

Color and direction-invariant nonlocal self-similarity prior and its application to color image denoising

Qi XIE¹, Qian ZHAO¹, Zongben XU¹ & Deyu MENG^{1,2*}

¹*School of Mathematics and Statistics, Xi'an Jiaotong University, Shaanxi 710049, China;*

²*Macau Institute of Systems Engineering, Macau University of Science and Technology, Macau 999078, China*

Received 4 January 2020/Revised 9 April 2020/Accepted 14 April 2020/Published online 2 November 2020

Abstract Nonlocal self-similarity (NSS) is one of the most commonly used priors in computer vision and image processing. It aims to make use of the fact that a natural image often possesses many repetitive local patterns, and thus a local image patch always has many similar patches across the image. Through compensatively integrating these similar image patches, their insightful patterns hiding under corrupted noises can be intrinsically extracted. However, for using this prior knowledge, current methods search the similar patches by using simple block matching strategy with Euclidean distance, which largely ignores those patches containing similar local patterns but with different texture-directions and colors. To more sufficiently explore similar patches over an image, in this paper, we propose two new representations for image patches, which facilitate an easy NSS prior for measuring direction-invariant and color-invariant nonlocal self-similarity possessed by image patches. Specifically, based on this prior term, we formulate the color image denoising problem as a concise Bayesian posterior estimation framework, and design an efficient expectation-maximization (EM) algorithm to solve it. A series of experiments implemented on simulated and real noisy color images demonstrate the superiority of the proposed method as compared with the state-of-the-arts both visually and quantitatively, verifying the potential usefulness of this new NSS prior.

Keywords color image denoising, nonlocal self-similarity, Gaussian mixture model, maximum a posterior (MAP) model, EM algorithm

Citation Xie Q, Zhao Q, Xu Z B, et al. Color and direction-invariant nonlocal self-similarity prior and its application to color image denoising. *Sci China Inf Sci*, 2020, 63(12): 222101, <https://doi.org/10.1007/s11432-020-2880-3>

1 Introduction

In most conventional methods designed for various image processing tasks, the key point we need to consider is to finely explore and encode the general prior structure knowledge underlying images. One of the most commonly utilized priors in current research is the so called nonlocal self-similarity (NSS) prior [1], referring to the fact that a natural image (both gray-scale and color ones) often has many repetitive local patterns, and thus a local image patch always has some similar patches across the image. Through compensatively integrating these similar image patches, the negative effects brought by noises can be effectively suppressed and their insightful patterns can be intrinsically extracted. By taking advantage of such NSS prior, various methods have been proposed for different image processing tasks. Let's take the well known image denoising task as an example. Multiple effective denoising methods have been designed for gray-scale images by considering this NSS prior and achieved excellent effect, like nonlocal means [1], BM3D [2], WNNM [3], and so on [4–7]. Afterwards, as natural extensions of the BM3D and WNNM methods, the state-of-the-art methods for the color image denoising task, CBM3D [8]

* Corresponding author (email: dymeng@xjtu.edu.cn)

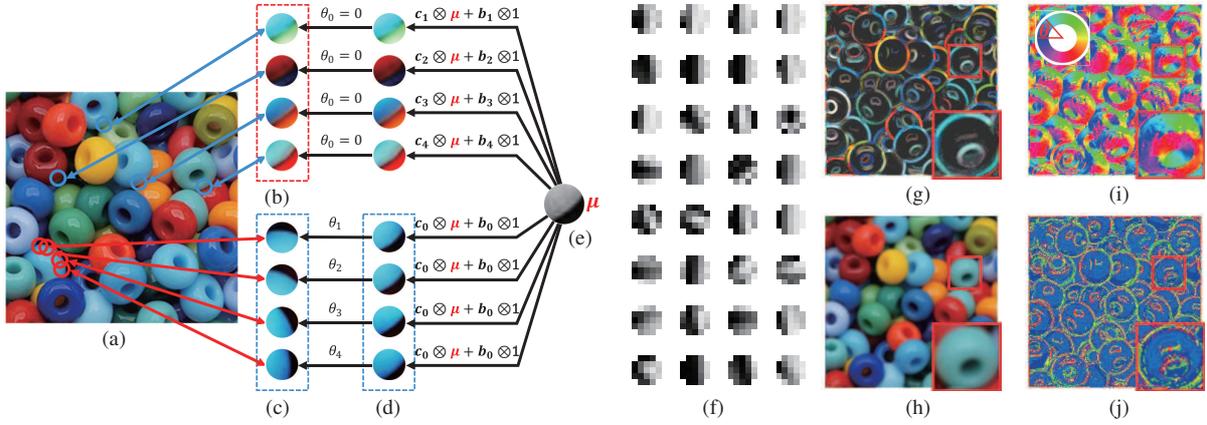


Figure 1 Illustration of the CDI-NSS in a color image. (a) A natural color image. (b) Image patches with similar structure but different colors. (c) Image patches with similar structure but different texture directions. (d) Angel-adjusting results of patches in (c). (e) The core structural patch of image patches in (b) and (d). (f) Several core patches obtained by adopting the proposed method to the clean image in (a). (g), (h) Illustrations of color vectors \mathbf{c}_s and \mathbf{b}_s estimated by the proposed method. Since they are $H \times W$ 3-dimensional vectors, we illustrate them as $H \times W \times 3$ RGB images. (i) Rotation angles $\theta \in \mathbb{R}^{H \times W}$ estimated by the proposed method, where the correspondence between angle and color is shown at the top left corner. (j) The class labels of patches obtained by the proposed method, where different color refer to different class label.

and MCWNNM [9], respectively, also fully capitalize on such prior on 3-D color local patches instead of 2-D gray-scale ones.

However, for using this prior knowledge, current methods search the similar patches for a local patch by using the simple block matching strategy with Euclidean distance imposed on the image pixel values. Such simple regime largely under-estimates the insightful complexity and variety of similar patches on an image, especially a color image. For example, most previous NSS techniques neglect the following two prior similarity knowledge intrinsically possessed by a (color) image.

- Similarity among image patches with similar structure but different texture directions. In a natural image, there always exist image patches with similar structural configurations while with different directions. This means that only after a proper angel adjusting they can be appropriately measured by block matching, as can be easily seen in Figure 1(c). It should be noted that such similarity exists in both gray-scale and color images, while in color images it is more expected to extract such similarity since more channel information can be used.

- Similarity among image patches with similar structure but different color. A color image might possibly contain many local patches with similar texture/edge structures, while in different colors, as clearly shown in Figure 1(b). Traditional blocking matching way in NSS techniques cannot exploit such similarity knowledge since those patches are with relatively large Euclidean distances caused by their different RGB values.

Actually, in more general cases, for a local patch in a (color) image, there are always many patches sharing similar intrinsic structures while with both different colors and structure directions from it, as clearly depicted in Figure 1. Making a similarity measure color-invariant and direction-invariant among patches is thus very helpful to facilitate finding more insightfully similar local patch groups and further prompt the performance of subsequent image processing tasks. For convenience, we call such expected similarity as color and direction invariant nonlocal self-similarity, or CDI-NSS briefly.

To the best of our knowledge, the previous image processing techniques have not yet considered such CDI-NSS priors in their methods, albeit the general NSS prior has been commonly used in various applications. Actually, such CDI-NSS is not easy to be mathematically formulated in a concise way. For example, to measure the CDI-NNS of two patches with different texture directions, one generally needs to first rotate a local patch possibly close to the other one, and do interpolation to image grids on the rotated pixels (possibly not located in the image grids). Such implementation is toilsome and with relatively high computational cost, and thus is generally not employed in practice.

To the aforementioned issue, this paper attempts to develop two concise formulations to facilitate a direction-invariant and a color-invariant similarity measure on local patches of an image (the latter one is especially considered for a color image). Specifically, we consider the color image denoising task as a typical example, and combine the proposed prior term to construct a new maximum a posterior (MAP) model for the task. In summary, this work mainly contains the following three-fold contributions.

