obtain line segments in Sobel Y convolved image
  7. **Detect and filter the circle**
     filter the circle only if it is not intersected with any edge obtained from Sobel Y convolved image or an image border by comparing it's radius with perpendicular distance to the edge or border

```
                    ┌─────────────────────┐
                    │   Load the image    │
                    └─────────────────────┘
                              │
                              ▼
                    ┌─────────────────────┐
                    │   Resize the image  │
                    └─────────────────────┘
                              │
                              ▼
                    ┌─────────────────────┐
                    │ Convert to Grayscale│
                    └─────────────────────┘
                              │
                              ▼
                    ┌─────────────────────┐
                    │  Apply Gaussian Blur│
                    └─────────────────────┘
```

Load the image

Resize the image

Convert to Grayscale

Apply Gaussian Blur

Apply Sobel Y

Apply houghcircles

Apply HoughLines

filter the circle

Does the circle intersect with edges or borders of image Text

yes

No

Not Filtered

Filtered

- **Visual outputs**

  Run **test_per_image.py** for each of the input images to get the following results.
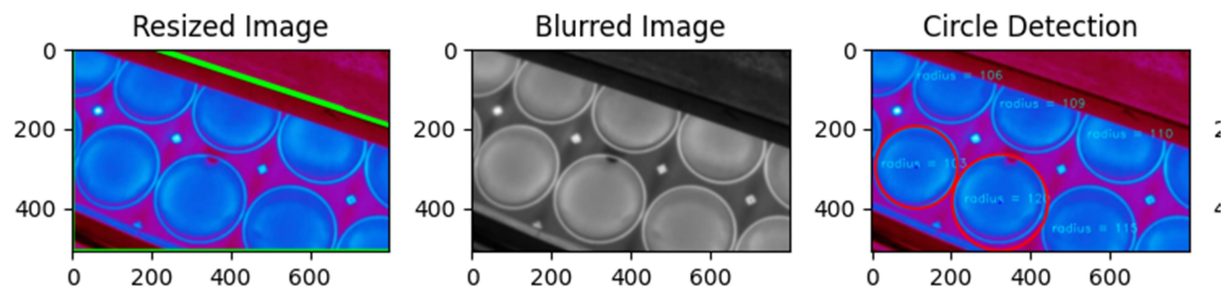
```
# Load image
image = cv2.imread("images/image1.png")
```

