Asad Haroon

SP17-BCS-012

Contra Harmonic Mean Filter

Note: I am Using Scipy v1.1.0. In case of any errors you can install scipy 1.1.0 by the following command: pip install scipy==1.1.0

In [24]:
```
import scipy
scipy.__version__
```

Out[24]:
```
'1.1.0'
```

To download images Click on this url: https://drive.google.com/drive/folders/1pcaTwofZGfoCxZ3Hv2X6vW6xf_1i88eb?usp=sharing (https://drive.google.com/drive/folders/1pcaTwofZGfoCxZ3Hv2X6vW6xf_1i88eb?usp=sharing)

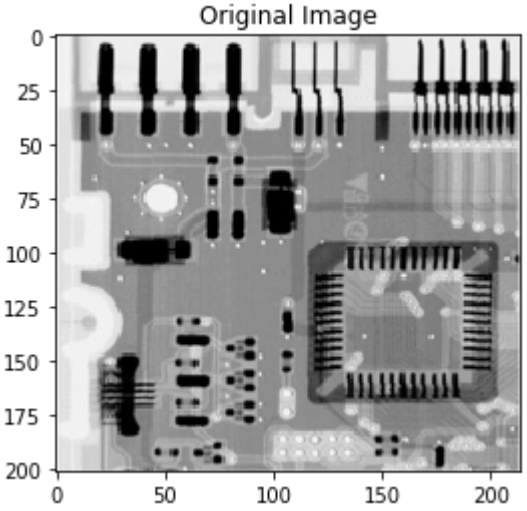
Import Libraries

In [17]:
```
import cv2
from scipy.misc import imread
import matplotlib.pyplot as plt
import numpy as np
from skimage.util import random_noise
from skimage.filters import rank
```

Read Image

In [18]:
```
img=imread("circuit_image.png",False,'L')
img=img.astype(np.uint8)
img_2=img.copy()
rows,cols=img.shape[:2]
plt.title("Original Image")
plt.imshow(img,plt.cm.gray)
plt.show()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: DeprecationWarning: `imread` is deprecate
d!
`imread` is deprecated in SciPy 1.0.0, and will be removed in 1.2.0.
Use ``imageio.imread`` instead.
  """Entry point for launching an IPython kernel.
```



