SIViP (2013) 7:111–118
DOI 10.1007/s11760-011-0210-3

ORIGINAL PAPER

Salt-and-pepper noise removal by adaptive median-based lifting filter using second-generation wavelets

P. Syamala Jaya Sree · Pradeep Kumar · Rajesh Siddavatam · Ravikant Verma

Received: 14 June 2010 / Revised: 12 January 2011 / Accepted: 13 January 2011 / Published online: 27 February 2011

Abstract In this paper, we propose a novel adaptive median-based lifting filter for image de-noising which has been corrupted by homogeneous salt and pepper noise. The median-based lifting filter removes the noise of the input image by calculating the median of the neighboring significant pixels. The algorithm for image noise removal uses the lifting scheme of the second-generation wavelets in conjunction with the proposed adaptive median-based lifting filter. The experimental results demonstrate the efficiency of the proposed method. The proposed algorithm is compared with all the basic filters, and it is found that our method outperforms many other algorithms and it can remove salt and pepper noise with a noise level as high as 90%. The algorithm works exceedingly well for all levels of noise, as illustrated in terms of peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) measures.

Keywords Adaptive median filter · Salt and pepper noise · Second-generation wavelets · Lifting filter

P. S. J. Sree · R. Siddavatam · R. Verma
Department of Computer Science & IT,
Jaypee University of Information Technology, Waknaghat,
Solan, HP, 173215, India
e-mail: jayasree.syamala@gmail.com

R. Siddavatam
e-mail: srajesh@juit.ac.in

P. Kumar (✉)
Department of Electronics and Communication Engineering,
Jaypee University of Information Technology,
Waknaghat, Solan, HP, 173215, India
e-mail: pradeep.kumar@juit.ac.in

1 Introduction

In the process of image acquisition and transmission, the digital images are often corrupted with impulse noise. This is mainly because of the errors in the sensors or in the communication channel. It is highly imperative that this noise be removed prior to its subjection for further processing, such as edge detection, image segmentation, object recognition, and pattern recognition. Various methods [1–10] have been developed for impulse noise removal from corrupted images. Median filters have been extensively used for the removal of impulse noises. They are simple yet very effective in the removal of salt- and pepper-type impulse noise. Median filters often tend to modify good pixels too; therefore, impulse detection algorithms play a crucial role in noise removal.

In [4], the progressive switching median (PSM) filter has been developed for removing impulse noise from highly corrupted images. It works by using an impulse detection algorithm and then iteratively detecting and filtering impulse noise, and hence, it performs heavy computation. In [5], a detail-preserving adaptive median filter has been proposed for image processing. In [6], alpha-trimmed mean has been used in impulse noise detection and then the noisy pixels are corrected with the help of original value of the pixel and the median of the window of the noisy pixel.

Second-generation wavelets developed by Swelden [7] and [11] have been efficiently used for many applications of image processing. Generating set of most significant samples for de-noising and then using them to generate an image is a highly non-linear and computationally expensive task. A set of significant de-noising samples is obtained after estimating the detail coefficients. Highly sparse, noise-removed significant samples are used to approximate an image. In [8], a decision-based median filter, which consists of two functions decision making and noise filtering, has been used for noise removal. In [9], an adaptive median-based filter has been used for noise removal from images corrupted with various kinds of noises. In [10], an iterative procedure utilizing adaptive center-weighted median filter has been used for removing random-valued impulse noise.

The general scheme followed for noise removal using first-generation wavelets is described as follows. Consider an original image denoted as A and noisy image denoted as B. Here, we assume that the original image A is corrupted by homogeneous salt and pepper noise resulting in the image B. Hence, we get a model of the type

$$B(x, y) = A(x, y) + \varepsilon(x, y) \quad (1)$$

where ε is homogeneous salt and pepper noise.

In this paper, we have proposed a novel median-based lifting filter using second-generation wavelets which is combination of work [2] and [3] developed by Siddavatam Rajesh et al. We use the lifting scheme first for separating the significant pixels from the insignificant pixels and then for filtering the corrupted image using the median-based lifting filter. The lifting filter was used in [2] by us for noise removal. Upon improving the lifting filter in [2] by calculating the median value of the neighboring pixels of the corrupted pixel in its window, we were able to improve the results previously generated in [2]. The proposed algorithm gives excellent results for all levels of noise. The rest of the paper is organized as follows. Section 2 describes the proposed adaptive median-based lifting filter. In Sect. 3, the theoretical framework for second-generation wavelets is elaborated. In Sect. 4, our proposed lifting algorithm for image noise removal has been given. In Sect. 5, the significance measures PSNR and SSIM have been described to assess the quality of the de-noised images. The efficiency of our proposed method with results has been shown in Sect. 6. Finally, Sect. 7 concludes the work.

2 Proposed adaptive median-based lifting filter

2.1 Review of the adaptive median filter [1]

Let $a_{x,y}$, for $(x, y) \in A \equiv \{1, \ldots, M\} \times \{1, \ldots, N\}$, be the gray level of a true M –by –N image, A at pixel location (x, y), and $[s_{\min}, s_{\max}]$ be the dynamic range of A, i.e., $s_{\min} \le a_{x,y} \le s_{\max}$ for all $(x, y) \in A$ (Fig. 1).

Let B is a noisy image in the classical salt-and-pepper noise model, the observed gray level at pixel location (x, y) given by:

$$b_{x,y} = \begin{cases} s_{\min} & \text{with probability} \quad p \\ s_{\max} & \text{with probability} \quad q \\ a_{x,y} & \text{with probability} \quad 1-r \end{cases}$$

where $r = p+q$ defines the noise level. Let $S^{w}_{x,y}$ be a window size of $w \times w$ centered at (x, y) i.e.,

$$S^{w}_{x,y} = \{(k, l) : |k - i| \le w \quad \text{and} \quad |j - l| \le w\}$$

And let $w_{\max} \times w_{\max}$ be the maximum window size. The algorithm tries to identify the noise candidates and then replace each $b_{x,y}$ by the median of the pixels in $S^{w}_{x,y}$.

2.2 Algorithm for adaptive median filter

For each pixel location do the following:

1. Initialize $w = 3$.
2. Compute $S^{\min,w}_{x,y}$, $S^{\text{med},w}_{x,y}$ and $S^{\max,w}_{x,y}$, which are the minimum, median, and maximum values of the pixel values in the window, respectively.
3. If $S^{\min,w}_{x,y} < S^{\text{med},w}_{x,y} < S^{\max,w}_{x,y}$, then go to step 5, otherwise set $w = w + 2$.
4. If $w \le w_{\max}$, go to step 2, otherwise replace $b_{x,y}$ by $S^{\text{med},w_{\max}}_{x,y}$.

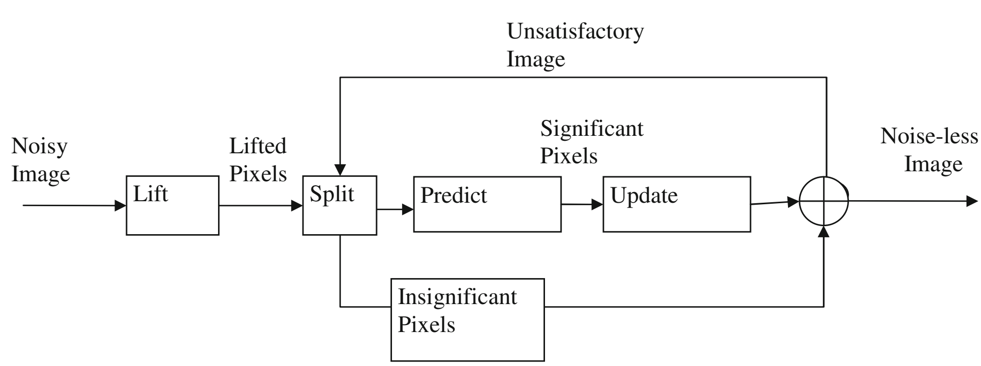

Fig. 1 General framework of lifting filter

5. If $S^{\min,w}_{x,y} < b_{x,y} < S^{\max,w}_{x,y}$, then $b_{x,y}$ is not a noise candidate, else replace $b_{x,y}$ by $S^{\text{med},w}_{x,y}$.

The most seminal contributions for the adaptive median filter related to noise smoothing filter for images have been elaborated in [12–14], and [15].

The adaptive structure of the filter ensures that most of the impulse noise is detected at a high noise level provided that the window size is large enough. Notice that the noise candidates are replaced by the median, while the remaining pixels are left unaltered.

3 Theoretical framework for second-generation wavelets

The first-generation wavelets could be easily used for periodic and infinite domain signals. But there was no clear-cut way of using it for bounded domain signals. Thus, the second-generation wavelets came into picture. The second-generation wavelets have all the useful properties like time-frequency localization and fast implementation of the first-generation wavelets in addition to being able to represent the signals which are bounded. This has been achieved by removing the translation and dilation of the mother wavelet. Instead, we use our *Lifting Scheme* proposed by Siddavatam Rajesh et al. in [3]. The main advantage of second-generation lifting is that it does not use any Fourier analysis and all functions are derived in the spatial domain. The use of spatial domain results in an intuitively appealing solution.

Lifting scheme is better suited for image de-nosing as it can easily be generalized to complex geometric situations of high non-uniformity. The lifting scheme is a tool for constructing second-generation wavelets [7] and [11], which are no longer, dilates and translates of one single function the mother wavelet. In contrast to first-generation wavelets, which used the Fourier transform for wavelet construction, a construction using lifting is performed exclusively in spatial domain, and thus, wavelets can be custom-designed for complex domains like irregular noise samples.

A new mathematical formulation proposed by Swelden [16] based on spatial construction of the wavelets is called the lifting-based wavelet transform. The underlying principle of this approach [16,17] is to break up the high-pass and the low-pass wavelet filter into a sequence of smaller filters that in turn can be converted into a sequence of alternating upper and lower triangular matrices and a diagonal matrix with constants. The factorization is obtained by using an extension of the Euclidean algorithm. The resulting formulation can be implemented by means of banded matrix multiplications [18].

Let $\tilde{h}(z)$ and $\tilde{g}(z)$ be the low-pass and high-pass analysis filters and $h(z)$ and $g(z)$ be the low-pass and high-pass synthesis filters. The polyphase representation of the filter h is expressed as (Fig. 2):

$$h(z) = h_e(z^2) + z^{-1}h_o(z^2) \tag{2}$$

where h_e contains the even filter coefficients and h_o contains the odd filter coefficients of the FIR filter. Similarly, the polyphase representation of the filters $g(z)$, $\tilde{h}(z)$, and $\tilde{g}(z)$ is expressed as follows:

$$g(z) = g_e(z^2) + z^{-1}g_o(z^2) \tag{3}$$
$$\tilde{h}(z) = \tilde{h}_e(z^2) + z^{-1}\tilde{h}_o(z^2) \tag{4}$$
$$\tilde{g}(z) = \tilde{g}_e(z^2) + z^{-1}\tilde{g}_o(z^2) \tag{5}$$

Based on the formulation in Eq. (2)–(5) , the polyphase matrix representation of the filters can be given as follows:

$$P(z) = \begin{bmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{bmatrix}$$

$$\tilde{P}(z) = \begin{bmatrix} \tilde{h}_e(z) & \tilde{g}_e(z) \\ \tilde{h}_o(z) & \tilde{g}_o(z) \end{bmatrix}$$

The two matrices, i.e., $P(z)$ and $\tilde{P}(z)$, are called dual of each other. The two polyphase matrices must satisfy the following condition for perfect reconstruction:

$$P(z)\tilde{P}(z^{-1})^T = 1 \tag{6}$$

When the determinant of $P(z)$ is unity, the synthesis filter bank (h, g) is called complementary and so is the analysis filter pair $(\tilde{h}, \tilde{g})$. However, when $h(z) = \tilde{h}(z) = g(z) = \tilde{g}(z) = 1$, the DWT simply splits an input signal into two subsequences, one with all the odd samples and one with all the even sequences. This is called the lazy wavelet transform. The lifting scheme [16] is an easy relationship between perfect reconstruction filter pairs that have the same low-pass or high-pass filter. One can then start from the Lazy wavelet and use lifting to gradually build one's way up to a multiresolution analysis with particular properties (Fig. 3).

The lifting technique used in the present approach is primal lifting, which lifts the low-pass subband with the help of high- pass subband. According to the lifting theorem [17,18], if the wavelet filter pair (h, g) is complementary, then any other filter g^{new} that is complementary is of the form

$$g^{\text{new}}(z) = g(z) + h(z)s(z^2) \tag{7}$$

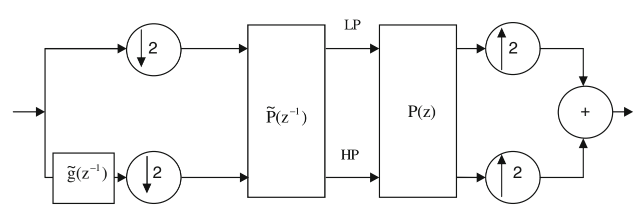

Fig. 2 Polyphase representation of wavelet transform: first subsample into even and odd, then apply the dual polyphase matrix. For the inverse transform: first apply the polyphase matrix and then join even and odd

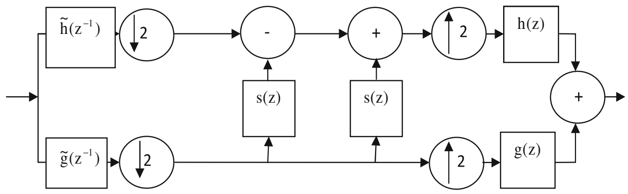

Fig. 3 The primal lifting scheme: first applying a lazy wavelet transform and then lifting the low-pass subband with the help of the high-pass subband

The polyphase matrix of $g^{\text{new}}(z)$ can be defined as

$$P^{\text{new}}(z) = \begin{bmatrix} h_e(z) & g_e(z) + h_e(z)s(z) \\ h_o(z) & g_o(z) + h_o(z)s(z) \end{bmatrix}$$

Using Eq. (6), the above polyphase matrix of $g^{\text{new}}(z)$ can be modified as

$$P^{\text{new}}(z) = [P(z)]^{-1} \begin{bmatrix} 1 & 0 \\ -s(z^{-1)} & 1 \end{bmatrix}$$

Hence, the lifting created a new low-pass filter

$$\tilde{h}^{\text{new}}(z) = \tilde{h}(z) - \tilde{g}(z)s(z^{-2}) \tag{8}$$

The lifting wavelet transform essentially means first applying the lazy wavelet transform on the input stream, then executing primal lifting, and finally scaling the output streams to produce low-pass and high-pass subbands.

4 Proposed lifting algorithm

Wavelets and approximations of the data can be constructed on the basis of this hierarchical structure of partitioning. In this second-generation wavelet representation, the process does not depend on a regular setting for the data; therefore, it can be used in both the regular and irregular data sets for noise sampling. This is an important advantage of the lifting scheme.
The proposed lifting algorithm is performed in two steps:

4.1 Algorithm I : image lifting algorithm

Consider a data set to be partitioned into two groups: significant pixels sig and insignificant pixels $insig$. If the original pixel set can be partitioned in a hierarchical structure, then the above process can be iteratively applied to different sets. A hierarchical structure has the following form

$$sig^0 \subset sig^1 \subset sig^2 \cdots \subset sig^n \tag{9}$$

where sig^n denotes the finest representation of the geometry. sig^n can be partitioned into sig^{n-1} and $insig^{n-1}$; then sig^{n-1} can be partitioned into sig^{n-2} and $insig^{n-2}$ and so on. Note that the larger number in superscripts represents finer resolution.

$$sig^k \cup insig^k = sig^{k+1}, k = 0, 1, 2, \ldots, n-1 \tag{10}$$

4.1.1 Median-based lifting filter (med_lift)

The median-based lifting filter (med_lift) is used for finding the correct filtered value by which a corrupted noisy pixel should be replaced. For finding the filtered value for a noisy pixel under consideration, the median-based lifting filter considers a window around the noisy pixel and finds the filtered median value of the non-impulse noise pixels which are neighbors of the corrupted pixel and are present in that window. In case there is no non-impulse neighbor of the corrupted pixel in the current window, the filter increases the window size to the next possible higher value and calculates the filtered median value. The median-based lifting filter always uses a window size of 3×3; the first it determines the filtered value for a corrupted pixel. As mentioned above if there is no non-impulse neighbor of the corrupted pixel in this window of size 3×3, the window size is then increased to the next higher odd integer and the window size becomes 5×5. In this way, the filter dynamically increases the window size as and when needed. After this when the filter considers some other noisy pixel, the initial window size is again 3×3. The non-impulse neighbors of the corrupted pixel in a window of size $W \times W$ are given by

$$\Omega^W_{x,y} = (j = j_1, j_2) | x - \frac{W-1}{2} \le j_1 \le x + \frac{W-1}{2}, \\ y - \frac{W-1}{2} \le j_2 \le y + \frac{W-1}{2} \tag{11}$$

where (x, y) are the coordinates of the corrupted pixel. Here, W can be any odd integer greater than or equal to 3. But while calculating the filtered value for a new corrupted pixel, its value is 3 and can be increased later as explained above. After finding the non-impulse neighbors above, the median-based lifting filter can then calculate the filtered median value $\text{med}^n_{x,y}$ as given below:

$$\text{med}^n_{x,y} = \text{med\_lift}(B^n_j | j \in \Omega^W_{x,y}) \tag{12}$$

where n is the current number of iteration of the image denoising algorithm and the med_lift is defined as the adaptive median-based lifting filter as described above.

The image lifting algorithm works by following the three steps of the lifting scheme split, predict, and update.
Input: Noisy image $B_{x,y}$ or $B^0_{x,y}$

1. Lifting (Split): If $B^n_{x,y} = 0$ or $B^n_{x,y} = 255$ then

$$X^n_{x,y} = B^n_{x,y} \tag{13}$$

2. (a) Predict: If $B^n_{x,y} = 0$ or $B^n_{x,y} = 255$, using (11) and (12), $\Omega^W_{x,y}$ and $med^n_{x,y}$ are calculated respectively.
   (b) Find Detail

$$d^{n+1}_{x,y} = |B^n_{x,y} - d^n_{x,y}| \tag{14}$$

   (c) Build significant and insignificant sets
   If $(d^n_{x,y} > T)$

$$sig^{n+1}_{x,y} = B^n_{x,y} \tag{15}$$

   else

$$insig^{n+1}_{x,y} = B^n_{x,y} \tag{16}$$

3. Update: For all the significant pixels update

$$B^n_{x,y} = \text{med}^n_{x,y}, \quad \text{if } B^{n+1}_{x,y} \in sig^{n+1}_{x,y} \tag{17}$$

4. Repeat steps 1 to 3 on the output of the previous run until satisfactory results are obtained.

Output: Restored Image B^{n+1}.

In the first step, we lift (split) all the pixels of the corrupted input image that are suspected to be noisy pixels. In the case of salt and pepper noise, this is done by separating those pixels whose grayscale image value is either 0 or 255 because these are the grayscale values that correspond to pepper and salt. Images corrupted by salt and pepper noise are contaminated with pixels which have either a very high value (255 in grayscale image) or a very low value (0 in a grayscale image). For the current iteration n of the algorithm, these pixels are stored in an array $X^{n+1}_{x,y}$. Now some of the pixels in the array $X^{n+1}_{x,y}$ can also be image data. To separate those pixels, we perform the predict step of our algorithm. In this step, first we find $\text{med}^n_{x,y}$ using the median-based lifting filter. Then, we calculate the detail d using Eq. (14). Then, the detail d of the pixel (x, y) are compared under consideration with threshold T to determine whether this pixel (x, y) is corrupted or not. As the data of images are smooth varying, the neighboring pixels of any pixel will not have values that are significantly different from its value. Hence, the filtered value found by the median-based lifting filter will also be close to the value of the pixel (x, y) if it is image data and not a noisy pixel. This is the reason, why upon comparing the detail with a suitably narrow threshold, we can successfully separate image data from noisy data. In our work, we put the noisy pixels and image pixels in iteration n in $sig^n_{x,y}$ and $insig^n_{x,y}$. Lastly, in the update step, we correct all the pixels in $sig^{n+1}_{x,y}$ with their filtered median value, where $(n+1)$ is the current number of iteration for which the algorithm has been executed. After executing the three steps above if the result is unsatisfactory, we execute the algorithm again on the output of the previous run.

4.2 Algorithm II : significant noise removal algorithm

The following steps are used to obtain a nested subset of significant noise pixels. Input: Noisy Image B

1. Let $B^{n+1} = B$: data set $(sig^k \cup insig^k)$
2. Use Algorithm I to find a set of new significant noise pixels (NP) to be filtered using the lifting filter.
3. Get $sig^k = sig^{k+1} - NP$.
4. Repeat the step 2 to 3 to get the desired image quality (sig^k).
5. Check the quality of image obtained from the data. The process is stopped, if a good image is generated.

Output: Denoised Image B^{n+1}.

5 Significance measures

5.1 PSNR

We have tested our proposed algorithm for different levels of noise ranging from as low as 5% to as high as 90%. The experimental results have been gauged using the mean square error (MSE) and peak signal-to-noise ratio (PSNR) measures that have been given below.

$$MSE = \frac{1}{mn} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} (A(x, y) - R(x, y))^2 \tag{18}$$

where A and R are the original and the restored images having a resolution of m*n.

$$PSNR = 10 \log_{10} \left(\frac{\max^2}{MSE} \right) \tag{19}$$

where max is the maximum possible pixel value of the image and its value is 255 in the case of a grayscale image.

5.2 SSIM

The SSIM index [19] is a full reference metric, in other words, the measuring of image quality based on an initial uncompressed or distortion-free image as reference. SSIM is designed to improve on traditional methods like peak signal-to-noise ratio and mean-squared error, which have proved to be inconsistent with human eye perception. SSIM is a new paradigm for quality assessment, based on the hypothesis that the HVS is highly adapted for extracting structural information. The measure of structural similarity compares local patterns of pixel intensities that have been normalized for luminance and contrast. In practice, a single overall index is sufficient enough to evaluate the overall image quality; hence, a mean SSIM (MSSIM) index is used as the quality measurement metric.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_1)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_1)} \quad (20)$$

$$MSSIM(x, y) = \frac{1}{M}\sum_{m=1}^{M} SSIM(x_m, y_m) \quad (21)$$

6 Results and discussion

The proposed algorithm using median-based lifting filter generates results that are superior to the existing methods for noise removal. The algorithm has been seen to yield better PSNR than given in [1,2] and [4–10]. First, we compare our results with Algorithm II of Salt-and-Pepper Noise Removal by Median-Type Noise Detectors and Detail-Preserving Regularization, which is defined as Edge Preservation Filter [EPF] of [1]. The comparative evaluation of our proposed method with [1] has been done in Table 1. Our proposed algorithm gives excellent PSNR values greater than 43 dB for noise ratios less than or equal to 10%. For noise ratios between 10% and 20%, the PSNR varies between 43.0821 and 39.0591 dB. Even for a noise ratio of 50%, we get a PSNR of 33.2258 dB with 2 iterations of the algorithm. All the results generated for noise ratios less than 50% have been generated by a single iteration of the algorithm. For noise ratios of 80% and 95%, we get PSNR values of 24.4037 and 23.8363 dB respectively. It took 3 iterations of the algorithm to get the results in the case of 80% noise and 4 iterations of the algorithm in the case of 95% noise. For implementing our algorithm, we have used MATLAB 7 on a 1.73-GHz Pentium *M* Processor with 256 MB of RAM. The algorithm has been found to be pretty fast. For restoring 512 × 512 Lena image corrupted with 95% noise, we needed 4 iterations of our algorithm and the execution took 96 s. The execution time

Table 1 Comparative evaluation of proposed method

Test image	Algorithm used	Noise density (%)	PSNR (in dB)
Lena 512 × 512	Proposed method	10	43.0821
		20	39.0591
		50	33.2258
		70	25.2000
	EPF [1]	10	35.0000
		20	32.5000
		50	27.5000
		70	24.6000

Table 2 Time complexity—comparison of CPU time (in seconds)

Test image	Noise density (%)	Adaptive MF	Edge PF [1]	Proposed
Lena 512 × 512	70	23	6865	53
	90	311	> 12000	91
Bridge 512 × 512	70	56	8003	54
	90	311	> 12000	92

Table 3 MSSIM value of proposed algorithm for 512 × 512 Lena image

Noise density (%)	MSSIM
10	0.9691
20	0.9528
50	0.9284
70	0.7339

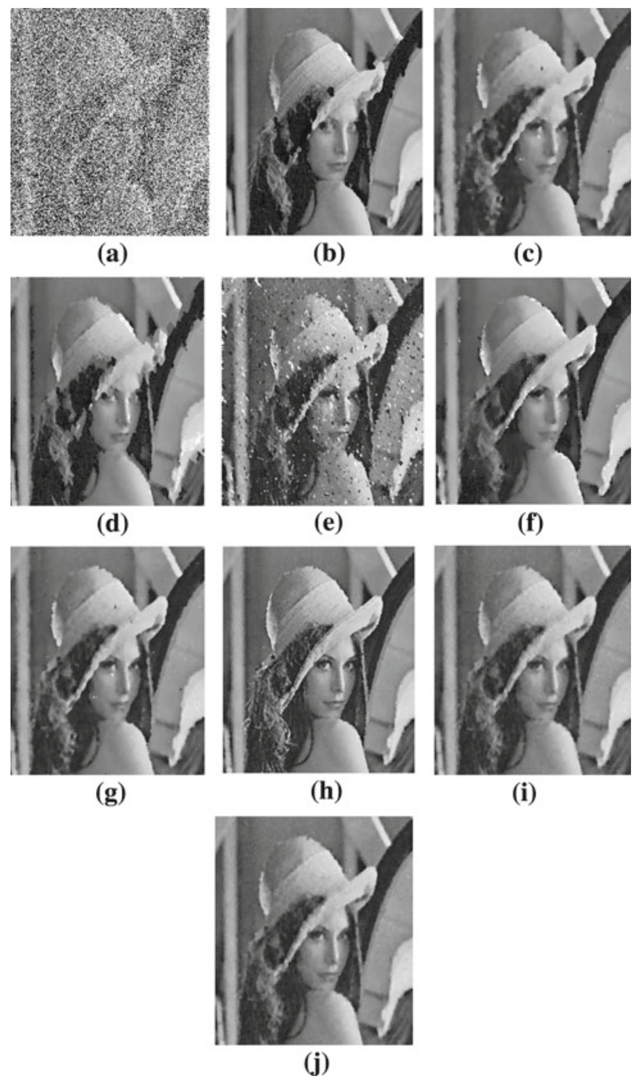

Fig. 4 De-noising results of various filters for Lena image, **a** Image with 70% Salt and Pepper Noise (6.7 dB). **b** MED filter (23.2 dB). **c** PSM filter (19.5 dB). **d** MSM filter (19.0 dB). **e** DDBSM filter (17.5 dB). **f** NASM Filter (21.8 dB). **g** ISM filter (23.4 dB). **h** Adaptive MED (25.8 dB). **i** Edge preservation filter [1] (24.6 dB). **j** Proposed algorithm (25.2 dB) (MSSIM = 0.7339)

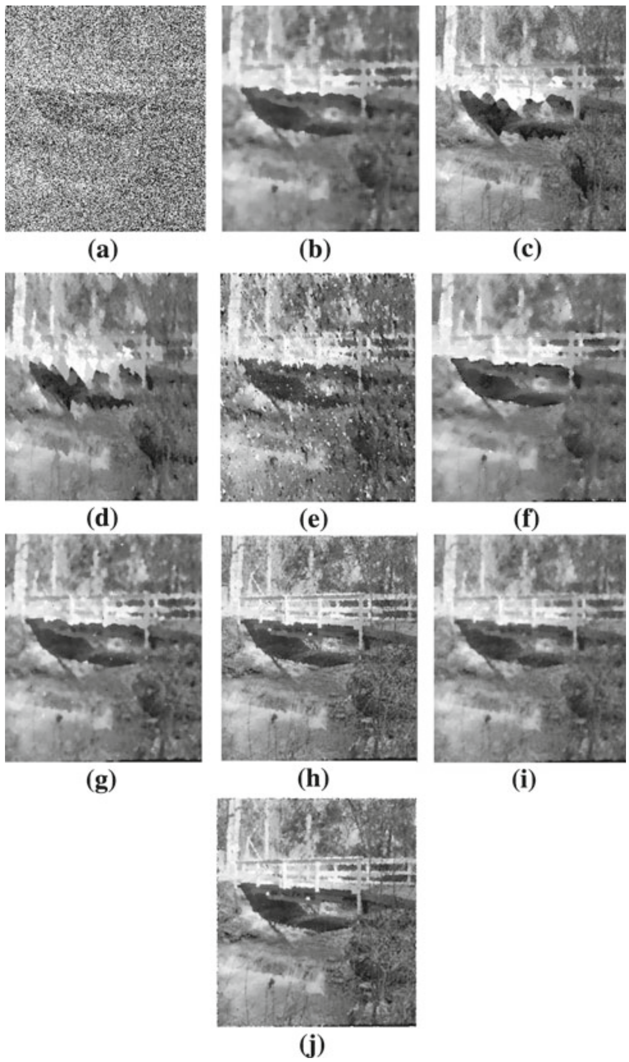

Fig. 5 De-noising results of various filters for bridge image, **a** Image with 70% salt and pepper noise (6.8 dB). **b** MED filter (19.8 dB). **c** PSM filter (17.0 dB). **d** MSM filter (16.4 dB). **e** DDBSM filter (15.9 dB). **f** NASM filter (19.9 dB). **g** ISM filter (20.1 dB). **h** Adaptive MED (21.8 dB). **i** Edge preservation filter [1] (21.1 dB). **j** Proposed algorithm (23.4 dB) (MSSIM = 0.7112)

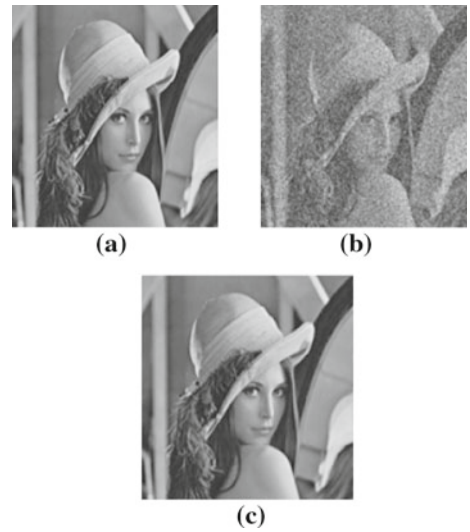

Fig. 6 Lena reconstruction results of [2] **a** Original Lena image. **b** Lena with 50% noise. **c** Reconstructed Lena PSNR = 31.8077 dB

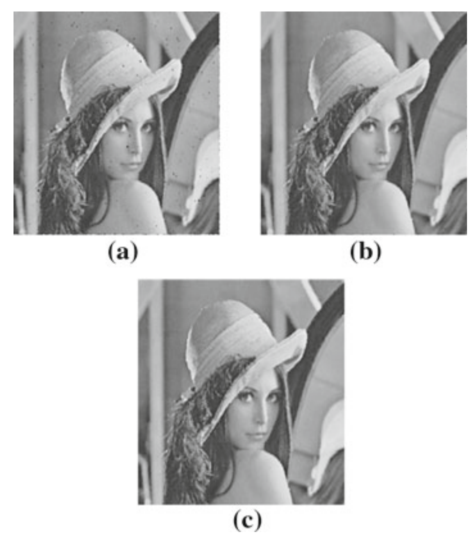

Fig. 7 Lena reconstruction results with PSNR stabilization compared with [2] **a** Proposed scheme 50% noise—run1 (PSNR = 31.5786 dB. **b** Proposed scheme—run2 (PSNR = 32.9453 dB). **c** Proposed scheme–run3 (PSNR = 33.2258 dB)

of our algorithm together with PSNR for different levels of noise has been shown in Table 2. The MSSIM of the proposed algorithm is shown in Table 3. Hence, the algorithm can be seen to be efficient both in terms of its dexterity to restore a corrupted image exceedingly well and to be less computation intensive.

Further, the performance of the proposed algorithm is tested for various levels of noise corruption and compared with standard filters, namely standard median (MED) filter, and also recently proposed filters like the progressive switching median (PSM) filter, the multistate median (MSM) filter, the noise adaptive soft-switching median (NASM) filter, the directional difference-based switching median (DDBSM) filter, and the improved switching median (ISM) filter. The results for 512 × 512 Lena image and 512 × 512 Bridge image for 70% salt and pepper noise are shown in Figs. 4 and 5, respectively. From these figures, it is clear that our proposed filter gives better image quality (PSNR) than the above filters (Figs. 6, 7). The variation of PSNR with noise level of salt and pepper noise for Lena image and Bridge image is depicted in Figs. 8 and 9, respectively. From these figures, it can be concluded that our proposed filter is superior than the other filters.

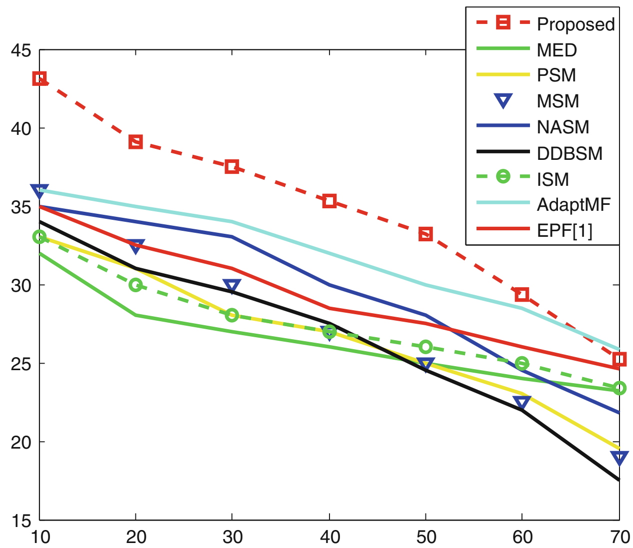

Fig. 8 PSNR versus noise level for Lena image

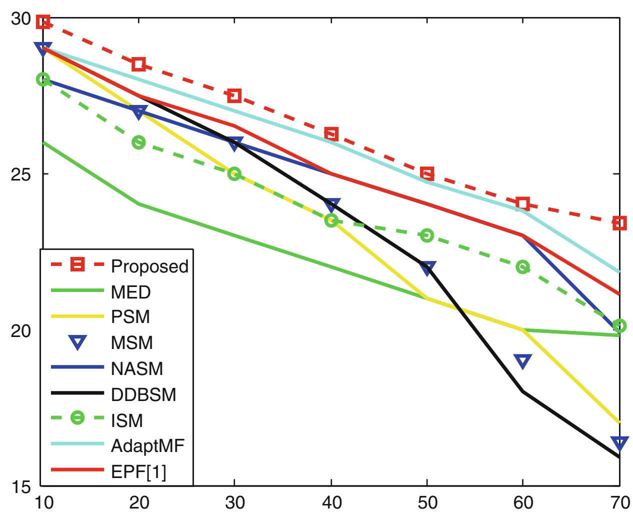

Fig. 9 PSNR versus noise level for Boat image

7 Conclusion

In this paper, a novel adaptive median-based lifting filter has been proposed for de-noising the salt and pepper noise. The lifting scheme of the second-generation wavelets with the adaptive median-based lifting filter is used. The computed results of the proposed filter is compared with various existing filters. The comparison of these results shows the superiority of the proposed filter in terms of image quality as well as the time complexity. The proposed filter is able to remove salt and pepper noise with higher levels of noise density.

References

1. Chan, R.H., Ho, C.W., Nikolova, M.: Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. IEEE Trans. Image Process. **14**(10), 1479–1485 (2005)
2. Jaiswal, T., Siddavatam, R.: Image noise cancellation by lifting filter using second generation wavelets. In: Proceedings of IEEE International Conference on Advances in Recent Technologies in Communication and Computing, pp. 667–671. Kerala, India, Oct. 27–28 (2009)
3. Siddavatam, R., Sandeep, K., Mittal, R.K.: A fast progressive image sampling using lifting scheme and non-uniform B-Splines. In: Proceedings of IEEE International Symposium on Industrial Electronics ISIE -07, pp. 1645–1650. Vigo, Spain, June 4–7 (2007)
4. Wang, Z., Zhang, D.: Progressive switching median filter for the removal of impulse noise from highly corrupted images. IEEE Trans. Circ. Syst.—II: Analog Digit. Signal Process. **46**(1), 78–80 (1999)
5. Sun, T., Neuvo, Y.: Detail-preserving median based filters in image processing. Pattern Recognit. Lett. **15**, 341–347 (1994)
6. Luo, W.: An efficient detail-preserving approach for removing impulse noise in images. IEEE Signal Process. Lett. **13**(7), 413–416 (2006)
7. Swelden, W.: The lifting scheme : a custom design construction of biorthogonal wavelets. Appl. Comput. Harmon. Anal. **3**(2), 186–200 (1996)
8. Pok, G., Liu, J.C., Nair, A.S.: Selective removal of impulse noise based on homogeneity level information. IEEE Trans. Image Process. **12**(1), 85–92 (2003)
9. Chen, T., Wu, H.R.: Application of partition-based median type filters for suppressing noise in images. IEEE Trans. Image Process. **10**(6), 829–836 (2001)
10. Chan, R.H., Hu, C., Nikolova, M.: An iterative procedure for removing random-valued impulse noise. IEEE Signal Process. Lett. **11**(12), 921–924 (2004)
11. Swelden, W.: The lifting scheme: a construction of second generation wavelets. SIAM J. Math. Anal. **29**(2), 511–546 (1998)
12. Lee, J.S.: Digital image enhancement and filtering by use of local statistics. IEEE Trans. Pattern Anal. Machine Intell. **PAMI-2**, 165–168 (1980)
13. Kuan, D.T., Sawchuk, A.A., Strand, T.C., Chavel, P.: Adaptive noise smoothing filter for images with signal-dependent noise. IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-7**(2), 165–177 (1985)
14. Bernstein, R.: Adaptive nonlinear filters for simultaneous removal of different kinds of noise in images. IEEE Trans. Circ. Syst. **CAS-34**(11), 1275–1291 (1987)
15. Sun, X.Z., Venetsanopoulos, A.N.: Adaptive schemes for noise filtering and edge detection by use of local statistics. IEEE Trans. Circ. Syst. **CAS-35**(1), 57–69 (1988)
16. Sweldens, W.: The Lifting Scheme: A New Philosophy in Biorthogonal Wavelet Constructions. Proceedings of SPIE, **2569**, 68–79 (1995)
17. Daubechies, I., Sweldens, W.: Factoring wavelet transforms into lifting schemes. J. Fourier Anal. Appl. **4**, 247–269 (1998)
18. Acharya, T., Tsai, P.S.: JPEG 2000 Standard for Image Compression: Concepts, Algorithms and VLSI Architectures. Wiley, London (2005)
19. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)