

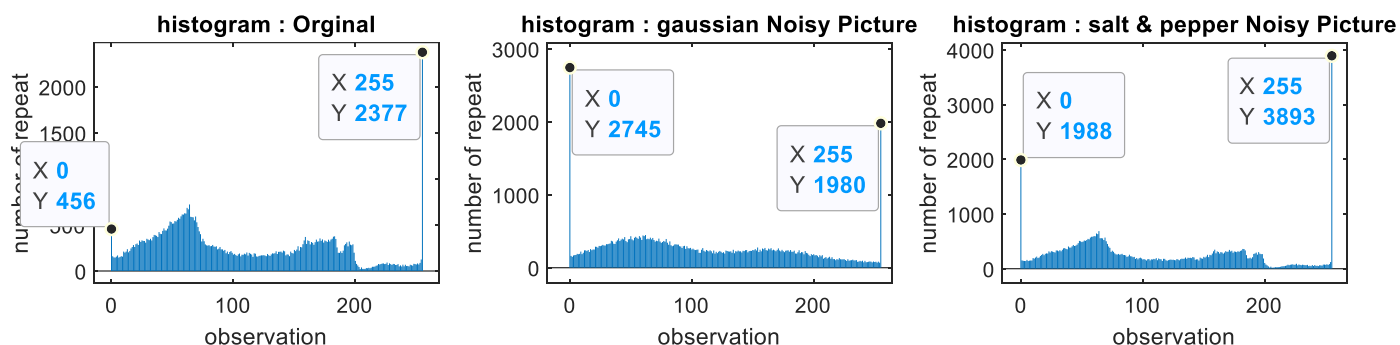
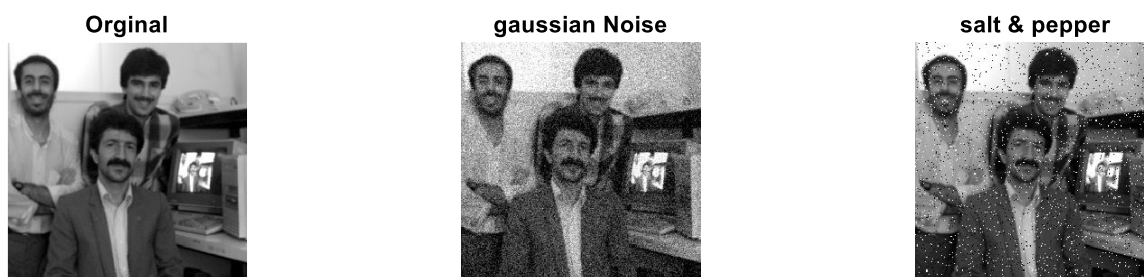
هدف

در این پروژه دو نوع نویز به یک تصویر اضافه می‌شود. نویز گوسی و نویز فلغل‌نمکی. سپس برای حذف نویز از دو نوع فیلتر متوسط گیر و میانه با سه سایز مختلف 3×3 و 5×5 و 7×7 استفاده خواهیم نمود و نتایج خروجی هر فیلتر برای هر نوع نویز را مشاهده خواهیم نمود. همچنین با رسم هیستوگرام هر تصویر اثری که هر نوع فیلتر با افزایش سایز خود بر هر تصویر می‌گذارد را مشاهده خواهیم نمود.

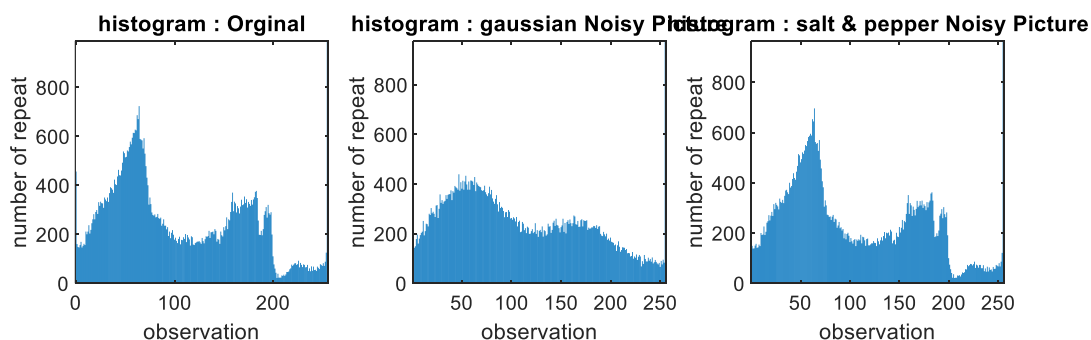
نتایج

در ابتدا تصویر اصلی را قبل و بعد از اضافه کردن نویز مشاهده خواهیم نمود.