Firstly, we propose a new representation manner $G^{\text{DI}}(\mathbf{a}; \theta)$ for a local image patch \mathbf{Z} under the polynomial fitting bases, where \mathbf{a} denotes the polynomial coefficients and θ denotes an angle to rotate. Such a representation can help easily measure the direction-invariant similarity among different patches of an image. Specifically, if two patches can be approximately represented as $G^{\text{DI}}(\mathbf{a}; \theta_1)$ and $G^{\text{DI}}(\mathbf{a}; \theta_2)$, respectively, we then know the patches are with similar shapes after a rotation operators imposed on them. Besides, to alleviate the unexpected edge effect on similar patches with different directions, we first suggest to use the circle patches instead of the traditional square ones, which further helps improve the searching accuracy of direction-invariant similar patches.

Secondly, we design a new representation scheme $G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}, \mathbf{b})$ for representing a color image patch, where \mathbf{c} and \mathbf{b} denote the color-rendering and color-shifting transformation parameters, of the transformed patch $G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}, \mathbf{b})$, respectively, against its intrinsic structure $\boldsymbol{\mu}$. Specifically, this representation helps easily measure color-invariant similarity between two patches. That is, if two patches are with expressions $G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}_1, \mathbf{b}_1)$ and $G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}_2, \mathbf{b}_2)$, we can know that the two patches share a similar intrinsic structure depicted as $\boldsymbol{\mu}$, with only different imposed colors.

Thirdly, through combinationally considering two aforementioned representations for image patches, we formulate a concise prior form to encode CDI-NSS structure inside local patches of a color image. A complete MAP framework can then be naturally constructed for the color image denoising task. An expectation-maximization (EM) algorithm [10] is readily designed for solving the model. Each step in the algorithm can be efficiently implemented, and all parameters involved in the model, including the rotation angel (θ), the color transformation parameters (\mathbf{c} , \mathbf{b}), the core structure patches ($\boldsymbol{\mu}$) and to-be-recovered image can be easily solved (Figure 1(f)–(j) show examples for easy understanding of these parameters). Experiments on simulated and real noisy color images demonstrate the superiority of the proposed method as compared with the state-of-the-art methods along this line, substituting the potential usefulness of the proposed NSS prior representations.

The paper is organized as follows. Section 2 introduces the related works on generally used priors for image processing and typical image denoising methods. Section 3 proposes the necessary notations and preliminaries utilized in describing our designed priors and color image denoising method. Section 4 provides the constructed CDI-NSS representations for measuring similarity between local patches. Section 5 presents the color image denoising method by employing the designed prior representations. Section 6 demonstrates the experimental results for performance evaluation of the proposed method as compared with other competing ones. The paper is finally concluded.

2 Related work

Designing rational prior terms is one of the most important and necessary steps in handling conventional image processing tasks, especially for most low-level computer vision ones. We take the typical image restoration task as an example. From a probability-based perspective, this task can be formulated as a Bayesian posterior estimation problem: $p(x|y) = p(y|x)p(x)/p(y)$, where $p(y|x)$ is the data model, $p(x)$ is the prior distribution of the to-be-estimated image x and $p(y)$ is a constant once the input image y is given. As aforementioned, the most general technique for the task is to design proper prior forms to help possibly deliver the intrinsic structure underlying the to-be-estimated image. Most representative priors are described as follows. The classic total variation (TV) based method assumes that natural image gradients are depicted as heavy-tail distributions, and models this prior by Laplacian or hyper-Laplacian distributions [11, 12]. Many statistical prior models assumed on wavelet coefficients have also been proposed, such as generalized Gaussian [13] and Gaussian scale mixture [14] models. By exploiting

the NSS prior, nonlocal means [1] and nonlocal regularization [15] methods achieve good performance for image denoising. Other methods exploiting NSS in different ways also achieve great success for various image restoration tasks [5–7]. The BM3D method [2] is a typical example along this line, which is verified to be effective and has become a benchmark in image denoising. Zoran and Weiss [16] modeled the categorical property of patches in clean natural image patches using mixture of Gaussian models, and reconstructed the latent image by maximizing the expected patch log likelihood (EPLL). Recently, low-rank approximation methods have exhibited exciting performance on image recovery [4, 17, 18]. For example, WNNM [3] adopted a weighted nuclear norm minimization on the grouped similar patches and has obtained excellent denoising and super-resolution results. Some discriminative denoising methods have also been developed by learning discriminative priors from pairs of clean and noisy images [19, 20].

Since we adopt color image denoising to evaluate the performance of the proposed method, we also introduce the related developments of this task. RGB color image denoising is the most direct extension of the gray-scale image denoising, aiming to recover the latent clean RGB color image from its noise-corrupted version. In recent years, there are many denoising methods raised for handling gray-scale image, by employing various techniques [20, 21], including dictionary learning [22], low-rank approximation [3], collaborative filtering [2], and most recently, deep neural networks [19, 23–25]. As a comparison, the techniques specifically designed for color image denoising have been attracted relatively less attention since those gray-scale image techniques are supposed can be easily extended to color cases.

Specifically, nowadays there are mainly three categories of techniques utilized for constructing unsupervised color image denoising models [9, 26–32]. The first category is to easily apply a gray-scale image denoising method to each channel of an RGB image. The shortcoming of this approach is that it ignores the spectral correlation among RGB channels of a color image, while such prior structure is one of the most insightful characteristics possessed by color images beyond gray-scale ones. This line of methods thus always cannot perform sufficiently satisfactory for the task. The second category of methods is to transform the RGB image into a less correlated color space, such as YCbCr [33], and then perform denoising in each channel of the transformed space [8, 34]. The CBM3D method [8] is a representative one among these methods, which is a natural extension of BM3D [2]. In CBM3D method, an RGB image is first transformed into a luminance-chrominance space (e.g., YCbCr) and patch groups for all three channels are obtained by the group matching on only the luminance channel (Y channel). In this way, CBM3D takes effect on using the NSS prior knowledge and reduce the patch color effect to a certain extent, while still cannot take full use of the CDI-NSS.

The state-of-the-art model for unsupervised color image denoising is represented by the third category of methods along this research line, which intrinsically takes spectral correlation among RGB channels of a color image into account [27–29]. The commonly used strategy is to concatenate the RGB channels by rearranging a local 3-D patch of a color image as a long vector and performing low-rank approximation on similar patch groups across the image space. For example, MCWNNM [9] extends the weighted nuclear norm minimization designed on gray-scale patch groups to color patch groups, and achieves good results. In this manner, the correlation among RGB channels can be finely utilized, naturally leading to the better performance of this approach. However, these methods still have not considered full use the intrinsic CDI-NSS prior structures underlying a color image. There is thus still room to further improve performance for the current methods along this line.

3 Notations and preliminaries

In this paper, we denote scalar, vector, matrix, and tensor as non-bold lower case, bold lower case, bold upper case, and calligraphic upper case letters, respectively.

A tensor of order N is denoted as $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$. An element of \mathcal{A} is denoted as $a_{i_1, \dots, i_n, \dots, i_N}$ where $1 \leq i_n \leq I_n$. We denote the vectorization of tensor \mathcal{A} (or matrix \mathbf{A}), as $\mathbf{a} = \text{vec}(\mathcal{A})$ (or $\mathbf{a} = \text{vec}(\mathbf{A})$ for matrix case). Specially, the i th elements of a vector \mathbf{a} is denoted as a_i . We denote identity matrix as \mathbf{I} . We further denote the Frobenius norm a tensor \mathcal{X} as $\|\mathcal{X}\|_F = \sqrt{\sum_{i_1, \dots, i_N} x_{i_1, \dots, i_N}^2}$.

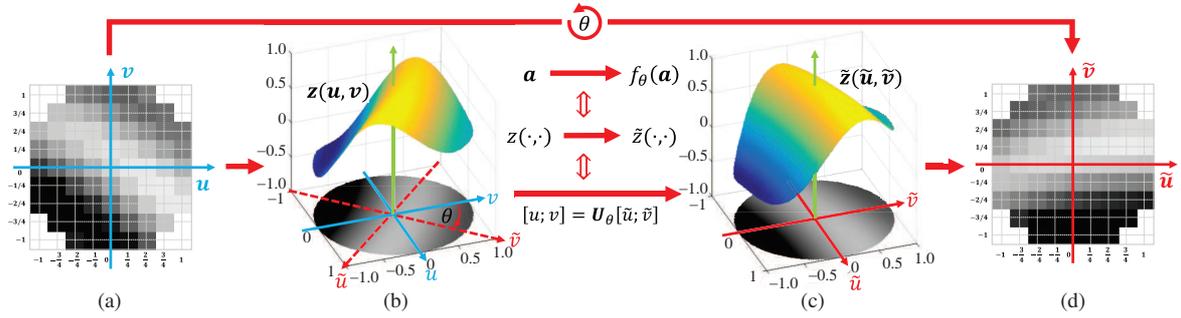


Figure 2 (a) An example of circle-like patch with diameter $m = 9$ and $p = 4$. (b) Illusion of fitting function $z(\cdot, \cdot)$, whose polynomial coefficients is \mathbf{a} . (c) Illusion of function $\tilde{z}(\cdot, \cdot)$, whose polynomial coefficients is $f_\theta(\mathbf{a})$. (d) Rotated result of the patch shown in (a).

