Mohaymen Sameen
627650
Assignment1

Assignment 1A
Code Snippet

```python
import glob
from os import path
import cv2 as cv
from PIL import ImageFilter
from PIL import Image
from matplotlib import pyplot as plt
from skimage.util import random_noise


def noisetype(noise_type, image):
    img = cv.imread(image)

    if noise_type == "gauss":
        gauss = random_noise(img, mode='gaussian', seed=None, clip=True)
        plt.imshow(gauss[..., ::-1]), plt.title('Gaussian')
        plt.savefig('(%s)_gaussian.jpg' % image, dpi=300)
        plt.show()
    elif noise_type == "s&p":
        sp = random_noise(img, mode='s&p', seed=None, clip=True)
        plt.imshow(sp[..., ::-1]), plt.title('Salt & Pepper')
        plt.savefig('(%s)_s&p.jpg' % image, dpi=300)
        plt.show()
    elif noise_type == "poisson":
        poiss = random_noise(img, mode='poisson', seed=None, clip=True)
        plt.imshow(poiss[..., ::-1]), plt.title('Poisson')
        plt.savefig('(%s)_poisson.jpg' % image, dpi=300)
        plt.show()
    elif noise_type == "spectacle":
        spec = random_noise(img, mode='speckle', seed=None, clip=True)
        plt.imshow(spec[..., ::-1]), plt.title('Spectacle')
        plt.savefig('(%s)_spectacle.jpg' % image, dpi=300)
        plt.show()
    else:
        print("invalid noise type")

    return plt


def filter(filter_type):
    images = []
    for imagepath in glob.glob("mohaymen/mountain/*.jpg"):
        images.append(imagepath)
    count = 0
    outpath = "mohaymen/mountain/%s" % filter_type
    for img in images:
        img2 = cv.imread(img)
        if filter_type == "mean":
            new_image = cv.blur(img2, (9, 9))
            plt.imshow(new_image[..., ::-1]), plt.title('Mean')
            plt.savefig(path.join(outpath, "%d_mean.jpg" % count))
            plt.show()
            count += 1
        elif filter_type == "gaussblur":
            new_image = cv.GaussianBlur(img2, (9, 9), 0)
            plt.imshow(new_image[..., ::-1]), plt.title('Gaussian Blur')
            plt.savefig(path.join(outpath, "%d_gaussian_blur.jpg" % count))
            plt.show()
            count += 1
        elif filter_type == "median":
            new_image = cv.medianBlur(img2, 9)
            plt.imshow(new_image[..., ::-1]), plt.title('Median Filter')
            plt.savefig(path.join(outpath, "%d_median_filter.jpg" % count))
            plt.show()
            count += 1
        elif filter_type == "laplacian":
            new_image = cv.Laplacian(img2, cv.CV_64F)
            plt.imshow(img2 + new_image[..., ::-1]), plt.title('Laplacian')
            plt.savefig(path.join(outpath, "%d_laplacian.jpg" % count))
            plt.show()
            count += 1
        elif filter_type == "unsharp":
            img2 = img2[..., ::-1]
            img2 = Image.fromarray(img2.astype('uint8'))
            new_image = img2.filter(ImageFilter.UnsharpMask(radius=2, percent=150))
            plt.imshow(new_image), plt.title('unsharp')
            plt.savefig(path.join(outpath, "%d_unsharp.jpg" % count))
            plt.show()
            count += 1
        else:
            print("invalid filter type")
    return plt


# noisetype("spectacle", "mountain.jpg")
filter("unsharp")
```

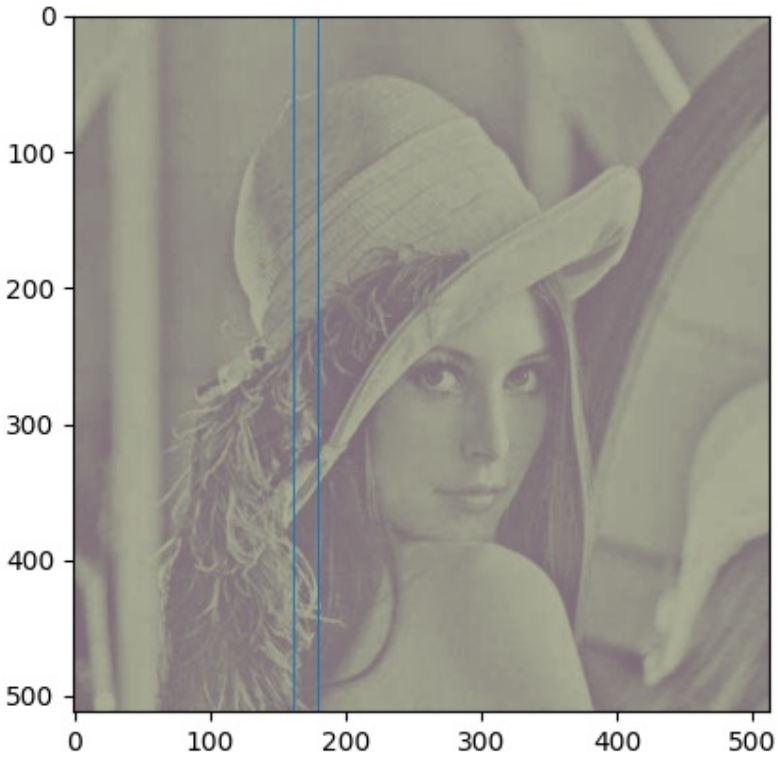
Assignment1B Code Snippet

```python
import cv2 as cv
import numpy as np
from matplotlib import pyplot as plt


def image_channels(image):
    img = cv.imread(image)
    img = img[..., ::-1]
    blue_channel = img[:, :, 0]
    green_channel = img[:, :, 1]
    red_channel = img[:, :, 2]

    blue_img = np.zeros(img.shape)
    green_img = np.zeros(img.shape)
    red_img = np.zeros(img.shape)
    blue_img[:, :, 0] = blue_channel
    green_img[:, :, 1] = green_channel
    red_img[:, :, 2] = red_channel

    plt.imshow(blue_img)
    plt.savefig("mohaymen/mountain_red_img.jpg", dpi=300)
    plt.show()
    return plt


image_channels("mohaymen/mountain/mountain.jpg")
```

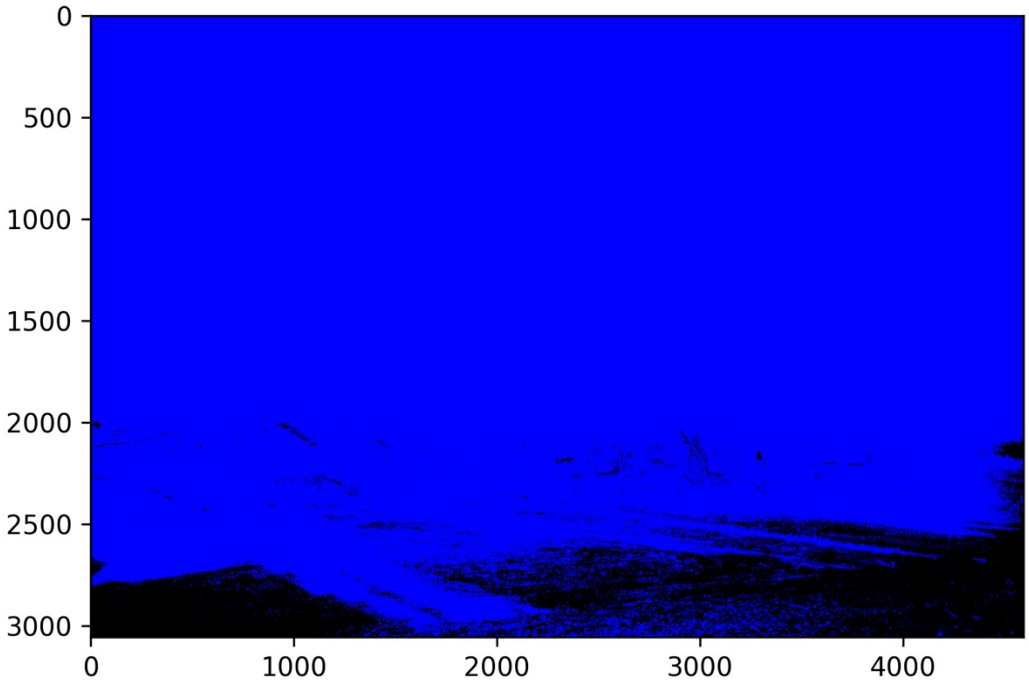
Assignment 1C Code Snippet

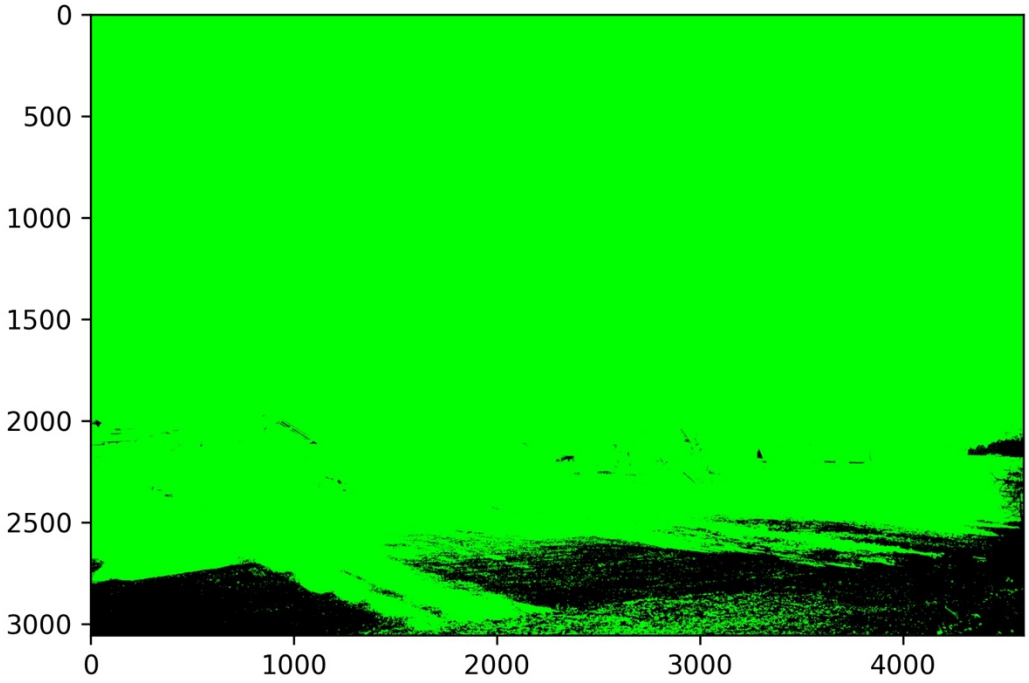
```python
import cv2
from cv2 import COLOR_BGR2LAB, COLOR_RGB2Lab
from matplotlib import pyplot as plt


image = cv2.imread("mohaymen/Low contrast Image.jpg")
lab = cv2.cvtColor(image, COLOR_BGR2LAB)
l, a, b = cv2.split(lab)
plt.hist(l.flat, bins=20000, range=(0, 255))
newL = cv2.equalizeHist(l)
merge = cv2.merge([newL, a, b])
convert = cv2.cvtColor(merge, COLOR_RGB2Lab)

clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8,8))
cl1 = clahe.apply(l)
newimg = cv2.merge([cl1, a, b])
convert2 = cv2.cvtColor(newimg, COLOR_RGB2Lab)
plt.imshow(convert2)
plt.show()
```

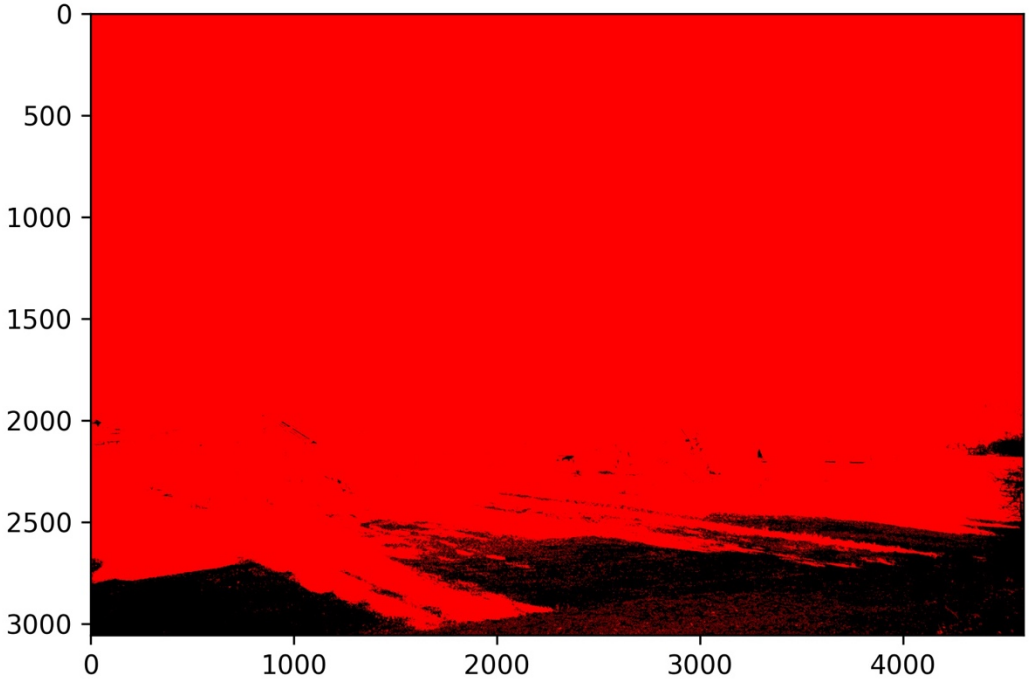
Assignment 1C Image output:



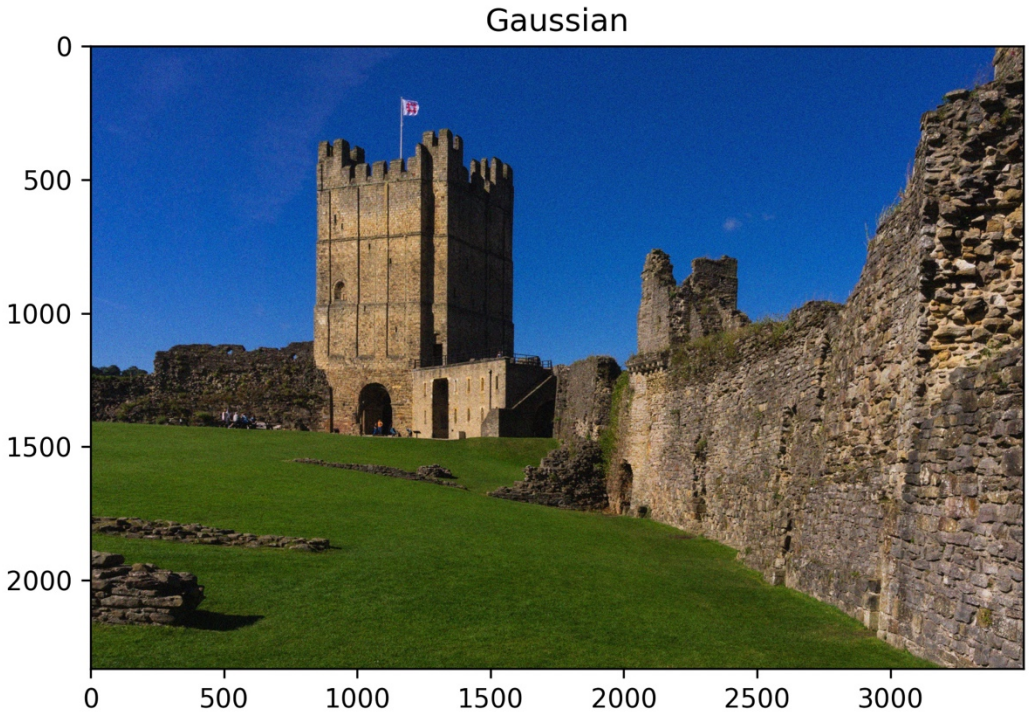
Assignment 1B output images (Mountain Image):

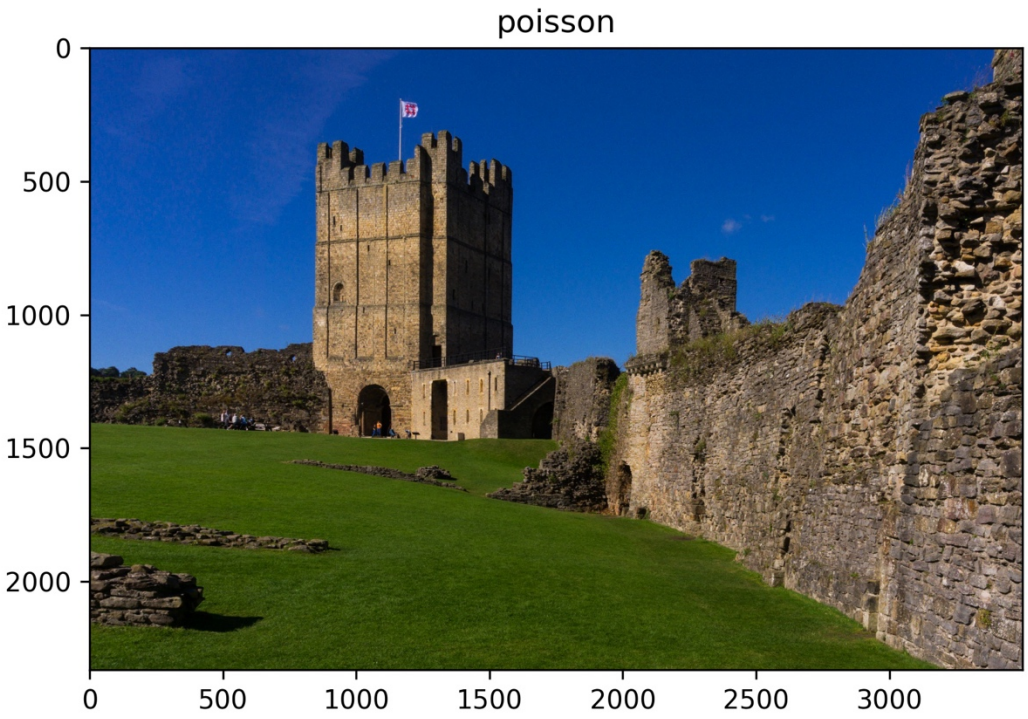



0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

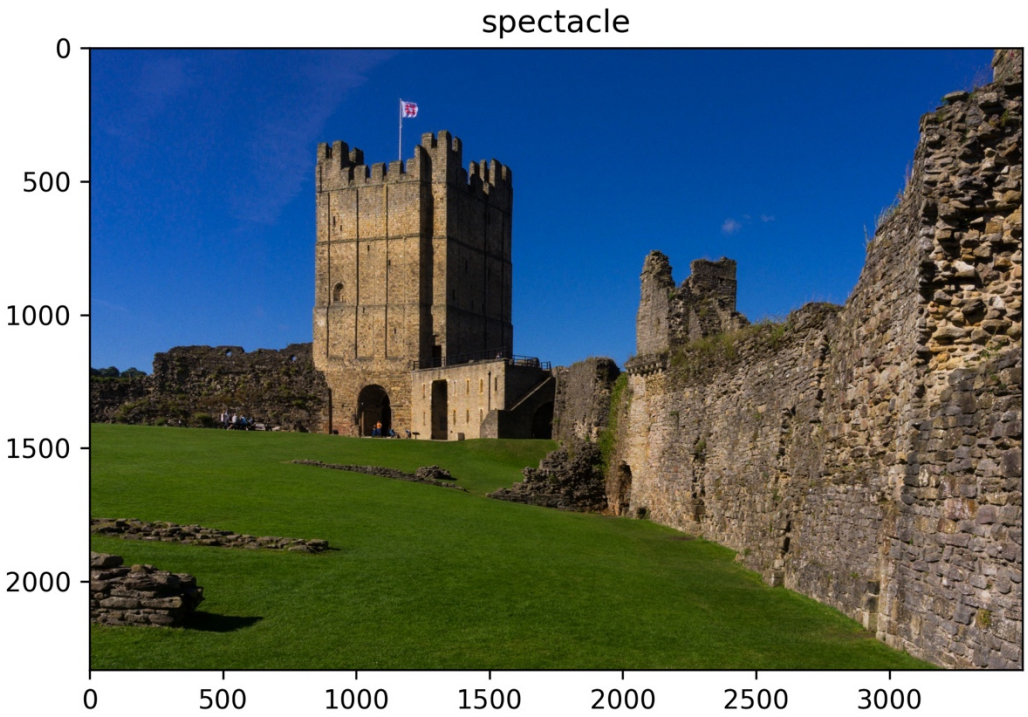

0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

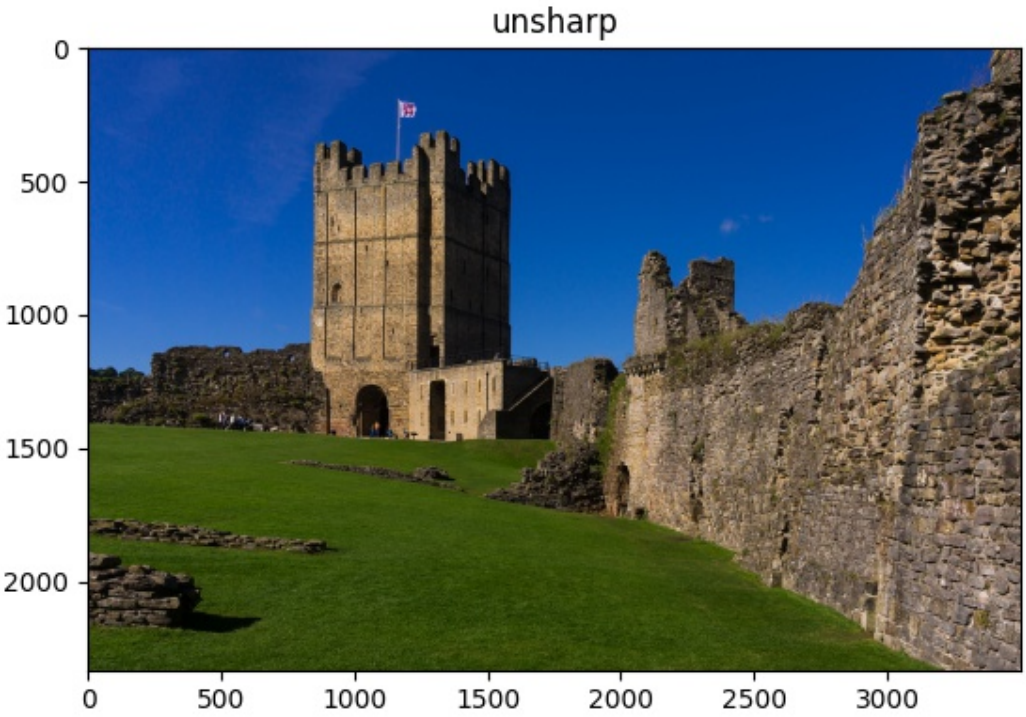
Assignment 1A output images: Original Images are in order (Castle, Cloud, Field, Mountain)