picture's



با توجه به هیستوگرام هر عکس مشاهده میکنیم با افزودن نویز گوسی پیکسل‌های سیاه رنگ تعدادشان زیادت‌ر و پیکسل‌های تماماً سفید تعدادشان کمتر می‌شود همچنین با محدود کردن اندازه هیستوگرام مشاهده می‌شود



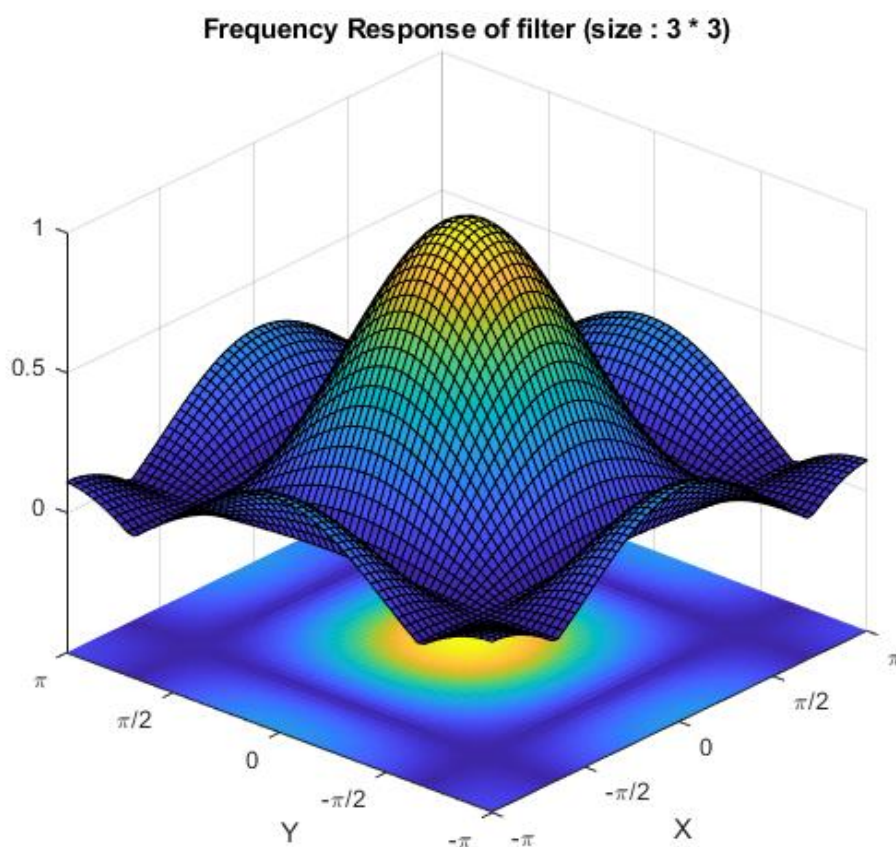
مشاهده می شود نویز گوسی به نوعی دارد تصویر را نرم میکند و تمام رنگ ها را پخش می کند و هیستوگرام یک شکل نسبتاً نرم گرفته ولی نویز فلفل نمکی به صورت رندوم بعضی پیکسل ها را سیاه کامل یا سفید کامل کرد است و شکل کلی هیستوگرام را کاملاً به هم نزده است، از همین رو انتظار داریم نویز فلفل نمکی با یک فیلترینگ مناسب بهتر از نویز گوسی قابل حذف باشد.

