Iraqi Journal of Physics, 2008 *Vol. 5, No. 1, PP. 90-103*

Smoothing of Image using adaptive Lowpass Spatial Filtering

Suhad A. Hamdan
Department of Physics, College of Science, University of Baghdad

Abstract

Lowpass spatial filters are adopted to match the noise statistics of the degradation seeking good quality smoothed images. This study imply different size and shape of smoothing windows. The study shows that using a window square frame shape gives good quality smoothing and at the same time preserving a certain level of high frequency components in comparsion with standard smoothing filters.

تنعيم الصورة باستخدام مرشح الترددات المكانية الواطئة المحور

سهاد عبد الكريم حمدان

الخلاصة:

يتضمن هذا البحث استخدام طريقة من طرق تحسين الصور المشوشة وهي Lowpass Spatial Filtering مستخدمين نوافذ باشكال واحجام مختلفة للبحث عن صورة محسنة بجودة عالية. وتمت دراسة تاثير شكل وحجم النافذة على تحسن الصورة المشوشة .هذه الدراسة بينت ان النافذة ذات الشكل Square Frame أعطت نتائج جيدة وحافظت على الترددات العالية في الصورة في نفس الوقت مقارنة بطرق التحسين القياسية.

Introduction

Noise in an image generally has a higher spatial frequency spectrum than the normal image components. Image noise arising from a noisy sensor or channel transmission errors usually appears as discrete isolated pixel variations [pratt: 78].

Smoothing (by local weighted averaging) is an effective image regularization method that has been used for denoising, restoration, and enhancement. A drawback is that smoothing can damage image features such as edges, lines, and textures [Ros:82,Carsten:2000]. To avoid the damage, the smoothing has to be adaptively controlled with two principls: 1) control of the amount of smoothing, i.e., less smoothing in the locations with strong image features, and more smoothing in the locations with weak image features; and 2) control of the direction of smoothing , i.e., minimal smoothing in the directions across the image features , and maximal smoothing in the directions along the image features [Rene:98].

In general, image can be filtered (enhanced) either in the frequency or in the spatial domain.

A-<u>Spatial domain filters</u>

Those types of filters appled directly on the pixels composing an image. Different kinds of spatial filters have been designed and implemented e.g. lowpass filter , median, ………., etc.

B-<u>Frequency domain filters</u>

Refers to those types of filters applied on the transformed coefficients (frequencies) instead of the real pixels values, taking into consideration that the transformed coefficients give an indication to the variation in the value of the pixels as a function of the pixel coordinates, e.g. low-pass or high – pass filters can be employed to smooth the picture, or to enhance the sharp edges. The design of these filters has been well developed in digital signal processing [Oppenheim 1975, Huang 1975].

Enhancement technique based on various combination of methods from these two categories [Gonzalaz:87, Gonzalaz:92, Gonzalaz:2000].

In this paper we study smoothing image in spatial domain using lowpass spatial filtering.

Digital Convolution and Filtering

This section focused on a brief discrbtion of the mathematical convolution and filtering.

Digital convolution

Linear systems theory is a branch of mathematics that provides the mathematical basis for some digital filters. This means that if the filter satisfies certain conditions (i.e, it is linear and shift-invariant), then the output of such filters can be expressed mathematically in terms of a convolution equation [Awcock 1995].

For one-dimensional digital signal the equation is,

$$g(i) = \sum_{k=-\infty}^{+\infty} f(k)h(i-k)\ldots\ldots\ldots\ldots\ldots(1)$$

where $g(i)$ is the ouput image at point (i) and h is the point spread function or impulse response which identifies the filter of the system. The (h) functin is usually equal to zero outside some range, the above equation may be rewritten as:

$$g(i) = \sum_{k=i-w}^{i+w} f(k)h(i-k)\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

where $(-w,+w)$ is the range over which h is nonzero.

Thus the output g(i) at point (i) is given by a weighted sum of input pixels surrounding (i) where the weights are given by h(k). To create the output at the next pixel (i+1) the function h(k) is shifted by one, and the weighted sum is recomputed. The full output is created by a series of shift-multiply-sum operations, and this is called a digital covoluation.