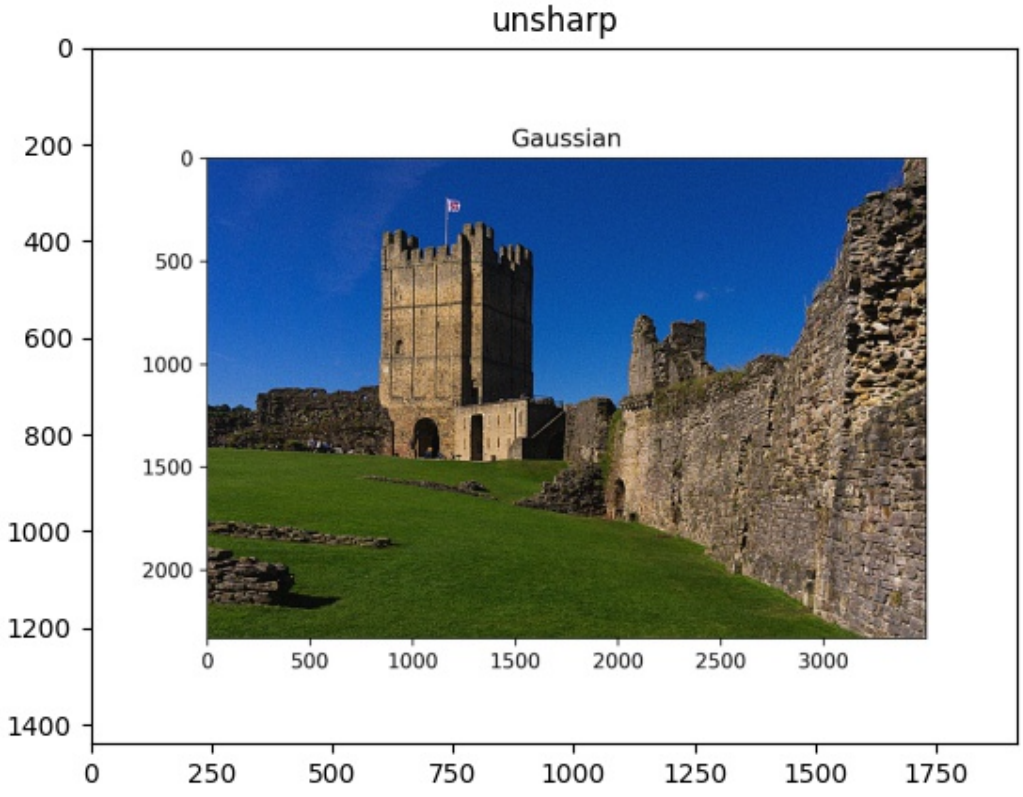


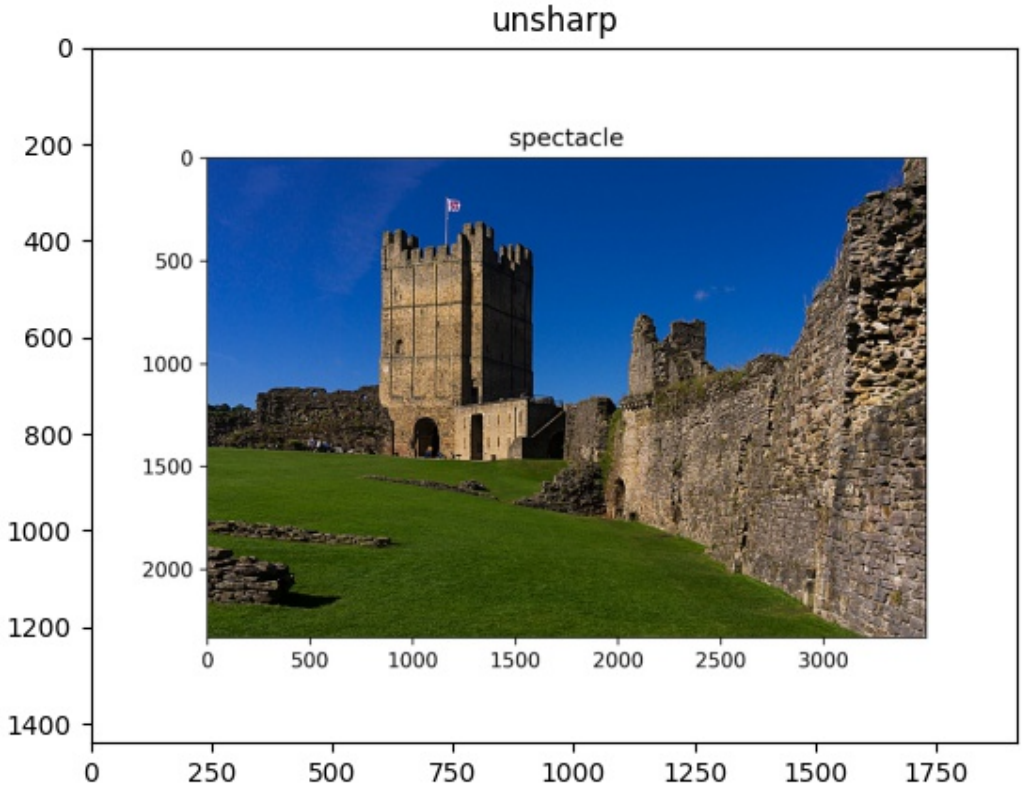

poisson
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

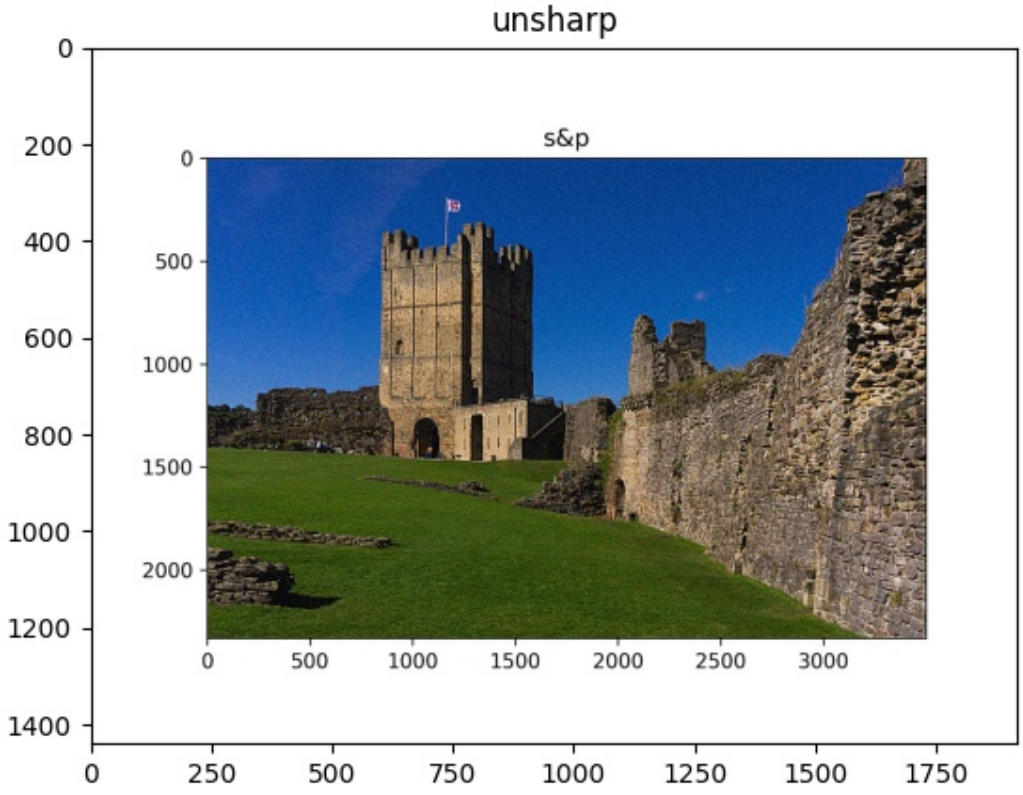

s&p
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

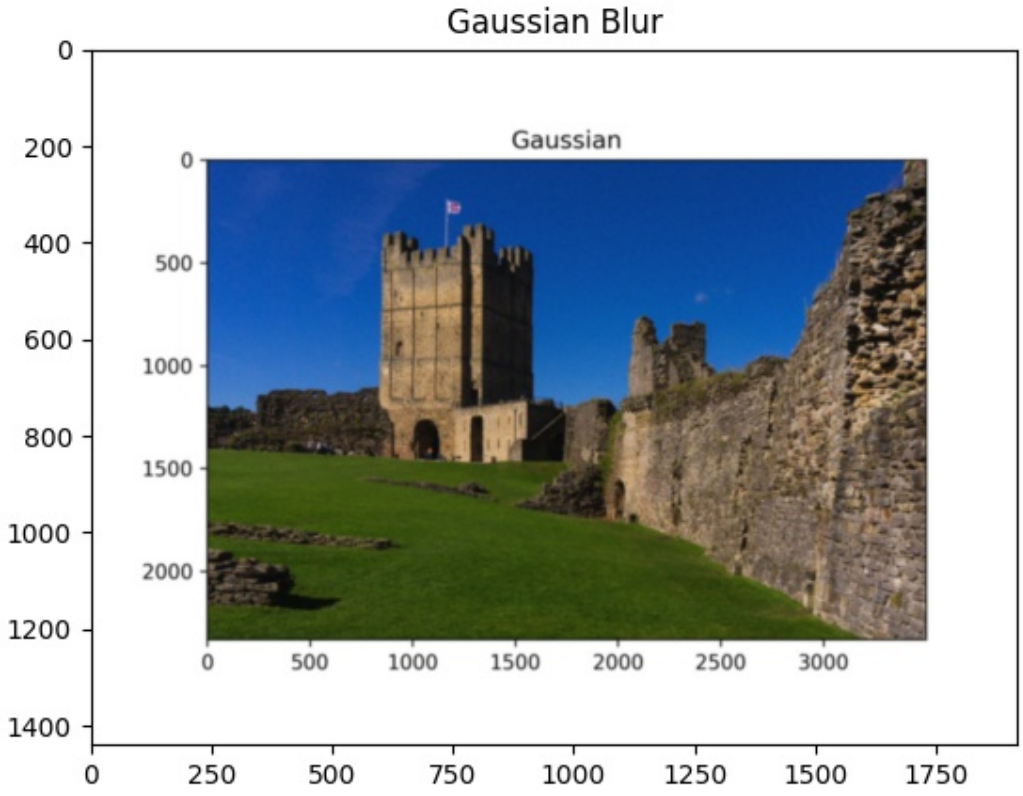

spectacle
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

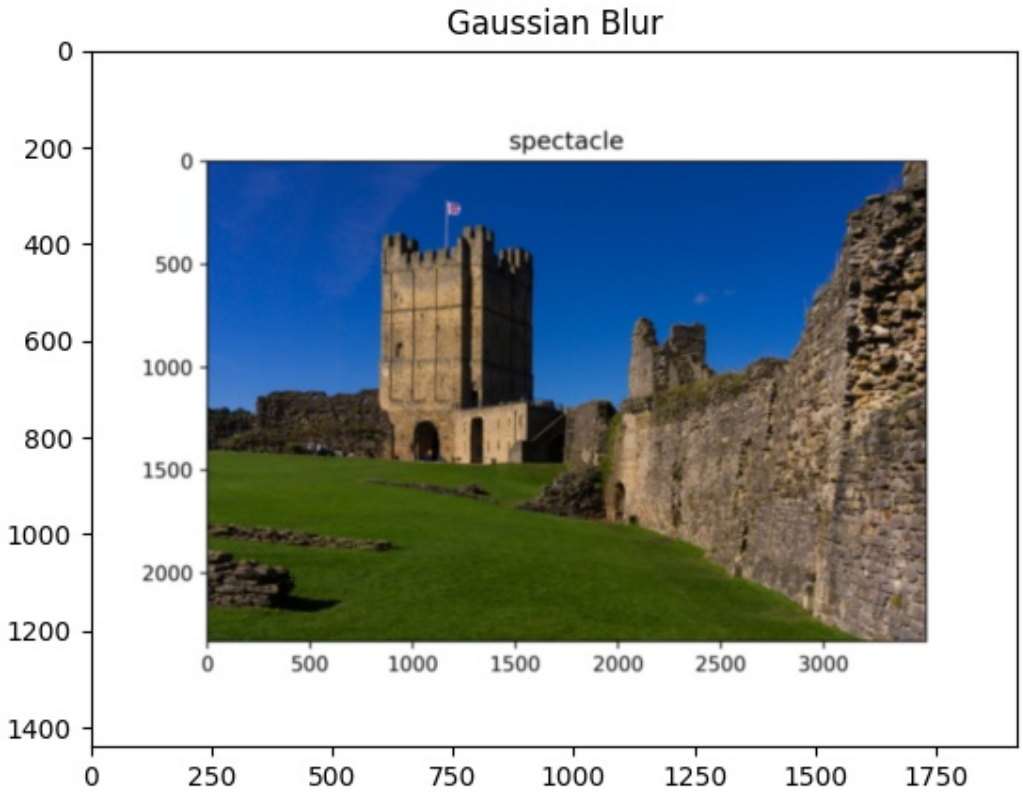

unsharp
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

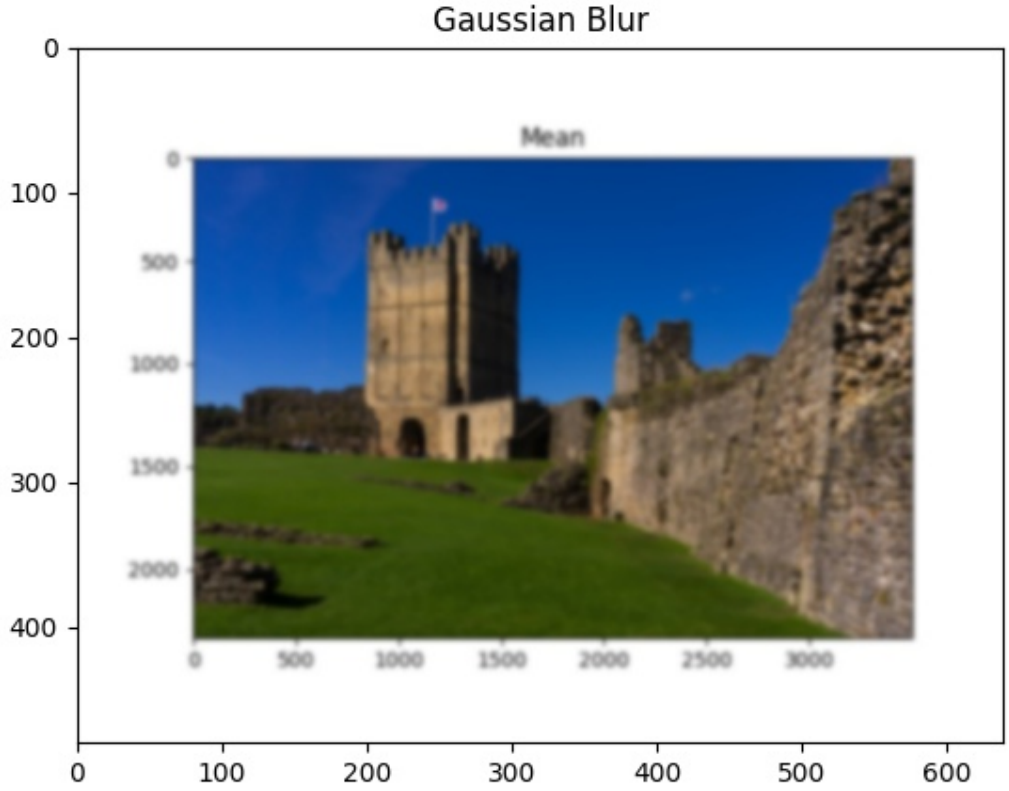

unsharp
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

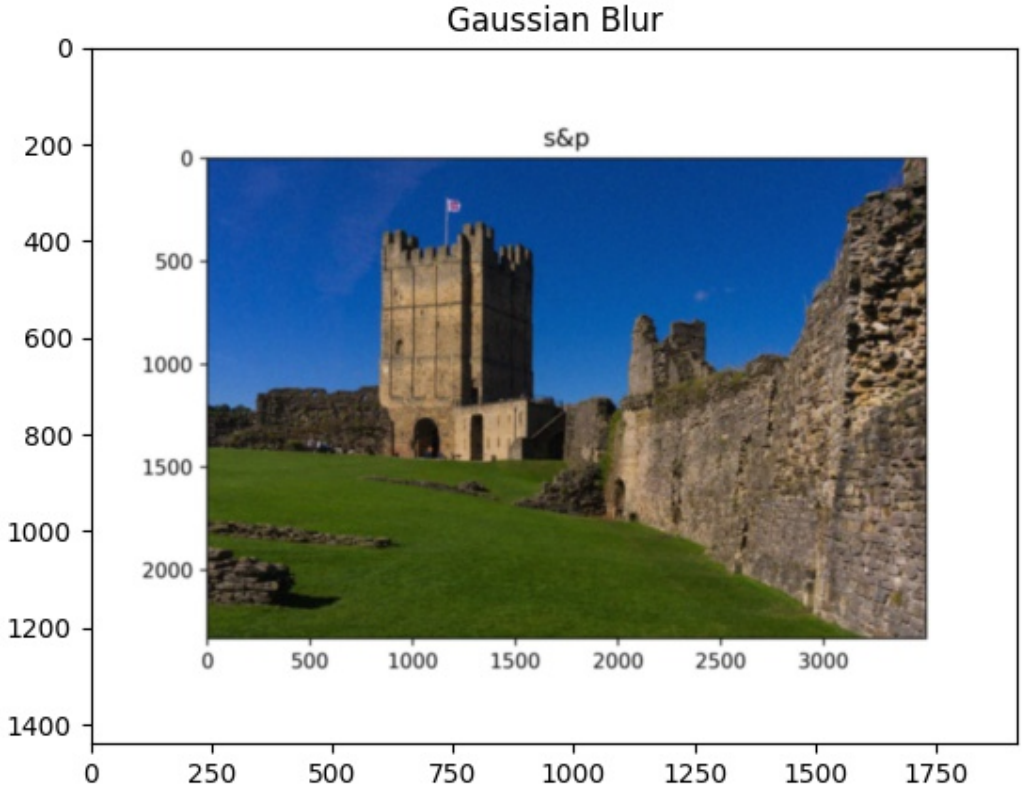

unsharp
spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000

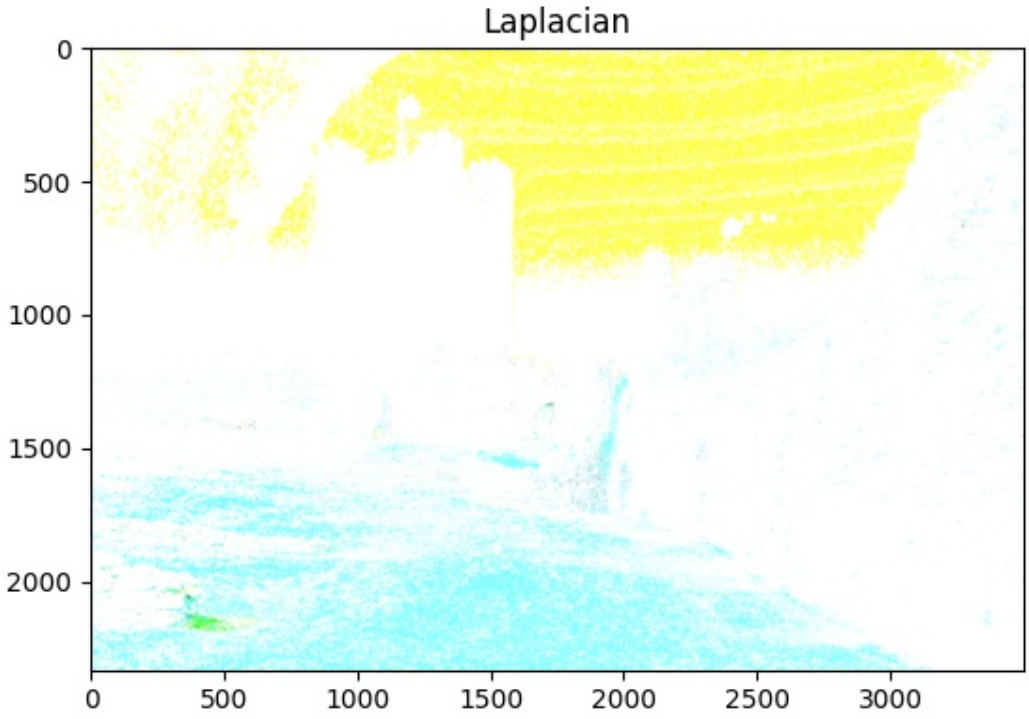

unsharp
s&p
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000

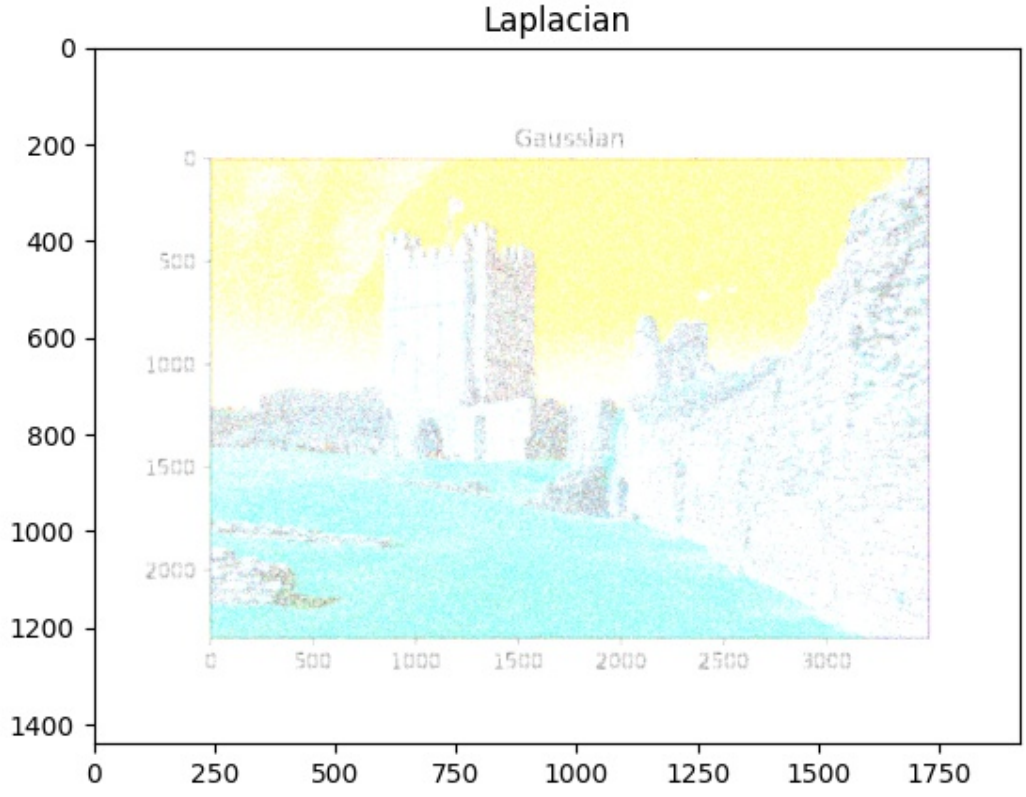

Gaussian Blur
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000

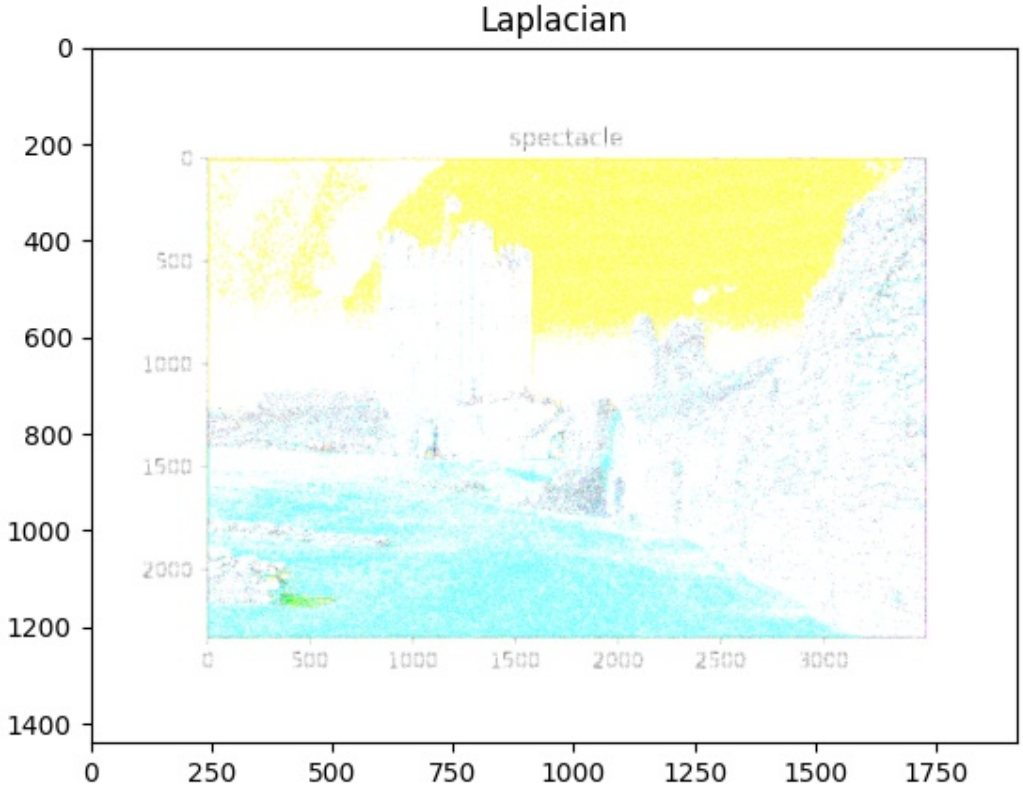

Gaussian Blur
spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000

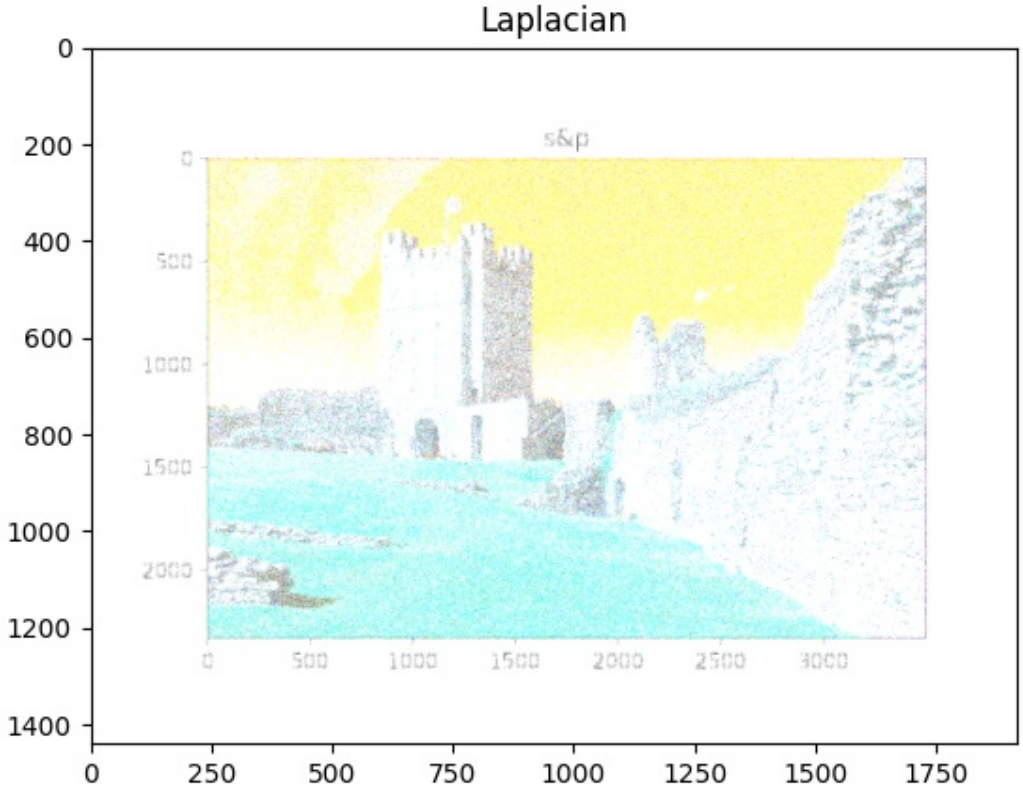

Gaussian Blur
Mean


Gaussian Blur
s&p

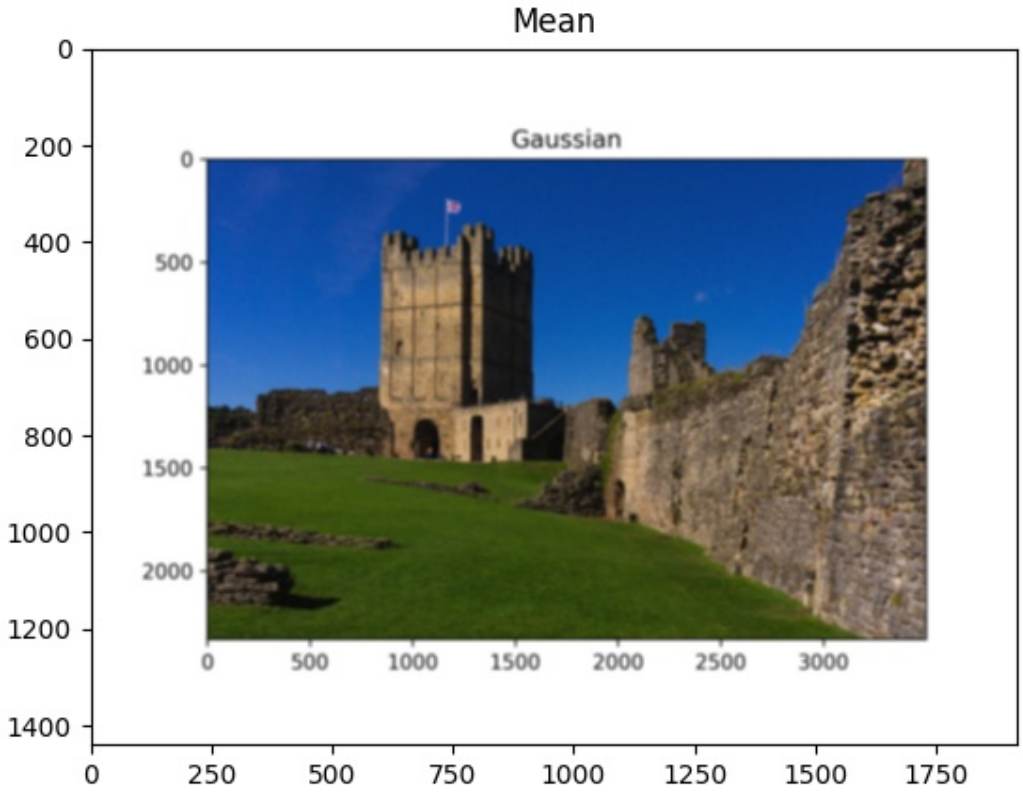

Laplacian
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

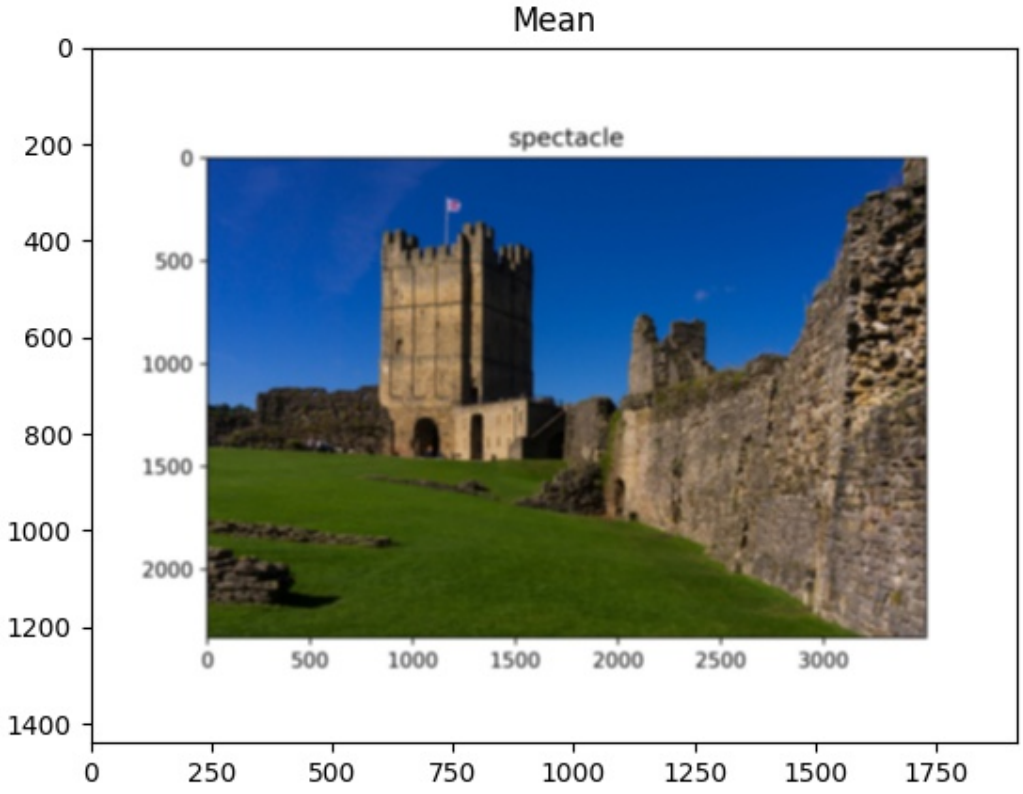

Laplacian
Gausslan
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

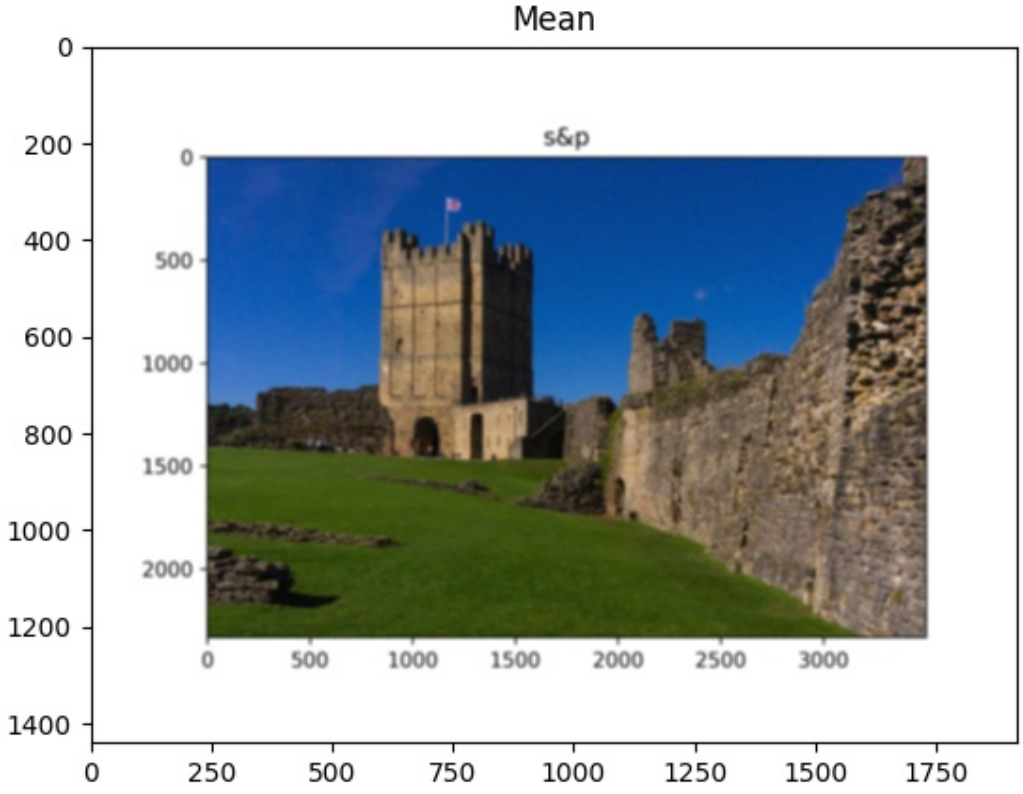

Laplacian
spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

