PAPER

Deblocking Artifact of Satellite Image Based on Adaptive Soft-Threshold Anisotropic Filter Using Wavelet

RISNANDAR[†*a)], ***Nonmember and*** **Masayoshi ARITSUGI**[††b)], ***Senior Member***

SUMMARY New deblocking artifact, or blocking artifact reduction, algorithms based on nonlinear adaptive soft-threshold anisotropic filter in wavelet are proposed. Our deblocking algorithm uses soft-threshold, adaptive wavelet direction, adaptive anisotropic filter, and estimation. The novelties of this paper are an adaptive soft-threshold for deblocking artifact and an optimal intersection of confidence intervals (OICI) method in deblocking artifact estimation. The soft-threshold values are adaptable to different thresholds of flat area, texture area, and blocking artifact. The OICI is a reconstruction technique of estimated deblocking artifact which improves acceptable quality level of estimated deblocking artifact and reduces execution time of deblocking artifact estimation compared to the other methods. Our adaptive OICI method outperforms other adaptive deblocking artifact methods. Our estimated deblocking artifact algorithms have up to 98% of MSE improvement, up to 89% of RMSE improvement, and up to 99% of MAE improvement. We also got up to 77.98% reduction of computational time of deblocking artifact estimations, compared to other methods. We have estimated shift and add algorithms by using $Euler++(E++)$ and Runge-Kutta of order 4++ ($RK4++$) algorithms which iterate one step an ordinary differential equation integration method. Experimental results showed that our $E++$ and $RK4++$ algorithms could reduce computational time in terms of shift and add, and $RK4++$ algorithm is superior to $E++$ algorithm.

key words: *deblocking artifact, soft-threshold, reconstruction, estimation*

1. Introduction

The discrete cosine transform (DCT) is the most popular image compression technique because its performance is known to be optimal in the mean squared error (MSE) term [1]–[4]. Our previous study [5] and the block DCT (B-DCT) which used in JPEG [6], [7], H.264 [8], and MPEG [9], show that DCT leaves blocking artifacts problem with linear filtering [2]–[5]. This is caused by the higher compression.

Various algorithms have been proposed for removing the blocking artifacts. The methods of [10]–[12] used the projection onto convex sets (POCS) techniques. POCS uses a dynamic focus-plus context (DF+C) [13] and the improved weighted projection onto convex sets (IWPOCS) [14]. The techniques make good effects, but the main disadvantages of DF+C and IWPOCS methods are non-unique solution, always slow convergence, high computational time [15] and also unstable of numerical computation [16] in reconstruction of the detailed structures of the satellite image. The other disadvantages of spatially adaptive techniques in DF+C and IWPOCS methods are much greater precomputation and searching processes than wavelet transform, and also the limited reference of the satellite image.

In this paper, an adaptive deblocking artifact method based on the intersection of confidence intervals (ICI) rule [17]–[19] is discussed. The ICI rule is an adaptive procedure for selecting the appropriate adaptive scale parameter in each pixel of blocking artifact to get deblocking artifact. Nevertheless the methods in [17]–[19] tend to have a result of low quality and high computational time of deblocking artifact. To overcome these drawbacks, we propose deblocking artifact based on nonlinear adaptive soft-threshold anisotropic filter in wavelet.

Our adaptive soft-threshold removes blocking artifact significantly from both distorted and undistorted satellite images in different sub-bands of wavelet transform. It is adaptable to high transitions signal criteria of blocking artifact, flat area, and texture area. The other contribution of this paper is a reconstruction technique of deblocking artifact by using an optimal ICI (OICI) method. Our OICI improves quality of estimated deblocking artifact which is determined by soft-threhold in confidence intervals for every scale parameter. For large scale parameter, soft-threshold in confidence intervals and deblocking artifact estimation are more ideal. Our OICI also reduces execution time of deblocking artifact estimation. In order to further improve estimated deblocking artifact performance, OICI determines scale parameter from the left or right side kernel only, while original ICIs [17]–[19] determine scale parameter from both the left and right side kernels, which have higher computational time.

In computational complexity, we propose $E++$ and $RK4++$ algorithms for the computation of elementary functions like exponential, logarithm, trigonometric functions, hyperbolic functions, and their reciprocals in fixed precision, typically the computer single or double precision. Our proposed method combines shift and add algorithms and both classical methods, Euler and Runge-Kutta methods for the numerical integration of ordinary differential equations (ODEs).

The paper is organized as follows. Related work is given in Sect. 2. Section 3 provides two steps of our proposal. The first step is proposed for deblocking artifact of

Manuscript received January 11, 2018.
Manuscript publicized February 26, 2018.

[†]The author is with the Research Center for Informatics, Indonesian Institute of Sciences, Bandung, Indonesia.
[††]The author is with the Faculty of Advanced Science and Technology, Kumamoto University, Kumamoto-shi, Japan.
[*]Presently, with the Dept. of Computer Science and Electrical Engineering, Kumamoto University.
a) E-mail: risnandar@lipi.go.id
b) E-mail: aritsugi@cs.kumamoto-u.ac.jp

DOI: 10.1587/transinf.2018EDP7013

satellite image by using soft-threshold in wavelet, followed by an adaptive anisotropic filter. The second step is reconstruction and smoothing, followed by LPA estimation, OICI algorithm, and fusion of deblocking artifact as final estimation. Section 4 shows our experimental results and algorithm performances. Conclusions are found in Sect. 5.

2. Related Work

2.1 Deblocking Artifact

In [20], [21] methods for deblocking artifact of image were developed by using the wavelet for solving the drawbacks in [10]–[14]. The method of [20] applies a variety of wavelet-based multitemporal DInSAR algorithms to remove artifacts, such as spatially correlated and temporally uncorrelated components, the digital elevation model, and orbital errors of an image. However, in various images which have low bit rates, the method of [20] often removes some part of undistorted satellite image, like texture area and flat area that are assumed as distorted images. The method of [21] also can remove blocking artifacts by using orthogonal projections of wavelets from an upside down pyramid-shaped region in a multi-dimensional space. The method of [21] also has drawbacks, such as lack of shift invariance, lack of wavelet symmetry, and poor directionality, when projection-based wavelet [21] uses dyadic wavelet transform [22]–[24], logarithmic dyadic wavelet transform (LDWT) [25], and dyadic bounded mean oscillating (BMO) [26]. Blocking artifact removal methods in [22]–[26] and overcomplete wavelet representation in [27], [28] are not suitable for various undistorted satellite images which have texture area and flat area, either. Discrete and continuous-time soft-threshold [29] and a linear expansion of thresholds (LET) [30] can remove blocking artifact and treat the drawbacks of [22]–[28].

Another deblocking artifact method [31] uses linear filtering and its estimation. Unfortunately, it does not work with adaptation. Sparse representation [31] removes blocking artifacts from a set of training images by using K-singular value decomposition (K-SVD) and the orthogonal matching pursuit (OMP) for the estimated error threshold. However, since K-SVD is a linear representation of the data, K-SVD has to face nonlinear distributions of the data. It often leads to poor performance, it lacks the capability to separate different classes and also dictionary learning redundant. OMP builds the solution by adding one vector of estimated error threshold in every distorted image criterion and also OMP results depend on the elementary signals, or atoms, in dictionary learning [31]. However, it produces a big number of iterations, which is difficult to determine an appropriate threshold value of different distorted images. For a large number of data and iterations, performing an atom decomposition can take high computational time. In addition, they [22]–[31] did not consider soft-threshold criteria for different images and their levels of distorted images.

Thus, removing blocking artifact has been left problem for linear filtering [20]–[22], [24]–[31] in satellite image processing analysis. The linear filter methods [20]–[22], [24]–[31] give blurring effect in edges and other fine satellite image details. It is difficult for image analysis of linear filter to reconstruct an undistorted image from a distorted image. This has been treated in [32]–[35]. They [20]–[22], [24]–[31] allow us to find an alternative method by using nonlinear filtering. In particular, median filtering tends to give good results. Median filter is quite popular for nonlinear filtering in image processing applications [36] because it is simple and can preserve edges. A variety of median filters, such as stack filters and adaptive stack filters [37]–[39], the mean absolute error (MAE) of stack filter [40], multilevel median and FIR-median hybrid filter [41], adaptive switching median filter [42], optimal weighted vector median [43], [44], generalized Gaussian median filter [45], and relaxed median filter [46] have been developed to treat these drawbacks of [20]–[22], [24]–[31].

Although median filter and its various methods [37]–[46] are useful as nonlinear filter for satellite image denoising and enhancement, yet they also have some drawbacks. The median filter removes both distorted and undistorted satellite images when it cannot detect the differences between them. In case of satellite image of small size, it has minimal effect on the value of the median. This effect problem is still filtered out. The crucial element of estimations in nonlinear filter methods is adaptation.

2.2 Soft-Threshold

The soft-threshold is one of the most popular threshold methods used for deblocking artifact [47]. The performances of these methods [47], [48] are close to an ideal coefficient selection method if the coefficients of the observed blocking artifact signal are known in advance. The soft-threshold is more efficient than hard-threshold in deblocking artifact. The authors of [47], [48] gave the mechanism for finding a universal threshold value, which is called VisuShrink. VisuShrink depends on the noise power and the number of samples in the satellite image, or signal size. VisuShrink is a single value of threshold which is applied to all the wavelet coefficients. This is derived by proving an approximation error in the limit of an arbitrary large signal size. This method consists of the magnitudes of signal which are less than the threshold and the magnitudes of noise which are more than the given threshold. These methods [47], [48] would cut off the parts of the true signal and also leave some noise in distorted satellite image. In this paper, we propose an adaptive soft-threshold based on the level of distorted satellite image to overcome the drawbacks in [47], [48]. Our soft-threshold values use a nonlinear method which is adaptive to various satellite images and their blocking artifacts.

The authors of [29], [30] proposed soft-threshold for removing blocking artifacts. In [29], the iterative soft-threshold algorithm (ISTA) was presented for performing a discrete gradient step followed by a soft-threshold operation at each iteration. At the same time, this method [29] would pay high computational cost resulting in the slow convergence rate in fusion of deblocking artifact. The method of [29] would be difficult to analyze the noise properties in distorted satellite image. In [30], a linear expansion of thresholds (LET) is an excellent estimate of the noiseless image by decomposing the denoising process and optimizing the coefficients of this representation using an estimate of the Stein's unbiased risk estimate (SURE). Unfortunately, they showed unstable and slow convergence performance, in particular for various satellite images.

Thus, we propose the accelerated gradient of convergence in fusion step for treating these drawbacks of [29] and [30]. The accelerated gradient of convergence is a function value of a kernel which improves convergence performance in estimation, particularly in the fusion of deblocking artifact. The accelerated gradient of convergence which is demonstrated in the kernel builds iteration steps. The step taken at each iteration depends on the previous iterations, where the accelerated gradient of convergence grows from one iteration to the next iteration. When we take the current iteration as the new starting point, this erases the previous iterations and resets the accelerated gradient of convergence back to zero.

Many adaptation techniques have been proposed. They used wavelet shrinkage [48], wavelet shrinkage and adaptive elliptical kernel for image smoothing [49], and generalized shrinkage-threshold for deblurring images [50]. The Lepski's methods like 1D estimator [51], minimax adaptive estimation [52], spatial adaptation inhomogeneous smoothness [53], and pointwise adaptive [54] are adopted in our paper. Our adaptation is proposed based on a nonlinear estimation of distorted satellite image with its edge recovering. A nonlinear estimator is applied, which is derived from a local polynomial approach of satellite image in a wavelet window. Generalized linear model [55], estimating regression [56], smoothing adaptation [57], sharp adaptive estimation [58], signal dependent noise (SDN) model [59], and an adaptive jump-preserving (AJP) estimation [60] discussed the kernel by using an adaptive estimation. Yet, they did not apply Lepski's methods as we did. The further idea of [51] was proposed as SDN model [59] for single image noisy estimation by using the noise level function of signal-dependent noise which assumed the generalized signal-dependent noise model and the Poisson-Gaussian noise model. Moreover the adaptive methods [55]–[60] are difficult to analyze mathematically because they apply a diffusion process by using derivatives of the evolving image for smoothing and enhancing the important feature of distorted satellite image, such as edges and the details of fine satellite images. To deal with these drawbacks of [55]–[60], we propose an adaptive nonlinear diffusion filter by using anisotropic diffusion filter. Our anisotropic adaptation filter method finds a point of estimation. Our polynomial component of signal fits well with the entire distorted satellite image. Distorted satellite image estimations are calculated for a grid of window size. Each grid is compared to the other grids of window size in wavelet. The adaptation window size is defined as the largest window size in the grid.