In two-dimensions, h(k) becomes h(k, l) and eq.(2) becomes a double summation:

$$g(i,j) = \sum_{k=i-w}^{i+w} \sum_{l=j-v}^{j+v} f(k,l)h(i-k, j-l)\ldots\ldots(3)$$

Here again , g(i, j) is created by a series of shift – multiply – sum operations. The values of h are also referred to as the filter weights, the filter kernel, or the filter mask (window) which is moved over the complete digital image f(i, j) . For reasons of symmetry, h(i, j) is almost always chosen to be of size m×n where both m and n are odd. Often m=n. There is a reflection between the filter weights and the point spread function of a filter (h), if h(k, l) is the point spread function, then the weights or filter mask is usually given as h(-k, -l).

In physical systems, the kernel h must be always non-negative, which results in some blurring or averaging of the image. The basic idea of convolution, the weights of (h) may be varied over the image, and the size and shape of the window varied. These operations are no longer linear, and are no longer convolutions, but become general “moving window Operations” which are very common in digital image processing. With this flexibility, a wide range of linear, nonlinear, and adaptive filters may be implemented such as for edge enhancement or selective smoothing. [Niblack1986].

Spatial Filtering

The use of spatial masks for image processing usually is called spatial filtering , and the masks themselves are called spatial filters [Gonzalaz:92]. Spatial filtering is performed by convolving the image with a mask or a kernel [pratt:78].

The image smoothing filtering categories:

A- Linear Filters

The type of linear filter used, the basic approach is to sum products between the mask coefficients and the intensities of the pixels under the mask at a specific location in the image. Fig.(1) shows a general 3×3 mask. Denoting the gray levels of pixels under the mask at any location by $p1, p2, p3, ..., p9$, the response of a linear mask is [Gonzalaz:92, Gonzalaz:2000].

$$R = w1p1 + w2p2 + ...w9p9.....(4)$$

It is called linear filters because the filter's output values are related with all the value of adjacent pixels (covered by the filter area) by a linear relationship e.g. lowpass filter [Pratt:75, Justusso:78].

P1	P2	P3
P4	P5	P6
P7	P8	P9

Fig. (1): a 3×3 Mask with arbitrary coefficients (weights)

B- Nonlinear Filter

It refers to all other kinds of filters doesn't obey a linear relationship between the filter's output values and the adjacent pixels values.[pratt:1975, Nonlinear spatial filters operate on neighborhoods. Their operation is based on directly on the values of the pixels in the neighborhood under consideration e.g. the median filter in which a pixel value is replaced by the median of a set of it's neighbors [Gonzalaz:92, Gonzalaz:2000].

Numerical implementations and Results

The shape of the impluse respose needed to implement a lowpass (smoothing) spatial filter indicates that the filter has to have all positive coefficients (see Fig. (2))

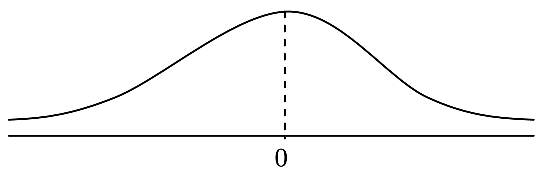


Fig. (2): Cross section of corresponding spatial domain filters

Although the spatial filter shape shown in Fig. (2) could be modeled by, say, a sampled Gaussian function, the key requirement is that all the coefficients be positive. For a 3×3 spatial filter, the simplest arrangement would be a mask in which all coefficients have a value 1. From eq.(4), the response would then be the sum of gray levels for nine pixels, which could cause R to be out of the valid gray-level range. The solution is to scale the sum by dividing R by 9. Fig. (3a) shows the resulting mask. Larger mask follow the same concept, as Fig. (3b). Note that, in all these cases, the response R would simply be the average of all the pixels in the area of the mask. For this reason, the use of masks of the form shown in Fig. (3) is often referred to as neighborhood averaging [Gonzalaz:92, Gonzalaz: 2000]. If the required noise mask becomes large in order to achieve adequate noise cleaning, it is usually more computationally efficient to perform the convolution operation [pratt:78].