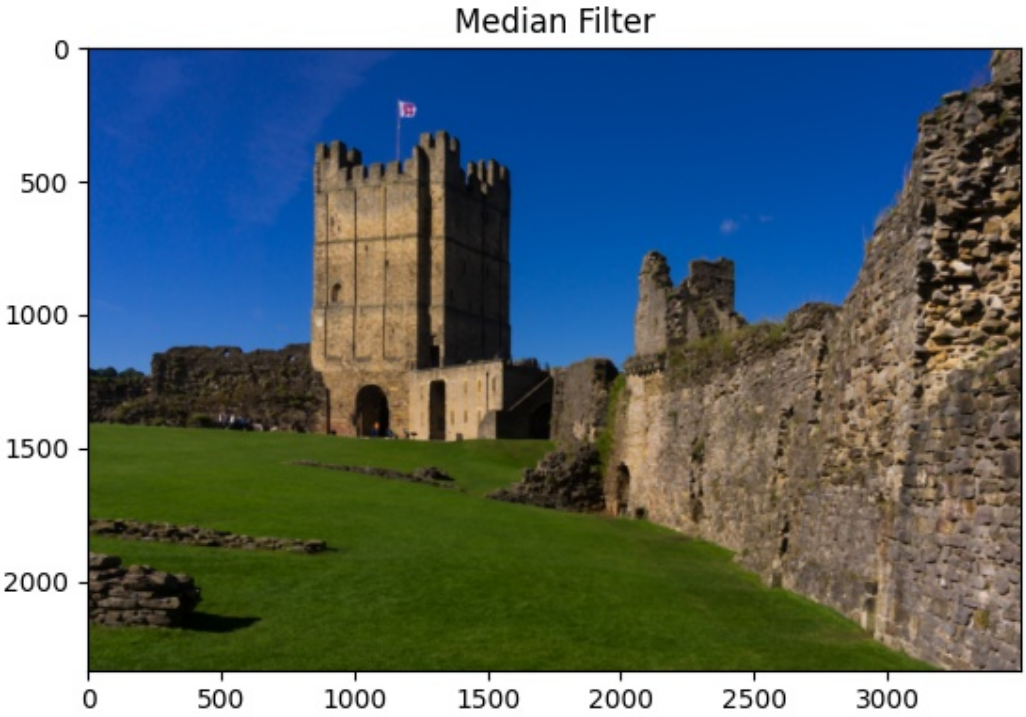

Laplacian
s&p
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

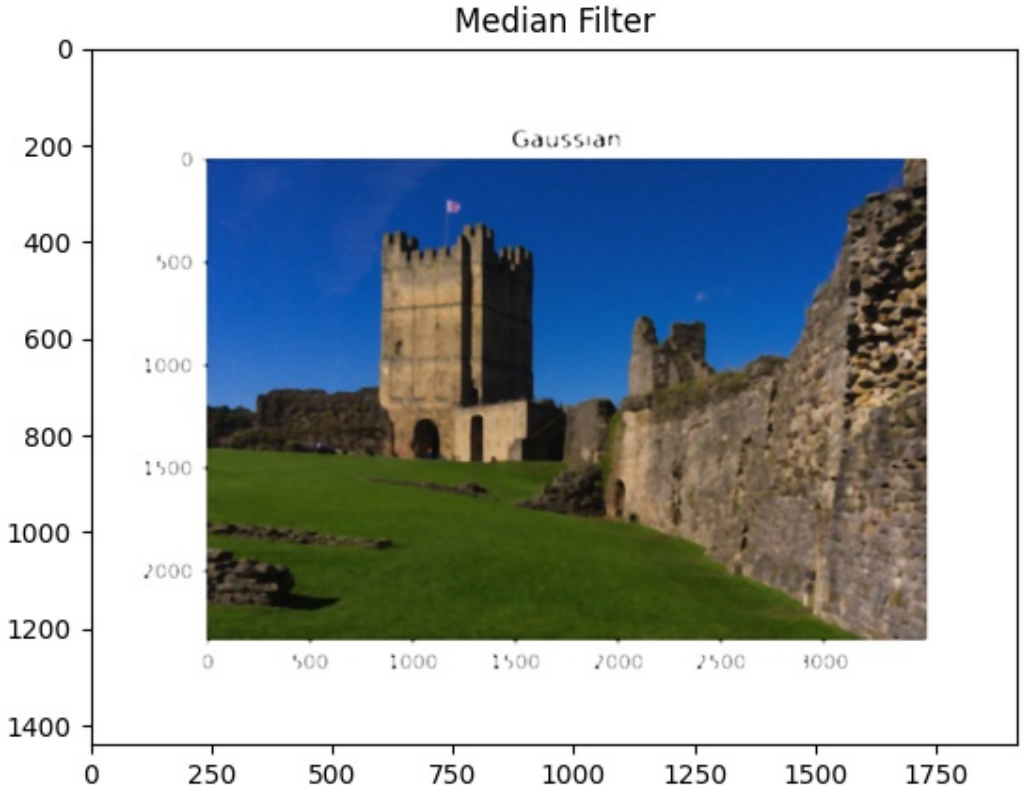

Mean
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

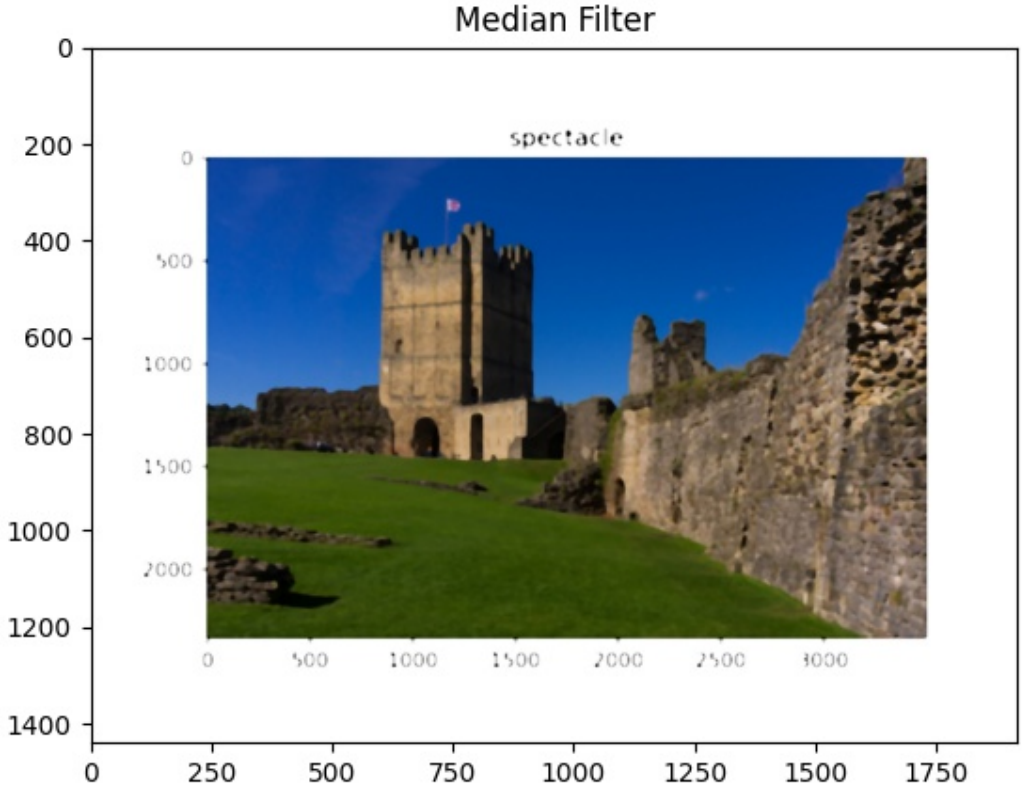

Mean
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

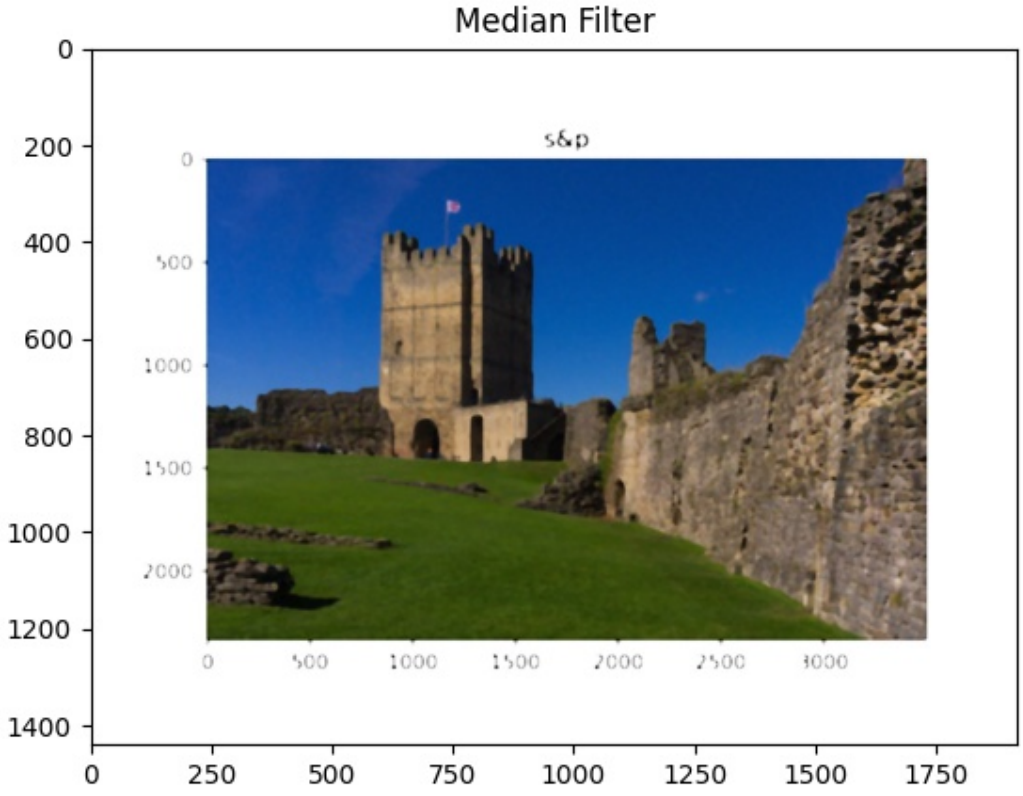

Mean
spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000

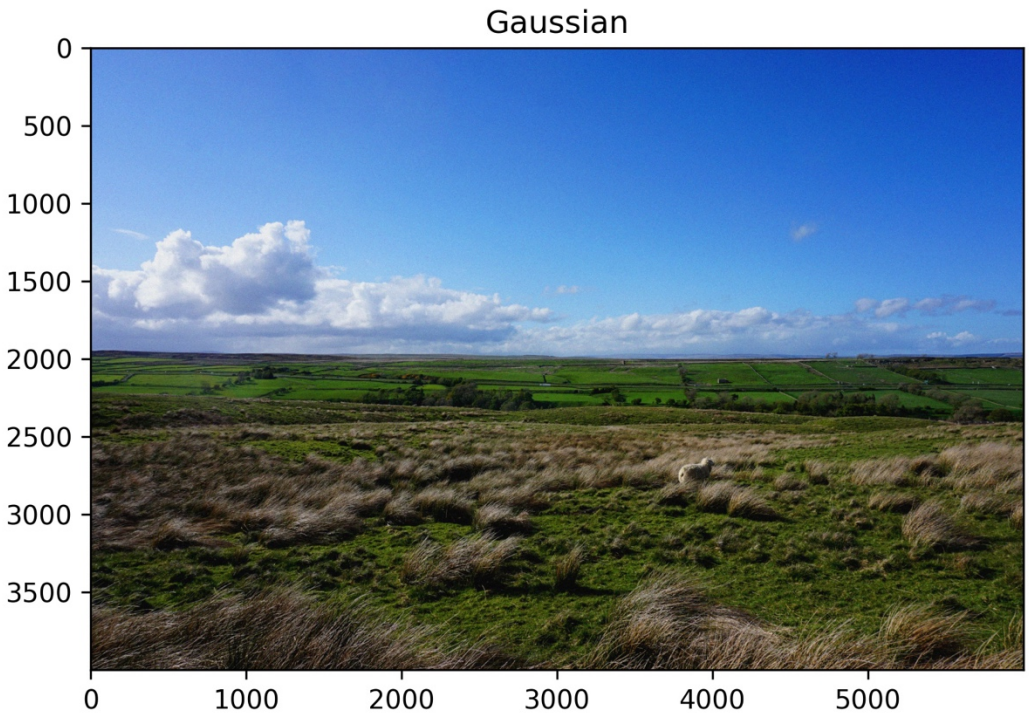

Mean
s&p
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000

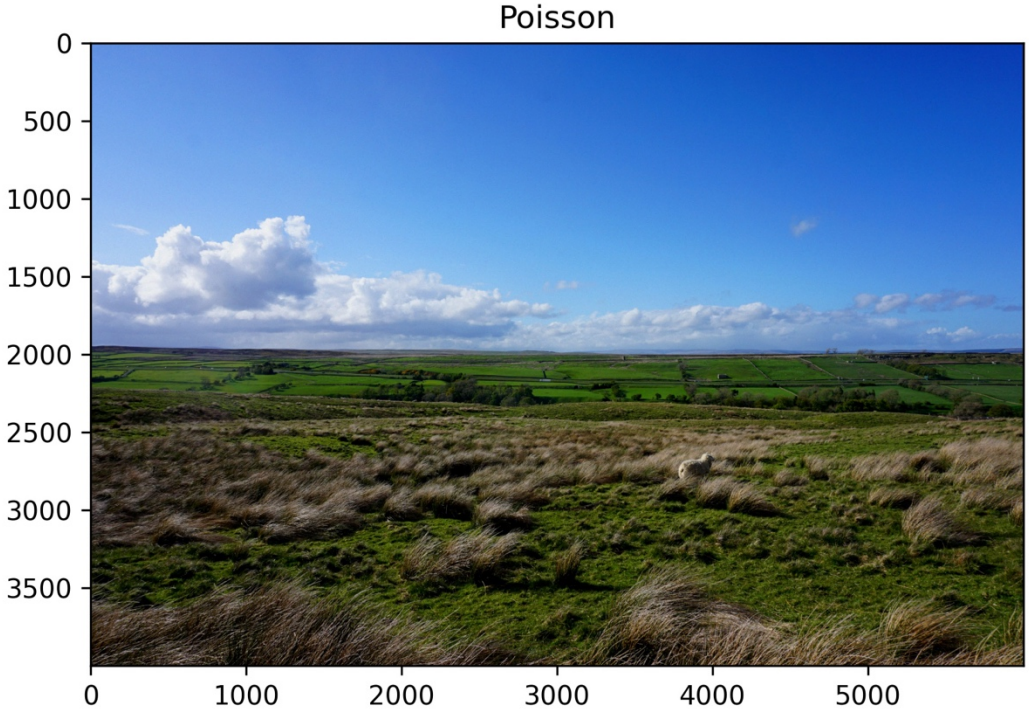

Median Filter
0
500
1000
1500
2000
0
500
1000
1500
2000
2500
3000

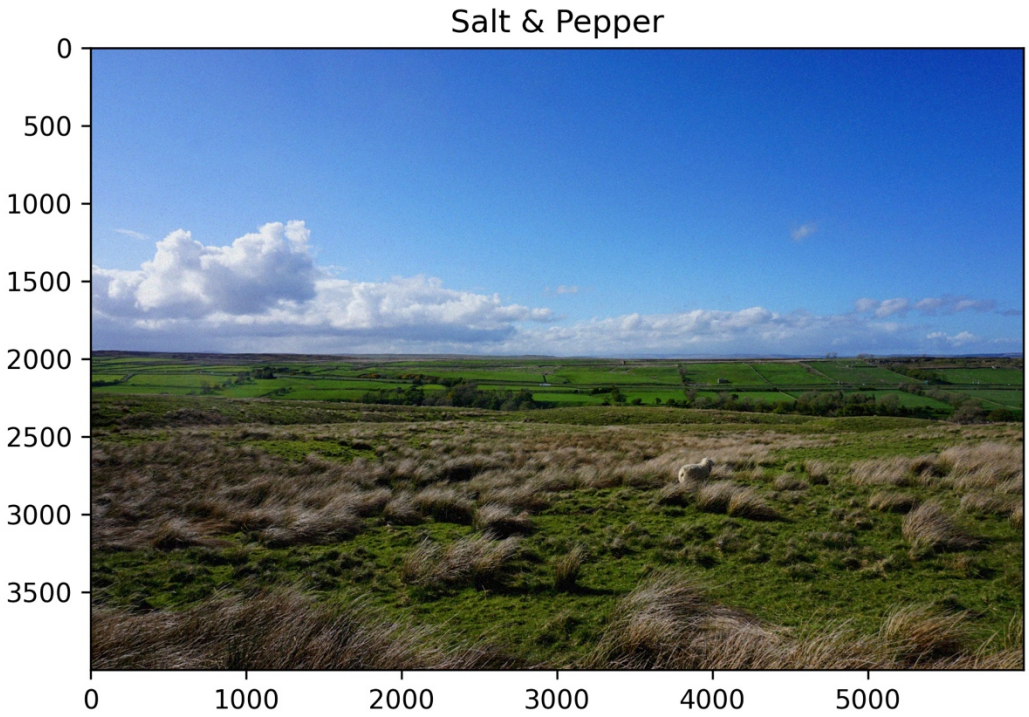

Median Filter
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

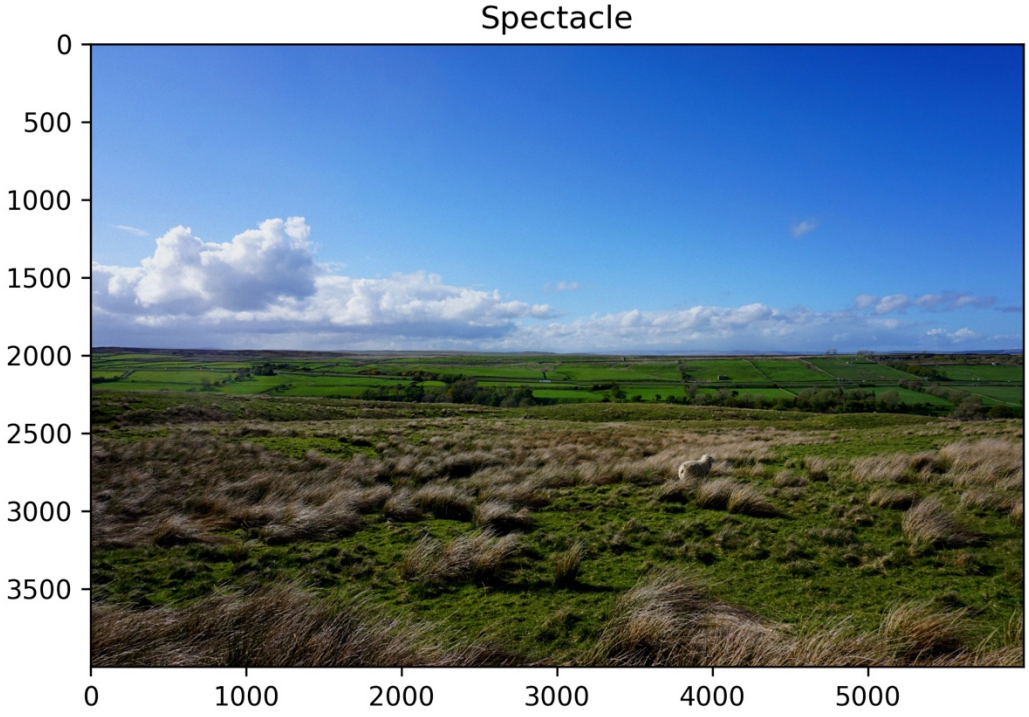

Median Filter
spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

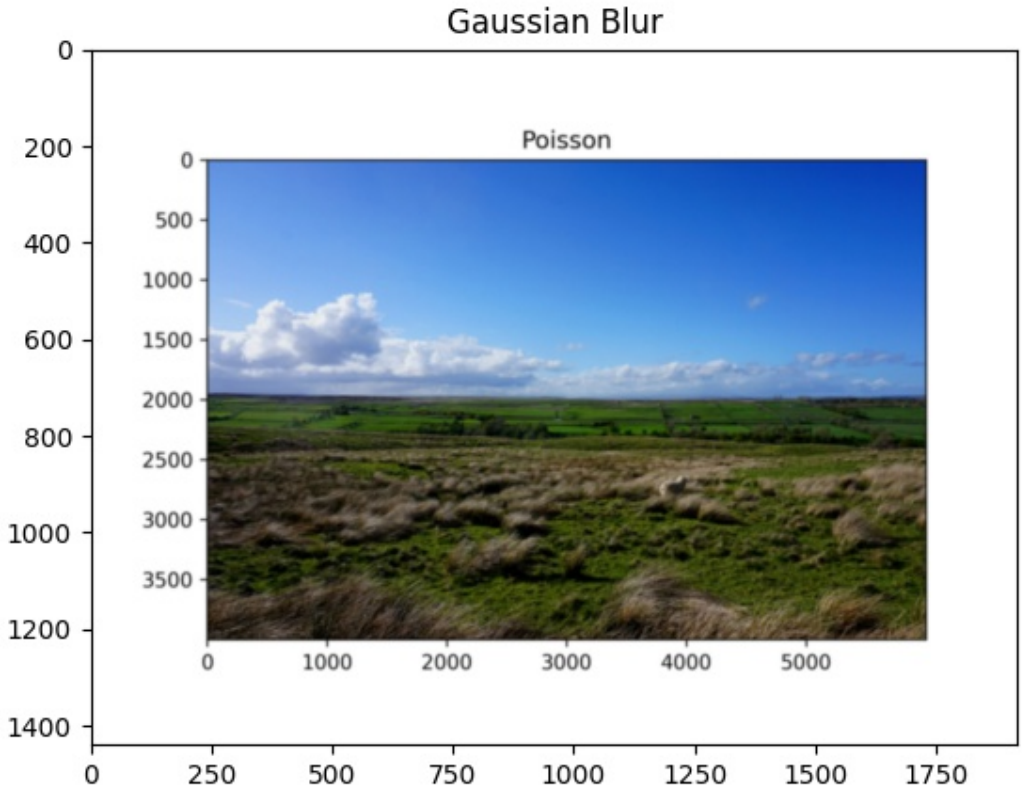

Median Filter
s&p
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

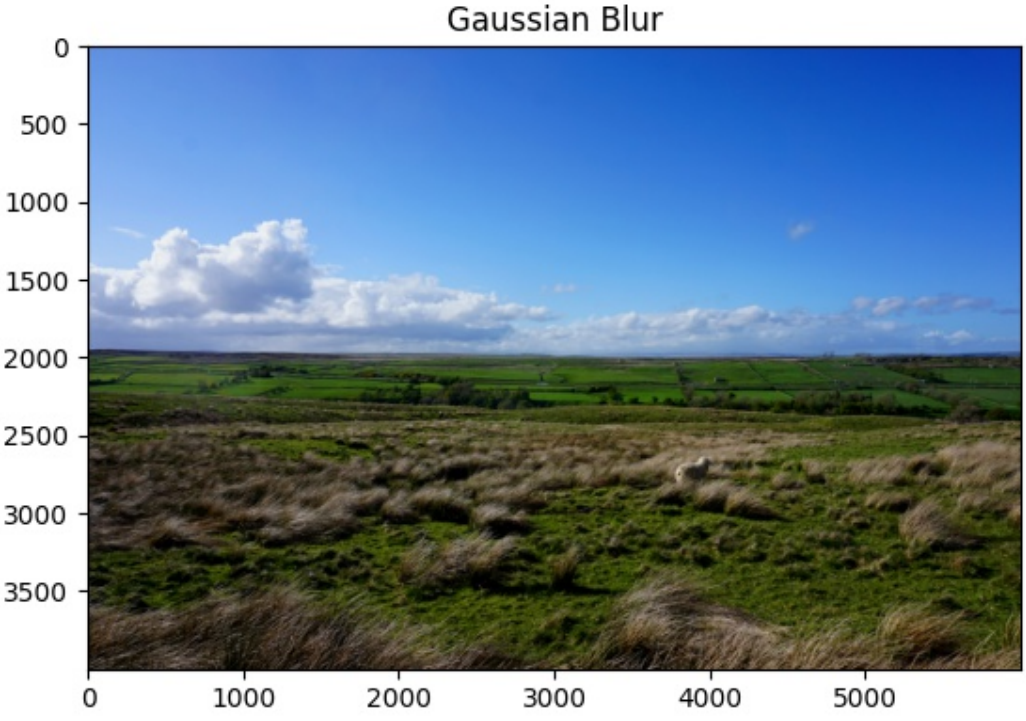

Gaussian
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

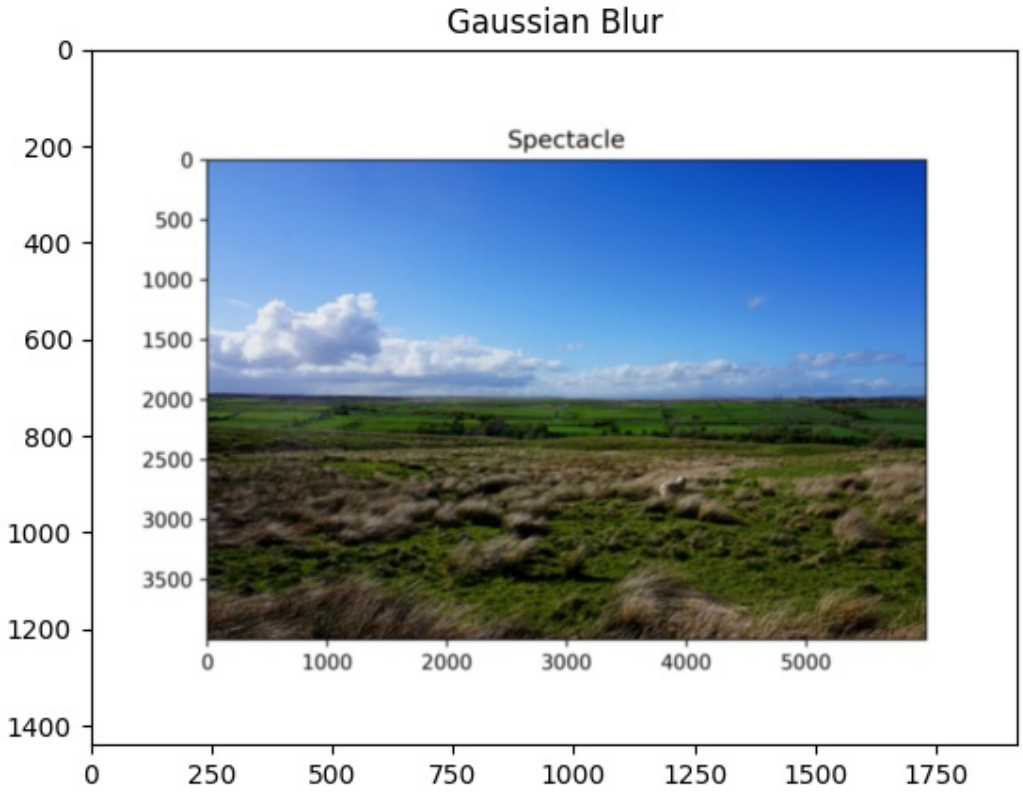

Poisson
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

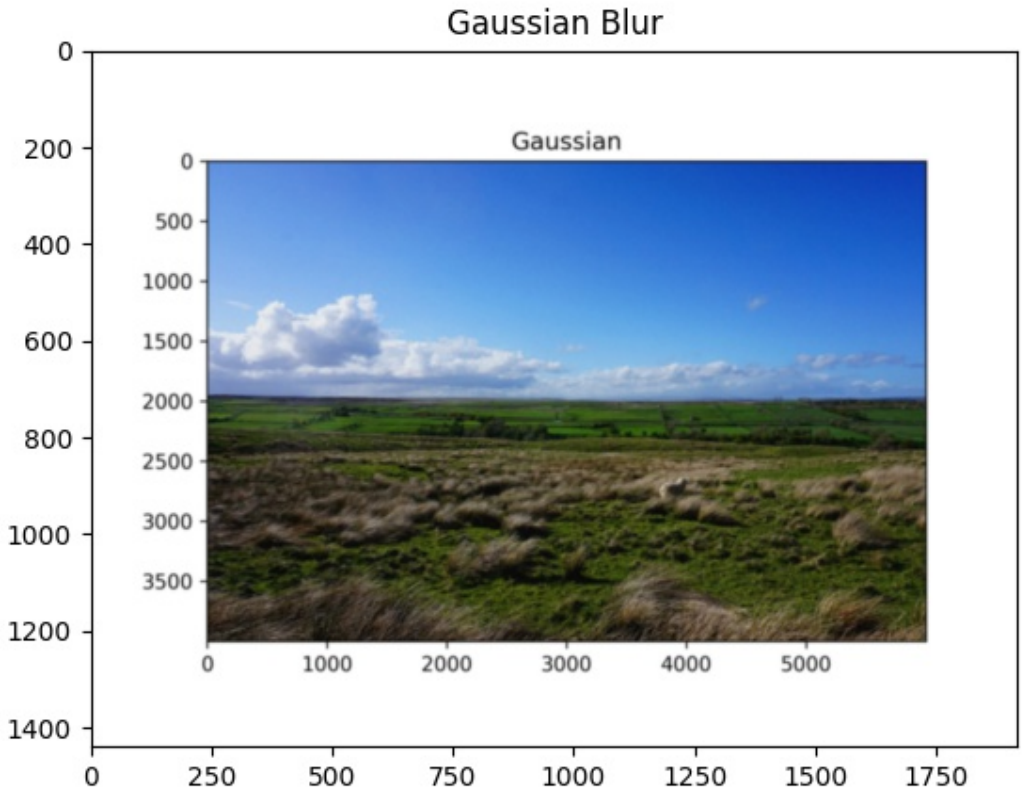

Salt & Pepper
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

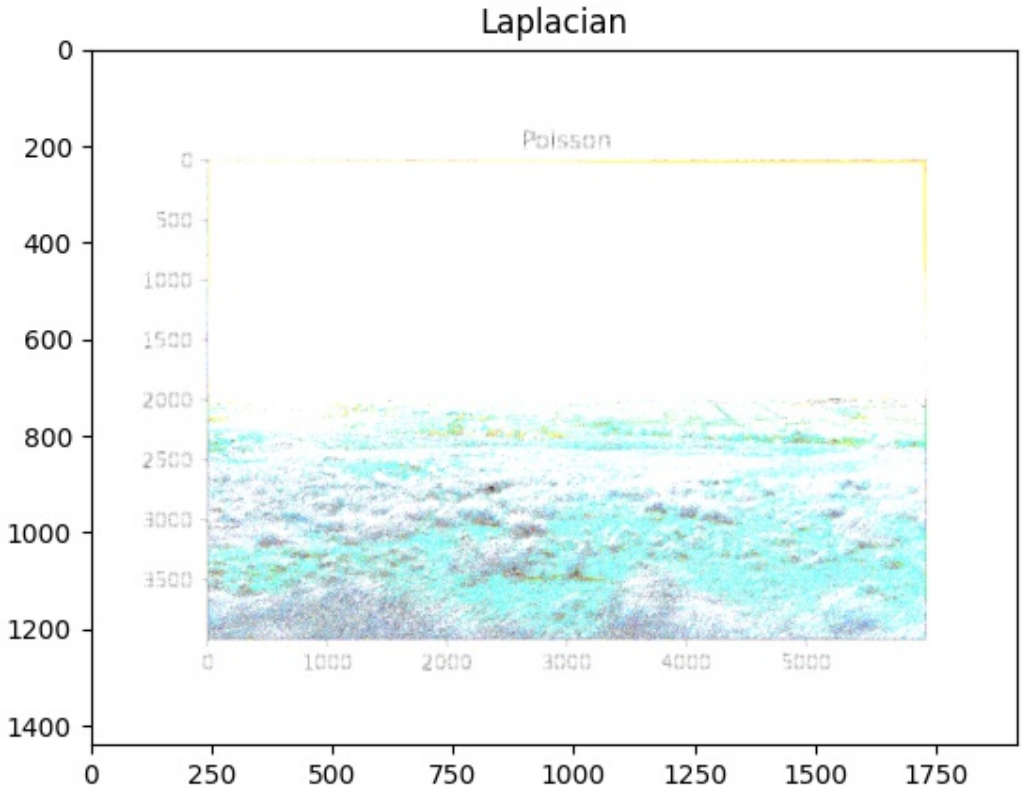

Spectacle
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