The product between matrices can be generalized to the product of a tensor and a matrix. The mode- n product of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ by a matrix $\mathbf{B} \in \mathbb{R}^{J_n \times I_n}$, denoted by $\mathcal{A} \times_n \mathbf{B}$, is an N -order tensor $\mathcal{C} \in \mathbb{R}^{I_1 \times \dots \times I_n \times \dots \times I_N}$ with entries: $c_{i_1 \times \dots \times i_{n-1} \times j_n \times i_{n+1} \times \dots \times i_N} = \sum_{i_n} a_{i_1, \dots, i_n, \dots, i_N} b_{j_n i_n}$.

We denote the Kronecker product between $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $\mathbf{B} \in \mathbb{R}^{p \times q}$ by

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & \dots & a_{1n}\mathbf{B} \\ \vdots & & \vdots \\ a_{m1}\mathbf{B} & \dots & a_{mn}\mathbf{B} \end{pmatrix}. \quad (1)$$

We further denote the convolution of a filter $\mathbf{F} \in \mathbb{R}^{m \times m}$ and a matrix $\mathbf{X} \in \mathbb{R}^{H \times W}$ by $\mathbf{F} * \mathbf{X} \in \mathbb{R}^{(H-m+1) \times (W-m+1)}$.

4 Constructing representations for measuring direction and color invariant similarities

In this section, we aim to formulate two representations for local patches of an image to easily conduct the direction-invariant and color-invariant similarity among patches, respectively.

4.1 Representation for measuring direction-invariant similarity

We first discuss the case of gray-scale image for facilitating an easy understanding of readers, and then extend it to more general color image case latter. Firstly, we introduce a specific basis set, the polynomial bases, for representing an image patch.

Represent an image patch by polynomial bases. For an image patch $\mathbf{Z} \in \mathbb{R}^{m \times m}$ with its odd number diameter $m = 2p + 1$ ¹⁾, as shown in Figure 2(a) and (b). Our aim is then to achieve the optimal polynomial approximation to represent all image pixels in \mathbf{Z} , i.e., finding a polynomial function that satisfies $z(u_i, v_j) \approx z_{ij}$, where $z(\cdot, \cdot)$ is a polynomial function, z_{ij} is a pixel in \mathbf{Z} and $\mathbf{u} = \mathbf{v} = [-p, -(p-1), \dots, 0, \dots, p-1, p]$ denote the grid axes. For easy calculation, we normalize this axis as $[-1, \frac{-(p-1)}{p}, \dots, 0, \dots, \frac{(p-1)}{p}, 1]$ in the following.

Without loss of generality, we first take the 2-order polynomial bases as an example for easy understanding. Denoting the vector $[u_i; v_j]$ as \mathbf{w}_{ij} , the 2-order polynomial approximation $z(u_i, v_i)$ can be represented as

$$\begin{aligned} z_{ij} \approx z(u_i, v_j) &= a_{11}u_i^2 + a_{12}u_iv_j + a_{21}v_iu_j + a_{22}v_j^2 + a_1u_i + a_2v_j + a_0 \\ &= \mathbf{w}_{ij}^T \mathbf{A}_2 \mathbf{w}_{ij} + \mathbf{a}_1^T \mathbf{w}_{ij} + a_0, \end{aligned} \quad (2)$$

1) If m is a even number, the deduction is the same except that $m = 2p$.

where $\mathbf{A}_2 = [a_{11}, a_{12}; a_{21}, a_{22}]$, $\mathbf{a}_1 = [a_1; a_2]$ and a_0 are the coefficients imposed on the 2-order, 1-order, and 0-order polynomial basis functions, respectively. Note that albeit having manifold values only on its grids position (u_i, v_j) ($i, j = 1, \dots, 2p + 1$), after we calculate all representation coefficients \mathbf{A}_2 , \mathbf{a}_1 , and a_0 , we can then get the approximate expression for the latent manifold and attain values in any positions on the manifold.

It is easy to see that when a patch contains few textures/edges and with few variations, it can be finely approximated by a 0-order polynomial expression; when it contains even color change from one side to the other, it can then be well expressed by a 1-order one. 2-order one can evidently represent patches with more complex and various configurations.

By vectoring both sides of Eq. (2), it is easy to find that it is equitant to the following concise form:

$$\mathbf{z} \approx \mathbf{D}\mathbf{a}, \tag{3}$$

where $\mathbf{D} \in \mathbb{R}^{m^2 \times K}$, K is the total number of polynomial basis functions with order less than 2,

$$\mathbf{z} = \text{vec}(\mathbf{Z}) = \begin{pmatrix} z_{11} \\ \vdots \\ z_{ij} \\ \vdots \\ z_{mm} \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} u_1u_1 & u_1v_1 & v_1u_1 & v_1v_1 & u_1 & v_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_iu_i & u_iv_j & v_iu_j & v_jv_j & u_i & v_j & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_mu_m & u_mv_m & v_mu_m & v_mv_m & u_m & v_m & 1 \end{pmatrix}, \tag{4}$$

and $\mathbf{a} = [a_{11}; a_{12}; a_{21}; a_{22}; a_1; a_2; a_0] \in \mathbb{R}^7$ denotes the polynomial coefficients.

In general, the r -order polynomial fitting function for an image patch can be similarly constructed as follows:

$$\mathbf{z}(u_i, v_j) = \mathcal{A}_r \times_1 \mathbf{w}_{ij} \cdots \times_r \mathbf{w}_{ij} + \cdots + \mathbf{w}_{ij}^T \mathbf{A}_2 \mathbf{w}_{ij} + \mathbf{a}_1^T \mathbf{w}_{ij} + a_0, \tag{5}$$

where \mathcal{A}_r is a r -order tensor and $\{\mathcal{A}_r; \dots; \mathbf{A}; \mathbf{a}_1; a_0\}$ denotes the polynomial coefficients. By vectoring both sides of (5), we can also transfer (5) into a more concise formulation of (3), with $\mathbf{D} \in \mathbb{R}^{m^2 \times K}$, where K is the total number of polynomial basis functions involved here, and

$$\mathbf{D} = \begin{pmatrix} \mathbf{w}_{11}^T \otimes \cdots \otimes \mathbf{w}_{11}^T & \cdots & \mathbf{w}_{11}^T \otimes \mathbf{w}_{11}^T & \mathbf{w}_{11}^T & 1 \\ \vdots & & \vdots & \vdots & \vdots \\ \mathbf{w}_{ij}^T \otimes \cdots \otimes \mathbf{w}_{ij}^T & \cdots & \mathbf{w}_{ij}^T \otimes \mathbf{w}_{ij}^T & \mathbf{w}_{ij}^T & 1 \\ \vdots & & \vdots & \vdots & \vdots \\ \mathbf{w}_{mm}^T \otimes \cdots \otimes \mathbf{w}_{mm}^T & \cdots & \mathbf{w}_{mm}^T \otimes \mathbf{w}_{mm}^T & \mathbf{w}_{mm}^T & 1 \end{pmatrix}, \tag{6}$$

and $\mathbf{a} = [\text{vec}(\mathcal{A}_r); \dots; \text{vec}(\mathbf{A}); \mathbf{a}_1; a_0] \in \mathbb{R}^K$.

