Analysis of Color Image Filtering Methods

Jihad Nader, PhD
Al-Balqa' Applied University
Faculty of Engineering Technology
Amman – Jordan

Ziad A. A. Alqadi
Al-Balqa' Applied University
Faculty of Engineering Technology
Amman – Jordan

Bilal Zahran
Al-Balqa' Applied University
Faculty of Engineering Technology
Amman – Jordan

ABSTRACT
The quality of the RGB color image degrades from the minute it is captured to the time it is displayed to the human observer. The image is subject to many kinds of distortions during the stages that it might pass through such as storing, processing, compressing, and transmitting, thus enhancing the image quality is a major important issue. This paper will introduce the effects of salt&pepper and Gaussian noises. A comparative experimental analysis will be done and some recommendation will be approved in which filter to use to reduce the effect of the noise.

General Terms
Color image filtering methods

Keywords
Adaptive median filter, median filter, average filter, noise, PSNR..

1. INTRODUCTION
The RGB color model is an additive color model in which red, green and blue light are added together in various ways to reproduce a broad array of colors. The name of the model comes from the initials of the three additive primary colors, red, green and blue.

The main purpose of the RGB color model is for the sensing, representation and display of images in electronic systems, such as televisions and computers, though it has also been used in conventional photography. Before the electronic age, the RGB color model already had a solid theory behind it, based in human perception of colors.

RGB is a device-dependent color model: different devices detect or reproduce a given RGB value differently, since the color elements (such as phosphors or dyes) and their response to the individual R, G and B levels vary from manufacturer to manufacturer, or even in the same device over time. Thus an RGB value does not define the same color across devices without some kind of color management.

An RGB image sometimes referred to as a "true color" image and it can be stored as an m-by-n-by-3 data array that defines red, green and blue color components for each individual pixel. RGB images do not use a palette. The color of each pixel is determined by the combination of the red, green and blue intensities stored in each color plane at the pixel's location [1], [2].

2. COLOR IMAGE QUALITY
The quality of the RGB color image degrades from the minute it is captured to the time it is displayed to the human observer. The image is subject to many kinds of distortions during the stages that it might pass through such as storing, processing, compressing, and transmitting, etc...[3], [4] RGB color image may be affected by different kinds of noises such as salt&pepper noise and Gaussian noise. These noises can affect 1 or more channels in RGB color image, thus disturbing image contrast brightness and making the color image not suitable for seeing [5[,[6], because quality of the image was decreased. Here in this paper we will focus on salt&pepper and Gaussian noises.

Salt-and-pepper noise is a form of noise sometimes seen on images. It presents itself as sparsely occurring white and black pixels. An effective noise reduction method for this type of noise is a filter such as average filter [6], [7].

Gaussian noise is statistical noise having a probability density function (PDF) equal to that of the normal distribution, which is also known as the Gaussian distribution[8]. In other words, the values that the noise can take on are Gaussian-distributed. A special case is white Gaussian noise, in which the values at any pair of times are identically distributed and statistically independent (and hence uncorrelated). In communication channel testing and modeling, Gaussian noise is used as additive white noise to generate additive white Gaussian noise [9],[10].

In telecommunications and computer networking, communication channels can be affected by wideband Gaussian noise coming from many natural sources, such as the thermal vibrations of atoms in conductors (referred to as thermal noise or Johnson-Nyquist noise), shot noise, black body radiation from the earth and other warm objects, and from celestial sources such as the Sun.

Principal sources of Gaussian noise in digital images arise during acquisition e.g. sensor noise caused by poor illumination and/or high temperature, and/or transmission e.g. electronic circuit noise [11], [12]. In digital image processing Gaussian noise can be reduced using a spatial filter, though when smoothing an image, an undesirable outcome may result in the blurring of fine-scaled image edges and details because they also correspond to blocked high frequencies. Conventional spatial filtering techniques for noise removal include: mean (convolution) filtering, median filtering and Gaussian smoothing [11],[12].