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

(a)

$$1/25\times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

(b)

Fig. (3) spatial lowpass Filter of various sizes

We have studied the effects of the shape and size of the used window, many filters can be used, we take the weight and multiply it by mask; then sum the corresponding values. Often a 3×3 a square kernel is used as shown in Fig. (4), although larger kernels (e.g. 5×5 squares) can be used for more severe smoothing as shown in Fig. (5). Various standard kernels exist for specific applications, where the size and the form of the kernel determine the characteristics of the operation. The idea of average filtering is simply to replace each pixel value in an image with the average value of it's neighbors, including itself. This has the effect of eliminating pixel value which are unrepresentative of their surroundings. Average filtering is usually though of as a convolution filter. Like other convolutions it is based around a kernel, which represents the shape and size of the neighborhood to be sampled when calculating the average. The weighted average on a pixel Neighborhood, in the averaging of equation (4) the pixels were all weighted the same by the value $W_k = 1/9$ and all 9 of these weights added up to unity. The sum of all mask entries multiplied by the mask factor should equal unity to keep the output average brightness the same as in the original image. Here the weights 1/9 are all equal and we say that the average has equal weighting. Consider the following weighting

$$P_k = \sum_{k=1,9} W_k P_k \ldots\ldots\ldots\ldots\ldots(5)$$

It is convenient to put the weights in an array called a mask of the same size as the neighborhood, and to define an operation for the weighted average. Consider the operation of a mask and neighborhood shown below:

$$P_{new} = \begin{vmatrix} W1 & W2 & W3 \\ W4 & W5 & W6 \\ W7 & W8 & W9 \end{vmatrix} * \begin{vmatrix} P1 & P2 & P3 \\ P4 & P5 & P6 \\ P7 & P8 & P9 \end{vmatrix} = \sum_{K=1,9} W_k P_k$$

we call this operation convolution (as explaned in previous section). An example of weighted averaging convolution mask and its mask factor (1/5) that forces the weights to sum to unity, the 3×3 plus mask operations on a 3×3 neighborhood is:

$$P_{new} = (1/5)\begin{vmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{vmatrix} * \begin{vmatrix} P1 & P2 & P3 \\ P4 & P5 & P6 \\ P7 & P8 & P9 \end{vmatrix} =$$

$$\{P2 + P4 + P5 + P6 + P8\}/5$$

In same approch we get masks in 3×3 size and Horizontal line shape, Vertical line shape,Square frame shape, Digonal shape, and plus with zero center shape as shown in Fig. (4)

as well as we obtained masks in 5×5 size as shown in figure (5).

In general, the weighted averaging filter. It can be expressed as follows [Roesnfeld:1969, Prewitt:1970, Davis: 1975, and Rosenfeld:1976]:

$$g(x,y) = f(x,y) * h(x,y) = \sum_{i=-m}^{m} \sum_{j=-n}^{n} f(i,j)h(x-i, y-j.) \ldots (6)$$

where f & g are the input and output functions respectively, and the weighting factors of the filter h are equal to : $\dfrac{1}{(2m+1)(2n+1)}$

This study represent the theartical results of lowpass spatial filtering using many window with different size and shapes. For testing, the performance the filters, the original medical image of size (256*256) pixel and it's histogram are shown in Fig. (6a). This image is corrupted by a Gaussian noise. The degraded image and it's histogram are show in Fig. (6b).

Fig.s (7,8,9,10) show the smoothed images, their histogram , and the type of smoothing filters that using in the process.

After implementing all of the previous filters on test image, we compare the performance of the filters using statistical analysis such as standard deviation (Stdv.), minimum, maximum, and mean as shown in table (1). In additions to that PSNR (Peak Signal to Noise Ratio) and MSE (Mean Square Error) are culculated to check the quality of smoothed images as shown in table (2) this table demonstrat the superiority of our presented filter.

Conclusions

Depending on the previous results, we can conclude that using a square frame window gave better quality smoothed images this is because this filter preserve the histogram and not creating artifacts in the smoothed image.

$$1/3\times \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

a- Horizontal Line Shape

$$1/3\times \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

b-Vertical Line Shape

$$1/8\times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

c- A square frame shape

$$1/5\times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

d-Pluse shape

$$1/3\times \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

e-Diagonal shape

$$1/4\times \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

f-plus shape with zero center

Fig. (4) Representation modified filters with different shapes and with 3× 3 size window