The fitting polynomial coefficients \mathbf{a} for an image patch vector \mathbf{z} can be easily calculated by solving the following optimization problem:

$$\min_{\mathbf{a}} \|\mathbf{z} - \mathbf{D}\mathbf{a}\|_2. \tag{7}$$

Eq. (7) has a closed-form solution $\mathbf{a} = \mathbf{D}^\dagger \mathbf{z}$, where $\mathbf{D}^\dagger \in \mathbb{R}^{K \times m^2}$ denotes the pseudo inverse (more details of pseudo inverse can be found in Appendix A) of \mathbf{D} , which is defined as

$$\mathbf{D}^\dagger = (\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T. \tag{8}$$

It should be noted that the calculation of fitting polynomial coefficients for all patches in an image $X \in \mathbb{R}^{H \times W}$ can be performed efficiently by 2-D convolutions. Denote the k th row of \mathbf{D}^\dagger as $\mathbf{f}_k^T \in \mathbb{R}^{m^2}$, and then we have $a_k = \mathbf{f}_k^T \mathbf{z}$. Since \mathbf{f}_k is an $m^2 \times 1$ vector, it can be reshaped as an $m \times m$ filter \mathbf{F}_k , which is with the same size to the patch \mathbf{Z} , and all \mathbf{F}_k s can be stacked to form a filter tensor $\mathcal{F} \in \mathbb{R}^{m \times m \times K}$.

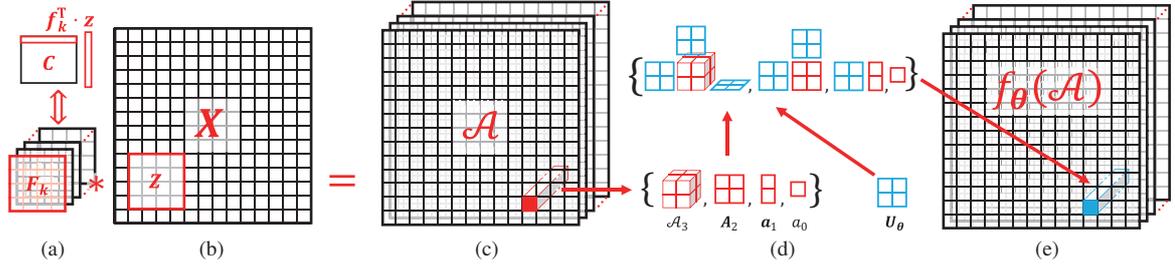


Figure 3 (a) Performing $\mathbf{f}_k^T \mathbf{z}$ for all image patches in an image \mathbf{X} is equivalent to performing the convolution $\mathbf{F}_k * \mathbf{X}$. (b) An image matrix \mathbf{X} . (c) Polynomial coefficients of \mathbf{X} . (d) Illusion of rotating a set of polynomial coefficients by rotation matrix \mathbf{U}_θ , where we take 3 order case as an example. (e) The rotated polynomial coefficients.

Thus, as shown in Figure 3(a)–(c), performing $\mathbf{f}_k^T \mathbf{z}$ for all image patches in a gray-scale image $\mathbf{X} \in \mathbb{R}^{H \times W}$ is equivalent to perform the convolution:

$$\mathbf{A}_k = \mathbf{F}_k * \mathbf{X}, \quad (9)$$

where $\mathbf{A}_k \in \mathbb{R}^{H \times W}$ is the matrix of the k th polynomial coefficients of all the patches in different location in \mathbf{X} ²⁾. Thus, with K 2-D convolutions, we can obtain fitting polynomial coefficients for all the patches in an image.

For an RGB color image $\mathcal{X}^{H \times W \times 3}$, we can perform the convolution defined in (9) for the R, G, B channels, respectively, to obtain the fitting polynomial coefficients for all of the 3 channels. Denote K as the number of polynomial coefficients for each patch, and then the tensor stacked by $\{\mathbf{A}_k^r, \mathbf{A}_k^g, \mathbf{A}_k^b\}_{k=1}^K$ is represented by $\mathcal{A} \in \mathbb{R}^{H \times W \times 3 \times K}$ and the tensor stacked by $\{\mathbf{F}_k\}_{k=1}^K$ is represented by $\mathcal{F} \in \mathbb{R}^{m \times m \times K}$. For convenience, we denote (9) for all $k = 1, 2, \dots, K$ and 3 channels as

$$\mathcal{A} = \mathcal{F} * \mathcal{X}. \quad (10)$$

Image patch rotation under polynomial representation. We then introduce how to use the aforementioned polynomial representation to help measure direction-invariant similarity among image patches.

Considering a set of polynomial coefficients of the image patch \mathbf{Z} in any RGB channel, which can be represented under the polynomial bases as $\mathbf{a} = \{A_r, \dots, A_2, a_1, a_0\}$ as shown in Figure 3(c) and (d). If we assume that another patch $\tilde{\mathbf{z}}$ is similar to \mathbf{z} but with a θ degree rotation from it, $\tilde{\mathbf{z}}$ can then be understood as being with the similar coefficients \mathbf{a} while under rotated axes $[\tilde{u}; \tilde{v}] = \mathbf{U}_\theta[u; v]$, where \mathbf{U}_θ is the rotation matrix with the form

$$\mathbf{U}_\theta = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}, \quad (11)$$

which can be easily understood by observing Figure 2. Moreover, we can prove Theorem 1.

Theorem 1. For two r -order polynomial functions $z_A(u, v)$ and $z_B(u, v)$, with $\{A_r; \dots; A_1; a_0\}$ and $\{B_r; \dots; B_1; b_0\}$ denoting their polynomial coefficients, respectively. When $a_0 = b_0$ and $B_i = A_i \times_1 \mathbf{U}_\theta \cdots \times_i \mathbf{U}_\theta$, for all $i = 1, 2, \dots, r$, $z_B(u, v)$ is the result of counterclockwise rotating $z_A(u, v)$ with θ degrees angle, i.e.,

$$z_B(u, v) = z_A(\tilde{u}, \tilde{v}), \quad (12)$$

where $[\tilde{u}; \tilde{v}] = \mathbf{U}_\theta[u; v]$.

Proof. We denote the vector $[u; v]$ as \mathbf{w} , and denote $[\tilde{u}; \tilde{v}]$ as $\tilde{\mathbf{w}}$. Then, by the definition in (5), we have

$$\begin{aligned} z_B(u, v) &= \mathcal{B}_r \times_1 \mathbf{w} \cdots \times_r \mathbf{w} + \cdots + \mathbf{w}^T \mathcal{B}_2 \mathbf{w} + \mathbf{b}_1^w + b_0 \\ &= (\mathcal{A}_r \times_1 \mathbf{U}_\theta \cdots \times_r \mathbf{U}_\theta) \times_1 \mathbf{w} \cdots \times_r \mathbf{w} + \mathbf{w}^T (\mathbf{U}_\theta^T \mathcal{A}_2 \mathbf{U}_\theta) \mathbf{w} + \mathbf{a}_1^T \mathbf{w} + b_0 \end{aligned}$$

2) We pad edges of \mathbf{X} with zeros for p -pixel width while performing convolution.

$$\begin{aligned}
 &= \mathcal{A}_r \times_1 (\mathbf{U}_\theta \mathbf{w}) \cdots \times_r (\mathbf{U}_\theta \mathbf{w}) \cdots + (\mathbf{U}_\theta \mathbf{w})^\top \mathbf{A}_2 (\mathbf{U}_\theta \mathbf{w}) + \mathbf{a}_1^\top (\mathbf{U}_\theta \mathbf{w}) + a_0 \\
 &= \mathcal{A}_r \times_1 \tilde{\mathbf{w}} \cdots \times_r \tilde{\mathbf{w}} \cdots + \tilde{\mathbf{w}}^\top \mathbf{A}_2 \tilde{\mathbf{w}} + \mathbf{a}_1^\top \tilde{\mathbf{w}} + a_0 \\
 &= z_A(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}).
 \end{aligned} \tag{13}$$

The proof is then completed.

By Theorem 1, we can rotate an image patch by keeping the axes (as well as the polynomial bases) fixed, while transforming the polynomial coefficients of \mathcal{A} as

$$f_\theta(\mathbf{a}) = [\text{vec}(\mathcal{A}_r \times_1 \mathbf{U}_\theta \cdots \times_r \mathbf{U}_\theta); \dots; \text{vec}(\mathbf{U}_\theta \mathbf{A}_2 \mathbf{U}_\theta^\top); \mathbf{U}_\theta \mathbf{a}_1; a_0]. \tag{14}$$

Please see Figure 2 and Figure 3(d) and (e) for easy understanding of such analysis.