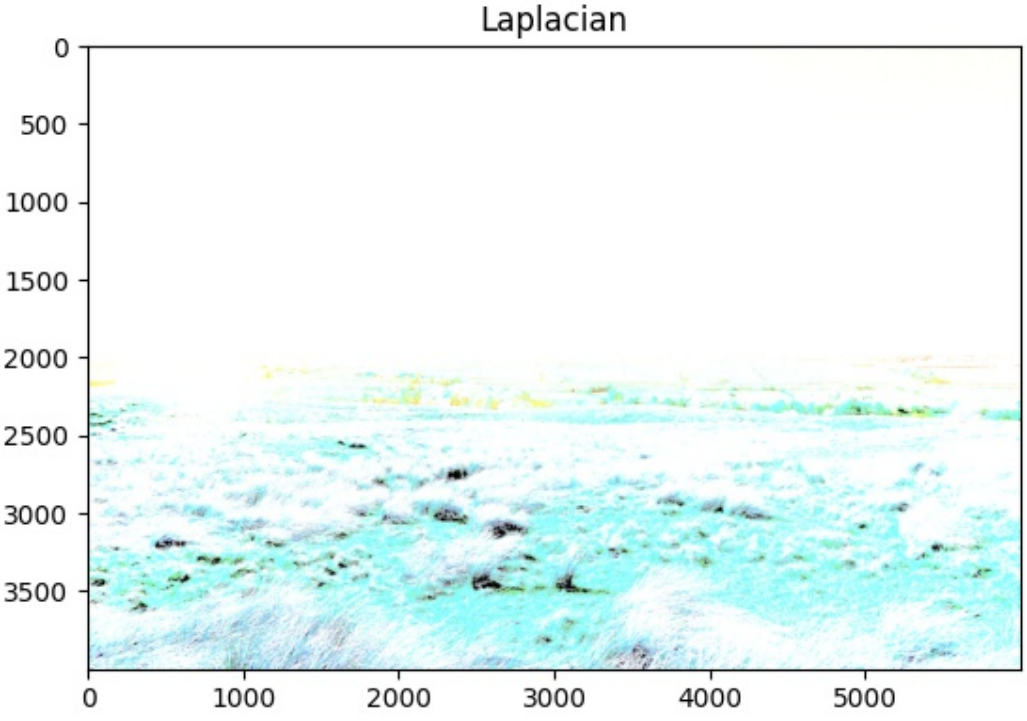

Gaussian Blur
Poisson
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

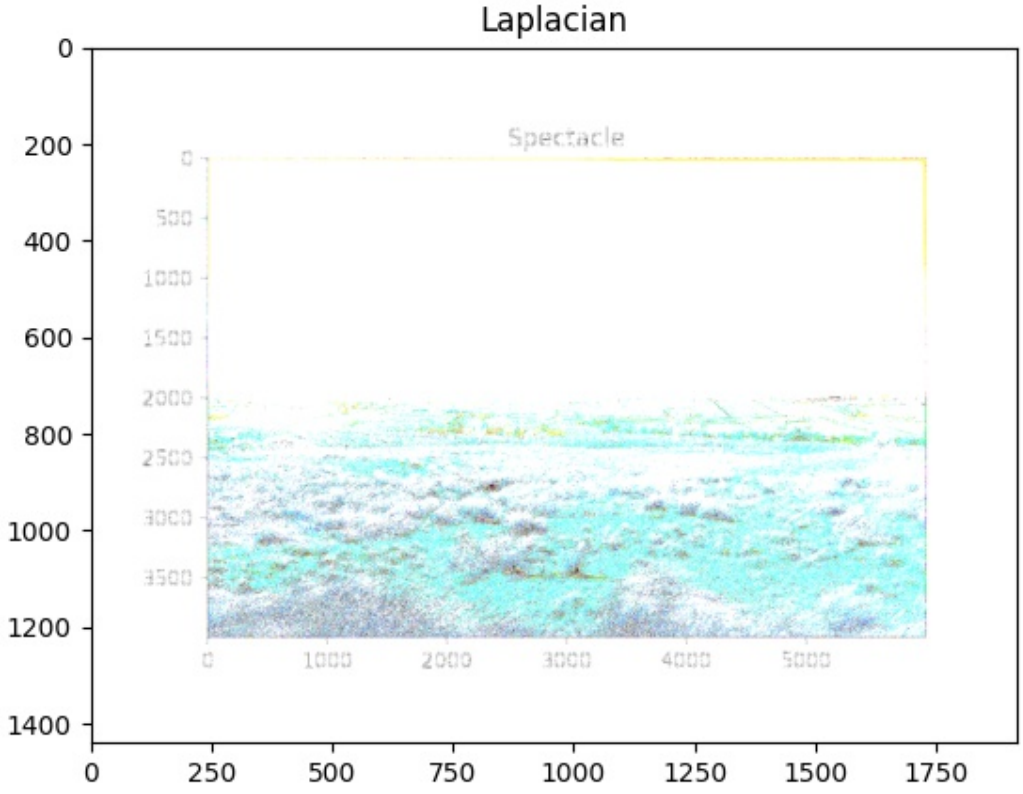

Gaussian Blur
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

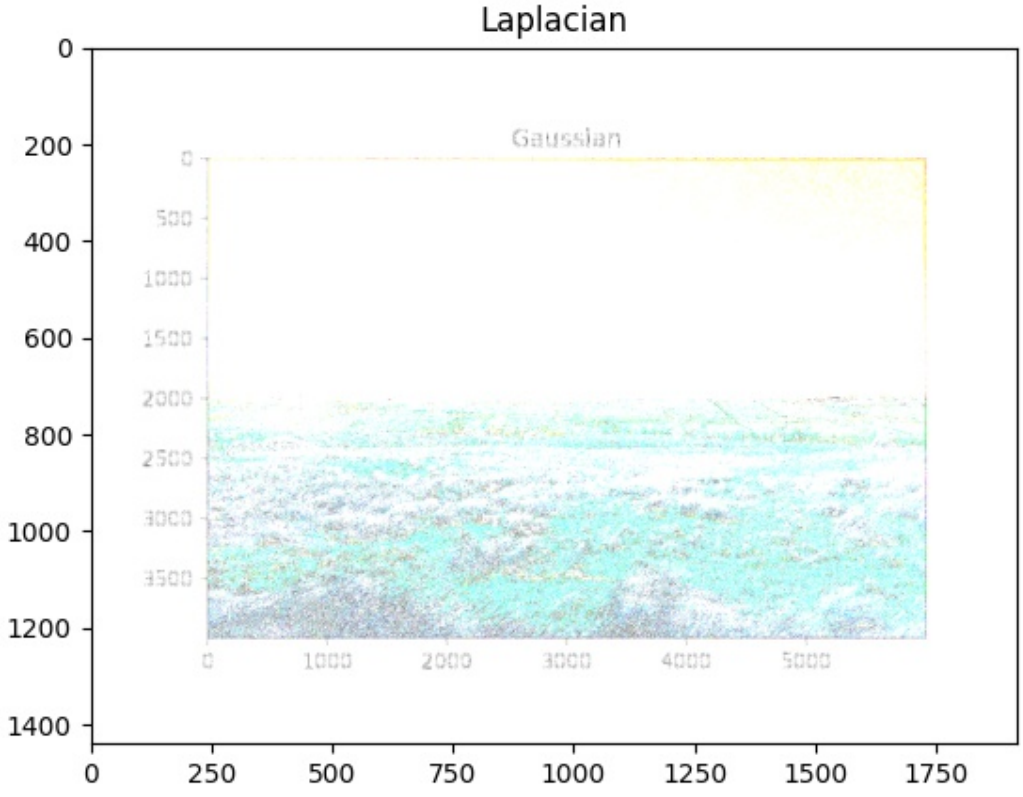

Gaussian Blur
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

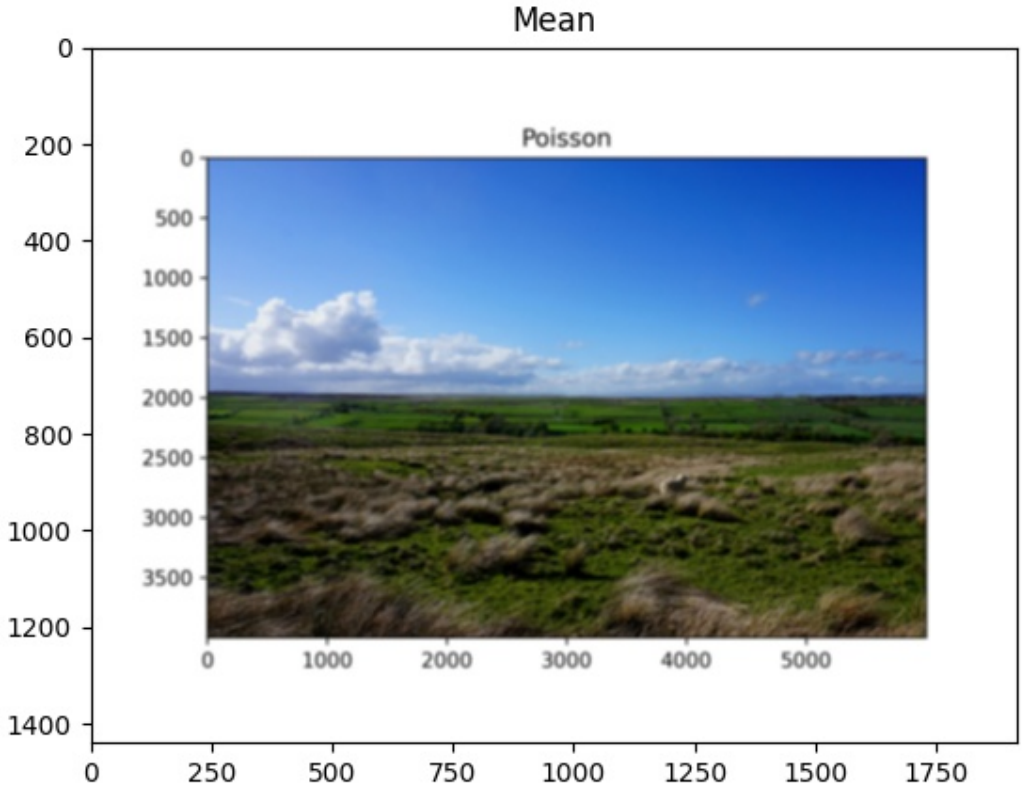

Gaussian Blur
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

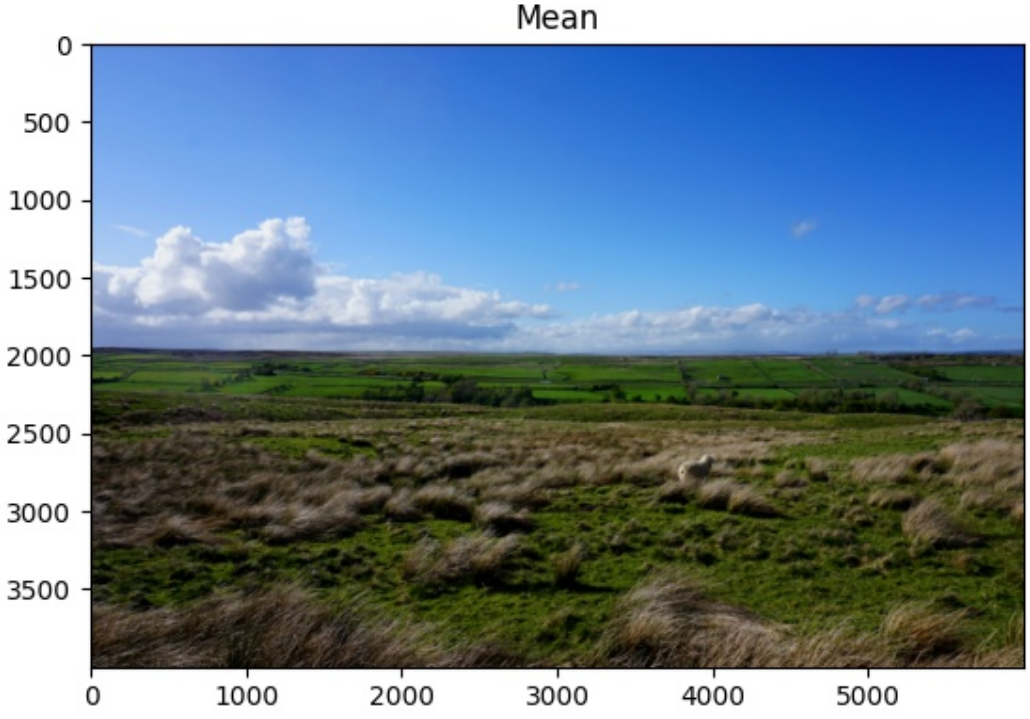

Laplacian
Poisson
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

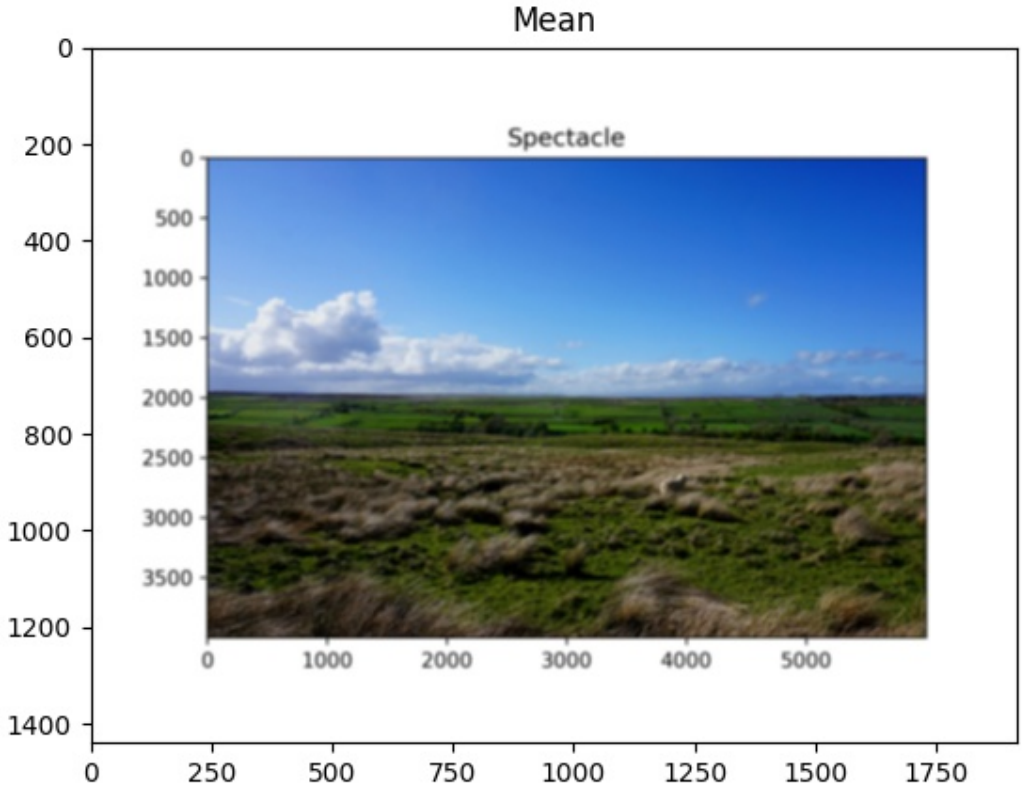

Laplacian
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

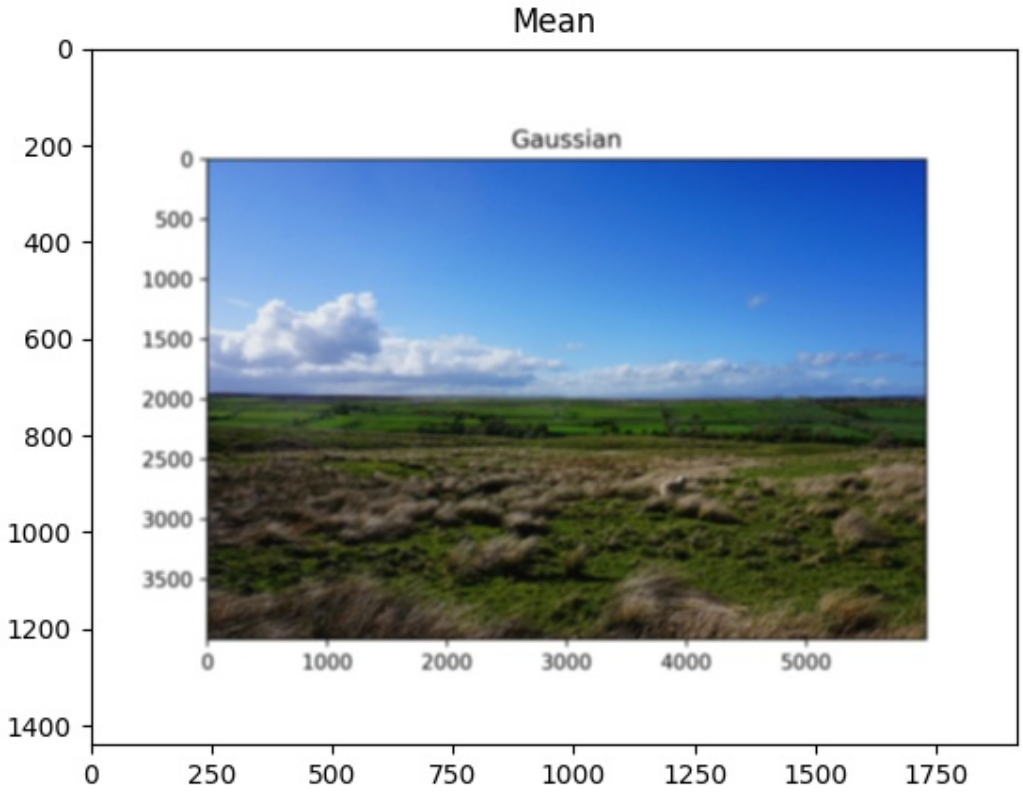

Laplacian
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

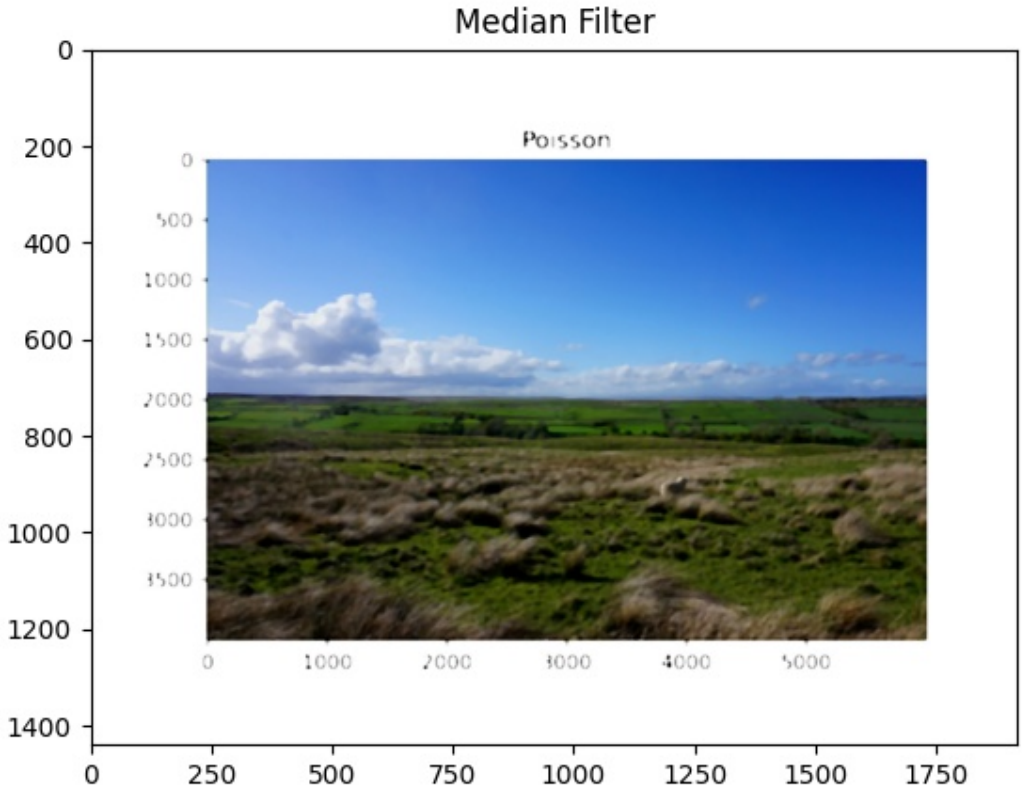

Laplacian
Gausslan
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

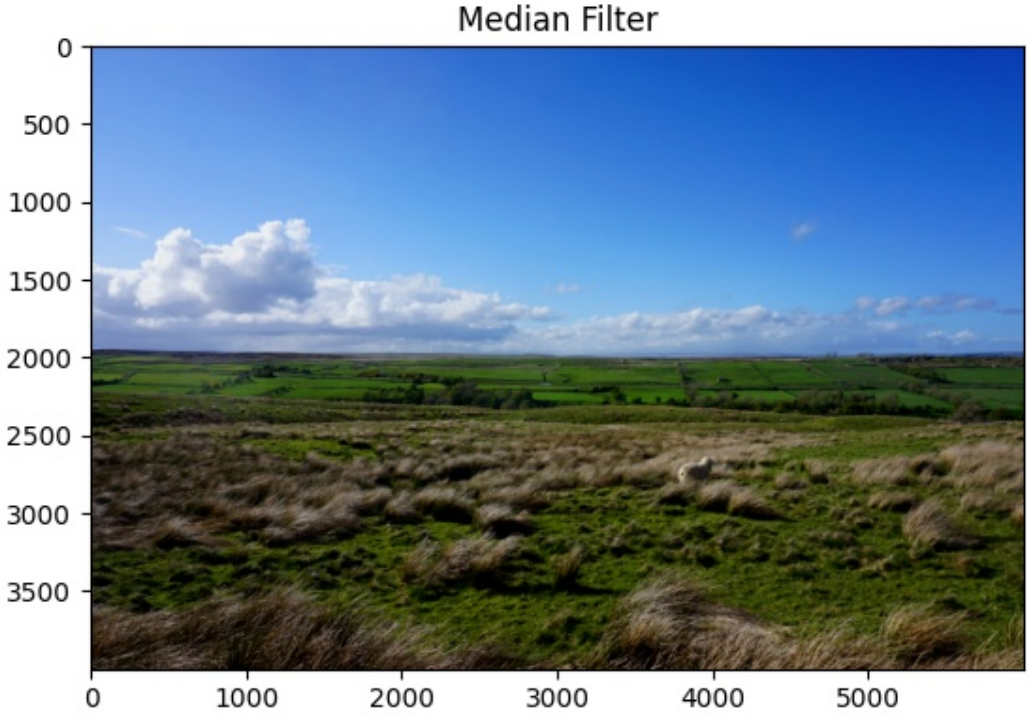

Mean
Poisson

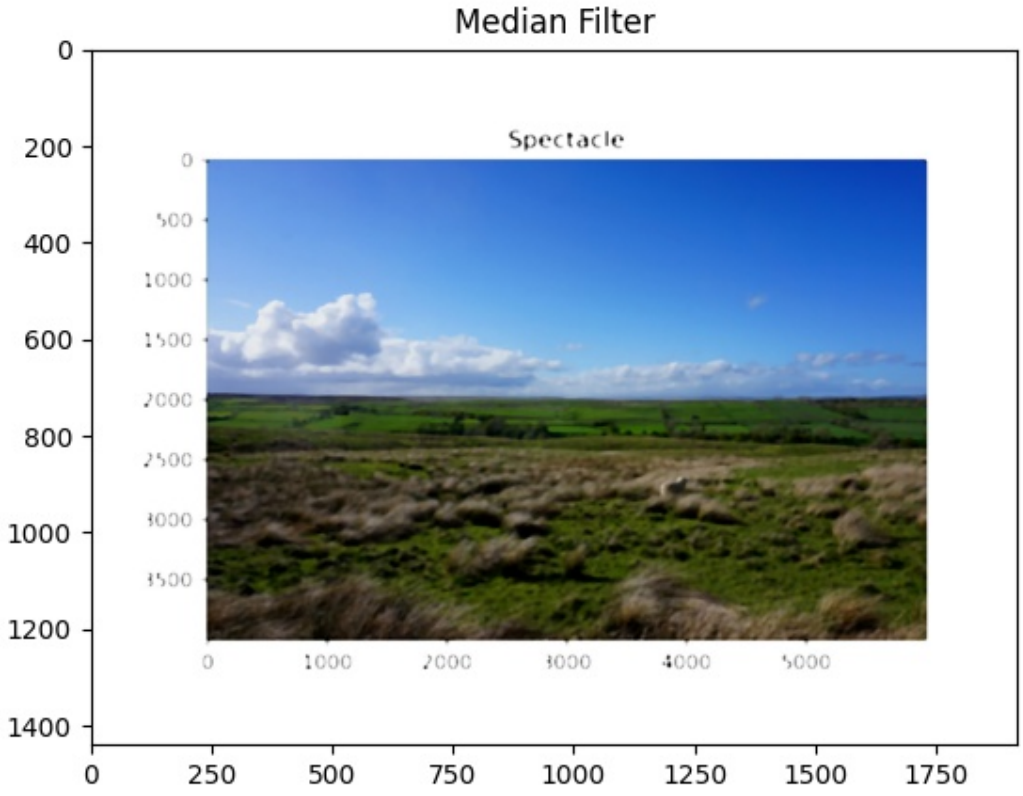

Mean

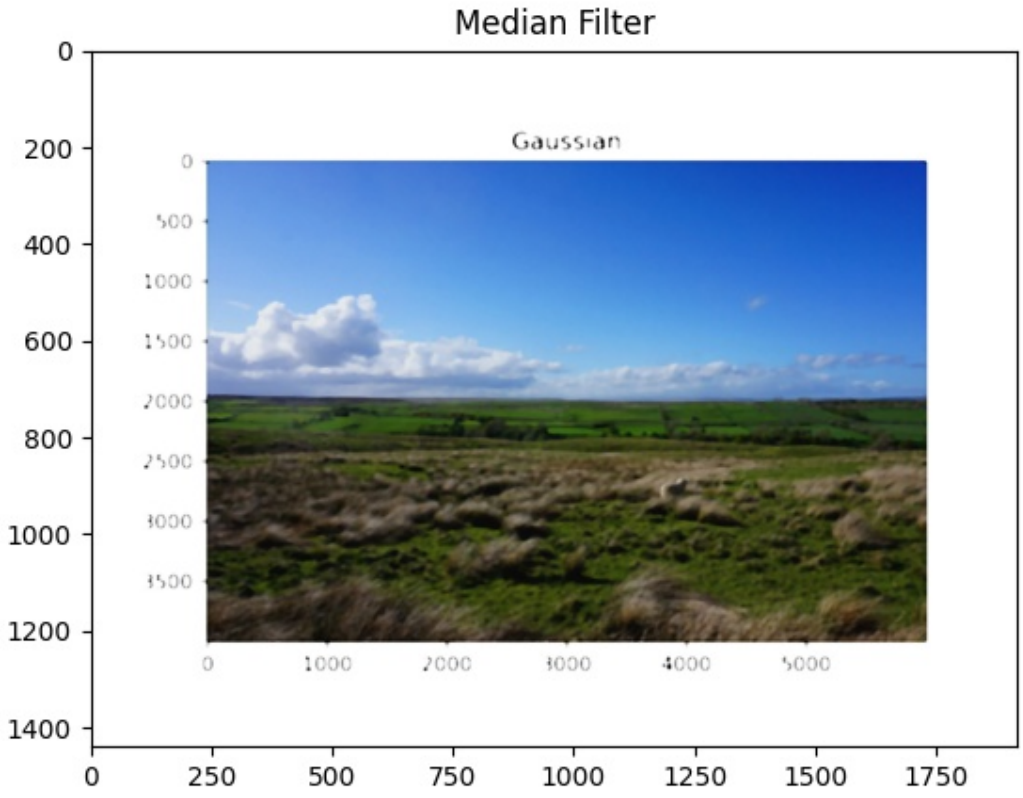

Mean
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

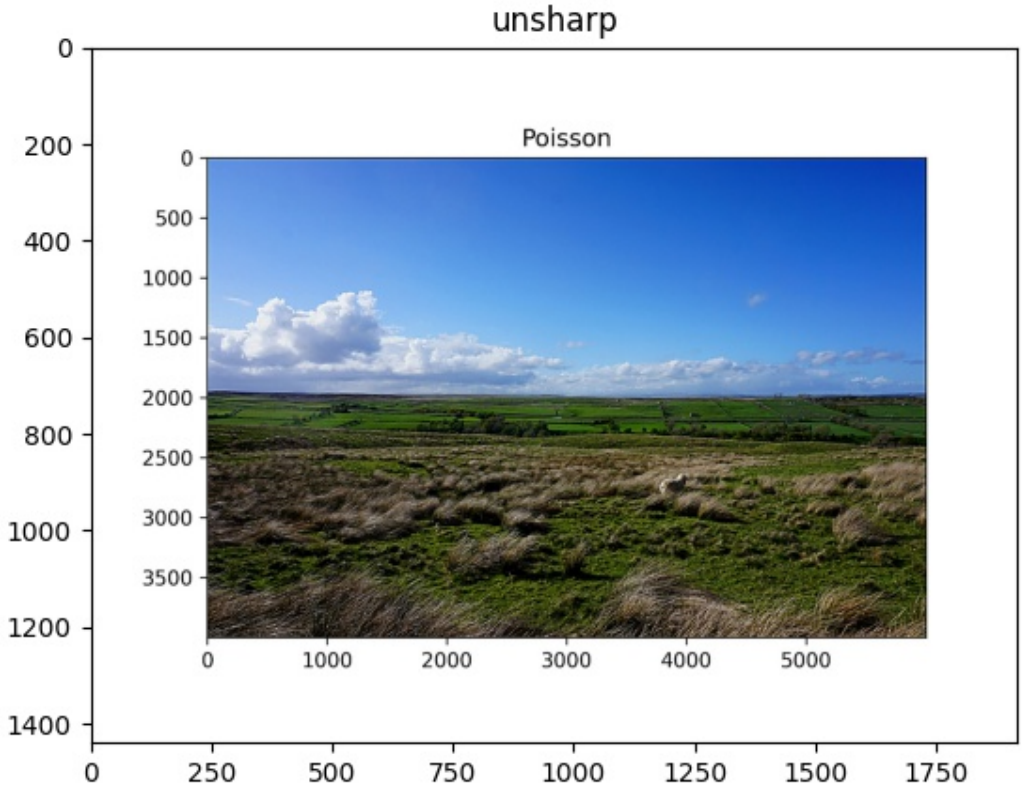

Mean
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

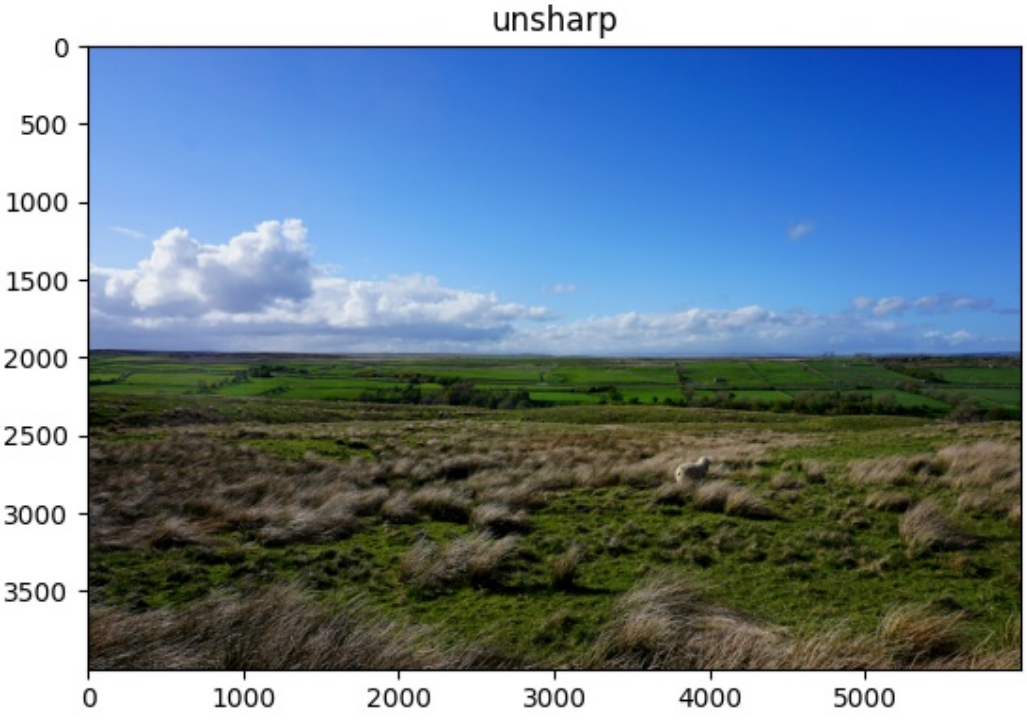

Median Filter
Poisson
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

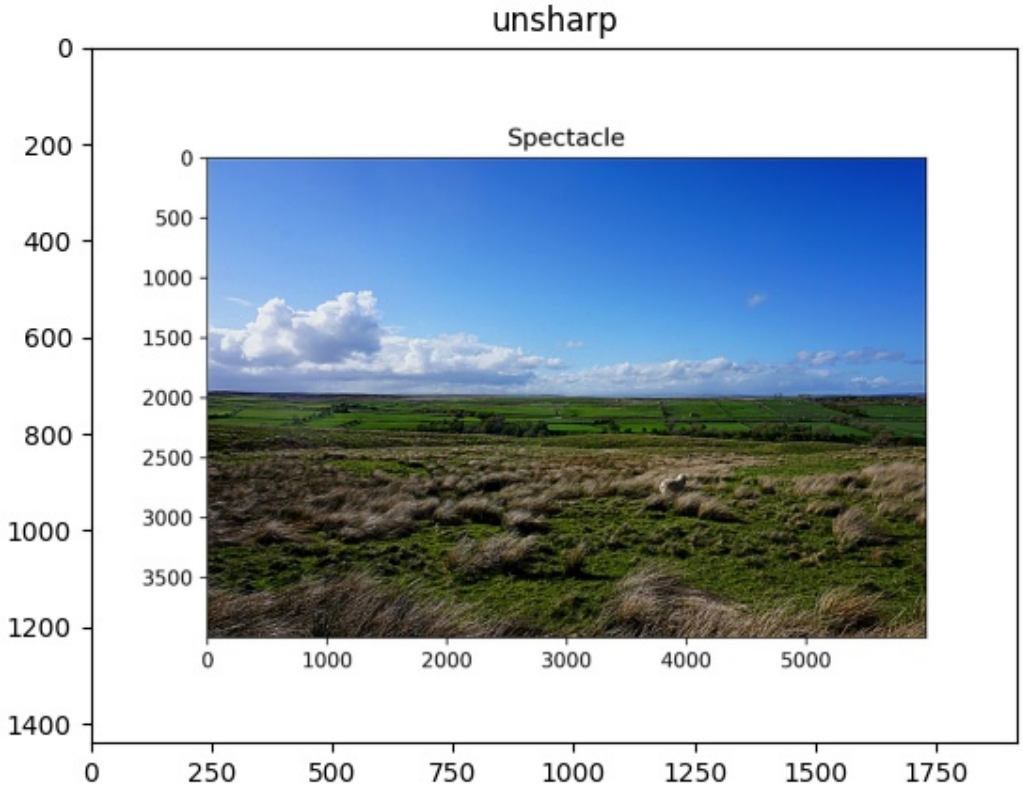

Median Filter
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