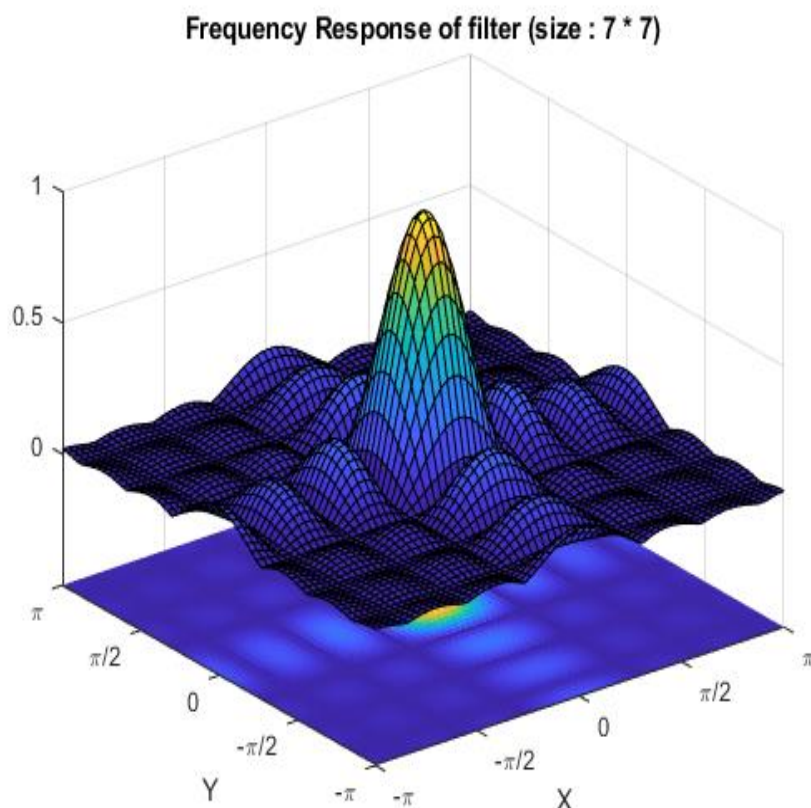
مشخصات فیلتر های میانگین گیر

برای این پروژه از سه فیلتر میانگین گیر استفاده شده است

فیلتر ۱

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$





فیلتر های میانه

فیلتر های میانه همگی یکسان و مانند فیلتر های میانگین هستند البته اساس کار این فیلتر ها مانند فیلتر های میانگین نیست و نمایش خاصی در حوزه فرکانس ندارند.

برای پیاده سازی فیلتر های میانه از دستور زیر استفاده شده است که به همراه یک مثال از عملکرد آن آورده شده است:

Median filter	B = ordfilt2(A,5,ones(3,3))	<table><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	1	1	1	<table><tr><td>88</td><td>16</td><td>56</td></tr><tr><td>5</td><td>3</td><td>30</td></tr><tr><td>21</td><td>63</td><td>42</td></tr></table>	88	16	56	5	3	30	21	63	42
1	1	1																			
1	1	1																			
1	1	1																			
88	16	56																			
5	3	30																			
21	63	42																			

دقت شود عدد ۵ نشان دهنده میانه است. یعنی در مربع سمت راست عدد 30 بعد از مرتب کردن از کوچک به بزرگ در خانه ۵ام است و در ۹ عدد مرتب شده خانه ۵ میانه است اگر سائز فیلتر ۵*۵ باشد خانه میانه بعد از مرتب کردن ۱۳ خواهد بود و فیلتر ۷*۷ خانه میانه ۲۵ خواهد بود. این را اینگونه مینویسیم.

(ordfilt2(pic,round(n^2/2),filter))

