Question 1:

Linear : $T[a_1 x_1(n) + a_2 x_2(n)] = a_1 T[x_1(n)] + a_2 T[x_2(n)]$

Time invariant : $T[x(n-n_0)] = y(n-n_0)$

Since we can express $x(n)$ as

$$x(n) = \sum_{n_0=-\infty}^{\infty} x(n_0)\,\delta(n-n_0)$$

Then,

$$T[x(n)] = T\left[\sum_{n_0=-\infty}^{\infty} x(n_0)\,\delta(n-n_0)\right]$$

$$= \sum_{n_0=-\infty}^{\infty} x(n_0)\, T[\delta(n-n_0)]$$

$$= \sum_{n_0=-\infty}^{\infty} x(n_0)\, h(n-n_0)$$ # by def of $T[\delta(n)] = h(n)$

$$= x(n) * h(n)$$

Question 2:

Let m and n be the highest degree for the polynomial $u(x)$ and $v(x)$ respectively, where $m, n \in \mathbb{N}$

Then, we can expressed $u(x)$ and $v(x)$ as vector u and v s.t

$$u = [u_0 \ u_1 \ \ldots\ldots \ u_{\max(m,n)-1}]$$

$$v = [v_0 \ v_1 \ \ldots\ldots \ v_{\max(m,n)-1}] \, ,$$

where $\{u_0, u_1, \ldots, u_{\max(m,n)-1}\}$ and $\{v_0, v_1, \ldots, v_{\max(m,n)-1}\} \in \mathbb{N}$

Then, the output of $u * v$ is defined as

$$z_i = \sum_{i=0}^{\max(m,n)-1} u_i v_{\max(m,n)-1-i}$$

The output vector is the coefficient of the multiplication of $u(x) = \sum_{i=0}^{m-1} u_i x^i$ and $v(x) = \sum_{i=0}^{n-1} v_i x^i$. Therefore, the convolution of u and v is equivalent to the multiplication of their correspondong polynomical.

Question 3:

Assume that we have two sets of coordonates : (x,y) and (u,v) where (u,v) is the coordinate obtained by rotating the coordinate (x,y) by an angle θ

$$\begin{cases} u = x\cos\theta - y\sin\theta \\ v = x\sin\theta + y\cos\theta \end{cases}$$

To obtain the 1st differential order:

$$\frac{du}{dx} = \cos\theta \qquad \frac{du}{dy} = -\sin\theta$$

$$\frac{dv}{dx} = \sin\theta \qquad \frac{dv}{dy} = \cos\theta$$

$$\therefore \frac{d}{dx} = \cos\theta + \sin\theta$$

$$\frac{d}{dy} = -\sin\theta + \cos\theta$$

To find $\frac{d^2f}{dx^2}$,

let $\frac{d^2f}{dx^2} = \frac{d}{dx}\left(\frac{df}{dx}\right)$ and

$$g = \frac{df}{dx} = \frac{du}{dx}\cdot\frac{df}{du} + \frac{dv}{dx}\cdot\frac{df}{dv}$$

$$= \cos\theta\cdot\frac{df}{du} + \sin\theta\cdot\frac{df}{du} \quad \text{\# as calculated in the 1st order differential} \quad -①$$

Then, $\frac{d^2f}{dx^2} = \frac{dg}{dx}$

$$= \frac{du}{dx}\cdot\frac{dg}{du} + \frac{dv}{dx}\cdot\frac{dg}{dv}$$

$$= \cos\theta\frac{dg}{du} + \sin\theta\cdot\frac{dg}{dv}$$

$$= \cos\theta\frac{d}{du}\left(\cos\theta\cdot\frac{df}{du} + \sin\theta\cdot\frac{df}{dv}\right) + \sin\theta\frac{d}{dv}\left(\cos\theta\cdot\frac{df}{du} + \sin\theta\cdot\frac{df}{dv}\right) \quad \text{\# by eq. ①}$$

$$= \cos^2\theta\frac{d^2f}{du^2} + \sin^2\theta\frac{d^2f}{dv^2} + 2\cos\theta\sin\theta\frac{d^2f}{dudv} \quad -②$$