2.3 Intersection of Confidence Intervals (ICI)

The ICI rule [17]–[19] is an automatic adaptive procedure for selecting the appropriate adaptive filter of scale parameter m for each pixel in each satellite image in order to obtain a deblocking artifact with minimal estimation error and scale parameter m is non-negative vector. Let us consider distorted satellite image $d(r,s)$ and undistorted satellite image $u(r,s)$, where r and s are the pixel indices. The absolute estimation error e of them can be calculated as follows:

$$|e(r,s,m)| = |u(r,s) - \hat{d}(r,s,m)| \tag{1}$$

where $|\hat{d}(r,s,m)|$ is the estimated distorted satellite image, obtained using the scale parameter m. As mentioned in [18], the absolute estimation error is

$$|e(r,s,m)| \leqslant |\hat{E}(r,s,m)| + |e^0(r,s,m)| \tag{2}$$

where $|\hat{E}(r,s,m)|$ is the maximum value of the estimation bias and $|e^0(r,s,m)|$ is random error with probability density $N(0,\sigma_d^2(r,s,m))$.

It was shown in [18] the following inequality holds

$$|e^0(r,s,m)| \leqslant G_{(1-\frac{\alpha}{2})}\sigma_{\hat{d}_m}(r,s,m) \tag{3}$$

where $G_{(1-\frac{\alpha}{2})}$ is $(1-\frac{\alpha}{2})$-th quantile of the standard Gaussian distribution and $\sigma_{\hat{d}_m}(r,s,m)$ is the standard deviation of distorted satellite image in orientation α. The following inequality can be derived.

$$|e(r,s,m)| \leqslant |\hat{E}(r,s,m)| + G_{(1-\frac{\alpha}{2})}\hat{d}_m(r,s,m) \tag{4}$$

We define an abbreviated Eq. (4) to Eq. (5) as follows:

$$|e(r,s,m)| \leqslant \delta\tau\sigma_{\hat{d}_m}(r,s,m) \tag{5}$$

for $m \leqslant m'(r,s)$, where $m'(r,s)$ is the estimated scale parameter m of the estimated distorted satellite image $\hat{d}(r,s,m)$ and soft-threshold $\delta\tau$ is used in confidence interval.

The adaptive scale parameter m procedure based on the ICI rule [61] introduces a finite set of parameter sizes and calculates a sequence of confidence intervals limits of the biased estimates for each satellite image pixel separately and independently to its left or right hand side. The one side of confidence interval limits of our proposed algorithm can be defined as follows:

$$U_i(r,s,m_i) = \hat{d}_m(r,s,m_i) + \sigma\delta\tau_{\hat{d}_m}(r,s,m_i), \tag{6}$$

$$L_i(r,s,m_i) = \hat{d}_m(r,s,m_i) - \sigma\delta\tau_{\hat{d}_m}(r,s,m_i), \tag{7}$$

where $U_i(r,s,m_i)$ and $L_i(r,s,m_i)$ are the i-th upper and lower confidence interval limits, respectively. $\hat{d}_m(r,s,m_i)$ is estimated deblocking artifact of scale parameter m. We calculate the intervals and their intersections which are being

Table 1 Conventional vs non-conventional deblocking artifact methods

No.	Method	Advantages	Disadvantages
1.	**Conventional**	-	No estimation, no adaptation, and many iterations.
	a. Improved weighted projection onto convex sets (IWPOCS) [14]	Most effective and good effects for deblocking artifact.	Non-unique solution, slow convergence, high computational time and also unstable of numerical computation in reconstruction of deblocking artifact.
	b. Overcomplete wavelet [28]	Good shift invariance, wavelet symmetry, and directionality.	Difficult to analyze because all of the distorted and undistorted satellite images are unseparated. It is not suitable for various undistorted satellite images which have texture area and flat area, either.
2.	**Non-conventional**	Adaptive filtering and estimation and also limited iterations.	-
	a. LPA-ICI [18], [19]	Cross-validation which was proven to be a good criterion for selection of threshold resulting in estimation accuracy improvement.	High consuming time for deblocking artifact, because it has two iterations from both, right and left side. It also has low quality deblocking artifact.
	b. Linear expansion of thresholds (LET) [30]	An excellent estimate of deblocking artifact.	Unstable and slow convergence performance, in particular for various satellite images.
	c. Signal dependent noise (SDN) [59]	Can simultaneously estimate all parameters of deblocking artifact by SDN and can evaluate distorted satellite images.	Difficult to analyze because it uses a diffusion process by using derivatives of the evolving satellite image for smoothing and enhancing deblocking artifact, such as edges and the details of fine satellite images.
	d. Multivariate wavelet denoising (MWD) [66]	To improve deblocking artifact over the conventional univariate wavelet directly.	Deleting texture area causes loss of fine satellite images. and flat area. Insufficient of deblocking artifact in the low bit of distorted satellite image.

tracked with the following Eq. (8).

$$\max_{i=1,\ldots,D-1} L_i(r,s,m_i) \leqslant \min_{i=1,\ldots,D-1} U_i(r,s,m_i) \tag{8}$$

The largest i is the proper scale parameter m in the number of deblocking artifact D. The left or right hand side of distorted satellite image $d(r,s)$ represents as blocking artifact which is combined and also used in estimated deblocking artifact $\hat{d}(r,s,m)$. Soft-threshold $\delta\tau$ has important role for confidence intervals and estimated deblocking artifact efficiency. For small values of soft-threshold $\delta\tau$, variance increase and bias estimation of scale parameter m decrease.

Deblocking artifact by using ICI based method [18], [19] has high computational time. To deal with this drawback, we propose an optimal ICI (OICI) method. OICI method improves a quality level of estimated deblocking artifact and reduces estimated deblocking artifact computational time compared to the ICI method. Our OICI method does not only calculate scale parameter for each estimated deblocking artifact pixel like the ICI method, but detects the appropriate time regions for each pixel in time as well. It calculates the first pixel value in the exacted blocking artifact time region of distorted satellite image. It uses a scale parameter in the detected time region for all estimated undistorted satellite images pixel values. Thus, the optimal time region is an important role in our estimated deblocking artifact quality. Cross-validation method [18], [61] and local adaptive transform [19] used the ICI method. They were proven to be a good estimation improvement. However, these previous studies [18], [19], [61] have higher computational time of deblocking artifact estimation compared to our OICI method. We describe the pros and cons comparison between conventional and non-conventional deblocking artifact methods in Table 1.

2.4 Ordinary Differential Equations (ODEs)

In this paper we propose fixed precision which is provided by the float type on any computer. The previous study [62] mentioned that some methods could be employed by using Padé approximations. However, since a fixed number of digits is required, pre-computed tables can be used. These tables have a fixed number of entries that are used in the same fixed storage. Thus, these methods [62] have been developed for fixed precision computations of elementary functions (exponential, logarithm, trigonometric functions, hyperbolic functions, and their reciprocals). Shift and add methods [62] belong computations of elementary functions. The methods decompose their arguments into a number of decomposition which is performed by means of additions only. Furthermore, shift and add algorithms are computed along with the decomposition and requires only additions and multiplications or divisions by 2, which are realized on a computer by shifts. Since additions and shifts are very efficiently performed, the estimated iteration is very efficient. Moreover, the number of the estimated iterations is small.

3. Deblocking Artifact and Reconstruction Algorithms

Deblocking artifact of our proposed algorithms starts with the detection of blocking artifact discussed in our previous study [5]. In this section, firstly, we propose new algorithms of deblocking artifact by using soft-threshold, adaptive wavelet direction, and adaptive anisotropic filter. Secondly, we propose new algorithms of deblocking artifact reconstruction and estimations by using LPA estimation, OICI estimation, and fusion estimate. Our proposed algorithms are illustrated in Fig. 1.

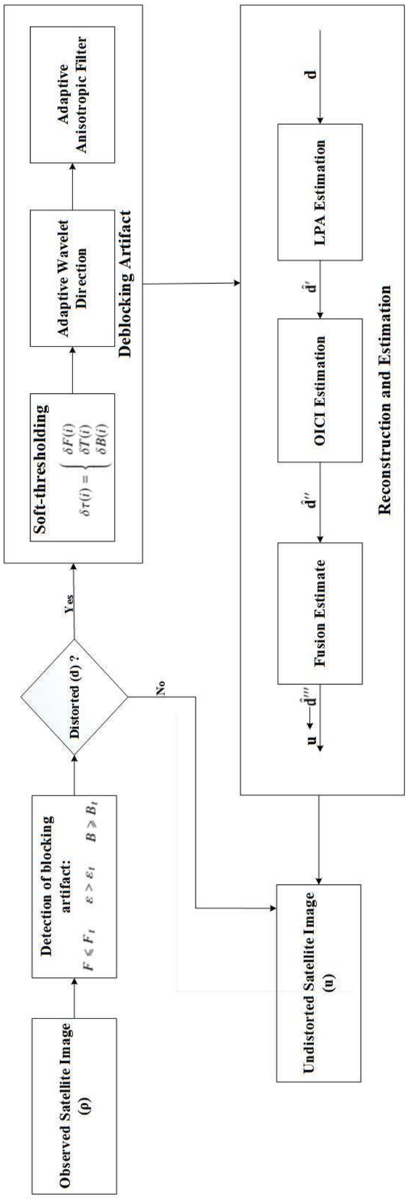


Fig. 1 Proposed algorithms

3.1 Deblocking Artifact

3.1.1 Soft-Threshold

In our proposed algorithms, we have a wavelet transform and adaptive filter for wavelet analysis. Our wavelet filter should keep a low computational time in soft-threshold. Thus, we need a symmetrical wavelet filter to keep spatial positions of blocking artifact in different scales. In the decomposition process, we have three high-pass filters, namely, horizontal high-pass or vertical low-pass (HL), horizontal low-pass or vertical high-pass (LH), and horizontal high-pass or vertical high-pass (HH). These are very important for the exacted blocking artifact position as we mentioned in [5].

We denote high-low, low-high, and high-high subband as $\{HL_F(i), LH_F(i), HH_F(i)\}$, $\{HL_\varepsilon(i), LH_\varepsilon(i), HH_\varepsilon(i)\}$, and $\{HL_B(i), LH_B(i), HH_B(i)\}$ for flat area, texture area, and blocking artifact, respectively. Blocking artifacts appear at the horizontal and vertical of edge directions. We have verified and compared a specific distorted satellite image with other compressed satellite image in our previous study [5].

Distorted satellite image is indicated as blocking artifact. Flat area, e.g., plain, plateau, steppe, tableland, tundra, and texture area, e.g., forest, wave of the sea, and mountain are indicated as an undistorted satellite image or normal compressed satellite image. However, some detections of blocking artifact detect flat area and/or texture area as blocking artifact.

To overcome this drawback, we find the thresholds of flat area, texture area, and blocking artifact, respectively, as described in our previous study [5]. We determine threshold criteria which depend on the exacted blocking artifact location in every decomposition level i-th, by comparing wavelet coefficient values of flat area to the flat area threshold ($F \leqslant F_t$), the entropy of texture area to the entropy threshold ($\varepsilon > \varepsilon_t$), and blocking artifact to the blocking artifact threshold ($B \geqslant B_t$). The thresholds are determined by Eqs. (12) and (13). We define the relations of them in Eqs. (9)–(11), respectively.

$$F = \begin{cases} |C_{HL_F(i)}(r,s)| \leqslant |C_{HL_{F_t}(i)}(r,s)| \\ |C_{LH_F(i)}(r,s)| \leqslant |C_{LH_{F_t}(i)}(r,s)| \\ |C_{HH_F(i)}(r,s)| \leqslant |C_{HH_{F_t}(i)}(r,s)| \end{cases} \tag{9}$$

$$T = \begin{cases} |C_{HL_\varepsilon(i)}(r,s)| > |C_{HL_{\varepsilon_t}(i)}(r,s)| \\ |C_{LH_\varepsilon(i)}(r,s)| > |C_{LH_{\varepsilon_t}(i)}(r,s)| \\ |C_{HH_\varepsilon(i)}(r,s)| > |C_{HH_{\varepsilon_t}(i)}(r,s)| \end{cases} \tag{10}$$

$$B = \begin{cases} |C_{HL_B(i)}(r,s)| \geqslant |C_{HL_{B_t}(i)}(r,s)| \\ |C_{LH_B(i)}(r,s)| \geqslant |C_{LH_{B_t}(i)}(r,s)| \\ |C_{HH_B(i)}(r,s)| \geqslant |C_{HH_{B_t}(i)}(r,s)| \end{cases} \tag{11}$$

where $C_{subband}(r,s)$ indicates the wavelet coefficient in the position of subband coordinate *(r,s)*.

We can obtain the soft-threshold values of flat area δF, texture area δT, and blocking artifact δB, as described in Eqs. (14)–(16), respectively, from the wavelet decomposition process. After the wavelet decomposition, blocking artifact appears as horizontal line-shaped edge in the *LH* subband and vertical line-shaped edge in the *HL* subband. For the whole satellite images containing blocking artifacts, we calculate the three level wavelet decompositions. According to the wavelet decomposition, we block the high-low subband of each satellite image block together from left to right and then top to bottom. We also block the high-low, low-high, and high-high subbands of i-th level decomposition of the whole satellite images.

The high-low, low-high, and high-high subbands only appear on the block boundary in i-th level wavelet decomposition. The wavelet decomposition is a process downsampled by the length of the high-pass which equals to 3, and blocking artifacts appear on the block boundary.

