Software Implementation Of Hybrid Median Filter

Hussien Ali Hussien*

Abstract

Image filtering is one of the fundamental problems in the field of image processing and computer vision. Filtering an image improves the details and consequently the usability of an image.

In this paper the algorithm of hybrid median filter is presented and then software is implemented using Matlab package.

The features of the implemented hybrid median filter are compared with normal median filter under different conditions and circumstances and from testing results many points are concluded.

Key words: Software Implementation , hybrid median filter , computer vision

* College Of Engineering /Al –Mustansriya University

1. Introduction

When an image is acquired by a camera or other imaging system, often the vision system for which it is intended is unable to use it directly. The image may be corrupted by random variations in intensity, variations in illumination, or poor contrast that must be dealt with in the early stages of vision processing [1].

This paper discusses some of the methods for image enhancement aimed at eliminating these undesirable characteristics.

Then one of these methods which is called hybrid median filter is software implemented using Matlab package and the features of this method are compared with normal median filter for different images and for various noise types.

Finally, from testing results many points are concluded.

2. Types of noise [2, 3, 4]

In this section four different noise types will be discussed as given below:

1. Salt an pepper noise:

It is called also impulsive noise, shot noise, or binary noise. This degradation can be caused by sharp, sudden disturbance in the image signal, its appearance is randomly scattered white or black (or both) pixels over the image.

2. Gaussian noise

Gaussian noise is an idealized form of white noise, which is caused by random fluctuations in the signal. We can observe white noise by watching a television which is slightly mistuned to a particular channel. Gaussian noise is white noise which is normally distributed. If the image is represented as I, and the Gaussian noise by N, then we can model a noisy image by simply adding the two:

$$I + N \qquad (1)$$

3. Speckle noise

Whereas Gaussian noise can be modeled by random values added to an image, speckle noise (or more simply just speckle) can be modeled by random values multiplied by pixel values, hence it is also called multiplicative noise. Speckle noise is a major problem in some radar applications.

4.Periodic noise

If the image signal is subject to a periodic, rather than a random disturbance, we might obtain an image corrupted by periodic noise. The effect is of bars over the image.

Salt and pepper noise, Gaussian noise and speckle noise can all be removed by using spatial filtering techniques. Periodic noise, however, requires image transforms for best results.

3. Filtering Techniques [2, 3, 4, 5, 6, 7]

In image processing, filters are mainly used to suppress either the high frequencies in the image, i.e. smoothing the image, or the low frequencies, i.e. enhancing or detecting edges in the image. An image can be filtered either in the frequency or in the spatial domain. The first approach involves transforming the image into the frequency domain, multiplying it with the frequency filter function and re-transforming the result into the spatial domain. The filter function is shaped so as to attenuate some frequencies and enhance others. For example low pass function is 1 for frequencies smaller than the cut-off frequency and 0 for all others. The corresponding process in the spatial domain is to convolve the input image f (i,j) with the filter function h(i,j).

This can be written as:

$$g(i,j) = h(i,j) \otimes f(i,j) \quad (2)$$

The mathematical operation is identical to the multiplication in the frequency space, but the results of the digital implementations vary,

since the filter function must be approximated with a discrete and finite kernel (also called impulse response).

The discrete convolution can be defined as a 'shift and multiply' operation, where the kernel is shifted over the image and then its values is multiplied with the corresponding pixel values of the image. For a square kernel with size M×M , the output image can be

calculated with the following formula:

$$g(i,j) = \sum_{m=-\frac{M}{2}}^{\frac{M}{2}} \sum_{n=-\frac{M}{2}}^{\frac{M}{2}} h(m,n)\ f(i-m, j-n) \quad (3)$$

Various standard kernels exist for specific applications, where the size and the form of the kernel determine the characteristics of the operation. The most important of them are discussed below:

1. Mean filter

The mean filter is a simple sliding-window spatial filter that replaces the center value in the window with the average (mean) of the pixel values in the window. The window, or kernel, is usually square but can be any shape. An example of mean filtering of a single 3 × 3 window of values is given below:

If the unfiltered values are as shown,

5	3	6
2	1	9
8	4	7

then by taking the mean of the nine values:

5 + 3 + 6 + 2 + 1 + 9 + 8 + 4 + 7 = 45

45 / 9 = 5

Then the mean filtered output is as shown,

*	*	*
*	5	*
*	*	*

The center value (previously 1) is replaced by the mean of the nine values (5). For more than 3 × 3 window the same procedure can be applied.