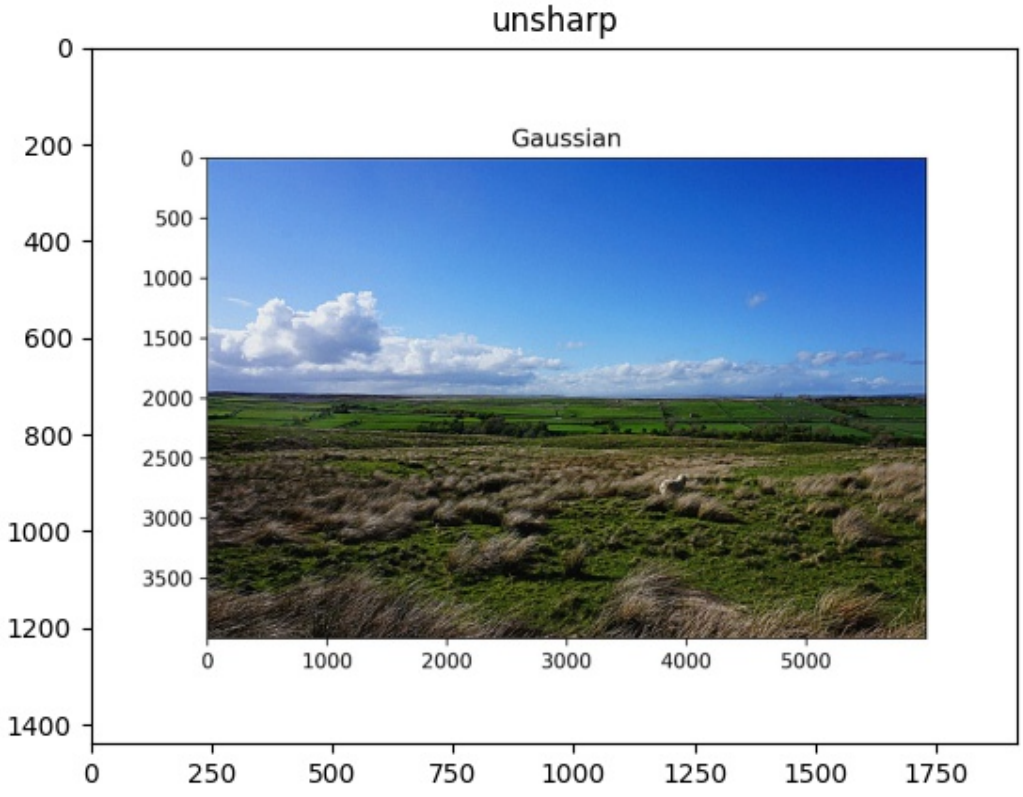

Median Filter
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750


Median Filter
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750


unsharp
Poisson
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750


unsharp
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

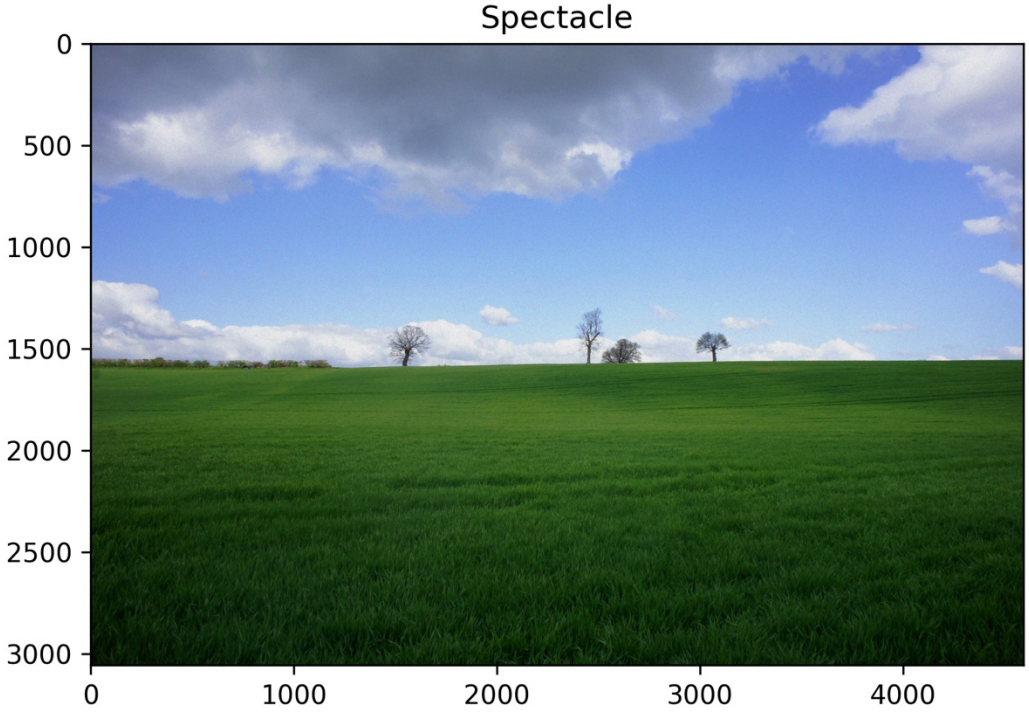

unsharp
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

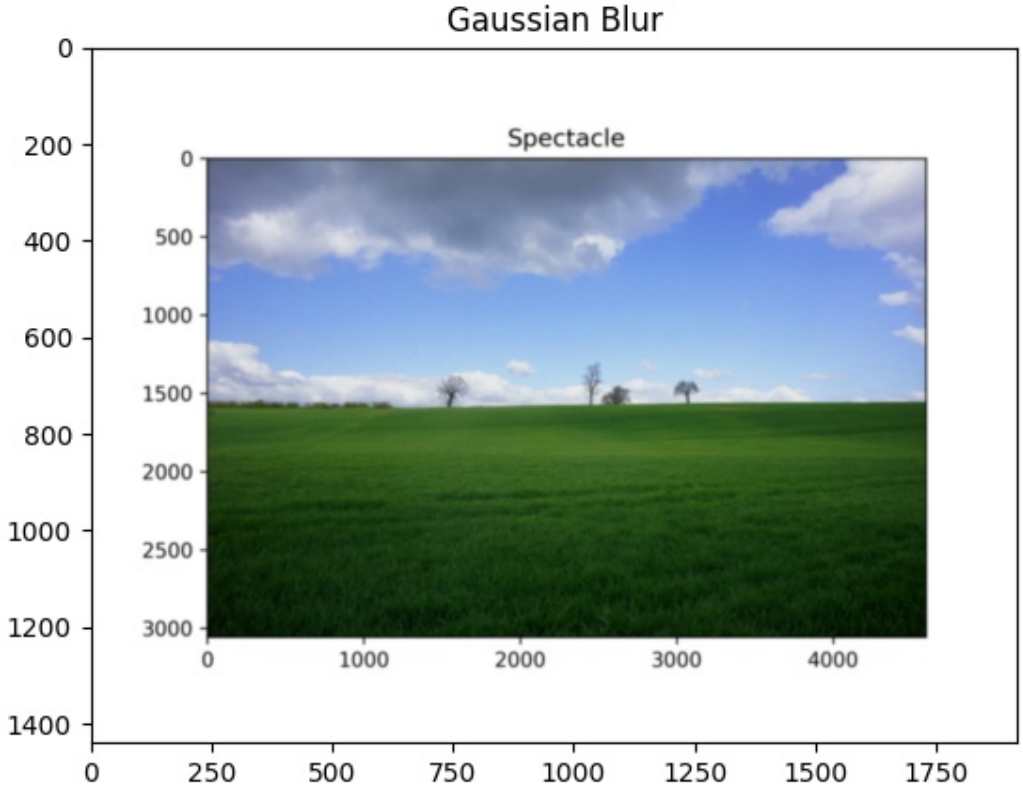

unsharp
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000

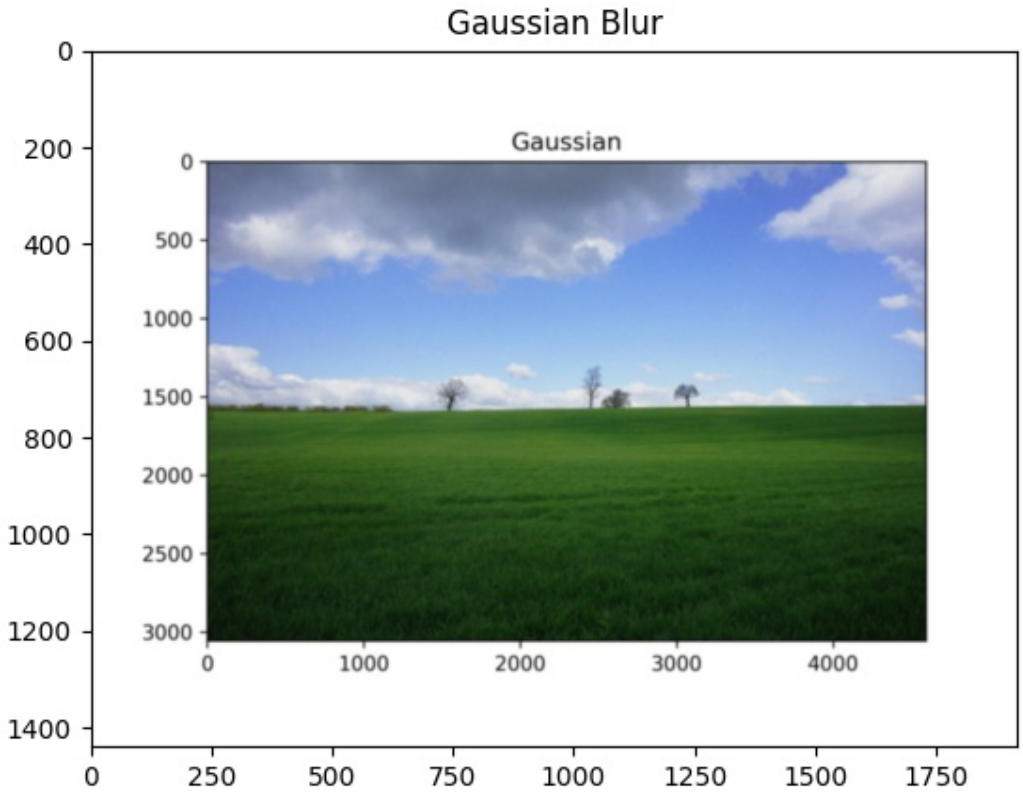

Gaussian
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

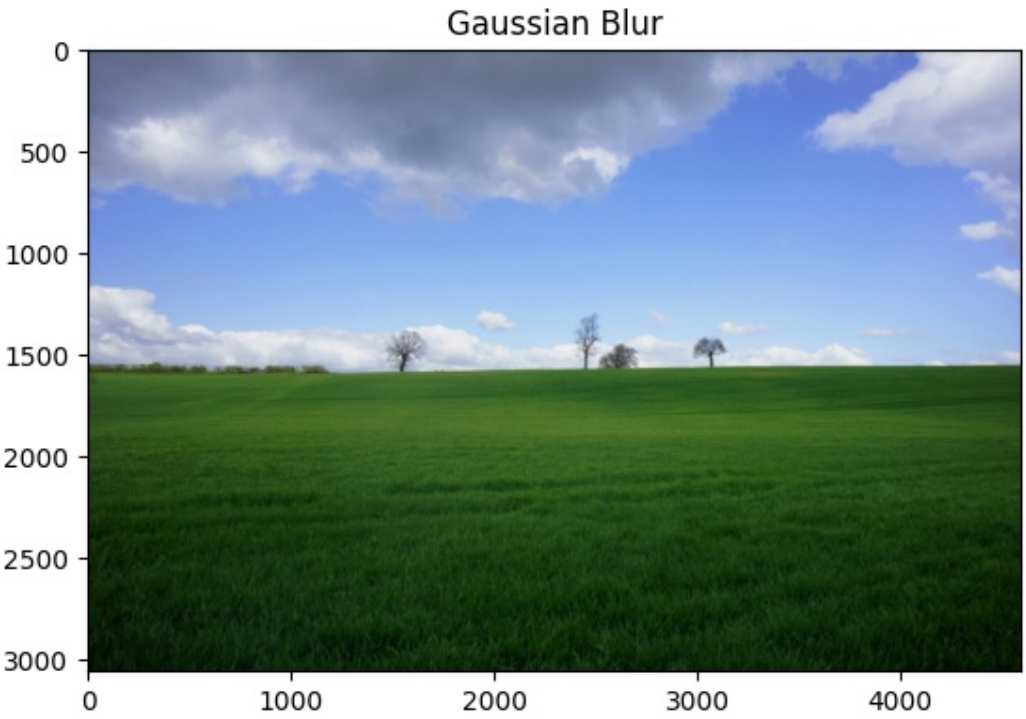

Poisson
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

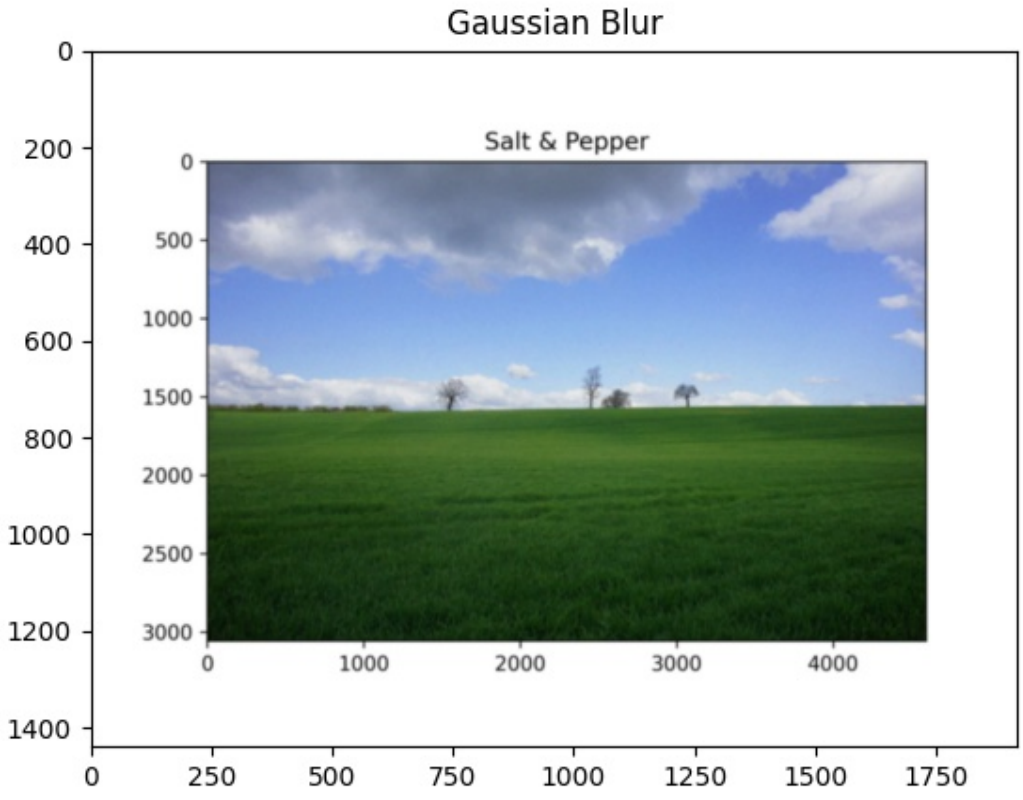

Salt & Pepper
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

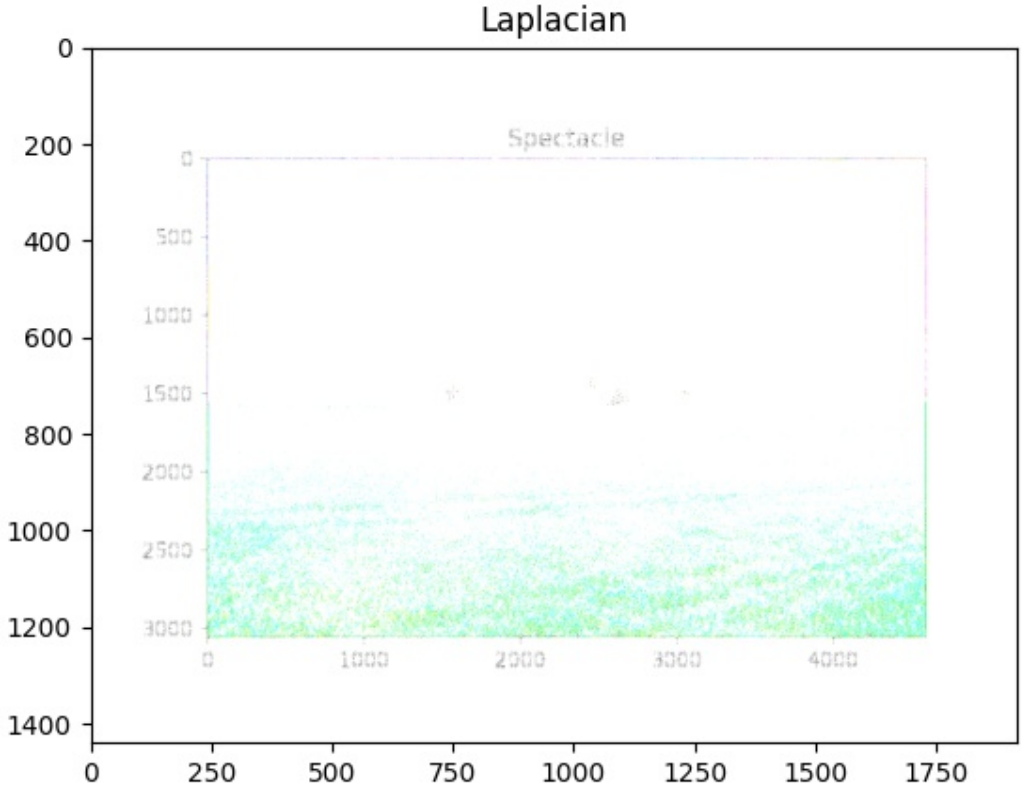

Spectacle
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

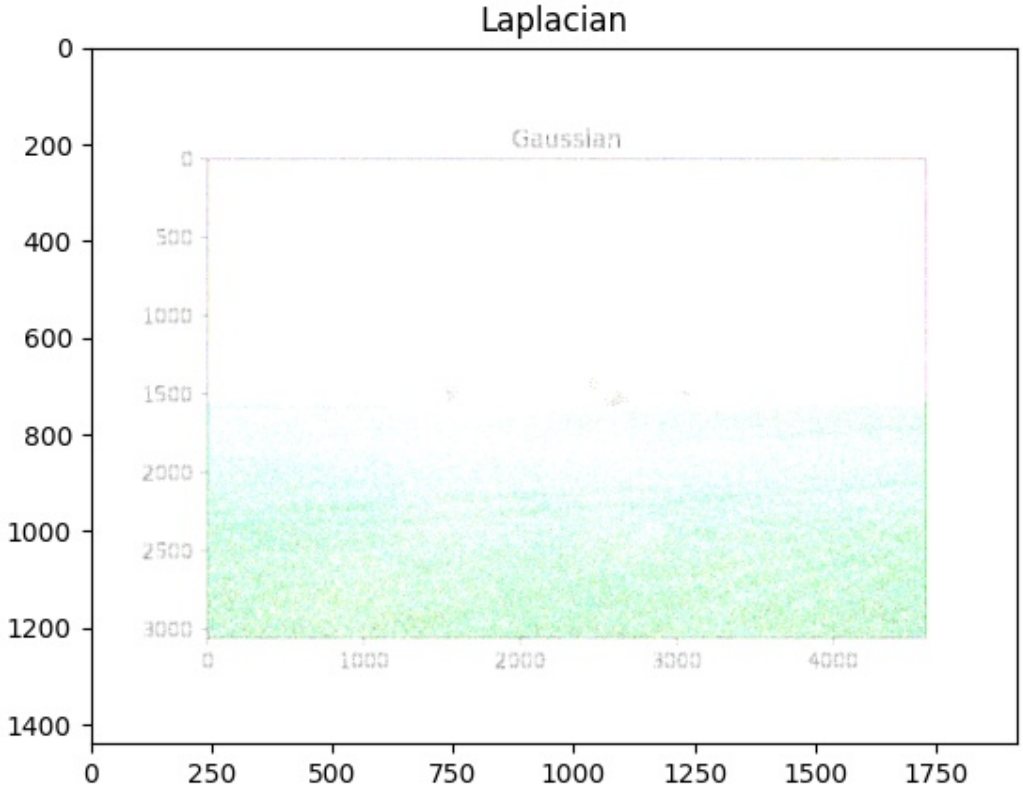

Gaussian Blur
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

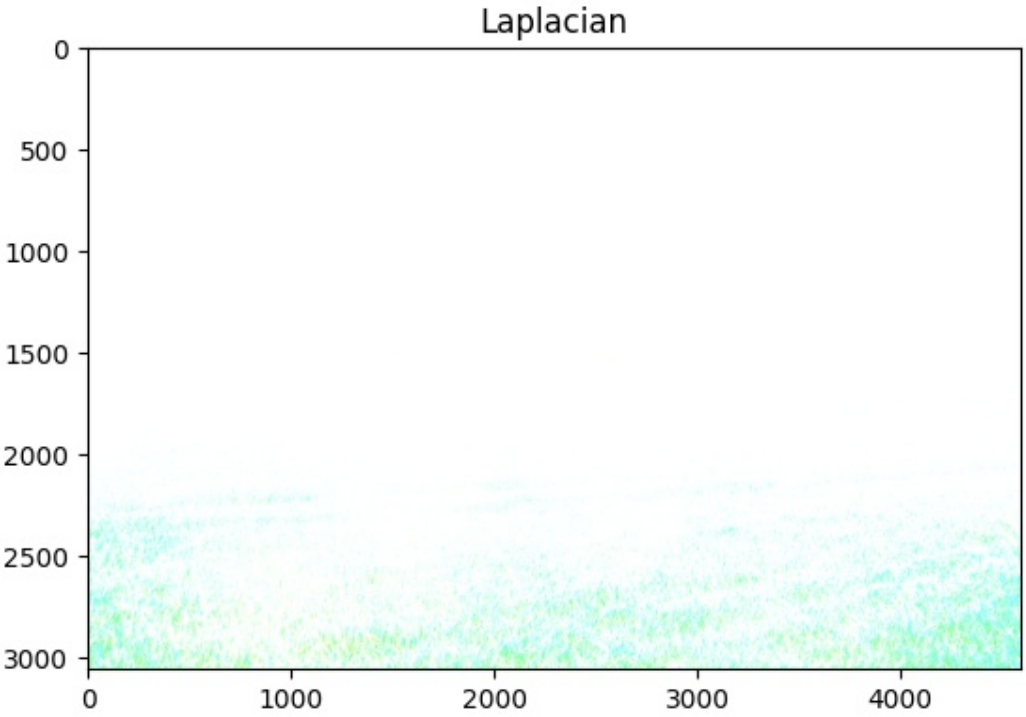

Gaussian Blur
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

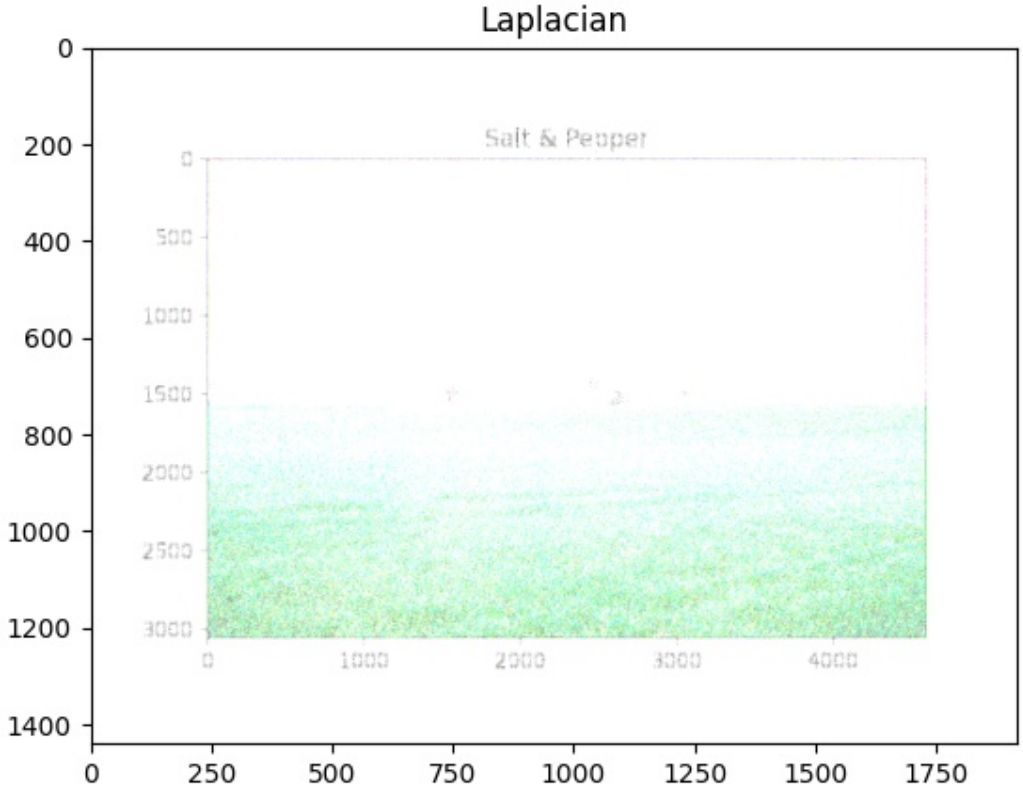

Gaussian Blur
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

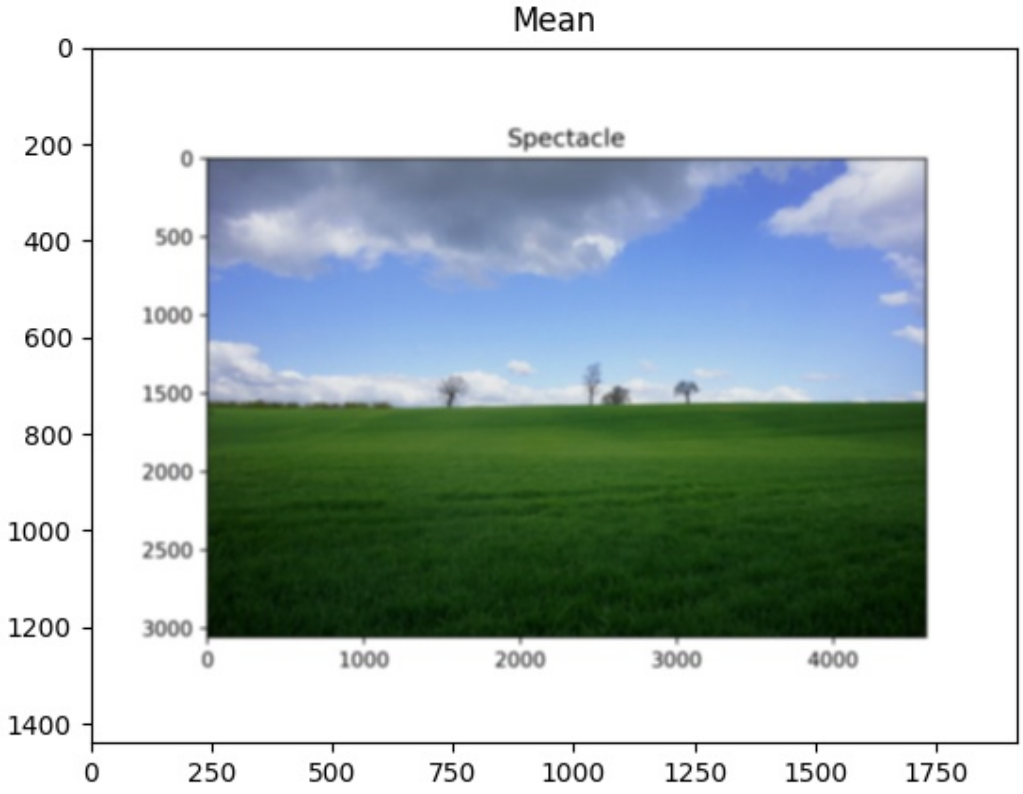

Gaussian Blur
Salt & Pepper
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