The soft-threshold criteria of flat area δF, texture area δT, and blocking artifact δB are determined by calculating difference of wavelet subbands coefficient between $HL_{F_t}(i)$ and $HL_F(i)$, $LH_{F_t}(i)$ and $LH_F(i)$, $HH_{F_t}(i)$ and $HH_F(i)$, $HL_\varepsilon(i)$ and $HL_{\varepsilon_t}(i)$, $LH_\varepsilon(i)$ and $LH_{\varepsilon_t}(i)$, $HH_\varepsilon(i)$ and $HH_{\varepsilon_t}(i)$, $HL_{B_t}(i)$ and $HL_B(i)$, $LH_{B_t}(i)$ and $LH_B(i)$, and the last one $HH_{B_t}(i)$ and $HH_B(i)$. Then, difference of wavelet subbands coefficient divided by $\{N_{HL_F}(i), N_{LH_F}(i), N_{HH_F}(i)\}$, $\{N_{HL_T}(i), N_{LH_T}(i), N_{HH_T}(i)\}$, and $\{N_{HL_B}(i), N_{LH_B}(i), N_{HH_B}(i)\}$, which stand for the numbers of the wavelet subbands coefficient of flat area, texture area, and blocking artifact, respectively.

We calculate flat area threshold F_t in Eq. (14), entropy threshold ε_t in Eq. (15), and blocking artifact threshold B_t in Eq. (16) by using probability density function (PDF) [63] as follows:

$$f(\rho) = \frac{\beta^\gamma}{\Gamma(\gamma)}\rho^{\gamma-1}e^{-\beta\rho}; \rho > 0, \beta > 0, \gamma > 0 \tag{12}$$

where the observed satellite image ρ has a certain procedure consisting of γ independent steps, and each step takes the number of exponential Gamma distributions β per unit time. A complete Gamma function $\Gamma(\gamma)$ is defined as follows:

$$\Gamma(\gamma) = \int_0^\infty \beta^\gamma \rho^{\gamma-1} e^{-\beta\rho} d\rho \tag{13}$$

Finally, to determine F_t, ε_t, and B_t in soft-threshold values of flat area, texture area, and blocking artifact in Eqs. (14)–(16), we use Eqs. (12) and (13). By using Eqs. (12) and (13), we got the distribution signals of $F_t = 0.25$, $\varepsilon_t = 0.26$, and $B_t = 0.29$. Then, we define range values $0.23 \leqslant F_t < \varepsilon_t < B_t \leqslant 0.31$ for the wavelet coefficient subband $C_{subband}(r, s)$ in the coordinate *(r,s)*.

The soft-threshold $\delta\tau$ in every decomposition level i-th of a satellite image can be defined as Eq. (17).

$$\delta\tau(i) = \begin{cases} \delta F(i); & \rho > 0, \beta > 0, \gamma > 0, 0.23 \leqslant F_t \leqslant 0.25 \\ \delta T(i); & \rho > 0, \beta > 0, \gamma > 0, 0.26 \leqslant \varepsilon_t \leqslant 0.28 \\ \delta B(i); & \rho > 0, \beta > 0, \gamma > 0, 0.29 \leqslant B_t \leqslant 0.31 \end{cases} \tag{17}$$

3.1.2 Adaptive Wavelet Direction

Here, we propose 2D adaptive wavelet direction for the kernel which consists of two stages. First stage, or a 1D wavelet, is applied to blocking artifact followed by a vertical direction to obtain the vertical high-pass or horizontal low-pass subband (*LH*). The second stage, or the other 1D wavelet, is applied to blocking artifact followed by a horizontal direction to obtain the horizontal high-pass (HL) and horizontal or the vertical high-pass (*HH*). Each high-pass subband of the vertical or horizontal direction in wavelet can select a wavelet direction that detects blocking artifact. Blocking artifact can be represented in $d(x_0, x_i)$ coordinates of 2D distorted satellite image. We define an adaptive wavelet direction Ψ as follows:

$$\Psi_{m,\alpha,\delta\tau}(i) = \frac{1}{\sqrt{m_i}}\left[\frac{(x_0 \cos\alpha) + (x_i \sin\alpha)}{m_i \delta\tau}\right] \tag{18}$$

where m is scale parameter or width of kernel in wavelet and non-negative vector. Scale parameter m_i is represented as vectors m_1 and m_2. α is an orientation angle of kernel displacement. x_0 and x_i are the first and i-th positions of kernel in wavelet direction, respectively, as shown in Fig. 2.

Scale parameter m is determined by ridglet. Since the ridglet is constant along the ridge line $x_0 \cos\alpha + x_i \sin\alpha = 0$ and it is a wavelet along the orthogonal direction. Then, for small scale parameter m which is identical with m_2, ridglet is quite clear and localized along the ridge line. For larger

$$\delta F(i) = \begin{cases} \delta HL_F(i) = F_t - \left(\left(\left|\left|\sum_{|C_{HL_{F_t}(i)}(r,s)| \leqslant F_t} C_{HL_{F_t}(i)}(r,s)\right| - \left|\sum_{|C_{HL_F(i)}(r,s)| \leqslant F_t} C_{HL_F(i)}(r,s)\right|\right|\right) / N_{HL_F(i)}\right) \\ \delta LH_F(i) = F_t - \left(\left(\left|\left|\sum_{|C_{LH_{F_t}(i)}(r,s)| \leqslant F_t} C_{LH_{F_t}(i)}(r,s)\right| - \left|\sum_{C_{LH_F(i)}(r,s)| \leqslant F_t} C_{LH_F(i)}(r,s)\right|\right|\right) / N_{LH_F(i)}\right) \\ \delta HH_F(i) = F_t - \left(\left(\left|\left|\sum_{|C_{HH_{F_t}(i)}(r,s)| \leqslant F_t} C_{HH_{F_t}(i)}(r,s)\right| - \left|\sum_{|C_{HH_F(i)}(r,s)| \leqslant F_t} C_{HH_F(i)}(r,s)\right|\right|\right) / N_{HH_F(i)}\right) \end{cases} \tag{14}$$

$$\delta T(i) = \begin{cases} \delta HL_T(i) = \varepsilon_t + \left(\left(\left|\left|\sum_{|C_{HL_\varepsilon(i)}(r,s)| > \varepsilon_t} C_{HL_\varepsilon(i)}(r,s)\right| - \left|\sum_{|C_{HL_{\varepsilon_t}(i)}(r,s)| > \varepsilon_t} C_{HL_{\varepsilon_t}(i)}(r,s)\right|\right|\right) / N_{HL_T(i)}\right) \\ \delta LH_T(i) = \varepsilon_t + \left(\left(\left|\left|\sum_{|C_{LH_\varepsilon(i)}(r,s)| > \varepsilon_t} C_{LH_\varepsilon(i)}(r,s)\right| - \left|\sum_{|C_{LH_{\varepsilon_t}(i)}(r,s)| > \varepsilon_t} C_{LH_{\varepsilon_t}(i)}(r,s)\right|\right|\right) / N_{LH_T(i)}\right) \\ \delta HH_T(i) = \varepsilon_t + \left(\left(\left|\left|\sum_{|C_{HH_\varepsilon(i)}(r,s)| > \varepsilon_t} C_{HH_\varepsilon(i)}(r,s)\right| - \left|\sum_{|C_{HH_{\varepsilon_t}(i)}(r,s)| > \varepsilon_t} C_{HH_{\varepsilon_t}(i)}(r,s)\right|\right|\right) / N_{HH_T(i)}\right) \end{cases} \tag{15}$$

$$\delta B(i) = \begin{cases} \delta HL_B(i) = B_t + \left(\left(\left|\left|\sum_{|C_{HL_{B_t}(i)}(r,s)| \geqslant B_t} C_{HL_{B_t}(i)}(r,s)\right| - \left|\sum_{|C_{HL_B(i)}(r,s)| \geqslant B_t} C_{HL_B(i)}(r,s)\right|\right|\right) / N_{HL_B(i)}\right) \\ \delta LH_B(i) = B_t + \left(\left(\left|\left|\sum_{|C_{LH_{B_t}(i)}(r,s)| \geqslant B_t} C_{LH_{B_t}(i)}(r,s)\right| - \left|\sum_{|C_{LH_B(i)}(r,s)| \geqslant B_t} C_{LH_B(i)}(r,s)\right|\right|\right) / N_{LH_B(i)}\right) \\ \delta HH_B(i) = B_t + \left(\left(\left|\left|\sum_{|C_{HH_{B_t}(i)}(r,s)| \geqslant B_t} C_{HH_{B_t}(i)}(r,s)\right| - \left|\sum_{|C_{HH_B(i)}(r,s)| \geqslant B_t} C_{HH_B(i)}(r,s)\right|\right|\right) / N_{HH_B(i)}\right) \end{cases} \tag{16}$$

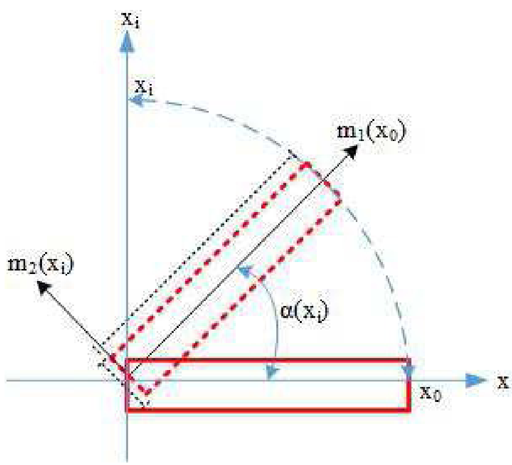


Fig. 2 Proposed the directional kernel of an adaptive wavelet

scale parameter m_1, ridglet can be a smooth function with smooth waves. Thus, different scale parameter m in ridglets can be used for directional approximations in smooth and straight-line edge area as we describe in Eqs. (19)–(22) as follows:

$$m_1 = x_0 \cos\alpha + x_i \sin\alpha \tag{19}$$

$$m_2 = -x_0 \sin\alpha + x_i \cos\alpha \tag{20}$$

$$x_0 = m_1 \cos\alpha - m_2 \sin\alpha \tag{21}$$

$$x_i = m_1 \sin\alpha + m_2 \cos\alpha \tag{22}$$

Figure 2 shows an adaptive wavelet which consists of two steps. The first step is a rotation of the window size function $w_m(x)$. The second step is the local polynomial approximation (LPA) design of the kernel with scale parameter m as coordinate systems as we describe in Eqs. (19)–(22). Location of distorted pixels or blocking artifacts are inside of the rotation angle α. The LPA in the variables m_1 and m_2 is used in order to obtain the accurate polynomial kernel for any discrete after rotation. The rotated kernels are reproduced with respect to the signals polynomial and also satisfy to the corresponding vanishing moment conditions of scale parameter m.

3.1.3 Adaptive Anisotropic Filter

We propose an adaptive anisotropic filter which uses an anisotropic measure of distorted satellite image level to control the shape of the kernel. The kernel k is applied at each pixel of distorted satellite image d with distorted level σ in coordinate $(x_{0,d}, x_{i,d})$ which is defined as follows:

$$k(x_{0,d}, x_{i,d}) = w_m(x)\alpha(x_{i,d} - x_{0,d}) * \left(\exp\left(-\delta\tau \left(\frac{((x_{i,d} - x_{0,d})m_1)^2}{\sigma^2{}_{1,d}(x_{0,d})} + \frac{((x_{i,d} - x_{0,d})m_2)^2}{\sigma^2{}_{2,d}(x_{0,d})}\right)\right)\right) \tag{23}$$

where the window size function $w_m(x)$ of neighborhood x in Eq. (23) is identical to that in Eq. (24) as follows:

$$w_m(x) = \frac{\frac{x}{m}}{m} = \frac{x}{m^2} \tag{24}$$

The symbol $\alpha(x_{i,d} - x_{0,d})$ in Eq. (21) represents a positive direction and the kernel rotation function where the condition $\alpha(x_i) = 1$ if $|x_i| \leqslant m_1$, and m_1 is the maximum scale parameter m. Distorted satellite image parameters of $\sigma^2{}_{1,d}(x_{0,d})$ and $\sigma^2{}_{2,d}(x_{0,d})$ are used to control the shape of kernel $k(x_{0,d}, x_{i,d})$.