نتایج مربوط به تصویر با نویز گوسی

Gaussian Noise , Average Filter

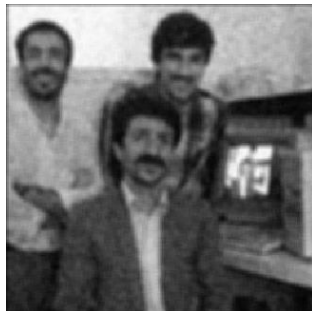
Noisy Picture



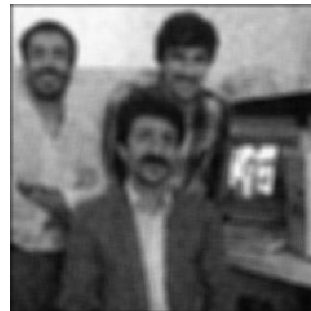
filter Size : 3 * 3



filter Size : 5 * 5

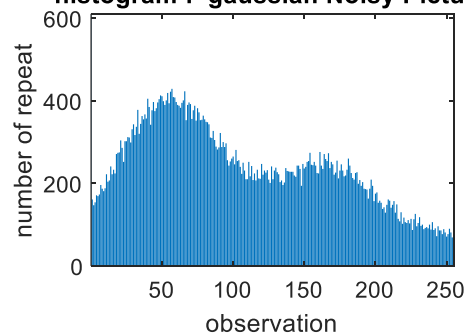


filter Size : 7 * 7

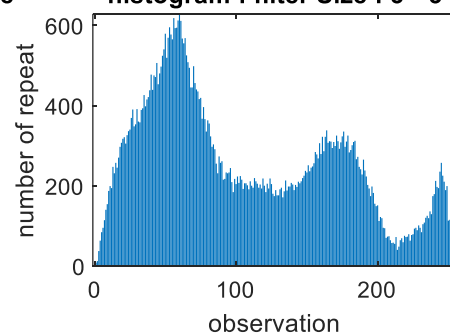


Gaussian Noise , Average Filter , Histogram

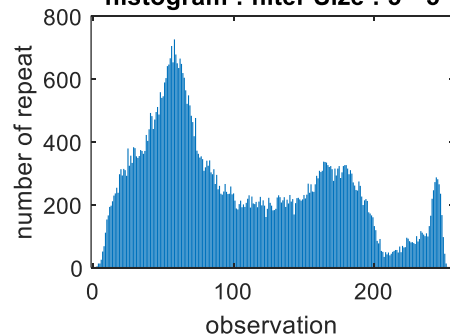
histogram : gaussian Noisy Picture



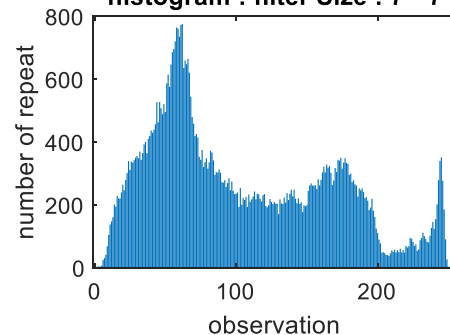
histogram : filter Size : 3 * 3



histogram : filter Size : 5 * 5



histogram : filter Size : 7 * 7



در ابتدا دقت شود هیستوگرام تصویر نویزی با محدود کردن بین صفر تا ۶۰۰ رسم شده است و نمونه‌ی بدون محدودیت آن در نمودار اول گزارش مشاهده شده است. ولی هیستوگرام سایر تصاویر بدون محدود کردن است.

اولاً مشاهده می‌کنیم با افزایش سایز فیلتر هیستوگرام از نقاط 0 و 255 فاصله بیشتری می‌گیرد که قابل درک است چرا.

ثانیاً با افزایش سایز فیلتر و با توجه به این که فیلتر متوسط گیر است هیستوگرام به سمت یکسری اعداد میل می‌کند. در واقع این اعداد میانگین نوری‌های محلی هستند در نمودارها مشخص است این اعداد نزدیک عدد ۵۰ و ۱۷۰ می‌باشد. اگر سایز فیلتر بزرگ و بزرگتر شود هیستوگرام در نهایت به سمت میانگین کل هیستوگرام اصلی میل میکند مثلاً در اینجا انتظار داریم به سمت ۷۰ که تقریباً میانگین کل هیستوگرام است میل کند.

در مورد تصاویر نیز کیفیت تصاویر با فیلتر ۳*۳ بهتر است و هنوز آنقدر مات نشده است. ولی خروجی فیلترهای ۵ و ۷ میزان محو شدگی بالا رفته و تصویر مناسب نیست. حتی در تصویر خروجی فیلتر ۷*۷ لبخندها از بین رفته است. البته با توجه به پاسخ فرکانسی فیلترها نیز قابل درک است چرا این اتفاق افتاده در واقع داریم تغییرات نرم و فرکانس پایین را نگه میداریم و بقیه را حذف میکنیم.

