A Comparative Study of Mixed Noise Removal Techniques

Ajay Kumar Nain\(^1\), Surbhi Singhania\(^2\), Shailender Gupta\(^3\) and Bharat Bhushan\(^4\)

Electronics and Communication Engineering Department, YMCA University of Science and Technology, Faridabad, India
\(^1\)aknain90@gmail.com, \(^2\)surbhidec24@gmail.com, \(^3\)shailender81@gmail.com, \(^4\)bhrts@gmail

Abstract

Mixed noises are a characteristic of combined noises acting on a single carrier. Various mechanisms in recent past have been given in literature to restore images corrupted with Poisson and impulse mixed noise. This paper compares mixed noise removal techniques such as: Peer Group averaging (PGA), Vector Median Filter (VMF), Vector Direction Filter (VDF), Fuzzy Peer Group Averaging (FPGA), and Fuzzy Vector Median Filter (FVMF) on the basis of performance metrics such as Peak Signal to Noise Ratio (PSNR), Mean Absolute Error (MAE), Mean Square Error (MSE) and time complexity. The image size and the noise density is varied so as record these performance metrics. All the above mentioned techniques were implemented in MATLAB-11. The simulation and result shows that FVMF introduces blurring of edged but provide an output of highest PSNR value, especially for large sized images. However, for smaller images PGA provides best results of PSNR and hence a good quality of de-noised image. Also it is observed that with increase in image size the quality of the resulting image improves as the value of PSNR also increases but on increasing the impulse noise density with constant image size the image quality decreases with a constant decrease in the PSNR value.

Keywords: Non-Liner Filter, MSE, MAE, PSNR, Color image De-noising, impulse noise, poison noise

1. Introduction

Noise \([4]\) is a result of errors in image acquisition which causes random variations in its brightness. It may be caused due to film grain in case of digital cameras acquisition or electronic transmission faults as observed in television broadcasting. Various researchers \([6-8]\) have proposed mechanism to remove single noise from images. Over the last few years the research has focused on removal of Poisson corrupted with impulse mixed noise \([1-4]\). Poisson noise \([4, 6]\) or photo shot noise is caused by random variation of photons, which cause more photons to enter one sensor than the other. In real world photography, if enough images are taken, it will be seen that the deviation in intensity found for each image follows the well-known Poisson distribution. In effect, we cannot be sure that the intensity we have measured in a particular image represents the "true" intensity as we know that this value will deviate from the average. It is this deviation which is considered to be the noise associated with the image. As the deviation is known to follow a Poisson distribution, we know that the likely deviation will be plus or minus the square root of the signal intensity measured. Thus, if we measure a signal intensity of one hundred photons, then the noise on this signal will be ten photons. If we measure a signal intensity of one thousand photons in the image, then the noise on this signal will be about thirty one photons.

Impulse noise \([12-13, 29]\) on the other hand is created due to transmission discrepancy. For instance, in case of satellite transmission over long distances impulse noise is
prominent. Also known as the “salt and pepper noise”, it replaces the pixels with a zero or maximum pixel value leaving black and white spots on the image. Impulse noise density is defined as the number of pixels distorted with respect to a certain image size.

Many researchers have worked on attempting to remove a single type of noise [6-8], and fair amount of work has also been done on removing mixed noises [4-5, 12, 29-31] as follows:

- The simplest way to remove mixed noise is to apply separate filters [17, 27] on the noise but this leads to high computational complexity. Therefore, there have been methods to remove mixed noises with lower complexity.


- The Fuzzy Vector Median Filter (FVMF) [1] performs a weighted averaging where the weight of each pixel is computed according to its similarity to the robust vector median.

This paper is a comparative study of all the above mentioned techniques to remove Poisson-Impulse noise. For this purpose all the techniques were implemented in MATLAB-11. The results of PSNR, MAE, MSE and time complexity were recorded by varying image size and noise density. At last we show which technique is the best in terms of given performance metric.

The rest of the paper has been divided into the following sections. Section 2 describes the techniques that are compared in this paper for mixed noise removal. Section 3 explains the simulation setup parameters used for comparison of the above mentioned techniques. Section 4 illustrates the results followed by conclusion and references.