2.1 Filtering an RGB color image, with a linear spatial filter consists of the following steps:
1. Extract the three component images:

```
>>fR= fc( : , : , 1); fG= fc( : , : , 2); fB= fc( : , : , 3);
```

2. Filter each component images individually. Letting w represent a filter generated using fspecial, we smooth the red component images as follows:

```
>> fR_filtered= imfilter(fR, w);
```

3. Reconstruct the filter RGB images:

```
>>fc_filtered= cat(3 , fR_filtered, fG_filtered, fB_filtered);
```

2.2 Peak Signal-to-Noise Ratio (PSNR)

The PSNR is often used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the filtered image, or the reconstructed image[10].

The Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR) are the two error metrics used to compare image compression quality. The MSE represents the cumulative squared error between the compressed and the original image, whereas PSNR represents a measure of the peak error. The lower the value of MSE, the lower the error and the higher value of PSNR is the higher of image quality.

To compute the PSNR, we can use the following equations [15], [16]:

MSE(X,Y)=SUM(1,n)SUM(1,m)SUM(1,p) [X(i,j,k)-Y(i,j,k)]

PSNR(X,Y)=10.log(d.^2 / MSE)

3. RGB COLOR IMAGE FILTERS

Here we will focus on the most popular filters used to filter RGB color image [13], [14]:

3.1 Average filter

Replace each pixel by the average of pixels in a square window surrounding this pixel.

3.2 Median filter

Taking the median value instead of the average or weighted average of pixels in the window Median: sort all the pixels in an increasing order, take the middle one.

3.3 Adaptive median filter

The adaptive median filtering has been introduced [10] as an improvement to the standard median filtering, as we explained before that the Median filtering can detect the noise but in the same it can't differentiate between the fine details and the noise. So the main idea in the Adaptive Median Filter is to perform a spatial processing to determine which pixels in an image have been affected by impulse noise, and run the filter only in this pixel. The Adaptive Median Filter classifies pixels as noise by comparing each pixel in the image to its surrounding neighbor pixels. The size of the neighborhood is adjustable, as well as the threshold for the comparison. A pixel that is different from a majority of its neighbors, as well as being not structurally aligned with those pixels to which it is similar, is labeled as impulse noise. These noise pixels are then replaced by the median pixel value of the pixels in the neighborhood that have passed the noise labeling test.

Keep in mind that the output of the filter is a single value used to replace the value of the pixel at (x,y), the particular point on which the window Sxy is centered at given time.

Consider the following notation:

- ✓ **zmin = minimum gray level in Sxy**
- ✓ **zmax = maximum gray level in Sxy**
- ✓ **zmed = median of gray levels in Sxy**
- ✓ **zxy = gray level at coordinates (x, y)**
- ✓ **Smax =maximum allowed size of Sxy**

The adaptive median filtering algorithm works in two levels, denoted level A and level B, as follows:

 Level A:

- ✓ *A1 = zmed – zmin*
- ✓ *A2 = zmed – zmax*
- ✓ If *A1* > 0 and *A2* < 0, Go to level B
- ✓ Else increase the window size
- ✓ If window size ≤ *Smax* repeat level A
- ✓ Else output *zxy*

 Level B:

- ✓ *B1 = zxy – zmin*
- ✓ *B2 = zxy – zmax*
- ✓ If *B1* > 0 and *B2* < 0, output *zxy*
- ✓ Else output *zmed*

The following matlab function can be used to implement this filter:

```
function f = adpmedian(g, Smax)
if (Smax <= 1) | (Smax/2 == round(Smax/2)) | (Smax
~= round(Smax))
 error('SMAX must be an odd integer > 1.')
end
[M, N] = size(g);
f = g; f(:) = 0;
alreadyProcessed = false(size(g));
for k = 3:2:Smax
 zmin = ordfilt2(g, 1, ones(k, k));
 zmax = ordfilt2(g, k * k, ones(k, k));
 zmed = medfilt2(g, [k k]);
 processUsingLevelB = (zmed > zmin) & (zmax >
zmed) & ...
 ~alreadyProcessed;
 zB = (g > zmin) & (zmax > g);
 outputZxy = processUsingLevelB & zB;
 outputZmed = processUsingLevelB & ~zB;
 f(outputZxy) = g(outputZxy);
 f(outputZmed) = zmed(outputZmed);
 alreadyProcessed = alreadyProcessed |
processUsingLevelB;
 if all(alreadyProcessed(:))
 break;  end end
```

4. IMPLEMENTATION AND EXPERIMENTAL RESULTS

A matlab code was written to implement the previous mentioned filters, different color images were used by applying salt&pepper noise with different densities and Gaussian noise for one or more channels of the RGB color image. Fig. 1 shows the corrupted (with salt&pepper noise) and the resulting images of each filter.