3.2 Reconstruction

3.2.1 Local Polynomial Approximation (LPA) Estimation

Our idea of LPA estimation for deblocking artifact is to fit a polynomial to distorted satellite image in the neighborhood x and use it to estimate the value of the blocking artifact at the considered point. The polynomial is developed locally, as a linear combination of the considered point and fitted using the weighted least squares (WLS) method. We define that the quantization noise in blocking artifact is the sum of undistorted satellite image and distorted satellite image which corresponds to standard deviation σ. The observed satellite image ρ is expressed as follows:

$$\rho(r, s) = u(r, s) + \sigma d(r, s) \tag{25}$$

where u is undistorted satellite image and d is distorted satellite image or noise component of satellite image. Particularly, distorted satellite image is blocking artifact. *(r, s)*-th is the pixel coordinate in the block of satellite images and noise σ. Equation (25) and Taylor's series in [59], [60] are used for an approximation of deblocking artifact function in distorted satellite image d and neighborhood x, as follows:

$$d(r, s) \approx d(x) + d_1(x)\left(\frac{(x - (r, s))}{1!}\right) + d_2(x)\left(\frac{(x - (r, s))^2}{2!}\right) + d_3(x)\left(\frac{(x - (r, s))^3}{3!}\right) + \ldots \tag{26}$$

where d is the function of distorted satellite image and d_1, d_2, and d_3 are the first, second, and third derivative functions of distorted satellite image, respectively. $d_{\acute{d}}$ is the $\acute{d}$-th of distorted satellite image d estimation which can be given in the kernel k of nonlinear filter as follows:

$$\hat{d}_{\acute{d}}(x) = \sum_n k(x)\rho_n; \quad \acute{d} = 1, 2, 3, \ldots, \infty \tag{27}$$

Thus, the directional LPA kernel $k_{m,\alpha}(x)$ is defined as $\hat{d}_{m,\alpha_i}(x, m) = (k_{m,\alpha_i} * \rho)(x, m)$. Kernel $k_{m,\alpha}(x)$ and the observed satellite image ρ are convolved (*) at direction $\alpha_i(x)$, neighborhood x, and scale parameter m. The derivative directional LPA kernel $k_{m,\alpha}$ of estimated scale parameter $m_{\acute{d}}$ in $\acute{d}$-th estimation can be defined as follows:

$$\hat{d}_{m_{\acute{d}},\alpha_i}(x, m_{\acute{d}}) = (k_{m_{\acute{d}},\alpha_i} * \rho)(x, m_{\acute{d}}); \quad \acute{d} = 1, 2, 3, \ldots, \infty \tag{28}$$

We also obtain estimated variance in n number of deblocking artifacts which is computed as follows:

$$\begin{aligned}&\sigma^2_{\hat{d}_{m_{\hat{d}},\alpha_i}}(i, m_{\hat{d}}) = \sigma^2 \sum_{i\in\rho^d} |k_{m_{\hat{d}}}(n)|^2;\\&\hat{d} = 1, 2, 3, \ldots, \infty\end{aligned} \tag{29}$$

where ρ^d is a set of i-th observed satellite images ρ which is ranked by distorted satellite image d. Thus, our derivative estimated variances of Eq. (29) do not depend on neighborhood x.

3.2.2 Optimal Intersection of Confidence Interval (OICI)

The adaptive scale parameter m of each distorted satellite image can be seen in Eqs. (6)–(7). The intervals and their intersections have calculated in Eq. (8).

Our OICI method finds scale parameter m by calculating the confidence intervals from the left or the right side only. However, the original ICI based method calculates them from both the left and the right sides. The same procedure is repeated for the left or the right side of our deblocking artifact in distorted satellite image pixel *d(r,s,m)*. The smaller values of the confidence interval soft-threshold $\delta\tau$ can increase variance σ in scale parameter m. On the other hand, the larger values of the confidence interval soft-threshold $\delta\tau$ reduce variance in scale parameter m. Our OICI method does not only calculate scale parameter for each deblocking artifact pixel like ICI method, but detects the appropriate time regions for each pixel as well. Deblocking artifact pixel in time *t(r,s)* of the confidence intervals is calculated by using our OICI method as follows:

$$U_i(r, s, m_i) = \hat{d}_m(r, s, m_i) + \frac{\sigma_{\hat{d}_m}}{2\delta\tau i^2}(r, s, m_i), \tag{30}$$

$$L_i(r, s, m_i) = \hat{d}_m(r, s, m_i) - \frac{\sigma_{\hat{d}_m}}{2\delta\tau i^2}(r, s, m_i) \tag{31}$$

Based on Eqs. (30) and (31), our adaptive scale parameter m is calculated from the first to all pixel values in the detected time region of deblocking artifact from the left or the right side. Our OICI size ratio is calculated as follows:

$$OICI = \left(\frac{minU_i - maxL_i}{U_i(x_i, m_i) - L_i(x_i, m_i)}\right)^i \tag{32}$$

Our OICI based deblocking artifact algorithms which start by considering distorted satellite image pixel value in time which is denoted as $\hat{d}_m = (r, s, m_i)$, where $i = 1, 2, \ldots, D$, and D is the number of deblocking artifacts. OICI calculates upper and lower confidence intervals of each i by using the procedure in Eq. (8). If $m_1 \neq D$, the procedure in Eq. (8) is repeated until $m_i = D$ as shown in Eq. (33).

$$\sum_{i=1}^{t(r,s)} m_i = D \tag{33}$$

where *t(r,s)* is the number of detected time regions for the deblocking artifact pixel. Our OICI algorithm is expressed as follows.

Algorithm 1 OICI

Initialized
$\hat{d}_m, \sigma, \delta\tau, x, D, oici$
Declare
i : *integer*
x : *integer* ▹ *discrete value of deblocking artifact*
D : *integer* ▹ *number of deblocking artifacts*
while $x \leqslant D$ **do**
    $i \leftarrow 1$
    $max_{Li} \leftarrow -\infty$
    $min_{Ui} \leftarrow +\infty$
    while $max_{Li} \leqslant min_{Ui}$ **do**
        $U_i(x) \rightarrow \hat{d}_m(x) + \frac{\sigma_{\hat{d}_m}}{2\delta\tau i^2}(x)$
        $L_i(x) \rightarrow \hat{d}_m(x) - \frac{\sigma_{\hat{d}_m}}{2\delta\tau i^2}(x)$
        $max_{Li} \rightarrow max\{L_i(x), max_{Li}\}$
        $min_{Ui} \rightarrow min\{U_i(x), min_{Ui}\}$
        $oici \rightarrow \left(\frac{min_{Ui} - max_{Li}}{U_i(x,m_i) - L_i(x,m_i)}\right)^i$
        $i \rightarrow i + 1$
    end while
    $\hat{d}(x) \rightarrow \hat{d}_m(x, m)$
    $x \rightarrow x + 1$
end while
return $\hat{d}(x)$

Our OICI reduces computational time for $1 \leqslant t(r, s) \leqslant D$ at *t(r,s)* time. Then, we have a directional estimated kernel of our OICI by using Eq. (28). Estimated variance of OICI is the adaptive scale which is calculated as follows:

$$\begin{aligned}&\sigma^2_{\hat{d}_{m_{\hat{d}},\alpha_i}}(x, m_{\hat{d}}) = \sigma^2 \sum_{i=1}^{m} k^2_{m_{\hat{d}},\alpha_i}(x, m_{\hat{d}});\\&\hat{d} = 1, 2, 3, \ldots, \infty\end{aligned} \tag{34}$$

The adaptive scale estimate in Eq. (34) is the combined estimates which are exploited to obtain the final estimated deblocking artifact in the fusion process.

3.2.3 Fusion of Deblocking Artifact

Estimated fusion is a final estimation of estimated deblocking artifact which is computed as follows:

$$\begin{aligned}&\hat{d}_{m_{\hat{d}},\alpha_i}(x, m_{\hat{d}}) = \left(k_{m_{\hat{d}},\alpha_i} * \lambda_i \hat{d}^{(m-1)}_{m_{\hat{d}},\alpha_i}\right)(x, m_{\hat{d}});\\&\hat{d} = 1, 2, 3, \ldots, \infty\end{aligned} \tag{35}$$

Finally, we calculate new estimated variance by using soft-threshold $\delta\tau$ in fusion of deblocking artifact as follows:

$$\begin{aligned}&\sigma^2_{\hat{d}_{m_{\hat{d}},\alpha_i}}(x, m_{\hat{d}}) = \delta\tau\left(k^2_{m_{\hat{d}},\alpha_i} * \sigma^2_{\hat{d}^{(m-1)}_{m_{\hat{d}},\alpha_i}}\right)(x, m_{\hat{d}});\\&\hat{d} = 1, 2, 3, \ldots, \infty\end{aligned} \tag{36}$$

Equation (36) is estimated fusion variance of OICI.

3.3 Integration of ODE

First, we use Euler and Runge-Kutta-4 (RK4) methods. They are a family of implicit and explicit iterative methods for the integration of an ODE. Second, we use shift and add methods which cover algorithms for exponential and logarithm as well as a CORDIC method for trigonometric functions. We propose $Euler++$ $(E++)$ and Runge-Kutta of order 4++ $(RK4{+}{+})$ methods which are combined with shift and add methods. In an ODE, y is a vector. $y(x) = [y_1(x), y_2(x), \ldots,$ and $y_m(x)]$ is function. $\mathbf{F}$ is a vector valued function of y and its derivatives in n order of differential equation described as follows:

$$y^{(n)} = \mathbf{F}(x, y, y', y'', \ldots, y^{(n-1)}) \tag{37}$$

or Eq. (38) can be written as follows:

$$y' = F(t, y) \tag{38}$$

where y' is the first derivative of y. y is a function of one variable t (one-dimensional) to real $\mathbb{R}^p$. $\mathbf{F}$ is a given function which is defined on a domain $\mathfrak{D}_F \subset \mathbb{R}x\mathbb{R}^p$ to $\mathbb{R}^p$. Usually, an initial condition is written as $y(t_0) = \eta \in \mathbb{R}^p$. The corresponding Cauchy problem [64] or initial value problem is as follows:

$$Cauchy \begin{cases} y' = F(t, y) \\ y(t_0) = \eta \end{cases} \tag{39}$$

Let interval factor $[\zeta; \theta] \subset \mathbb{R}$ with $t_0 \in [\zeta; \theta]$, a C^1 function $Y : t \in [\zeta; \theta] \longmapsto Y(t) \in \mathbb{R}^p$ is a solution if

$$\begin{cases} (t, Y(t)) \in \mathfrak{D}_F \quad \forall t \in [\zeta; \theta] \\ Y(t_0) = \eta \\ Y'(t) = F(t, Y(t)) \quad \forall t \in [\zeta; \theta] \end{cases} \text{are fulfilled.} \tag{40}$$

To integrate an ODE, numerical methods obtain approximate values of the solution at a set of points $t_0 < t_1 < \cdots < t_n < \cdots < t_N$. The approximate value y_n of $Y(t_n)$ is computed by using some of the values obtained from $Y(t_{n-1})$. Our explicit $E++$ and $RK4++$ methods are considered in the initial value problem $Y(t)$.

3.4 Explicit Euler and Runge-Kutta Methods

Explicit Euler method is a single step method of order 1 defined on $[t_0; \xi]$ by

$$\begin{cases} y_{n+1} = y_n + hF(t_n, y_n) \\ y(t_0) = \eta \end{cases} \tag{41}$$

where h is the size of every step, t is a set of points in n number of point, ξ is the last range a set of points, and N is number of step. We define $h = \frac{\xi - t_0}{N}$ and $t_n = t_0 + nh$.

Runge-Kutta method aims to achieve a given order evaluating any derivatives of F and involves the evaluation of F at intermediary points. Among these methods, RK4 is 4 orders.

$$\begin{aligned} \mathcal{K}_1 &= F(t_n, y_n) \\ \mathcal{K}_2 &= F(t_n + \tfrac{h}{2}, y_n + \tfrac{h}{2}\mathcal{K}_1) \\ \mathcal{K}_3 &= F(t_n + \tfrac{h}{2}, y_n + \tfrac{h}{2}\mathcal{K}_2) \\ \mathcal{K}_4 &= F(t_n + h, y_n + h\mathcal{K}_3) \\ y_{n+1} &= y_n + \tfrac{h}{6}(\mathcal{K}_1 + 2\mathcal{K}_2 + 2\mathcal{K}_3 + \mathcal{K}_4) \\ y_0 &= \eta \end{aligned} \tag{42}$$

We consider the computation of $Y(t)$ for any $t \in [t_0; \xi]$. $E++$ and $RK4++$ methods compute $Y(t_{\mathcal{K}})$ where $t_{\mathcal{K}}$ is an intermediary point used to reach t. We consider only the error $\mathcal{E} = y_n - Y(t_n)$ in order to choose h. For $E++$, h has to satisfy $h \leqslant \sqrt{\frac{2\mathcal{E}}{\|y''\|_\infty}}$. The maximum of $\|\cdot\|_\infty$ is $[t_0; \xi]$. For $RK4++$, h has to satisfy a more complex inequality $h \leqslant \sqrt[5]{\frac{2880\mathcal{E}}{49\|\frac{\partial^4 F(t,y(t))}{\partial t^4}\|_\infty}}$. The sums of h for a given precision $\mathcal{E} = 2^{-24}$ correspond to the single precision (32 bits) and for a given precision $\mathcal{E} = 2^{-53}$ correspond to the double precision (64 bits) of each elementary function in our explicit $E++$ and $RK4++$ methods.

4. Experimental Results

We used 200 reference distorted satellite images which had blocking artifacts and each size was $32x32$, $64x64$, $128x128$, $256x256$, $512x512$, and $1024x1024$ pixels [65] in our experiment.