To find $\frac{d^2f}{dy^2}$,

Let $h = \frac{df}{dy} = \frac{du}{dy} \cdot \frac{df}{du} + \frac{dv}{dy} \cdot \frac{df}{dv}$

$= -\sin\theta \frac{df}{du} + \cos\theta \frac{df}{dv}$ # by 1st order differential — ③

$$\frac{d^2f}{dy^2} = \frac{dh}{dy}$$

$$= \frac{du}{dy} \cdot \frac{df}{du} + \frac{dv}{dy} \cdot \frac{df}{dv}$$

$$= -\sin\theta \frac{d}{du}\left(-\sin\theta \frac{df}{du} + \cos\theta \frac{df}{dv}\right) +$$

$$\cos\theta \frac{d}{dv}\left(-\sin\theta \frac{df}{du} + \cos\theta \frac{df}{dv}\right)$$ # by eq ③

$$= \sin^2\theta \frac{d^2f}{du^2} + \cos^2\theta \frac{d^2f}{dv^2} - 2\sin\theta\cos\theta \frac{d^2f}{dudv}$$ — ④

Then,

$$\frac{d^2f}{dx^2} + \frac{d^2f}{dy^2} = ② + ④$$

$$= \cos^2\theta \frac{d^2f}{du^2} + \sin^2\theta \frac{d^2f}{dv^2} +$$

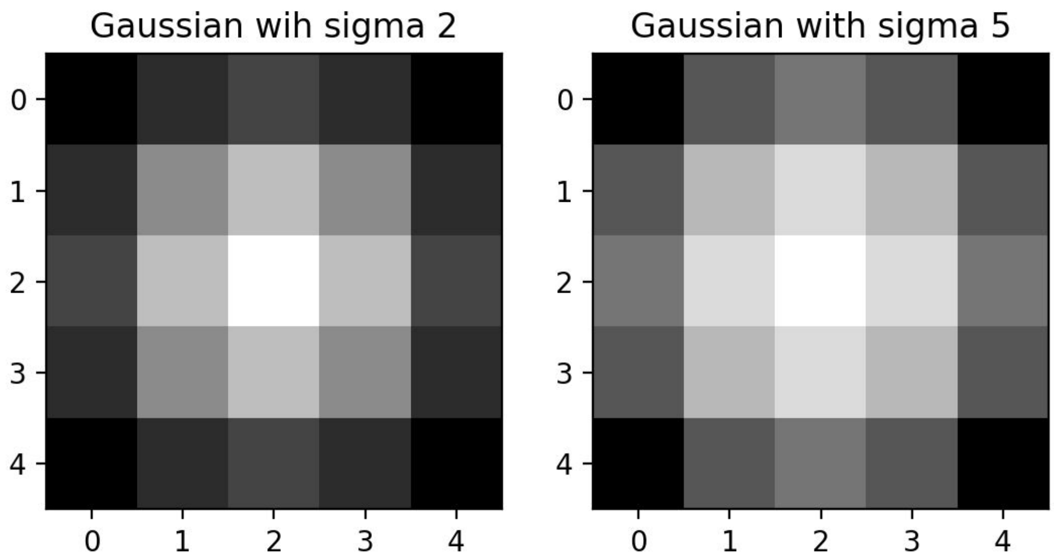
$$\sin^2\theta \frac{d^2f}{du^2} + \cos^2\theta \frac{d^2f}{dv^2}$$

$$= (\cos^2\theta + \sin^2\theta) \frac{d^2f}{du^2} + (\sin^2\theta + \cos^2\theta) \frac{d^2f}{dv^2}$$

$= 1 \cdot \frac{d^2f}{du^2} + 1 \cdot \frac{d^2f}{dv^2}$ # by trigo identity $\sin^2\theta + \cos^2\theta = 1$

