

```
In [1]:  
import cv2 as cv  
import numpy as np  
import matplotlib.pyplot as plt  
import os  
%matplotlib inline
```

```
In [2]:  
def load(path):  
    img=cv.imread(path)  
    img=cv.cvtColor(img,cv.COLOR_BGR2RGB)  
    return img
```

```
In [3]:  
def display(img,cmap="gray"):  
    fig=plt.figure(figsize=(12,16))  
    ax=fig.add_subplot()  
    ax.imshow(img,cmap="gray")
```

```
In [4]:  
img=load("/Users/mehradhq/Computer_Vision/Research_2/dataset/train/Prohibition_Signs/17.jpeg")  
display(img)
```



In [5]:

```
#python built-in blur with kernel size 11*11
img_blur=cv.blur(img,(11,11))
display(img_blur)
```



In [6]:

```
#gaussian blur with standard deviation 10 and kernel size 11*11
img_gaussian_blur=cv.GaussianBlur(img,(11,11),10)
display(img_gaussian_blur)
```



In [7]:

```
#median blur with kernel size 11*11
img_median=cv.medianBlur(img,11)
display(img_median)
```



In [8]:

```
#bilateral filter with distance 9 and sigma color 75 and sigma space 75
img_bilateral=cv.bilateralFilter(img,9,75,75)
display(img_bilateral)
```



In [9]:

```
#morphological filter opening with kernel size 11*11  
img_opening=cv.morphologyEx(img, cv.MORPH_OPEN,(11,11))  
display(img_opening)
```



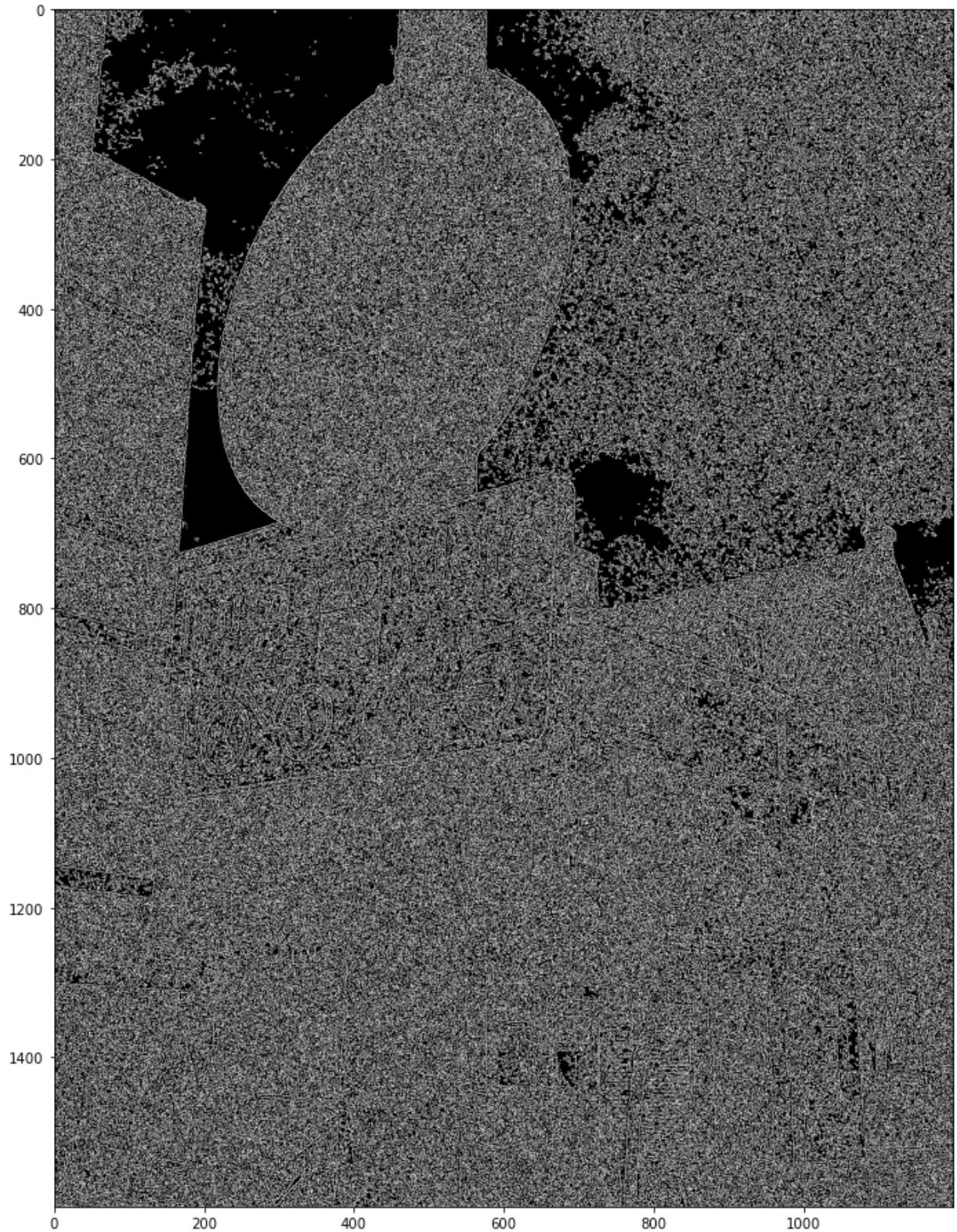
In [10]:

```
#morphological filter closing with kernel size 11*11
img_closing=cv.morphologyEx(img, cv.MORPH_CLOSE,(11,11))
display(img_closing)
```