We demonstrate deblocking artifact algorithms performance five blocking artifact satellite images to represent 200 distorted satellite images [65]. The size of each satellite image is 512x512 pixels. Deblocking artifacts show that blocking artifacts were removed. The estimated deblocking artifacts show the resulting of deblocking artifacts which are estimated and reconstructed by our LPA and OICI. We calculate our deblocking artifact and estimated deblocking artifact performance by using root-mean-square error (RMSE), signal-to-noise ratio (SNR), peak signal-to-noise ratio (PSNR), improvement in signal-to-noise ratio (ISNR), mean absolute error (MAE), and maximum MAE (MaxMAE).

We compare our proposed algorithms to conventional methods [14], [28] which are no estimation nor adaptation, and also to non-conventional methods [18], [19], [30], [59], [64] which use adapative filtering and estimation. We use non-conventional methods of an adaptive soft-threshold which deals for deblocking artifact. Our OICI is adaptable to a crucial element of nonlinear filtering in reconstruction and estimation. The other non-conventional methods [18], [19], [30], [59], [64] have left larger distorted effect and also have removed finer signals of estimated deblocking artifact in reconstruction and estimation.

In Fig. 3, we show blocking artifacts which consist of detected blocking artifacts (a.1-e.1) denoted by red regions, deblocking artifacts (a.2-e.2) denoted by yellow regions, and final estimated deblocking artifacts (a.3-e.3) denoted by green regions. Based on Fig. 3 (a.2-e.2), we calculate SNR,

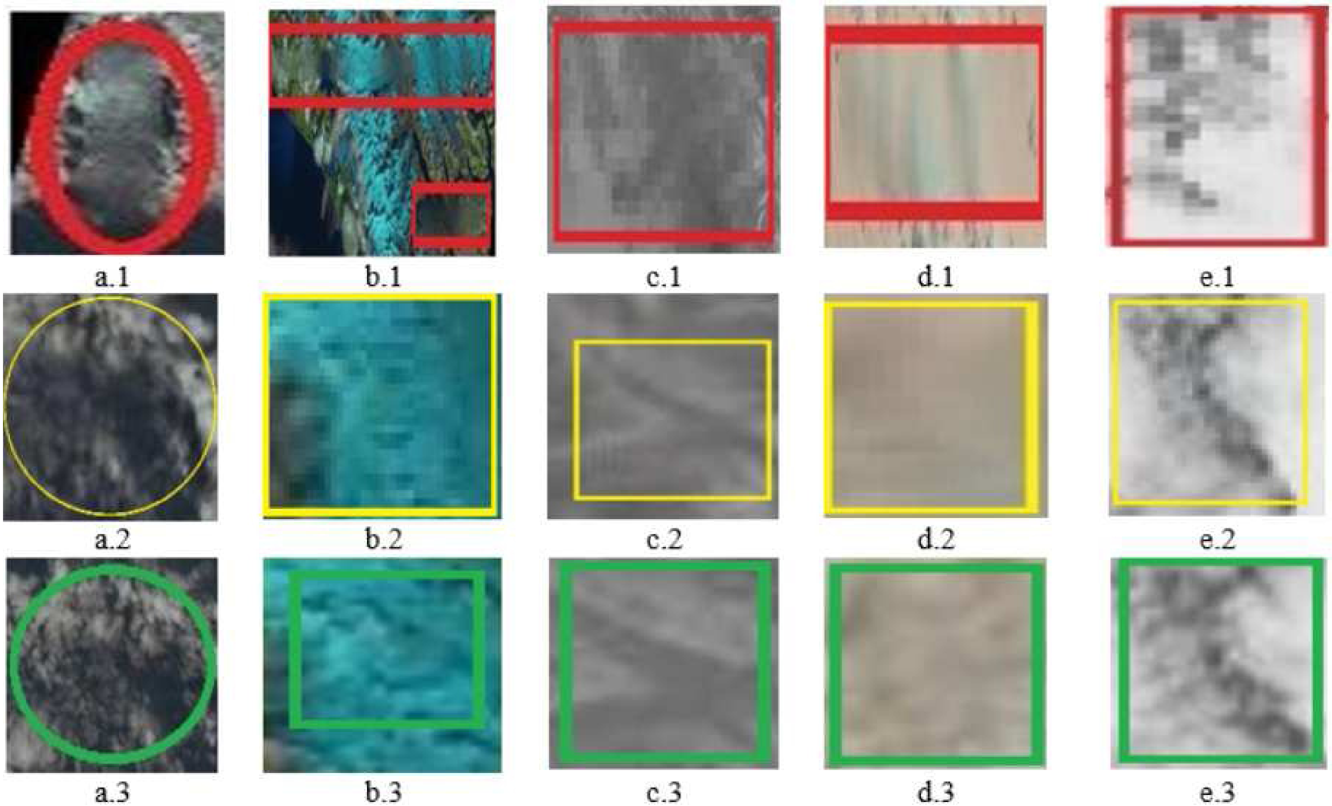


Fig. 3 Proposed deblocking artifact and estimated deblocking artifact satellite image results

Table 2 SNR (dB) and PSNR (dB) values of deblocking artifact

Method	Fig. 3.a.2		Fig. 3.b.2		Fig. 3.c.2		Fig. 3.d.2		Fig. 3.e.2	
	SNR	PSNR	SNR	PSNR	SNR	PSNR	SNR	PSNR	SNR	PSNR
Proposed Method	14.2999	28.9821	15.3515	26.5763	21.8369	22.2386	19.7635	26.9717	18.5423	27.2805
IWPOCS [14]	9.1519	18.5485	9.8250	17.0088	13.9756	14.2327	12.6486	17.2619	11.8671	17.4595
Overcomplete Wavelet [28]	10.4389	21.1569	11.2066	19.4007	15.9409	16.2342	14.4274	19.6893	13.5359	19.9148
LET [30]	9.7239	19.7078	10.4390	18.0719	14.8491	15.1222	13.4392	18.3408	12.6088	18.5507
LPA-ICI [18], [19]	11.4399	23.1857	12.2812	21.2610	20.3047	24.0417	15.8108	21.5774	14.8338	21.8244
SDN [59]	10.3288	20.9338	11.0884	19.1961	15.7728	16.0629	14.2752	19.4817	13.3931	19.7047
MWD [66]	10.8822	22.0554	11.6825	20.2246	16.6179	16.9236	15.0400	20.5255	14.1107	20.7605

Table 3 MSE and standard deviation (σ) values of deblocking artifact

Method	Fig. 3.a.2		Fig. 3.b.2		Fig. 3.c.2		Fig. 3.d.2		Fig. 3.e.2	
	MSE	σ	MSE	σ	MSE	σ	MSE	σ	MSE	σ
Proposed Method	82.2005	798.0025	143.0382	539.0027	64.2490	231.4716	130.5891	358.8028	121.6279	409.0464
IWPOCS [14]	109.3267	1061.3433	190.2408	716.8736	85.4512	307.8573	173.6835	477.2077	161.7651	544.0318
Overcomplete Wavelet [28]	100.2846	973.5630	174.5066	657.5833	78.3838	282.3954	159.3187	437.7394	148.3860	499.0367
LET [30]	113.4367	1101.2434	197.3927	743.8237	88.6636	319.4309	180.2130	495.1479	167.8465	564.4841
LPA-ICI [18], [19]	131.5208	1276.8039	228.8611	862.4043	102.7984	370.3546	208.9426	574.0845	194.6046	654.4743
SDN [59]	95.3526	925.6828	165.9243	625.2431	74.5288	268.5071	151.4834	416.2113	141.0884	474.4939
MWD [66]	99.4626	965.5830	173.0762	652.1933	77.7413	280.0807	158.0128	434.1514	147.1698	494.9462

PSNR, MSE, and standard deviations σ which start from deblocking artifact level as we show in Tables 2 and 3. Final estimated deblocking artifact evaluation of Fig. 3 (a.3-e.3) are shown in Tables 4–8. Tables 2 and 3 show that our proposed deblocking artifact algorithms have higher SNR and PSNR values and lower MSE and standard deviation values than other methods.

Our proposed deblocking artifact algorithms have finer signal of the deblocking artifacts. Table 4 shows SNR (dB) and PSNR (dB) values which have a significant increasing SNR and the PSNR values compared to our deblocking artifact in Table 2 and also other methods in Table 4. Table 5 shows MSE and standard deviation σ values of our final estimated deblocking artifact algorithms Fig. 3 (a.3-e.3) which have decreased MSE values compared to our deblocking artifact in Table 3 and also other methods in Table 5. We

Table 4 SNR (dB) and PSNR (dB) values of estimated deblocking artifact

Method	Fig. 3.a.3		Fig. 3.b.3		Fig. 3.c.3		Fig. 3.d.3		Fig. 3.e.3	
	SNR	PSNR	SNR	PSNR	SNR	PSNR	SNR	PSNR	SNR	PSNR
Proposed Method	53.1913	54.1965	48.4288	49.4319	54.5921	55.5964	49.7015	50.7068	49.9274	50.9327
LET [30]	37.2339	37.9376	33.9001	34.6023	38.2145	38.9175	34.7911	35.4948	34.9492	35.6529
LPA-ICI [18], [19]	42.5530	43.3572	38.7430	39.5455	43.6737	44.4771	39.7612	40.5654	39.9419	40.7462
SDN [59]	21.6905	26.1660	23.2856	22.3174	33.1229	18.0760	29.9779	21.4239	28.1255	19.9181
MWD [66]	13.0176	23.1878	13.7260	20.5367	20.1431	20.9758	22.3635	23.9503	14.4650	21.8909

Table 5 MSE and standard deviation (σ) values of estimated deblocking artifact

Method	Fig. 3.a.3		Fig. 3.b.3		Fig. 3.c.3		Fig. 3.d.3		Fig. 3.e.3	
	MSE	σ	MSE	σ	MSE	σ	MSE	σ	MSE	σ
Proposed Method	1.1131	9.0986	1.8255	5.9661	0.4428	1.8828	0.9526	3.0026	1.5519	4.5271
LET [30]	80.9257	785.6270	140.8200	530.6439	63.2526	227.8820	128.5639	353.2385	119.7417	402.7030
LPA-ICI [18], [19]	41.1003	638.4020	71.5191	431.2022	32.1245	185.1773	65.2946	287.0422	60.8140	327.2372
SDN [59]	40.5535	393.6929	70.5676	265.9159	31.6971	114.1961	64.4259	177.0146	60.0049	201.8022
MWD [66]	68.2413	662.4865	118.7476	447.4698	53.3383	192.1634	108.4126	297.8713	100.9732	339.5826

Table 6 RMSE values of estimated deblocking artifact

Method	Fig. 3.a.3	Fig. 3.b.3	Fig. 3.c.3	Fig. 3.d.3	Fig. 3.e.3
Proposed Method	1.0550	1.3511	0.6654	0.9760	1.2457
LET [30]	8.9959	11.8668	7.9532	11.3386	10.9427
LPA-ICI [18], [19]	6.4109	8.4569	5.6678	8.0805	7.7983
SDN [59]	6.3682	8.4005	5.6300	8.0266	7.7463
MWD [66]	8.2608	10.8971	7.3033	10.4121	10.0485

Table 7 MAE and MaxMAE values of estimated deblocking artifact

Method	Fig. 3.a.3		Fig. 3.b.3		Fig. 3.c.3		Fig. 3.d.3		Fig. 3.e.3	
	MAE	MaxMAE	MAE	MaxMAE	MAE	MaxMAE	MAE	MaxMAE	MAE	MaxMAE
Proposed Method	0.0735	2.1337	0.1075	2.1381	0.0506	1.6801	0.0691	1.2481	0.1033	2.1568
LET [30]	2.1317	61.8764	3.1180	62.0035	1.4683	48.7234	2.0037	36.1942	2.9960	62.5484
LPA-ICI [18], [19]	1.9847	57.6091	2.9029	57.7274	1.3670	45.3631	1.8655	33.6981	2.7893	58.2348
SDN [59]	21.6160	153.4572	34.0845	214.9195	21.5847	178.0194	28.0893	132.5318	34.1891	102.8781
MWD [66]	25.2187	168.0334	39.7653	239.7394	25.1822	181.0226	32.7708	142.9538	39.2581	124.4252

Table 8 Computational time (CT) in second and ISNR (dB) values of estimated deblocking artifact

Method	Fig. 3.a.3		Fig. 3.b.3		Fig. 3.c.3		Fig. 3.d.3		Fig. 3.e.3	
	CT	ISNR	CT	ISNR	CT	ISNR	CT	ISNR	CT	ISNR
Proposed Method	0.5767	38.8914	0.5948	33.0773	0.5593	32.7552	0.5525	29.9380	0.5773	31.3851
LET [30]	42.8919	30.3699	44.2291	26.5314	41.6058	27.7328	41.1003	25.3046	42.9326	26.0489
LPA-ICI [18], [19]	42.6773	31.1131	44.0145	26.4618	41.3912	23.3690	40.8857	23.9504	42.7180	25.1081
SDN [59]	43.9897	11.3617	45.3269	12.1972	42.7036	17.3501	42.1981	15.7027	44.0304	14.7324
MWD [66]	48.1999	2.1354	49.5371	2.0435	46.9138	3.5252	46.4083	7.3235	48.2406	0.3543

achieved 98.65% (Fig. 3.a.3), 98.72% (Fig. 3.b.3), 99.31% (Fig. 3.c.35), 99.27% (Fig. 3.d.3), and 98.72% (Fig. 3.e.3) MSE improvement. Thus, Tables 4 and 5 show that our estimated deblocking artifact algorithms have lower error and higher quality level than the other methods.