We can then get a polynomial representation for a rotated image patch as

$$\tilde{\mathbf{z}} = G^{\text{DI}}(\mathbf{a}; \theta) = \mathbf{D} f_\theta(\mathbf{a}). \tag{15}$$

Such a representation helps conduct a direction-invariant similarity for image patches. That is, if two patches are with expressions $G^{\text{DI}}(\mathbf{a}; \theta_1)$ and $G^{\text{DI}}(\mathbf{a}; \theta_2)$, this means that they possess similar intrinsic structures while with different rotation angles, as those shown in Figure 1(c).

Similarly, for a color image patch case, we can get its coefficients by imposing $f_\theta(\cdot)$ on its three-channel coefficients, respectively. Note that the three channels share the same rotation angle θ , which facilitates a more stable calculation for this parameter through compensating three channel information.

Circle patch. Note that as compared with the conventional square patch, an approximate circle patch tend to better reduce the edge artifacts when rotating a local patch to another. We thus prefer to use circle-like patch in our method as shown in Figure 2(a). Define an indicator matrix $\Omega \in \mathbb{R}^{m \times m}$:

$$\Omega_{ij} = \begin{cases} 1, & \text{if } u_i^2 + v_j^2 \leq R^2, \\ 0, & \text{if } u_i^2 + v_j^2 > R^2, \end{cases} \tag{16}$$

where R is the radius of the circle patch, which can be set as $1 + \frac{1}{2p}$ in our setting, and let $\omega = \text{vec}(\Omega)$, and remove the k th row in \mathbf{D} , as defined in (4), if $\omega_k = 0$. Then we can obtain a $\bar{\mathbf{D}}$ with less rows than \mathbf{D} . By using $\bar{\mathbf{D}}$ to replace \mathbf{D} in the above equations, we are actually performing our method on approximating circle patches. The only difference is that the column number of $\bar{\mathbf{D}}^\dagger$ is less than \mathbf{D}^\dagger , which make its rows cannot be directly reshaped to the filters (i.e., F_k s in (9)). This issue can be easily solved by setting the elements in the filter with indicator (i, j) as zero when $\Omega_{ij} = 0$, and setting the other elements as the correlated elements in the k th row of $\bar{\mathbf{D}}^\dagger$.

For easy notations, we still use the symbols \mathbf{D} instead of $\bar{\mathbf{D}}$ in the following, while it actually means the circle-like patches in this paper.

4.2 Representation for measuring color-invariant similarity

We then formulate the following representation for delivering the color-invariant similarity among color image patches:

$$\mathbf{z} = G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}, \mathbf{b}) = \mathbf{c} \otimes \boldsymbol{\mu} + \mathbf{b} \otimes \mathbf{1}, \tag{17}$$

where \mathbf{z} is a given color image patch, $\boldsymbol{\mu} \in \mathbb{R}^M$ is the intrinsic structure of the color patch, M is the number of pixels in a channel of patch, $G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}, \mathbf{b}) \in \mathbb{R}^{3M}$, $\mathbf{1}$ denotes an M vector with all elements equivalent to 1, and $\mathbf{c}, \mathbf{b} \in \mathbb{R}^3$ denote the color-rendering and color-shifting transformation parameters of the transformed patch $G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}, \mathbf{b})$ against its intrinsic structure $\boldsymbol{\mu}$. For two color patches with expressions $G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}_1, \mathbf{b}_1)$ and $G^{\text{CI}}(\boldsymbol{\mu}; \mathbf{c}_2, \mathbf{b}_2)$, it is seen that they share similar intrinsic structures like textures/edges, while with a color-rendering and color-shifting variations. The rationality of this representation can be easily understood by observing Figure 1(b)–(e), which shows that variations of \mathbf{c}, \mathbf{b} do not affect the similar structures possessed by corresponding local color patches.

5 CDI-NSS-based method for color image denoising

By employing the aforementioned CDI-NSS prior, we aim to propose a new MAP model for color image denoising. The details are introduced as follows.

5.1 MAP model

The first goal now is to embed the CDI-NSS into the general statistical model to form a complete MAP framework for color image denoising.

Firstly, we design a prior term on the to-be-estimated color image $\mathcal{X} \in \mathbb{R}^{H \times W \times 3}$ to reflect the direction and color invariant similarities among image patches. Specifically, for each local patch \mathbf{Z}_{hw} at location (h, w) of \mathcal{X} , its fitting polynomial coefficient vector is $\mathbf{a}_{hw} = (\mathcal{F} * \mathcal{X})_{hw}$. We expect to eliminate the direction variation freedom θ_{hw} and color one \mathbf{c}_{hw} , \mathbf{b}_{hw} of the patch, so as to properly categorize the \mathbf{Z}_{hw} to a group with intrinsic structure shape $\boldsymbol{\mu}_l \in \mathbb{R}^M$, where $l = 1, \dots, L$, L represents the structure group numbers of the entire local patches across the image and M is the number of pixels in a circle-like patch. Through readily utilizing mixture of Gaussians [10] for reflecting such similar patch grouping characteristic, we can construct the following prior term:

$$p(\mathcal{X} | \boldsymbol{\theta}, \boldsymbol{\mu}, \mathbf{c}, \mathbf{b}, \boldsymbol{\pi}, \sigma) = \prod_{hw} \sum_{l=1}^L \pi_l \mathcal{N}(G^{\text{DI}}((\mathcal{F} * \mathcal{X})_{hw}; \theta_{hw}) | G^{\text{CI}}(\boldsymbol{\mu}_l; \mathbf{c}_{hw}, \mathbf{b}_{hw}), \sigma \mathbf{I}), \quad (18)$$

where the operators G^{DI} and G^{CI} are defined in (14) and (17), respectively, σ denotes the variance for all Gaussian components, π_l is the mixing proportion in MoG, and L is the number of Gaussian components as well as that of the intrinsic structure patches $\boldsymbol{\mu}_l$. Note that this prior implies that the patches categorized in one Gaussian share similar structure patch shape $\boldsymbol{\mu}_l$, after removing its direction and color variance. We set the variance of each Gaussian group the same to guarantee a possible balanced clustering effect in-between all groups.

For a noisy input color image \mathcal{Y} , we can further formulate the following likelihood term on it:

$$p(\mathcal{Y} | \mathcal{X}) = \prod_{hw} \prod_{q=1}^3 \mathcal{N}(y_{hwq} | x_{hwq}, \lambda), \quad (19)$$

where \mathcal{X} is the to-be-recovered image, $q \in \{1, 2, 3\}$ denoted the three channels and λ is the variance parameter. Note that we simply assume i.i.d. Gaussian noise on it. In practical cases, however, we can use more flexible and proper noise modeling technique to better rectify this term [35, 36]. Since this paper mainly focused on the prior term, we still use this simple noise assumption throughout the paper.

By combining the prior (18) and the likelihood (40), and adopting non-informative prior to other to-be-estimated parameters, we can then obtain the following posteriori distribution:

$$p(\mathcal{X}, \boldsymbol{\theta}, \boldsymbol{\mu}, \mathbf{c}, \mathbf{b}, \boldsymbol{\pi}, \sigma | \mathcal{Y}) \propto p(\mathcal{X} | \boldsymbol{\theta}, \boldsymbol{\mu}, \mathbf{c}, \mathbf{b}, \boldsymbol{\pi}, \sigma) p(\mathcal{Y} | \mathcal{X}). \quad (20)$$

Then we can estimate the clean image \mathcal{X} as well as other involved parameters by MAP estimation.

5.2 EM algorithm

The EM algorithm [37] can be readily used to estimate the parameters $(\mathcal{X}, \boldsymbol{\theta}, \boldsymbol{\mu}, \mathbf{c}, \mathbf{b}, \boldsymbol{\pi}, \sigma)$ that maximize the posterior (20). The proposed algorithm will iterate between calculating responsibilities of all Gaussian components (E step) and maximizing the parameters of the model (M step).