2. Techniques Compared

To have better understanding for users we would like to explain the techniques used in this paper for comparison purpose in detail. The techniques that were implemented in MATLAB are as follows:

2.1. Average Median Filter (AMF)

This is the simplest type of filter to remove noise from image. In this technique, each pixel is replaced by arithmetic mean of neighboring pixels. The explanation of averaging theory is that images typically vary slowly over space, so neighboring pixels are likely to have similar values, and it is therefore appropriate to average them together. But this assumption of slow variation in pixels fails at edges and does not preserve them. This can remove light noise but introduces blurriness in image.

2.2. Weiner filter

It is an adaptive low-pass filter which uses a pixel-wise adaptive Wiener method based on statistics estimated from a local neighbour-hood of each pixel. For colour image it can be implemented on red, green and blue colour planes separately.
2.3. Vector Median Filter (VMF)

Astola et al. [2] introduced the vector median filter as an extension of the median filter to multivariate data. For an observation window $\Omega = \{x_1, x_2, \ldots, x_N\}$, the output of the vector median filter is defined as

$$x_{VM} = \arg \min_{i=1}^{N} \| x - x_i \|_p,$$

where $\| \cdot \|_p$ denotes the Lp norm. This technique is simple and fast but only suitable for lightly corrupted color image. The impulse response of this type of filter is zero; therefore, it is good for removal of impulse noise.

2.4. Vector Directional Filter

Trahanias and Venetsanopoulos [34] proposed a vector directional filter (VDF) for directional processing, which is a generalized basic vector directional filter (BVDF). For an observation window, the output of theBVDF is defined as:

$$x_{BVD} = \arg \min_{i=1}^{N} A(x, x_i),$$

Where $A(x, x_i)$ denotes the angle between $x$ and $x_i$. The computation of the XBVD is similar to that of the XVM except that the angle between vectors is used to compute the distance, i.e., $d(x) = \sum_{i=1}^{N} A(x, x_i)$. This technique provides better result than VMF but not suitable for heavily corrupted noisy image.

2.5. Peer Group Averaging Filter (PGA)

This is another non-linear filter where the output is the weighted average of the peer group members of the pixel under evaluation. The aim of this filter is to average over a peer group rather than the whole window [11] because the latter leads to blurring of edges and fine details. Peer group is defined as the most similar pixels. The distance with respect to central element is calculated as in equation-3 and sorted similar to the method mentioned in VMF mechanism.

$$d_i(n) = \| x(n) - x_i(n) \|, \quad i = 0, \ldots, k$$

where $d_0(n) \leq d_1(n) \leq \cdots \leq d_k(n)$

After sorting, one method is to select a threshold value [3]. The peer group $P(n)$ of size $m(n)$ for $x(n)$ is defined as

$$P(n) = \{x_i(n), i = 0, 1, \ldots, m(n) - 1\}$$

The value of $m(n)$ decides the efficiency of PGA filtering [3]. One approach is to select a fixed number of members in each window while other approach is to use Fisher Linear Discriminant (FLD). Here we used fixed approach by selecting $m(n)$ as 3/4th of window size.

2.6. Fuzzy Vector Median Filter (FVMF)

Utilizing the techniques of fuzzy set theory, Y. Shen and K. E. Barner proposed fuzzy vector median (FVM) based surface smoothing [1] which utilize the information regarding the spread of samples in image pixels. In this technique one membership function $\mu$ is used to calculate degree of pixel as below:
where $\alpha$ controls the spread of membership function. Then output is calculated as:

$$F_{FVMF} = \frac{\sum_{i=1}^{N} F_{i} \mu(F_{o}, F_{i})}{\sum_{i=1}^{N} \mu(F_{o}, F_{i})}$$

where $F_{o}$ is the central pixel and $F_{i}$ is the current pixel in neighbourhood. This method provides more accurate results but this technique is not suitable for high density Gaussian noise in image.

2.6. Fuzzy Peer Group Averaging (FPGA)

S. Morillas et al., [12] proposed FPGA for removing mixed noise from colour image. In that paper, a fuzzy peer group concept which extends the peer group concept in the fuzzy setting is introduced. A fuzzy peer group will be defined as a fuzzy set that takes a peer groups support set and where the membership degree of each peer group member will be given by its fuzzy similarity with respect to the pixel under processing. The fuzzy peer group of each image pixel will be determined by means of a novel fuzzy logic-based procedure. Then output is calculated by weighting averaging operation as below where weighting coefficients for each pixel vector is its membership degree.

$$F_{FPGA} = \frac{\sum_{i=1}^{m} F_{i} \times \sum_{k=1}^{m} F_{P_{mk}}}{\sum_{i=1}^{m} \sum_{k=1}^{m} F_{P_{mk}}}$$

This technique performs well on mixed noisy image but it tends to blur the edges of image.