Table 6 shows RMSE values of our estimated deblocking artifact which compared to other methods. We achieved 88.36% (Fig. 3.a.3), 88.70% (Fig. 3.b.3), 91.70% (Fig. 3.c.3), 91.46% (Fig. 3.d.3), and 88.70% (Fig. 3.e.3) RMSE improvement. Thus, our proposed algorithms have lower RMSE values than the other methods. This means that our proposed algorithms have lower error than other methods. Table 7 shows MAE and MaxMAE values which are close or equal to 0 compared to the other methods. Our proposed algorithms have lower error in mean values. We achieved 99.27% (Fig. 3.a.3), 99.89% (Fig. 3.b.3), 99.95% (Fig. 3.c.3), 99.93% (Fig. 3.d.3), and 99.90% (Fig. 3.e.3) of MAE improvement. In other words, our proposed algorithms achieved higher quality of estimated deblocking artifact.

We achieve improvement over 90% of lower MSEs and standard deviations σ of deblocking artifact as shown in Table 3 compared with both conventional [14], [28] and non-conventional methods [18], [19], [30], [59], [64]. Our deblocking artifact outperform other methods which have many errors in classifying and removing of blocking

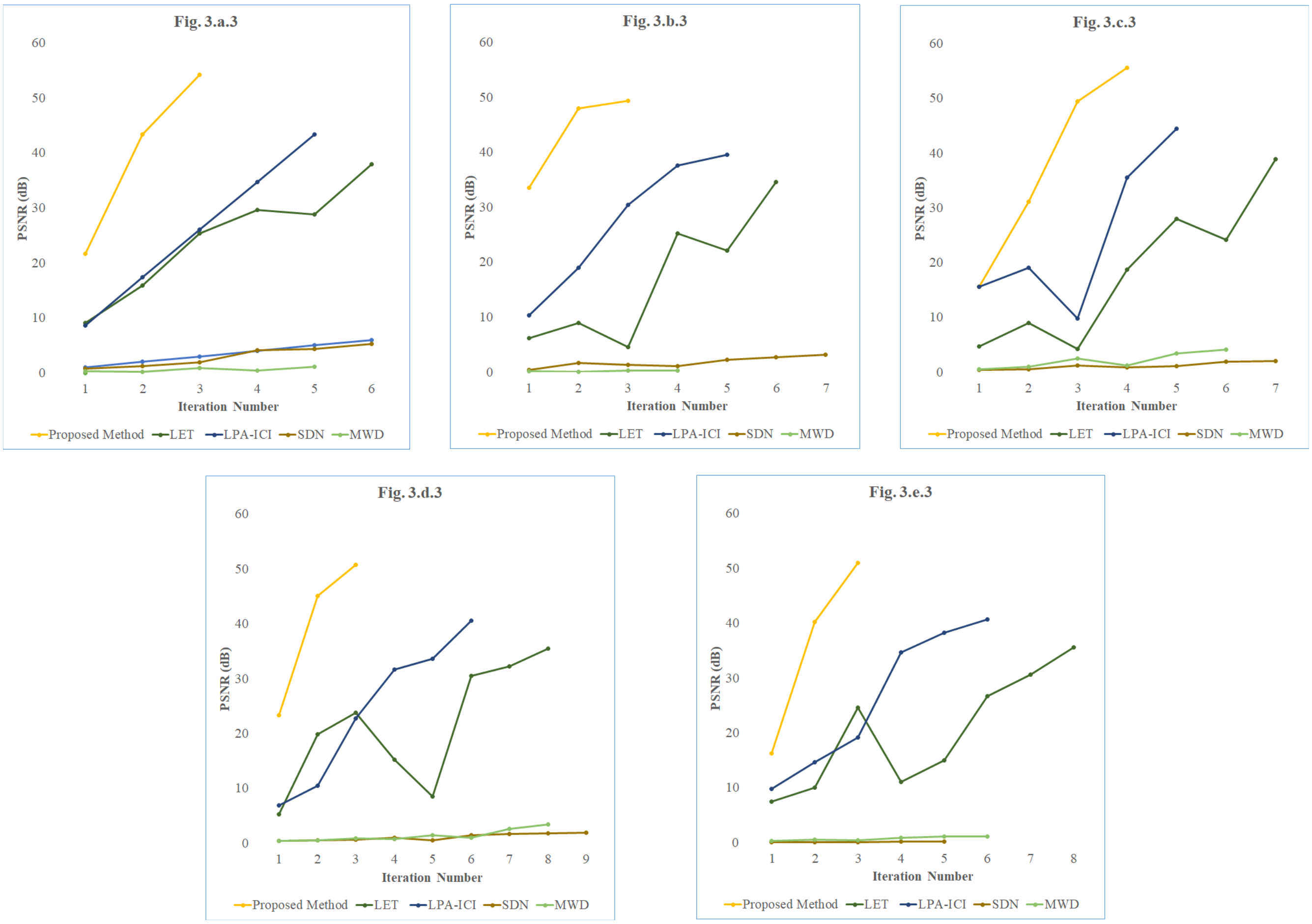


Fig. 4 Iterations of estimated deblocking artifact and PSNR

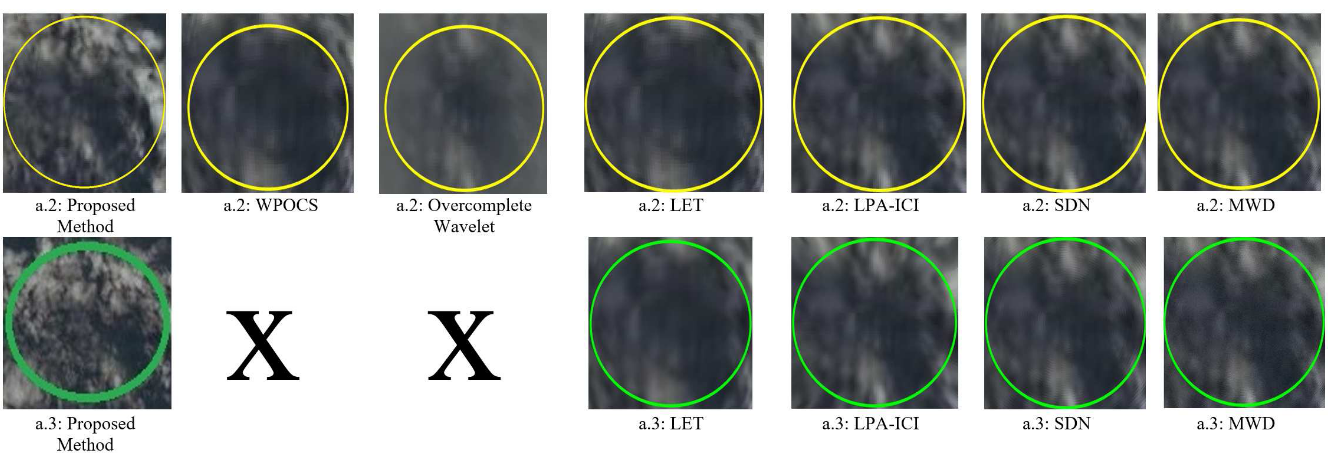


Fig. 5 Visual comparison of deblocking artifact (a.2) and estimated deblocking artifact (a.3)

artifact. Our proposed method successfully outperforms in terms of reducing MSEs, standard deviations, RMSE values, MAE and MaxMAE values of estimated deblocking artifact as shown in Table 5, Table 6, and Table 7, respectively, compared to other non-conventional methods [18], [19], [30], [59], [64]. We use non-conventional methods of an adaptive soft-threshold which deals for deblocking artifact in various satellite images, both undistorted and distorted satellite images.

Furthermore, computational times of estimated deblocking artifacts are shown in Table 8. We used a personal computer which had the specifications of Processor Intel

Table 9 The values of I correspond to the single and double floating-point numbers for various elementary functions by using $E++$ and $RK4++$.

Method	Function	Fig. 3.a.3				Fig. 3.b.3				Fig. 3.c.3				Fig. 3.d.3				Fig. 3.e.3			
		Euler++		$RK4++$		Euler++		$RK4++$		Euler++		$RK4++$		Euler++		$RK4++$		Euler++		$RK4++$	
		I_{E++}	I_{E++}	I_{RK4++}	I_{RK4++}	I_{E++}	I_{E++}	I_{RK4++}	I_{RK4++}	I_{E++}	I_{E++}	I_{RK4++}	I_{RK4++}	I_{E++}	I_{E++}	I_{RK4++}	I_{RK4++}	I_{E++}	I_{E++}	I_{RK4++}	I_{RK4++}
		Single	Double	Single	Double	Single	Double	Single	Double	Single	Double	Single	Double	Single	Double	Single	Double	Single	Double	Single	Double
Proposed Method	exp	4	8	2	4	4	9	2	4	4	8	3	6	3	7	3	6	4	9	2	4
	ln	5	9	3	5	4	7	2	4	3	5	2	4	4	8	2	4	3	6	2	3
	$\begin{pmatrix}\sin t\\ \cos t\end{pmatrix}$	4	7	3	5	3	6	2	4	3	5	2	4	4	6	3	5	2	4	3	5
	arctan	5	9	3	5	3	6	2	4	4	8	3	5	3	6	4	7	3	5	2	3
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$	5	9	2	3	4	7	3	6	5	10	4	8	4	8	5	9	4	8	3	6
	arg tanh	5	9	3	6	4	7	3	5	4	7	4	8	3	5	4	7	3	5	2	4
LET [30]	exp	18	36	8	16	16	34	8	18	16	32	10	18	18	34	10	18	14	30	6	14
	ln	16	32	6	12	14	28	6	12	14	30	8	16	16	30	8	14	12	26	6	12
	$\begin{pmatrix}\sin t\\ \cos t\end{pmatrix}$	16	32	6	12	12	24	6	12	12	24	8	16	14	28	8	16	10	20	6	12
	arctan	16	28	6	10	12	22	6	12	10	18	10	18	12	24	10	9	10	20	8	14
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$	18	34	8	16	14	28	8	16	12	24	12	22	14	26	12	24	12	22	10	20
	arg tanh	20	35	10	18	16	30	10	18	14	26	14	28	16	30	14	26	10	20	8	16
LPA-ICI [18], [19]	exp	16	32	8	14	14	30	7	14	13	28	9	18	16	32	10	17	10	20	6	12
	ln	14	28	5	11	12	24	5	11	13	28	7	15	15	29	7	13	11	23	6	11
	$\begin{pmatrix}\sin t\\ \cos t\end{pmatrix}$	15	31	6	11	11	23	5	9	11	22	7	15	13	26	7	15	9	19	6	11
	arctan	15	30	5	11	11	21	5	9	9	17	10	19	11	23	9	19	9	18	7	13
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$	17	32	7	15	13	26	7	15	11	22	11	21	13	25	11	23	11	20	9	19
	arg tanh	19	33	9	17	15	29	9	17	13	25	13	27	15	29	13	25	9	18	7	15
SDN [59]	exp	20	40	10	20	20	39	15	30	18	36	13	26	21	40	12	24	16	30	9	18
	ln	18	35	8	15	17	33	8	16	16	32	10	20	18	36	10	19	14	28	7	14
	$\begin{pmatrix}\sin t\\ \cos t\end{pmatrix}$	18	34	8	16	15	30	8	15	13	25	10	18	17	32	9	17	12	23	7	14
	arctan	17	33	8	15	13	26	8	16	12	23	12	23	13	25	11	21	11	22	10	19
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$	19	38	10	19	15	29	9	18	14	28	13	26	15	30	14	28	14	26	12	23
	arg tanh	21	41	13	25	17	34	13	26	15	30	15	30	17	34	16	31	11	21	9	18
MWD [66]	exp	22	43	13	26	23	45	17	34	20	39	15	30	23	45	14	28	18	34	11	21
	ln	20	40	10	19	19	37	10	19	18	36	12	23	20	39	12	23	15	29	8	15
	$\begin{pmatrix}\sin t\\ \cos t\end{pmatrix}$	19	37	9	17	17	33	10	19	15	30	11	21	19	38	10	19	13	25	9	18
	arctan	19	37	9	16	14	27	10	19	14	27	13	25	15	29	13	25	14	26	11	21
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$	20	39	11	21	16	31	10	19	16	32	15	29	16	32	15	30	16	32	13	25
	arg tanh	22	42	15	29	18	35	14	28	16	32	16	31	18	36	17	34	12	24	10	19

