Journal of Mathematical Imaging and Vision (2021) 63:503–527
https://doi.org/10.1007/s10851-020-01004-0

Adaptive Periodic Noise Reduction in Digital Images Using Fuzzy Transform

Najmeh Alibabaie[1] · **AliMohammad Latif**[1]

Received: 26 October 2019 / Accepted: 20 November 2020 / Published online: 22 January 2021

Abstract
Periodic noise degrades the image quality by overlaying similar patterns. This noise appears as peaks in the image spectrum. In this research, a method based on fuzzy transform has been developed to identify and reduce the peaks adaptively. We convert the periodic noise removal task as image compression and a smoothing problem. We first utilize the direct and inverse fuzzy transform of the spectrum to detect periodic noise peaks. Second, we propose a fuzzy transform-based notch filter for spectral smoothing and separating the original image from the periodic noise components. This noise correction approach filters out a portion (given by fuzzy transform) of the noise component. Extensive experiments on both synthetic and non-synthetic noisy images have been carried out to validate the effectiveness and efficiency of the proposed algorithm. The simulation results demonstrate that the proposed method outperforms state of the art algorithms both visually and quantitatively.

Keywords Image noise removal · Fuzzy transform · Periodic noise · Stripping noise

1 Introduction

Image noise is any random variation of the pixel value which can be produced in digital image acquisition by sensors, scanner, and circuits of digital cameras [19]. The periodic noise, as one type of image noise, is generated by electrical or magnetic interference [39]. This noise is seen in some visual applications such as medicine [18], remote sensing [10], television [35], traffic control [25], and real-time applications. Periodic noise is detected as a repetitive pattern on the image, which degrades the image quality. Due to its abundance, periodic noise removal is one of the important issues in image processing.

Periodic noise not only sharply degrades the image quality in the visual effect but also risks its suitability for subsequent processing, e.g., image un-mixing and classification. The goal of our work is to remove periodic noise and improve image quality before subsequent interpretation.

✉ AliMohammad Latif
Alatif@yazd.ac.ir

Najmeh Alibabaie
n.alibabaie@stu.yazd.ac.ir

[1] Computer Engineering Department, Yazd University, Yazd, Iran

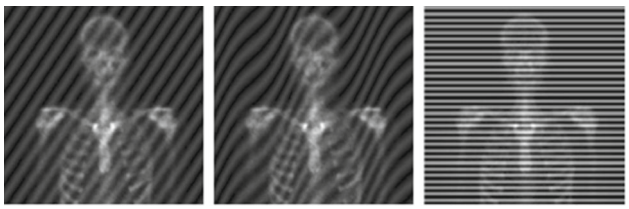
Periodic noise for an image of size M, N is spatially modeled through the summation of several sinusoidal functions (S) with different parameters [8] as follows:

$$n(x, y) = \sum_{i=1}^{S} A_i \sin\left(\left[\frac{2\pi u_{0i}(x + B_{xi})}{M}\right] + \left[\frac{2\pi v_{0i}(y + B_{yi})}{N}\right]\right) \tag{1}$$

where A_i denotes the amplitude, u_{0i} and v_{0i} are the ith sinusoidal frequency along the axes x and y and B_{xi} and B_{yi} are the phase displacements with respect to the origin and $S \in Z^+$.

Periodic noise is divided into global, local, and stripping categories. Global periodic noise, the noise parameters (amplitude, frequency, and phase) are spatial-independent. In local periodic noise, these parameters are spatial-dependent [33]. The stripping noise is a kind of periodic noise in which the peak is along the vertical or horizontal axis in the Fourier domain. Figure 1 shows three types of periodic noise.

Periodic noise reduction methods are divided into two general groups. The groups are spatial methods and spectral methods. Numerous spatial approaches have been proposed for the stripping noise due to it is spatially simpler than other periodic noise. Spatial domain methods operate directly on the image pixels for periodic noise suppression [17], while spectral methods are based on frequency domain filtering [7].



(a) Global **(b)** Local **(c)** Stripping

Fig. 1 X-Ray imaging of the human skeleton corrupted by periodic noise [21]

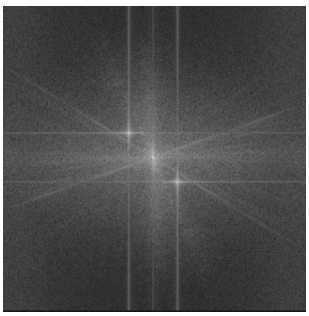


Fig. 2 Fourier transform of an image contaminated by periodic noise

From the methodological perspective, spatial methods are divided into two categories namely: optimization based methods and statistical based methods.

The main idea in statistical-based methods is to correct the distribution of the data to a reference distribution [32]. In the optimization based methods, denoising tasks are modeled as inverse problems to be tackled by regularization [9,14].

Since of scattering in the spatial domain, the efficiency of spatial methods decreases with global and local periodic noise [13]. Despite the spread of periodic noise in the spatial domain, it is concentrated in one or more adjacent coefficients in the frequency domain. Figure 2 shows the periodic noise at the frequency domain. Therefore, the frequency domain approaches are usually preferred.

In the frequency domain, noise reduction is performed in two steps. The first step is to find the location of the noisy frequencies. The main challenge in this step is to find out the location of the noise frequencies. The second step is how to repair the noisy frequencies and to get the restored image. Spectral approaches can be divided into subcategories according to the function they provide.

The first subcategory consists of algorithms that try to detect the peaks such as threshold-based methods [21,22,39], histogram analysis-based methods [8], clustering-based methods [12], spectral modeling methods [40,41], statistical-based methods [1–3,36–38].

The second subcategory consists of algorithms that try to repair noise frequencies. In the simplest case, noise frequencies were replaced by zero or divided by a constant [1,12]. To improve the repair operation, the noise component replaced by the minimum value [39] or median value [3,41] of its neighbors. In several studies, once the peaks detected, each peak together with its closest surroundings corrected by a windowed Gaussian notch filter [2,22,38]. Meanwhile, Gaussian notch filter not only rejected the central noise frequency but also suppressed its neighbors [5]. Gaussian-star filter and sinc-based filter were proposed due to the star-like appearance of noise peak [6,8].

Moreover, it should be noted that the Low-Frequency Region (LFR) should not be confused with periodic noise. In the simplest case, the LFR protected by protecting the DC component [1] or a square/circle in the center of the spectrum [2,3,36,38]. In addition, the region growing method [8] and connected component analysis method [12] were used to separate the low-frequency region.

This paper holds mainly two twofold. First, periodic noise peaks are found adaptively. The first hypothesis is that a compression method can be used for adaptive detection of the peaks because of the content loss is one part of the inherent property of lossy compression methods. If the content loss in a compression method takes place in edge pixels, it can also be used for adaptive peak detection. Second, the noise frequencies are repaired. The hypothesis is that filtering of the noise frequencies should be performed using a smoothing method.

In this research, the fuzzy transform (shortly, F-transform) is used to test hypotheses. Fuzzy transform is an approximation technique with a simple idea. The domain of a continuous function is partitioned using overlapping fuzzy sets. The theoretical preliminaries of the fuzzy transform have been described in [27], and its practical applications have been developed in time series analysis and forecasting [23], smoothing [15], edge detection [29], image fusion, image reconstruction [30], image compression [20], and image analysis [31,34]. The fuzzy transform is suitable for the purpose intended in this research because the content loss in the fuzzy transform take place on edges. Therefore, this method is used to detect periodic noise peaks. In addition, F-transform has been successfully applied in smoothing applications. Hence, a fuzzy notch filter can be presented according to the F-transform smoother while saving the computational time as well. Comparative study shows that our proposed algorithm outperforms both statistically and visually for various types of pure, quasi periodic noises including low, high, and multi-frequency periodic noise.

The main contribution of this paper is summarized as follows:

1. In this research, the problem of periodic noise reduction task is converted to a form of the compression, and smoothing problem and a fuzzy-based model is proposed for adaptive periodic noise removal.

2. The content loss has been used to construct a model for detection of noise frequencies which offers a new perspective on periodic noise reduction task.
3. Using the F-transform in a spectral method requires by taking into account for decreasing trend of the spectrum. In this research, F-transform has been used in the frequency domain by changing the partition space.
4. The noise peaks must be smoothed after detecting the noise frequencies. A fuzzy notch filter is proposed to repair the noise frequencies.

The structure of the paper is organized as follows. In Sect. 2, we briefly overview the main notions of the F-transform theory. Section 3 is devoted to a theoretical explanation of the efficiency of F-transform in the problem of periodic noise reduction. It goes into the details of the algorithm in Sect. 4. Comparison results between the proposed algorithm and some conventional methods are given and discussed in Sect. 5. The final section also presents the conclusion.

2 Preliminaries

F-transform is used as a tool in image processing applications and has two phases: direct and inverse. An image after normalization can be considered as a fuzzy matrix (relation). In the fuzzy partition, this matrix is divided into several submatrices known as blocks.

In the direct F-transform, each block is transformed to achieve the F-transform component. In fact, this component is the weighted mean value of the block that the weights given by the basic functions. In the inverse F-transform, a reconstructed block is obtained using basic functions resulting in an approximation of the original one [20]. The concepts and theories related to the fuzzy transform are explained below.

Fuzzy Partition

If the rectangle area $D^2 = [a, b]\times[c, d]$ is a common domain of all real-valued functions, the main idea consists of construction of two fuzzy partitions for intervals $[a, b]$ and $[c, d]$.

Definition 1 [24] If the interval $[a, b]$ is partitioned by basic functions $A_1, \ldots, A_n \subseteq [a, b]$, $n > 2$, and the interval $[c, d]$ is partitioned by basic functions $B_1, \ldots, B_m \subseteq [c, d]$, $m > 2$, the fuzzy partition of D^2 would be obtained by fuzzy Cartesian product $\{A_1, \ldots, A_n\}\times_{\odot}\{B_1, \ldots, B_m\}$, with respect to the product t-norm of these two fuzzy partitions. In case, both fuzzy partitions of particular axes are uniform, then the overall fuzzy partition is also uniform.

The direct and inverse Fuzzy transform of two variables function is a generalization of the case of one variable.

Direct Fuzzy Transform

Definition 2 [16] Let $A_1, \vdots A_n \subseteq [a, b]$ and $B_1, \vdots B_m \subseteq [c, d]$ be basic functions that form fuzzy partition of $[a, b]\times[c, d]$. In the discrete case, an original function $w : (p_i, q_j) \in ([a, b]\times[c, d]) \rightarrow R$ is known only at points (p_i, q_j), where $i = 1, \vdots N$, $j = 1, \vdots M$. In addition, the discrete F-transform $FT[w]$ of w is given by the following matrix of components:

$$FT[w] = (FT_{kl}[w])_{nm} = \begin{bmatrix} FT_{11}[w] & \ldots & FT_{1m}[w] \\ \vdots & \ddots & \vdots \\ FT_{n1}[w] & \ldots & FT_{nm}[w] \end{bmatrix}$$

where for $k = 1, \ldots, n, l = 1, \ldots, m$, the components of FT_{kl} are given by

$$FT_{kl} = \frac{\sum_{j=1}^{N}\sum_{i=1}^{M} w(p_i, q_j)A_k(p_i)B_l(q_j)}{\sum_{j=1}^{N}\sum_{i=1}^{M} A_k(p_i)B_l(q_j)} \tag{2}$$

Inverse Fuzzy Transform

Definition 3 [24] Let $A_1, \ldots, A_n \subseteq [a, b]$ and $B_1, \ldots, B_m \subseteq [c, d]$ be fuzzy partitions. Let $FT[w]$ be the F-transform of $w : [a, b]\times[c, d] \rightarrow R$ with respect to the basic functions $A_1, \ldots, A_n$ and $B_1, \ldots, B_m$. Then, inverse fuzzy transform of $\tilde{w}$ would be defined as:

$$\tilde{w}(p_i, q_j) = \sum_{k=1}^{n}\sum_{l=1}^{m} FT_{kl}A_k(p_i)B_l(q_j) \tag{3}$$

3 Theoretical Framework

F-transform is a method that is successfully used in image compression. Lossy image compression methods tend to introduce distortions such as general loss of sharpness and oscillations around high-contrast edges, blocking structure and loss of color details. The considered distortion in F-transform takes place at edges. This property to detect the noise frequencies adaptively in this research as used, meaning that the noise frequency, like edge pixels, has a rapid change in the intensity. So, if the image spectrum undergoes to this transform, the periodic noise peaks will be detected.

In this section, the main goal is to show that the F-transform enables a extraction of the main image frequencies and the sub-component forming the noise component by F -transform can be either completely removed or significantly reduced. The explanation will be given on the example of a discrete function that corresponds to the function w. Let w be represented by the discrete function $w : P \rightarrow R$ of two

variables where $P = (i, j)|i = 1, \ldots, N, j = 1, \ldots, M$ is an $N \times M$ array.

Proposition 1 *The klth component* $FT[w]_{kl} (k = 1, \ldots, n, l = 1, \ldots, m)$ *minimizes the function*

$$\Psi(y) = \sum_{j=1}^{l} (w(i, j) - y)^2 A_k(i) B_l(j) \tag{4}$$

The next statement [28] describes a representation of the discrete Fourier transform of the F-transform components.

Proposition 2 *Let* $Z_l = \{0, 1, \ldots, l-1\}$ *and* $\bar{w}$ *be the Fourier transform of a function* $w : Z_l \to R$. *Let* $n \geq 3$ *and* $A_1, \ldots, A_{n-1}$ *be a fuzzy partition* $[a, b]$. *Let* $W : Z_l \to R$ *be the discrete function given by*

$$W(t) = \sum_{j=0}^{l-1} A(t - j) w(j); t = 0, \ldots, l-1 \tag{5}$$

which contains the F-transform components of w among its values. The Fourier transform of W is given by

$$\begin{aligned} &\bar{W}(0) = \bar{w}(0) \\ &\bar{W}(k) \approx \frac{mn^2}{2\pi^2 k^2} \exp(-2\pi i k/n) \left(1 - \cos\frac{2\pi k}{n}.\bar{w}(k)\right. \\ &\quad k = 1, \ldots, l-1 \end{aligned} \tag{6}$$

where m is a fixed parameter.

By Proposition 2, the influence of the Fourier coefficient $\bar{w}(k)$ in the presentation (6) is weakened by the factor $\frac{1}{k^2}, k = 1, \ldots, l-1$, i.e., every F-transform component works as a low pass filter of an original function. So, the content loss takes place in edge pixels.

In each case, the difference between an original function and its inverse F-transform works as a high-pass filter of the former have been concluded. Therefore, this difference can be used as a membership function of the fuzzy set "noisy component" in respects the following rule: the bigger is the value n, the "higher" are the corresponding periodic noise bandwidth. This rule is a core of the proposed algorithm.

Peaks in a noisy spectrum are an area that can be characterized by a significant change of intensity. This characterization can be formalized. Firstly, the characteristic function of a given crisp set will be formed which corresponds, to the above-given characteristics and then provide a fuzzified version.

$$\begin{aligned} E = \{(x_i, y_j) \in P | \exists (x, y) \in P (d((x_i, y_j), (x, y)) \leq \lambda) \\ \text{and } |w(x_i, y_j) - w(x, y)| \geq \kappa\} \end{aligned} \tag{7}$$

where d is a distance, λ determines a size of the "important" neighborhood or in other way, bandwidth of the noise, and κ specifies a "significance" of the intensity change. The right-hand side expresses the fact that the value of a derivative of w at the point $(x_i, y_j) \in E$ is "rather big."

Now, the fuzzy set with the membership function is characterized for a noisy component that assigns a degree of belongness to every frequency. Given membership function fuzzifies the characteristic function (7) of a noisy component. Both properties Propositions 1 and 2 of the F-transform components will be used in the proposed characterization.

Let n, m be numbers of fuzzy sets in a fuzzy partition of the domain $P = [1, N] \times [1, M]$ of w. Numbers n, m relate to a size of a chosen neighborhood in (7) was characterized by the inequality $d((x_i, y_j), (x, y)) \leq \lambda$. In our approach, the above inequality is characterized by $A_i(x) \cdot B_j(y)$, where λ relates to lengths of supports of A_i, B_j. If $FT[w]_{i,j}, 1 \leq i \leq n, 1 \leq j \leq m$ are components of the F-transform of w, then the difference $|w(x, y) - FT[w]_{i,j}|$ approximates the respective difference $|w(x_i, y_j) - w(x, y)|$ in (7). Thus, the following fuzzy set is:

$$\begin{aligned} E = \{(x, y) \in P | (\exists i, j)(x \in A_i) \\ \text{and } (y \in B_j) \text{ and } |w(x, y) - FT[w]_{i,j}|\} \end{aligned} \tag{8}$$

The membership function of a noisy component E is represented in (8) by the formal expression with the following interpretation: $\exists$ relates to the addition, "and" to the product, $(x \in A_i)$ is interpreted as $A_i(x)$ (similarly, $y \in B_j$), and finally $|w(x, y) - FT[w]_{i,j}|$ is re-scaled to [0, 1] value. Therefore, the membership degree $E(x, y)$ is equal to $|\sum_{i=1}^{n} \sum_{j=1}^{m} A_i(x) B_j(y) (w(x, y) - FT[w]_{i,j})|$. The following Proposition gives another representation of the value of $E(x, y)$ which will be used in the periodic noise reduction algorithm.

Proposition 3 *Let the assumptions above be fulfilled. Then, for all* $(x, y) \in P$,

$$\begin{aligned} E(x, y) &= \left| \sum_{i=1}^{n} \sum_{j=1}^{m} A_i(x) B_j(y) (w(x, y) - FT[w]_{i,j}) \right| \\ &= |w(x, y) - w_{n.m}(x, y)| \leq \\ &\quad \sum_{i=1}^{n} \sum_{j=1}^{m} A_i(x) B_j(y) |(w(x, y) - FT[w]_{i,j})| \end{aligned} \tag{9}$$

4 Experimental Verification

In this section, the main steps of the periodic noise reduction algorithm demonstrate how the above-described method

works on a noisy image, it will be compared with the classical methods.

In this section, we will explain the main steps of the periodic noise reduction algorithm and demonstrate how the above-described method works on noisy images. We will also compare it with the classical methods for periodic noise reduction.

F-transform-Based Algorithm for Periodic Noise Reduction

This research is based on F-transform, and two main steps are considered in it. The first step is the periodic noise frequencies detection, and the second one is the restoration step.

Let $f(x, y)$ and $g(x, y)$ and $\hat{f}(x, y)$ be the pixel values of noise-less and noisy image and restored image in the (x, y) coordinate, respectively. The origin shifted Fourier transform of a noisy image is defined by

$$\bar{G}(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (-1)^{x+y} g(x, y) e^{-j2\pi(\frac{ux}{M} + \frac{vy}{N})} \tag{10}$$

where $(-1)^{(x+y)}$ denotes the origin shifting operation. The diagram of the proposed method is shown in Algorithm 1, and the peak detection algorithm is detailed through the following steps.

The Preprocessing

The preprocessing stage of the algorithm is normalizing the magnitude spectrum. The operation is performed by dividing the magnitude spectrum by its largest value and defined as:

$$\begin{aligned} ns(u, v) &= \frac{\left|\bar{G}(u, v)\right|}{\max_{u=1,\ldots,M, v=1,\ldots,N} \left|\bar{G}(u, v)\right|} \\ \left|\bar{G}(u, v)\right| &= \sqrt{R^2(u, v) + I^2(u, v)} \end{aligned} \tag{11}$$

where $R(u, v)$ and $I(u, v)$ are real and imaginary values of $\bar{G}(u, v)$.

Adaptive Peak Detection and Filtering

The proposed method is a spectral domain method. Thus, fuzzy partition, direct, and inverse F-transform have to be matched with spectrum properties. In an origin shifted Fourier spectrum, the amplitudes often decrease as the frequency increases in each direction away from the origin. Therefore, the frequency coefficients in polar coordinates have been used and defined fuzzy partitioning into

Algorithm 1 PSEUDOCODE FOR F- TRANSFORM- BASED PERIODIC NOISE REMOVAL

function FTBPNR($NoisyImage$)
  $Ncoef(i, j) = 1, \forall(i, j)$
  $F \leftarrow$ CONVERT TO FREQUENCY DOMAIN(NoisyImage);
  $ns \leftarrow$ NORMALIZE(F)/ $*\, Equation5$
  $NS \leftarrow$ DIRECTF- TRANSFORM(ns);
  $\hat{ns} \leftarrow$ INVERSEF- TRANSFORM(NS);
  for all (i, j) **do**
    if $\hat{ns}(i, j)/ns(i, j) < \chi(i, j)$ **then**
      $\mathcal{W} = \{(\hat{u}_i, \hat{v}_j)\} \leftarrow$ PLUS- SHAPEDNEIGHBORHOOD(i, j);
      $STD_{\mathcal{W}} = \sqrt{(ns_{\mathcal{W}} - \hat{ns}_{\mathcal{W}})^2}$;
      for each$(u, v) \in \mathcal{W}$ **do**
        if $(\hat{ns}(u, v)/ns(u, v) < \theta_2) and (STD(u, v) > \theta_1$ **then**
          $Ncoef(u, v) = \hat{ns}(u, v)/ns(u, v)$;
        end if
      end for
    end if
  end for
  $SPC \leftarrow$ SYMMETRICALLYPADDEDCONVOLUTION(Ncoef)/ $*\, Equation13$
  $FTNotchFilter \leftarrow min(Ncoef, SPC)$
  $\hat{F} \leftarrow F. * FTNotchFilter$
  $RestoredIMG \leftarrow$ CONVERTTOSPATIALDOMAIN($\hat{F}$);
  return $restoredIMG$
end function

$[1, max(\rho)]$ and $[0, \pi]$ intervals and adapted the direct and inverse F-transform to this domain. Note that (ρ, ϕ) is the polar coordinates of the position (u, v) in the spectrum of size $M \times N$. It is given by:

$$\rho = \sqrt{\left(\frac{u}{M}\right)^2 + \left(\frac{v}{N}\right)^2} \tag{12}$$

where ρ is the radial distance from $(0, 0)$ and ϕ is the counterclockwise angle from the u-axis.

The direct and inverse F-transform are modified as follows. Suppose that the input domain is a rectangle $[a, b] \times [c, d] \in \mathbb{Z} \times \mathbb{R}$ and $x_1 < \ldots < x_n$ are fixed nodes from $[a, b]$ and $y_1 < \ldots < y_n$ are fixed nodes from $[c, d]$, such that $x_1 = 1, x_n = max(\rho), y_1 = 0, y_m = \pi$ and $n, m \geq 2$. Assuming that $A_1, \ldots, A_n \subseteq [a, b]$ and $B_1, \ldots, B_m \subseteq [c, d]$ are basic functions defined on these nodes. The direct and inverse fuzzy transforms are given by

$$NS_{kl} = \frac{\sum_{j=0}^{\pi} \sum_{i=1}^{\max(\rho)} ns(\rho_i, \phi_j) A_k(\rho_i) B_l(\phi_j)}{\sum_{j=0}^{\pi} \sum_{i=1}^{\max(\rho)} A_k(\rho_i) B_l(\phi_j)} \tag{13}$$

for each $k = 1, \ldots, n, l = 1, \ldots, m$.

$$\hat{ns}(\rho_i, \phi_j) = \sum_{k=0}^{\pi} \sum_{l=1}^{\max(\rho)} NS_{kl} A_k(\rho_i) B_l(\phi_j) \tag{14}$$

The zero frequency coefficient is excluded in the calculation since its amplitude dominates over the others. The F-transform components of the normalized spectrum have been computed. Then, we obtain an approximation of the original one using the inverse F-transform.

In the proposed method, we consider practical consideration. It is expected that the coefficients of the noise-less image spectrum will not be changed. Firstly, the algorithm identifies the maximum frequency value of a noise peak. Then, the neighboring noisy areas associated with the peak are identified. In a noise-less image, no coefficient should be determined as center frequency and the neighboring coefficients should not undergo subsequent changes.

The center frequency in the normalized spectrum has a clear difference with its approximation. Additionally, it was mentioned that the amplitudes in the spectrum usually decrease as frequency increases in any direction away from the origin. So, we tried to utilize a descending threshold function to detect noise center frequencies as shown in the following equations:

$$\begin{aligned} CF(u,v) &= \begin{cases} 1 \text{ if } \frac{\hat{ns}(u,v)}{ns(u,v)} < \chi(u,v), \\ 0 \text{ otherwise} \end{cases} \\ \chi(u,v) &= \max\left(\epsilon, \frac{(\rho(u,v) - \max(\rho))^{power}}{2(\max(\rho)-1)^{power}}\right) \end{aligned} \tag{15}$$

where χ is a descending threshold function and ϵ is the greatest lower bound to identify the noisy center.

After that, the algorithm considers all neighborhood into a binary map image defined by:

$$\begin{aligned} &NOISEMAP(\hat{u},\hat{v}) = 1, \forall \hat{u} \in \{u-1, u, u+1\}, \hat{v} \in \{1,\ldots,N\}, \\ &\forall \hat{v} \in \{v-1, v, v+1\}, \hat{u} \in \{1,\ldots,M\} \end{aligned} \tag{16}$$

where (u, v) is the position of the coefficient that satisfies the condition in (15).

If the frequency center placed at the spectrum border, the binary map will wrap around its opposite side and will flag those pixels in the $NOIZEMAP$. Moreover, if noise-bandwidth increases, then many coefficients in a row or column would be involved. In this case, the condition in (15) is satisfied by several components in the center of the peak, and a greater number of adjacent rows and columns are flagged in the binary map.

We want to filter out only the noisy portion effectively without damaging the non-noisy part as much as possible. So, an adaptive filtration framework is proposed by exploiting the F-transform. Now, the flagged components are investigated with two considerations. First, it is important that the value of the flagged component in the normalized spectrum is greater than its approximation. Second, there is a large gap between this and the mean value. The approximation values obtained from the inverse fuzzy transform are used as mean. So, a filtering mask for flagged components of $NOISEMAP$ defined by:

$$Ncoef(u,v) = \begin{cases} \frac{\hat{ns}(u,v)}{ns(u,v)} & \text{if } STD(u,v) > \theta_1 \text{and} \frac{\hat{ns}(u,v)}{ns(u,v)} < \theta_2 \\ 1 & \text{otherwise} \end{cases} \tag{17}$$

$$\begin{aligned} &STD(u,v) = \sqrt{(ns(u,v) - \hat{ns}(u,v))^2} \\ &\forall (u,v) \in NOISEMAP \end{aligned} \tag{18}$$

where θ_1 is the minimum distance from the mean and θ_2 is the least upper bound for the ratio between the normalized coefficient value and its approximation.

We aim to form a filtering mask that reduces a portion of frequency value corresponding to the periodic noise. So, the term $\frac{\hat{ns}(u,v)}{ns(u,v)}$ is used in filtering mask.

Up to now, the noisy frequencies are identified. Then, the F-transform notch filter is obtained after convolving the outlier map, $Ncoef$, by a Gaussian kernel defined as:

$$\begin{aligned} FTmask(u,v) &= \min(Nooef(u,v), \\ &\quad \sum_p \sum_q Nooef(p,q) Ga(u-p+1, v-q+1) \\ Ga(n_1,n_2) &= \frac{h_g(n_1,n_2)}{\sum_{n_1}\sum_{n_2} h_g} \\ h_g(n_1,n_2) &= e^{\frac{-(n_1^2+n_2^2)}{2\sigma^2}} \end{aligned} \tag{19}$$

This filter is used to compensate for noisy components. Finally, the algorithm performs inverse shifted Fourier transform to reconstruct the restored image.

$$\check{F}(u,v) = \bar{G}(u,v) \times FTmask(u,v) \tag{20}$$

5 Implementation and Experimental Results

The proposed method is implemented in the MATLAB environment. The performance of this method has been objectively/subjectively assessed by other state-of-the-art algorithms in terms of the mean absolute error (MAE), the standard deviation of absolute error (STD), the peak signal-to-noise ratio ($PSNR$), the mean structural similarity index measure ($SSIM$), and the edges fall-out and miss rate(EFM). The performance analysis of different algorithms is carried out with synthetic and non-synthetic corrupted images.

If f and $\hat{f}$ are noise-less image and restored image of size $M \times N$, respectively, then the objective performance metrics

are expressed by:

$$MAE = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left|f(x,y)-\hat{f}(x,y)\right|}{M\times N} \tag{21}$$

$$STD = \left[\frac{1}{M\times N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\left[\left|f(x,y)-\hat{f}(x,y)\right|-MAE\right]^2\right]^{\frac{1}{2}} \tag{22}$$

$PSNR$ is the ratio of the power of peak signal to the power of distorting noise and it is usually expressed in terms of the logarithmic decibel scale [4]. It is mathematically defined as:

$$\begin{aligned} PSNR &= 10\log_{10}\left(\frac{255^2}{MSE}\right)(\mathrm{d}B)\\ MSE &= \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left(f(x,y)-\hat{f}(x,y)\right)^2}{M\times N} \end{aligned} \tag{23}$$

$SSIM$ takes into account the biological factors of the human vision system when comparing the quality of restored images [42]. It is expressed by

$$SSIM = \frac{(2\mu_f\mu_{\hat{f}}+C_1)\times(2\sigma_{f\hat{f}}+C_2)}{(\mu_f^2+\mu_{\hat{f}}^2+C_1)\times(\sigma_f^2+\sigma_{\hat{f}}^2+C_2)} \tag{24}$$

where μ_f and σ_f are the mean intensity and standard deviation of pixels in the original image, and $\mu_{\hat{f}}$ and $\sigma_{\hat{f}}$ are mean intensity and standard deviation of pixels in the de-noised image. C_1 and C_2 are added to avoid instability.

Fall-out and Miss Rate of the Edge Pixels

One of the obvious effects of the periodic noise is seen on the edge pixels; i.e., periodic noise may introduce some extra edges or destroy some of the edges. A good restoration algorithm must be able to remove the undesired edges and to reconstruct the decayed ones. Since false positive and true negative have a high chance of occurring, the performance of the restoration algorithms can also be evaluated in terms of fall-out and miss rate of the edge pixels [41]. For calculating those parameters, we need the information about true edge points and false edge points. Here, the Canny method is employed for edge detection.

(A) The miss rate of the edge pixels : This measure is the percentage of edges present in the edge image of the noise-less image but not present in the edge image of the restored image to the total number of edge pixels present in the noise-less image and is defined by:

$$\begin{aligned} \xi_1 &= \left(\frac{\sum_{x=1}^{M}\sum_{y=1}^{N}N_{\xi_1}(f,\hat{f})}{\sum_{x=1}^{M}\sum_{y=1}^{N}O(f,\hat{f})}\right)\times 100(\%)\\ N_{\xi_1}(f,\hat{f}) &= \begin{cases} 1 \text{ if } O(f,\hat{f})=1 \text{ and } R(f,\hat{f})=0\\ 0 \text{ otherwise} \end{cases} \end{aligned} \tag{25}$$

where O and R are the edge images of the original image (f), and the restored image ($\hat{f}$), respectively.

(B) The fall-out of the edge pixels: It is the percentage of edges present in the edge image of the restored image but are not present in the edge image of the noise-less image to the total number of non-edge pixels present in the noise-less image and is defined by

$$\begin{aligned} \xi_2 &= \left(\frac{\sum_{x=1}^{M}\sum_{y=1}^{N}N_{\xi_2}(f,\hat{f})}{MN-\sum_{x=1}^{M}\sum_{y=1}^{N}O(f,\hat{f})}\right)\times 100(\%)\\ N_{\xi_2}(f,\hat{f}) &= \begin{cases} 1 \text{ if } O(f,\hat{f})=0 \text{ and } R(f,\hat{f})=1\\ 0 \text{ otherwise} \end{cases} \end{aligned} \tag{26}$$

$$EFM = \xi_1+\xi_2 \tag{27}$$

An efficient algorithm in noise reduction would have a high value of $SSIM$ and $PSNR$ and low value of MAE, STD, and EFM.

The performance of the proposed method is compared with Windowed Adaptive Switching Minimum Filter (WASMF) [40], Adaptive Threshold-Based Frequency domain filter (ATBF) [39], Laplacian-based Frequency Domain Filter (LFDF) [41], Median Filter in spectral domain (AMF1) [3], Mean Filter in spectral domain (AMF2) [1], Windowed Gaussian Notch Filter (WGNF) [2], Adaptive Gaussian Notch Filter (AGNF) [22], Adaptive Optimum Notch Filter (AONF) [21], A-Contrario Automated Removal of quasi-Periodic noise using frequency domain statistics (ACARP) [36], Automated Removal of quasi-Periodic noise using frequency domain statistics (ARP) [38], soft morphological filter (SMF) [17], Adaptive Sinc Restoration Filter (ASRF) [6]. Table 1 shows the parameters of compared methods.

5.1 Results and Discussion

The proposed method was tested in several steps as follows

- Synthetic periodic noise
- Low-frequency periodic noise of various noise strengths
- High-frequency periodic noise of various noise strengths
- Pure and quasi-periodic noise

Table 1 The parameters of compared methods

Method	Parameter	value	Method	Parameter	value
AMF2(Mean)	Window size	11×11	LFDF	C_1	0.4
	Threshold	7		C_2	1.1
	Normalizing Divider	50		γ	0.9
ARP	Patch size	128	AONF	w	3
WGNF	Window size	11×11	ASRF	C_1	10
	Threshold	7		C_2	2.5
	A	0.1		Structure element size	5
	B	1.0		Smallest filtering window(P)	2
AMF1(Median Like)	Window size	11×11	WASMF	C_1	0.4
	Threshold	7		C_2	1.1
ATBF	Alpha	1.8	SMF	Structure element size	5
AGNF	w	3	ACARP	Patch size	128
	A	1.0		LogNFAthresh	0
	B	0.01			

- Striping noise
- Multi-frequency periodic noise
- Periodic noise on a tonal gradient image
- Non-synthetic periodic noise
- Noise-less image
- Periodic noise removal in RGB images

Then, the computational complexity analysis of the proposed method is discussed. The synthetic corrupted images are created by adding artificially generated sinusoidal noise patterns to the uncorrupted reference images. The performance of spectral domain techniques is strongly dependent on the test image and the noise parameters; therefore, the results averaged on the 40 repetitions under test conditions. In all tests, the test images are 256×256 pixels. In all tables, the "noisy image" column is added to evaluate the quality of the degraded noisy image. For that, the performance values are averaged for each set of noisy images. Tables 2, 3, 4, 5, 6, and 7 show the performance comparison of our proposed algorithm with comparison with other approaches in terms of performance measures.

Synthetic Periodic Noise

Low-frequency Periodic Noise of Various Noise Strengths

In this step, low-frequency periodic noise structures are considered as a noise source. In this case, the simulations are carried out on the images with noise pattern of Eq. 1 with $u_0, v_0 \subseteq [2, 14]$. The simulation results are shown in Table 2.

This table shows the performance of the compared algorithm for synthetic periodic noise with varying noise amplitudes. Generally, the restoration method should restore a more inferior image when increasing the noise amplitude. It is evident from Tables 2 that the proposed algorithm is capable of achieving higher performances than the others.

The LFR radius detection in AGNF is not optimum in [8]. In AGNF, the coefficients are averaged to calculate the LFR radius. As DC and its nearby region are usually enriched with higher spectral amplitude values, these values will be affected the average value when the noise peak is revealed near the LFR;

In this case, the noise peak may not be detected and mistaken as part of the LFR. The simulations show that some methods are incapable of peak detection when the noise peak appears near the DC component. This issue is reflected in the results as in Table 2. It is due to WASMF, LFDF, ACARP, and ARP use static prefixed windows for the LFR radius, irrespective of noise conditions; in this case, the noisy peak area may place in this region and exclude this peak from restoration. Of course, ACARP and ARP show better performance because the LFR radius is relatively small in these methods. Also, in ASRF, the noise area is added to the boundaries of the LFR in the dilation process and is excluded from restoration.

As shown in Table 2, a higher PSNR value is not necessarily associated with a higher SSIM value. In other words, the method may have a higher PSNR and lower SSIM than the other, or vice versa. This will better capture the challenges of periodic noise reduction. Noisy components should be replaced by appropriate values after the restoration method to find the noise peaks. Otherwise, the brightness of the image may be affected. In this case, a restoration method has a higher value of the SSIM metric and a lower value of the

Table 2 Comparison among different restoration algorithms for restoring sample images corrupted by low-frequency periodic noise in terms of performance metrics ($\epsilon = 0.08$, $power = 10$, $\theta_1 = 0.5$, $\theta_2 = 9 \times 10^{(-4)}$)

A	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*	*K*	*L*	*M*	*N*	*O*	*Q*	*P*
Boat	0.1	1	28.80	27.56	28.80	26.81	21.52	31.39	26.72	28.34	19.10	30.38	29.52	20.83	29.22	33.53
		2	0.933	0.867	0.933	0.838	0.941	0.961	0.957	0.894	0.734	0.951	0.940	0.507	0.947	0.976
		3	4.36	5.81	4.36	6.69	10.29	3.82	7.64	5.59	23.46	3.99	4.24	16.54	4.57	3.16
		4	23.73	38.10	23.73	43.84	18.22	18.84	24.94	33.50	57.54	19.28	21.41	99.19	20.77	13.39
	0.3	1	20.56	20.40	22.00	20.36	20.40	24.68	23.41	25.00	16.47	23.04	22.50	18.62	21.74	26.26
		2	0.697	0.668	0.748	0.662	0.859	0.855	0.857	0.872	0.536	0.804	0.781	0.365	0.750	0.906
		3	11.61	12.13	10.61	12.63	13.37	8.61	16.10	8.81	26.11	9.76	10.21	18.27	10.91	7.53
		4	57.44	63.42	49.61	64.54	30.49	34.52	32.11	32.30	73.92	42.93	45.80	101.9	48.83	26.29
	0.6	1	16.28	16.25	18.92	16.27	19.11	20.94	19.69	20.46	13.77	18.33	18.11	16.13	18.12	22.24
		2	0.495	0.483	0.645	0.482	0.760	0.752	0.708	0.752	0.366	0.626	0.613	0.257	0.616	0.821
		3	19.53	19.73	16.23	20.14	16.53	13.42	18.28	13.34	30.30	16.92	17.24	22.54	17.47	11.99
		4	77.96	80.27	59.75	81.24	48.72	46.67	46.70	40.61	87.31	63.20	64.29	102.2	61.32	36.43
	0.9	1	14.23	14.22	18.18	14.43	18.53	19.36	18.42	19.22	12.20	16.56	16.51	14.68	16.68	20.44
		2	0.388	0.377	0.631	0.391	0.716	0.705	0.702	0.708	0.285	0.548	0.544	0.208	0.557	0.773
		3	24.95	25.11	18.31	25.07	18.19	16.42	20.54	17.22	34.90	21.10	21.17	26.06	21.26	14.97
		4	86.45	88.02	59.83	87.30	49.41	50.39	45.68	48.65	92.34	70.81	70.97	101.8	63.48	39.02
	1.2	1	13.01	13.01	16.30	13.16	17.44	17.43	17.60	17.30	11.04	14.15	13.98	13.43	14.81	18.37
		2	0.364	0.351	0.563	0.361	0.648	0.632	0.641	0.633	0.259	0.440	0.430	0.196	0.494	0.687
		3	28.84	28.99	22.55	28.94	20.49	20.07	22.26	20.95	39.15	26.57	26.95	29.49	25.66	18.62
		4	86.17	87.96	65.82	87.80	56.44	57.04	57.84	56.01	94.00	79.87	80.28	101.0	70.26	49.24

Table 2 continued

A	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*I*	*J*	*K*	*L*	*M*	*N*	*O*	*Q*	*P*
Lake	0.1	[1]	28.35	27.16	28.35	25.81	16.70	28.48	24.94	27.08	5.16	29.49	27.83	19.99	25.45	32.02
		[2]	0.930	0.858	0.930	0.811	0.884	0.932	0.916	0.882	0.430	0.947	0.917	0.552	0.881	0.970
		[3]	4.62	6.08	4.26	7.60	17.21	4.59	8.63	6.22	118.7	4.33	5.81	18.09	8.27	3.63
		[4]	15.49	24.93	15.49	31.99	12.39	15.14	15.61	23.41	83.55	13.02	18.82	99.12	25.83	9.29
	0.3	[1]	20.15	19.98	20.15	19.82	16.38	22.88	20.78	23.24	6.24	22.32	21.70	17.80	19.69	24.88
		[2]	0.710	0.684	0.710	0.668	0.845	0.823	0.879	0.843	0.335	0.810	0.783	0.413	0.685	0.909
		[3]	12.37	12.92	12.37	13.74	18.36	10.15	13.75	10.01	104.1	10.50	11.44	20.34	14.87	8.18
		[4]	44.73	49.07	44.73	51.65	21.32	29.48	21.47	27.96	88.27	31.71	36.36	100.8	49.58	18.55
	0.6	[1]	15.58	15.54	17.50	15.52	15.54	18.66	18.17	19.70	10.71	17.17	16.98	15.20	16.15	20.24
		[2]	0.517	0.505	0.646	0.500	0.782	0.728	0.790	0.796	0.327	0.637	0.620	0.293	0.557	0.831
		[3]	22.38	22.60	18.57	23.06	20.95	16.62	16.12	14.12	55.32	19.16	19.70	26.03	21.63	12.57
		[4]	68.28	70.43	51.64	70.98	30.45	41.96	44.93	31.44	86.72	52.83	55.43	102.4	62.74	27.05
	0.9	[1]	13.47	13.46	16.25	13.59	15.00	16.69	16.57	17.30	9.49	15.30	15.15	13.70	13.97	18.14
		[2]	0.409	0.401	0.627	0.409	0.745	0.662	0.750	0.712	0.251	0.563	0.545	0.236	0.460	0.779
		[3]	29.09	29.21	21.48	29.12	21.73	20.25	18.10	17.91	59.58	24.00	24.52	30.63	27.63	15.02
		[4]	79.13	80.64	53.72	79.84	41.77	48.94	38.83	39.50	92.94	60.76	63.46	102.4	71.73	29.62
	1.2	[1]	12.22	12.22	14.69	12.31	14.33	14.88	15.47	15.29	9.38	13.08	12.98	12.47	12.34	16.31
		[2]	0.377	0.695	0.567	0.373	0.659	0.586	0.659	0.622	0.240	0.449	0.436	0.217	0.397	0.698
		[3]	33.92	34.01	25.93	33.95	23.64	25.30	22.05	23.30	54.19	31.14	31.53	34.98	33.90	19.24
		[4]	79.98	81.52	58.42	81.58	47.37	55.87	45.20	48.46	93.70	72.18	74.07	101.6	77.20	39.02

[1] PSNR [2] SSIM [3] STD [4] EFM [A] Image [B] Amplitude [C] Performance metric [D] Noisy Image [E] WASMF [F] ATBF [G] LFDF [H] AMF1 [I] AMF2 [J] WGNF [K] AGNF [L] AONF [M] ACARP [N] ARP [O] SMF [Q] ASRF [P] Proposed Method

PSNR. Obviously, this is more important for low-frequency noise due to the high values of the low-frequency region.

High-frequency Periodic Noise of Various Noise Strengths

In Table 3, high-frequency periodic noise structures are considered. In this case, the simulations are carried out on the images with noise pattern of Eq. 1 with $u_0, v_0 > 20$. The simulation results are shown in Table 3. It is clear in Tables 2 and 3 that the restoration algorithms have better performance in high-frequency periodic noise fading.

When the bandwidth of the periodic noise increases, several rows or columns of the spectrum may be involved, so a restoration method with better detection and correction capabilities has a higher efficiency.

The region growing approach in AGNF curbs the number of detected coefficients. In ASRF, the noise affected zone is limited by the position of the nearest fundamental peak and the DC component. This method is better to expand the correction of coefficients in rows and columns, but as the noisy zone has expanded across the spectrum, many coefficients undergo changes inevitably.

In addition, for increasing the noise power tends to decrease the performance of spatial-domain methods such as SMF and AONF regardless of frequency bands. Naturally, AMF1, AMF2, and WGNF may perform efficiently for a specific image-noise combination, but they are not adaptive.

Pure and Quasi-periodic Noise

In this step, pure periodic noise is added to the original images and the performance of the proposed method is evaluated. In this case, the simulations are carried out on the images with noise pattern of Eq. 1 with integer values for u_0 and v_0. The simulation results are shown in Table 4.

Periodic noise is detected as peaks in pure and quasi-periodic case. In pure case, the periodic noise affects only one frequency, and quasi-periodic case influences a set of frequencies.

The size of the filtering window in WGNF is not adaptive to noisy peak areas. This decreases the performance of this method for pure periodic noise. In ARP and ACARP, the noise peak is characterized in the average power spectrum which is smaller than the original spectrum size. Then, the outlier map is expanded by interpolation to match the size of the spectrum. The interpolation operation propagates filtering from bright spots to the surrounding area. It may fail to produce satisfactory results for pure periodic noise.

Stripping Noise

In this step, stripping noise structures are considered as a noise source. The simulation results are shown in Table 5.

Stripping noise signal always appears as spike along the vertical or horizontal axis in the Fourier domain. In this case, the simulations are carried out on the images with noise patterns of Eq. 1 with $u_0 = 0$ and $v_0 = 0$. De-striping requires more precision in the detection and correction of noisy frequencies. In this case, the filtering problem consists of removing the striping noise of the images without introducing any blurring effect or eliminating the horizontal/vertical components of the spectrum. So, only prominent peaks must be considered.

Multi-frequency Periodic Noise

In this case, the simulations are carried out on the images with noise pattern of Eq. 1 with $s \subseteq 1, 2, 3, 4, 5$. The simulation results are shown in Table 6. Qualitative evaluation of the restoration algorithms is shown in Fig. 6 for Lena image corrupted by multi-frequency periodic noise.

Generally, the restoration algorithms work well for single frequency periodic noise, but multi-frequency periodic noise fading is a challenging problem. Most of the non-synthetic images are corrupted by multi-frequency periodic noise. In this simulation, the reference images are presented with additional multi-frequency periodic noise.

Periodic Noise in a Tonal Gradient Image

Periodic noise increases the value of one or more coefficients in the spectrum. If the noisy component is replaced by zero value, the damage to image frequency is notable. This effect may not be obvious in high-frequency noise condition, but the degree of damage may be severe in low-frequency noise. However, the compared methods used various correction approach to reinstate the noisy frequency. The result must preserve the dynamic range of the intensity values, accuracies and sharpness of the edges. In this simulation, a tonal gradient corrupted by synthetic periodic noise is considered. Qualitative results of the restoration algorithms are shown in Fig. 5.

One of the advantages of the proposed method is its noise-correction-manner framework. The main slogan is to filter out only a portion of the flagged component, and this portion is given by F-transform. Therefore, the tonal attributes of the restored image will be preserved. As depicted in Fig. 5, the proposed method preserves gray-scale values, while it removes the periodic patterns. In this case, F-transform helps to replace the noise coefficient by a proper value.

Table 3 Performance values of different methods for high-frequency periodic noise under various noise levels($(\epsilon = 0.08,\ power = 10, \theta_1 = 0.5, \theta_2 = 9 \times 10^{(-4)})$)

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	Q	P
Boat	0.1	1	28.84	32.92	34.28	29.52	21.83	35.65	23.86	36.81	26.21	38.00	37.95	21.62	33.84	38.05
		2	0.758	0.920	0.908	0.842	0.941	0.938	0.929	0.956	0.723	0.981	0.979	0.561	0.957	0.981
		3	4.34	3.61	2.93	5.18	9.40	2.49	11.17	2.13	8.36	1.75	1.84	16.29	3.74	1.73
		4	11.00	28.52	8.27	38.82	10.27	5.83	20.95	6.41	25.17	8.95	9.00	94.40	16.94	8.23
	0.3	1	20.59	27.96	25.80	23.11	21.14	26.48	22.56	27.56	19.56	28.74	28.66	21.40	28.60	28.98
		2	0.401	0.877	0.727	0.574	0.871	0.766	0.895	0.829	0.374	0.924	0.921	0.552	0.922	0.938
		3	11.58	6.76	7.78	10.43	12.00	7.33	13.12	6.79	23.92	6.31	6.35	15.79	6.40	6.12
		4	34.65	28.02	17.54	44.25	13.97	15.34	22.85	12.17	48.71	13.02	13.07	85.03	14.85	12.03
	0.6	1	16.29	23.16	21.04	18.81	19.69	21.35	20.79	22.54	15.30	23.39	23.34	20.34	23.74	23.90
		2	0.221	0.561	0.411	0.750	0.579	0.579	0.837	0.682	0.210	0.839	0.835	0.526	0.851	0.871
		3	19.52	11.58	13.43	17.06	15.70	12.99	15.87	11.92	30.52	11.22	11.26	16.57	10.92	10.66
		4	40.82	28.28	25.16	45.24	18.88	23.73	24.22	16.00	54.25	13.45	13.56	78.41	13.21	13.93
	0.9	1	14.26	21.28	19.81	15.89	18.86	19.69	19.65	20.35	14.08	21.40	21.36	19.48	21.57	21.67
		2	0.145	0.783	0.558	0.264	0.727	0.540	0.788	0.595	0.140	0.803	0.799	0.506	0.805	0.824
		3	24.90	14.27	15.86	22.92	17.75	15.98	17.82	15.27	25.99	13.97	14.01	17.68	13.64	13.64
		4	32.49	25.25	18.45	42.88	15.90	17.90	24.49	16.50	52.20	11.96	11.13	77.35	13.78	10.27
	1.2	1	13.04	19.78	17.86	14.02	17.98	17.79	18.81	18.74	12.27	19.77	19.72	18.59	18.98	20.18
		2	0.115	0.716	0.434	0.184	0.621	0.411	0.739	0.496	0.108	0.730	0.724	0.480	0.708	0.758
		3	28.79	16.80	19.45	27.82	19.62	19.33	19.39	18.13	38.90	16.61	16.67	19.24	18.67	15.97
		4	37.42	25.36	23.38	46.98	18.50	23.59	26.17	22.57	59.54	18.48	18.77	76.40	28.72	17.88

Table 3 continued

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	Q	P
Lake	0.1	1	28.52	32.27	33.32	28.33	16.89	34.54	26.32	36.40	23.02	29.39	29.54	20.72	28.19	38.11
		2	0.767	0.906	0.902	0.824	0.899	0.935	0.953	0.960	0.686	0.984	0.966	0.612	0.909	0.984
		3	4.46	3.84	3.24	5.87	16.22	2.88	7.57	2.39	25.77	4.41	4.68	17.88	6.34	1.88
		4	7.22	19.34	6.18	27.94	6.91	3.94	10.72	4.46	25.74	4.30	10.17	94.95	22.41	3.56
	0.3	1	20.15	26.53	24.73	23.06	16.71	24.73	23.25	26.50	18.82	24.05	24.02	20.28	22.08	27.78
		2	0.447	0.869	0.747	0.563	0.859	0.747	0.923	0.847	0.414	0.942	0.927	0.606	0.806	0.950
		3	12.49	6.40	8.35	10.36	16.26	8.44	9.82	6.49	32.35	5.68	6.93	16.35	12.13	4.83
		4	27.90	22.46	13.28	30.78	10.15	13.36	12.65	7.54	41.00	6.64	11.41	83.17	34.54	6.02
	0.3	1	15.60	21.00	19.45	17.96	15.92	18.98	19.76	20.60	14.19	20.01	19.99	18.62	20.63	21.51
		2	0.265	0.821	0.591	0.482	0.760	0.546	0.861	0.704	0.246	0.865	0.854	0.577	0.834	0.882
		3	22.42	10.19	14.65	18.00	18.80	15.90	12.08	11.97	42.36	14.09	14.54	15.20	11.00	8.96
		4	36.51	16.30	20.01	37.72	14.07	21.02	13.89	11.31	49.43	8.99	14.21	72.60	22.85	8.66
	0.9	1	13.50	18.65	17.74	15.02	15.31	17.05	17.93	18.15	13.30	17.69	17.68	17.33	18.95	18.95
		2	0.181	0.787	0.580	0.321	0.729	0.491	0.808	0.614	0.175	0.826	0.815	0.547	0.826	0826
		3	29.13	12.58	16.55	25.05	19.90	18.98	13.69	15.46	30.22	17.96	18.07	15.47	11.65	11.61
		4	28.24	14.18	15.13	34.89	12.68	16.70	13.76	13.45	45.56	8.49	13.01	71.11	8.78	8.65
	1.2	1	12.25	17.08	15.90	13.37	14.70	15.25	16.70	16.55	11.37	16.49	16.47	16.27	17.24	17.41
		2	0.144	0.714	0.458	0.245	0.634	0.378	0.749	0.516	0.135	0.754	0.744	0.512	0.752	0.760
		3	33.91	15.33	21.10	30.61	21.20	23.96	15.47	18.95	48.27	21.17	21.14	16.93	14.60	14.05
		4	33.08	14.89	19.68	39.73	13.74	20.93	14.53	18.31	53.01	10.90	16.16	69.06	10.50	10.47

[1] PSNR [2] SSIM [3] STD [4] EFM [A] Image [B] Amplitude [C] Performance metric [D] Noisy Image [E] WASMF [F] ATBF [G] LFDF [H] AMF1 [I] AMF2 [J] WGNF [K] AGNF [L] AONF [M] ACARP [N] ARP [O] SMF [Q] ASRF [P] Proposed Method

Table 4 Comparison among different restoration algorithms for restoring sample images corrupted by pure periodic noise in terms of performance metrics (($\epsilon = 0.08$, $power = 10$, $\theta_1 = 0.5$, $\theta_2 = 9 \times 10^{(-4)}$))

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Boat	1	16.20	21.53	17.60	16.43	19.79	24.40	21.89	23.87	15.53	24.21	24.7	19.45	23.53	24.40
	2	0.242	0.683	0.295	0.259	0.875	0.892	0.867	0.878	0.225	0.887	0.882	0.451	0.879	0.892
	3	21.58	14.82	18.81	21.64	15.71	11.57	14.28	11.84	23.96	11.66	11.76	18.20	12.29	11.57
	4	95.94	49.15	90.14	95.90	8.70	3.49	14.68	8.06	97.49	7.57	8.10	86.23	11.69	3.48
Lake	1	17.34	23.56	20.28	16.96	16.16	25.28	20.94	25.27	16.61	19.23	19.33	18.91	20.84	25.28
	2	0.327	0.857	0.823	0.311	0.876	0.923	0.888	0.923	0.317	0.924	0.909	0.597	0.840	0.923
	3	20.46	8.56	12.36	21.52	17.17	7.27	11.26	7.27	23.13	9.81	11.32	15.71	11.47	7.31
	4	20.06	20.50	13.11	39.47	5.43	3.15	10.06	3.47	32.98	2.71	9.49	74.04	20.05	2.60

[1] PSNR [2] SSIM [3] STD [4] EFM [A] Image [B] Performance metric [C] Noisy Image [D] WASMF [E] ATBF [F] LFDF [G] AMF1 [H] AMF2 [I] WGNF [J] AGNF [K] AONF [L] ACARP [M] ARP [N] SMF [O] ASRF [P] Proposed Method

Table 5 Comparison among different restoration algorithms for de-stripping sample images corrupted by stripping noise in terms of performance metrics (($\epsilon = 0.08$, $power = 10$, $\theta_1 = 0.5$, $\theta_2 = 9 \times 10^{(-4)}$))

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Boat	1	17.18	23.89	17.40	17.55	20.06	24.51	20.67	23.89	15.70	24.68	24.63	19.78	23.02	24.87
	2	0.299	0.828	0.319	0.325	0.872	0.792	0.833	0.747	0.276	0.863	0.859	0.488	0.866	0.875
	3	19.47	11.68	19.47	19.60	15.01	11.90	16.35	11.83	25.46	11.36	11.41	17.74	11.57	11.14
	4	43.53	30.88	42.53	55.68	12.69	13.42	25.76	16.69	73.92	11.28	11.42	85.22	15.99	10.51
Lake	1	16.57	21.81	17.93	17.31	15.97	22.14	20.01	22.23	13.46	22.25	22.39	18.14	20.21	22.81
	2	0.332	0.824	0.402	0.399	0.830	0.743	0.853	0.758	0.275	0.841	0.848	0.530	0.823	0.853
	3	22.38	10.44	21.26	21.53	17.75	11.48	12.47	11.59	38.35	14.02	14.87	17.11	12.46	10.44
	4	38.17	20.78	43.21	43.35	12.22	12.24	14.92	13.23	70.59	12.24	14.70	81.64	20.86	11.31

[1] PSNR [2] SSIM [3] STD [4] EFM [A] Image [B] Performance metric [C] Noisy Image [D] WASMF [E] ATBF [F] LFDF [G] AMF1 [H] AMF2 [I] WGNF [J] AGNF [K] AONF [L] ACARP [M] ARP [N] SMF [O] ASRF [P] Proposed Method

Table 6 Comparison among different restoration algorithms for restoring sample images corrupted by multi-frequency periodic noise in terms of performance metrics ($(\epsilon = 0.08, power = 10, \theta_1 = 0.5, \theta_2 = 9 \times 10^{(-4)})$)

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Lena	1	18.15	23.93	22.71	21.47	18.32	23.01	21.89	23.30	17.72	18.04	24.70	20.74	24.85	25.66
	2	0.265	0.739	0.570	0.501	0.812	0.609	0.824	0.597	0.256	0.262	0.820	0.532	0.853	0.921
	3	17.89	9.42	11.30	12.91	14.02	10.87	12.43	10.49	19.34	18.07	8.41	15.59	8.16	7.26
	4	57.22	43.11	35.20	48.68	22.59	26.75	40.65	32.48	64.42	57.97	27.68	82.72	28.18	9.98

[1] PSNR [2] SSIM [3] STD [4] EFM [A] Image [B] Performance metric [C] Noisy Image [D] WASMF [E] ATBF [F] LFDF [G] AMF1 [H] AMF2 [I] WGNF [J] AGNF [K] AONF [L] ACARP [M] ARP [N] SMF [O] ASRF [P] Proposed Method

Non-Synthetic Periodic Noise

The performance of the proposed method is evaluated in a real situation. In this case, the images from various fields, corrupted with different types of non-synthetic periodic noise structures, are tested as a benchmark.

As the distortion-free reference image is not within reach, the performance evaluation metrics cannot be computed. Hence, the performance is compared only visually. Figures 7 and 8 show that our method outperforms the most recent state-of-the-art algorithms for non-synthetically corrupted images in real situations. Figure 9 shows the restored outputs of the proposed algorithm while restoring a few non-synthetically corrupted images.

Periodic Noise Removal in RGB Images

Spectral periodic noise reduction methods can easily generalize from gray-scale images to RGB images. For this purpose, image planes are separated from each other. Then, the periodic noise reduction method is applied to each separated plane. The final RGB image is obtained from the combination of these results. Figure 3 shows the restored outputs of the proposed algorithm while restoring the corrupted color image.

Noise-less Image

It is expected that the coefficients of the noise-less image spectrum will not be changed, particularly when dealing with adaptive periodic noise detection. The local structures of the original image must be well preserved. In this simulation, the reference images are considered and the performance of the proposed method is evaluated. The simulation results are shown in Table 7. In this table, "INF" denotes infinity, i.e., the restored image is the same as the reference image.

While filtering permits to eradicate periodic noise, this requires first to discriminate between noise spikes and spectrum patterns caused by spatially localized textures or repetitive structures. It is important to note that the periodic structures and repetitive patterns are revealed as peaks at the spectrum. These peaks would be the weakness of the spectral methods.

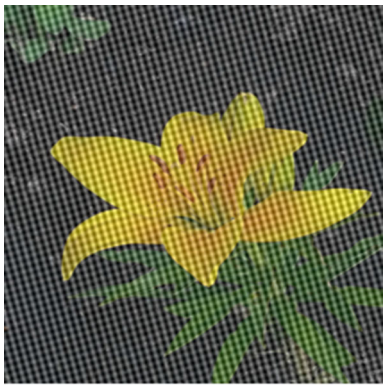
Parameter Analysis

This section makes the parameter analysis of the parameters used by the proposed algorithm. The parameter ϵ is used for ensuring that the descending function is above the uncorrupted version of the F-transformed spectrum. Figure

Table 7 Comparison among different restoration algorithms for restoring sample noise-less images in terms of performance metrics (($\epsilon = 0.08$, $power = 10$, $\theta_1 = 0.5$, $\theta_2 = 9 \times 10^{(-4)}$))

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Boat	PSNR	*Inf*	33.20	*Inf*	30.57	21.99	*Inf*	24.14	33.87	13.69	54.41	46.08	21.37	*Inf*	*Inf*
	SSIM	1	0.913	1	0.872	0.970	1	0.939	0.931	0.651	0.999	0.995	0.546	1	1
	STD	0	3.57	0	4.62	8.83	0	10.90	3.29	43.79	0.44	1.03	17.00	0	0
	EFM	0	31.87	0	40.01	8.70	0	19.46	28.42	71.72	0.37	1.71	99.40	0	0
Lake	PSNR	*Inf*	33.27	*Inf*	28.84	16.95	57.89	21.46	32.32	5.54	*Inf*	32.60	20.37	29.33	*Inf*
	SSIM	1	0.910	1	0.843	0.922	0.999	0.899	0.933	0.491	1	0.962	0.589	0.936	1
	STD	0	3.49	0	5.56	15.76	0.307	13.46	4.01	115.3	0	4.14	18.90	5.62	0
	EFM	0	18.66	0	29.03	5.34	0.06	15.87	18.25	84.39	0	13.39	99.10	19.66	0
Lena	PSNR	*Inf*	39.85	*Inf*	32.01	*Inf*	*Inf*	27.12	41.97	24.51	*Inf*	*Inf*	22.84	*Inf*	*Inf*
	SSIM	1	0.909	1	0.881	1	1	0.962	0.977	0.856	1	1	0.675	1	1
	STD	0	1.64	0	3.83	0	0	6.85	1.36	12.95	0	0	15.47	0	0
	EFM	0	17.37	0	32.50	0	0	10.87	5.93	42.07	0	0	97.94	0	0
woman	PSNR	*Inf*	41.85	*Inf*	34.52	19.65	*Inf*	21.58	45.70	10.32	*Inf*	42.33	23.38	*Inf*	*Inf*
	SSIM	1	0.975	1	0.919	0.906	1	0.860	0.987	0.432	1	0.986	0.809	1	1
	STD	0	1.41	0	2.91	11.55	0	12.96	1.06	10.62	0	1.68	15.70	0	0
	EFM	0	25.95	0	45.32	20.38	0	39.07	5.47	100	0	17.33	90.25	0	0

[1] PSNR [2] SSIM [3] STD [4] EFM [A] Image [B] Performance metric [C] Noisy Image [D] WASMF [E] ATBF [F] LFDF [G] AMF1 [H] AMF2 [I] WGNF [J] AGNF [K] AONF [L] ACARP [M] ARP [N] SMF [O] ASRF [P] Proposed Method



(a) Corrupted yellow lily

(b) De-noised yellow lily

Fig. 3 Periodic noise removal in RGB images using proposed method ($\epsilon = 0.08$, $power = 10, \theta_1 = 0.5, \theta_2 = 9 \times 10^{(-4)}$)

4a shows the average performance metrics values obtained from several restored images by varying noise strength and ϵ values. In this research, the value of ϵ is set to 0.08.

The parameter θ_1 is used to ensure that a component is greater than its fuzzy transformed component. Since the peak center passes the previous step, the parameters θ_1 and θ_2 are used to extend substitution across adjacent components in rows and columns. More neighboring frequencies may be changed by increasing the value of these parameters. Figure 4b shows the average performance metrics values obtained from several restored images by varying noise strength and θ_1 values. In this study, an attempt is made to change the minimum frequency components. Therefore, experiments conducted by varying θ_1. The parameter θ_1 is selected 0.5 in this research.

Each noise component must be dominant over its counterpart component in the F-transformed spectrum, and the algorithm controls the domination intensity by parameter θ_2. The increase in this parameter gradually decreases this sensitivity. Hence, according to Fig. 4c, more neighbors may be changed, and the performance measures may be reduced. In this research, the parameter θ_2 is selected $9 \times 10^{(-4)}$.

It is noted that in all of the experiments of this section, different types of periodic noise (low frequency, high frequency, pure, stripping, and multi-frequency) considered and the average value of criteria reported. In addition, for testing each parameter, the other two parameters are considered fixed and the experiments are conducted.

Computational Complexity Analysis

Since the proposed method is a spectral method, it is necessary to compute the Fourier transform of the image. If fast Fourier transform is used, the complexity of this step would be $O(MNlogMN)$ [11].

For a general type of partitions, the time consumption requirement is $o(Sr(2^r + \log(n))$ for direct F-transform and $o(r(2^r + \log(n)))$ for inverse F-transform [26] where r is the dimension of the original function domain that is to say the number of input variables. The frequency coefficients are presented in polar coordinates so, $r = 2$. Also, S is the count of the sample in which the original function is known or simply the number of input data vectors. In order to speed up further, the proposed method makes use of the symmetric property of the Fourier spectrum. It processes the noisy peak area detection to the first two quadrants. So, $S = (M \times N)/2$, M, and N are the spectrum size.

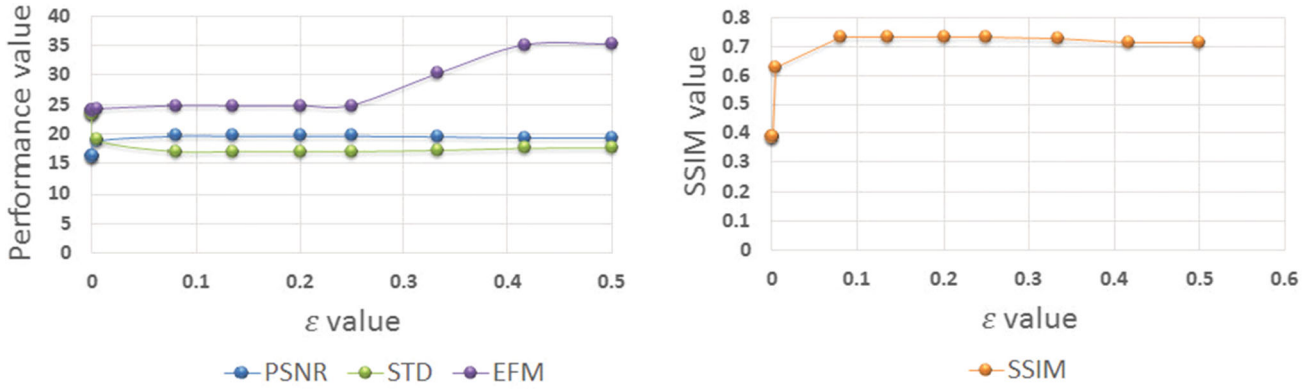
Finally, n is equal to the maximal number of basic functions from individual dimensions and defined by

$$n = \bigvee_{l=1}^{r} n_l \tag{28}$$

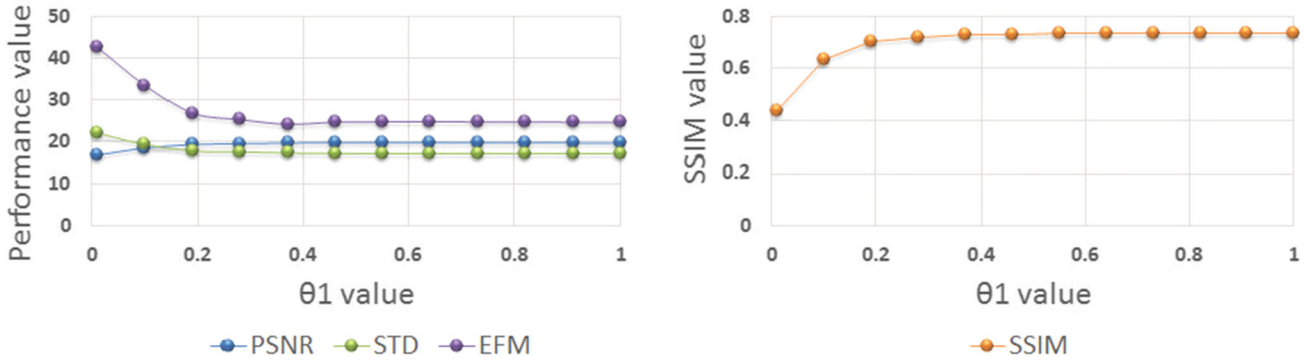
where $\bigvee$ denote the maximum operator and n_l is number of basic functions in dimension l.

Low-frequency components with large amplitude are located near the center of the Fourier spectrum and higher frequency components with small values are located toward the edges of the Fourier spectrum. Notably, non-uniform partitioning was considered in the proposed method. $\{2^i\}, i = 0, 1, \ldots, \log_2(\max(\rho))$ are considered as fixed nodes in the ρ dimension. In addition, a 10-uniform partition with $\{10(j-1), j = 1, \ldots, 19\}$ nodes is considered for the ϕ dimension.

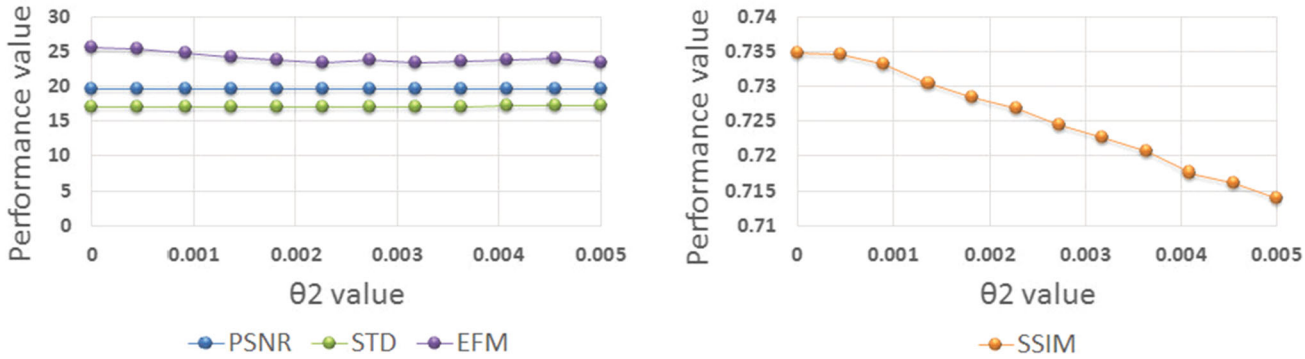
The number of noise coefficients of a spectrum is usually lower than non-noisy coefficients. If the computational complexity of the correction of these coefficients is assumed to be constant, the computational complexity of the algorithm is dominated by the Fourier transform. Therefore, the computational complexity of the proposed method is $O(MN \log(MN))$.



(a) Average performance metrics values obtained from several restored images by varying ϵ values.



(b) Average performance metrics values obtained from several restored images by varying θ_1 values.



(c) Average performance metrics values obtained from several restored images by varying θ_2 values.

Fig. 4 Parameter analysis on ϵ, θ_1, and θ_2

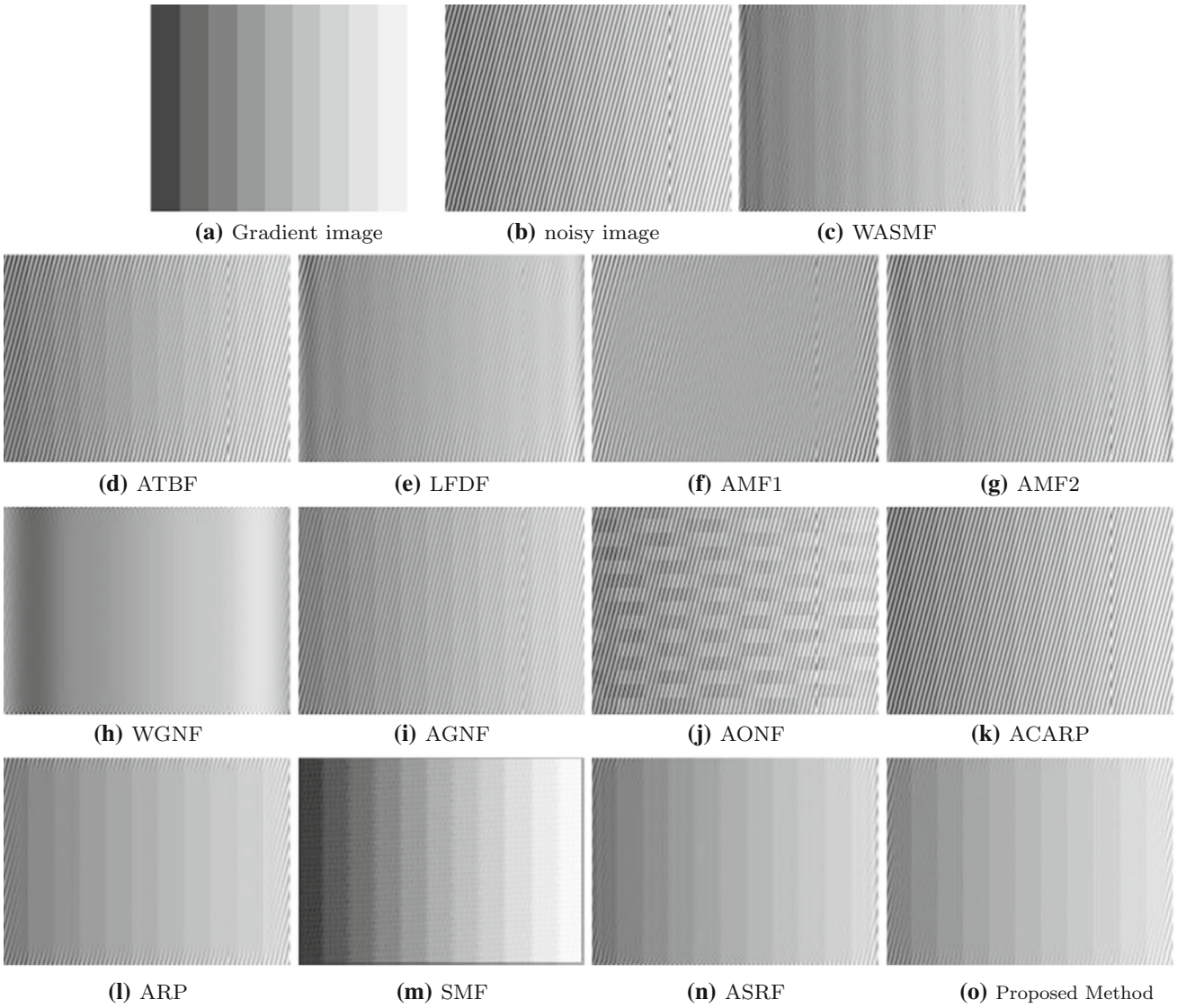


Fig. 5 Visual comparisons for different denoising algorithms for a synthetically corrupted tonal gradient image by periodic noise of strength $a = 1.3(\epsilon = 0.08, power = 10, \theta_1 = 0.5, \theta_2 = 9 \times 10^{(-4)})$

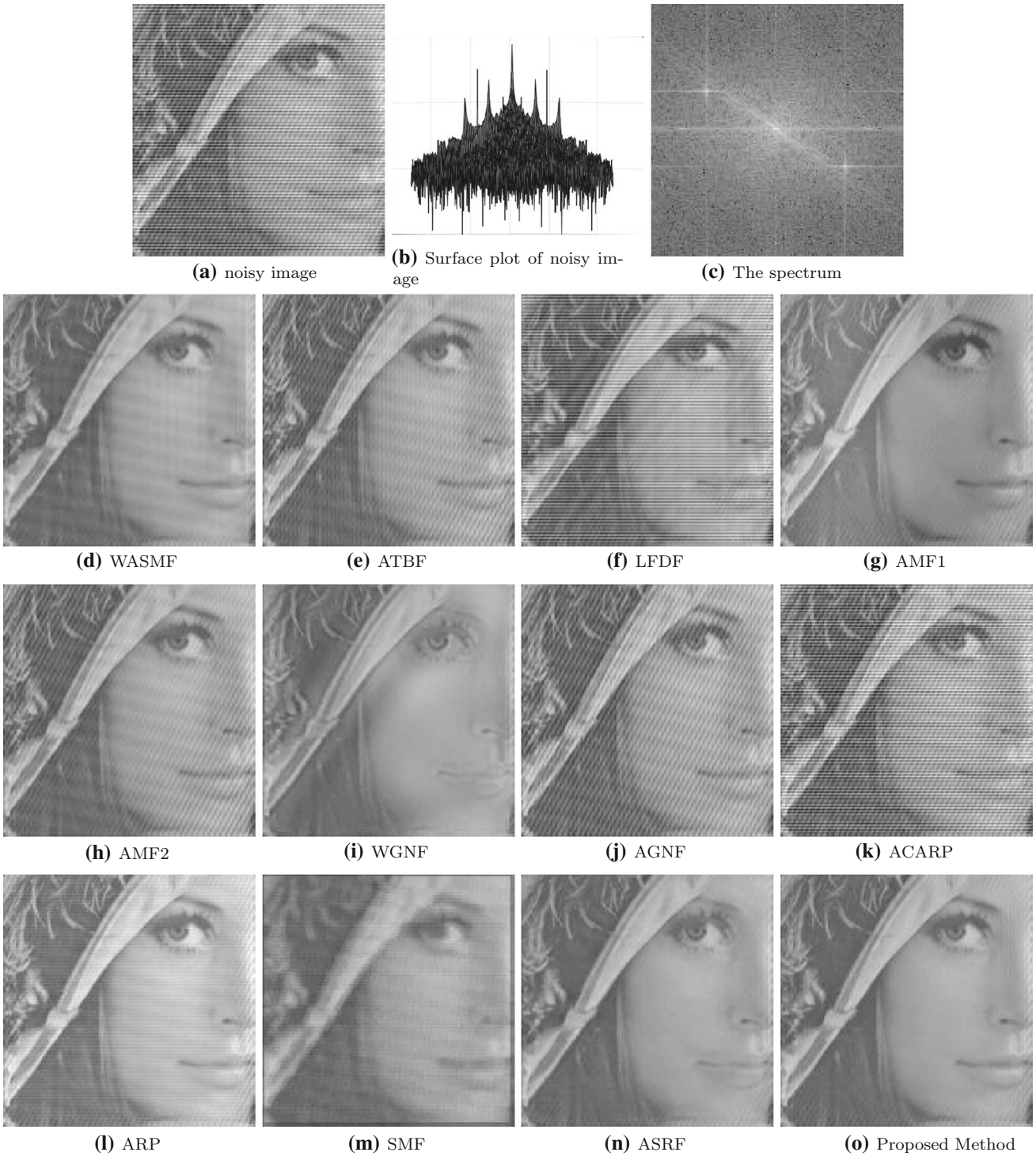


Fig. 6 Visual comparisons for different denoising algorithms for a synthetically corrupted sample image Lena by multi-frequency periodic noise($\epsilon = 0.08$, $power = 10, \theta_1 = 0.5, \theta_2 = 9 \times 10^{(-4)}$)

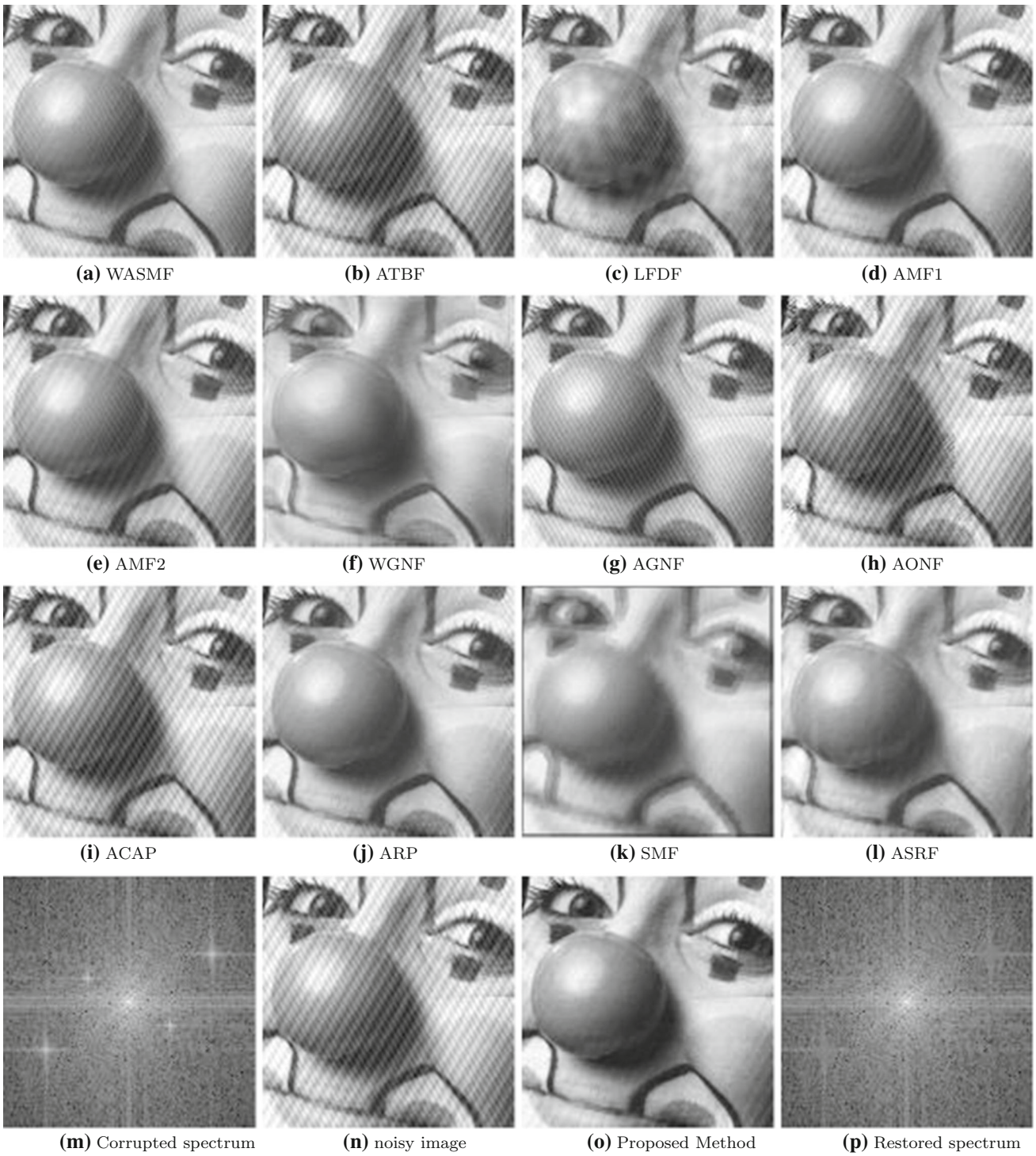


Fig. 7 Visual comparisons for different denoising algorithms for a non-synthetically corrupted image Clown ($\epsilon = 0.08, power = 10, \theta_1 = 0.5, \theta_2 = 9 \times 10^{(-4)}$)

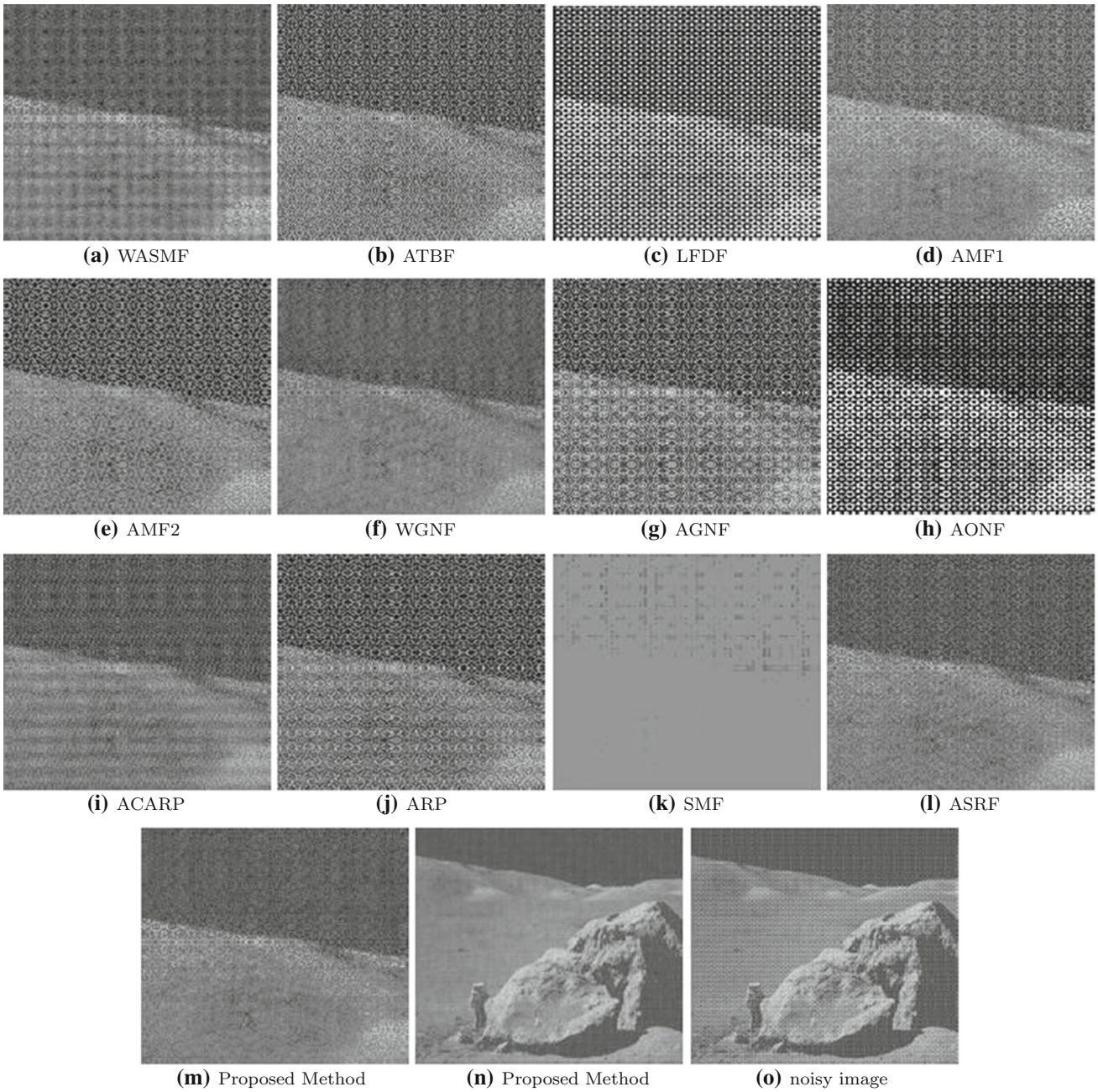
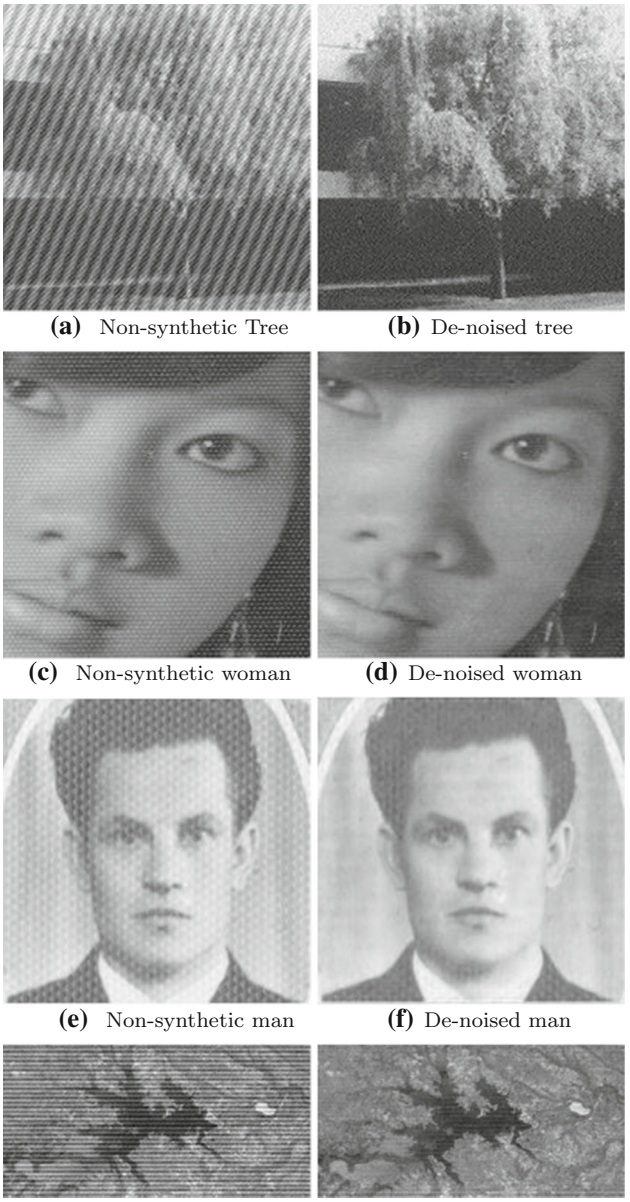


Fig. 8 Visual comparisons for different denoising algorithms for a non-synthetically corrupted image Moonlanding ($\epsilon = 0.08$, $power = 10$, $\theta_1 = 0.5$, $\theta_2 = 9 \times 10^{(-4)}$)



(a) Non-synthetic Tree **(b)** De-noised tree

(c) Non-synthetic woman **(d)** De-noised woman

(e) Non-synthetic man **(f)** De-noised man

(g) Non-synthetic Detector striping **(h)** De-noised Detector striping

Fig. 9 Real images corrupted by different non-synthetic periodic noise structures, and restored images using proposed method ($\epsilon = 0.08$, $power = 10, \theta_1 = 0.5, \theta_2 = 9 \times 10^{(-4)}$)

6 Conclusion

There is a various source of periodic noise and many studies have attempted to remove it from the images. However, there are still challenges about adaptive detection of noise frequencies and appropriate attenuation. This study showed that adaptive detection can be achieved by a compression method in which the content loss takes place in high-frequency positions. Moreover, in such a case decompressed result is smooth. However, the current findings suggest that F-transform smoothing property can be used as an attenuation factor for better restoration of noise frequencies. In this research, fuzzy partitioning was proposed in polar coordinates for the operation in the frequency domain. Experimental results in terms of subjective and objective metrics depicted that the performance of the proposed method in restoring periodic noise corrupted images is better as compared to the state-of-the-art algorithms used in the comparative study. Further work will be directed to the improvement of the developed solution. It is difficult to discriminate between noise peaks and spectrum peaks caused by spatially localized textures or repetitive structures. Furthermore, providing a dual-domain approach is one of the subjects of our future research. Further still, the proposed method is not a fully automatic method due to several parameters. As for the model to automate the parameters can be attempted as another future work by incorporating fuzzy, neural network, and optimization techniques.

References

1. Aizenberg, I., Butakoff, C.: Frequency domain median like filter for periodic and quasi-periodic noise removal. SPIE Proc. **4667**, 181–191 (2002)
2. Aizenberg, I., Butakoff, C.: A windowed Gaussian notch filter for quasi-periodic noise removal. Image Vis. Comput. **26**(10), 1347–1353 (2008). https://doi.org/10.1016/j.imavis.2007.08.011
3. Aizenberg, I., Butakoff, C., Astola, J., Egiazarian, K.: Nonlinear frequency domain filter for quasi periodic noise removal. In: International TICSP Workshop on Spectral Methods and Multirate Signal Processing, pp. 147–153. Toulouse, France (2002)
4. Al-Najjar, Y., Soong, D.: Comparison of image quality assessment: PSNR, HVS, SSIM, UIQI. Int. J. Sci Eng. Res. **3**(8), 1–5 (2012)
5. Atul, R.: An Empirical Study of Periodic Noise Filtering in Fourier Domain: An Introduction to Novel Autonomous Periodic Noise Removal Algorithms, 1st edn. Lap Lambert Academic Publishing, Latvia, European Union (2013)
6. Chakraborty, D., Chakraborty, A., Banerjee, A., Chaudhuri, S.R.B.: Automated spectral domain approach of quasi-periodic denoising in natural images using notch filtration with exact noise profile. IET Image Process. **12**(7), 1150–1163 (2018). https://doi.org/10.1049/iet-ipr.2017.0307
7. Chakraborty, D., Tarafder, M.K., Banerjee, A., Chaudhuri, S.R.B.: Gabor-based spectral domain automated notch-reject filter for quasi-periodic noise reduction from digital images. Multimed. Tools Appl. **78**(2), 1757–1783 (2019). https://doi.org/10.1007/s11042-018-6194-z
8. Chakraborty, D., Tarafder, M.K., Chakraborty, A., Banerjee, A.: A proficient method for periodic and quasi-periodic noise fading using spectral histogram thresholding with Sinc restoration filter. Aeu-Int. J. Electron. Commun. **70**(12), 1580–1592 (2016). https://doi.org/10.1016/j.aeue.2016.09.003
9. Chang, Y., Yan, L., Fang, H., Liu, H.: Simultaneous destriping and denoising for remote sensing images with unidirectional total variation and sparse representation. IEEE Geosci. Remote Sens. Lett. **11**(6), 1051–1055 (2014). https://doi.org/10.1109/LGRS.2013.2285124
10. Chang, Y., Yan, L., Wu, T., Zhong, S.: Remote sensing image stripe noise removal: from image decomposition perspective. IEEE

Trans. Geosci. Remote Sens. **54**(12), 7018–7031 (2016). https://doi.org/10.1109/TGRS.2016.2594080
11. Cooley, J., Lewis, P., Welch, P.: Historical notes on the fast Fourier transform. IEEE Trans. Audio Electroacoust. **15**(2), 76–79 (1967). https://doi.org/10.1109/TAU.1967.1161903
12. Dutta, S., Mallick, A., Roy, S., Kumar, U.: Periodic noise recognition and elimination using RFPCM clustering. In: International Conference on Electronics and Communication Systems, pp. 1–5. Coimbatore (2014). https://doi.org/10.1109/ECS.2014.6892633
13. Feuerstein, D., Parker, K.H., Boutelle, M.: Practical methods for noise removal: applications to spikes, nonstationary quasi-periodic noise, and baseline drift. Anal. Chem. **81**(12), 4987–4994 (2009). https://doi.org/10.1021/ac900161x
14. He, W., Zhang, H., Zhang, L., Shen, H.: Total-variation-regularized low-rank matrix factorization for hyperspectral image restoration. IEEE Trans. Geosci. Remote Sens. **54**(1), 178–188 (2016). https://doi.org/10.1109/TGRS.2015.2452812
15. Holčapek, M., Tichý, T.: A smoothing filter based on fuzzy transform. J. Fuzzy Sets Syst. **180**(1), 69–97 (2011). https://doi.org/10.1016/j.fss.2011.05.028
16. Hurtik, P., Perfilieva, I.: Image compression methodology based on fuzzy transform using block similarity. In: 8th Conference of the European Society for Fuzzy Logic and Technology, pp. 521–526. Atlantis Press (2013). https://doi.org/10.2991/eusflat.2013.79
17. Ji, Z., Liao, H., Zhang, X., Wu, Q.: Simple and efficient soft morphological filter in periodic noise reduction. In: IEEE Region 10 Conference TENCON, pp. 1–4 (2006). https://doi.org/10.1109/TENCON.2006.343712
18. Koukou, V., Martini, N., Michail, C., Sotiropoulou, P., Fountzoula, C., Kalyvas, N., Kandarakis, I., Nikiforidis, G., Fountos, G.: Dual energy method for breast imaging: A simulation study. Comput. Math. Methods Med. **2015**(574238), 1–8 (2015). https://doi.org/10.1155/2015/574238
19. Laus, F., Pierre, F., Steidl, G.: Nonlocal myriad filters for Cauchy noise removal. J. Math. Imaging Vis. **60**(8), 1324–1354 (2018). https://doi.org/10.1007/s10851-018-0816-y
20. Martino, F.D., Loia, V., Perfilieva, I., Sessa, S.: An image coding/decoding method based on direct and inverse fuzzy transforms. Int. J. Approx. Reason. **48**, 110–131 (2008). https://doi.org/10.1016/j.ijar.2007.06.008
21. Moallem, P., Behnampour, M.: Adaptive optimum notch filter for periodic noise reduction in digital images. Amirkabir Int. J. Elect. Electron. Eng. **42**(1), 1–7 (2010). https://doi.org/10.22060/eej.2010.94
22. Moallem, P., Masoumzadeh, M., Habibi, M.: A novel adaptive Gaussian restoration filter for reducing periodic noises in digital image. Signal, Image Video Process. **9**(5), 1179–1191 (2013). https://doi.org/10.1007/s11760-013-0560-0
23. Novák, V., Perfilieva, I., Holčapek, M., Kreinovich, V.: Filtering out high frequencies in time series using F-transform. Inf. Sci. **274**, 192–209 (2014). https://doi.org/10.1016/j.ins.2014.02.133
24. Novak, V., Perfilieva, I., Dvorak, A.: Insight into Fuzzy Modeling. Wiley and Sons, Hoboken, New Jersey (2016)
25. Patro, P.P., Panda, C.S.: A review on: Noise model in digital image processing. Int. J. Eng. Sci. Res. Technol. **5**(1), 891–897 (2016)
26. Pavliska, V.: Computational complexity of discrete fuzzy transform. Technical Report vol. 113, Publication of institute for research and applications of fuzzy modeling, University of Ostrava (2006)
27. Perfilieva, I.: Fuzzy transforms: Theory and applications. Fuzzy Sets Syst. **157**, 993–1023 (2006). https://doi.org/10.1016/j.fss.2005.11.012
28. Perfilieva, I., Hodáková, P.: Fuzzy and Fourier transforms. In: 7th Conference of the European Society for Fuzzy Logic and Technology, pp. 521–526. Atlantis Press, France (2011). https://doi.org/10.2991/eusflat.2011.12
29. Perfilieva, I., Hodáková, P., Hurtík, P.: Differentiation by the F-transform and application to edge detection. Fuzzy Sets Syst. **288**, 96–114 (2016). https://doi.org/10.1016/j.fss.2014.12.013
30. Perfilieva, I., Vlašánek, P.: Image reconstruction by means of F-transform. Knowl.-Based Syst. **70**, 55–63 (2014). https://doi.org/10.1016/j.knosys.2014.04.007
31. Perfilieva, I., Vlašánek, P.: Total variation with nonlocal FT-Laplacian for patch-based inpainting. Soft Comput. **23**(6), 1833–1841 (2019). https://doi.org/10.1007/s00500-018-3589-8
32. Rakwatin, P., Takeuchi, W., Yasuoka, Y.: Restoration of Aqua MODIS band 6 using histogram matching and local least squares fitting. IEEE Trans. Geosci. Remote Sens. **47**(2), 613–627 (2009). https://doi.org/10.1109/TGRS.2008.2003436
33. Schowengerdt, R.: Remote Sensing: Models and Methods for Image Processing, 3rd edn. Academic Press, Waltham (2007)
34. Schuster, T., Sussner, P.: An adaptive image filter based on the fuzzy transform for impulse noise reduction. Soft Comput. **21**(13), 3659–3672 (2017). https://doi.org/10.1007/s00500-017-2669-5
35. Smith, R.D.: Digital Transmission Systems, 3rd edn. Heidelberg Springer Science and Business Media, Berlin (2012)
36. Sur, F.: An a-contrario approach to quasi-periodic noise removal. In: Proceeding of the International Conference on IEEE Image Processing, pp. 3841–3845. , Quebec City, Canada (2015). https://doi.org/10.1109/ICIP.2015.7351524
37. Sur, F.: A non-local dual-domain approach to cartoon and texture decomposition. IEEE Trans. Image Process. **28**(4), 1882–1894 (2019). https://doi.org/10.1109/TIP.2018.2881906
38. Sur, F., Grediac, M.: Automated removal of quasiperiodic noise using frequency domain statistics. J. Electron. Imaging **24**(1), 1–19 (2015). https://doi.org/10.1117/1.JEI.24.1.013003
39. Varghese, J.: Adaptive threshold based frequency domain filter for periodic noise reduction. Aeu-Int. J. Electron. Commun. **70**(12), 1692–1701 (2016). https://doi.org/10.1016/j.aeue.2016.10.008
40. Varghese, J., Subash, S., Tairan, N.: Fourier transform-based windowed adaptive switching minimum filter for reducing periodic noise from digital images. IET Image Process. **10**(9), 646–656 (2016). https://doi.org/10.1049/iet-ipr.2015.0750
41. Varghese, J., Subash, S., Tairan, N., Babu, B.: Laplacian-based frequency domain filter for the restoration of digital images corrupted by periodic noise. Can. J. Electr. Comput. Eng. **39**(2), 82–91 (2016). https://doi.org/10.1109/CJECE.2015.2490598
42. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004). https://doi.org/10.1109/TIP.2003.819861

Najmeh Alibabaie received her B.Sc. degree in software engineering from the Jahad Daneshgahi University of Esfahan, Iran, in 2009 and M.Sc. degree in Decision science and knowledge engineering from the Kharazmi University of Tehran, Iran, in 2013. She is currently a Ph.D. candidate in Computer Engineering at Yazd University of Yazd, Iran. Her current research interests include image processing, fuzzy logic, evolutionary computation, pattern recognition and deep learning.

AliMohammad Latif obtained his first degree in Electronic Engineering from Isfahan University of Technology in 1993 and his M.Sc. in Electronic Engineering from Amirkabir University of Technology in 1996. In 2001, he joined the academic staff at electrical engineering department of Yazd University. He worked as a Ph.D. candidate at the University of Isfahan where he obtained his Ph.D. In 2011, he joined the academic staff at computer department of Yazd University.