1/5×

0	0	0	0	0
0	0	0	0	0
1	1	1	1	1
0	0	0	0	0
0	0	0	0	0

a- Horizontal Line Shape

1/5×

0	0	1	0	0
0	0	1	0	0
0	0	1	0	0
0	0	1	0	0
0	0	1	0	0

b- Vertical Line Shape

1/16×

1	1	1	1	1
1	0	0	0	1
1	0	0	0	1
1	0	0	0	1
1	1	1	1	1

c- A square frame shape

1/9×

0	0	1	0	0
0	0	1	0	0
1	1	1	1	1
0	0	1	0	0
0	0	1	0	0

d- pluse shaped

1/5×

1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

e- Diagonal shape

1/21×

0	1	1	1	0
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
0	1	1	1	0

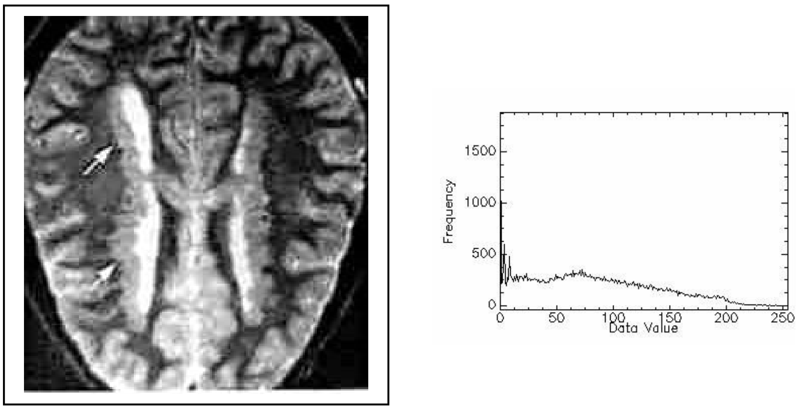
f- circular shape

1/16×

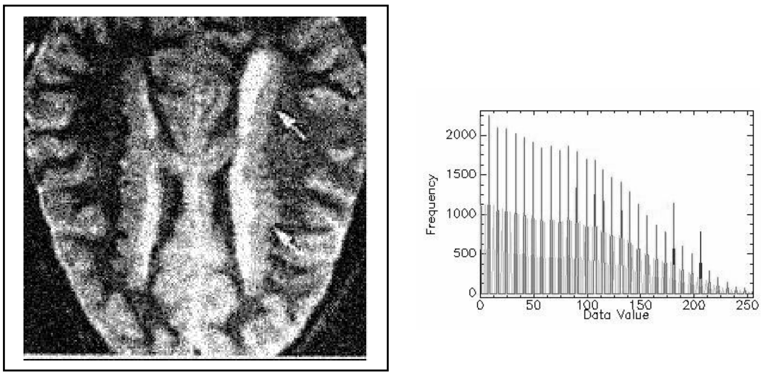
0	1	1	1	0
1	1	0	1	1
1	0	0	0	1
1	1	0	1	1
0	1	1	1	0

g- circular with zero center

Fig. (5) Representation modified filters with different shape and with 5×5 window size