3. Experimental setup

3.1. Performance Metrics

- Peak Signal to noise Ratio- is the measure of peak error. It is an expression used to depict the ratio [4, 5] of maximum possible power of image (signal) and the power of the corrupting noise that affects the quality of its representation. It is represented in terms of mean square error as:

$$PSNR = 10 \log_{10}\left(\frac{MAX_{I}^{2}}{MSE}\right)$$

$MAX_{I}$ is the maximum possible pixel value of the image. It is equal to 255 for 8 bit represented image.

- Mean square error- is the cumulative squared error between the final, de-noised image and the original image before introduction of noise. This enables us to compare mathematically as to which method provides better results under same conditions like image size noise, etc. It is mathematically stated as:

$$MSE = \frac{1}{mn} \sum_{y=1}^{n} \sum_{x=1}^{m} [I(x, y) - I'(x, y)]^2$$

Where the image size is $m \times n$. 
• Mean absolute error- is the absolute error between the original image and the de-noised image obtained after applying one of the filters. It is used to measure the closeness to the true or original value of the pixel with respect to the de-noised pixel. It is given by:

\[ MAE = \frac{1}{mn} \sum_{y=1}^{m} \sum_{x=1}^{n} | I(x,y) - I'(x,y) | \]

• Image Quality- The original image and the de-noised image were placed side by side to compare the variance in degradation of image quality. This was tested on the images of varying sizes.

• Time Complexity- is used to define the time taken by each method under varying parametric conditions like image size, noise density, etc. Time complexity defines the complexity of each algorithm and is hence used to define the algorithm with least and maximum computational cost.

3.2. Simulation Setup

Algorithms were developed in MATLAB to simulate the methods for filtering an image consisting of dual noise. The value of variance was varied and hence, a comparative result generated. The setup parameters are as shown below:

<table>
<thead>
<tr>
<th>Component</th>
<th>Parameter</th>
<th>Value of parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image</td>
<td>Image Size</td>
<td>128x128 pixel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>256x256 pixel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>512x512 pixel</td>
</tr>
<tr>
<td></td>
<td>Type</td>
<td>RGB</td>
</tr>
<tr>
<td>Noise</td>
<td>Type</td>
<td>Impulse + Poisson</td>
</tr>
<tr>
<td></td>
<td>Variance of impulse noise</td>
<td>0.05-0.2</td>
</tr>
<tr>
<td></td>
<td>Step size</td>
<td>0.05</td>
</tr>
<tr>
<td>Processor</td>
<td>Type</td>
<td>i5-64 bit</td>
</tr>
<tr>
<td></td>
<td>RAM</td>
<td>2 Gb</td>
</tr>
<tr>
<td></td>
<td>Speed</td>
<td>2370 MHz</td>
</tr>
<tr>
<td>Software</td>
<td></td>
<td>MATLAB 7.12.0</td>
</tr>
</tbody>
</table>

4. Result

The following tables and graphs present in this section depict a thorough comparison of all the methods. The table compares the values of PSNR, MSE and MAE for the de-noised images on using different method for variable image sizes and noise density. In the qualitative analysis, visual comparison between original and final de-noised images can be made. The purpose of graphs is to pictorially represent the trend of the increasing or decreasing PSNR on variation in the above mentioned parameters. We would discuss each one by one as follows:

4.1. Impact on PSNR, MSE, MAE

The Table shows the overall values of PSNR, MAE and MSE for different image sizes and noise density. The following inferences can be drawn:
On increasing the impulse noise factor while keeping the image size constant, PSNR decreases, MSE and MAE increases. This is for the reason that the ratio of image size to noise density decreases with increase in image size and constant noise, therefore the output image is lesser de-noised.

On increasing the image size with constant impulse noise density, PSNR increases, MSE and MAE decreases. This is because the ratio of image size to noise density increases with increasing image size and constant noise, therefore the output image is better de-noised.

