```
In [ ]:  import cv2
         import numpy as np
         from matplotlib import pyplot as plt
```

```
In [ ]:  # Load Image
         img = cv2.imread('Source/gray.bmp')
         # img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
         # img = cv2.bitwise_not(img)

         plt.imshow(img, cmap='gray')

         def imgDisplay(localImg1, localImg2):
             plt.figure(figsize= (11, 11))

             plt.subplot(1, 2, 1)
             plt.imshow(localImg1, cmap= 'gray')

             plt.subplot(1, 2, 2)
             plt.imshow(localImg2, cmap= 'gray')

             plt.show()
```
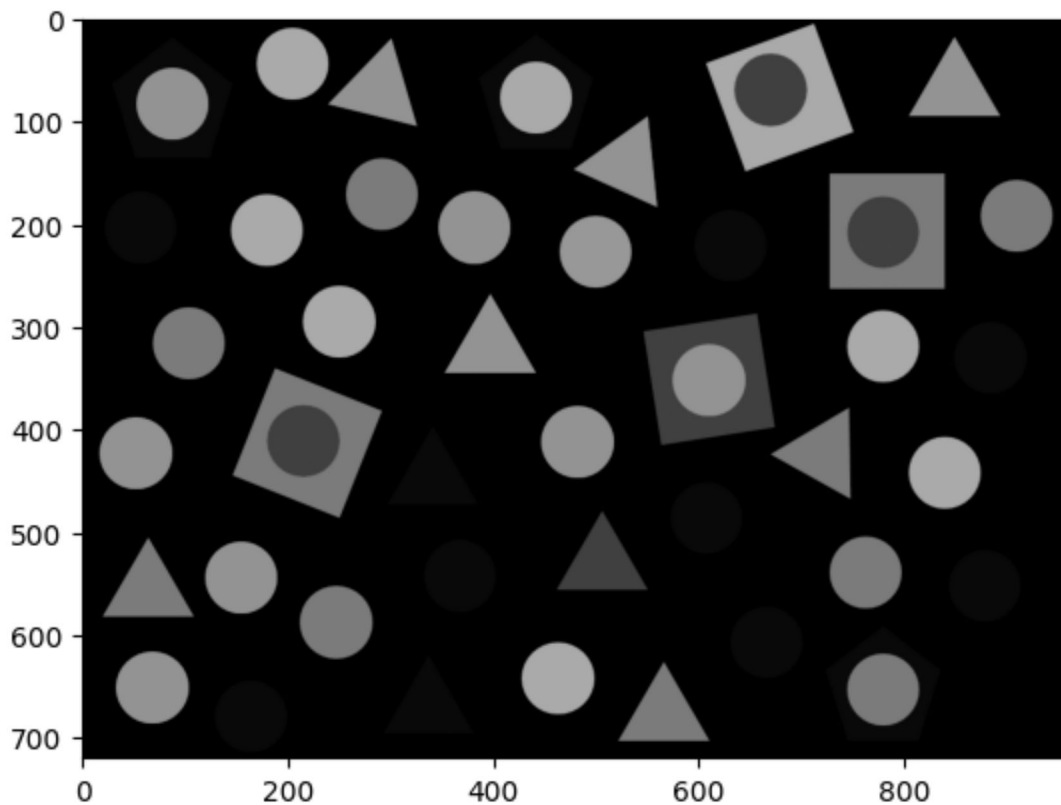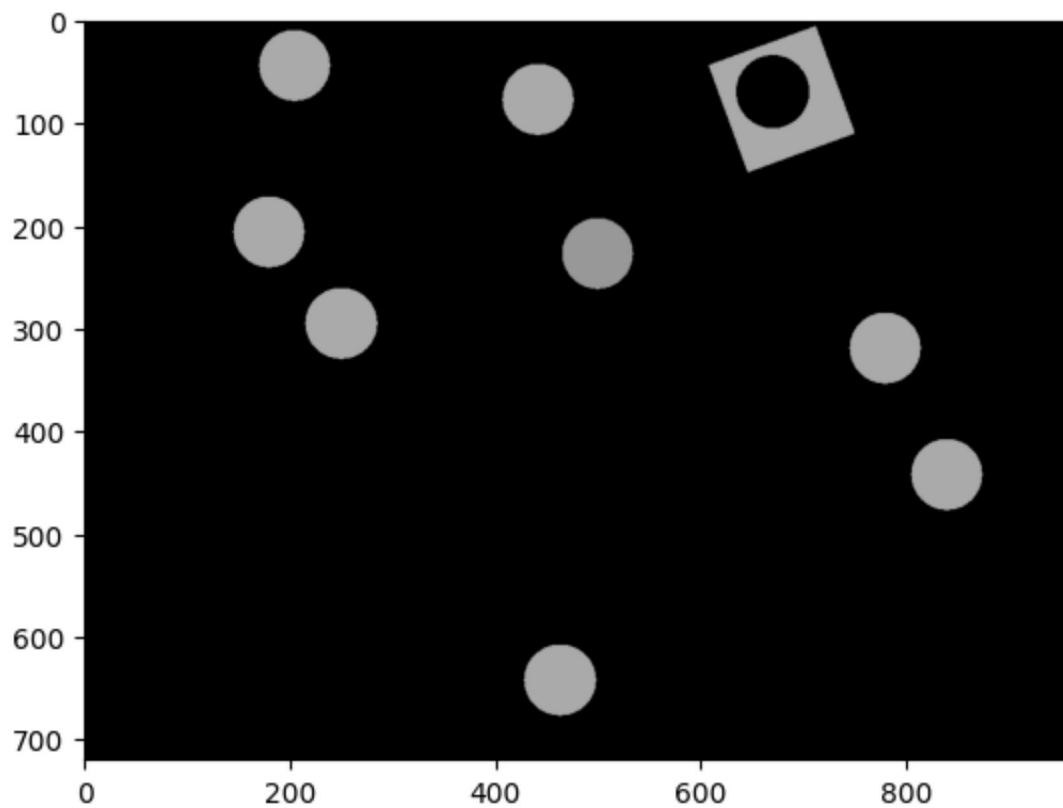


```
In [ ]:  ret, green = cv2.threshold(img, 150, 255, cv2.THRESH_TOZERO)

         plt.imshow(green, cmap= 'gray')
```
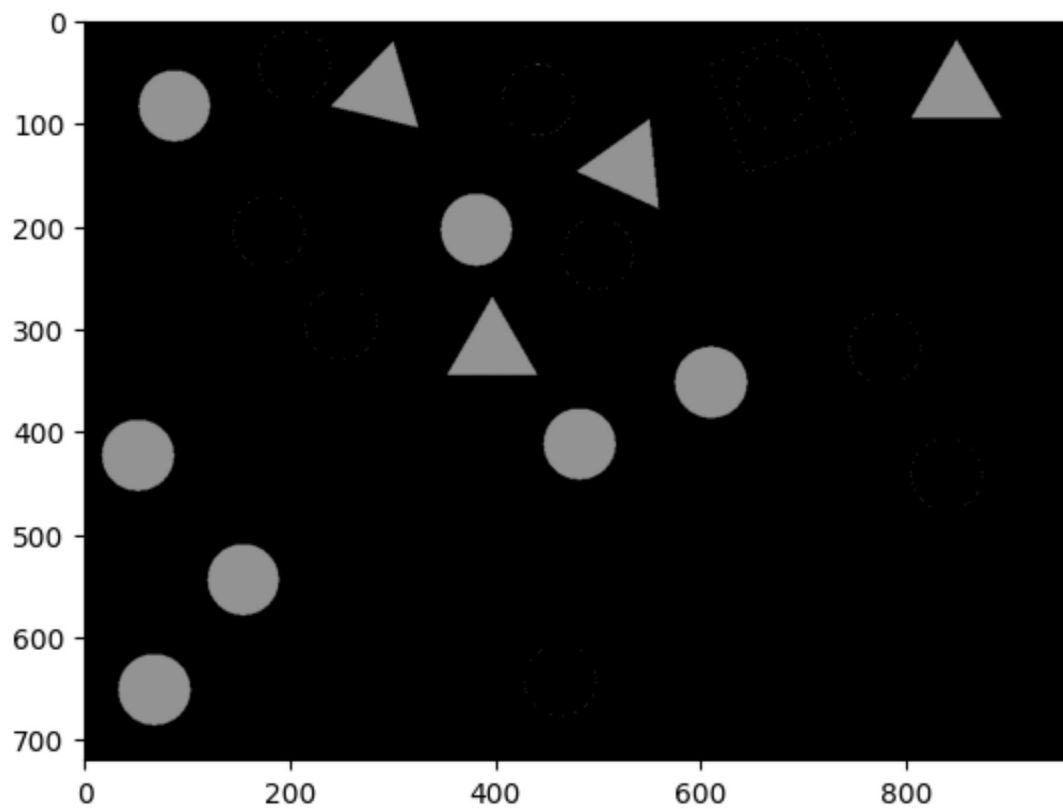
```
Out[ ]:  <matplotlib.image.AxesImage at 0x187e7500b20>
```

```
In [ ]:  ret, red = cv2.threshold(img, 140, 255, cv2.THRESH_TOZERO)
         red = red - green
         plt.imshow(red, cmap= 'gray')
```

Out[ ]: <matplotlib.image.AxesImage at 0x187e0a02ca0>

```python
In [ ]:  # Recolor image
         red_gray = cv2.cvtColor(red, cv2.COLOR_BGR2GRAY)
         green_gray = cv2.cvtColor(green, cv2.COLOR_BGR2GRAY)

         # red
         r = np.uint8(red_gray * 1)
         g = np.uint8(red_gray * 0)
         b = np.uint8(red_gray * 0)
         red_rgb = cv2.merge([r, g, b])

         # green
         r = np.uint8(green_gray * 0)
         g = np.uint8(green_gray * 1)
         b = np.uint8(green_gray * 0)
         green_rgb = cv2.merge([r, g, b])

         result = red_rgb + green_rgb
         # result_rgb = cv2.cvtColor(result, cv2.COLOR_RGB2BGR)


         plt.imshow(result)
```

Out[ ]: <matplotlib.image.AxesImage at 0x187e08f49a0>