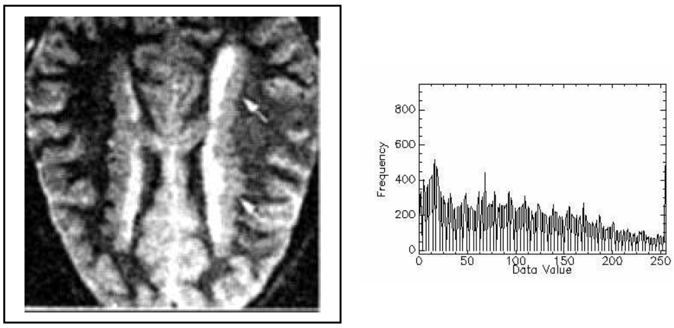


a



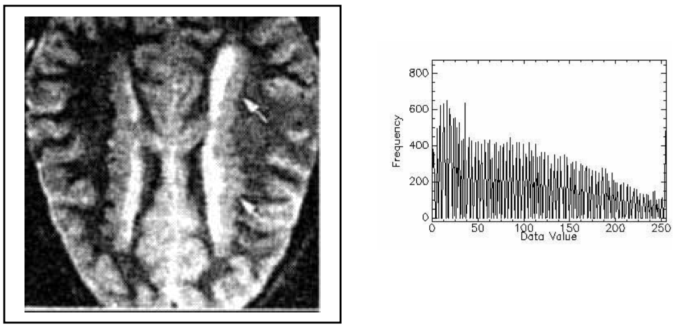
b

Fig. (6)
a. Original image ***b. noisy Image***



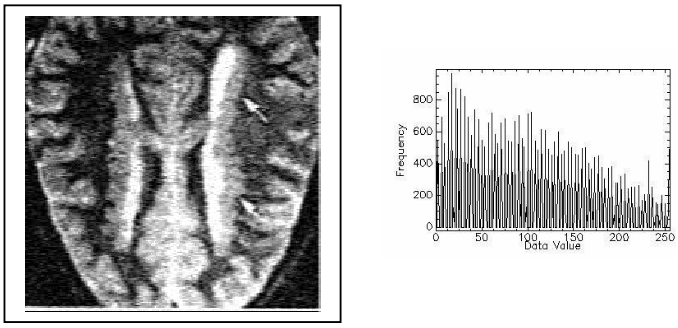
1 / 9	1/ 9	1/ 9
1 / 9	1 / 9	1 / 9
1 / 9	1 / 9	1 / 9

a



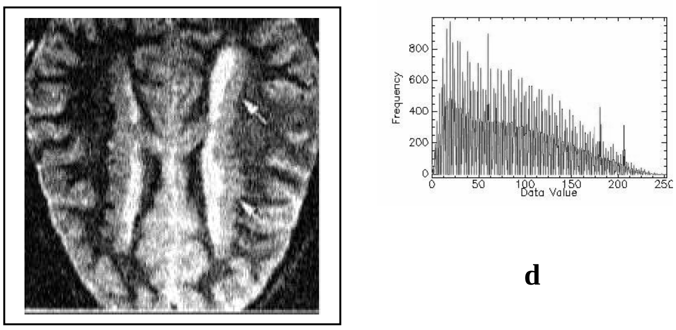
0	1/ 5	0
1/ 5	1/ 5	1/ 5
00	1/ 5	0

b



0	0	0
1/ 3	1/3	1/3
0	0	0

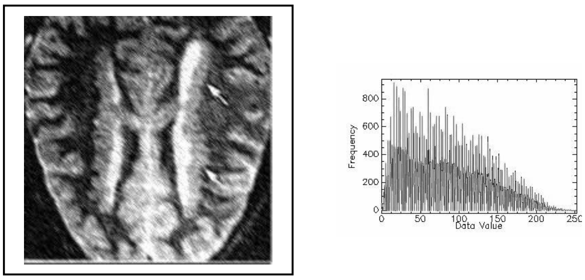
c



0	1/3	0
0	1/3	0
0	1/3	0

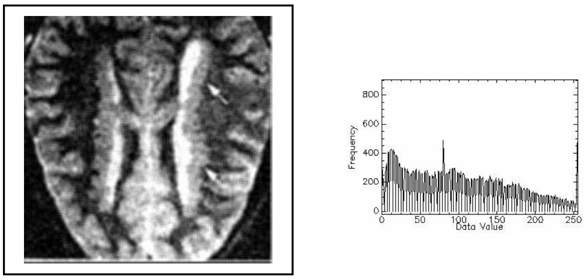
d

Fig. (7): Represent the smoothing of images using different classic and modification Block and its Histogram



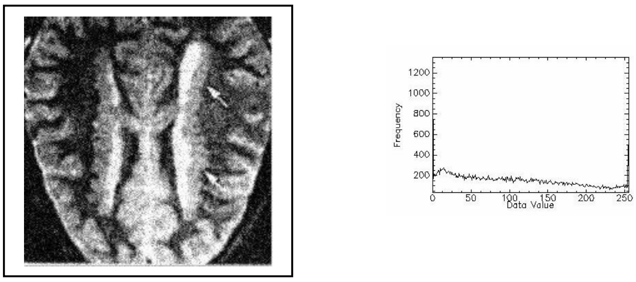
1/3	0	0
0	1/3	0
0	0	1/3

a



1/8	1/8	1/8
1/8	0	1/8
1/8	1/8	1/8

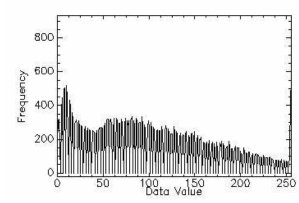
b



0	1/4	0
1/4	0	1/4
0	1/4	0

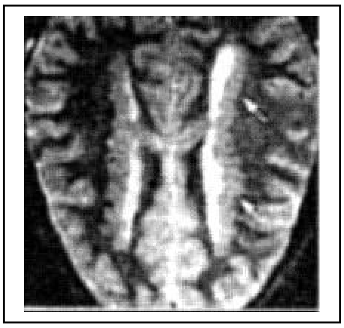
c

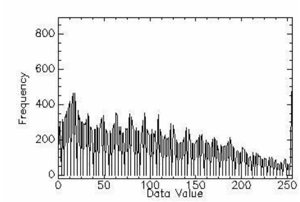
Fig. (8): Represent the smoothing of images using different classic and modification Block and it's Histogram