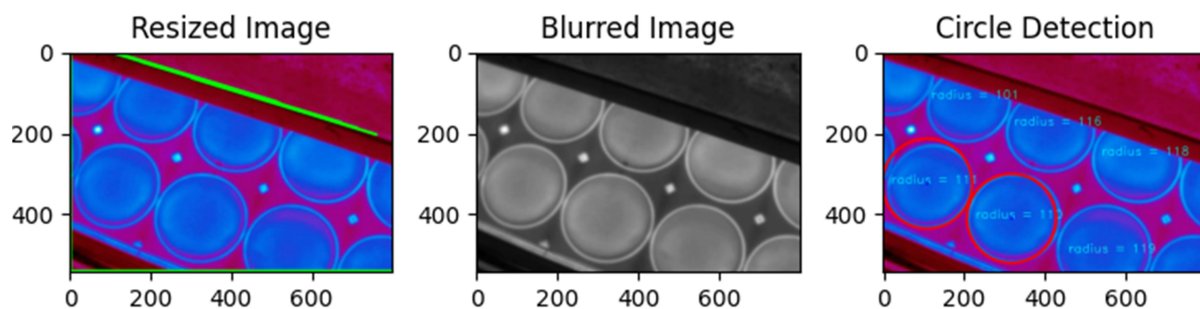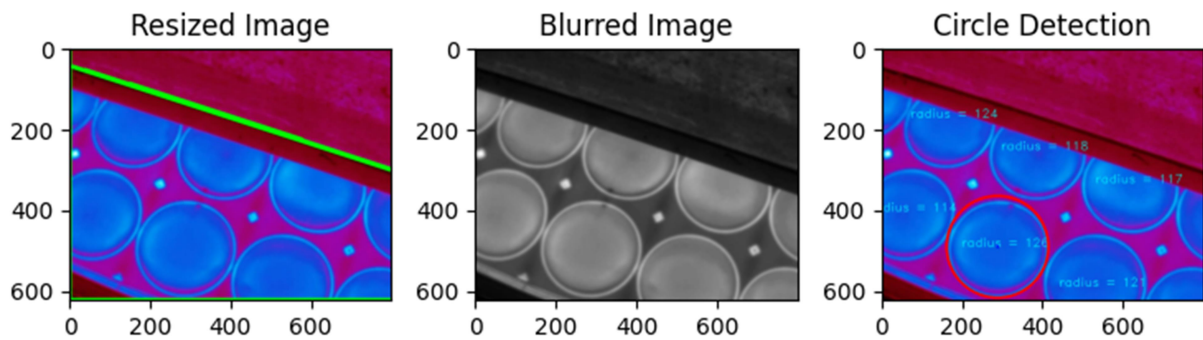 The edge lines are shown in the resized image.
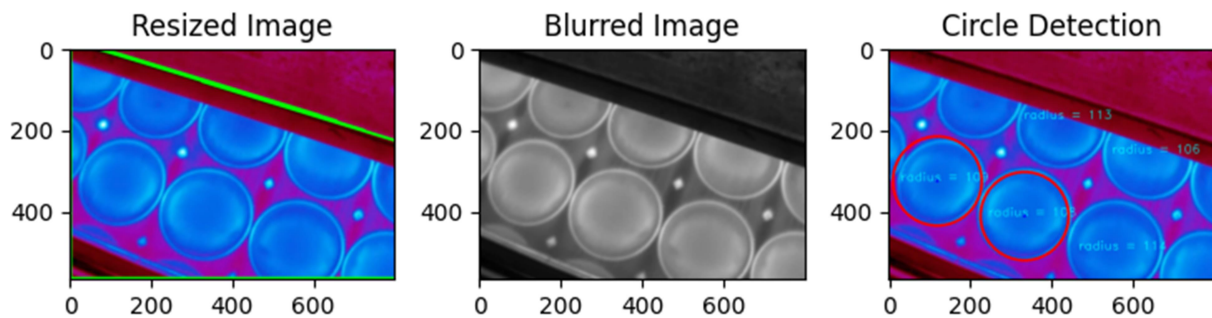
  ### 1. image1.png
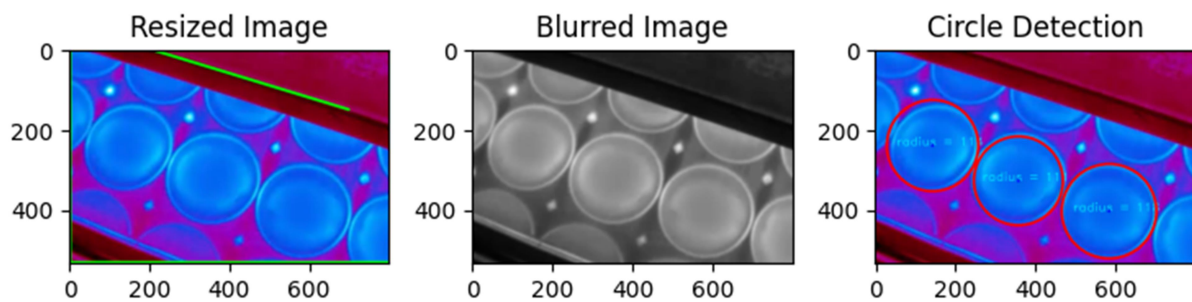
  ### 2. image2.png
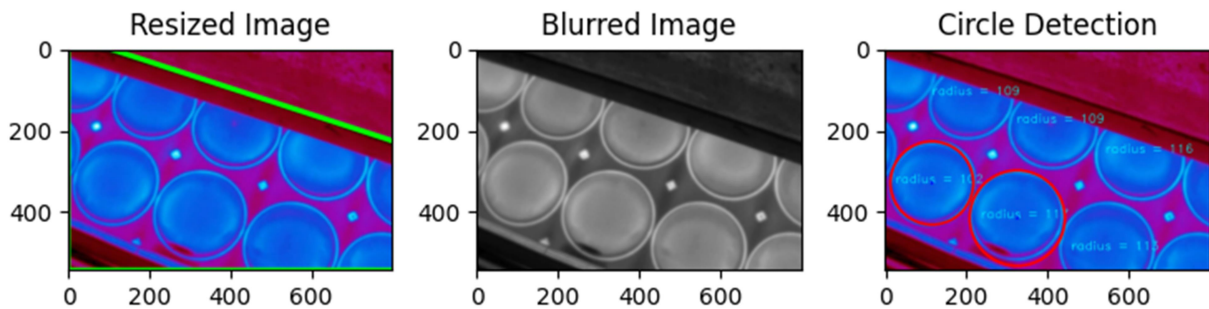

  ### 3. image3.png

4. image4.png

5. Image5.png

6. Image6.png
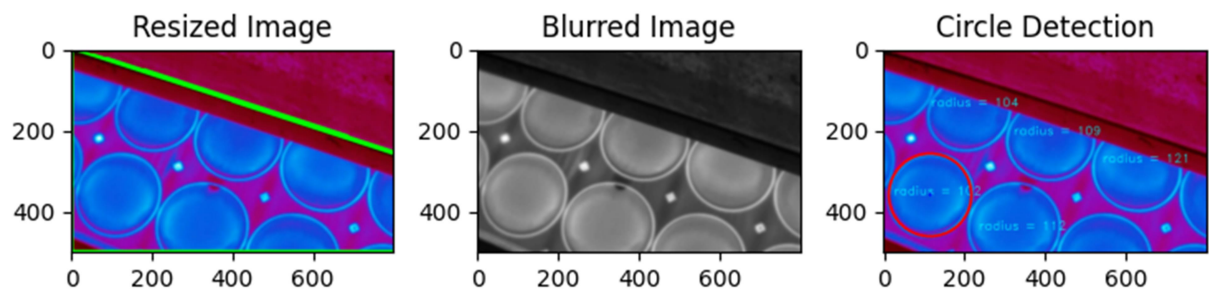

7. Image7.png

Run **Assignment1.py** to get the results for all input images in one figure