

A Novel Fuzzy Filter for Random Impulse Noise Removal in a Color Video

P. Vani

M.Tech in Wireless and
Mobile communication,
Anurag Engineering College,
Ananthagiri, Kodad,
Nalgonda (Dist.),
Telangana, India.

V. Santhosh Kumar

Assistant Professor,
Dept of ECE,
Anurag Engineering College,
Ananthagiri, Kodad,
Nalgonda (Dist.),
Telangana, India.

P. Rama Krishna

Assistant Professor,
Dept of ECE,
Dhanekula Institute of Engineering
& Technology, Ganguru,
Vijayawada, India.

Abstract :

Now a day's digital image processing applications are widely used in various fields such as medical, military, satellite, remote sensing and even web applications also. In any application denoising of image/video is a challenging task because noise removal will increase the digital quality of an image or video and will improve the perceptual visual quality. In spite of the great success of many denoising algorithms, they tend to smooth the fine scale image textures when removing noise, degrading the image visual quality. To address this problem, a new fuzzy filter for the removal of random impulse noise in color video is presented. By working with different successive filtering steps, a very good tradeoff between detail preservation and noise removal is obtained. The experiments show that the proposed method outperforms other state-of-the-art filters both visually and in terms of objective quality measures such as the mean absolute error (MAE) and the peak-signal-to-noise ratio (PSNR).

I.INTRODUCTION:

Images and videos captured from both digital cameras and conventional film cameras will be affected with the noise from a variety of sources. These noise elements will create some serious issues for further processing of images in practical applications such as computer vision, artistic work or marketing and also in many fields. There are many types of noises like salt and pepper, Gaussian, speckle and passion. In salt and pepper noise (sparse light and dark disturbances), pixels in the captured image are very different in intensity from their neighbouring pixels; the defining characteristic is that the intensity value of a noisy picture element bears no relation to the color of neighbouring pixels.

Generally this type of noise will only affect a small number of pixels in an image. When we viewed an image which is affected with salt and pepper noise, the image contains black and white dots, hence it is termed as salt and pepper noise. In Gaussian noise, noisy pixel value will be a small change of original value of a pixel. A histogram, a discrete plot of the amount of the distortion of intensity values against the frequency with which it occurs, it shows a normal distribution of noise. While other distributions are possible, the Gaussian (normal) distribution is usually a good model, due to the central limit theorem that says that the sum of different noises tends to approach a Gaussian distribution. In selecting a noise reduction algorithm, one must consider several factors:

- A digital camera must apply noise reduction in a fraction of a second using a tiny on-board CPU, while a desktop computer has much more power and time
- whether sacrificing some real detail information is acceptable if it allows more distortion or noise to be removed (how aggressively to decide whether the random variations in the image are noisy or not)

In real-world photographs, maximum variations in brightness ("luminance detail") will be consisted by the highest spatial frequency, rather than the random variations in hue ("chroma detail"). Since most of noise reducing techniques should attempt to remove noise without destroying of real detail from the captured photograph. In addition, most people find luminance noise in images less objectionable than chroma noise; the colored blobs are considered "digital-looking" and artificial, compared to the mealy appearance of luminance noise that some compare to film grain.

For these two reasons, most of digital image noise reduction algorithms split the image content into chroma and luminance components. One solution to eliminate noise is by convolving the original image with a mask that represents a low-pass filter or smoothing operation. For example, the Gaussian mask incorporates the elements determined by a Gaussian function. This operation brings the value of each pixel into closer harmony with the values of its neighbours. In general, a smoothing filter sets each pixel to the mean value, or a weighted mean, of itself and its nearby neighbours; the Gaussian filter is just one possible set of weights. However, spatial filtering approaches like mean filtering or average filtering, Savitzky filtering, Median filtering, bilateral filter and Wiener filters had been suffered with losing edges information.

All the filters that have been mentioned above were good at denoise of images but they will provide only low frequency content of an image it doesn't preserve the high frequency information. In order to overcome this issue Non Local mean approach has been introduced. More recently, noise reduction techniques based on the "NON-LOCAL MEANS (NLM) had developed to improve the performance of denoising mechanism [1][4][5][9]. It is a data-driven diffusion mechanism that was introduced by Buades et al. in [1]. It has been proved that it's a simple and powerful method for digital image denoising. In this, a given pixel is denoised using a weighted average of other pixels in the (noisy) image. In particular, given a noisy image n_i , and the denoised image \tilde{d}_i at pixel i is computed by using the formula.

$$\tilde{d}_i = \frac{\sum_j w_{ij} n_j}{\sum_j w_{ij}} \quad (1)$$

Where w_{ij} is some weight assigned to pixel i and j . The sum in (1) is ideally performed to whole image to denoise the noisy image. NLM at large noise levels will not give accurate results because the computation of weights of pixels will be different for some neighbourhood pixels which looks like same. Most of the standard algorithm used to denoise the noisy image and perform the individual filtering process. Denoise generally reduce the noise level but the image is either blurred or over smoothed due to losses like edges or lines. In the recent years there has been a fair amount of research on center pixel weight (CPW) for image denoising [3], because CPW provides an appropriate basis for separating noisy signal from the image signal.

Optimized CPW is good at energy compaction, the small coefficient are more likely due to noise and large coefficient due to important signal feature [8]. These small coefficients can be thresholded without affecting the significant features of the image. However, all the above mentioned techniques were not suitable for texture enhanced image denoising and will not preserve the fine details of image. In order to overcome the existing systems drawbacks, here in this we present a filter for the removal of random impulse noise in color image sequences, in which each of the color components is filtered separately based on fuzzy rules, in which information from the other color bands is integrated. To preserve the details as much as possible, the noise is removed by three successive filtering steps.

Only pixels that have been detected to be noisy are filtered. This filtering is done by block matching, a technique used for video compression that has already been adopted in video filters for the removal of Gaussian noise (e.g., [4]–[6]), but that has not really found its way to impulse noise filters yet. The correspondence between blocks is usually calculated by a mean absolute difference (MAD) that is heavily subject to noisy impulses. Therefore, we introduce a MAD measure that is adaptive to detected noisy pixels components. To benefit as much as possible from the spatial and temporal information available in the sequence, the search region for corresponding blocks contains pixel blocks both from the previous and current frame. The experiments show that the proposed method outperforms other state-of-the-art filters both visually and in terms of objective quality measures such as the MAE, PSNR and NCD.

The rest of this thesis has been organized as: Section II existing techniques such as Savitzky-golay, median, bilateral, wavelet filters, and NLM; Section III discusses the proposed fuzzy filter with successive steps; Section IV shows experimental comparisons for various techniques with the new solution; and Section V concludes the thesis.

II. CONVENTIONAL FILTERS:

In this section we discussed various spatial filters and their performance when a noisy input will be given to them. Here in this section we had explained about each filter in detail. Firstly, Savitzky-Golay (SG) filter:

it is a simplified method and uses least squares technique for calculating differentiation and smoothing of data. Its computational speed will be improved when compared least-squares techniques. The major drawback of this filter is: Some of first and last data point cannot smoothen out by the original Savitzky-Golay method. Assuming that, filter length or frame size (in S-G filter number of data sample read into the state vector at a time) N is odd, $N=2M+1$ and $N= d+1$, where d = polynomial order or polynomial degree. Second, Median filter: This is a nonlinear digital spatial filtering technique, often used to removal of noise from digital images. Median filtering has been widely used in most of the digital image processing applications.

The main idea of the median filter is to run through the image entry by pixel, replacing each pixel with the median value of neighboring pixels. The pattern of neighbors is called the “window”, which slides, pixel by pixel, over the entire image. Third, Bilateral filter: The bilateral filter is a nonlinear filter which does the spatial averaging without smoothing edges information. Because of this feature it has been shown that it’s an effective image denoising algorithm. Bilateral filter is presented by Tomasi and Manduchi in 1998. The concept of the bilateral filter was also presented in [8] as the SUSAN filter and in [3] as the neighborhood filter. It is mentionable that the Beltrami flow algorithm is considered as the theoretical origin of the bilateral filter [4] [5] [6], which produce a spectrum of image enhancing algorithms ranging from the linear diffusion to the nonlinear flows. The bilateral filter takes a weighted sum of the pixels in a local neighborhood; the weights depend on both the spatial distance and the intensity length.

In this way, edges are preserved well while noise is eliminated out. Next, Wavelet filtering: Signal denoising using the DWT consists of the three successive procedures, namely, signal decomposition, thresholding of the DWT coefficients, and signal reconstruction. Firstly, we carry out the wavelet analysis of a noisy signal up to a chosen level N . Secondly, we perform thresholding of the detail coefficients from level 1 to N . Lastly, we synthesize the signal using the altered detail coefficients from level 1 to N and approximation coefficients of level N . However, it is generally impossible to remove all the noise without corrupting the signal. As for thresholding, we can settle either a level-dependent threshold vector of length N or a global threshold of a constant value for all levels.

Classical Non Local Means: It is a data-driven diffusion mechanism that was introduced by Buades et al. in [1]. It has been proved that it’s a simple and powerful method for digital image denoising. In this, a given pixel is denoised using a weighted average of other pixels in the (noisy) image. In particular, given a noisy image n_i , and the denoised image $\hat{d}=(d_i)$ at pixel i is computed by using the formula.

$$\hat{d}_i = \frac{\sum_j w_{ij} n_j}{\sum_j w_{ij}} \quad (1)$$

Where w_{ij} is some weight assigned to pixel i and j . The sum in (1) is ideally performed to whole image to denoise the noisy image. NLM at large noise levels will not give accurate results because the computation of weights of pixels will be different for some neighbourhood pixels which looks like same.

$$w_{i,j} = \exp\left(\sum_{k \in P} G_{\beta} \left(\frac{(n_{i+k} - n_{j+k})^2}{2h}\right)\right) \quad (2)$$

In this each weight is computed by similarity quantification between two local patches around noisy pixels n_i and n_j as shown in eq. (2). Here, G_{β} is a Gaussian weakly smooth kernel [1] and P denotes the local patch, typically a square centered at the pixel and h is a temperature parameter controlling the behavior of the weight function. Another popular approach to image denoising is the variational method, where energy functional is minimized to search the desired estimation of x from its noisy observation y . The energy functional usually involves two terms: a data fidelity term which depends on the image degeneration process and a regularization term which models the prior of clean natural images [13], [16] and [17]. The statistical modeling of natural image priors is crucial to the success of image denoising. Motivated by the fact that natural image gradients and wavelet transform coefficients have a heavy-tailed distribution, sparsity priors are widely used in image denoising [10]–[12].

The well-known total variation minimization methods actually assume Laplacian distribution of image gradients [13]. The sparse Laplacian distribution is also used to model the high pass filter responses and wavelet/curvelet transforms coefficients [14], [15]. By representing image patches as a sparse linear combination of the atoms in an over-complete redundant dictionary, which can be analytically designed or learned from natural images, sparse coding has proved to be very effective in image denoising via l_0 -norm or l_1 -norm minimization [16], [17].

Another popular prior is the nonlocal self-similarity (NSS) prior that is, in natural images there are often many similar patches (i.e., nonlocal neighbors) to a given patch, which may be spatially far from it. The connection between NSS and the sparsity prior is discussed in [18], [19]. The joint use of sparsity prior and NSS prior has led to state-of-the-art image denoising results [19]–[14]. In spite of the great success of many denoising algorithms, however, they often fail to preserve the image fine scale texture structures, degrading much the image visual quality (please refer to Fig. 1 for example). With the rapid development of digital imaging technology, now the acquired images can contain tens of megapixels.

On one hand, more fine scale texture features of the scene will be captured; on the other hand, the captured high definition image is more prone to noise because the smaller size of each pixel makes the exposure less sufficient. Unfortunately, suppressing noise and preserving textures are difficult to achieve simultaneously, and this has been one of the most challenging problems in natural image denoising. Unlike large scale edges, the fine scale textures are much more complex and are hard to characterize by using a sparse model. Texture regions in an image are homogeneous and are composed of similar local patterns, which can be characterized by using local descriptors or textons. Cognitive studies have revealed that the first-order statistics, e.g., histograms, are the most significant descriptors for texture discrimination.

III. PROPOSED FRAME WORK:

The filtering framework presented in this paper is intended for color video corrupted by random impulse noise.

If we respectively denote the original (noise-free) sequence by I_o , the t_{th} frame of that sequence by $I_o(t)$ and the red, green and blue component of the color $I_o(x, y, t)$ of the pixel at the x_{th} row and y_{th} column in that frame by $I_o^R(x, y, t)$, $I_o^G(x, y, t)$ and $I_o^B(x, y, t)$ (i.e., $I_o(x, y, t) = (I_o^R(x, y, t), I_o^G(x, y, t), I_o^B(x, y, t))$), then the noisy sequence $I_n(t)$ is determined as follows [1], [2]:

$$I_n^c(x, y, t) = \begin{cases} I_o^c(x, y, t), & \text{with } 1 - p \\ \eta^c(x, y, t), & \text{with } p \end{cases}$$

Where $c \in \{R, G, B\}$ and $p \in [0, 1]$ denotes the probability that a pixel component value is corrupted and replaced by a identically distributed independent random noise value $\eta^c(x, y, t)$ coming from a uniform distributed on the interval of possible color component values. For the color videos used in the

experiments of this paper, 8 bits are used for the storage of the color component values and we work with a uniform distribution on the interval $[0, 255]$. Further, the probability that a given color component value is corrupted is independent on whether the neighboring values or the values in the other color components are corrupted or not. The proposed filtering framework consists of three successive filtering steps as depicted in Fig. 1. By removing the noise step by step, the details can be preserved as much as possible. Indeed, if a considerable part of the noise has already been removed in a previous step, and more noise-free neighbors to compare to are available, it will be easier to distinguish noise from small details.

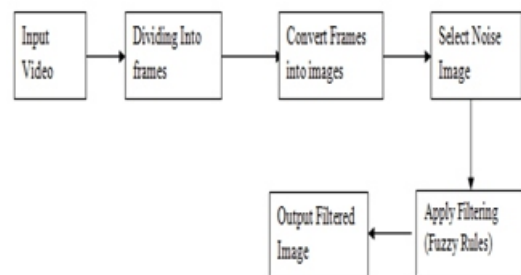


Fig1. Block diagram of proposed fuzzy filter for images

In the first step (with output denoted by $I_{(f_1)}$), we calculate for each pixel component a degree to which it is considered noise-free and a degree to which it is considered noisy. If the noisy degree is larger than the noise-free degree, the pixel component is filtered, otherwise it remains unchanged. The determination of both degrees is mainly based on temporal information (comparison to the corresponding pixel component in the previous frame).

Note, however, that only in non-moving areas can large temporal differences be assigned to noise. In areas where there is motion, such differences might also be caused by that motion. As a consequence, and as can be seen in Fig. 2, impulses in moving areas will not always be detected in this step. They can, however, be detected in the second step (outputl_f_2). Analogously as to the first step, again a noise-free degree and a noisy degree are calculated. However, the detection is now mainly based on color information. A pixel component can be seen as noisy if there is no similarity to its (spatio-temporal) neighbors in the given color, while there is in the other color bands.

The third step (outputl_f), finally, removes the remaining noise and refines the result by using as well temporal as spatial and color information. For example, homogeneous areas can be refined by removing small impulses that are relatively large in that region, but are not large enough to be detected in detailed regions and that thus have not been detected yet by the previous general detection steps. The results of the different successive filtering steps are illustrated for the 20th frame of the “Salesman” sequence.

First Filtering Step Detection:

In this detection step, we calculate for each of the components of each pixel a degree to which it is considered noise-free and a degree to which it is thought to be noisy. A component for which the noisy degree is larger than the noise free degree, i.e., that is more likely to be noisy than noise-free, will be filtered. Other pixel components will remain unchanged.

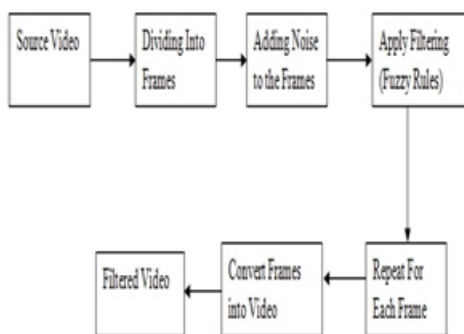


Fig2. Block diagram of proposed work for videos

The noise-free degree and the noisy degree are determined by fuzzy rules as follows.

We consider a pixel component to be noise-free if it is similar to the corresponding component of the pixel at the same spatial location in the previous or next frame and to the corresponding component of two neighboring pixels in the same frame. In the case of motion, the pixels in the previous frames cannot be used to determine whether a pixel component in the current frame is noise-free. Therefore, more confirmation (more similar neighbors or also similar in the other color components) is wanted instead. For the noise-free degree of the red component (and analogously for the other components), this is achieved by the following fuzzy rule.

Fuzzy Rule 1:

$$\begin{aligned}
 &IF \left(\left(|I_n^R(x, y, t) - I_f^R(x, y, t)| \text{ is NOT LARGE POSITIVE OR } |I_n^R(x, y, t) - I_n^R(x, y, t + 1)| \text{ is NOT LARGE POSITIVE} \right) \text{ AND, there are two neighbors } (x + k, y + l, t) \left(-2 \leq k, l \leq 2 \text{ and } (k, l) \neq (0, 0) \right) \text{ for which } |I_n^R(x, y, t) - I_n^R(x + k, y + l, t)| \text{ is NOT LARGE POSITIVE} \right)
 \end{aligned}$$

To represent the linguistic value large positive in the above rule, a fuzzy set is used, with a membership function as depicted in Fig. 3 (see Section III-A for the determination of the parameters). For the conjunctions (AND), disjunctions (OR) and negations (NOT) in fuzzy logic, triangular norms, triangular conorms and involutive negators [26] are used. In this paper, we will use the minimum operator, the maximum operator and the standard negator (N_s(x))=1-x, x [0,1] respectively. Those operators are simple in use and yielded the best results, but the difference compared to the results for another choice of operators is neglectable. The outcome of the rule, i.e., the degree to which the red component of the pixel at position (x,y,t) is considered noise-free, is determined as the degree to which the antecedent in the fuzzy rule is true:

$$\begin{aligned}
 &\mu_{noisyfree}^R(x, y, t) \\
 &= \max \left(\min \left(\max(\alpha_1(x, y, t), \alpha_2(x, y, t)), M_2(x, y, t) \right), \max(M_4(x, y, t), M_{2b}(x, y, t)) \right)
 \end{aligned}$$

Where

$$\begin{aligned}
 \alpha_1(x, y, t) &= \left(1 - \mu_{LP}(|I_n^R(x, y, t) - I_f^R(x, y, t - 1)|) \right)
 \end{aligned}$$

$$\alpha_2(x, y, t) = (1 - \mu_{LP}(|I_n^R(x, y, t) - I_n^R(x, y, t + 1)|))$$

and where $M_2(x, y, t)$ and $M_4(x, y, t)$ respectively denote the degree to which there are two (respectively four) neighbors for which the absolute difference in the red component value is not large positive, that is determined as the second largest element in the set

$$\{1 - \mu_{LP}(|I_n^R(x, y, t) - I_n^R(x + k, y + l, t)|) \parallel -2 \leq k, l \leq 2 \text{ and } (k, l) \neq (0, 0)\}$$

And $M_{2b}(x, y, t)$ denotes the degree to which there are two neighbors for which the absolute differences in the red component and one of the two color components are not large positive, determined as the second largest element in the set

$$\{ \min(1 - \mu_{LP}(|I_n^R(x, y, t) - I_n^R(x + k, y + l, t)|), \max(1 - \mu_{LP}(|I_n^G(x, y, t) - I_n^G(x + k, y + l, t)|), 1 - \mu_{LP}(|I_n^B(x, y, t) - I_n^B(x + k, y + l, t)|)) \parallel -2 \leq k, l \leq 2 \text{ and } (k, l) \neq (0, 0)\}$$

Analogously, a degree to which the component of a pixel is considered noisy is calculated. In this step, we consider a pixel component to be noisy if the absolute difference in that component is large positive compared to the pixel at the same spatial location in the previous frame and if not for five of its neighbors the absolute difference in this component and one of the other two color bands is large positive compared to the pixel at the same spatial location in the previous frame (which means that the difference is not caused by motion). Further, we also want a confirmation either by the fact that in this color band, there is a direction in which the differences between the considered pixel and the two respective neighbors in this direction are both large positive or large negative and if the absolute

difference between those two neighbors is not large positive (i.e., there is an impulse between two pixels that are expected to belong to the same object) or by the fact that there is no large difference between the considered pixel and the pixel at the same spatial location in the previous frame in one of the other two color bands. For the red component (and analogously the other components) this leads to the following fuzzy rule. In this subsection, we discuss the filtering for the red color band. The filtering of the other color bands is analogous. We decide to filter all red pixel components that are considered more likely to be noisy than noise-free, i.e., for which $\mu_{noisy}^R(x, y, t) > \mu_{noise-free}^R(x, y, t)$. The red components of the other pixels remain unchanged to avoid the filtering of noise-free pixels (that might have been incorrectly assigned a low noisy degree, but for which the high noise-free degree assures us that it is noise-free) and thus detail loss. On the other hand, noisy pixel components might remain unfiltered due to an incorrect high noise-free degree, but those pixels can still be detected in the next filtering step.

Second Filtering Step :

In our aim to preserve the details as much as possible, the noise is removed in successive steps. In this step, the noise is detected based on the output of the previous step (I_{f-1}). Also in this second filtering step, a degree to which a pixel component is expected to be noise-free and a degree to which a pixel component is expected to be noisy, is calculated. In the calculation of those degrees, we now take into account information from the other color bands. A color component of a pixel is considered noise-free if the difference between that pixel and the corresponding pixel in the previous frame is not large in the given component and also not large in one of the other two color components. It is also considered noise-free if there are two neighbors for which the difference in the given component and one of the other two components are not large. So, the other color bands are used here as a confirmation for the observations in the considered color band to make those more reliable.

Third Filtering Step :

The result from the previous steps is further refined based on temporal, spatial and color information. Namely, the red component (and analogously the green and blue component) of a pixel is refined in the following cases:

In non-moving areas, pixels will correspond to the pixels in the previous frame, which allows us to detect remaining isolated noisy pixels. If (x,y,t) lies in a non-moving 3×3 neighborhood, i.e., (with $\Delta(x,y,t)=|I_{f-2}(x,y,t)-I_{f-1}(x,y,t)|$)

EXPERIMENTAL RESULTS:

The experimental results have been done in MATLAB 2011a version and tested with different color image sequences and color videos also. The proposed work has been applied to a color video 'salesman.avi', and observed the denoising results in following figures. Original frame from a color video has shown in fig.1 and the noisy image which is corrupted by random impulse noise is shown in fig.2. And the denoised images after successive filtering steps have been shown in fig.3 (a) first filtering output (b) second stage output and fig.4 shows final filtering step output. Also calculated the Peak Signal to Noise Ratio (PSNR) and Mean Absolute Error (MAE) for the comparison of different filtering steps in terms of the visual quality of denoised images



Fig.1 Original frame of input color video



Fig.2 Noisy image



(a)



(b)

Fig3. (a) Output of fuzzy filter after first filtering step (b) output of fuzzy filter after second filtering step



Fig4. Output of final filtering step

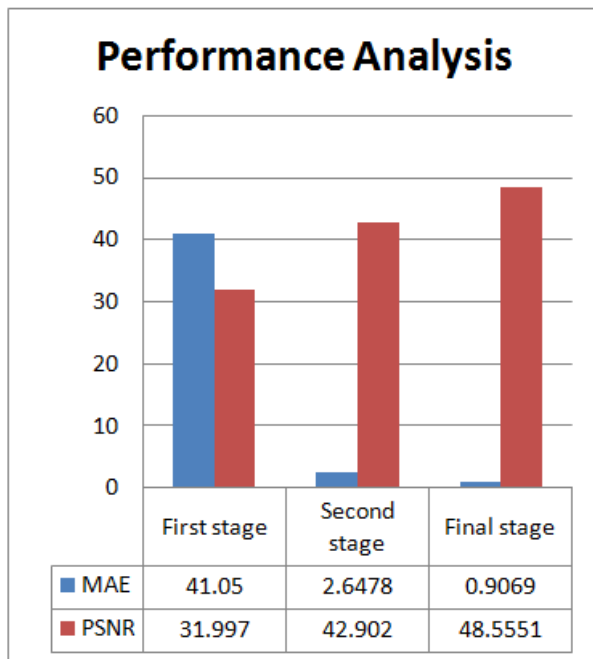


Fig5. Performance analysis of proposed filtering work

V.CONCLUSIONS:

Here in this letter, we had presented a novel fuzzy filtering framework for color videos, which has been corrupted by random impulse noise. To improve the efficiency, we followed a step by step process. Noised video has been denoised in successive filtering steps using proposed fuzzy rules. The experiments showed that the proposed method outperforms other state-of-the-art methods both in terms of objective measures such as MAE and PSNR and visually.

REFERENCES:

1. Buades, A, Coll, B, Morel J.M, "A non-local algorithm for image denoising", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 20-26 June 2005, San Diego, CA, USA.
2. C. Deledalle, V. Duval, and J. Salmon, "Non-local methods with shape adaptive patches (nlm-sap)," J. Math. Imag. Vis., vol. 43, no. 2, pp. 103-120, 2012.
3. V. Duval, J. Aujol, and Y. Gousseau, "A bias-variance approach for the nonlocal means," SIAM J. Imag. Sci., vol. 4, no. 2, pp. 760-788, 2011.
4. D. Van De Ville and M. Kocher, "Sure-based non-local means," IEEE Signal Process. Lett., vol. 16, no. 11, pp. 973-976, Nov. 2009.

5. J. Salmon, "On two parameters for denoising with non-local means," IEEE Signal Process. Lett., vol. 17, no. 3, pp. 269-272, Mar. 2010.
6. J. Darbon, A. Cunha, T. Chan, S. Osher, and G. Jensen, "Fast nonlocal filtering applied to electron cryomicroscopy," in IEEE Int. Symp. Biomedical Imaging: From Nano to Macro, 2008, pp. 1331-1334.
7. R. Lai and Y. Yang, "Accelerating non-local means algorithm with random projection," Electron. Lett., vol. 47, no. 3, pp. 182-183, 2011,3.
8. K. Chaudhury, "Acceleration of the shiftable $o(1)$ algorithm for bilateral filtering and non-local means," IEEE Trans. Image Process., vol. PP, no. 99, 2012, p. 1
9. G. Oppenheim J. M. Poggi M. Misiti, Y. Misiti. Wavelet Toolbox. The MathWorks, Inc., Natick, Massachusetts 01760, April 2001
10. R. Fergus, B. Singh, A. Hertzmann, S. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," in Proc. ACM SIGGRAPH, pp. 787-794, 2006.
11. A. Levin, R. Fergus, F. Durand, and W. T. Freeman, "Image and depth from a conventional camera with a coded aperture," in Proc. ACM SIGGRAPH, 2007.
12. D. Krishnan, R. Fergus, "Fast image deconvolution using hyperLaplacian priors," in Proc. Neural Inf. Process. Syst., pp. 1033-1041, 2009.
13. L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," Physica D, vol. 60, no. 1-4, pp. 259-268, Nov. 1992.
14. M. Wainwright and S. Simoncelli, "Scale mixtures of gaussians and the statistics of natural images," in Proc. Neural Inf. Process. Syst., vol. 12, pp. 855-861, 1999.
15. J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using a scale mixture of Gaussians in the wavelet domain," IEEE Trans. Image Process., vol. 12, no. 11, pp. 1338-1351, Nov. 2003.
16. M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," IEEE Trans. Image Process., vol. 15, no. 12, pp. 3736-3745, Dec. 2006.
17. W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super resolution by adaptive sparse domain selection and adaptive regularization," IEEE Trans. Image Process., vol. 20, no. 7, pp. 1838-1857, Jul. 2011.
18. A. Buades, B. Coll, and J. Morel, "A review of image denoising methods, with a new one," Multiscale Model. Simul., vol. 4, no. 2, pp. 490-530, 2005.
19. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," IEEE Trans. Image Process., vol. 16, no. 8, pp. 2080-2095, Aug. 2007.