Exercise 3:

1.

Gaussian case:

fig 1

fig2

fig 3

fig 4

In this case, we set mean=0, sigma=0 for fig 1, mean=5, sigma=20 for fig 2, mean=10, sigma=50 for fig 3, we set mean=20, sigma=100 for fig 4.

Salt-and-pepper case:

fig 5

fig 6

fig 7

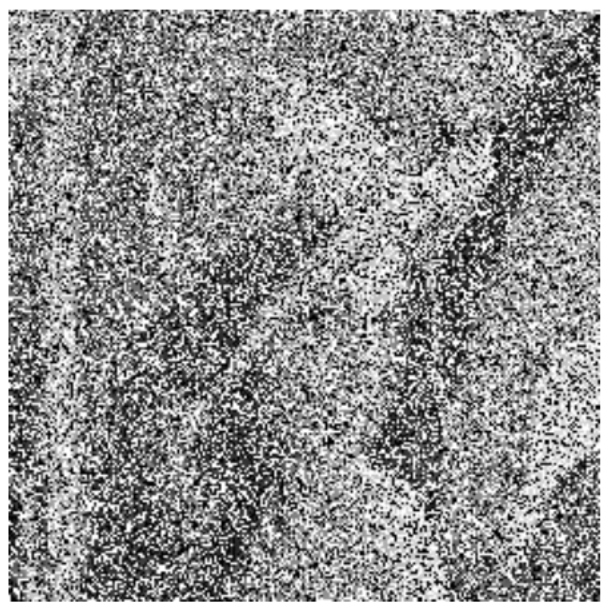

fig 8

In this case, we set pa=0.01,pb=0.01 for fig 5, pa=0.03,pb=0.03 for the fig 6, pa=0.05,pb=0.05 for the fig 7 and pa=0.4,pb=0.4 for the fig 8.

2.

kernel: 3*3:

fig 9

fig 10

fig 11

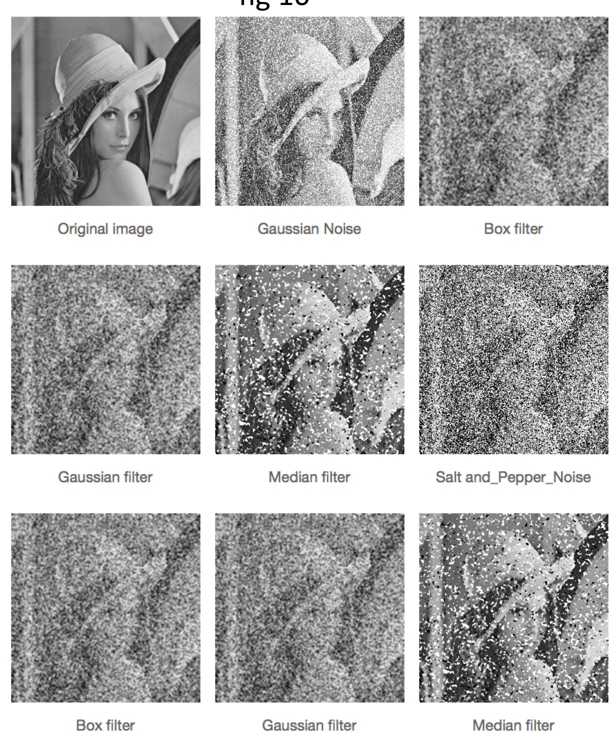

fig 12

For this part, we set the size of kernel to 3*3 for all figure from 9-12, for fig9: mean=0, sigma=0, pa=0.01, pb=0.01; for fig 10, mean=5, sigma=20, pa=0.03, pb=0.03; for fig 11, mean=10, sigma=50, pa=0.05, pb=0.05; and for fig 12, mean=20, sigma=100, pa=0.4, pb=0.4. By comparing the result, we can see that median filter works best for the noise values, no matter the kernel size is 3*3, 5*5 or 7*7. We can also know that when we get bigger kernel size, the filtered pictures are getting more blurring because there are less pixels in the picture.

Kernel size 5*5:

fig 13

fig 14

fig 15

fig 16

For this part, we set the size of kernel to 5*5 for all figure from 13-16, for fig13: mean=0, sigma=0, pa=0.01, pb=0.01; for fig 14, mean=5, sigma=20, pa=0.03, pb=0.03; for fig 15, mean=10, sigma=50, pa=0.05, pb=0.05; and for fig 16, mean=20, sigma=100, pa=0.4, pb=0.4. By comparing the result, we can see that median filter works best for the noise values.

Kernel size 7*7:

fig 17

fig 18

fig 19

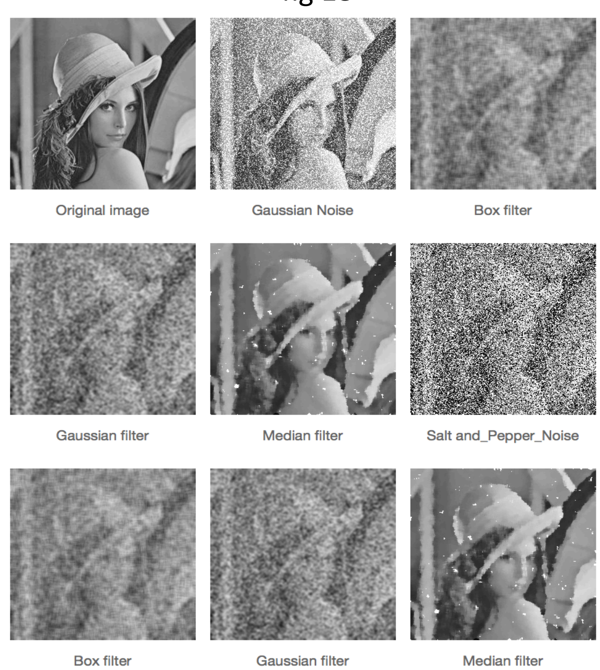

fig 20

For this part, we set the size of kernel to 7*7 for all figure from 17-20, for fig17: mean=0, sigma=0, pa=0.01, pb=0.01; for fig 18, mean=5, sigma=20, pa=0.03, pb=0.03; for fig 19, mean=10, sigma=50, pa=0.05, pb=0.05; and for fig 20, mean=20, sigma=100, pa=0.4, pb=0.4. By comparing the result, we can see that median filter works best for the noise values.