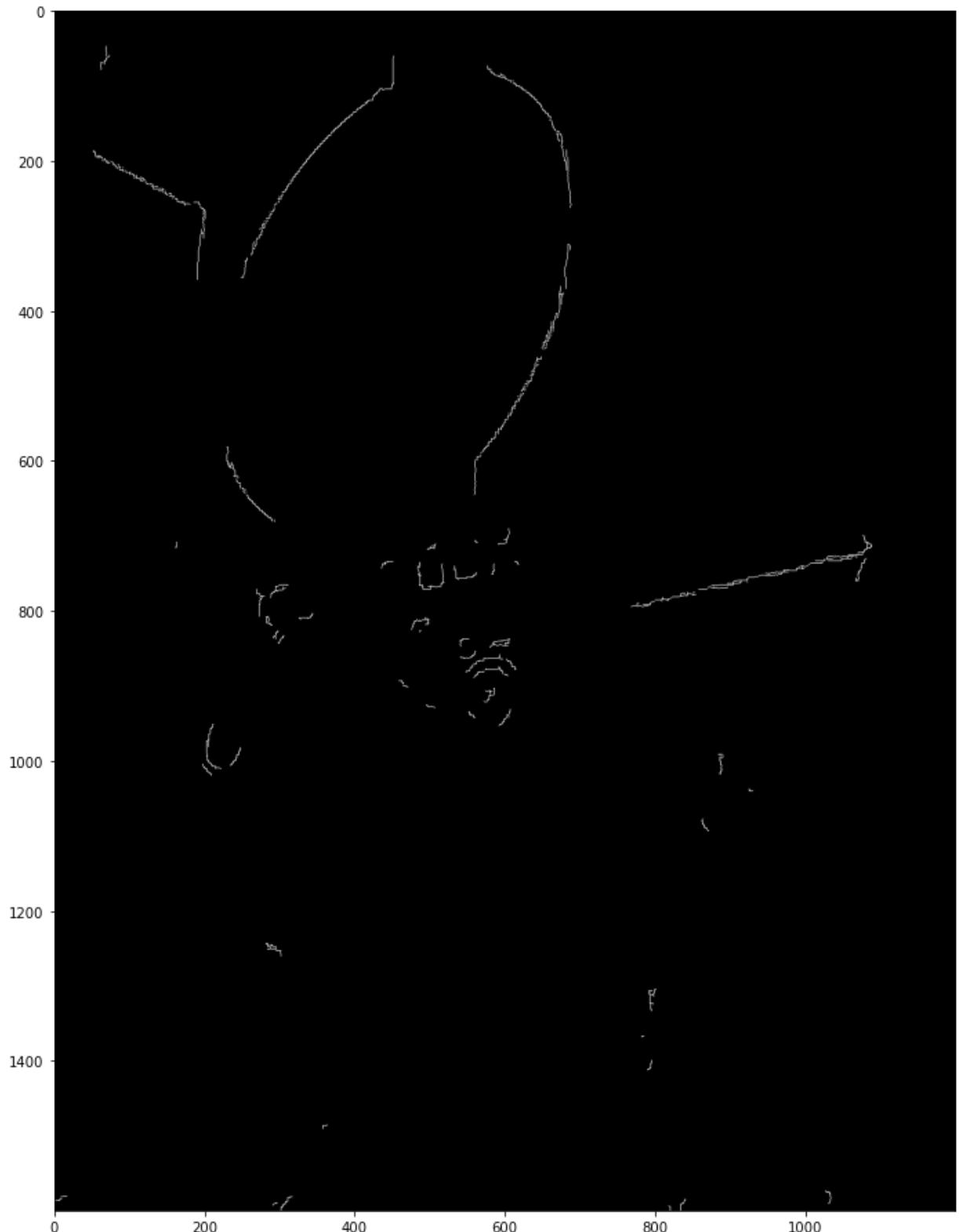
In [11]:

```
#now applying canny edge operator on the initial image
med_val=np.median(img)
lower=int(max(0,0.7*med_val))
upper=int(min(255, 1.3*med_val))
edges=cv.Canny(img,threshold1=lower,threshold2=upper)
display(edges)
```