Gaussian Noise , Median Filter

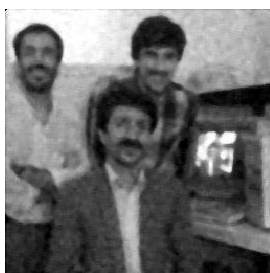
Noisy Picture



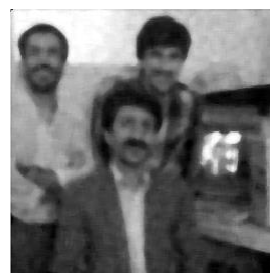
filter Size : 3 * 3



filter Size : 5 * 5

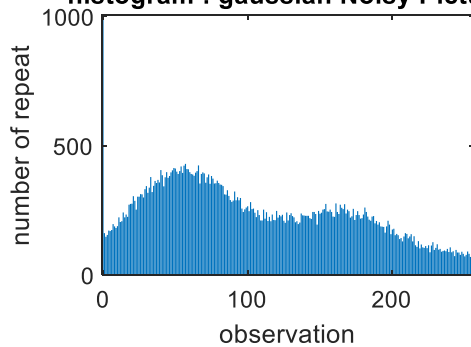


filter Size : 7 * 7

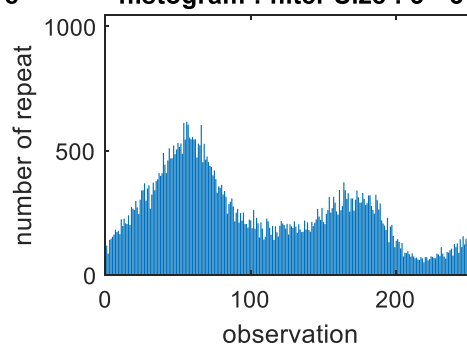


Gaussian Noise , Median Filter , Histogram

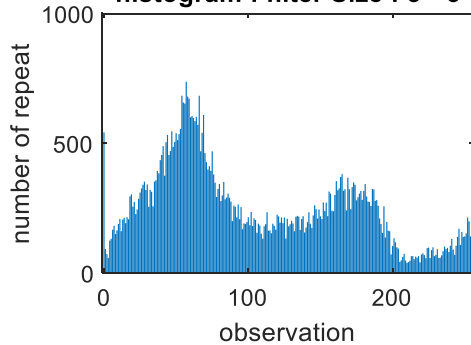
histogram : gaussian Noisy Picture



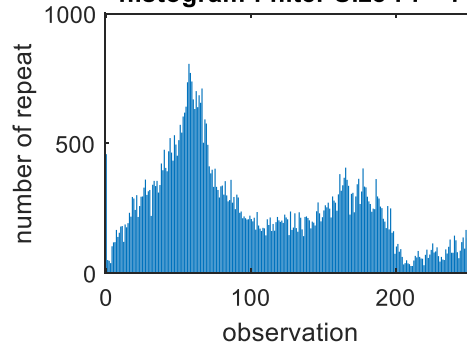
histogram : filter Size : 3 * 3



histogram : filter Size : 5 * 5



histogram : filter Size : 7 * 7



اینجا هم هیستوگرام تصویر اول بزرگنمایی شده است.

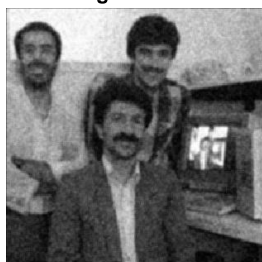
در اینجا مشاهده میکنیم با افزایش سایز فیلتر، نقاط ۲۵۵ و ۰ کاهش می‌یابند ولی به طور کامل حذف نمی‌شوند.

همچنین در اینجا نیز با افزایش سایز فیلتر اعداد به یکسری اعداد مشخص نزدیک میشوند که در اینجا نیز تقریباً همان میانگین است در اینجا نیز فیلتر با سایز 3×3 تصویر بهتری در اختیار ما میگذارد.

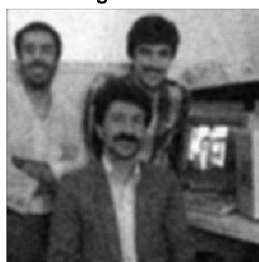
در نهایت ما تصویر نویز گوسی را از ۶ فیلتر عبور دادیم که برای مقایسه تصویر همه را با هم در پایین مشاهده میکنیم. (برای نتیجه گیری بهتر)

Gaussian Noise

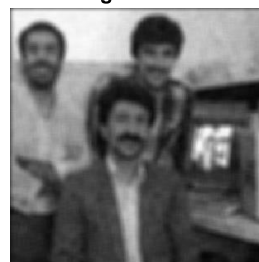
Average filter: 3×3



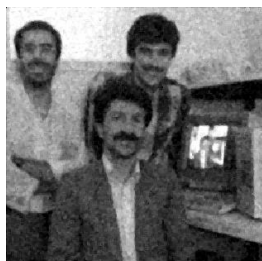
Average filter: 5×5



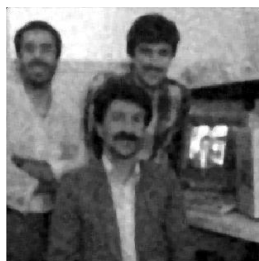
Average filter: 7×7



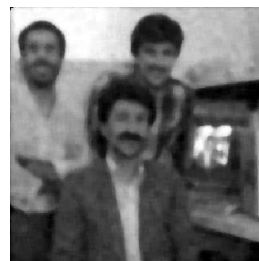
Median filter: 3×3



Median filter: 5×5



Median filter: 7×7



ترجیح من فیلتر متوسط گیر 3×3 است.