2. median filter

The median filter is also a sliding-window spatial filter, but it replaces the center value in the window with the median of the pixel values in the window. As for the mean filter, the kernel is usually square but can be any shape. An example of median filtering of a single 3 × 3 window of values is given below:

If the unfiltered values are as shown,

6	2	0
3	97	4
19	3	10

then by taking the median of the nine values:

0 , 2 , 3 , 3, 4 , 6 , 10 , 19 , 97

Then the median filtered output is as shown,

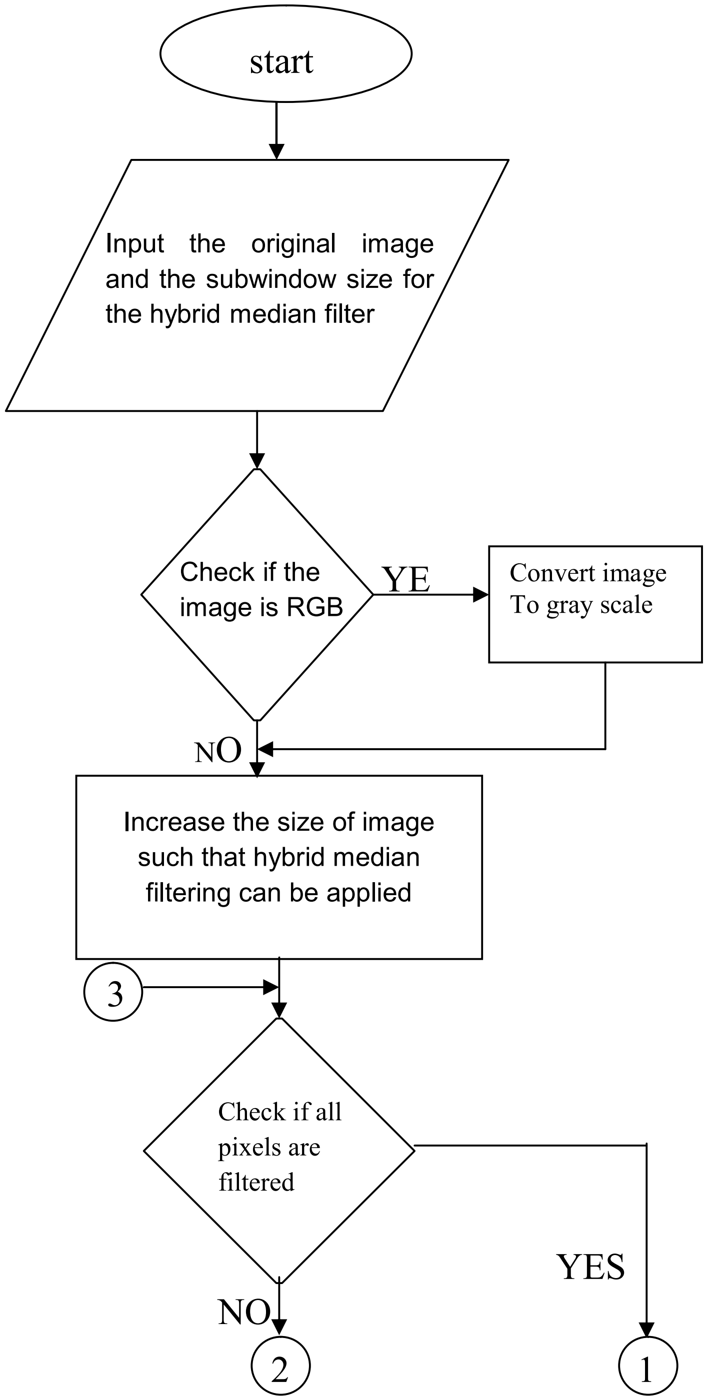
*	*	*
*	4	*
*	*	*

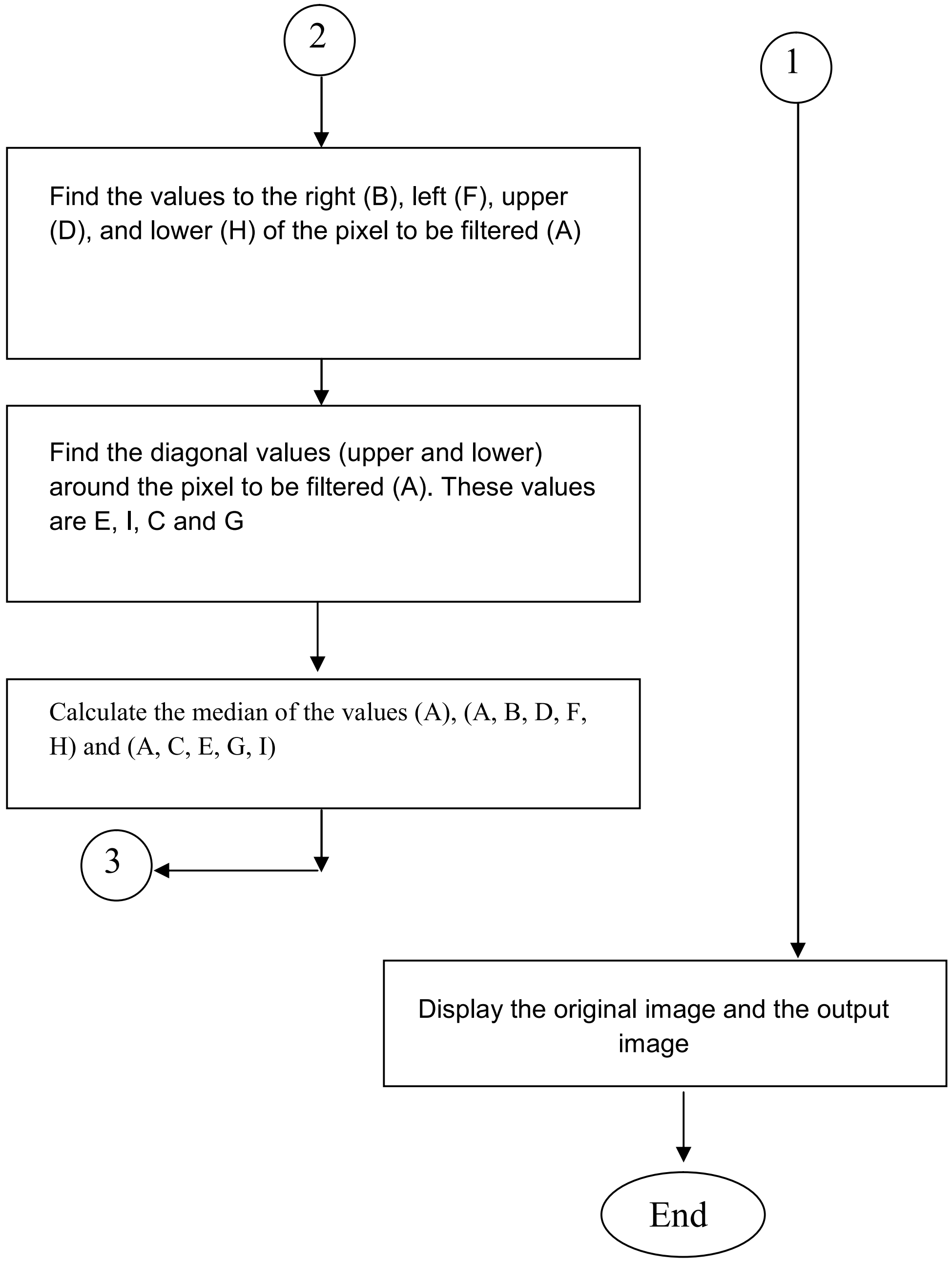
Center value (previously 97) is replaced by the mean of the nine values (4). For more than 3 × 3 window the same procedure can be applied.