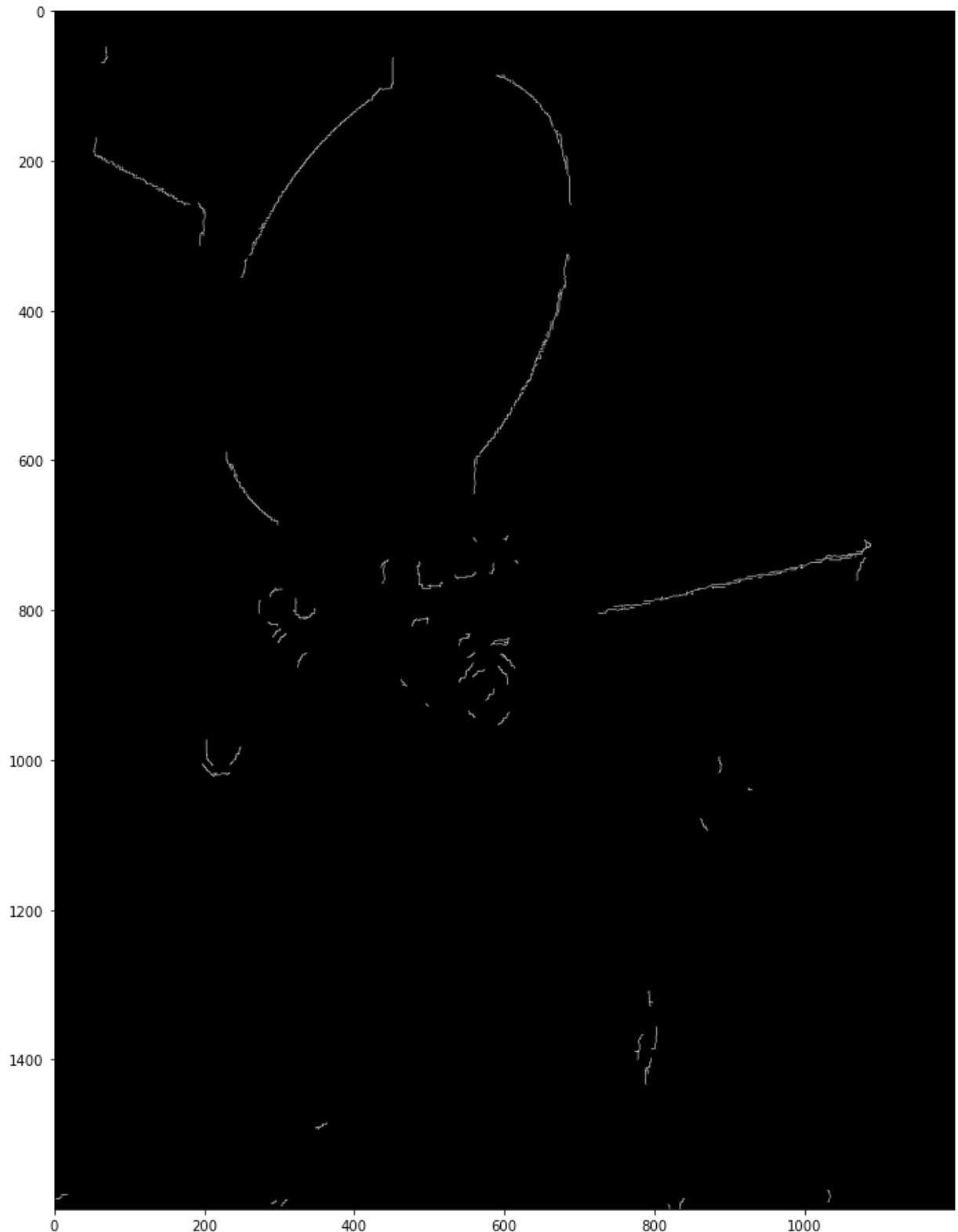
In [12]:

```
#now applying canny edge operator on the blurring image (python built-in)
med_val=np.median(img.blur)
lower=int(max(0,0.7*med_val))
upper=int(min(255, 1.3*med_val))
edges.blur=cv.Canny(img.blur,threshold1=lower,threshold2=upper)
display(edges.blur)
```