بررسی نتایج تصویر با نویز فلفل نمکی

Salt & Pepper Noise , Average Filter

Noisy Picture



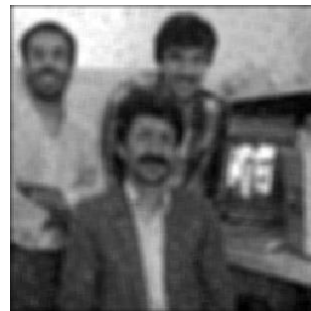
filter Size : 3 * 3



filter Size : 5 * 5

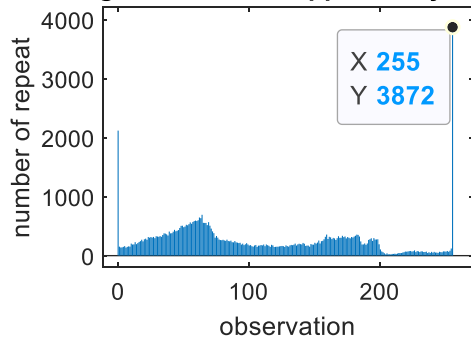


filter Size : 7 * 7

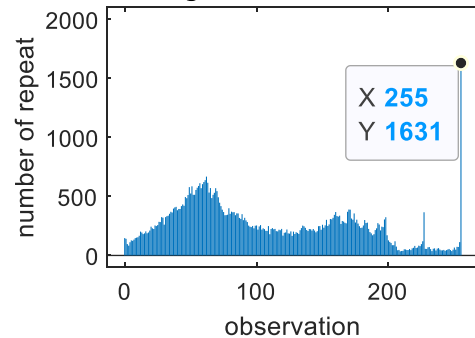


Salt & Pepper Noise , Average Filter , Histogram

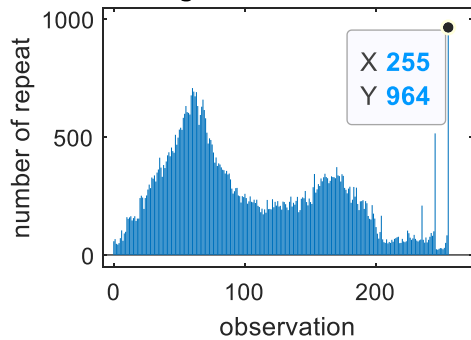
histogram : Salt & Pepper Noisy Picture



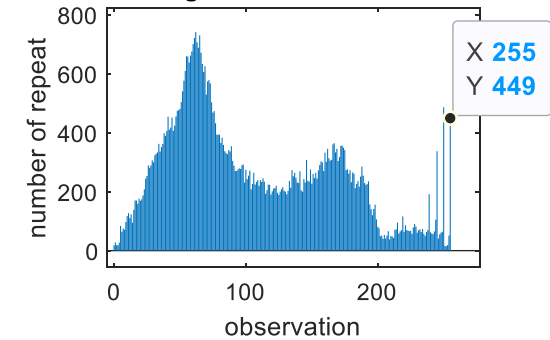
histogram : filter Size : 3 * 3



histogram : filter Size : 5 * 5



histogram : filter Size : 7 * 7



اولاً علت وجود همواره تعداد زیادی ۲۵۵ در هیستوگرام پنجره بالا سمت چپ در تصویر است که تماماً سفید است. و البته سایر نقاط تمام سفید این نقاط در حضور نویز گوسی برخی پیکسل هایشان از حالت تمام سفید فاصله می‌گرفتند و به همین دلیل با یک فیلتر متوسط از نقطه ۲۵۵ فاصله می‌گرفتند. در حضور نویز فلفل نمکی این تفاق نیفتاده است چون همچنان این نقاط تمام سفید مقدار زیادی هستند و میتوانند در کل فیلتر جا بگیرند. سایر نقاط در مورد هیستوگرام مثل قبل است.

در مجموع هیچ کدام از تصویر ها مناسب نیست ولی باز هم فیلتر 3×3 بهتر عمل کرده است.