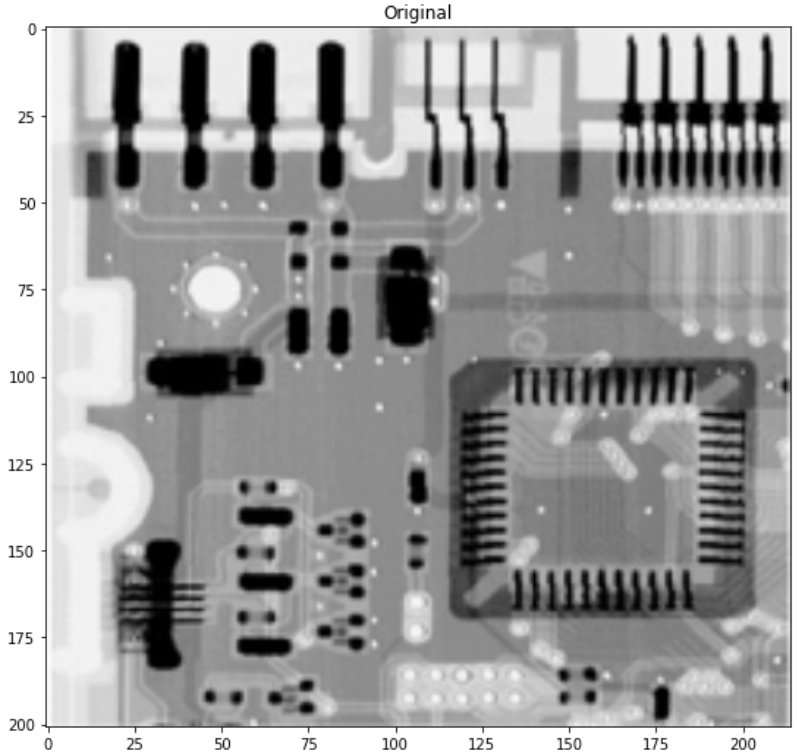
Adding Pepper Noise

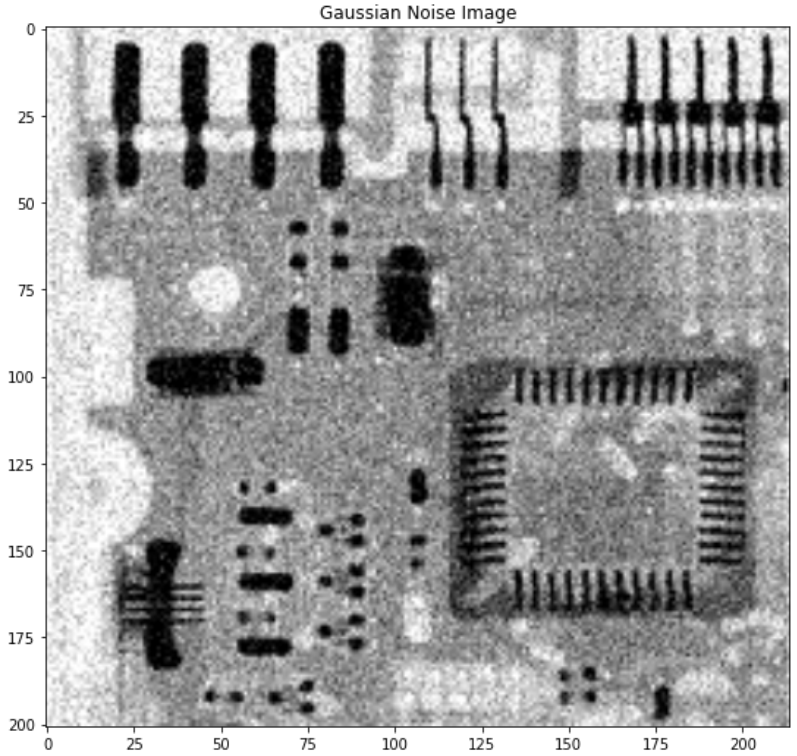
In [25]:
```
def pepper_noise(img):
    noise_img = random_noise(img, mode='pepper')
    noise_image=noise_img*255
    return noise_image
def gaussian_noise(img):
    noise_img = random_noise(img, mode='gaussian')
    noise_image=noise_img*255
    return noise_image
```

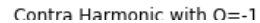
In [36]:
```
def contra_harmonic(Q=0,img=[]):
    rows,cols=img_2.shape[:2]
    img_contra_harmonic=np.zeros((rows,cols))
    for i in range(1,rows-1):
        for j in range(1,cols-1):
            ans=img[i-1:i+2,j-1:j+2]
            numerator=ans**(Q+1)
            if Q==-1:
                denominator=1/ans
            else:
                denominator=ans**(Q)
            ans1=np.sum(numerator)
            ans2=np.sum(denominator)
            ans3=ans1/ans2
            ans=round(ans3)
            img_contra_harmonic[i,j]=ans
    #print(img_contra_harmonic[1:5,1:5])
    return img_contra_harmonic
```