1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25
1/25	1/25	1/25	1/25	1/25

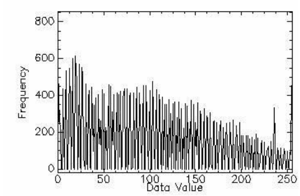
a





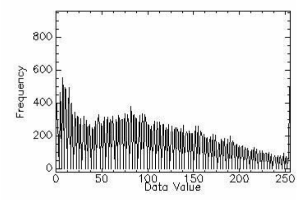
0	0	1/9	0	0
0	0	1/9	0	0
1/9	1/9	1/9	1/9	1/9
0	0	1/9	0	0
0	0	1/9	0	0

b



0	0	0	0	0
0	0	0	0	0
1/5	1/5	1/5	1/5	1/5
0	0	0	0	0
0	0	0	0	0

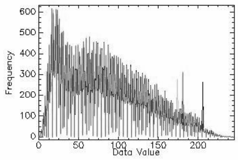
c



0	1/21	1/21	1/21	0
1/21	1/21	1/21	1/21	1/21
1/21	1/21	1/21	1/21	1/21
1/21	1/21	1/21	1/21	1/21
0	1/21	1/21	1/21	0

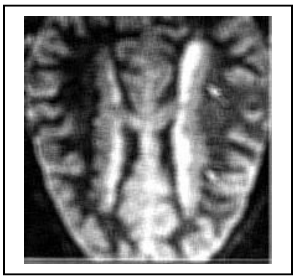
d

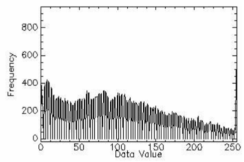
Fig. (9): Represent the smoothing of images using different classic and modification Block and its Histogram



0	0	1/5	0	0
0	0	1/5	0	0
0	0	1/5	0	0
0	0	1/5	0	0
0	0	1/5	0	0

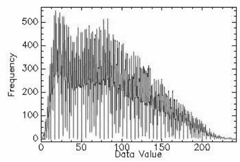
a





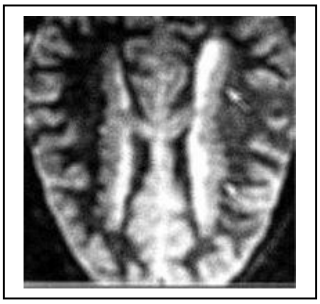
1/16	1/16	1/16	1/16	1/16
1/16	0	0	0	1/16
1/16	0	0	0	1/16
1/16	0	0	0	1/16
1/16	1/16	1/16	1/16	1/16

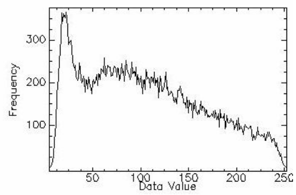
b



1/5	0	0	0	0
0	1/5	0	0	0
0	0	1/5	0	0
0	0	0	1/5	0
0	0	0	0	1/5

c





0	1/16	1/16	1/16	0
1/16	1/16	0	1/16	1/16
1/16	0	0	0	1/16
1/16	1/16	0	1/16	1/16
0	1/16	1/16	1/16	0

d

Fig. (10): Represent the smoothing of images using different classic and modification Block and its histogram

Table (1): The statistical properties for different modification filters and classical filter

	Shape ofwindow	**SizeOf window**	**min**	**Max**	**mean**	**Stdv.**
	Original image		0	254	79.595	54.359
	Noisy image		0	255	84.5974	595.45
1	Square	3	0	255	101.8004	70.952
2	Pluse	3	0	255	101.8195	70.995
3	Horizontal	3	0	255	103.1477	70.7105
4	Vertical	3	0	255	84.6007	54.2808
5	diagonal	3	0	252	84.6012	52.9219
6	Square frame	3	0	255	102.179400	69.9918
7	Plus shap with zero center	3	0	255	106.155	75.067
8	Square	5	0	255	102.451	69.812
9	Pluse	5	0	255	102.259	70.283
10	Horizontal	5	0	255	103.1779	70.2699
11	Circular	5	0	255	101.452	70.3143
12	Vertical	5	0	255	84.3856	52.2118
13	Square frame	5	0	255	103.1976	69.31002
14	diagonal	5	0	242	84.5376	49.8358
15	Circular with zero center	5	5	252	103.857	62.243