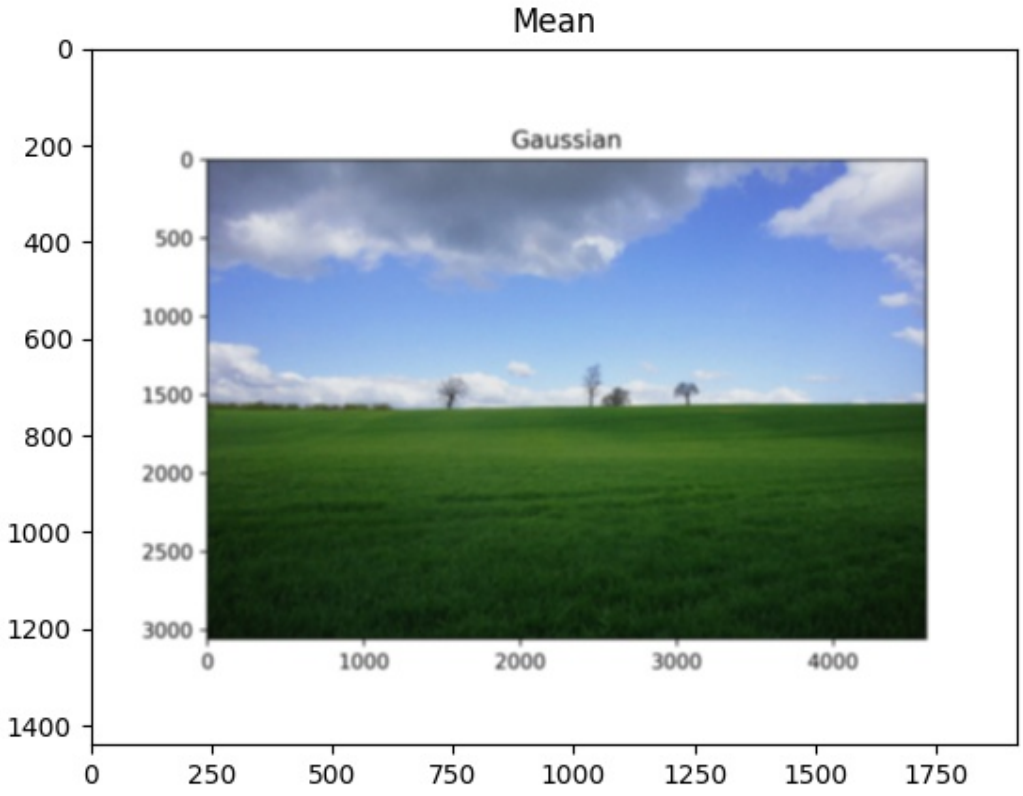

Laplacian
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

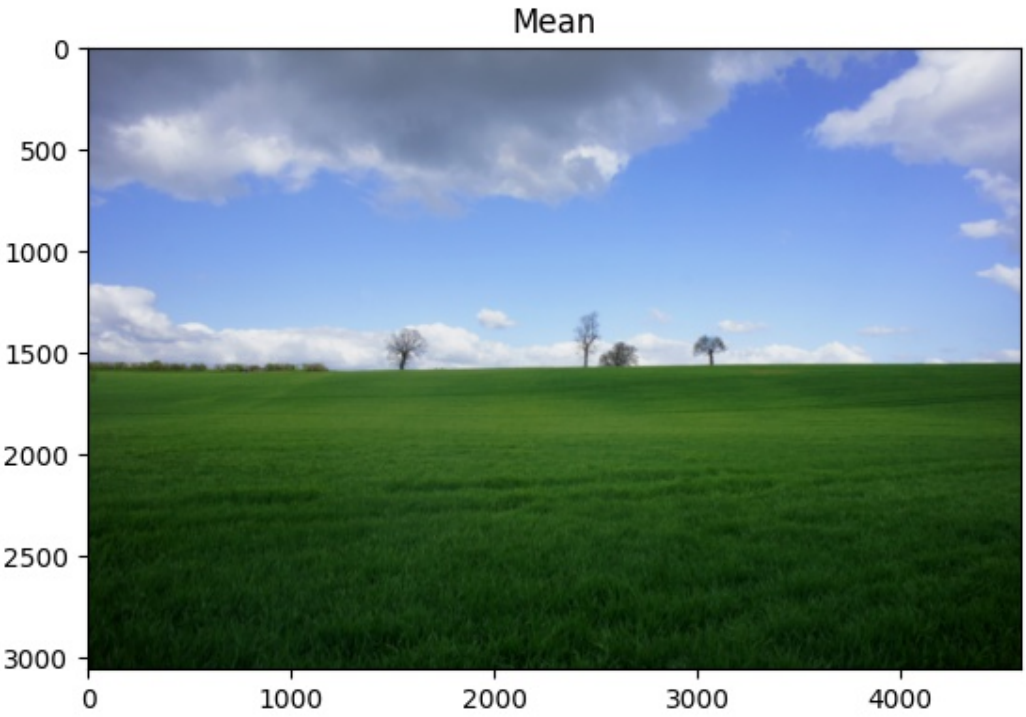

Laplacian
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

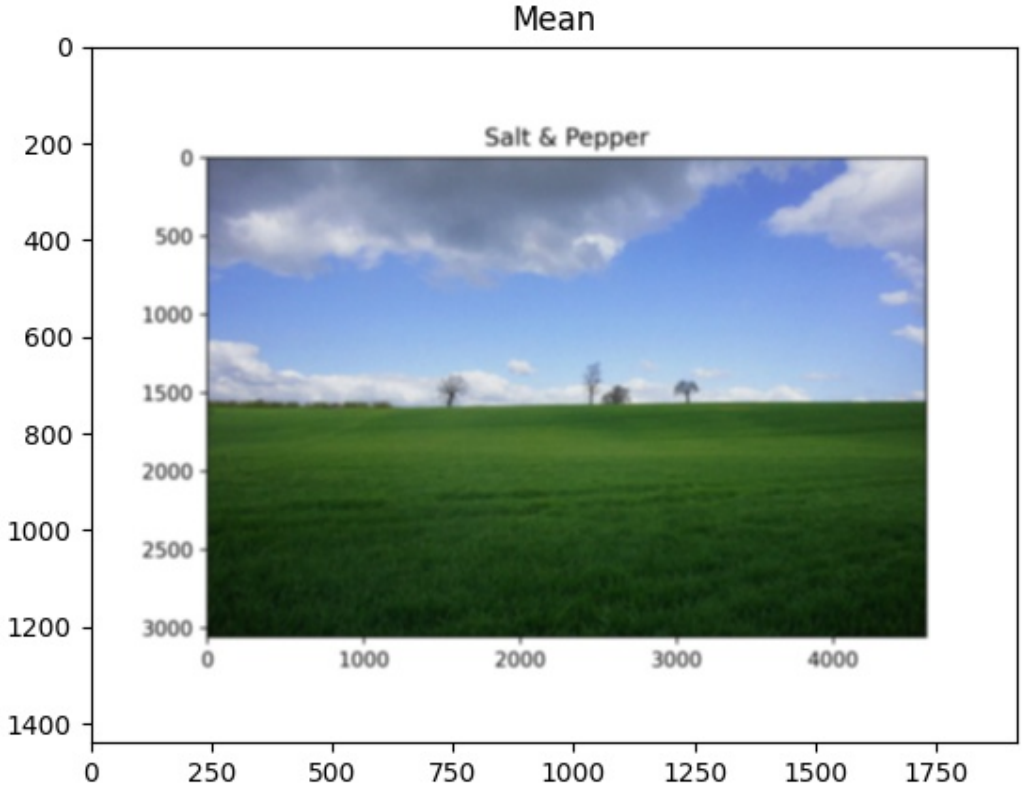

Laplacian
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

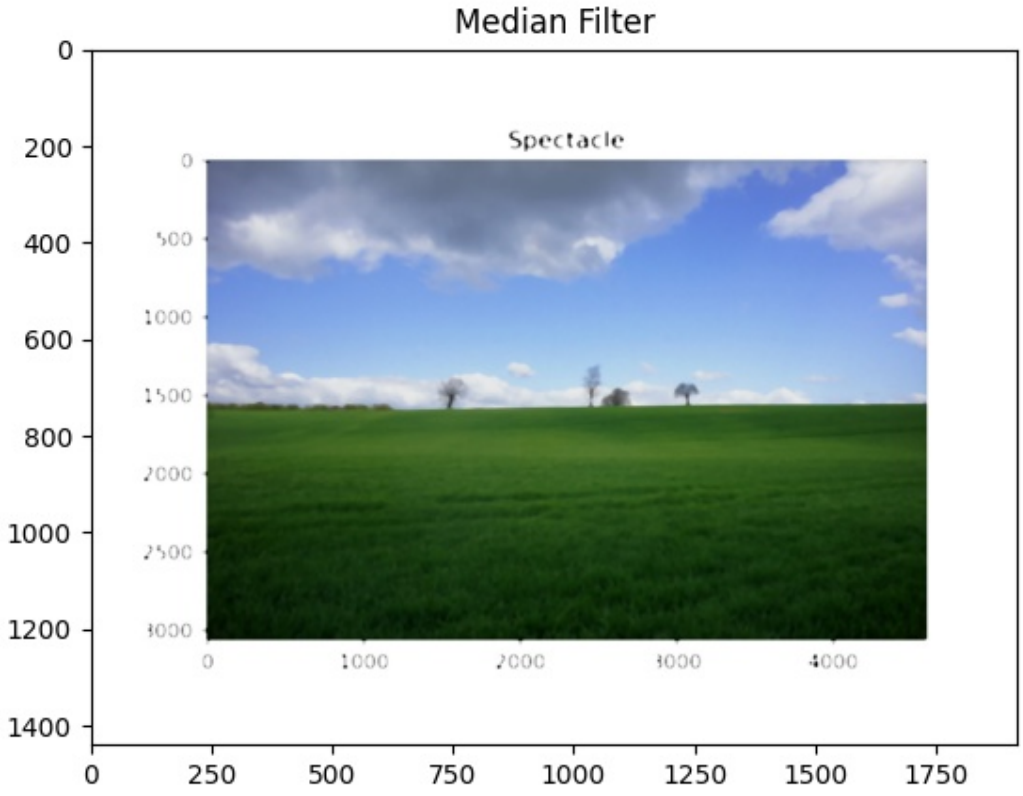

Laplacian
Salt & Pepper
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

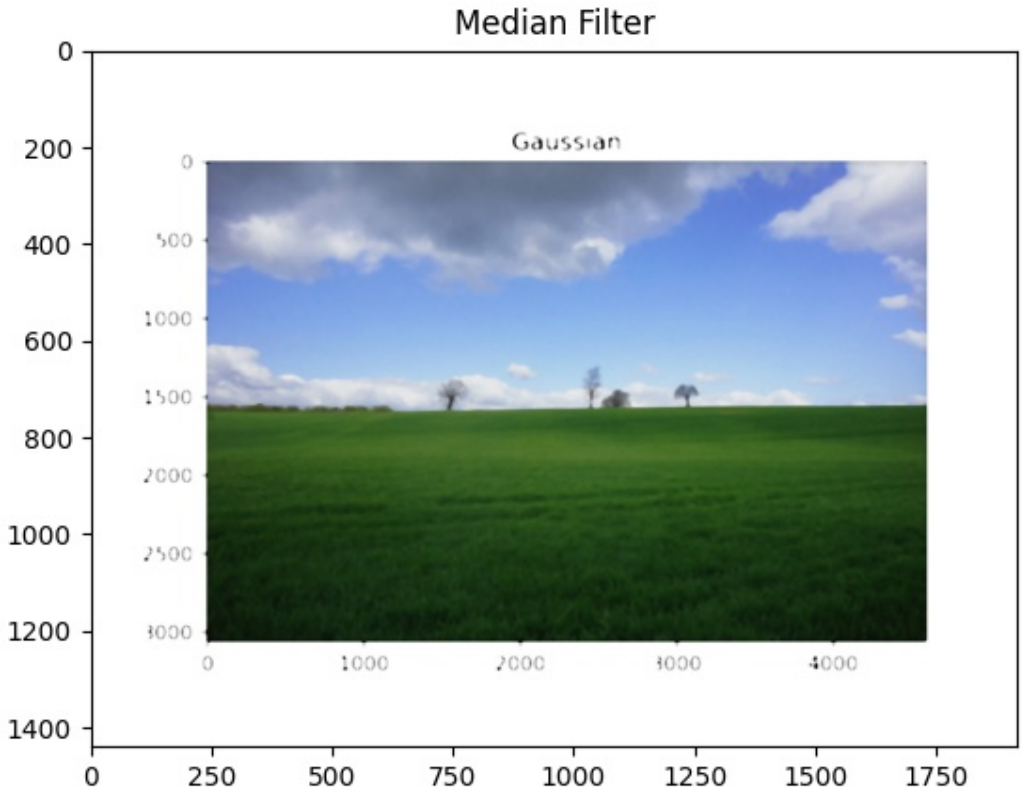

Mean
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

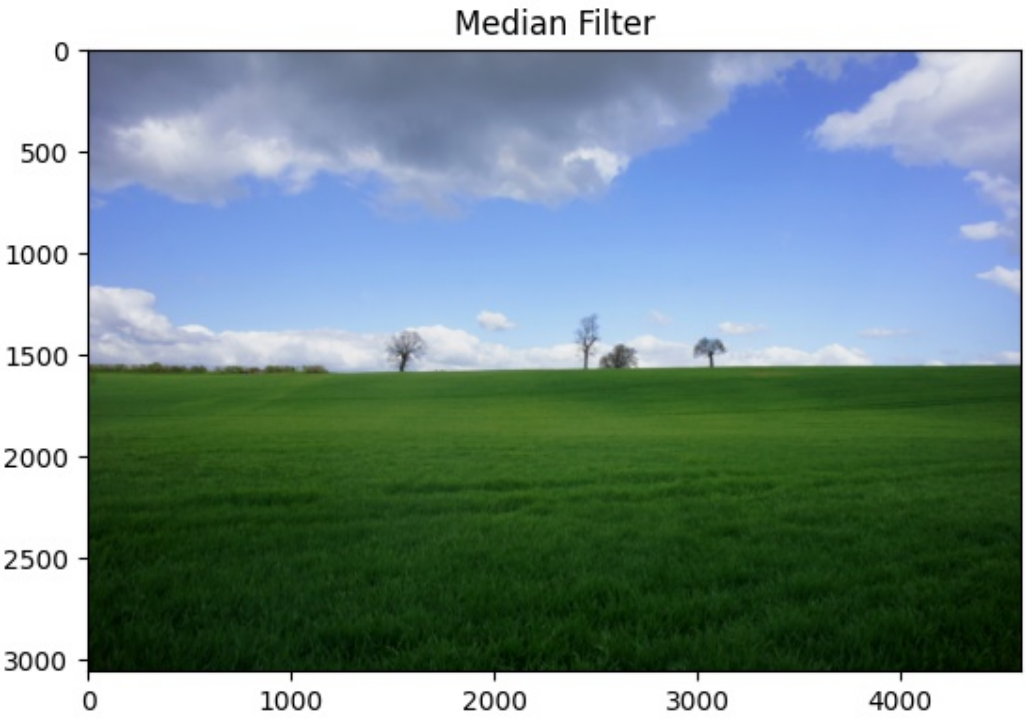

Mean
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

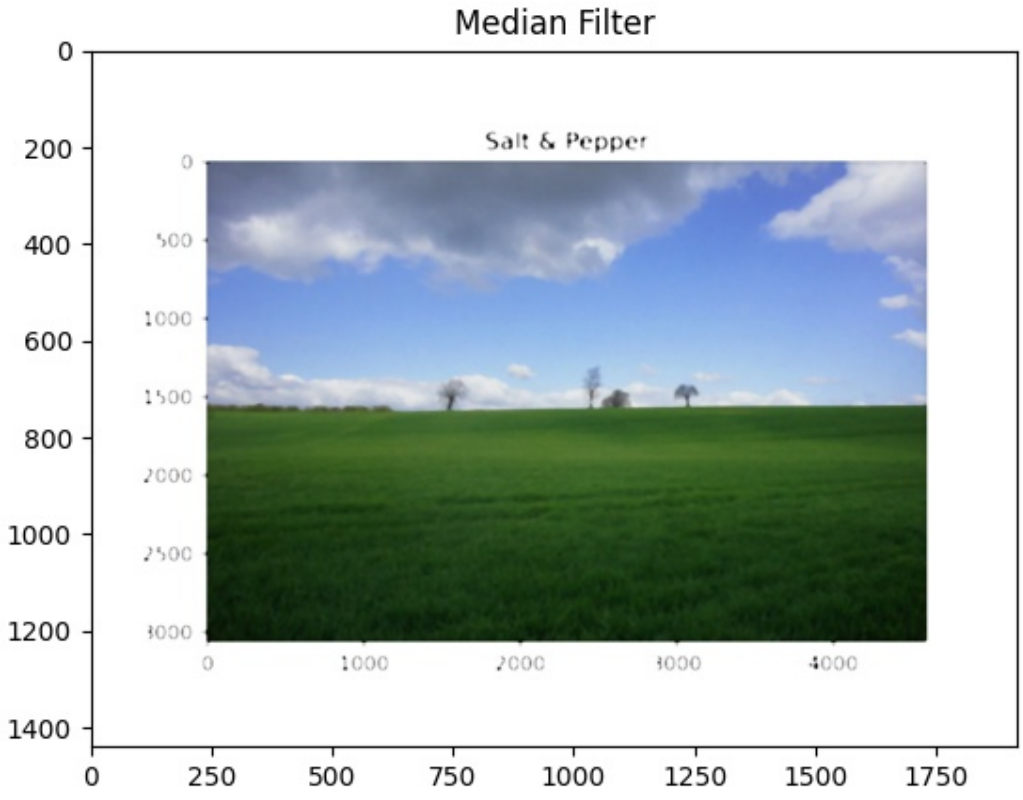

Mean
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

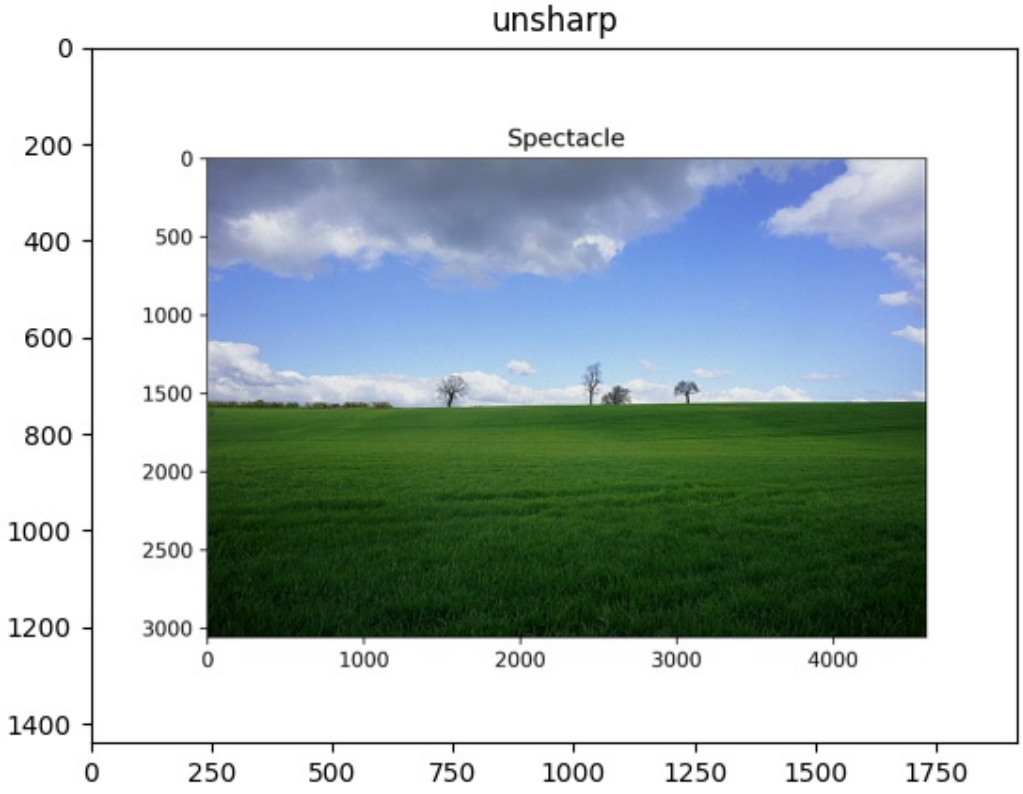

Mean
Salt & Pepper
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

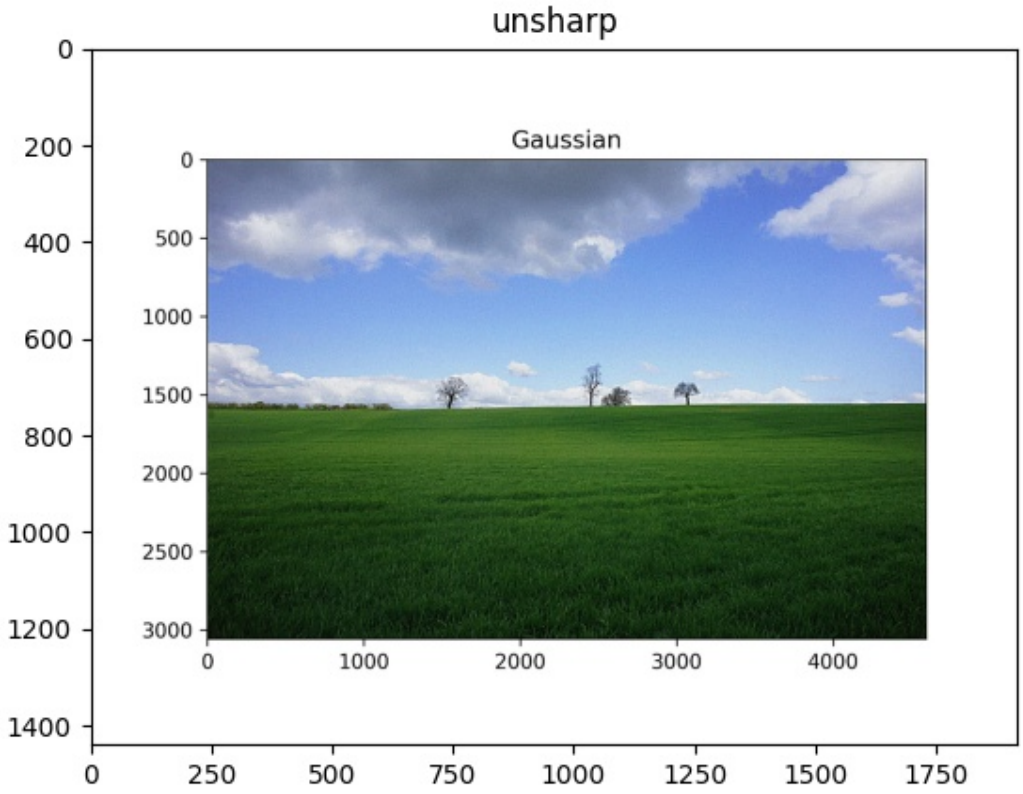

Median Filter
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

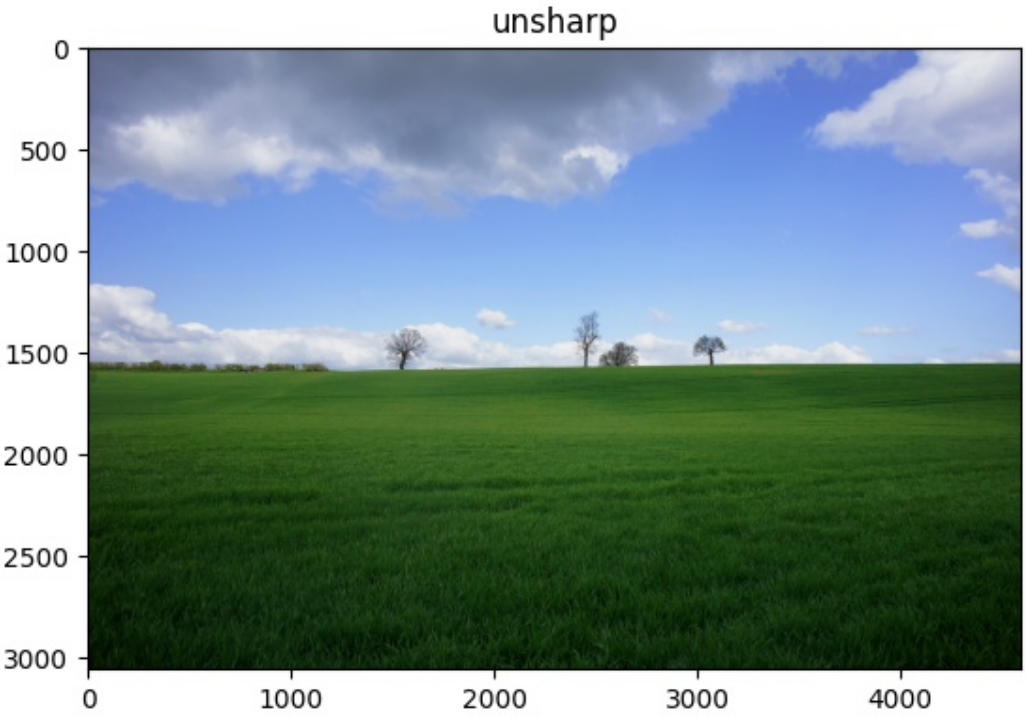

Median Filter
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

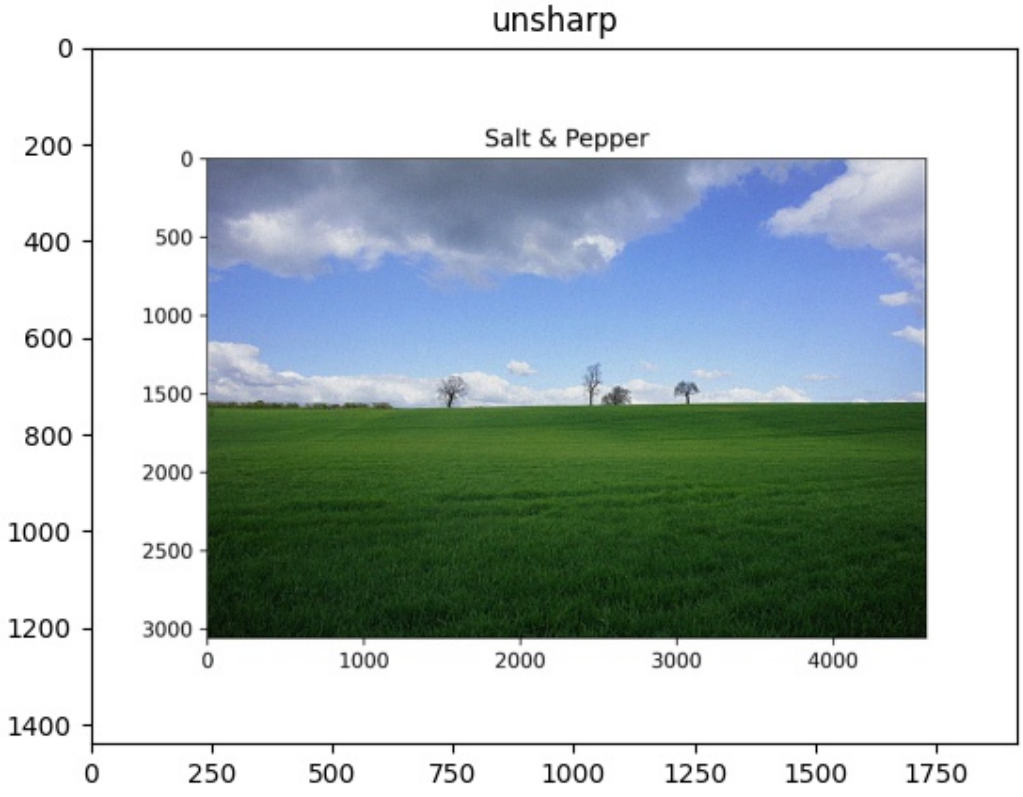

Median Filter
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

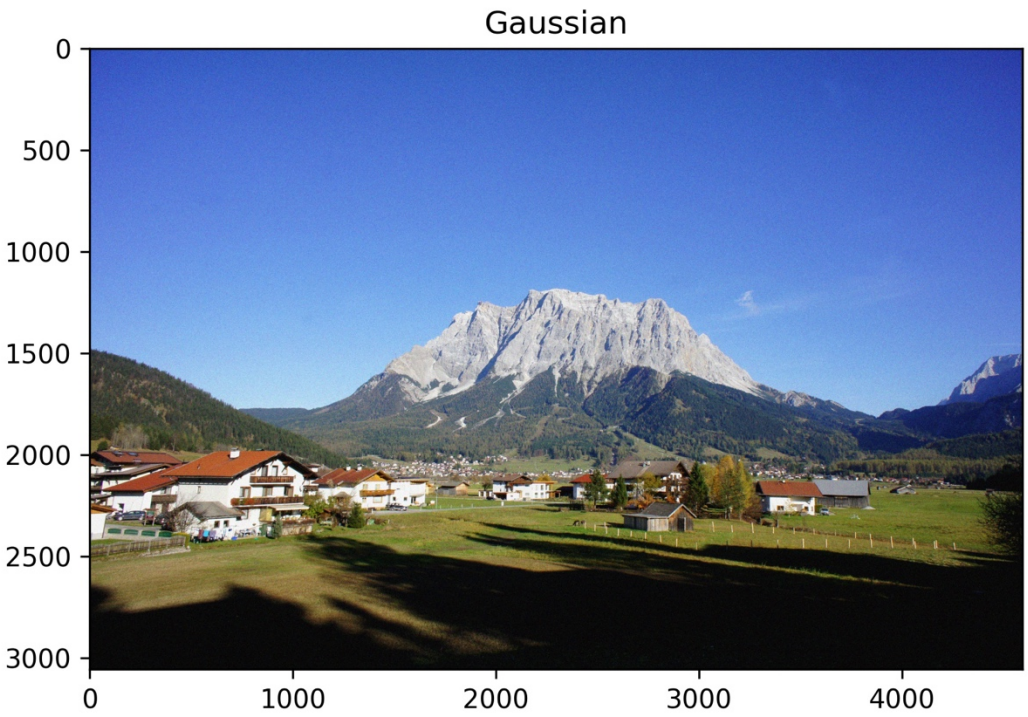

Median Filter
Salt & Pepper
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