Salt & Pepper Noise , Median Filter

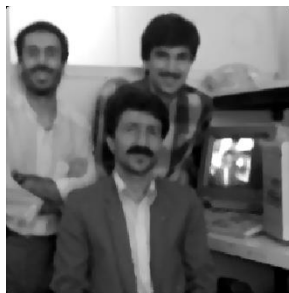
Noisy Picture



filter Size : 3 * 3



filter Size : 5 * 5

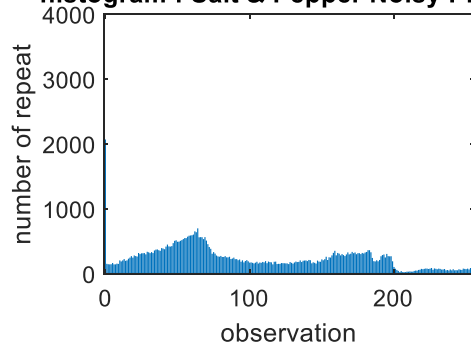


filter Size : 7 * 7

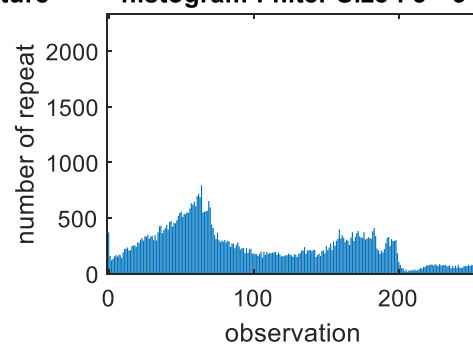


Salt & Pepper Noise , Median Filter , Histogram

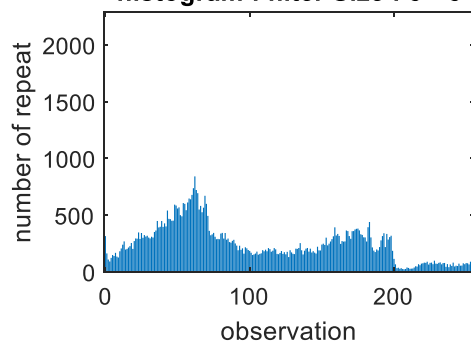
histogram : Salt & Pepper Noisy Picture



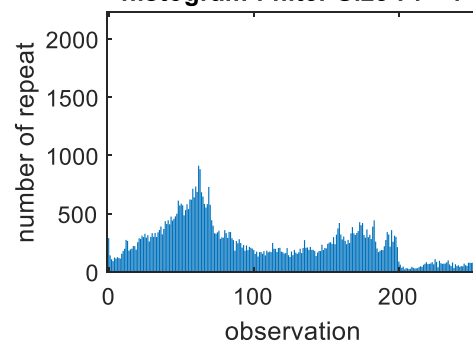
histogram : filter Size : 3 * 3



histogram : filter Size : 5 * 5



histogram : filter Size : 7 * 7



نکته جدیدی در مورد هیستوگرام ها موجود نیست و هر آنچه لازم باشد در قسمت های قبل گفتیم و درست است فقط در این جا نیز تعداد صفر ها به شدت کاهش می یابد و تعداد ۲۵۵ ها به نسبت رشد فیلتر کمتر می شود که دلیل این اتفاق هم نوع دیتا می باشد.

در مورد تصاویر نیز، باید گفت تصاویر شطرنجی شده اند و با افزایش سایز فیلتر محو شدگی نیز شدت میگیرد. کاملاً واضح است فیلتر 3×3 باز هم بهتر عمل کرده است.

در پایین تصاویر خروجی هر 3×3 فیلتر و هر نوع را مشاهده و مقایسه می کنیم.

Salt & Pepper Noise

Average filter: 3×3



Average filter: 5×5



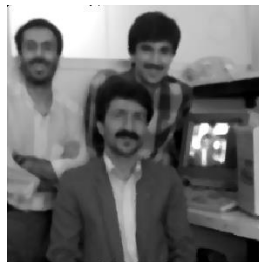
Average filter: 7×7



Median filter: 3×3



Median filter: 5×5



Median filter: 7×7



مشاهده میکنیم برای نویز فلفل نمکی فیلتر میانه بهتر عمل میکند و نقاط سیاه یا سفید شده را به طور کامل برمیدارد ولی فیلتر میانگین نقاط سیاه یا سفید شده را پخش میکند.

کد برنامه

سه فایل متلب وجود دارد یکی همان `pichist.m` برای رسم هیستوگرام است که در پروژه یک درست کردیم. و کد آن را آوردیم.

یک فایل `project2_Gaussian.m` که برنامه را برای تصویر با نویز گوسی اجرا میکند.

یک فایل هم project2_saltandpepper.m که برنامه را برای تصویر با نویز فلفل نمکی ران میکند.