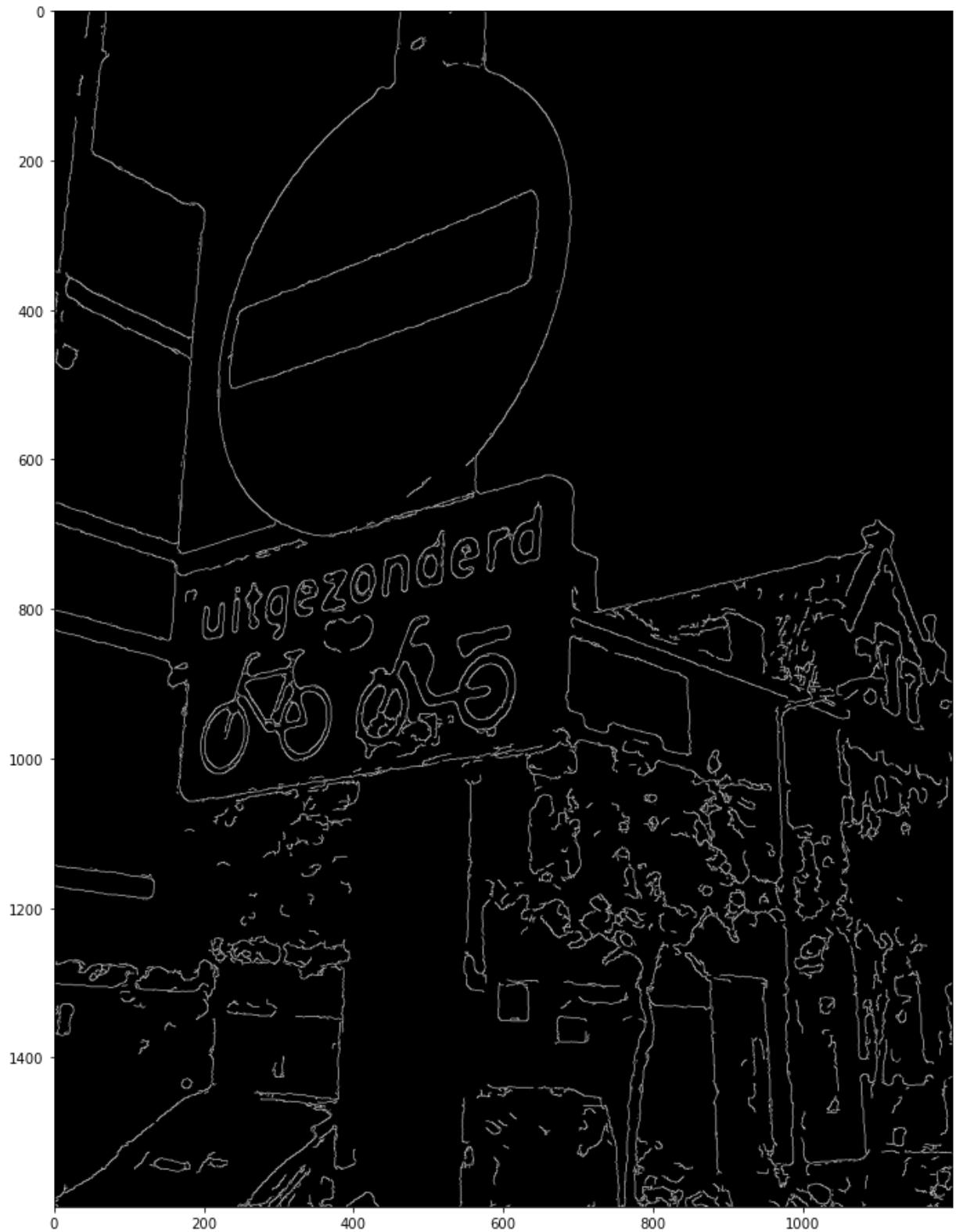
In [13]:

```
#now applying canny edge operator on the gaussian blurred image
med_val=np.median(img_gaussian_blur)
lower=int(max(0,0.7*med_val))
upper=int(min(255, 1.3*med_val))
edges_gaussian_blur=cv.Canny(img_gaussian_blur,threshold1=lower,threshold2=upper)
display(edges_gaussian_blur)
```