For small images of size 128x128 and 256x256 PGA gives best PSNR and hence least MSE and MAE. But for image size of 512x512 FVMF gives marginally better PSNR and corresponding least MSE and MAE.

The least PSNR and maximum MAE and MSE are given by VDF in all image cases.

On increasing image size with constant impulse noise density, PSNR increases, MSE and MAE decreases. This is because the ratio of image size to noise density increases with increase in image size and constant noise, therefore the output image is recovered. To show this we used only PGA technique since the results were same for the others.

### Figure 1. Comparison of PGA

#### 4.2. Qualitative Analysis

Table III shows the image quality of various mixed de-noising techniques for different parameters. The following inferences can be drawn:

- The image quality of de-noised image decreases with increase in impulse noise density for constant sized image (see Fig. z).
- The image quality of de-noised image increases with the increase in image size for a constant impulse noise density (see Table III- any method from left to right).
- FVMF causes maximum blurring in images when compared with other techniques (see Table III: (r)-(s)).
- PGA gives visually best results (see Table III: (n)-(q)). On the other hand FPGA preserves edges better as can be seen (see Table III: (v)-(y)).

#### 4.3. Time complexity

Increasing image size increases the time taken for execution of every method that is time complexity increases.
- Time Complexity is maximum for FPGA and minimum for AMF and Wiener Filter.
- For a certain image size, time complexity is independent of noise density.

### Table II. PSNR, MAE, MSE Results

#### 128x128

<table>
<thead>
<tr>
<th></th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>VDF</td>
<td>22.7</td>
<td>349.201</td>
<td>13.081</td>
<td>20.346</td>
<td>600.329</td>
<td>17.441</td>
<td>18.696</td>
<td>877.875</td>
<td>21.116</td>
<td>17.785</td>
<td>4.2E+03</td>
<td>24.7721</td>
</tr>
</tbody>
</table>

#### 256x256

<table>
<thead>
<tr>
<th></th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
</tr>
</thead>
</table>

#### 512x512

<table>
<thead>
<tr>
<th></th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
<th>PSNR</th>
<th>MSE</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMF</td>
<td>27.6257</td>
<td>275.533</td>
<td>8.0899</td>
<td>24.5144</td>
<td>229.955</td>
<td>11.6768</td>
<td>22.5958</td>
<td>357.6844</td>
<td>14.5677</td>
<td>21.1361</td>
<td>5.01E+02</td>
<td>17.1639</td>
</tr>
</tbody>
</table>
Table III. Qualitative Result

<table>
<thead>
<tr>
<th>Original Image (128X128)</th>
<th>5%</th>
<th>10%</th>
<th>15%</th>
<th>20%</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>(b)</td>
<td>(c)</td>
<td>(d)</td>
<td>(e)</td>
</tr>
<tr>
<td>VMF</td>
<td>(f)</td>
<td>(g)</td>
<td>(h)</td>
<td>(i)</td>
</tr>
<tr>
<td>VDF</td>
<td>(j)</td>
<td>(k)</td>
<td>(l)</td>
<td>(m)</td>
</tr>
<tr>
<td>PGA</td>
<td>(n)</td>
<td>(o)</td>
<td>(p)</td>
<td>(q)</td>
</tr>
<tr>
<td>FVMF</td>
<td>(r)</td>
<td>(s)</td>
<td>(t)</td>
<td>(u)</td>
</tr>
<tr>
<td>FPGA</td>
<td>(v)</td>
<td>(w)</td>
<td>(x)</td>
<td>(y)</td>
</tr>
</tbody>
</table>
5. Conclusion

In this paper a comparative study of mixed de-noising techniques is done. The paper takes various performance metrics such as MSE, PSNR, MAE and time complexity to evaluate the efficacy of these techniques. From all the above discussion the following points can prove beneficial for the researchers working in the direction of image processing as follows:

- When the image size is small it is advisable to use PGA but if the image size is large then its better to use FVMF as was also shown from our results.
- In terms of visual clarity, maximum blurring occurs when FVMF technique is used while PGA gives best visual quality results.
- As far as computational complexity is concerned best results were obtained for AMF filter followed by Wiener Filter and the least results were obtained for FPGA filter.
- Moreover, computational complexity is independent of impulse noise density.

References