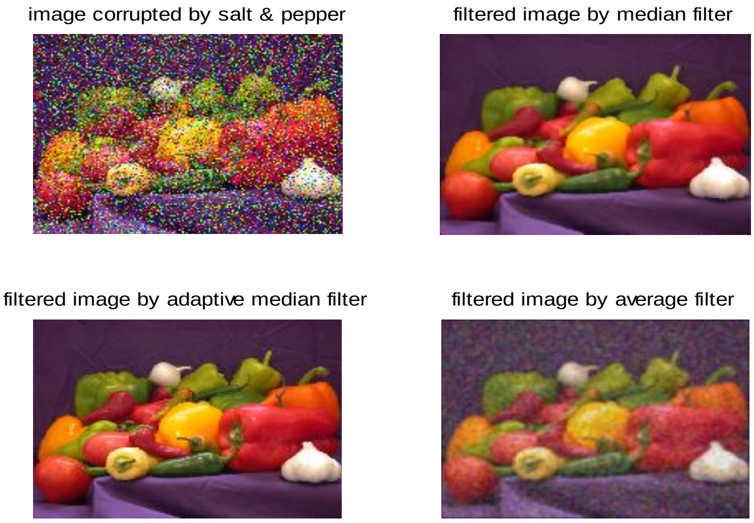

Fig 1: Applying filters to reduce salt&pepper noise

Fig.2 shows the R, G, and B color distribution of the same image treated in Fig.1;

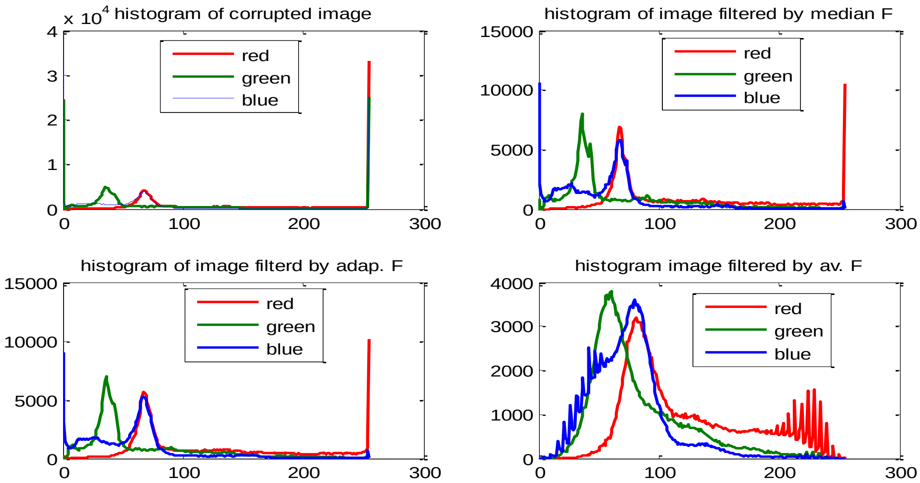

Fig.2: R, G, and B color distribution of the images treated in Fig.1

The same image was de-noised by Gaussian noise and Fig.3 shows the results of filtering:

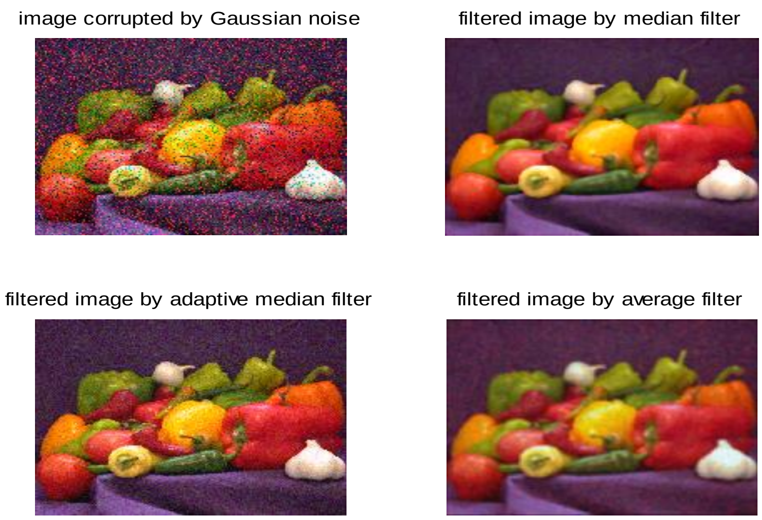

Fig.3: Fig.1: Applying filters to reduce Gaussian noise

Fig.4 shows the R, G, and B color distribution of the same image treated in Fig.3;

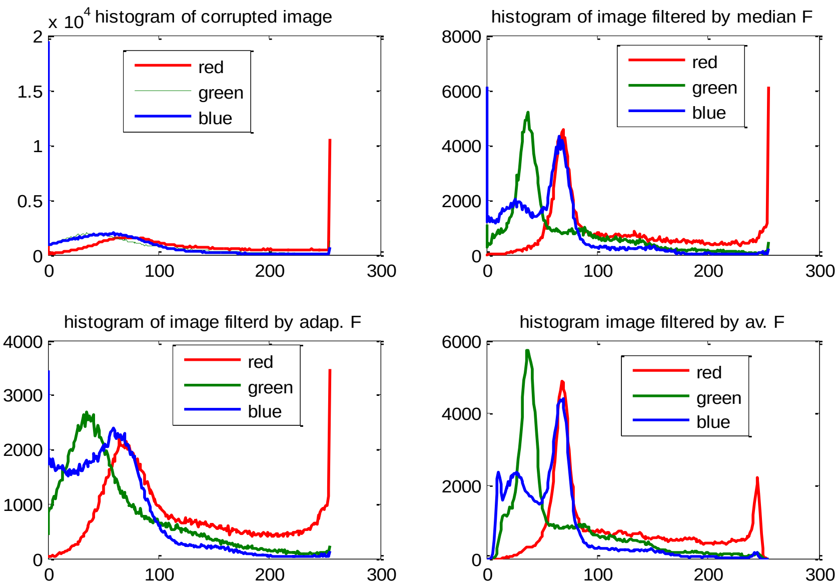

Fig.4: R, G, and B color distribution of the images treated in Fig.3

We varied the salt&pepper density and each time we calculate PSNR, the experimental results for this case are listed in table 1 (all the color image channels were affected by the noise):

Table 1: PSNR values for different values of salt&pepper noise density

Case	Salt&Pepper density	PSNR Adaptive filter	PSNR mean filter	PSNR average filter
1	0.1	**93.4534**	72.1088	58.0699
2	0.2	**85.0001**	69.0496	506201
3	0.3	**79.2870**	66.5321	44.8585
4	0.4	**73.4154**	63.5492	40.2328
5	0.5	**69.1835**	61.2619	36.4097
6	0.6	**62.4470**	55.5019	33.1252
7	0.7	**49.6734**	42.8051	30.4022
8	0.8	**34.5087**	28.5381	27.7985
9	0.9	22.0510	18.1334	**25.7079**

Then we mixed the salt&pepper noise with Gaussian noise and the obtained results are listed in table 2.

Table 2: Affected with noise channels and PSNR values (0, 0,G::means channels R and G are affected with 0 density of salt&pepper noise and blue channel affected with Gaussian noise.