Table 10 Elementary functions times (μ second) of $E++$ and $RK4++$ algorithms

Method	Function	Fig. 3.a.3						Fig. 3.b.3						Fig. 3.c.3						Fig. 3.d.3						Fig. 3.e.3					
		E++		RK4++		Shift~Add		E++		RK4++		Shift~Add		E++		RK4++		Shift~Add		E++		RK4++		Shift~Add		E++		RK4++		Shift~Add	
		S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D	S	D
Proposed	exp on [0;1.5]	7	12	4	5	33	66	5	10	2	4	15	30	4	7	2	3	16	28	7	11	4	6	22	36	7	11	4	6	14	26
	ln on [1;2]	8	14	5	7	24	42	6	9	3	5	17	31	6	11	3	5	18	35	8	14	4	7	25	47	9	15	4	7	17	33
	$\begin{pmatrix}\sin\\ \cos t\end{pmatrix}$ on t=[0; $\frac{\pi}{4}$]	10	14	5	9	32	47	7	12	4	8	26	49	7	12	4	7	28	48	8	12	5	6	26	50	8	14	5	8	26	51
	arctan on [0;1]	10	17	6	10	35	63	6	10	3	5	28	52	9	17	5	8	32	58	9	17	4	7	27	49	8	15	4	7	32	58
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$ on t=[0; 1]	9	14	4	6	32	60	6	9	4	7	31	62	8	16	4	8	31	57	7	12	5	8	41	66	7	14	3	5	33	59
	arg tanh on t=[0; 0.7]	7	12	3	6	42	63	7	14	4	8	36	67	7	11	3	6	30	54	7	14	4	7	38	53	6	12	4	7	36	61
LET [30]	exp on [0;1.5]	12	24	8	10	60	90	9	17	4	7	29	58	7	12	4	6	28	56	13	22	7	12	40	72	11	22	8	12	27	50
	ln on [1;2]	15	28	8	13	42	76	9	18	6	10	30	60	11	20	6	10	30	59	13	25	7	12	47	63	17	31	9	13	28	55
	$\begin{pmatrix}\sin\\ \cos t\end{pmatrix}$ on t=[0; $\frac{\pi}{4}$]	15	30	9	17	60	90	14	27	9	17	40	72	13	25	9	17	46	71	15	30	50	80	42	78	15	27	9	17	52	88
	arctan on [0;1]	19	37	11	20	64	95	17	32	10	16	44	80	16	32	10	19	50	92	17	34	59	92	45	90	18	34	10	18	54	93
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$ on t=[0; 1]	20	40	13	26	60	91	18	36	11	22	46	92	18	36	12	24	54	101	19	38	60	108	47	94	19	37	12	23	57	95
	arg tanh on t=[0; 0.7]	19	36	12	24	50	85	15	31	8	15	41	78	16	32	10	20	46	52	17	34	42	76	45	92	15	30	10	19	36	70
LPA-ICI [18], [19]	exp on [0;1.5]	10	20	7	14	56	87	7	14	3	6	25	50	6	12	5	10	24	48	11	22	6	12	37	74	9	18	7	13	24	46
	ln on [1;2]	13	26	7	14	38	72	7	14	6	12	27	54	9	17	5	10	26	52	12	23	6	11	40	72	15	30	8	16	25	47
	$\begin{pmatrix}\sin\\ \cos t\end{pmatrix}$ on t=[0; $\frac{\pi}{4}$]	12	24	8	16	54	88	10	20	8	16	35	70	12	24	8	16	41	80	14	28	46	74	32	60	13	26	8	15	45	76
	arctan on [0;1]	17	34	9	18	52	86	14	28	9	14	40	74	13	26	8	16	46	92	14	29	45	84	35	70	14	28	9	18	47	66
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$ on t=[0; 1]	19	37	12	24	45	90	17	34	10	20	42	82	16	32	11	22	51	98	15	30	53	94	42	84	15	30	10	20	37	72
	arg tanh on t=[0; 0.7]	17	34	10	20	43	79	12	24	7	13	35	69	14	28	8	16	37	48	15	30	34	68	41	81	13	24	9	17	31	60
SDN [59]	exp on [0;1.5]	15	30	10	19	63	103	11	20	6	12	32	64	9	18	6	12	30	59	17	34	8	16	45	87	14	28	9	17	31	59
	ln on [1;2]	17	34	9	18	44	88	10	20	7	14	32	62	13	25	8	15	33	64	15	30	8	16	49	97	18	36	11	22	32	64
	$\begin{pmatrix}\sin\\ \cos t\end{pmatrix}$ on t=[0; $\frac{\pi}{4}$]	19	38	11	22	62	98	15	30	10	20	41	82	14	28	10	20	47	89	16	32	53	97	45	89	17	32	12	24	56	98
	arctan on [0;1]	21	42	12	24	66	101	19	38	11	22	45	90	17	34	11	22	51	102	19	38	62	110	47	94	19	38	11	22	55	98
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$ on t=[0; 1]	22	44	14	28	61	94	20	40	12	24	47	95	21	42	13	26	55	107	20	40	61	110	48	96	20	40	13	26	59	101
	arg tanh on t=[0; 0.7]	23	46	15	30	51	96	17	34	9	17	42	84	17	34	11	22	47	63	18	36	44	88	46	92	17	33	11	21	39	72
MWD [66]	exp on [0;1.5]	17	34	11	22	64	109	12	24	7	14	33	66	10	20	7	14	32	64	18	36	9	17	47	92	15	30	10	20	32	64
	ln on [1;2]	18	36	11	22	45	90	12	24	8	15	32	62	14	28	9	17	34	68	16	32	9	17	50	99	19	38	13	26	34	66
	$\begin{pmatrix}\sin\\ \cos t\end{pmatrix}$ on t=[0; $\frac{\pi}{4}$]	21	41	13	25	63	110	16	32	11	21	43	85	15	30	11	22	49	97	18	36	54	106	46	91	19	38	14	28	57	111
	arctan on [0;1]	23	46	13	26	67	112	21	42	13	26	46	92	19	38	12	24	52	103	21	42	65	120	49	97	21	42	14	28	57	110
	$\begin{pmatrix}\sinh t\\ \cosh t\end{pmatrix}$ on t=[0; 1]	24	48	15	30	62	120	22	44	14	28	50	99	23	46	15	30	56	109	25	49	63	120	51	101	23	46	15	30	60	118
	arg tanh on t=[0; 0.7]	25	49	16	31	52	101	19	38	11	22	45	89	19	38	13	26	48	75	20	39	47	92	47	93	19	38	13	25	41	82

Core i7-2600K@3.40 GHz and 8 GB DDR3 RAM. Our proposed algorithms show smaller computational time than the other methods. We achieved 77.95% (Fig. 3.a.3), 77.83% (Fig. 3.b.3), 78.08% (Fig. 3.c.3), 78.12% (Fig. 3.d.3), and 77.94% (Fig. 3.e.3) computational time improvement compared to the other methods. Our OICI estimates scale parameter m from one way, from the left or right side kernel only. In contrast, original ICIs [17]–[19] and other methods [30], [59], [64] have a big number of iterations in estimation as shown in Fig. 5. Also, the ISNR values of our proposal are positive and higher values of SNR up to 30 dB than the other methods.

In Table 9, $Y(t)$ was computed with the required accuracy 2^{-n}. In each step $\mathcal{K}$ of the shift and add algorithms, the absolute error on t is $|t - t_{\mathcal{K}}| \leqslant 2^{-\mathcal{K}+1}$ with $t_{\mathcal{K}} = \sum_{j=0}^{\mathcal{K}} \mathfrak{D}_{\mathcal{K}}$ and the value computed at step $\mathcal{K}$, $y_{\mathcal{K}}$, was exactly equal to $Y(t_{\mathcal{K}})$. It was determined the step h of the numerical integration method corresponding to a method error less than 2^n. Let I be the smallest integer for $h \geqslant 2^{I+1}$. Shift and add algorithms are stopped after the $I - th$ iteration by giving t_I and y_I. If an iteration of the numerical integration method is performed by using a step $h^{'} = t - t_I$, the error is bounded by $|y_{I+1} - Y(t)| \leqslant 2^{-n}$.

We have determined the value of I which corresponded to the single (32 bits) and double (64 bits) IEEE floating-point numbers, for the various elementary functions when the numerical integration method is either explicit $E++$ or $RK4++$. The error has to be less than 2^{-24} for the single precision and less than 2^{-53} for the double precision. Table 9 shows our $E++$ and $RK4++$ algorithms which compute floating-point numbers for various elementary functions.

We have estimated the number of clock cycles. $E++$, $RK4++$, and shift and add methods were performed in one clock cycle, whereas a division was ten times longer. We observe that this is in good performance with the experimental times. Our $E++$ and $RK4++$ algorithms took over shift and add ones by a multiplicative factor between 1.9 and 3.6. We have estimated the number of clock cycles with a favorable configuration of Intel Core i7-2600K@3.40 GHz processor. It was performed a shift in one clock cycle, an addition or a multiplication in 5 clock cycles, a single precision division in 20 clock cycles and a double precision division in 30 clock cycles. For this processor, our $E++$ and $RK4++$ algorithms were reduced computational time of the shift and add ones. Furthermore, $RK4++$ algorithm is superior to $E++$ algorithm as described in Table 10.

5. Conclusions

We proposed new deblocking artifact algorithms by using adaptive soft-threshold anisotropic filter values in wavelet. Our deblocking artifact algorithms outperform other methods, both conventional and non-conventional methods. Our deblocking artifact algorithms are adaptable to different blocking artifact in distorted and undistorted satellite image. In reconstruction and estimation, we proposed OICI for estimated deblocking artifact. Our OICI improves MSE of estimated deblocking artifact up to 98%, RMSE up to 89%, and MAE up to 99%. Computational time was reduced up to 77.98% compared to the other methods.

We have accelerated shift and add algorithms by substituting some iterations $E++$ and $RK4++$ algorithms. These combine the advantages of the two kinds of methods, without suffering their drawbacks. Our $E++$ and $RK4++$ algorithms have performed less steps, but shift and add methods have performed more steps. We have compared the number of clock cycles required by each method. On a processor, our $E++$ and $RK4++$ algorithms have improved and reduced computational time in terms of the shift and add. $RK4++$ algorithm is superior to $E++$ algorithm.

References

[1] A.K. Jain, Fundamental of digital image processing, Englewood Cliffs, New Jersey: Prentice-Hall, 1989.
[2] K.R. Rao and P.C. Yip, The transform and data compression handbook, Boca Raton, Florida: CRC Press LLC, 2001.
[3] K. Sayood, Introduction to data compression, 4th edition, Elsevier Inc. and Morgan Kaufmann Press, 2012.
[4] V. Britanak, P.C. Yip, and K.R. Rao, Discrete cosine and sine transforms: general properties, fast algorithms and integer approximations, Elsevier Ltd., 2006.
[5] Risnandar and M. Aritsugi, "Detection of blocking artifact on satellite image and its new evaluator," The 13th International Conference on Signal Processing (ICSP2016), IEEE, pp.794–798, 2016.
[6] CCITT Recommendation/ISO/IEC International standard, Joint Photographic Expert Group, JPEG Technical Specification, 1992.
[7] D.S. Taubman and M.W. Marcellin, JPEG2000: image compression fundamentals, standards, and practice, Springer Science and Business Media, LLC, 2002.
[8] ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, ITU-T recommendation and international standard of joint video specification (ITU-T Rec. H.264|ISO/IEC 14496-10 AVC), 2003.
[9] M.C. Angelides and H. Agius, The Handbook of mpeg applications: standards in practice, United Kingdom: John Wiley & Sons Ltd., 2011.
[10] A.W.-C. Liew, H. Yan, and N.-F. Law, "POCS-based blocking artifacts suppression using a smoothness constraint set with explicit region modeling," IEEE Trans. on Circuits & Syst. for Video Technol., vol.15, no.6, pp.795–800, June 2005. doi: 10.1109/TCSVT.2005.848303.
[11] R. Heylen, M.A. Akhter, and P. Scheunders, "On using projection onto convex sets for solving the hyperspectral unmixing problem," IEEE Geosci. & Remote Sens. Letters, vol.10, no.6, pp.1522–1526, Nov. 2013. doi: 10.1109/lgrs.2013.2261276.
[12] X. Yang, Y. Zhang, D. Zhou, and R. Yang, "An improved iterative back projection algorithm based on ringing artifacts suppression," Elsevier, Neurocomputing, vol.162, pp.171–179, Aug. 2015. doi: 10.1016/j.neucom.2015.03.055.
[13] D. Iwai, K. Kodama, and K. Sato, "Reducing motion blur artifact of foveal projection for a dynamic focus-plus-context display," IEEE Trans. on Circuits & Syst. for Video Technol., vol.25, no.4, pp.547–556, April 2015. doi: 10.1109/TCSVT.2014.2352500.
[14] B. Wang, X. Chen, J. Li, and J. Cao, "An improved weighted projection onto convex sets method for seismic data interpolation and denoising," IEEE Journal of Selected Topics in Appl. Earth Observ. & Remote Sens., vol.9, no.1, pp.228–235, Jan. 2016. doi: 10.1109/JSTARS.2015.2496374.
[15] S.Q. Wang and J.H. Zhang, "Fast image inpainting using exponential-threshold POCS plus conjugate gradient," The Imaging Science Journal, vol.62, no.3, pp.161–170, June 2014.