It is clear that for the first (top) example, the median filter would also return a value of 5, since the ordered values are 1, 2, 3, 4, 5, 6, 7, 8, 9. For the second (bottom) example, the mean filter returns a value of 16 since the sum of the nine values in the window is 144 and 144 / 9 = 16. This illustrates one of the celebrated features of the median filter which is its ability to remove 'impulse' noise (outlying values, either high or low). The median filter is also widely claimed to be 'edge-preserving' since it theoretically preserves step edges without blurring. However, in the presence of noise it does blur edges in images slightly.

4. Hybrid median filtering

This technique is a modified version of median filtering technique. This technique gives better "edge-preserving" characteristics than normal median filter. The process of hybrid median filtering is best understood by the following flow chart.


start
Input the original image and the subwindow size for the hybrid median filter
Check if the image is RGB
YE
Convert image To gray scale
NO
Increase the size of image such that hybrid median filtering can be applied
3
Check if all pixels are filtered
YES
NO
2
1


2
1
Find the values to the right (B), left (F), upper (D), and lower (H) of the pixel to be filtered (A)
Find the diagonal values (upper and lower) around the pixel to be filtered (A). These values are E, I, C and G
Calculate the median of the values (A), (A, B, D, F, H) and (A, C, E, G, I)
3
Display the original image and the output image
End

5. Results and Discussion

The flow chart in the previous section is software implementd. Then it is tested under different conditions and circumstances and from testing results many points are concluded. The features of hybrid median filter are compared with normal median filter to clarify the effect of hybrid median filter on noisy images.

5.1 Cleaning salt and pepper noise using hybrid median filter

The original image for this case is show in Figure (2).
Two noise levels are considered which are:
1. 0.1.
2. 0.3.

The performance of hybrid median filter is compared with normal median filter for the following cases:
1. Subwindow size of 5 × 5.
2. Subwindow size of 9 × 9.

The filtered images are show in Figure (3) to (6) and from these figures the following points are concluded:
1. The performance of hybrid median filter in general is better than normal median filter.
2. By increasing noise level it is clear that hybrid median filter is doing well with edge-preserving, however, for small size subwindow one can observe that normal median filter is better in noise removal than hybrid median filter.

5.2 Cleaning Gaussian noise using hybrid median filter

The original image for this case is shown in Figure (7).
The performance of hybrid median filter is compared with normal median filter for the following cases:
1. Subwindow size of 5 × 5.
2. Subwindow size of 7 ×7.
3. Subwindow size of 9 × 9.

The filtered images are shown in Figure (8) to (10).

From these figures it is clear that hybrid median filter is better than normal median filter of all cases. However, by increasing subwindow size the output image is degraded.

5.3 Cleaning spickle noise hybrid median filter

The original image for this case is shown in Figure (11). The same cases considered in subsection (5.2) are applied here.