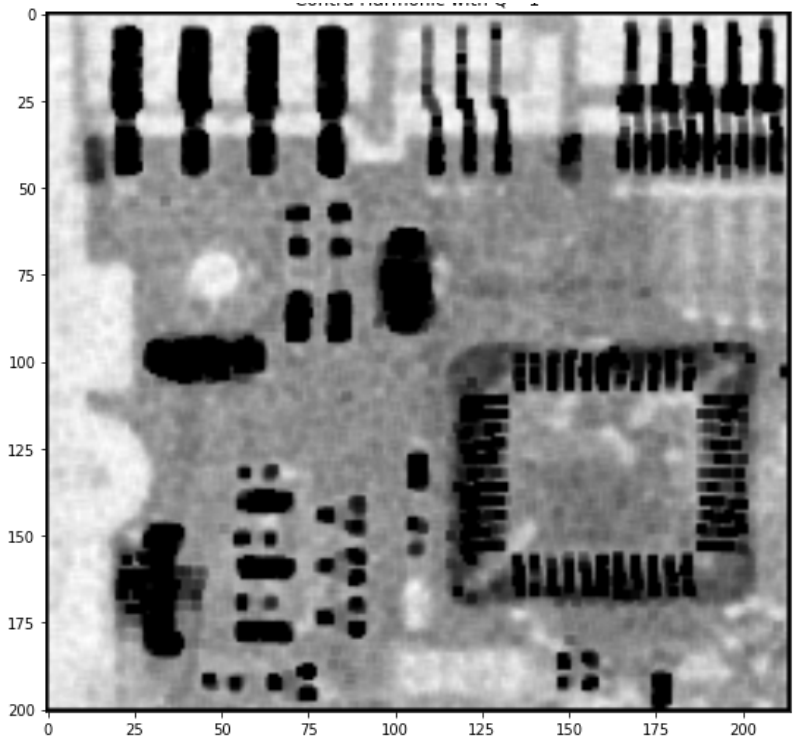
In [37]:
```
img_2=gaussian_noise(img)
a=contra_harmonic(Q=-1,img=img_2)
fig, axes = plt.subplots(2, 2,figsize=(20,20))
ax = axes.ravel()

ax[0].imshow(img,cmap=plt.cm.gray,interpolation='bilinear')
ax[0].set_title("Original")
ax[1].imshow(noise_image, cmap=plt.cm.gray,interpolation='bilinear')
ax[1].set_title("Gaussian Noise Image")
ax[2].imshow(a, cmap=plt.cm.gray,interpolation='bilinear')
ax[2].set_title("Contra Harmonic with Q=-1")
plt.show()
```

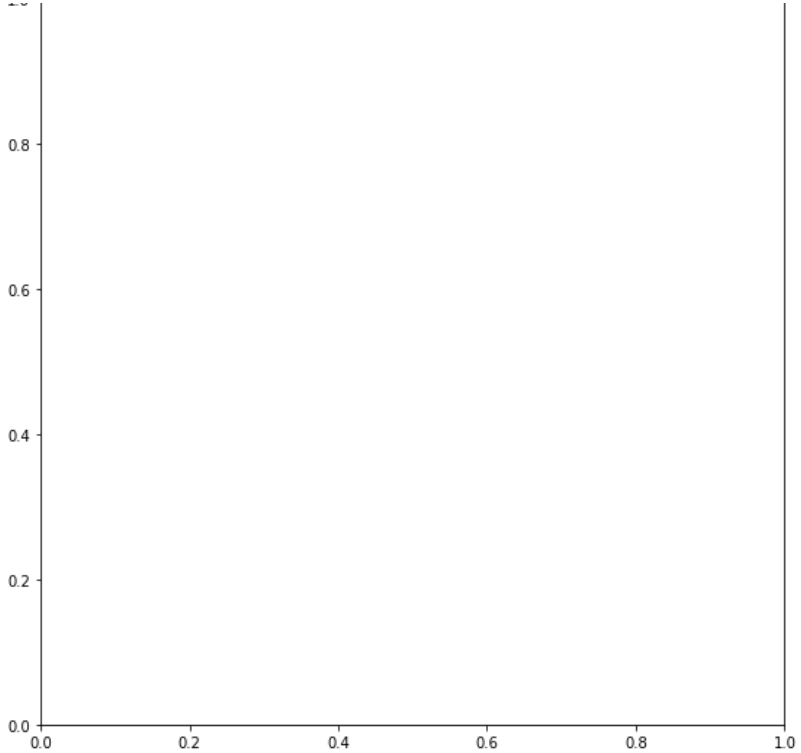
```
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:9: RuntimeWarning: divide by zero encountered
in true_divide
  if __name__ == '__main__':
```