در واقع برنامه فلفل نمکی همان برنامه گوسی است فقط برای وقت گیر نشدن برنامه نویسی کلمه gauss و Gaussian را با salt و Salt & pepper "Replace" کرده ام و هیچ فرق دیگری ندارد به همین جهت در پایین فقط کد گوسی را میگذارم.

```
clc
clear all
close all
%%
pic = imread("YAD1-256.bmp");
m = 0; %mean
varGauss = .01; %Variance
picGauss = imnoise(pic, 'gaussian', m, varGauss);
d = 0.05; %density (%)
picSalt = imnoise(pic, 'salt & pepper', d);

figure
sgtitle("picture's")

subplot(2,3,1)
imshow(pic)
title("Orginal")
subplot(2,3,2)
imshow(picGauss)
title("gaussian Noise")
subplot(2,3,3)
imshow(picSalt)
title("salt & pepper")

subplot(2,3,4)
hist1 = pichist(pic);
title("histogram : Orginal")
ylabel("number of repeat")
xlabel("observation")

subplot(2,3,5)
hist1 = pichist(picGauss);
title("histogram : gaussian Noisy Picture")
ylabel("number of repeat")
xlabel("observation")

subplot(2,3,6)
hist1 = pichist(picSalt);
title("histogram : salt & pepper Noisy Picture")
ylabel("number of repeat")
xlabel("observation")
%%
i = 0;
figure(5)
sgtitle("Gaussian Noise , Average Filter ")
subplot(2,2,1)
imshow(picGauss)
title("Noisy Picture")

figure(6)
sgtitle("Gaussian Noise , Average Filter , Histogram ")
subplot(2,2,1)
hist1 = pichist(picGauss);
```

```

title("histogram : gaussian Noisy Picture")
ylabel("number of repeat")
xlabel("observation")

figure(9)
sgtitle("Gaussian Noise")

for n = [3 5 7] %filter size
    i = i+1;
    avefilter = (1/(n*n))*ones(n,n);
    nfft = 64;
    Fmedfilter1 =fftshift(fft2(avefilter,nfft,nfft));

    freq = linspace(-pi,pi,nfft);
    offset = 0.5;
    figure
    s = surf(freq,freq,offset + abs(Fmedfilter1));
    hold on
    imagesc(freq,freq,offset + abs(Fmedfilter1));
    xticks([-pi -pi/2 0 pi/2 pi])
    xticklabels({'-\pi', '-\pi/2', '0', '\pi/2', '\pi'})
    yticks([-pi -pi/2 0 pi/2 pi])
    yticklabels({'-\pi', '-\pi/2', '0', '\pi/2', '\pi'})
    zticks([offset,offset+0.5,offset+1])
    zticklabels({'0', '0.5', '1'})
    % s.EdgeColor = 'none';
    title("Frequency Response of filter (size : "+ num2str(n) + " * "+num2str(n)+")")
    xlabel("X")
    ylabel("Y")
    xlim([-pi pi])
    ylim([-pi pi])

    picAveFiltered = uint8(conv2(picGauss,avefilter,'same'));

    figure(5)
    subplot(2,2,i+1)
    imshow(picAveFiltered)
    title("filter Size : "+ num2str(n) + " * "+num2str(n))

    figure(6)
    subplot(2,2,i+1)
    hist1 = pichist(picAveFiltered);
    title("histogram : filter Size : "+ num2str(n) + " * "+num2str(n))
    ylabel("number of repeat")
    xlabel("observation")
    figure(9)
    subplot(2,3,i)
    imshow(picAveFiltered)
    title("Average filter: "+ num2str(n) + " * "+num2str(n))
end
figure(7)
sgtitle("Gaussian Noise , Median Filter ")
subplot(2,2,1)
imshow(picGauss)
title("Noisy Picture")
figure(8)
sgtitle("Gaussian Noise , Median Filter , Histogarm ")
subplot(2,2,1)
hist1 = pichist(picGauss);
    
```



```
title("histogram : gaussian Noisy Picture")
ylabel("number of repeat")
xlabel("observation")

i = 0;
for n = [3 5 7]
    i = i+1;
    medfilter = ones(n);
    picMedFiltered = uint8(ordfilt2(picGauss,round(n^2/2),medfilter));
    figure(7)
    subplot(2,2,i+1)
    imshow(picMedFiltered)
    title("filter Size : "+ num2str(n) + " * "+num2str(n))
    figure(8)
    subplot(2,2,i+1)
    hist1 = pichist(picMedFiltered);
    title("histogram : filter Size : "+ num2str(n) + " * "+num2str(n))
    ylabel("number of repeat")
    xlabel("observation")

    figure(9)
    subplot(2,3,i+3)
    imshow(picMedFiltered)
    title("Median filter: "+ num2str(n) + " * "+num2str(n))
end
```