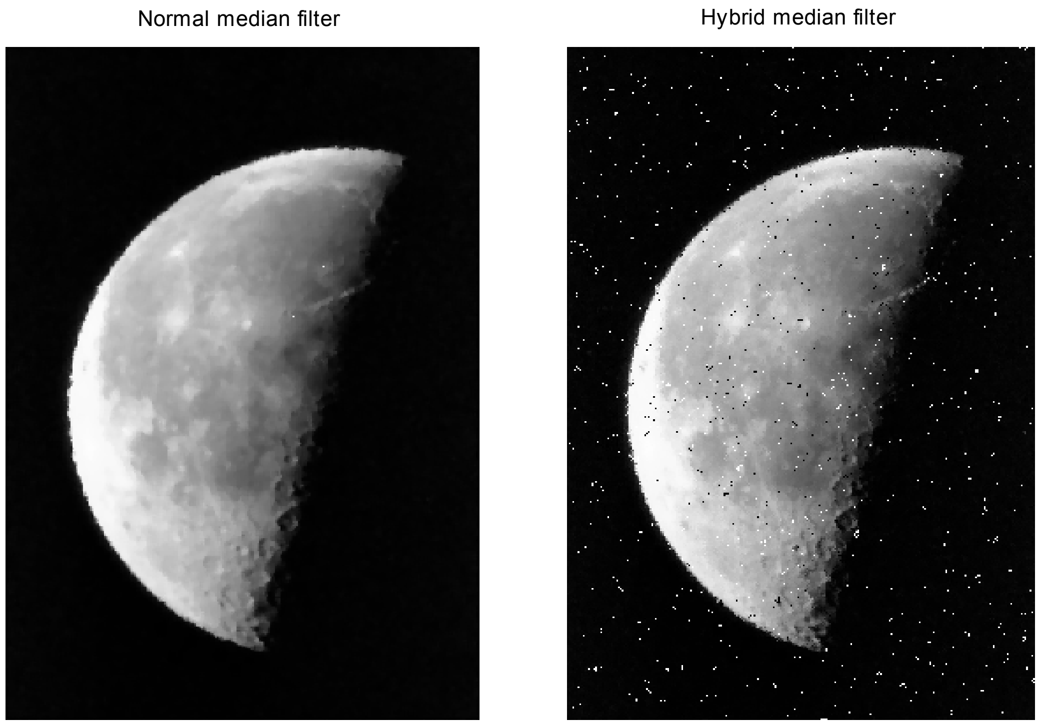
The filtered images are shown in Figure (12) to (14). From these figures, it is clear that hybrid median filter is better than normal median filter. However, by increasing subwindow size the output image is degraded.

original image

Figure (2) The original image for salt and pepper noise case.

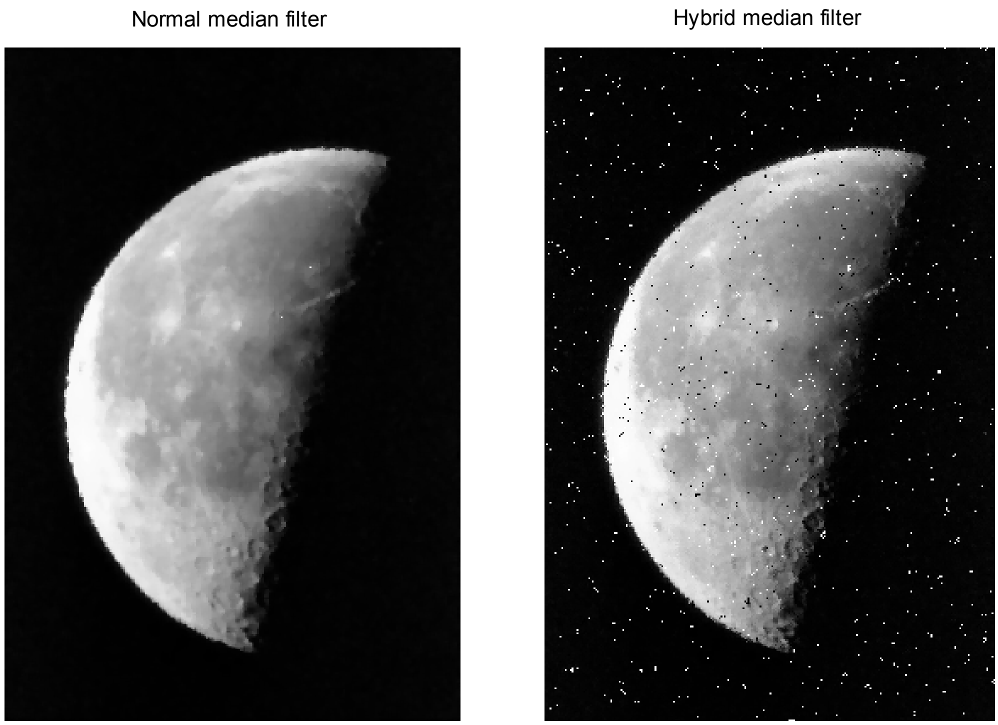
(a) (b)

Figure (3) The Output of: a) median filter b) hybrid median filter for noise level = 0.1 and subwindow size is 5×5.