E step. Assume a latent variable z_{hwl} in the model, with $z_{hwl} \in \{0, 1\}$ and $\sum_{l=1}^L z_{hwl} = 1$, indicating the assignment of the image patch \mathbf{p}_{hw} to a specific component of the mixture. The posterior responsibility of mixture l ($l = 1, 2, \dots, L$) for generating the patch \mathbf{x}_{hw} is then calculated by [37]

$$\gamma_{hwl} = \mathbb{E}\{z_{hwl}\} = \frac{\pi_l \mathcal{N}(G^{\text{DI}}((\mathcal{F} * \mathcal{X})_{hw}; \theta_{hw}) | G^{\text{CI}}(\boldsymbol{\mu}_l; \mathbf{c}_{hw}, \mathbf{b}_{hw}), \sigma \mathbf{I})}{\sum_{l=1}^L \pi_l \mathcal{N}(G^{\text{DI}}((\mathcal{F} * \mathcal{X})_{hw}; \theta_{hw}) | G^{\text{CI}}(\boldsymbol{\mu}_l; \mathbf{c}_{hw}, \mathbf{b}_{hw}), \sigma \mathbf{I})}. \quad (21)$$

M step. The M step maximizes the upper bound of logarithmic posterior function given by the E step, with respect to $\mathcal{X}, \theta, \mu, c, b, \pi, \sigma$ [37]:

$$\begin{aligned} & \mathbb{E}_{\mathcal{Z}} \{ \ln p(\mathcal{X}, \theta, \mu, c, b, \pi, \sigma, \mathcal{Z} | \mathcal{Y}) \} \\ &= -\frac{1}{2\lambda} \|\mathcal{X} - \mathcal{Y}\|_{\mathbb{F}}^2 - \frac{3HW}{2} \ln \sqrt{2\pi}\lambda \\ & \quad \cdot \sum_{hwl} \gamma_{hwl} \left(\ln \pi_l - \frac{M}{2} \ln \sqrt{2\pi}\sigma - \frac{1}{2\sigma} \|\mathbf{D}(f_{\theta_{hw}}((\mathcal{F} * \mathcal{X})_{hw})) - \mathbf{c}_{hw} \otimes \boldsymbol{\mu}_l - \mathbf{b}_{hw} \otimes \mathbf{1}\|_{\mathbb{F}}^2 \right). \end{aligned} \quad (22)$$

An easy way to solve this maximization problem is to alternatively update all the parameters as follows.

Update μ, c, b, σ . By setting the derivative with respect to these parameters to 0, closed-form updates for these parameters can be deduced, respectively. Denote c_{hwq} and b_{hwq} as the q th element in \mathbf{c}_{hw} and \mathbf{b}_{hw} , μ_{lm} as the m th pixel in core patch $\boldsymbol{\mu}_l$, and p_{hwmq} as the m th element in the q th channel of color patch $\mathbf{D}(f_{\theta_{hw}}((\mathcal{F} * \mathcal{X})_{hw}))$. Then the closed-form updates are

$$\begin{aligned} \mu_{lm}^+ &= \sum_{hwq} \gamma_{hwl} c_{hwq} (p_{hwmq} - b_{hwq}) / \sum_{hwq} \gamma_{hwl} c_{hwq}^2, \\ c_{hwq}^+ &= \sum_{lm} \gamma_{hwl} \mu_{lm} (p_{hwmq} - b_{hwq}) / \sum_{lm} \gamma_{lm} \mu_{lm}^2, \\ b_{hwq}^+ &= \sum_{lm} \gamma_{hwl} (p_{hwmq} - c_{hwq} \mu_{lm}) / M, \\ \sigma^+ &= \sum_{hwl} \gamma_{hwl} \|\mathbf{D}(f_{\theta_{hw}}((\mathcal{F} * \mathcal{X})_{hw})) - \mathbf{c}_{hw} \otimes \boldsymbol{\mu}_l - \mathbf{b}_{hw} \otimes \mathbf{1}\|_{\mathbb{F}}^2 / (3HWM). \end{aligned} \quad (23)$$

Update π . Closed-form update for this mixing proportion parameter is in the same formulation as standard EM algorithm for mixture of Gaussians [37]:

$$\pi_l^+ = \sum_{hw} \gamma_{hwl} / (HW). \quad (24)$$

Update θ . For θ_{hw} , by using $\sum_l \gamma_{hwl} = 1$, it is easy to deduce that it can be updated by solving

$$\theta_{hw}^+ = \arg \min_{\theta} Q(\theta) = \arg \min_{\theta} \|\mathbf{D}(f_{\theta}(\mathbf{a}_{hw})) - \mathbf{o}_{wh}\|_{\mathbb{F}}^2, \quad (25)$$

where $\mathbf{a}_{hw} = (\mathcal{F} * \mathcal{X})_{hw}$, $\mathbf{o}_{wh} = \mathbf{c}_{hw} \otimes (\sum_l \gamma_{hwl} \boldsymbol{\mu}_l) - \mathbf{b}_{hw} \otimes \mathbf{1}$, and for a coefficient vector $\mathbf{a} \in \mathbb{R}^K$ whose elements are $[\text{vec}(\mathcal{A}_r); \dots; \text{vec}(\mathbf{A}); \mathbf{a}_1; a_0]$,

$$f_{\theta}(\mathbf{a}) = [\text{vec}(\mathcal{A}_r \times_1 \mathbf{U}_{\theta} \cdots \times_r \mathbf{U}_{\theta}); \dots; \text{vec}(\mathbf{U}_{\theta} \mathbf{A}_2 \mathbf{U}_{\theta}^T); \mathbf{U}_{\theta} \mathbf{a}_1; a_0], \quad (26)$$

where $\forall s = 1, 2, \dots, r$, \mathcal{A}_s is a s order tensor, denoting the s order polynomial coefficients. We can deduce that, for any $s \in \mathbb{N}^+$ the derivative of $Q_s(\theta) = \mathcal{A}_s \times_1 \mathbf{U}_{\theta} \cdots \times_s \mathbf{U}_{\theta}$ is

$$Q'_s(\theta) = \sum_{i=1}^s \mathcal{A}_s \times_1 \mathbf{U}_{\theta} \times_2 \mathbf{U}_{\theta} \cdots \times_{i-1} \mathbf{U}_{\theta} \times_i \bar{\mathbf{U}}_{\theta} \times_{i+1} \mathbf{U}_{\theta} \cdots \times_s \mathbf{U}_{\theta}, \quad (27)$$

where

$$\bar{\mathbf{U}}_{\theta} = \frac{\partial \mathbf{U}_{\theta}}{\partial \theta} = \begin{pmatrix} -\sin(\theta) & \cos(\theta) \\ -\cos(\theta) & -\sin(\theta) \end{pmatrix}. \quad (28)$$

Then, we can deduce that the derivative of $F(\theta)$ is

$$Q'(\theta) = 2 \langle \mathbf{D}^T(\mathbf{D}(f_{\theta}(\mathbf{a}_{hw})) - \mathbf{o}_{wh}), [\text{vec}(Q'_r(\theta)); \dots; \text{vec}(Q'_2(\theta)); \text{vec}(Q'_1(\theta)); 0] \rangle, \quad (29)$$

which is also the gradient of problem (25). Then we can perform gradient descent method to solve problem (25).

In practice, we easily calculate the gradient direction by

$$g(\theta) = \begin{cases} 1, & \text{if } Q(\theta + \epsilon) < Q(\theta), \\ -1, & \text{if } Q(\theta + \epsilon) > Q(\theta), \end{cases} \quad (30)$$

where ϵ is a small positive number. Then we perform gradient descent method to solve problem (25). Though the closed-form solution of this problem is hard to deduce, the problem is a one dimensional optimization with smooth objective function, which can be easily optimized by off-the-shelf toolkits.

Update \mathcal{X} . By using $\sum_l \gamma_{hw} = 1$, it is easy to deduce that \mathcal{X} can be updated by solving following sub-problem:

$$\min_{\mathcal{X}} \frac{\sigma}{\lambda} \|\mathcal{X} - \mathcal{Y}\|_{\mathbb{F}}^2 + \sum_{hw} \|D(f_{\theta_{hw}}((\mathcal{F} * \mathcal{X})_{hw})) - \mathbf{o}_{wh}\|_{\mathbb{F}}^2. \quad (31)$$

We apply the alternating direction method of multipliers (ADMM) [38] to solve this problem. Firstly we introduce a tensor \mathcal{B} and equivalently reformulate (41) as follows:

$$\min_{\mathcal{X}, \mathcal{B}} \frac{\sigma}{\lambda} \|\mathcal{X} - \mathcal{Y}\|_{\mathbb{F}}^2 + \|H_D(\mathcal{B}) - \mathcal{O}\|_{\mathbb{F}}^2 \quad \text{s.t.} \quad \mathcal{B} = F(\mathcal{F} * \mathcal{X}, \boldsymbol{\theta}), \quad (32)$$

where \mathcal{O} is the tensor stacked by \mathbf{o}_{wh} s. $H_D(\mathcal{B})$ and $F(\mathcal{F} * \mathcal{X}, \boldsymbol{\theta})$ denote the result of performing $D \cdot \mathcal{B}_{hw}$ and $f_{\theta_{hw}}((\mathcal{F} * \mathcal{X})_{hw})$ for all h and w .