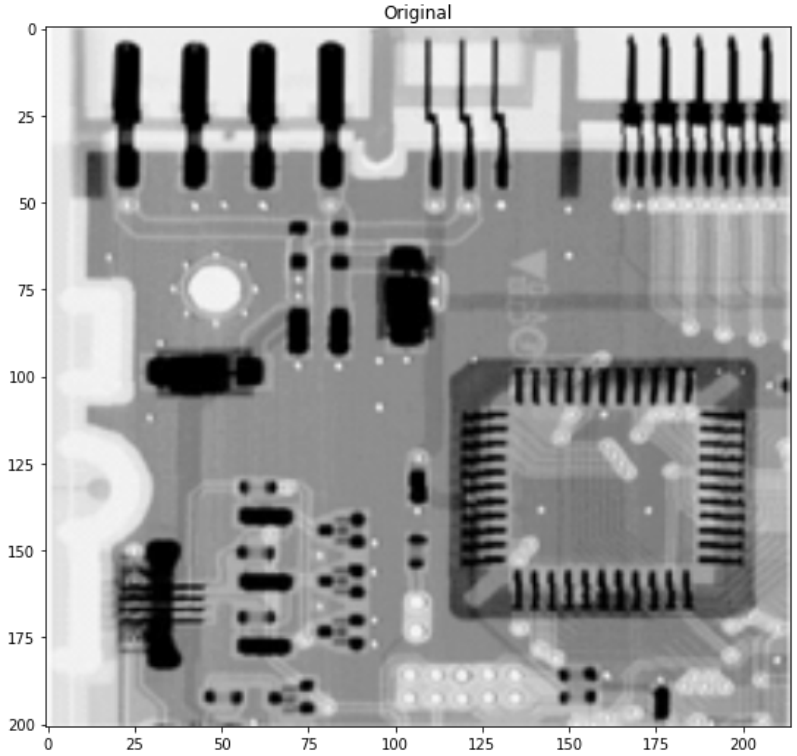


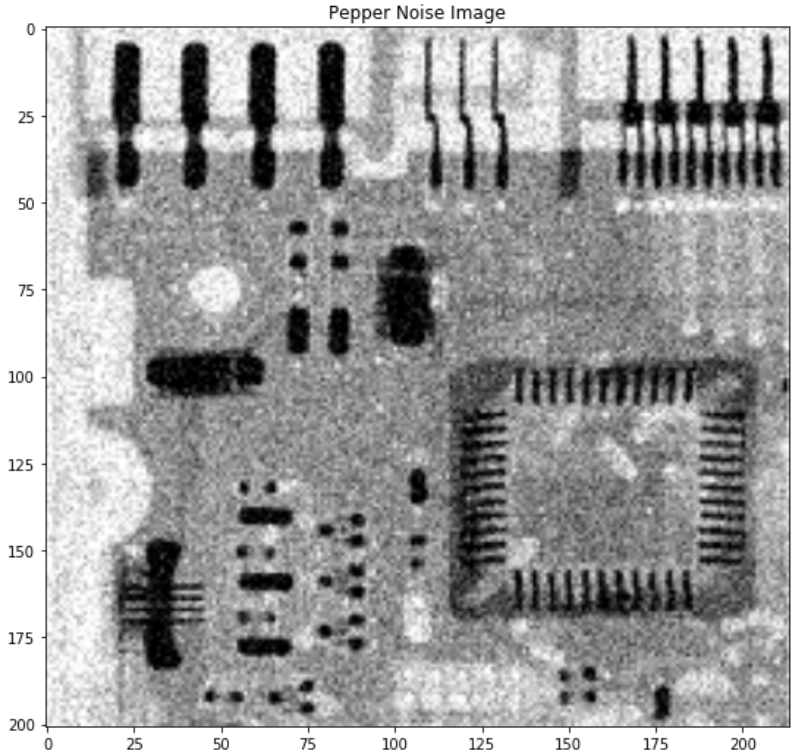

0
25
50
75
100
125
150
175
200
0
25
50
75
100
125
150
175
200


0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

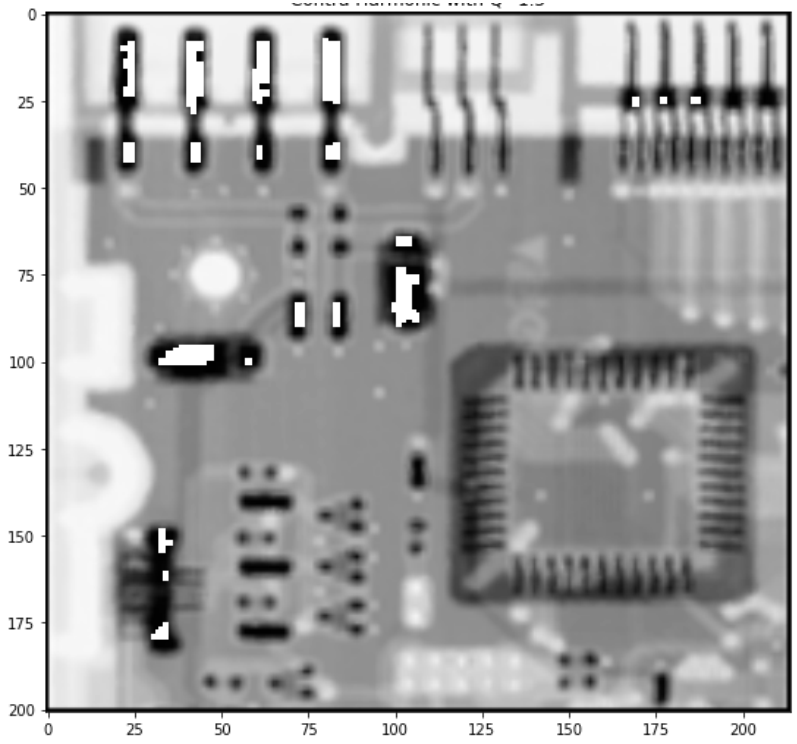
In [44]:
```
img_2=pepper_noise(img)
a=contra_harmonic(Q=0.1,img=img_2)
fig, axes = plt.subplots(2, 2,figsize=(20,20))
ax = axes.ravel()

ax[0].imshow(img,cmap=plt.cm.gray,interpolation='bilinear')
ax[0].set_title("Original")
ax[1].imshow(noise_image, cmap=plt.cm.gray,interpolation='bilinear')