(a) **(b)**

Figure (4) The Output of: a) median filter b) hybrid median filter for noise level = 0.1 and subwindow size is 9×9.



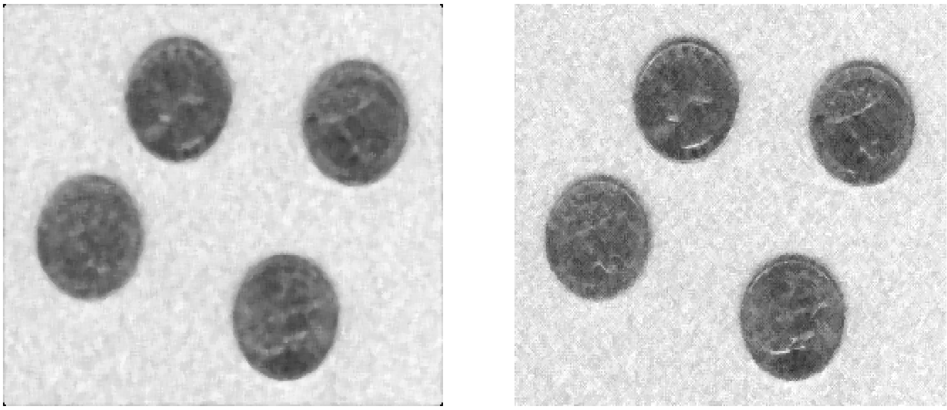
(a) (b)

Figure (5) The Output of: a) median filter b) hybrid median filter for noise level = 0.3 and subwindow size is 5×5.

(a) (b)

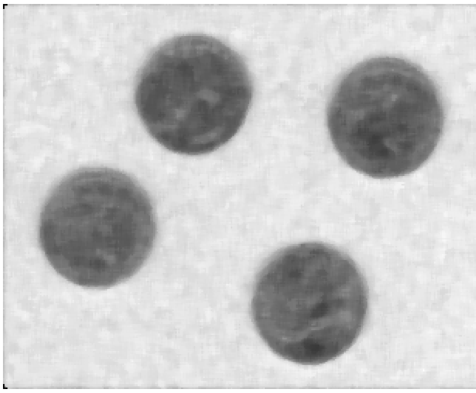
Figure (6) The Output of: a) median filter b) hybrid median filter for noise level = 0.3 and subwindow size is 9×9.

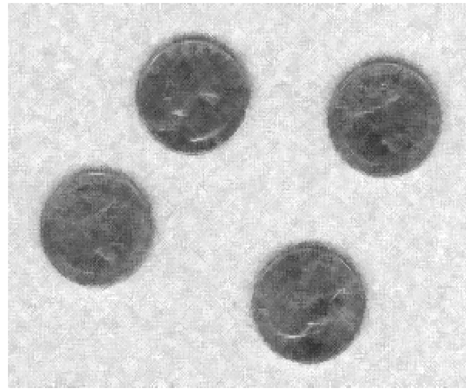
Figure (7) The original image for Gaussian noise case.



(a) (b)

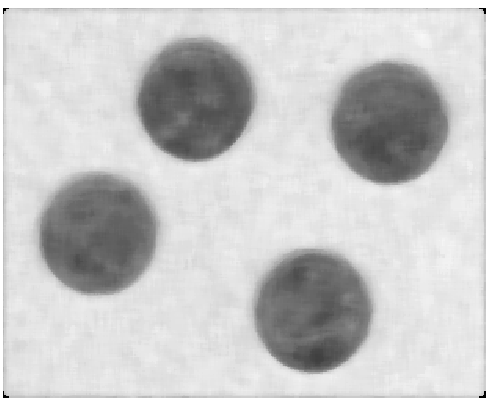
Figure (8) The Output of: a) median filter b) hybrid median filter for Gaussian noise and subwindow size is 5×5.

