Homework 9

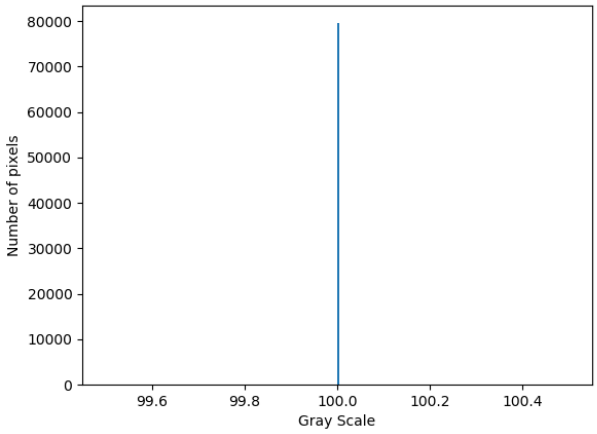
(1) Problem statement

(1) Create an image $g(x,y)$ whose pixels all have the same gray value of 100. Show the image $g(x,y)$.
(2) Generate Gaussian noise $n(x,y)$, with $\mu = 0$, $\sigma^2 = 15$, using methods 1 and 2. Show the noisy image $f(x,y)=g(x,y)+n(x,y)$.
(3) Display the histogram $h(i)$ of $f(x,y)$.
(4) Comment on your results.

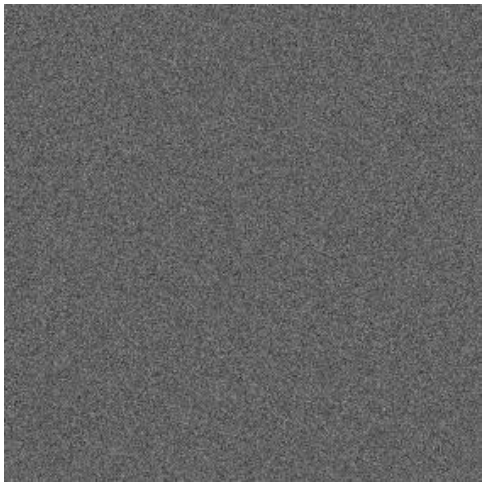
(2.1) Experimental results

Input image g(x,y) of gray value of 100

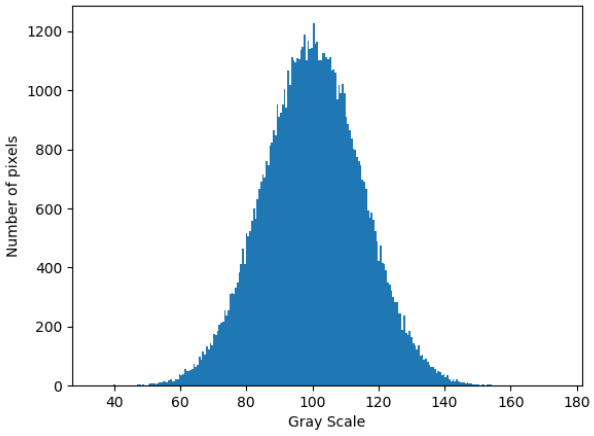
Histogram of g(x,y)



Noisy image f(x,y)



Histogram of f(x,y)



(2.2) Source code

(i) method1: generation of zero menu Gaussian noise

```python
# HW9-method1 (Generation of zero mean Gaussian noise)

from PIL import Image
import numpy as np
import matplotlib.pyplot as plt

# 1. Create an image g and the same gray values of 100
g = Image.open("g.png").convert("L")
for i in range(g.size[0]):
    for j in range(g.size[1]):
        g.putpixel((i, j), 100)

g.save("g.png")
g.show()

# Save image g histogram
a = np.array(g)
plt.hist(a.ravel(), bins=256)
plt.ylabel('Number of pixels')
plt.xlabel('Gray Scale')
plt.savefig('g_histogram.png')
plt.show()

# 2. Generate Gaussian noise n and Show noisy image f
# Calculate Gaussian noise with menu=0 and variance=15
menu = 0
variance = 15
noise = np.random.normal(menu, variance, g.size)
# Add the noise to the image g
f_array_noise = np.add(np.array(g), noise) # f = g + noise
f = Image.fromarray(f_array_noise)
# Show the noisy image f
f.show()

# 3. Save and display noisy image f histogram
a = np.array(f)
plt.hist(a.ravel(), bins=256)
plt.ylabel('Number of pixels')
plt.xlabel('Gray Scale')
plt.savefig('f_histogram.png')
plt.show()
```