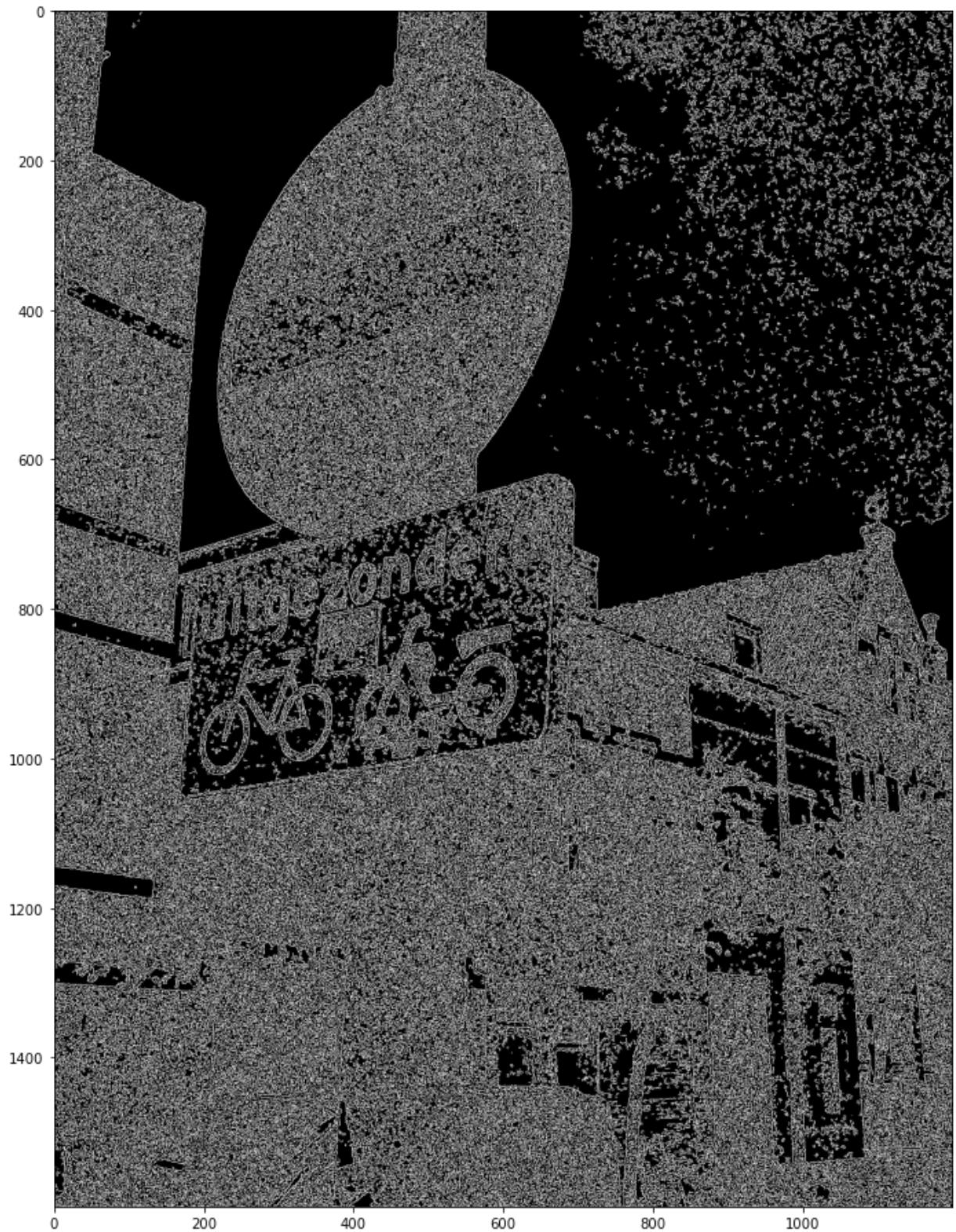
In [14]:

```
#now applying canny edge operator on the median blurred image.  
med_val=np.median(img_median)  
lower=int(max(0,0.7*med_val))  
upper=int(min(255, 1.3*med_val))  
edges_median=cv.Canny(img_median,threshold1=lower,threshold2=upper)  
display(edges_median)
```