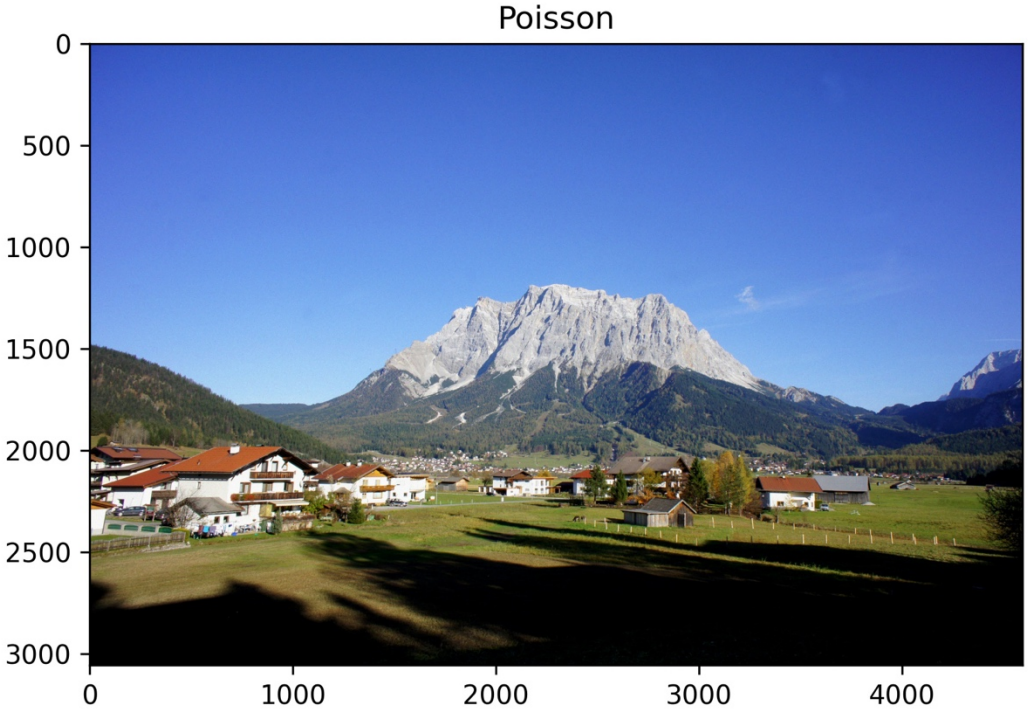

unsharp
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

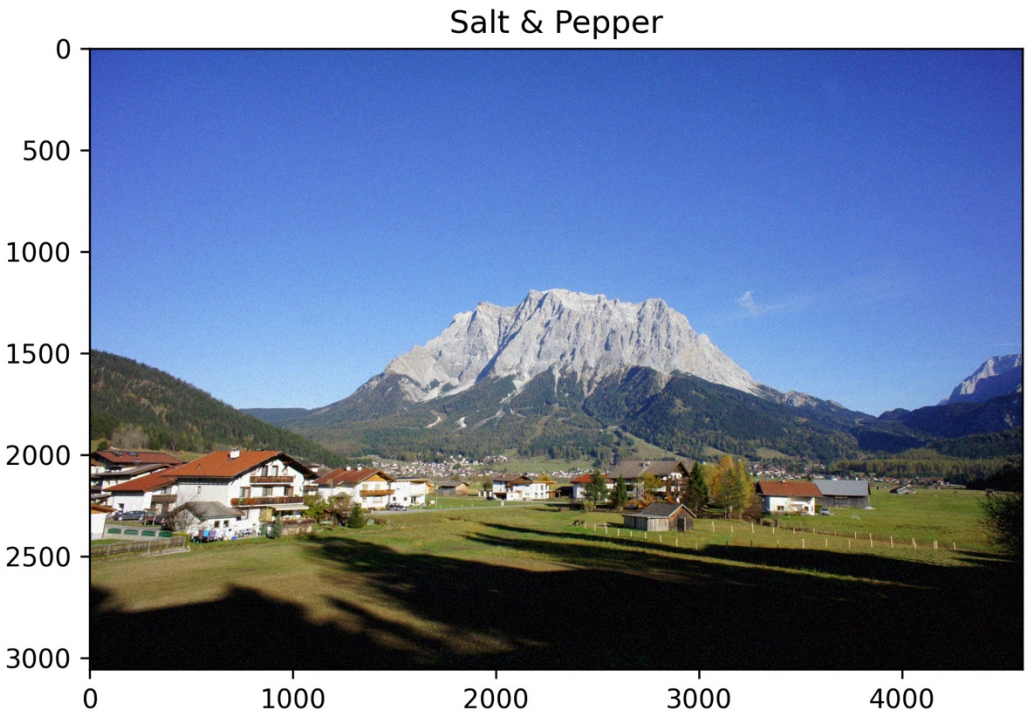

unsharp
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

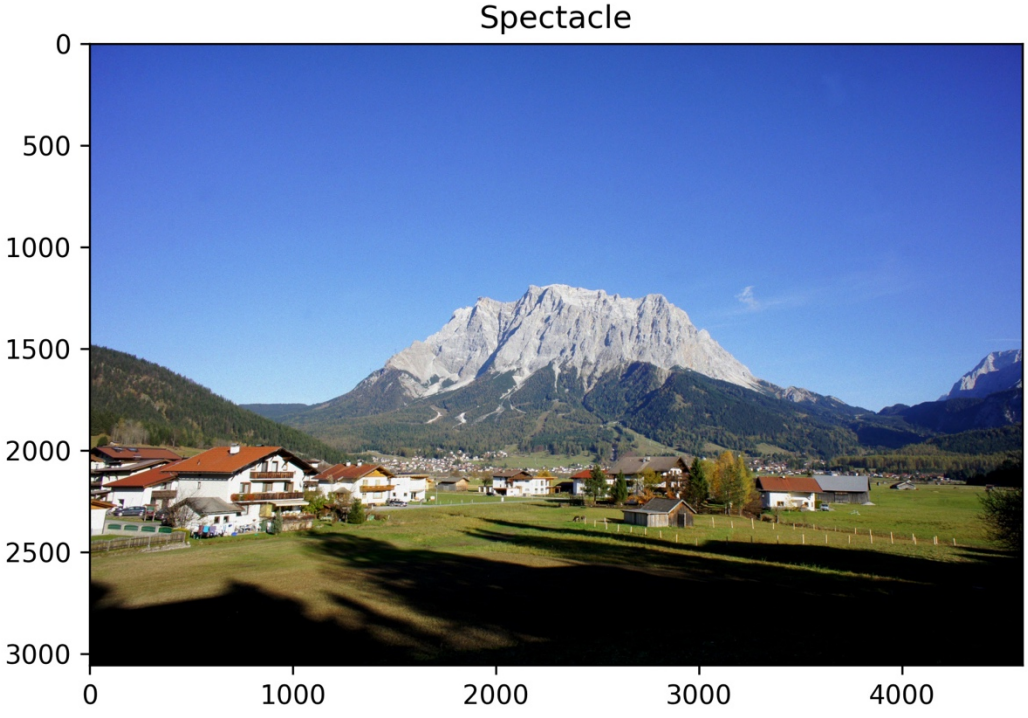

unsharp
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

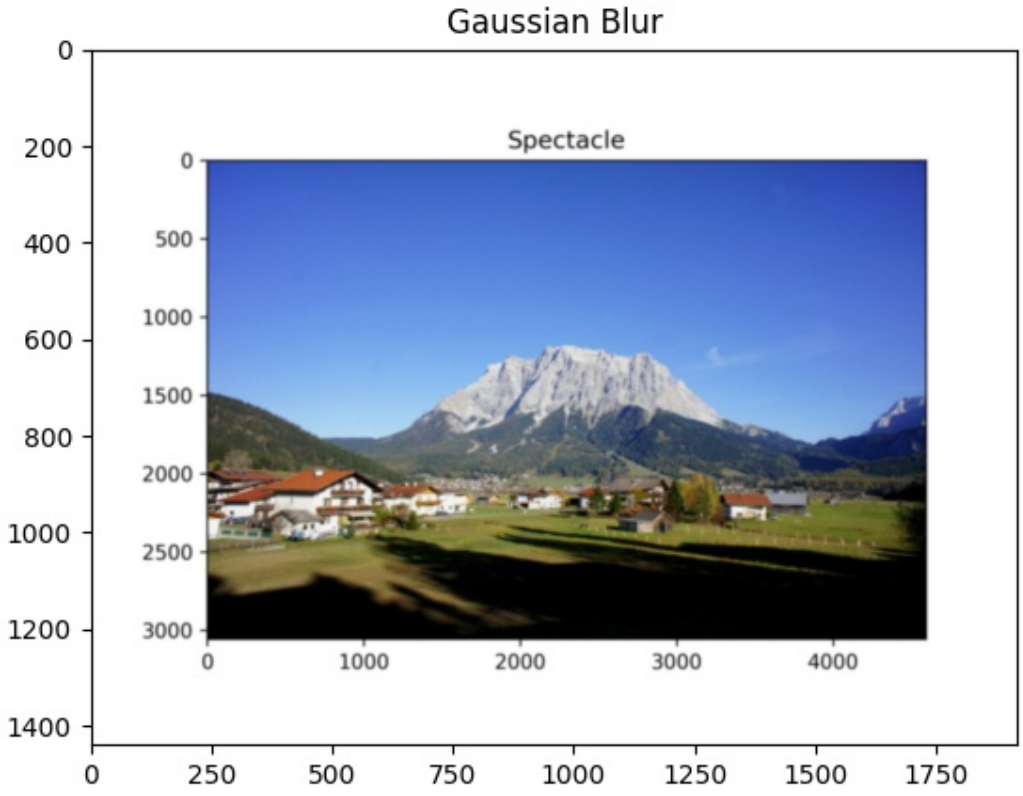

unsharp
Salt & Pepper
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

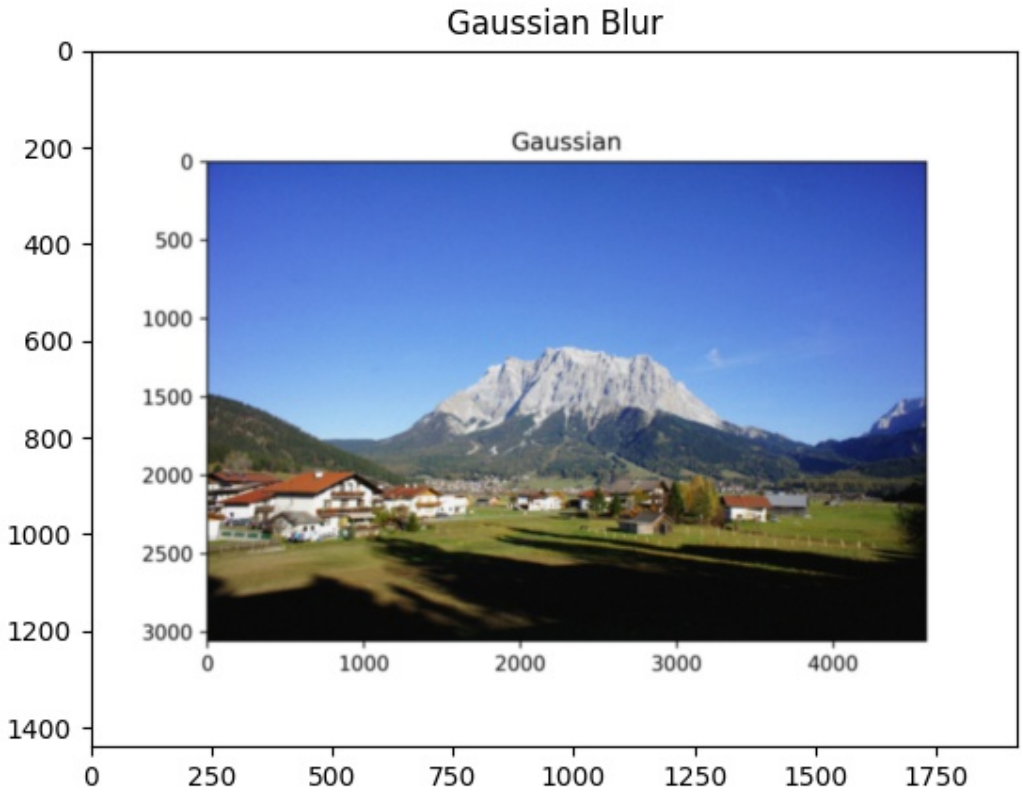

Gaussian
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

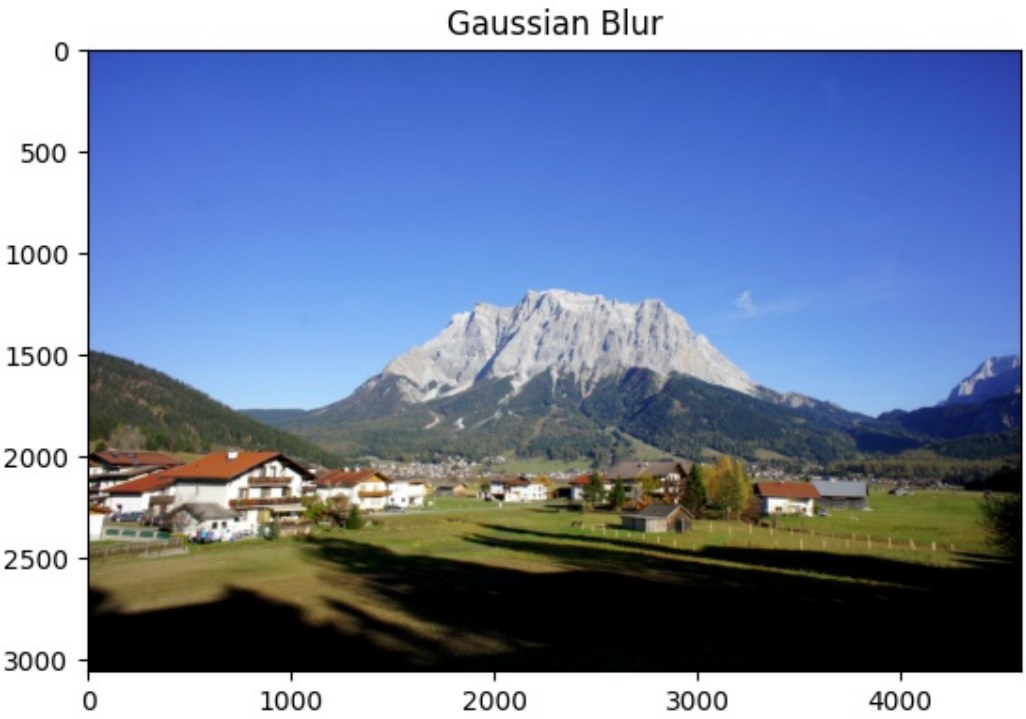

Poisson
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

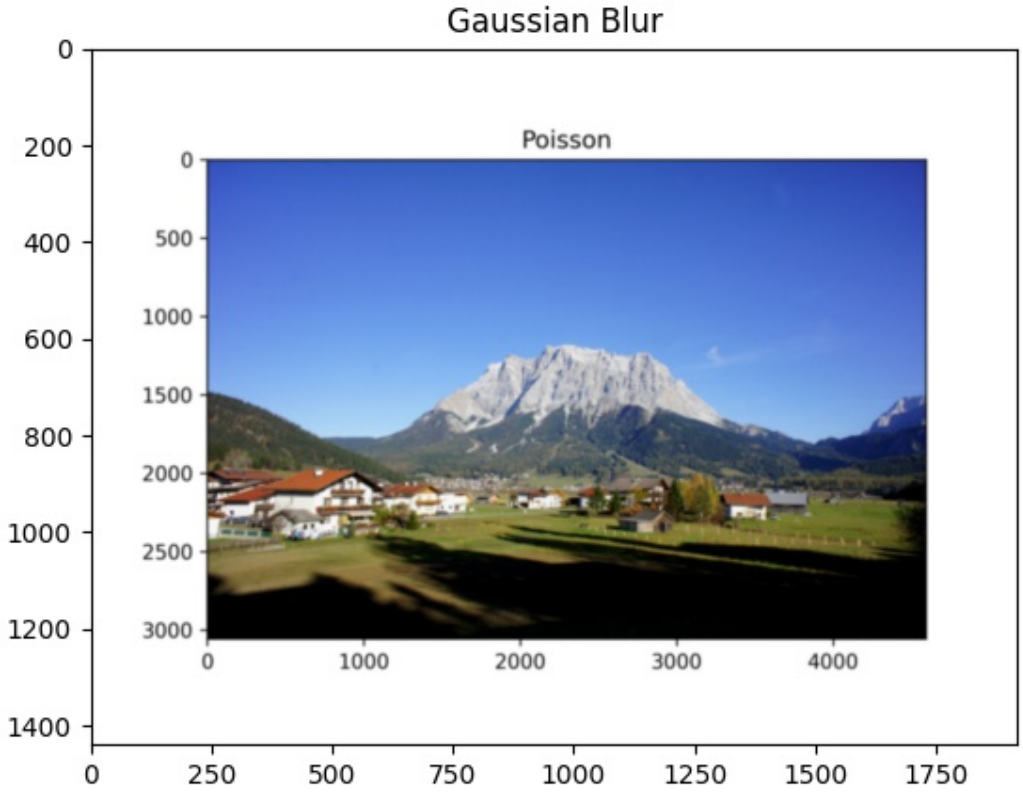

Salt & Pepper
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

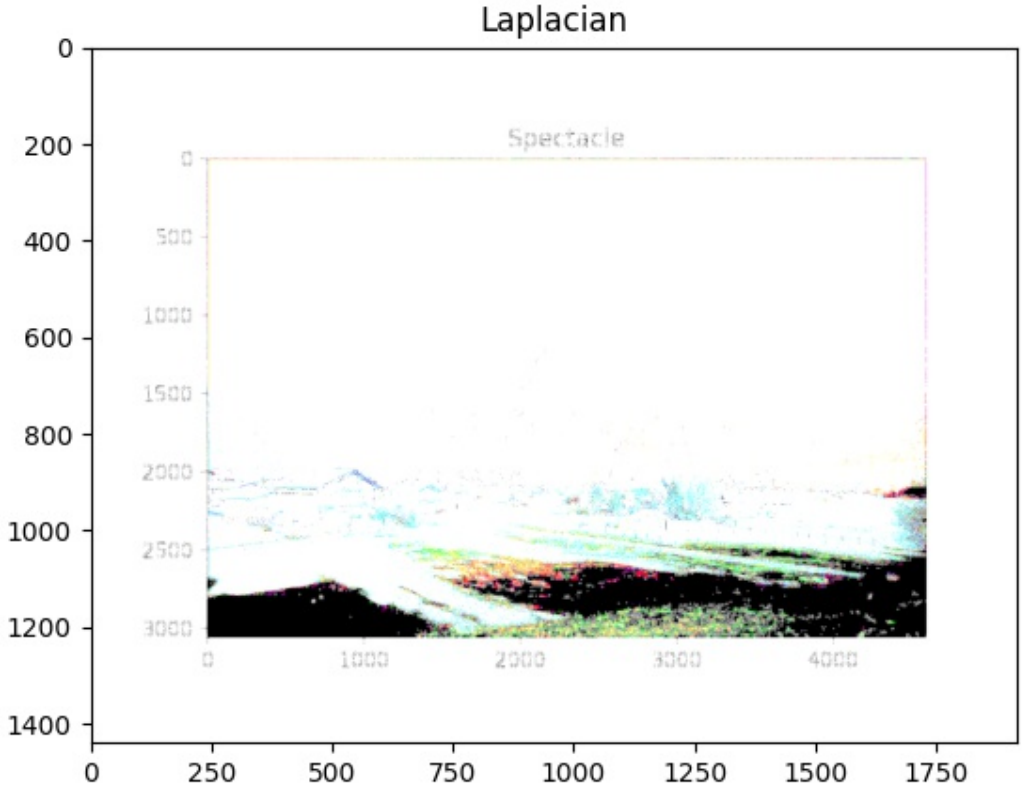

Spectacle
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

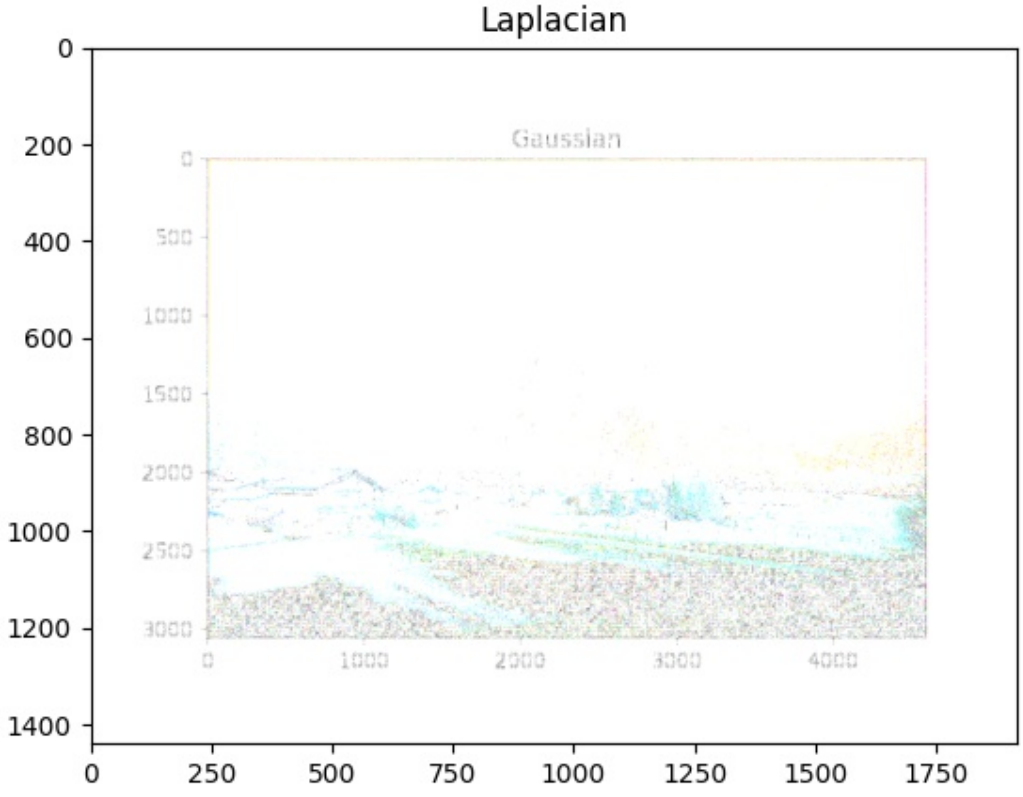

Gaussian Blur
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

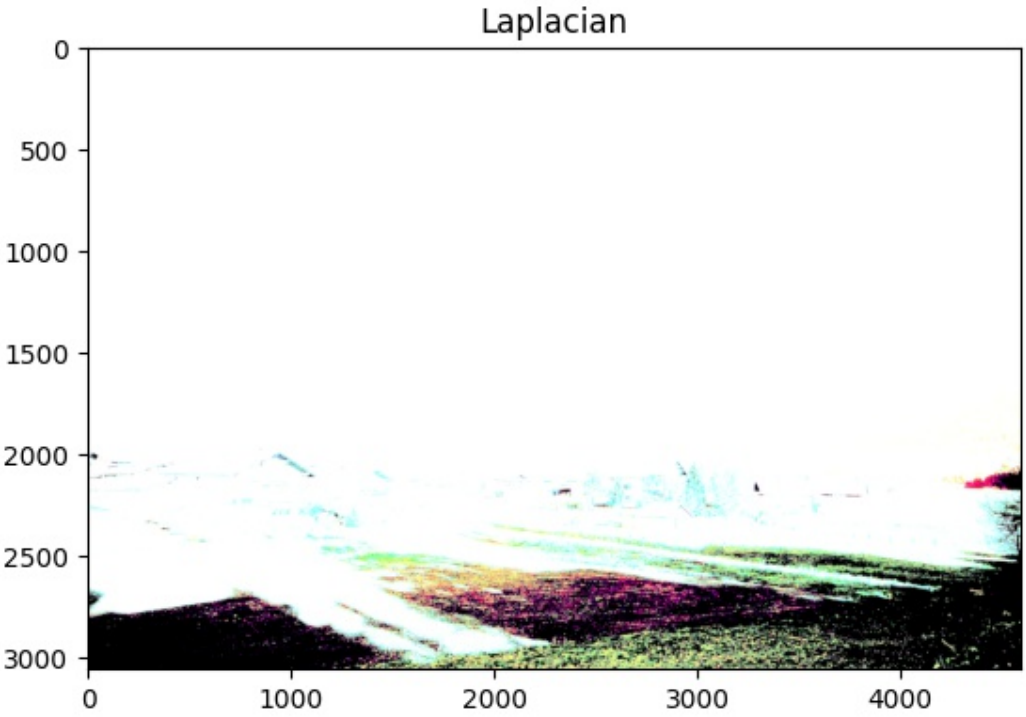

Gaussian Blur
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

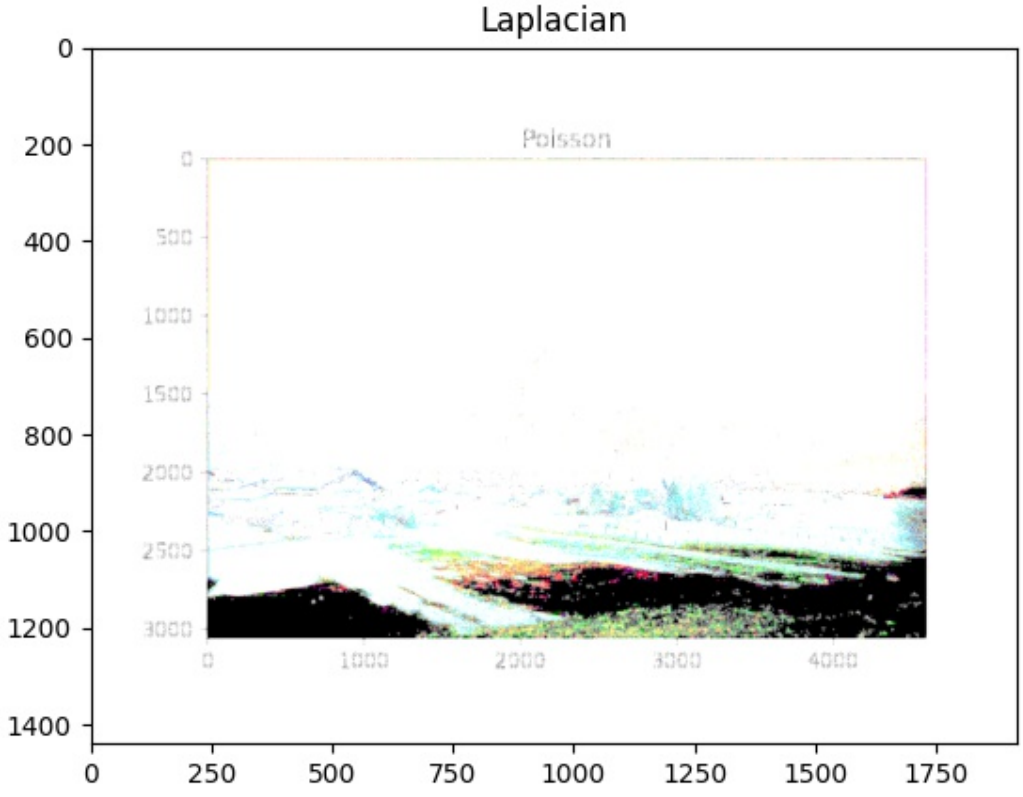

Gaussian Blur
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

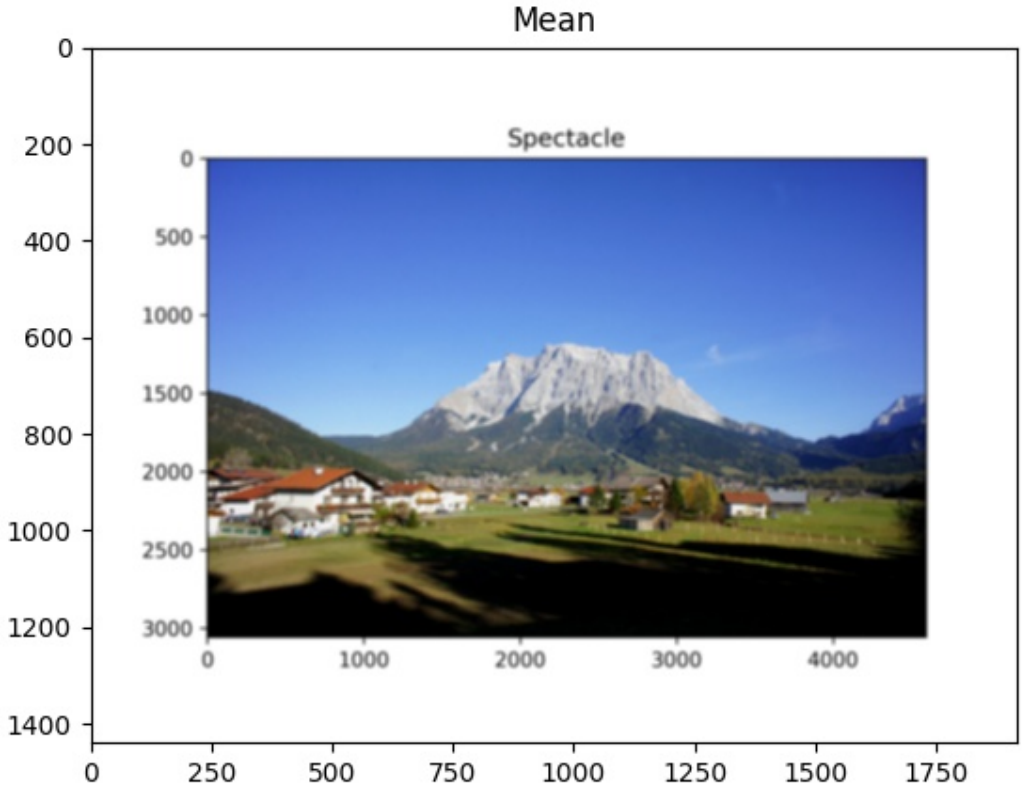

Gaussian Blur
Poisson
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

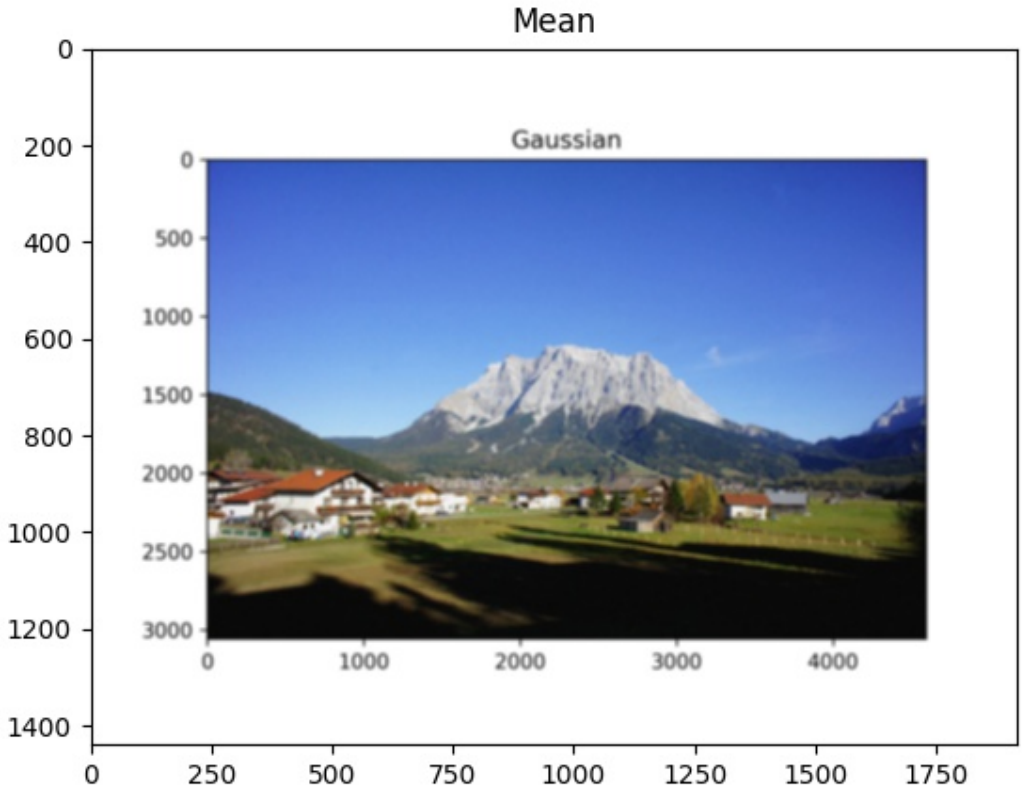

Laplacian
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

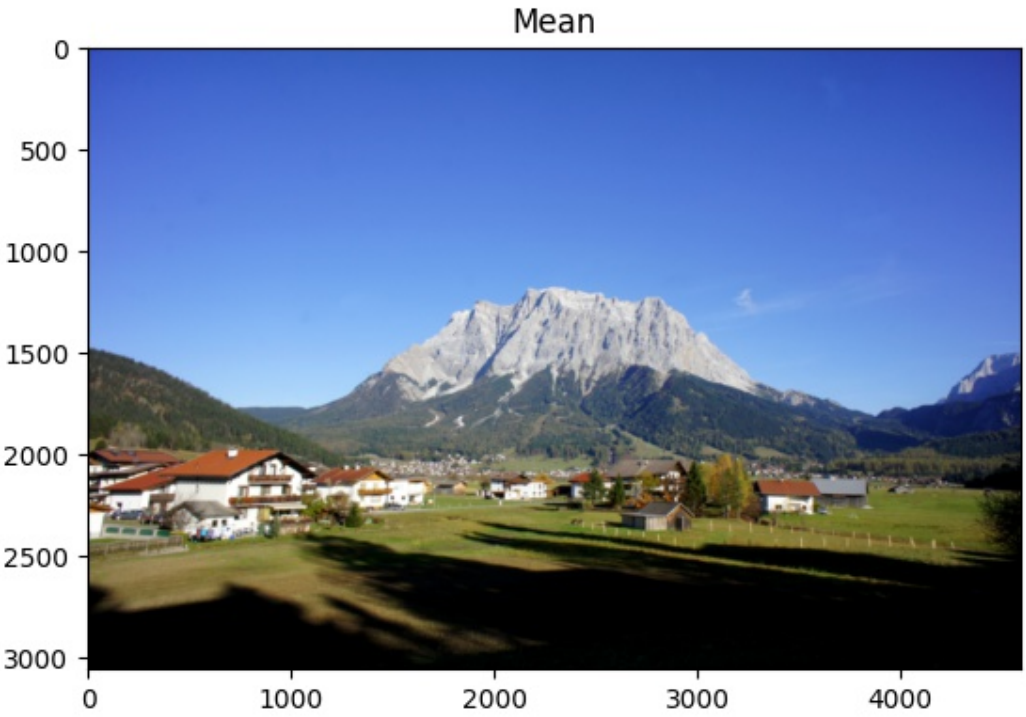

Laplacian
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