Case	Red channel	Green channel	Blue channel	PSNR: Adaptive median filter	PSNR: Median filter	PSNR: Average filter
1	0.25	0.25	0.25	81.5059	67.4412	47.6356
2	0.25	0.4	0,5	74.1682	64.5091	39.7938
3	0.4	0	0.5	73.1636	63.4131	40.3096
4	0,4	0.5	0,5	70.9441	62.3115	37.1349
5	0	0	0.25	89.1706	70.9729	55.2115
6	0	0.25	0.25	84.9792	69.3826	50.1141
7	**G**	**G**	**G**	54.1049	**66.4300**	63.9470
8	0.2	G	G	58.1125	**67.2891**	58.2257
9	0.2	0.2	G	**68.3770**	64.5739	53.6471
10	G	0.2	G	57.8981	**67.6796**	56.8587

From the results shown in table 1 we can see that adaptive median filter gave the best results of PSNR, except for higher values of salt&pepper density(>=0.9)(case 9) here it is better to use average filter. Fig.5 shows the relationship between PSNR and salt&pepper noise density for various filters.

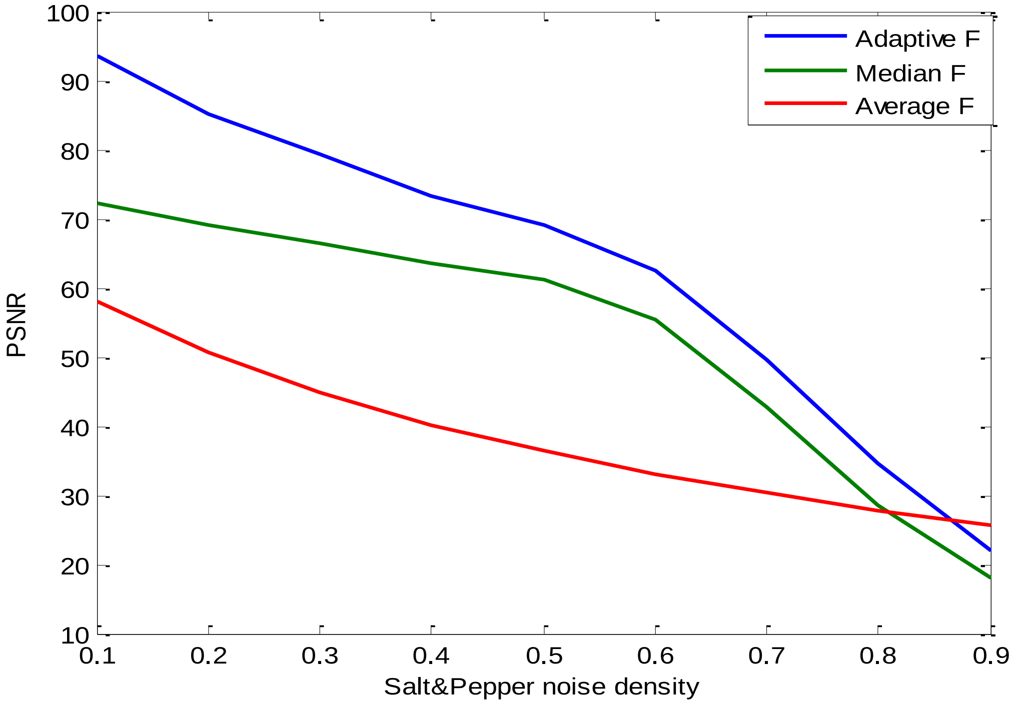

Fig.5 shows the relationship between PSNR and salt&pepper noise density for various filters.

From the results listed in table 2 we can see the following:

- ✓ To reduce the effects of salt&pepper noise with density <0.9 it is better to use adaptive median filter.
- ✓ If 2 or 3 channels of the RGB color image are affected with Gaussian noise it is better to use median filter (cases 7, 8 and 10).

5. CONCLUSIONS

A comparative analysis of adaptive median, median and average filters was performed based on the obtained values of PSNR. Refereeing to obtained experimental results we can conclude the following:

- If the 3 channels of the RGB color image were affected by salt&pepper noise with density less than 0, 9 it is preferable to use adaptive median filter.
- If the 3 channels of the RGB color image were affected by salt&pepper noise with density greater than or equal 0, 9 it is preferable to use average filter.
- If 2 or 3 channels of the RGB color image are affected with Gaussian noise it is preferable to use median filter
- If one channel of the RGB color image is affected with Gaussian noise, and the other 2 channels are affected with salt&pepper noise with density less than 0.9 it is preferable to use adaptive median filter.