[16] W.B. Wei, L. Tan, M.Q. Jia, and Z.K. Pan, "Normal vector projection method used for convex optimization of chan-vese model for image segmentation," Journal of Physics: Conference Series, vol.787, pp.1–9, May 2017. doi: 10.1088/1742-6596/787/1/012016.

[17] V. Katkovnik, H. Oktem, and K. Egiazarian, "Filtering heavy noised images using ICI rule for adaptive varying bandwidth," The 1999 International Symposium on Selection, Circuits and Systems (ISCAS99), IEEE, pp.195–198, 1999.

[18] Astola, "Adaptive window size image denoising based on ICI rule," International Conference on Acoustics, Speech, and Signal Processing (ICASSP'01), IEEE, pp.1869–1872, 2001.

[19] H. Oktem, V. Katkovnik, K. Egiazarin, and J. Astola, "Local adaptive transform based image denoising with varying window size," 2001 International Conference on Image Processing, IEEE, pp.273–276, 2001. doi: 10.1109/ICIP.2001.959006.

[20] M. Shirzaei, "A wavelet-based multitemporal DInSAR algorithm for monitoring ground surface motion," IEEE Geosci. & Remote Sens. Letters, vol.10, no.3, pp.456–460, May 2013. doi: 10.1109/LGRS.2012.2208935.

[21] A.E. Cetin and M. Tofighi, "Projection-based wavelet denoising," IEEE Signal Processing Magazine, vol.32, no.5, pp.120–124, Sept. 2015. doi: 10.1109/MSP.2015.2440051.

[22] S.G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," IEEE Trans. on Pattern Anal. & Mach. Intel., vol.11, no.7, pp.674–693, July 1989. doi: 10.1109/34.192463.

[23] S. Mallat, A wavelet tour of signal processing, second edition, California: Elsevier and Academic Press, 1999.

[24] Z.J. Xiang and P.J. Ramadge, "Edge-preserving image regularization based on morphological wavelets and dyadic trees," IEEE Trans. on Image Processing, vol.21, no.4, pp.1548–1560, April 2012. doi: 10.1109/TIP.2011.2181399.

[25] G.J. Tua and H. Karstoft, "Logarithmic dyadic wavelet transform with its applications in edge detection and reconstruction," Elsevier Applied Soft Computing Journal, vol.26, pp.193–201, Sept. 2014.

[26] T. Zhang, Q. Gao, and G. Tan, "Texture preserving image restoration with dyadic bounded mean oscillating constraints," IEEE Signal Processing Letters, vol.22, no.3, pp.322–326, March 2015. doi: 10.1109/LSP.2014.2359002.

[27] I. Bayram and I.W. Selesnick, "Frequency-domain design of overcomplete rational-dilation wavelet transforms," IEEE Trans. on Signal Processing, vol.57, no.8, pp.2957–2972, Aug. 2009. doi: 10.1109/TSP.2009.2020756.

[28] S.W. Myint, T. Zhu, and B. Zheng, "A novel image classification algorithm using overcomplete wavelet transforms," IEEE Geosci. & Remote Sens. Letters, vol.12, no.6, pp.1232–1236, June 2015. doi: 10.1109/LGRS.2015.2390133.

[29] A. Balavoine, C.J. Rozell, and J. Romberg, "Discrete and continuous-time soft-thresholding for dynamic signal recovery," IEEE Trans. Signal Process., vol.63, no.12, pp.1–16, June 2015.

[30] H. Pan and T. Blu, "An iterative linear expansion of thresholds for l1-based image restoration," IEEE Trans. on Image Processing, vol.22, no.9, pp.3715–3728, Sept. 2013. doi: 10.1109/TIP.2013.2270109.

[31] C. Jung, L. Jiao, H. Qi, and T. Sun, "Image deblocking via sparse representation," Elsevier, Signal Processing: Image Communication, vol.27, no.6, pp.663–677, July 2012. doi: 10.1016/j.image.2012.03.002.

[32] A. Rosenfeld and A.C. Kak, Digital picture processing, second edition, volume 2, New York: Academic Press Inc., 1982.

[33] A. Blake and A. Zisserman, Visual reconstruction, Massachussetts: The MIT Press, 1987.

[34] A.P. Korostelev and A.B. Tsybakov, Minimax theory of image reconstruction, New York: Springer-Verlag Inc., 1993.

[35] W.K. Pratt, Digital image processing, 4th edition, New Jersey: John Wiley & Sons, Inc., 2007.

[36] C. Xie, W. Zhong, and K. Mueller, "A Visual analytics approach for categorical joint distribution reconstruction from marginal projections," IEEE Trans. Vis. Comput. Graphics, vol.23, no.1, pp.51–60, Aug. 2016.

[37] P.D. Wendt, E.J. Coycle, and N.C. Gallagher, "Stack filters," IEEE Trans. Acoust., Speech, Signal Process., vol.34, no.4, pp.898–911, Sept. 1986.

[38] M.E. Buemi, J. Jacobo, and M. Mejail, "SAR image processing using adaptive stack filter," Elsevier, Pattern Recognition Letters, vol.31, no.4, pp.307–314, March 2010. doi: 10.1016/j.patrec.2009.02.008.

[39] M.E. Buemi, A.C. Frery, and H.S. Ramos, "Speckle reduction with adaptive stack filters," Elsevier, Pattern Recognition Letters, vol.36, pp.281–287, Jan. 2014. doi: 10.1016/j.patrec.2013.06.005.

[40] D. Dellamonica, P.J.S. Silva, C. Humes, N.S.T. Hirata, and J. Barrera, "An exact algorithm for optimal MAE stack filter design," IEEE Trans. on Image Processing, vol.16, no.2, pp.453–462, Feb. 2007. doi: 10.1109/TIP.2006.888358.

[41] G.R. Arce and R.E. Foster, "Detail-preserving ranked-order based filters for image processing," IEEE Trans. on Acoustics, Speech, Signal Processing, vol.37, no.1, pp.83–98, Jan. 1989. doi: 10.1109/29.17503.

[42] A. Fabijańska and D. Sankowski, "Noise adaptive switching median-based filter for impulse noise removal from extremely corrupted images," vol.5, no.5, pp.472–480, 2011.

[43] Y. Shen and K.E. Barner, "Fast adaptive optimization of weighted vector median filters," IEEE Trans. on Signal Processing, vol.54, no.7, pp.2497–2510, July 2006. doi: 10.1109/TSP.2006.874028.

[44] T.C. Aysal and K.E. Barner, "Quadratic weighted median filters for edge enhancement of noisy images," IEEE Trans. on Image Processing, vol.15, no.11, pp.3294–3310, Nov. 2006. doi: 10.1109/TIP.2006.882010.

[45] W. Fan, K. Wang, F. Cayre, and Z. Xiong, "Median filtered image quality enhancement and antiforensics via variational deconvolution," IEEE Trans. on Inf. Forensics & Security, vol.10, no.5, pp.1076–1091, Jan. 2015. doi: 10.1109/TIFS.2015.2398362.

[46] A.B. Hamza, P.L. Luque-Escamilla, J. Martinez-Aroza, and R. Román-Roldán, "Removing noise and preserving details with relaxed median filters," Springer, Journal of Mathematical Imaging and Vision, vol.11, no.2, pp.161–177, Oct. 1999. doi: 10.1023/A:1008395514426.

[47] D.L. Donoho and J.M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," Oxford Academic, Biometrika, vol.81, no.3, pp.425–455, Sept. 1994. doi: 10.2307/2337118.

[48] D.L. Donoho, "De-noising by soft-thresholding," IEEE Trans. on Inf. Theory, vol.41, no.3, pp.613–627, May 1995. doi: 10.1109/18.382009.

[49] F. Li, L. Pi, and T. Zeng, "Explicit coherence enhancing filter with spatial adaptive elliptical kernel," IEEE Signal Processing Letters, vol.19, no.9, pp.555–558, Sept. 2012. doi: 10.1109/LSP.2012.2206582.

[50] W. Zuo, D. Ren, D. Zhang, S. Gu, and L. Zhang, "Learning iteration-wise generalized shrinkagethresholding operators for blind deconvolution," IEEE Trans. on Image Processing, vol.25, no.4, pp.1751–1764, April 2016. doi: 10.1109/TIP.2016.2531905.

[51] O.V. Lepskii, "On a problem of adaptive estimation in Gaussian white noise," SIAM, Theory Probab. & Its Appl., vol.35, no.3, pp.454–466, 1991. doi: 10.1137/1135065.

[52] O.V. Lepski, "Asymptotic minimax adaptive estimation. II. Scheme without optimal adaptation: adaptive estimators," Theory Probability and Its Application, vol.37, no.3, pp.433–448, 1992.

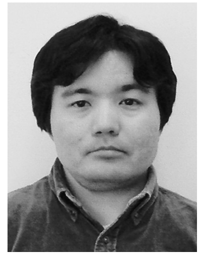
[53] O.V. Lepski, E. Mammen, and V.G. Spokoiny, "Optimal spatial adaptation to inhomogeneous smoothness: an approach based on kernel estimates with variable bandwidth selectors," JSTOR, The Annals of Statistics, vol.25, no.3, pp.929–947, June 1997.

[54] O.V. Lepski and V.G. Spokoiny, "Optimal pointwise adaptive methods in nonparametric estimation," AMS, The Annals of Statistics, vol.25, no.6, pp.2512–2546, 1997.

[55] G. Paul, J. Cardinale, and I.F. Sbalzarini, "Coupling image restora-

tion and segmentation: a generalized linear model/Bregman perspective," Springer, International Journal Computer Vision, vol.104, no.1, pp.69–93, 2013. doi: 10.1007/s11263-013-0615-2.
[56] A. Nemirovski, Topics in non-parametric statistics, Lecture notes in mathematics 1738, pp.85–277, Springer, N.Y., 2000.
[57] A. Barron, L. Birge, and P. Massart, "Risk bounds for model selection via penalization," Probability Theory and Relation Fields, vol.113, no.3, pp.301–413, Feb. 1999.
[58] J. Klemela and A.B. Tsybakov, "Sharp adaptive estimation of linear functionals," JSTOR, The Annals of Statistics, vol.29, no.6, pp.1567–1600, Dec. 2001.
[59] X. Liu, M. Tanaka, and M. Okutomi, "Practical signal-dependent noise parameter estimation from a single noisy image," IEEE Trans. Image Process., vol.23, no.10, pp.4361–4362, Oct. 2014.
[60] Y.-Y. Zhao, J.-G. Lin, X.-F. Huang, and H.-X. Wang, "Adaptive jump-preserving estimates in varying-coefficient models," Elsevier, Journal of Multivariate Analysis, vol.149, pp.65–80, July 2016. doi: 10.1016/j.jmva.2016.03.005.
[61] V. Katkovnik, "A new method for varying adaptive bandwidth selection," IEEE Trans. on Signal Processing, vol.47, no.9, pp.2567–2571, Sept. 1999. doi: 10.1109/78.782208.
[62] J.M. Muller, Elementary functions: algorithms and implementation, 2nd edition, Boston: Springer, 2006.
[63] M. Baron, Probability and statistics for computer scientists, 2nd Ed., USA: Taylor and Francis Group, LLC, 2014.
[64] Y. Choquet-Bruhat, "Beginnings of the Cauchy problem," arXiv:1410.3490v1, pp.1–19, Oct. 2014.
[65] TERA-MODIS NEA, Indonesian National Board for Disaster Management, Jan.-Dec., 2015.
[66] A. Ahrabian and D.P. Mandic, "A class of multivariate denoising algorithms based on synchrosqueezing," IEEE Trans. on Signal Processing, vol.63, no.9, pp.2196–2208, May 2015. doi: 10.1109/TSP.2015.2404307.

Risnandar was born in Sukabumi, Indonesia. He received two B.E. degrees from Dept. of Electrical Engineering, Institut Teknologi Bandung (ITB) and School of Informatics, STMIK Jabar, Indonesia, in 2006. His M.E. degrees are double degree program from Dept. of Computer Science, University College London (UCL), and School of Electrical Engineering and Informatics, ITB in 2010. From 2014 until now, he has been studying for his Ph.D degree at Dept. of Computer Science and Electrical Engineering, Kumamoto University, Japan. Since 2006, he has worked as full researcher at the Research Center for Informatics, Indonesian Institute of Sciences (LIPI), particularly in image processing and its analysis and also digital forensic research fields.



Masayoshi Aritsugi received his B.E. & D.E. degrees in computer science and communication engineering from Kyushu University, Japan, in 1991 and 1996, respectively. From 1996 to 2007, he was with the Department of Computer Science, Gunma University, Japan. Since 2007, he has been a Professor at Kumamoto University, Japan. His research interests include database systems and parallel/distributed data processing. He is a senior member of IPSJ and a member of ACM, IEEE, and DBSJ.