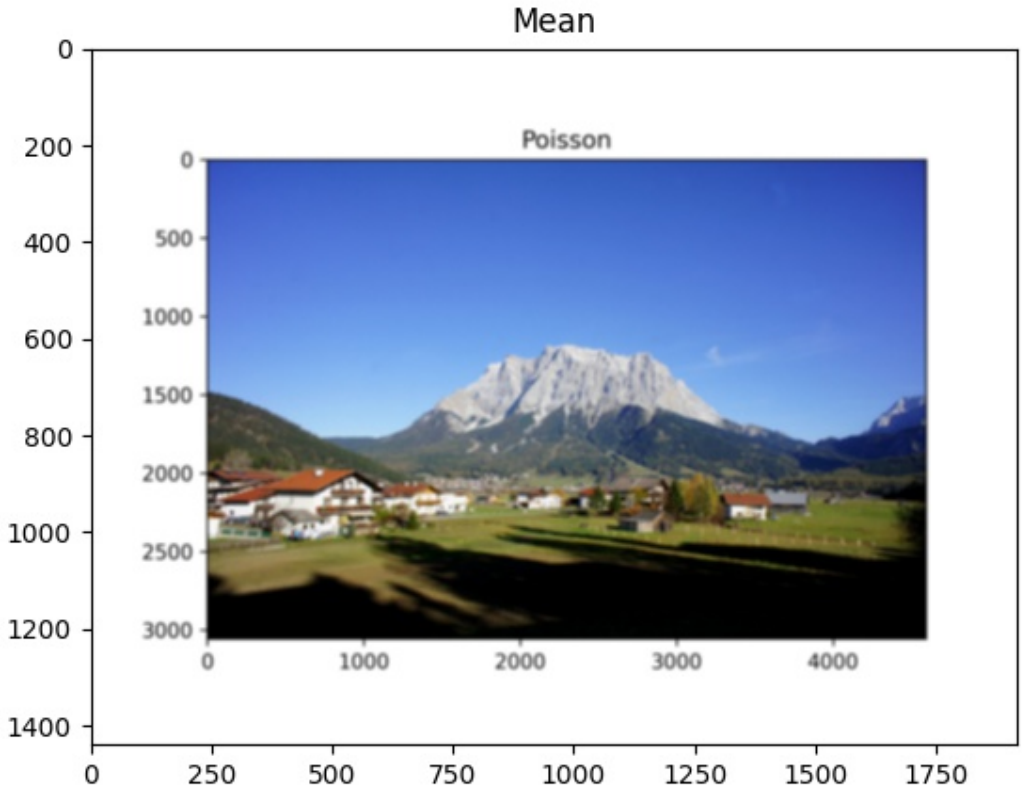

Laplacian
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

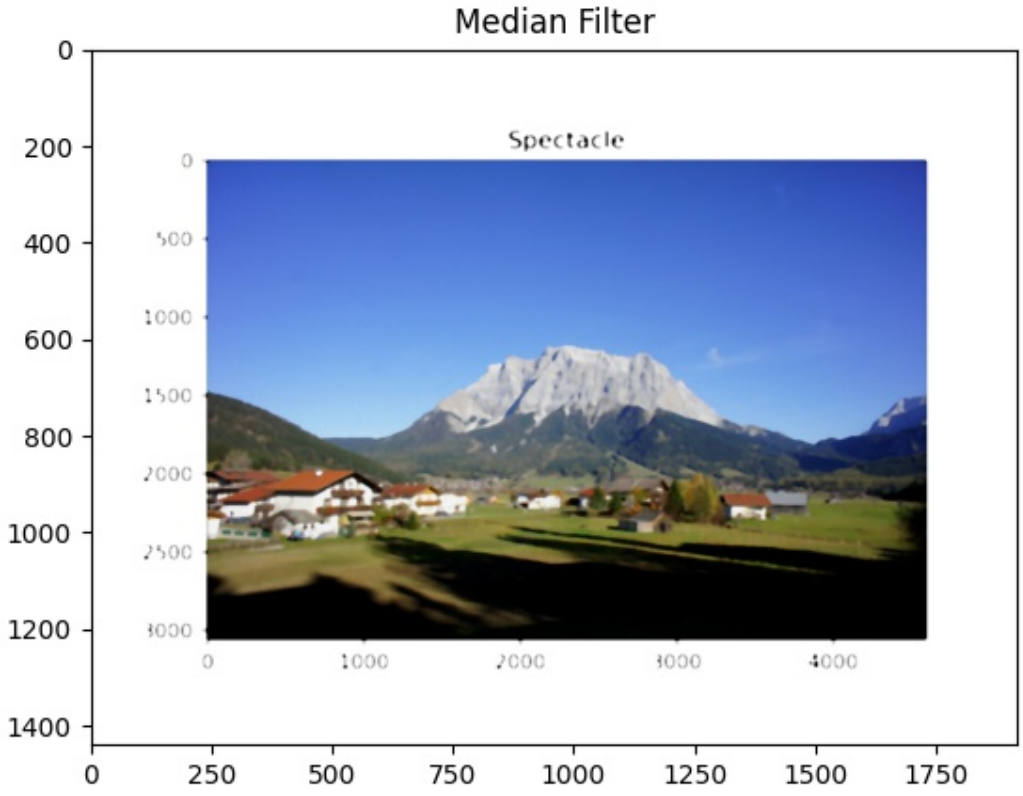

Laplacian
Poisson
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

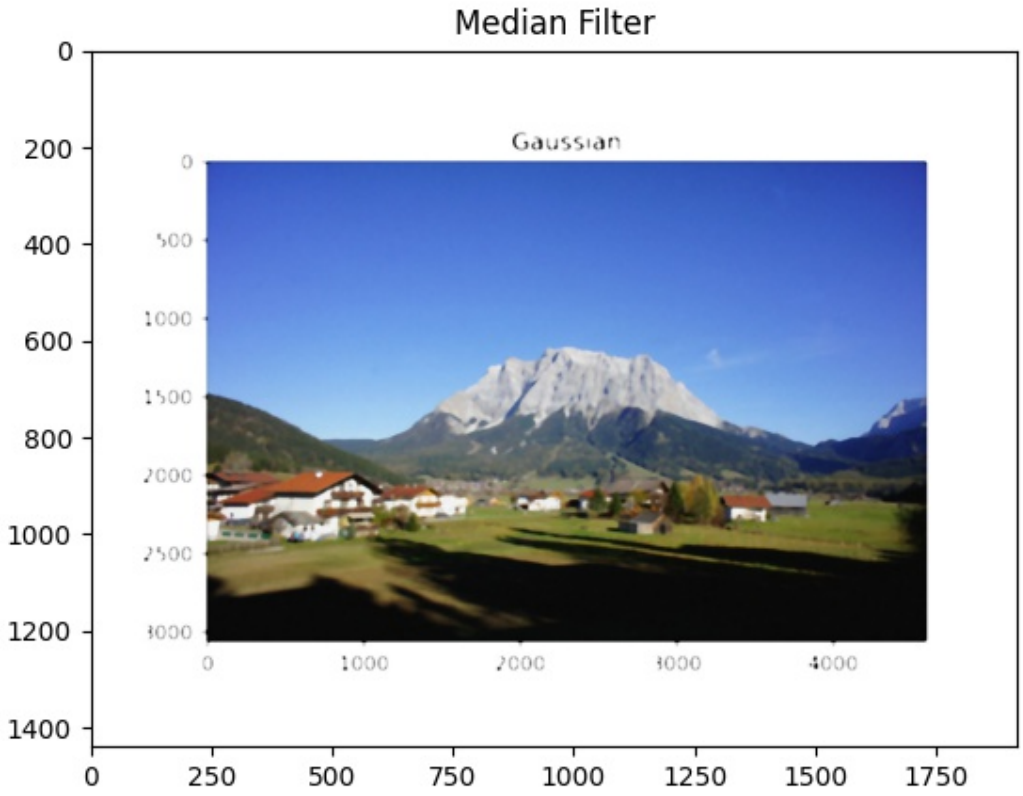

Mean
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

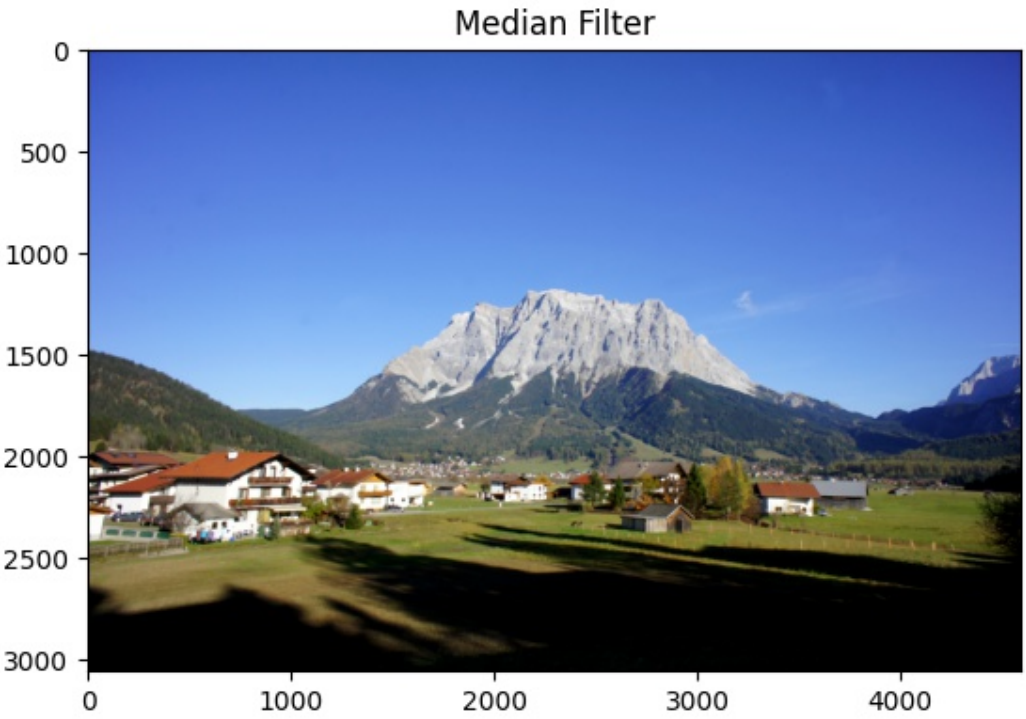

Mean
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

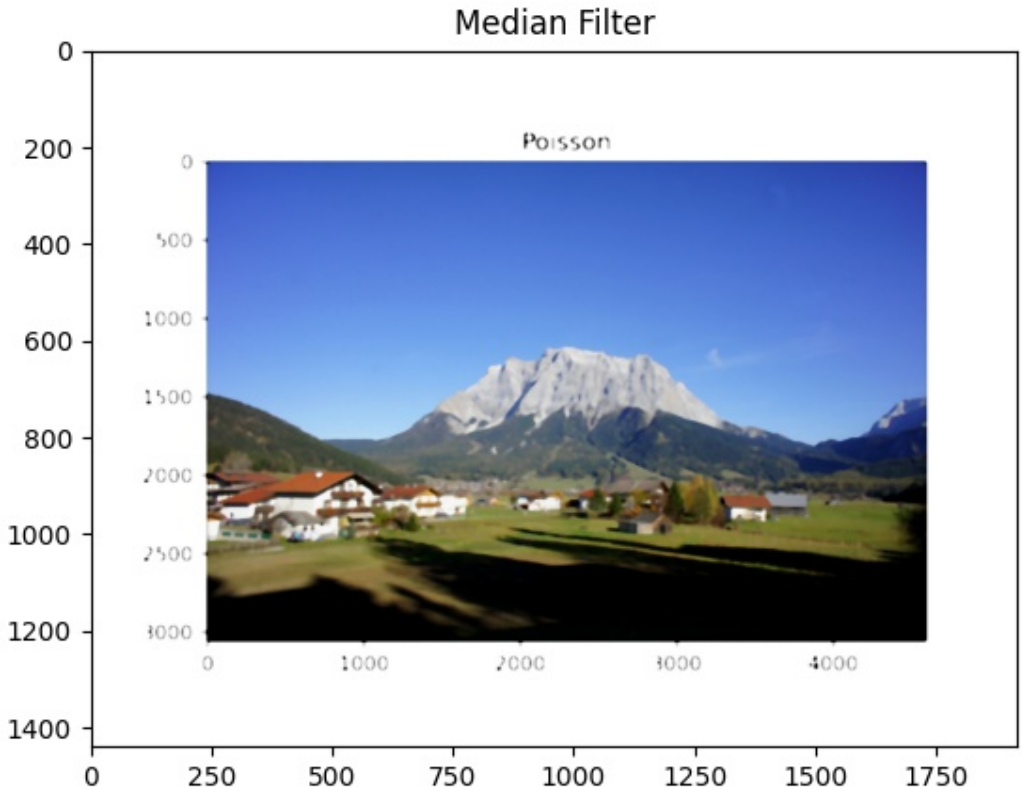

Mean
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

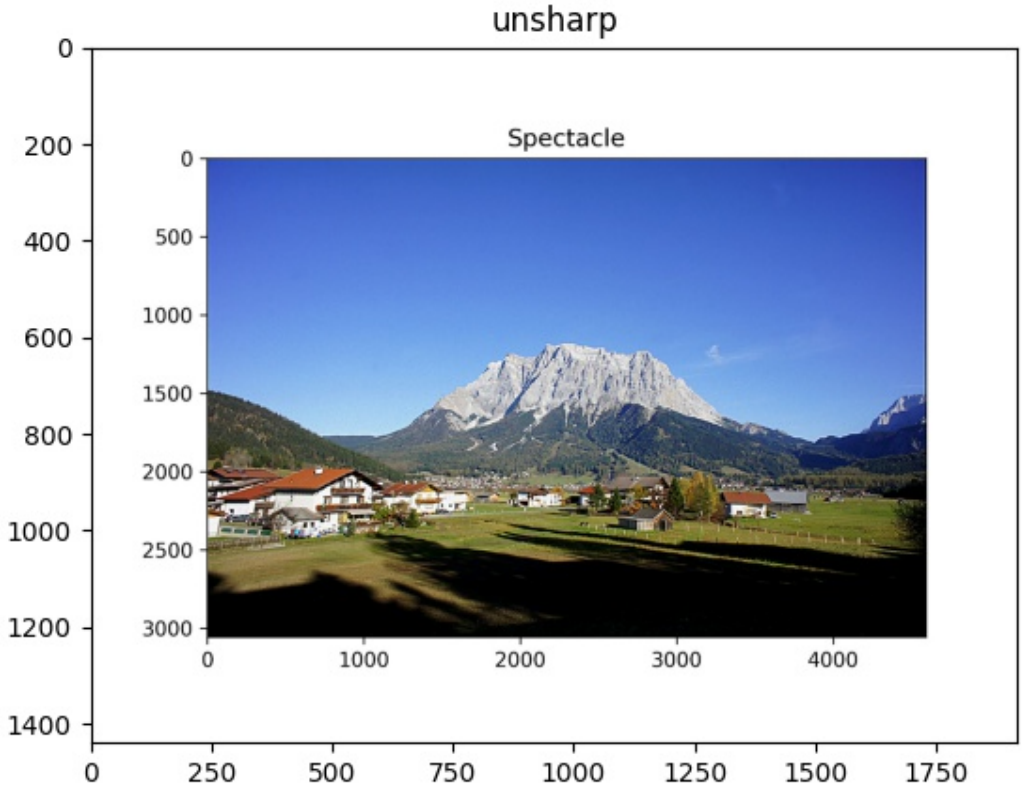

Mean
Poisson
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

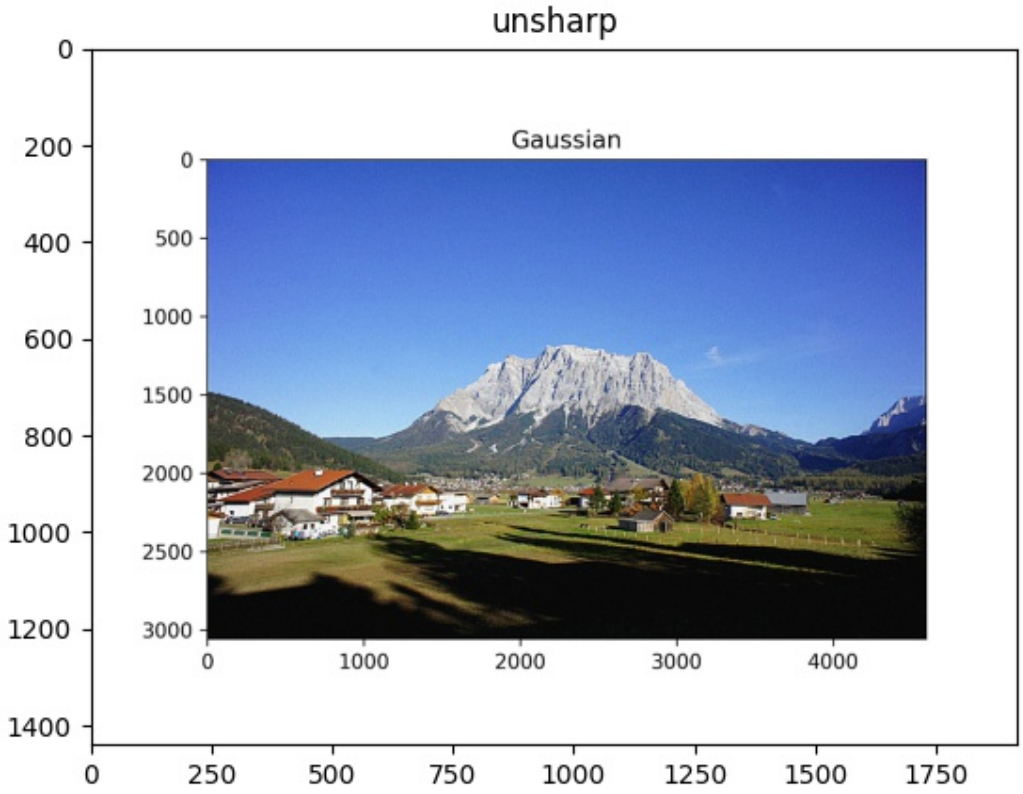

Median Filter
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

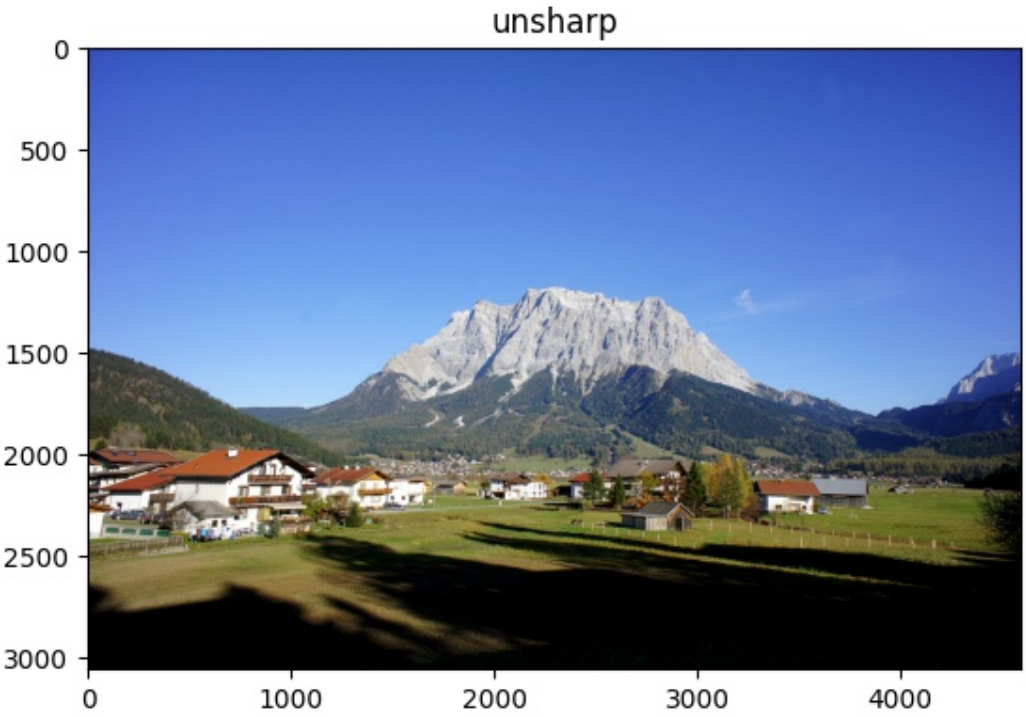

Median Filter
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750

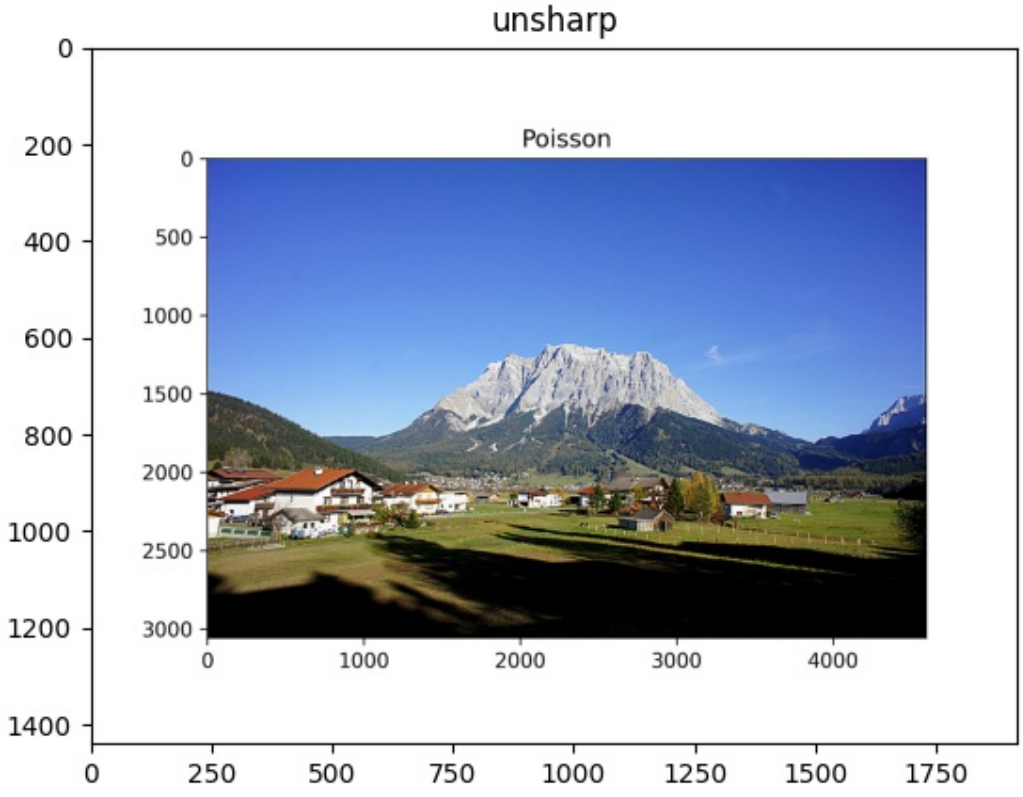

Median Filter
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000

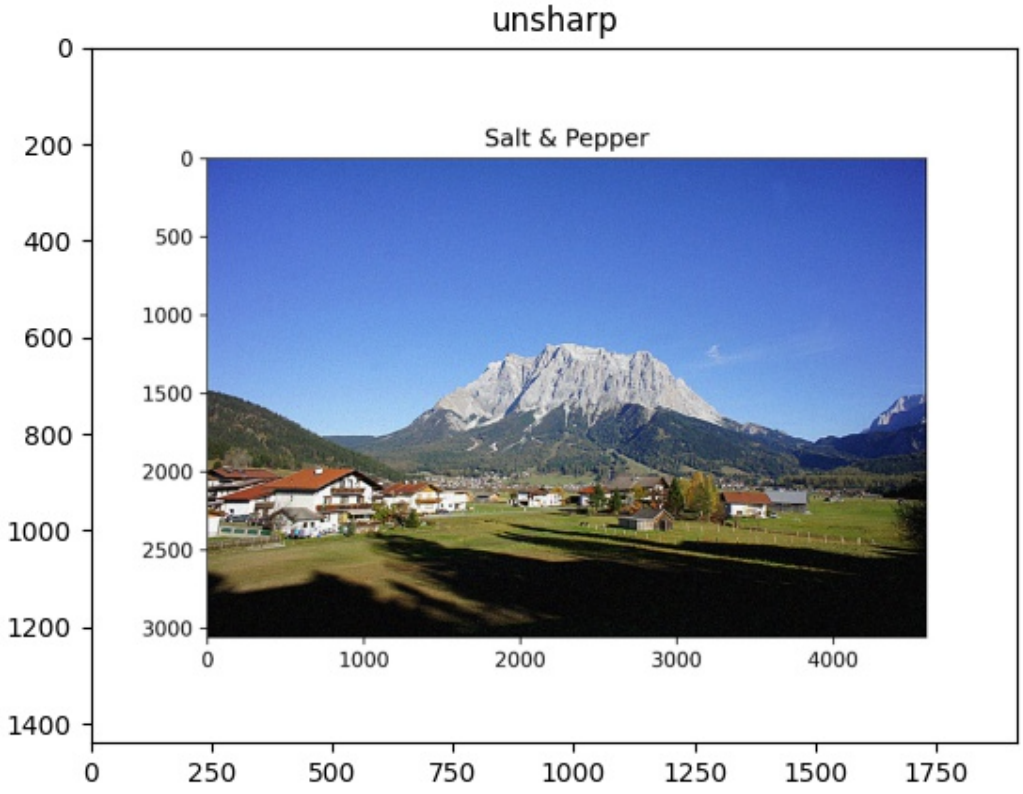

Median Filter
Poisson
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750


unsharp
Spectacle
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000


unsharp
Gaussian
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000


unsharp
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000


unsharp
Poisson
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000


unsharp
Salt & Pepper
0
200
400
600
800
1000
1200
1400
0
250
500
750
1000
1250
1500
1750
0
500
1000
1500
2000
2500
3000
0
1000
2000
3000
4000