Then, the augmented Lagrangian function, i.e., $L_{\rho}(\mathcal{X}, \mathcal{B}, \mathcal{L})$, is [38]

$$\frac{\sigma}{\lambda} \|\mathcal{X} - \mathcal{Y}\|_{\mathbb{F}}^2 + \|H_D(\mathcal{B}) - \mathcal{O}\|_{\mathbb{F}}^2 + \langle \mathcal{B} - F(\mathcal{F} * \mathcal{X}, \boldsymbol{\theta}), \mathcal{L} \rangle + \frac{\rho}{2} \|\mathcal{B} - F(\mathcal{F} * \mathcal{X}, \boldsymbol{\theta}) * \mathcal{X}\|_{\mathbb{F}}^2, \quad (33)$$

where \mathcal{L} is the Lagrange multiplier and ρ is a positive scalar. According to the ADMM framework, we update \mathcal{X} , \mathcal{B} , \mathcal{L} alternatively.

With other parameters fixed, \mathcal{X} can be updated by solving $\min_{\mathcal{X}} L_{\rho}(\mathcal{X}, \mathcal{B}, \mathcal{L})$ which is equivalent to the following problem:

$$\min_{\mathcal{X}} \frac{\sigma}{\lambda} \|\mathcal{X} - \mathcal{Y}\|_{\mathbb{F}}^2 + \frac{\rho}{2} \|F(\mathcal{F} * \mathcal{X}, \boldsymbol{\theta}) - \mathcal{B} - \rho^{-1} \mathcal{L}\|_{\mathbb{F}}^2. \quad (34)$$

Since the operator $f_{\theta}(\cdot)$ is constructed by several rotation matrices $\mathbf{U}_{\theta_{hw}}$ s, it is easy to deduce that $\|f_{\theta}(\cdot)\|_{\mathbb{F}}^2 = \|\cdot\|_{\mathbb{F}}^2$. Thus Eq. (34) is equivalent to

$$\min_{\mathcal{X}} \frac{\sigma}{\lambda} \|\mathcal{X} - \mathcal{Y}\|_{\mathbb{F}}^2 + \frac{\rho}{2} \|\mathcal{F} * \mathcal{X} - F(\mathcal{B} + \rho^{-1} \mathcal{L}, -\boldsymbol{\theta})\|_{\mathbb{F}}^2, \quad (35)$$

which has been proved to have a closed-form solution [39]

$$\mathcal{X}^+ = \text{fft}^{-1} \left(\frac{\frac{\sigma}{\lambda} \text{fft}(\mathcal{Y}) + \frac{\rho}{2} \sum_k \text{fft}(F_k)^* \odot \text{fft}((F(\mathcal{B} + \rho^{-1} \mathcal{L}, -\boldsymbol{\theta}))_k)}{\frac{\sigma}{\lambda} + \frac{\rho}{2} \sum_k (\text{fft}(F_k) \odot \text{fft}(F_k)^*)} \right), \quad (36)$$

where \odot denotes element-by-element multiplication.

With other parameters fixed, \mathcal{B} can be updated by solving $\min_{\mathcal{B}} L_{\rho}(\mathcal{X}, \mathcal{B}, \mathcal{L})$ which is equivalent to the following problem:

$$\min_{\mathcal{B}} \|H_D(\mathcal{B}) - \mathcal{O}\|_{\mathbb{F}}^2 + \frac{\rho}{2} \|\mathcal{B} - F(\mathcal{F} * \mathcal{X}, \boldsymbol{\theta}) + \rho^{-1} \mathcal{L}\|_{\mathbb{F}}^2. \quad (37)$$

Since the operator $H_D(\cdot)$ is constructed by the same matrix D product the correspondence vectors in \mathcal{B} , Eq. (37) is a quadratic program. Its closed-form solution is

$$\mathcal{B}^+ = H_{(D^T D + \frac{\rho}{2} I)^{-1}} \left(H_{D^T}(\mathcal{O}) + \frac{\rho}{2} (F(\mathcal{F} * \mathcal{X}, \boldsymbol{\theta}) - \rho^{-1} \mathcal{L}) \right). \quad (38)$$

Finally the Lagrange multiplier \mathcal{L} can be update in closed-form [38]

$$\mathcal{L}^+ = \mathcal{L} + \rho(\mathcal{B} - F(\mathcal{F} * \mathcal{X}, \boldsymbol{\theta})). \quad (39)$$

The proposed algorithm can then be summarized in Algorithm 1, and we denote the proposed method as CDI-MoG (color and direction invariant mixture of Gaussian denoising method). An example of the estimated $\boldsymbol{\mu}$, \mathbf{c} , \mathbf{b} , θ , γ is illustrated in Figure 1(f)–(j) for easy understanding. Please refer to Appendix B for more details of the algorithm.

Algorithm 1 Algorithm for CDI-MoG method

Input: Noisy image \mathcal{Y} .

- 1: Initialize $\mathcal{X}^{(0)}$, $\boldsymbol{\theta}^{(0)}$, $\mathbf{c}^{(0)}$, $\mathbf{b}^{(0)}$, $\boldsymbol{\mu}^{(0)}$, $\boldsymbol{\pi}^{(0)}$, and $\sigma^{(0)}$;
- 2: **for** $l = 1 : L$ **do**
- 3: Update E step by Eq. (21);
- 4: Update $\boldsymbol{\mu}$, \mathbf{c} , \mathbf{b} , σ , $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$ by Eqs. (23)–(25);
- 5: **while** not convergence **do**
- 6: Update \mathcal{X} by Eq. (36) and update \mathcal{B} by Eq. (38);
- 7: Update \mathcal{L} by Eq. (39) and let $\mu := \rho\mu$;
- 8: **end while**
- 9: **end for**

Output: Denoised image $\mathcal{X}^{(L)}$.

Computational complexity. For an input image $Y \in \mathbb{R}^{H \times W \times 3}$, the cost for updating $\boldsymbol{\mu}$, \mathbf{c} , \mathbf{b} are around $O(HWLM)$, where M is the number of pixels in an image patch, and L is the number of Gaussian components in our model. When updating θ , the cost depends on the order of the fitting polynomial function r . We can deduce that the number of coefficients is $K = \frac{(r+2)(r+1)}{2}$. Each time when we calculate $f_\theta(\mathbf{a})$ by (26), the cost is $O(\frac{M(r+2)(r+1)}{2} + 2^{r+2} - 1)$. Thus the cost on updating $\boldsymbol{\theta}$ is $O(\frac{HWM(r+2)(r+1)}{2} + (2^{r+2} - 1)HW)$. Note that when we set $r = 5$ and $L = 200$ (as we did in our experiments), the cost of updating $\boldsymbol{\mu}$, \mathbf{c} , \mathbf{b} are similar to that of updating $\boldsymbol{\theta}$. The cost of updating \mathcal{X} mostly lies on the calculation of fast Fourier transform, which is $O(HWK \log_2(H))$. While the cost of updating \mathcal{B} is $O((K+M)MKHW)$. From the above analysis, we can find that the per-iteration computational cost of our method is around $O((LM + \frac{M(r+2)(r+1)}{2} + (2^{r+2} - 1) + K \log_2(H) + (K+M)MK)HW)$. It should be noted that the per-iteration cost of the proposed method, is comparable to that of the state-of-the-art method, MCWNNM [9], which is $O(\max(M^2P, P^3)HW)$, where P denote the number of patches in each similar patch group.