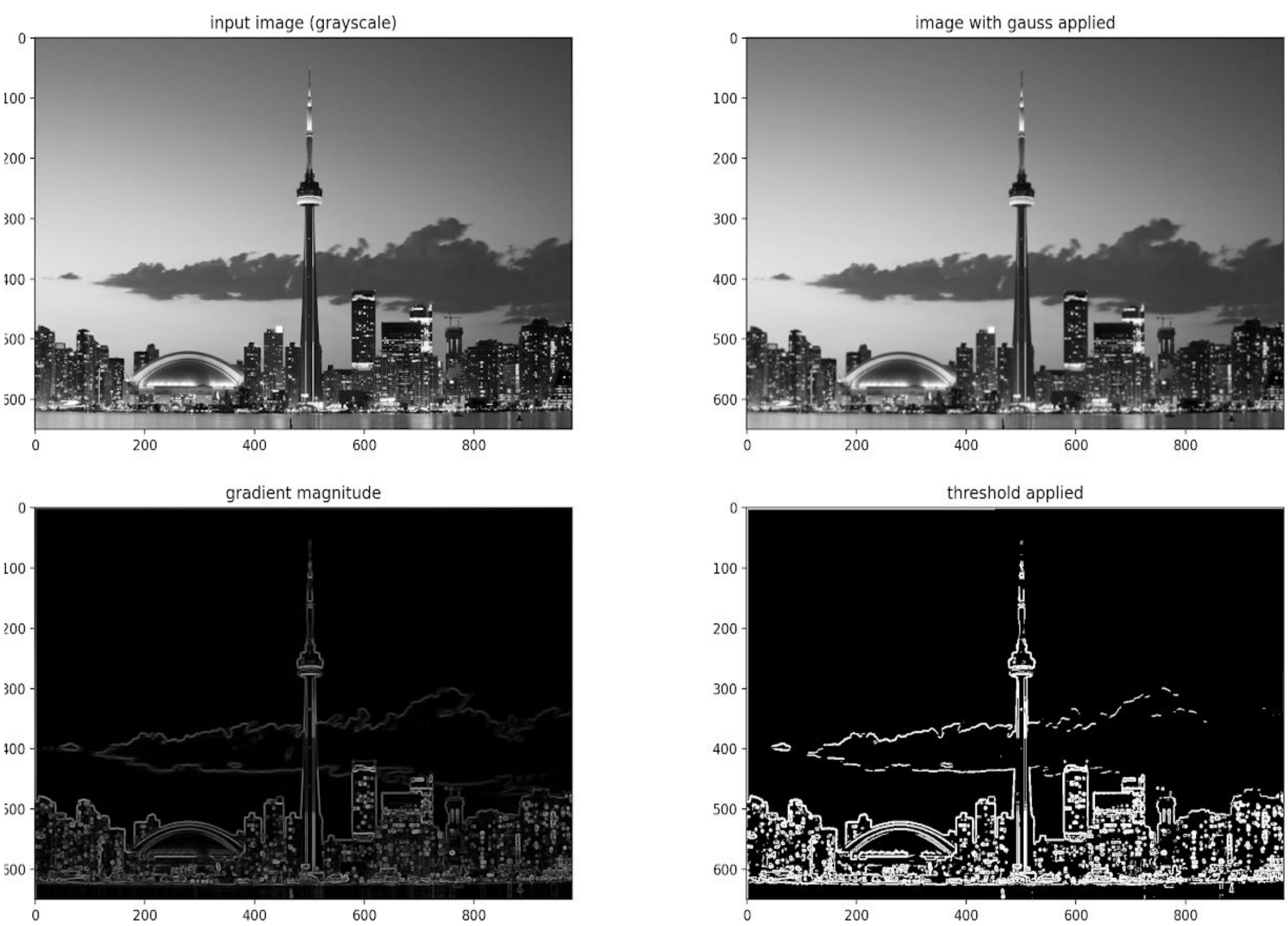
$$= \frac{d^2f}{du^2} + \frac{d^2f}{dv^2}$$ ✗