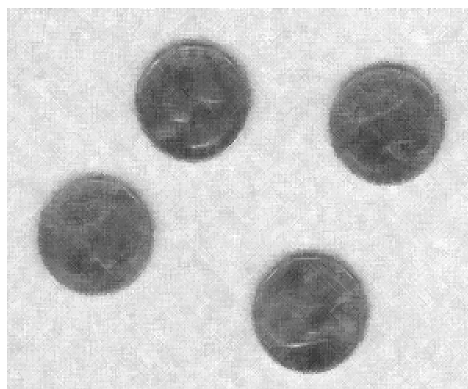




(a) (b

Figure (9) The Output of: a) median filter b) hybrid median filter for Gaussian noise and subwindow size is 7×7.





(a) (b)

Figure (10) The Output of: a) median filter b) hybrid median filter for Gaussian noise and subwindow size is 9×9.

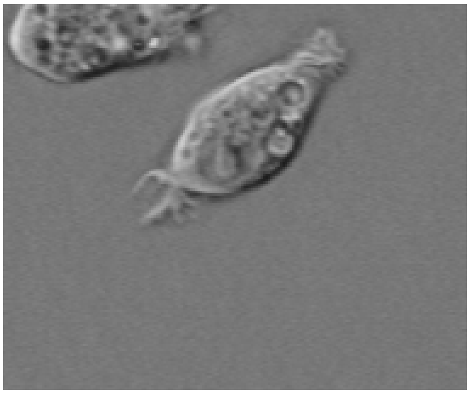
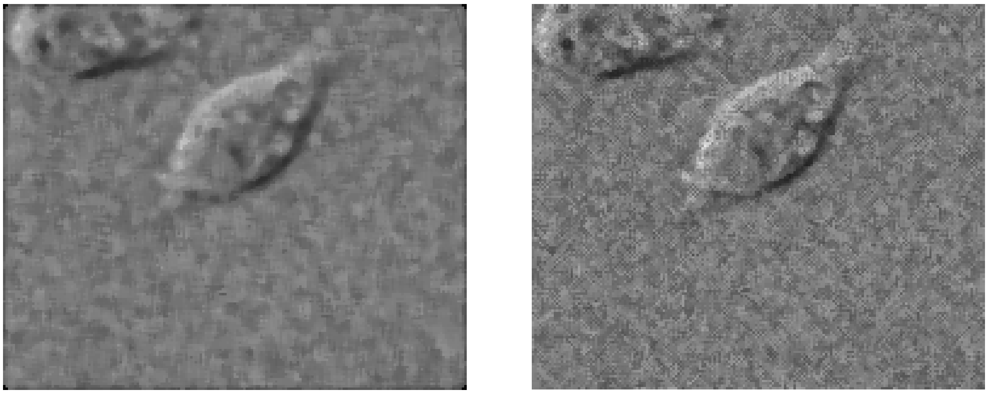
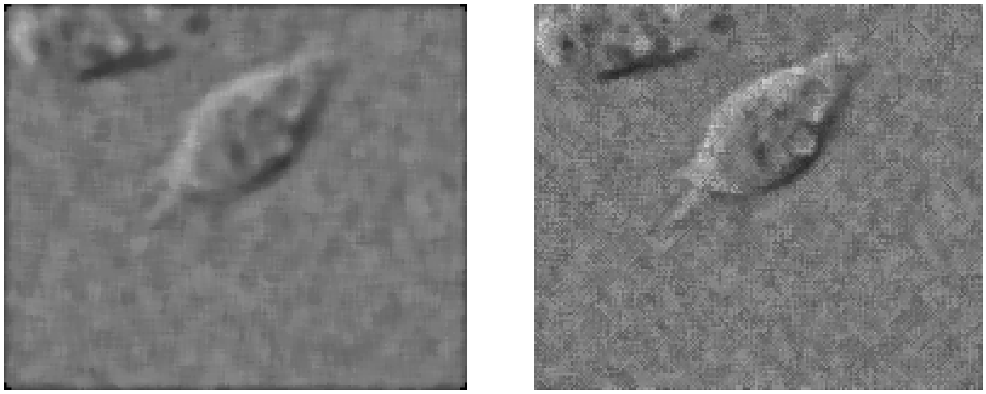


Figure (11) The original image for Spickle noise case.