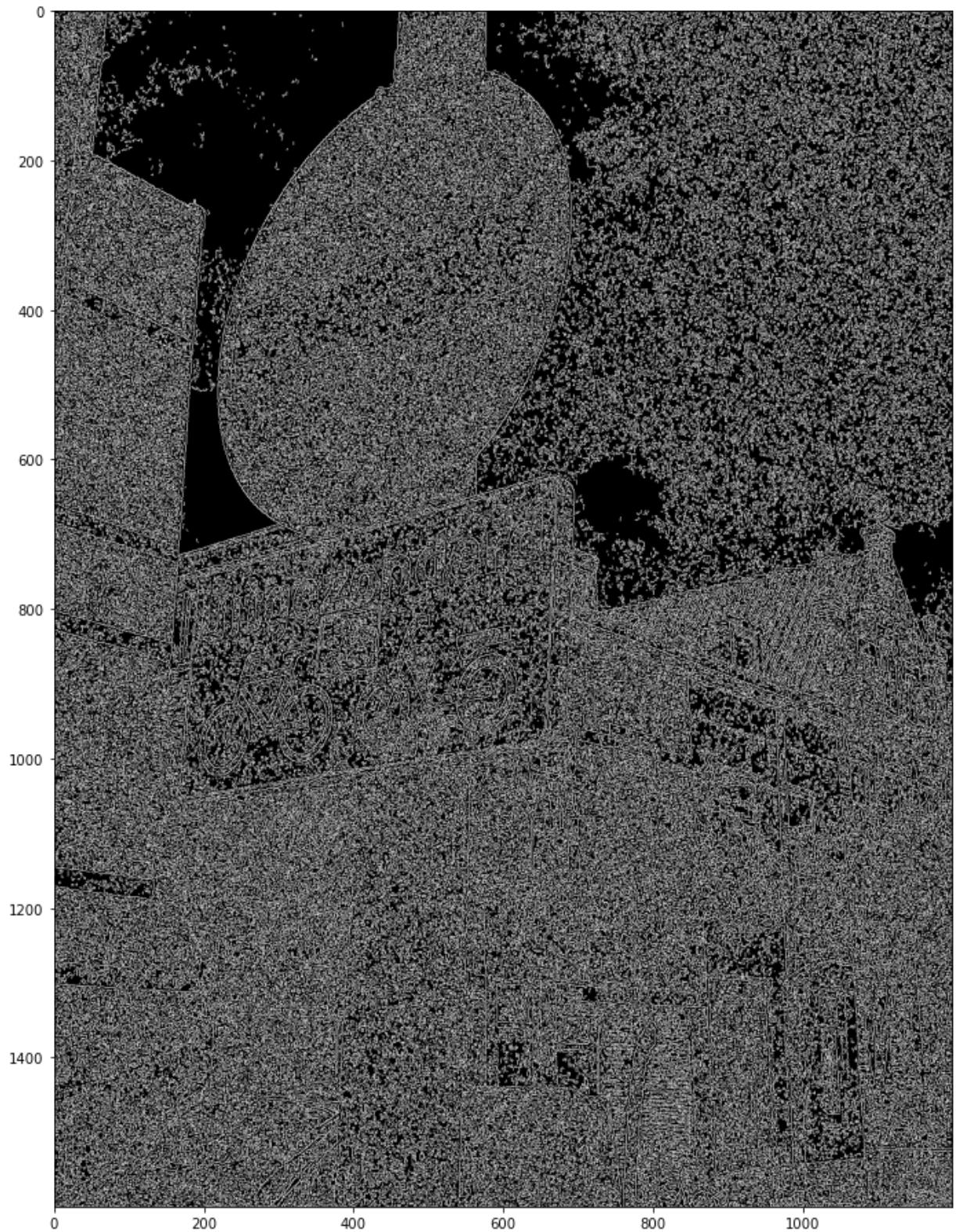
In [15]:

```
#now applying canny edge operator on the bilateral filtered image.  
med_val=np.median(img_bilateral)  
lower=int(max(0,0.7*med_val))  
upper=int(min(255, 1.3*med_val))  
edges_bilateral=cv.Canny(img_bilateral,threshold1=lower,threshold2=upper)  
display(edges_bilateral)
```



In [16]:

```
#now applying canny edge operator on the morphological filtered image (opening)
med_val=np.median(img_opening)
lower=int(max(0,0.7*med_val))
upper=int(min(255, 1.3*med_val))
edges_opening=cv.Canny(img_opening,threshold1=lower,threshold2=upper)
display(edges_opening)
```



In [17]:

```
#now applying canny edge operator on the morphological filtered image (closing)
med_val=np.median(img_closing)
lower=int(max(0,0.7*med_val))
upper=int(min(255, 1.3*med_val))
edges_closing=cv.Canny(img_closing,threshold1=lower,threshold2=upper)
display(edges_closing)
```