Question 4:
Refer q4_q5_q6.py for the code

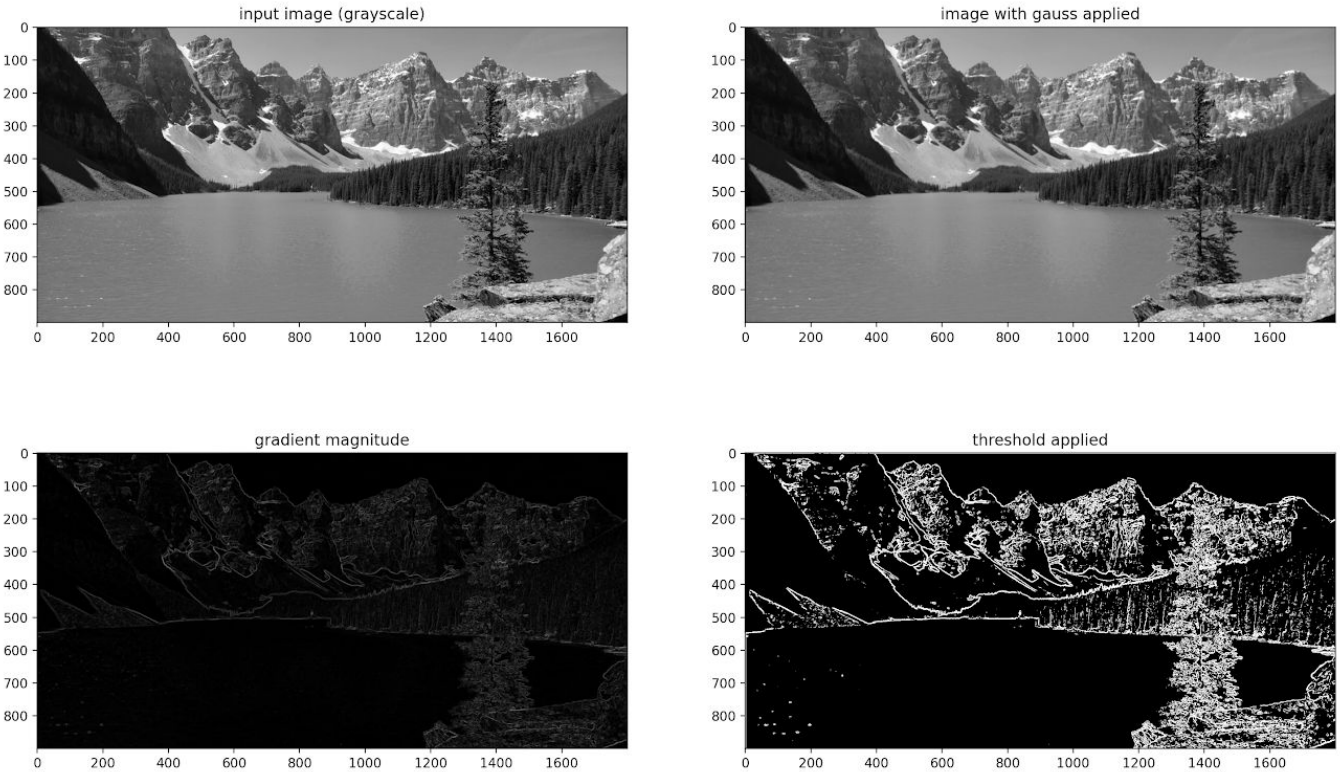


This is the result for the coefficient generated from question 4 part 1. The outputs have a filter size of 5 x 5 with sigma set to 2 and 3 respectively. As a result, the filter with larger sigma value has a larger value for most of the pixels, which correspond to brighter pixels.