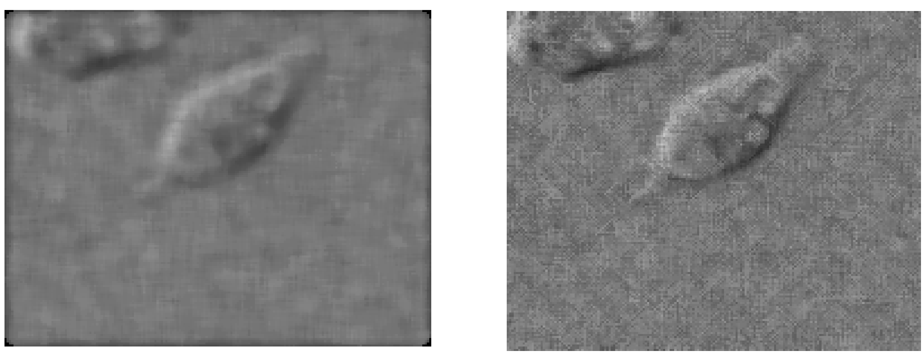
(a) (b)

Figure (12) The Output of: a) median filter b) hybrid median filter for Spickle noise and subwindow size is 5×5.



(a) (b)

Figure (13) The Output of: a) median filter b) hybrid median filter for Spickle noise and subwindow size is 7×7



(a) (b)

Figure (14) The Output of: a) median filter b) hybrid median filter for Spickle noise and subwindow size is 9×9.

6. Conclusions

There are many points concluded from this work which are:

1. For the case of salt and pepper noise, the performance of hybrid median filter in general is better than normal median filter. By increasing noise level it is clear that hybrid median filter is doing well with edge-preserving, however, for small size subwindow one can observe that normal median filter is better in noise removal than hybrid median filter.

2. For the case of Guassian noise, it is clear that hybrid median filter is better than normal median filter of all cases. However, by increasing subwindow size the output image is degraded.

3. For the case of Spickle noise, it is clear that hybrid median filter is better than normal median filter of all cases. However, by increasing subwindow size the output image is degraded.

7. References

1. Baxes, Gregory A., Digital Image Processing, Wiley, 1994.

2. Davis, E.R., Machine Vision Theory, Algorithms, Practicalities, Elsevier, 2006.

3. Gonzalez, Rafael C., and Richard. E. Woods, Digital Image Processing, 3rd edition, Addison-Wesley, 2008.

4. Russ, John C., The Image Processing Handbook, sixth Edition, CRC Press, 2007.

5. Parker, J.R., Algorithms for Image Processing and Computer Vision, 2nd edition, John Wiley & Sons, 2011.

6. Awcock, G.J. and R.Thomas, Applied Image Processing, McGraw-Hill, Inc., 1996.

7. Gonzalez, R., and Woods, R., Digital Image Processing, 2nd edition, Prentice hall, 2002.

البناء الحاسوبي لخوارزمية مرشح الوسيط الهجين

م.م.حسين علي حسين*

المستخلص

أن أزالة الضوضاء من الصور هو أحد المشاكل الجوهرية في مجال معالجة الصور والرؤية الحاسوبية. أن أزالة الضوضاء من الصور يؤدي الى زيادة دقة المعالم للصور وتحسين التفاصيل و من ثم زيادة المنفعة والفائدة من الصور.

في هذا البحث تم تقديم و دراسة خصائص أحدى خوارزميات المرشحات المستخدمة في أزالة الضوضاء من الصور. أن هذه الخوارزمية هي خوارزمية مرشح الوسيط الهجين (hybrid median filter).

تم في هذا البحث بناء برنامج بأستخدام الحزمة البرمجية (Matlab) يمثل خوارزمية (hybrid median filtering).

تم مقارنة خصائص مرشح الوسيط الهجين (hybrid median filter) مع خصائص مرشح الوسيط الأعتيادي (normal median filter) في حالات وظروف مختلفة ومن نتائج الأختبار تم الأستنتاج عدة نقاط.

* الجامعة المستنصرية، كلية الهندسة، قسم الهندسة الكهربائية