ax[1].set_title("Pepper Noise Image")
ax[2].imshow(a, cmap=plt.cm.gray,interpolation='bilinear')
ax[2].set_title("Contra Harmonic with Q=1.5")
plt.show()
```

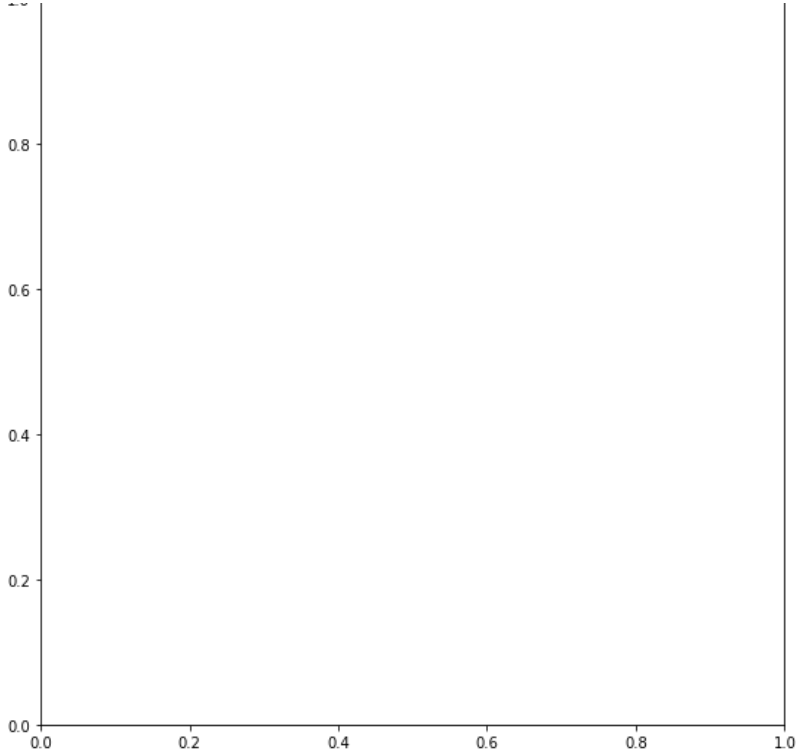
```
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:14: RuntimeWarning: invalid value encountered
in double_scalars
```





Contra Harmonic with Q=1.5


0
25
50
75
100
125
150
175
200
0
25
50
75
100
125
150
175
200


0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

In [ ]: