IMAGE DENOISING USING NON-LOCAL FUZZY MEANS

Rushi Lan, Yicong Zhou, Yuan Yan Tang, C. L. Philip Chen*

Department of Computer and Information Science
University of Macau, Macau 999078, China

ABSTRACT

Due to the fact that the dissimilarity between the centered and other patches within an image dynamically changes in each denoising iteration, this paper proposes a new non-local image denoising algorithm called the non-local fuzzy means (NLFM). It considers the weights as fuzzy variables and adaptively update their values by solving an energy minimization problem. A new exponential parameter is introduced to the weights, offering a nonlinear mapping to enhance the NLFM's denoising performance. Two strategies are proposed to solve the energy minimization problem in different parameter settings. Experiments demonstrate that the NLFM outperforms several existing non-local means algorithms in terms of visual quality and quantitative measures.

Index Terms— image denoising, non-local means (NLM), non-local fuzzy means (NLFM)

1. INTRODUCTION

Image denoising is a classical research field in image processing and computer vision due to the fact that the image is usually corrupted in some processing stages, such as image acquisition, quantization and transmission. The quality of corrupted images should be improved before further processing, like image segmentation, recognition, and registration, etc.

Recent years, a large number of algorithms have been developed to reduce the Gaussian noise with zero mean from different perspectives. Among those denoising methods, non-local means (NLM) [1, 2], as a classical spatial domain method, has brought significant attentions of researchers. NLM denoises the noisy image by calculating the weighted average values of the neighboring pixels in a searching window S_i centered at the pixel i, which can be expressed by:

$$\hat{X}(i) = \frac{\sum_{j \in S_i} \omega_{ij} Y(j)}{\sum_{j \in S_i} \omega_{ij}}, \tag{1}$$

where $Y(j)$ is the value at location j in noisy image Y, and $\hat{X}(i)$ is the restoration result. The weights ω_{ij} describe the dissimilarity between two patches as following

$$\omega_{ij} = \exp\left(-\frac{\|Y_i - Y_j\|^2}{h^2}\right) \tag{2}$$

where $\|\cdot\|$ is the Euclidean norm, Y_i and Y_j represent the image patches in Y, and h is a specified parameter.

Extensive investigations have proved that the non-local processing is suitable for image denoising. Examples include K-SVD [3], BM3D [4], LPG-PCA [5], PLOW [6], HOSVD [7], and PEWA [8], etc. Therefore, many improved versions of NLM have been proposed to enhance its denoising performance using different principles, such as computation acceleration [9, 10], dissimilarity measure [11], weight selection [12–15], and many others.

Studies have shown that the weights ω_{ij} play a significant role in the NLM framework [12–14]. However, in NLM and its improvements, the weights are often set to be constants during the iterative denosing processes. This setting is inappropriate because the corrupted images become cleaner and image contents change dramatically after each new denoising iteration.

In order to adaptively determine the weights, this paper proposes a novel NLM algorithm, called the non-local fuzzy means (NLFM). Our main contributions can be listed as follows:

1. Unlike existing NLM methods which use fixed weights, the proposed NLFM considers the weights as fuzzy variables, and an energy function is established to find their optimal values.
2. We introduce a new exponential parameter m to the weights ω_{ij} in order to nonlinearly stretch their values to a proper space to enhance the NLFM's denoising performance.
3. Under different ranges of parameter m, we propose two strategies to solve the energy minimization problem.

The remainder of the paper is organized as follows. In Section 2 we introduce the proposed NLFM algorithm in detail. In Section 3 we describe the experimental results. Section 4 offers our conclusions.

This work was supported in part by the Macau Science and Technology Development Fund under Grant FDCT/106/2013/A3 and by the Research Committee at University of Macau under Grants MYRG2014-00003-FST, MRG017/ZYC/2014/FST, MYRG113(Y1-L3)-FST12-ZYC and MRG001/ZYC/2013/FST.

*Corresponding author. Tel.: (853) 88228458; Fax: (853) 88222426, Email address: yicongzhou@umac.mo.

2. NON-LOCAL FUZZY MEANS

2.1. Optimization Models

The NLM can be transformed into an optimization problem [12]. This is another understanding of NLM. We further extend this concept to other classical image denoising methods including the mean filtering and Gaussian filtering. They can also be transformed into the corresponding optimization models from the mathematic point of view. The results are given in Table I.

Table 1: The optimization models of denoising methods.

Method	Optimization Model
Mean Filtering	$\hat{X}(i) = \text{argmin}_{X(i)} \sum_{j\in S_i}(Y(j) - X(i))^2$
Gaussian Filtering	$\hat{X}(i) = \text{argmin}_{X(i)} \sum_{j\in S_i} \omega_{ij}(Y(j) - X(i))^2$
NLM [12]	$\hat{X}_i = \text{argmin}_{X_i} \sum_{j\in S_i} \omega_{ij}\|Y_j - X_i\|^2$
NLEM [11]	$\hat{X}_i = \text{argmin}_{X_i} \sum_{j\in S_i} \omega_{ij}\|Y_j - X_i\|$
INLEM [12]	$\hat{X}_i = \text{argmin}_{X_i} \sum_{j\in S_i} \sqrt{\omega_{ij}}\|Y_j - X_i\|$

From Table I, we can observe the development of image denoising techniques. The mean filtering processes the image by simply averaging all pixel values within a patch (searching window). It was improved by the Gaussian filtering using a weighted Gaussian kernel. They both use the local information or neighboring pixels.

The NLM algorithm further improves the Gaussian filtering from the pixel-wise into patch-wise manner in terms of the weight calculation and dissimilarity measure. Although this improvement is quite simple, the NLM has been proved to be a powerful tool of image denoising. Two improved NLM versions the non-local Euclidean medians (NLEM) [11] and improved NLEM (INLEM) [12], promote the NLM's denoising performance by revising the weights ω_{ij} or/and the dissimilarity measure $\|X_i - Y_j\|^2$ in the optimization models shown in Table I.

These methods calculate the weights ω_{ij} from the noisy image patches and keep their values fixed for all denoising iterations. The weights ω_{ij} are rough estimates of the dissimilarity between two noisy patches. Due to the fact that the image becomes cleaner after each iteration, it is necessary to adaptively update the weights based on the denoising results in the previous iteration. In Section 2.2, we propose a novel NLM approach whose weights are set as fuzzy variables.

2.2. NLFM

This section introduces a novel non-local algorithm called non-local fuzzy means (NLFM) for image denoising. It considers the weights as fuzzy variables and updates their values in each denoising iteration. The proposed NLFM is summarized in Algorithm 1.

In above algorithm, the input parameter s and k denote the radii of the searching window and patch in the image respectively. Note that $\omega_{ij}^{(0)}$ and $X_i^{(0)}$ in Step 2(a) are used as initial values for the iterative calculation, and ω_{ij} in Step 2(b) are variables determined by the solutions of optimization problems.

Algorithm 1 Non-local Fuzzy Means (NLFM)

Input: The noisy image Y and parameters h, s, k, m.

- Step 1: Extract a patch, Y_i, with radius k centered at every pixel i in Y.
- Step 2: For every pixel i, do
  - **(a)** Set $\omega_{ij}^{(0)} = \exp\left(-\frac{\|Y_i - Y_j\|^2}{h^2}\right)$ for every $j \in S_i$, and compute $X_i^{(0)} = \sum_{j\in S_i} \frac{\omega_{ij}^{(0)} Y_j}{\sum_{j\in S_i}\omega_{ij}^{(0)}}$.
  - **(b)** Take $\omega_{ij}^{(0)}$ and $X_i^{(0)}$ as initial values, and find $\{\hat{X}_i, \hat{\omega}_{ij}\} = \text{argmin}_{X_i,\omega_{ij}} \sum_{j\in S_i} \omega_{ij}^m \|X_i - Y_j\|^2$, $s.t. \sum_{j\in S_i}\omega_{ij} = 1$ using an iterative way.
  - **(c)** Assign $\hat{X}(i)$ to be the value of the center pixel in $\hat{X}_i$.

Output: Denoised image $\hat{X}$.

The optimization model of the NLFM is defined by:

$$\{\hat{X}_i, \hat{\omega}_{ij}\} = \text{argmin}_{X_i,\omega_{ij}} \sum_{j\in S_i} \omega_{ij}^m \|X_i - Y_j\|^2. \quad (3)$$

In the above model, both X_i and ω_{ij} are unknown variables determined by the energy function. Weights ω_{ij} are adaptively updated in different iterations. These are the major differences between NLFM and the models in Table I.

Furthermore, we introduce an exponential parameter m ($m \geqslant 0$) to weights ω_{ij} which are usually normalized to $[0,\ 1]$. In the energy function, parameter m plays a significant role in the proposed NLFM. It performs a nonlinear mapping to weights ω_{ij} into different spaces. If $0 < m < 1$, small weights ω_{ij} are stretched into large values while large weights will be more compact. Opposite impacts appear when $m > 1$. As a result, m is tightly related to the contribution of $\|X_i - Y_j\|^2$ to the energy function, and the selection of m definitely effects the solution of the energy minimization problem as well as the denoising results. It is possible to tune the value of parameter m to achieve better denoising performance. Next, we will show how to implement the NLFM for different m values.

2.3. Optimization Strategies

The key step in the NLFM is to solve the optimization problem in Step 2(b). Different strategies are adopted according to different settings of parameter m.

2.3.1. $m = 0$

This means that weights ω_{ij} are neglected and NLFM reverts to the mean filtering.

2.3.2. $m > 0$ *but* $m \neq 1$

In this situation, we use the Lagrange multiplier method to solve the optimization model in Eq. (3). First, we define the following Lagrange function:

$$F = \sum_{j \in S_i} \omega_{ij}^m \|X_i - Y_j\|^2 + \lambda \left(\sum_{j \in S_i} \omega_{ij} - 1 \right). \quad (4)$$

By setting the derivative with respect to ω_{ij} equal to zero, we have

$$\frac{\partial F}{\partial \omega_{ij}} = m\omega_{ij}^{m-1} \|X_i - Y_j\|^2 + \lambda = 0. \quad (5)$$

Then ω_{ij} can be obtained in the following form,

$$\omega_{ij} = \left(\frac{-\lambda}{m\|X_i - Y_j\|^2} \right)^{\frac{1}{m-1}}. \quad (6)$$

Utilizing the constraint condition $\sum_{j \in S_i} \omega_{ij} = 1$, we obtain,

$$\omega_{ij} = \left(\sum_{l \in S_i} \left(\frac{\|X_i - Y_j\|}{\|X_i - Y_l\|} \right)^{\frac{2}{m-1}} \right)^{-1}. \quad (7)$$

When weights ω_{ij} have been determined by Eq. (7) under the constraint $\sum_{j \in S_i} \omega_{ij} = 1$, X_i can be obtained in the following way directly:

$$X_i = \sum_{j \in S_i} \omega_{ij} Y_j. \quad (8)$$

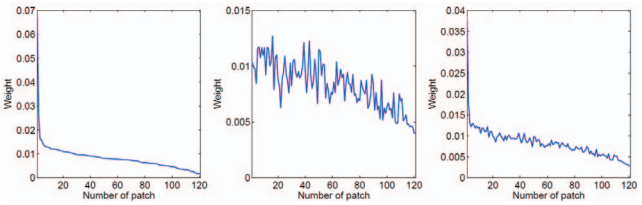
In practice, we apply an iterative way to to find optimal ω_{ij} and X_i. Namely, starting from an estimate $\omega_{ij}^{(t)}$ and $X_i^{(t)}$, the next iteration is set as

$$\omega_{ij}^{(t+1)} = \left(\sum_{l \in S_i} \left(\frac{\|X_i^{(t)} - Y_j\|}{\|X_i^{(t)} - Y_l\|} \right)^{\frac{2}{m-1}} \right)^{-1}, \quad (9)$$

and

$$X_i^{(t+1)} = \sum_{j \in S_i} \omega_{ij}^{(t+1)} Y_j. \quad (10)$$

As aforementioned, $X_i^{(0)}$ and $\omega_{ij}^{(0)}$ are initial conditions to start denoising iterations. An example is given in Fig. 1 to show the iterative processes. As illustrated in Fig. 1(a), weights $\omega_{ij}^{(0)}$, calculated from a noisy image, are plotted in a descending order. $X_i^{(0)}$ can be obtained using Eq. (10). Then we compute weights $\omega_{ij}^{(1)}$ according to Eq. (9), which are plotted in Fig. 1(b). It can be seen that the distribution of weights is greatly changed. If we directly utilize these weights, the image will be over-denoised. To avoid this problem, we update



(a) (b) (c)

Fig. 1: Weights ω_{ij} used in different steps of the NLFM: (a) Step 2(a) in Algorithm 1; (b) Eq. (9); (c) Eq. (11).

the weights by combining $\omega_{ij}^{(t+1)}$ and $\omega_{ij}^{(t)}$ in the following way:

$$\bar{\omega}_{ij}^{(t+1)} = \alpha\omega_{ij}^{(t+1)} + (1 - \alpha)\omega_{ij}^{(t)}, \quad (11)$$

where $0 \leqslant \alpha \leqslant 1$. Fig. 1(c) shows the weights obtained based on Eq. (11) with $\alpha = 0.5$. We can find that the weights updated in this way not only preserve the tendency of the original ones but also accommodate the weights for some specific patches. Finally, the iterations will stop on the condition that $\|X_i^{(t+1)} - X_i^{(t)}\|$ is smaller than a given threshold.

Note that m is recommended to be a value not more than 5. Otherwise, all ω_{ij}^m becomes small and similar values. The following two problems may arise. First, the weights has significantly reduced contribution to the minimization problem. Second, it requires more iterations to minimize the energy function.

2.3.3. $m = 1$

Because the Lagrange multiplier method fails to solve the optimization problem in this situation, we then update X_i and ω_{ij} using an approximation algorithm. Assume that, after t times iterations, a total of t pairs of $\omega_{ij}^{(t)}$ and $X_i^{(t)}$ are obtained, where $t = 0, 1, 2, \cdots$. Then we update ω_{ij} by

$$\omega_{ij}^{(t+1)} = \exp\left(-\frac{\|X_i^{(t)} - Y_j\|^2}{h^2} \right) \text{ for every } j \in S_i. \quad (12)$$

After that, $X_i^{(t)}$ is updated by

$$\hat{X}_i^{(t+1)} = \text{argmin}_{X_i} \sum_{j \in S_i} \omega_{ij}^{(t+1)} \|X_i - Y_j\|^2, \quad (13)$$

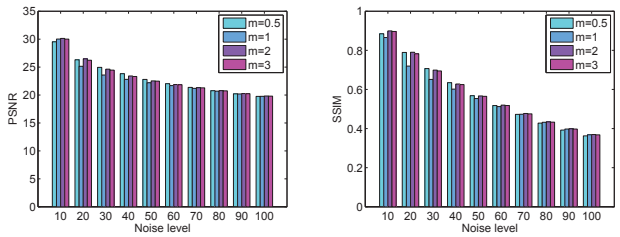
This is similar to the classical NLM in this step. However, different from the classical NLM, NLFM performs iterative denoising processes. Its denoised patches still are derived from the noisy patches $Y_j, j \in S_i$. The stop criterion of denoising iterations here is the same as those in the situation $m \neq 1$.

3. EXPERIMENTS

In this section, experiments are carried out to demonstrate denoising performance of the proposed NLFM. Similar to

Table 2: Comparison of the NLFM with other methods in terms of PSNR and SSIM at noise levels $\sigma = 10, 20, \cdots, 100$.

Image	Method	PSNR (dB)										SSIM									
		10	20	30	40	50	60	70	80	90	100	10	20	30	40	50	60	70	80	90	100
Peppers	NLM	**32.57**	27.84	25.29	23.69	22.58	21.87	21.22	20.69	20.15	19.67	0.895	0.820	0.760	0.703	0.651	0.603	0.559	0.515	0.477	0.439
	NLEM	32.04	28.01	25.44	23.86	22.65	21.89	21.27	20.75	20.25	19.76	0.892	0.822	0.762	**0.707**	**0.655**	**0.608**	0.565	0.520	0.483	0.444
	INLEM	31.52	28.14	26.10	24.37	23.06	22.17	21.43	20.84	20.27	19.75	0.891	0.822	0.754	0.683	0.621	0.568	0.522	0.478	0.444	0.406
	NLFM	32.04	**28.49**	**26.33**	**24.69**	**23.41**	**22.54**	**21.63**	**20.96**	**20.38**	**19.87**	**0.900**	**0.835**	**0.773**	**0.707**	0.647	**0.608**	**0.568**	**0.526**	**0.492**	**0.453**
C.man	NLM	31.69	27.96	24.72	23.09	22.12	21.45	20.92	20.46	19.79	19.35	0.874	0.798	0.715	0.642	0.572	0.519	0.473	0.436	0.396	0.372
	NLEM	31.40	28.03	24.95	23.15	22.16	21.49	20.98	20.52	19.86	19.45	0.874	0.801	0.720	0.647	0.578	0.524	0.479	0.442	0.402	0.377
	INLEM	31.91	27.97	25.86	23.90	22.55	21.72	21.10	20.57	19.87	19.43	0.895	0.800	0.717	0.631	0.551	0.492	0.443	0.407	0.368	0.345
	NLFM	**32.29**	**28.35**	**26.28**	**24.53**	**22.90**	**21.88**	**21.17**	**20.61**	**19.91**	**19.49**	**0.908**	**0.813**	**0.739**	**0.662**	**0.586**	**0.535**	**0.493**	**0.456**	**0.416**	**0.391**
Couples	NLM	30.49	25.55	24.06	23.30	22.72	22.23	21.80	21.24	20.77	20.51	0.847	0.689	0.618	0.559	0.508	0.470	0.429	0.391	0.366	0.340
	NLEM	30.44	25.66	24.04	23.28	22.72	22.24	21.82	21.27	20.82	20.57	0.852	0.691	0.617	0.560	0.508	0.471	0.430	0.392	0.368	0.342
	INLEM	30.45	26.51	24.67	23.62	22.88	22.29	21.80	21.20	20.73	20.46	0.869	0.735	0.642	0.566	0.502	0.457	0.412	0.372	0.348	0.323
	NLFM	**30.82**	**27.00**	**25.06**	**23.80**	**22.97**	**22.36**	**21.87**	**21.28**	**20.85**	**20.60**	**0.879**	**0.757**	**0.661**	**0.579**	**0.514**	**0.473**	**0.434**	**0.397**	**0.374**	**0.348**
House	NLM	**34.51**	30.20	27.62	26.27	25.35	24.61	23.82	23.07	22.28	21.71	0.881	0.815	0.750	0.688	0.631	0.580	0.529	0.481	0.438	0.412
	NLEM	34.22	30.47	27.76	26.34	25.40	24.68	23.92	23.16	22.39	21.82	0.880	0.818	**0.754**	**0.693**	**0.637**	0.586	0.536	0.488	0.444	0.418
	INLEM	33.15	30.43	28.17	26.56	25.48	24.63	23.79	22.99	22.21	21.63	0.878	0.808	0.730	0.652	0.586	0.531	0.480	0.434	0.394	0.371
	NLFM	33.99	**30.76**	**28.59**	**26.96**	**25.79**	**24.87**	**24.05**	**23.26**	**22.49**	**21.92**	**0.887**	**0.821**	0.753	0.683	0.626	**0.587**	**0.540**	**0.495**	**0.454**	**0.429**



(a) (b)

Fig. 2: Quantitative measures of image denoising results using the proposed NLFM with different m values for various noise levels: (a) PSNR; (b) SSIM.

most literatures, the restored images are evaluated by both the structural similarity (SSIM) index [16] and peak signal-to-noise rate (PSNR). The PSNR index is defined as

$$\text{PSNR} = 10 \cdot \log_{10} \frac{255^2 MN}{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left[X(i,j) - \hat{X}(i,j) \right]^2}. \quad (14)$$

First, the effect of parameter m in NLFM is investigated experimentally. Four common test images, "Lena", "Fingerprint", "Mandril", and "Boats", are selected for this test, and m is set to be 0.5, 1, 2, and 3 respectively. Each test image is corrupted by adding Gaussian noise with different σ values changing from 10 to 100 with a step of 10. The restoration results in terms of PSNR and SSIM are plotted in Fig. 2. They are the average results of all test images in different noise levels. As can be seen, the restored results of $m = 0.5$, 1, 2 and 3 are better than those of $m = 1$ for noise levels less than $\sigma = 60$, while all m values perform equally for larger noise levels. In observation of overall performance, $m = 2$ is superiors to other values in most cases. Therefore, m is set to 2 in the following experiments in this paper.

In the following experiments, we compare the proposed NLFM with NLM, NLEM, INLEM on several standard test images. All images, including the noise-free images and their corresponding noisy versions, are limited to the size of 256×256 and a range of pixel values between 0 to 255. The parameters for NLM, NLEM, and INLEM are set as recommended in [12]. For the NLFM, we set $h = 6\sigma, s = 10, k = 3$.

For a comparison purpose, all original test images are corrupted by different degrees of noise with the same parameters used in the former experiments. The corresponding PSNR and SSIM results are reported in Table II. It can be concluded from Table II that the NLFM achieves the best denoising results in most situations. Particularly, the NLFM obtains obvious improvements in the PSNR measures when $\sigma < 60$. Considering the SSIM measures, the improvements of the NLFM are better although it works slightly worse than the NLEM in few cases. This is because the optimization model for the NLFM is to minimize the square error sum with respect to the intensity of images and thus the structural details may not recover very well sometimes.

4. CONCLUSION

In this paper, we have introduced a new NLFM which is an improved version of the NLM for image denoising. Unlike the NLM and some of its improvements, whose weights are set to be constants in all denoising iternations, the proposed NLFM considers the weights as fuzzy variables which are determined by solving an energy minimization problem. In this way, the weights are able to be updated and adaptive to changes of image contents and denoising results. We also introduced an exponential parameter to the fuzzy weights to map them to a proper space for enhancing the NLFM's denoising performance. Experiments were carried out to evaluate the NLFM, and promising results have been achieved.

5. REFERENCES

[1] A. Buades, B. Coll, and J. M. Morel, "A non-local algorithm for image denoising," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, vol. 2, pp. 60–65 vol. 2.

[2] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *SIAM Interdisc. J.: Multiscale Modeling and Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[3] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[5] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognition*, vol. 43, no. 4, pp. 1531–1549, 2010.

[6] P. Chatterjee and P. Milanfar, "Patch-based near-optimal image denoising," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1635–1649, 2012.

[7] A. Rajwade, A. Rangarajan, and A. Banerjee, "Image denoising using the higher order singular value decomposition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 4, pp. 849–862, 2013.

[8] C. Kervrann, "PEWA: Patch-based exponentially weighted aggregation for image denoising," in *Advances in Neural Information Processing Systems*, 2014, pp. 2150–2158.

[9] N. Dowson and O. Salvado, "Hashed nonlocal means for rapid image filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 485–499, 2011.

[10] R. Vignesh, B. T. Oh, and C. C. J. Kuo, "Fast non-local means (NLM) computation with probabilistic early termination," *IEEE Signal Processing Letters*, vol. 17, no. 3, pp. 277–280, 2010.

[11] K. N. Chaudhury and A. Singer, "Non-local euclidean medians," *IEEE Signal Processing Letters*, vol. 19, no. 11, pp. 745–748, 2012.

[12] Z. Sun and S. Chen, "Analysis of non-local euclidean medians and its improvement," *IEEE Signal Processing Letters*, vol. 20, no. 4, pp. 303–306, 2013.

[13] Y. Wu, B. Tracey, P. Natarajan, and J. P. Noonan, "James-stein type center pixel weights for non-local means image denoising," *IEEE Signal Processing Letters*, vol. 20, no. 4, pp. 411–414, 2013.

[14] Y. Wu, B. Tracey, P. Natarajan, and J. P. Noonan, "Probabilistic non-local means," *IEEE Signal Processing Letters*, vol. 20, no. 8, pp. 763–766, 2013.

[15] L. Liu, CL P. Chen, Y. Zhou, and X. You, "A new weighted mean filter with a two-phase detector for removing impulse noise," *Information Sciences*, 2015.

[16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.