Table (2): The PSNR and MSE for different modification filters and classical filters

	Shape of window	**Size Of window**	**PSNR**	**MSE**
	Noisy image		9.3941731	7475.8924
1	Horizontal	3	9.70834	6954.1868
2	Vertical	3	9.461009	7361.72
3	Square frame	3	10.12947	6311.5098
4	Square	3	9.85723	6719.815
5	Pluse	3	9.77938	6841.3526
6	diagonal	3	10.0355	6449.4505
7	Plus shap with zero center	3	9.591725	7143.446
8	Horizontal	5	9.84423	6739.9494
9	Vertical	5	9.596219	7136.0585
10	Square frame	5	10.3237923	6035.3314
11	Square	5	10.057624	6416.791
12	Pluse	5	9.90821	6641.3946
13	diagonal	5	10.2223	6177.9286
14	Circular	5	10.02219	6469.3490
15	Circular with zero center	5	10.08684	6373.7691

Refrenceses

[1] [pratt:78], pratt, W.K., (1978), (Digital Image Processing), Wiely Inter. Science, NewYork.

[2] [Ros:82], Rosenfeld, A. & Kak, A.C.(1983), (Digital Picture Processing), 2nd Edition, Vol.1, Academic Press, Newyork..

[3] [Carsten:2000], Garnica, C., Boochs F., and Twardochlib M., (2000), (Anew Approach to Edge-Preserving smoothing for Edge Extraction and Image Segmentation), IAPRS, Vol.XXXIII, Amsterdam.

[4] [Rene:98], Rene A.Carmona, and Sifen Zhong, March 1998, (Adaptive smoothing Respecting Feature Direction), IEEE Transactions on Image Processing, Vol.7, No.3.

[5] [Oppenheim:75], Oppenhein, V. and Schaefer, R.W., 1975, (Digital Signal Processing), Printice Hall. Englewood Cliffs, N. J.

[6] [Huang: 75], Huang, T.S.(Ed.), 1975, (Picture Processing & Digital Filtering), Springer, NewYork.

[7] [Gonzalaz:87], Gonzalaz, R.C. & Wintz, P.,(1987), (Digital Image Processing), 2nd edition, Addison - Wesley publishing Company, Inc.

[8] [Gonzalaz:92], Gonzalaz, R.C. & Woods , R.E.(1992),(Digital Image Processing), 3rd edition, Addison-Wesley publishing Company , Inc.

[9] [Gonzalaz:2000], Gonzalaz, R.C. & Woods, R.E. (2000), (Digital Image Processing), Pearson Eduction Asia Pte Ltd.

[10] [Awcock:1995], Awcock,G J. and Thomas, R., 1995, (Applied Image Processing), Department of Electrical and Electronic Engineering, University of Brighten.

[11] [NIBLACK: 86], Niblack W., (1986), (An Introduction to Digital Image Processing), Prentice-Hall international Engleweed Cliffs, New Jersey.

[12] [Justusson 1978], Justusson, 1978, (Noise Reduction by Median Filtering), in Proc.4IJCPR, pp.502-504.

[13] [Ros:69], Rosenfeld, A.& Kak, A.C.(1969), (Picture Processing by Computer), Academic Press, Newyork.

[14] [Prewitt:1970], Prewitt, J.M., 1970, (Object Enhancement and Extraction in Picture Processing and Psychopictorics), B.S.Lipkin and A.Rosenfield, Eds.), p. Academic Press, New York.

[15] [Davis:1975], Davis, L.S., Rosenfeld, A., and Weszka, J.,1975, (Region Extraction by Averaging and Thresholding), IEEE Trans.Syst.Man Cybern. SMC-55,383.

[16] [Ros:1976], Rosenfeld, A.& Kak, A.C. (1976), (Digital Picture Processing), Computer Science and Applied Mathematics Series), Academic Press, Newyork.