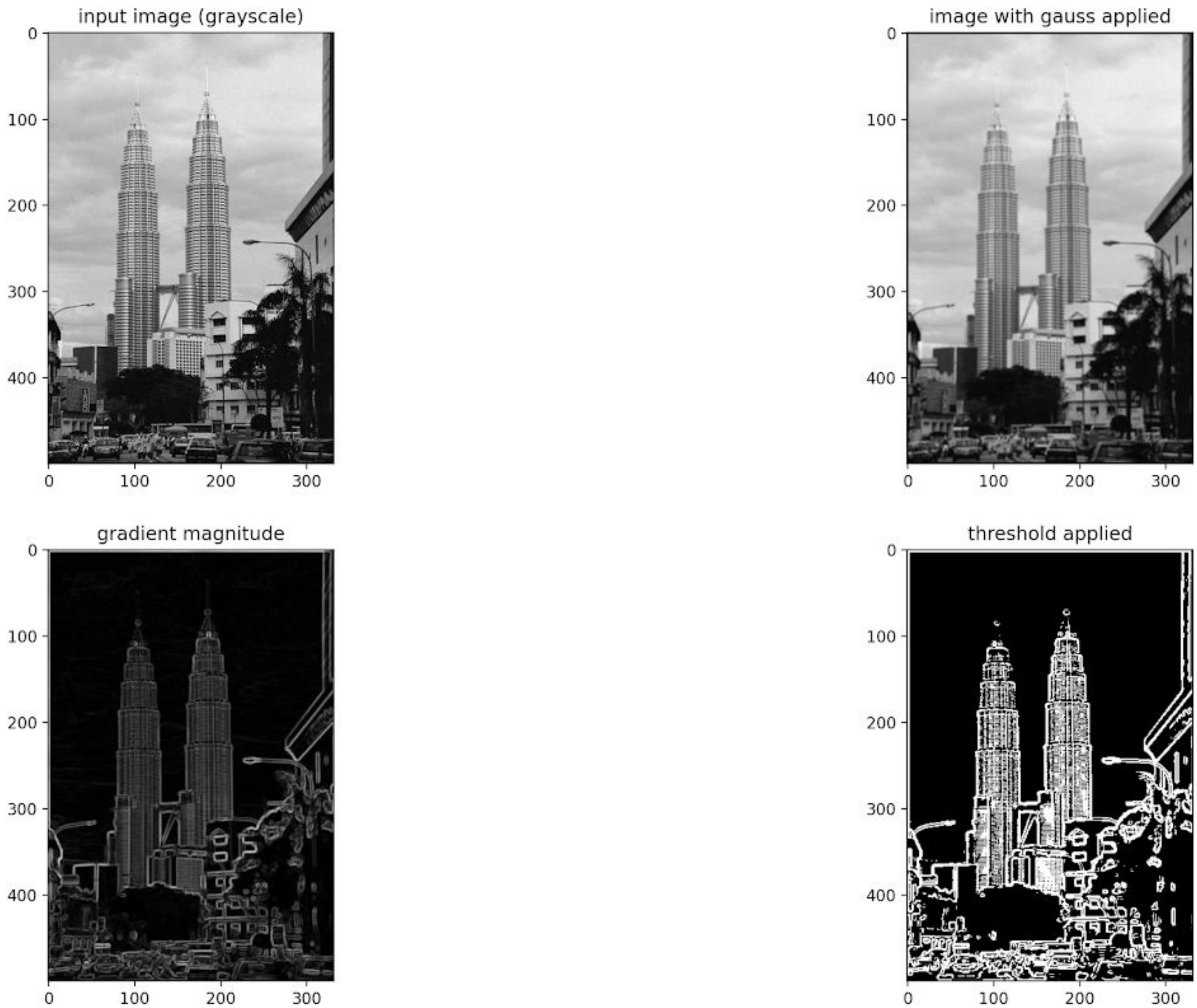
Output for Q4_image_1:



Output for Q4_image_2:



Output for own image:

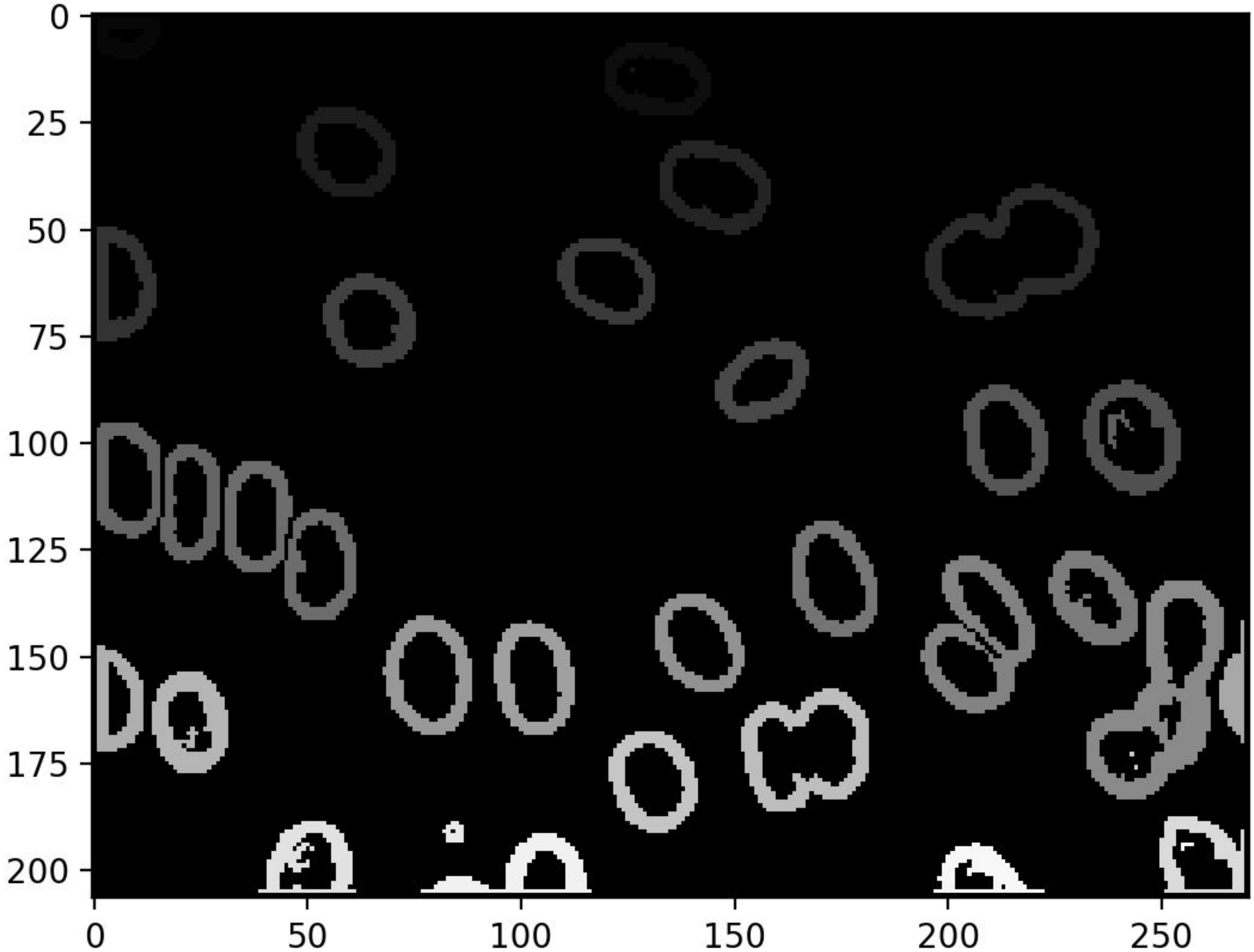


Discussion:
All the images are first converted to grayscale (top left corner). Then, the images were processed using the Gaussian filter with the size of 3 x 3 and sigma of 3, which produces a “softer” image than the original one (top right corner). After that, the “softer” image is used to calculate the gradient magnitude by applying the Sobel operation. Before applying the Sobel operation, convolution is performed on both filters by flipping each filter matrix horizontally and vertically, then multiply and sum the values with cell value in the image at their corresponding indexes.

This approach works well with images that have high contrast, where the gradient between multiple objects is high, and thus easier to tell each object apart. For example, in the 2nd row output of Q4_image_1, we can see that the part of CN Tower with clouds behind it is disconnected. The output tends to be a bit noisy as you can see the lights of the buildings around CN Tower. However, this issue might be less prominent if we use Gaussian filter with larger size and higher sigma value.

Question 5:
Refer q4_q5_q6.py for the code

Question 6:
Refer q4_q5_q6.py for the code



Estimated cell count: 36

The program struggles with situations where two or more cells are extremely close to each other. This causes the program to treat that group of cells as one entity. We could solve this by applying a higher threshold at step 3 of question 4, which makes the program to assign more pixels with value 0. By doing so, when we start labelling the neighbours around a pixel, we could better differentiate the cells

.