6. REFERENCES

[1] Majed O. Al-Dwairi, Ziad A. Alqadi, Amjad A. AbuJazar and Rushdi Abu Zneit, Optimized True-Color Image Processing, World Applied Sciences Journal 8 (10): 1175-1182, 2010 ISSN 1818-4952.

[2] Alqadi, Ziad A.; Moustafa, Akram A.; Alduari, Majed; Zneit, Rushdi abu Zneit, True Color Image Enhancement Using Morphological Operations, International Review on Computers & Software;Sep2009, Vol. 4 Issue 5, p557.

[3] J. Astola and P. Kuosmaneen. Fundamentals of nonlinear Digital Filtering. Boca Raton, FL: CRC, 1997.

[4] H. Hwang and R. A. Hadded. Adaptive median filter: New algorithmsand results, IEEE Trans. Image Process., Vol. 4, No. 4, pp. 499–502,Apr. 1995.

[5] S. Zhang and M. A. Karim. A new impulse detector for switching median filters, IEEE Signal Process. Lett., Vol. 9, No. 11, pp. 360–363, Nov. 2002.

[6] P. E. Ng and K. K. Ma. A switching median filter with boundary discriminative noise detection for extremely corrupted images, IEEE Trans. Image Process., Vol. 15, No. 6, pp. 1506–1516, Jun. 2006.

[7] K. S. Srinivasan and D. Ebenezer. A new fast and efficient decisionbased algorithm for removal of high density impulse noise, IEEE Signal Process. Lett., Vol. 14, No. 3, pp. 189–192, Mar. 2007.

[8] V. Jayaraj and D. Ebenezer. A new switching-based median filtering scheme and algorithm for removal of high-density salt and pepper noise in image, EURASIP J. Adv. Signal Process., 2010.

[9] K. Aiswarya, V. Jayaraj, and D. Ebenezer. A new and efficient algorithm for the removal of high density salt and pepper noise in images and videos, Second Int. Conf. Computer Modeling and Simulation, pp. 409–413, 2010.

[10] Medida.Amulya Bhanu, Gopichand Nelapati, Dr.Rajeyyagari Sivaram, Salt and Pepper Noise Detection and removal by Modified Decision based Unsymmetrical Trimmed Median Filter for Image Restoration, International Journal of Advanced Trends in Computer Science and Engineering, Volume 1, No.3, July – August 2012.

[11] Jump up to *Philippe Cattin (2012-04-24).* "Image Restoration: Introduction to Signal and Image Processing". *MIAC, University of Basel*. Retrieved 20 December 2016.

[12] Jump up *Robert Fisher; Simon Perkins; Ashley Walker; Erik Wolfart.* "Image Synthesis — Noise Generation". Retrieved 20 December 2016

[13] Haitham Alasha'ary , Abdullah Al-Hasanat, Khaled Matrouk ,Ziad Al-Qadi ,Hasan Al-Shalabi, A Novel Digital Filter for Enhancing Dark Gray Images, European Journal of Scientific Research ISSN 1450-216X / 1450-202X Vol.122 No.1, 2014, pp.99-106.

[14] Alqadi, Ziad A.; Moustafa, Akram A.; Alduari, Majed; Zneit, Rushdi abu Zneit,, True Color Image Enhancement Using Morphological Operations, International Review on Computers & Software;Sep2009, Vol. 4 Issue 5, p557.

[15] Yusra A. Y. Al-Najjar, Dr. Der Chen Soong, Comparison of Image Quality Assessment: PSNR, HVS, SSIM, UIQI, International Journal of Scientific & Engineering Research, Volume 3, Issue 8, August-2012 1 ISSN 2229-5518.

[16] C. A. B. R. H. S. a. P. E. S. Z. Wang, "Image quality assessment: from error visibility to structural similarity," IEEE Trans. Image Processing, vol. 13, no. 4, pp. 600-612, 2004.