Extension to general color image processing tasks. Note that the proposed CDI-NSS is easy to be applied to other color image processing tasks. Consider a general image processing problem, which can be modeled as $\mathcal{Y} = A(\mathcal{X})$ with \mathcal{Y} , \mathcal{X} and A denote the observation, original image and the linear degradation operator, respectively. Similar to Eq. (40), we can adopt the following generation distribution to the observation:

$$p(\mathcal{Y}|\mathcal{X}) = \prod_{hw} \prod_{q=1}^3 \mathcal{N}(y_{hwq} | (A(\mathcal{X}))_{hwq}, \lambda), \tag{40}$$

where h and w denote height and width of the image, respectively, $q \in \{1, 2, 3\}$ denotes the three channels of color image and λ is the variance parameter. Combining this distribution with the CDI-NSS prior distribution (18), we can obtain a posteriori distribution similar to Eq. (20). Then we can estimate \mathcal{X} by MAP estimation in a similar way as the CDI-MoG algorithm. The difference is that the subproblem (41) now is replaced by

$$\min_{\mathcal{X}} \frac{\sigma}{\lambda} \|A(\mathcal{X}) - \mathcal{Y}\|_{\mathbb{F}}^2 + \sum_{hw} \|D(f_{\theta_{hw}}((\mathcal{F} * \mathcal{X})_{hw})) - \mathbf{o}_{wh}\|_{\mathbb{F}}^2. \tag{41}$$

Here, we just show a very intuitive way as an example. One can exploit CDI-NSS in flexible ways depending on the specific situations.

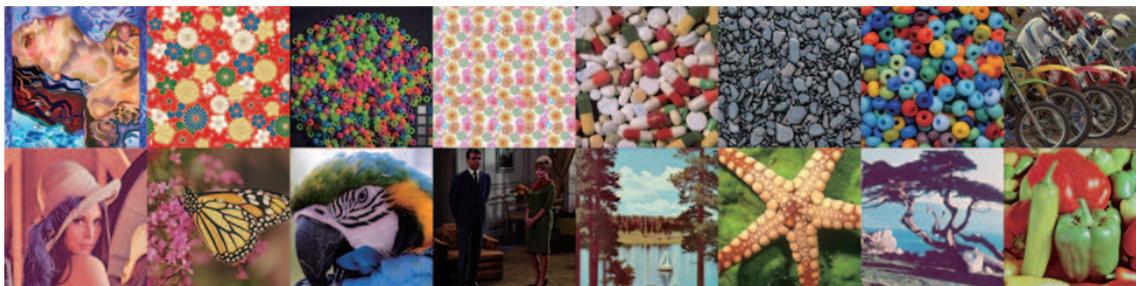


Figure 4 16 employed color images in synthetic experiments.

Table 1 Average performance of 8 competing methods with respect to 4 PQIs. For both settings, the results are obtained by averaging through the 16 images

Competing method	$\lambda = 0.10^2$				$\lambda = 0.15^2$				Average time (s)
	PSNR	SSIM	FSIM	MS-SSIM	PSNR	SSIM	FSIM	MS-SSIM	
Noisy	19.998	0.484	0.812	0.295	16.476	0.350	0.728	0.272	–
BM3D	27.913	0.845	0.925	0.320	25.781	0.782	0.893	0.312	0.47
WNNM	28.249	0.852	0.928	0.321	26.185	0.791	0.896	0.313	484.94
NCSR	27.963	0.843	0.923	0.320	25.769	0.777	0.883	0.311	1427.69
PCLR	28.341	0.855	0.930	0.321	26.281	0.796	0.897	0.314	279.19
EPLL	27.973	0.850	0.932	0.321	25.972	0.789	0.902	0.313	105.28
CBM3D	29.094	0.881	0.935	0.324	26.793	0.823	0.904	0.317	0.45
MCWNNM	29.119	0.874	0.930	0.323	26.926	0.819	0.896	0.316	115.60
CDI-MoG	29.257	0.882	0.937	0.324	27.130	0.831	0.910	0.317	984.05

6 Experimental results

We evaluate the proposed CDI-MoG method on synthetic and real noisy color images. We compare the proposed method with state-of-the-art denoising methods, including BM3D [2], WNNM [3], NCSR [7], PCLR [21], EPLL [16], CBM3D [8], and MCWNNM [9]. The later two methods represent state-of-the-arts color image denoising method, and the first 5 methods represent state-of-the-arts gray-scale denoising methods which is adopted to the RGB channels separably.

Implementation details. In proposed method, we set the number of Gaussian component $L = 200$, and set diameter of image patch to be 5. Since the updating of \mathcal{X} takes more computation than other parameters, we update \mathcal{X} only once after 15 iterations updating of other parameters.

6.1 Experiments on synthetic noisy color images

We first compare the proposed CDI-MoG method with 7 competing methods on 16 color images, as shown in Figure 4. Four quantitative picture quality indices (PQI) are employed for performance evaluation, including peak signal-to-noise ratio (PSNR), structure similarity (SSIM [40]), feature similarity (FSIM [41]), and multiscale structural similarity (MS-SSIM [42]). The larger these four measures are, the closer the target image is to the reference one.

For most of the competing denoising methods, the standard deviation of noise should be given as a parameter. In synthetic experiments, additive Gaussian noises with variance $\lambda = 0.1^2$ and 0.15^2 are added to these testing images to generate the noisy observations, and the noise levels λ is assumed to be known for all competing methods.

For each noise setting, all of the four PQI values for each competing MSI denoising methods on all 16 scenes have been calculated and recorded. Table 1 lists the average performance of the competing methods. From these quantitative comparison, the advantage of the proposed method can be evidently observed. Specifically, the average results of our method can outperform other competing methods with respect to all evaluation measures. Note that albeit with comparable per-iteration computational cost with current state-of-the-arts, the computation of the proposed method might be not that fast since on one hand we have not optimized our code in the Matlab platform, on the other hand it might need more

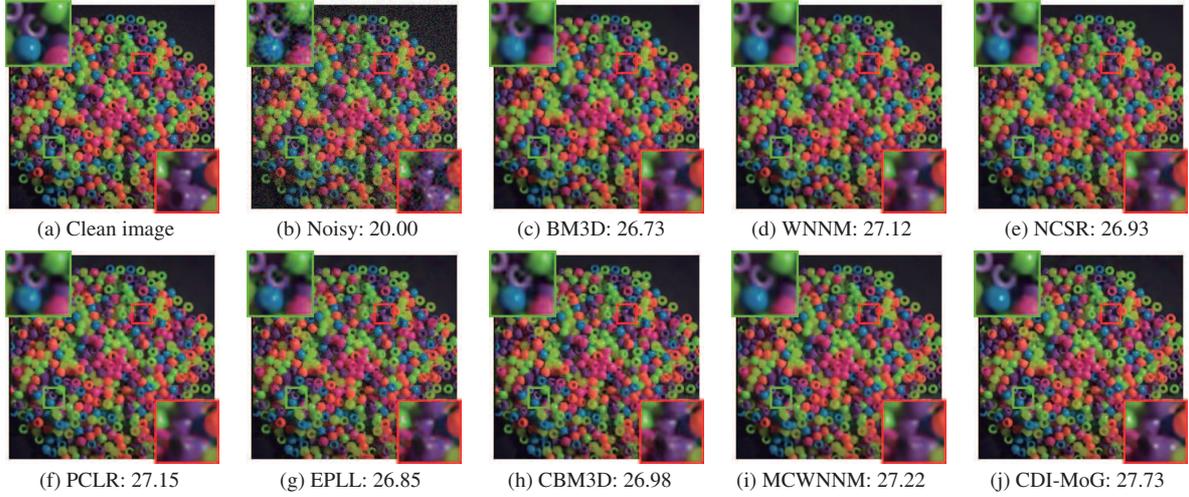


Figure 5 (a) Clean image. (b) Corresponding noisy image. (c)–(j) Restored images obtained by 8 competing methods. The demarcated areas are for easy observing the detail.

iteration steps for final convergence. We will further alleviate this efficiency issue in our future research.

To further depict the denoising performance of our method, we show in Figure 5 the results of the competing methods on 4 typical utilized images, with noise level $\lambda = 0.15^2$. From the figures, it is easy to observe that the proposed method performs better than other competing ones, both in the recovery of finer-grained textures and coarser-grained structures.

The good performance of the proposed CDI-MoG model can be rationally explained by observing Figure 1(f)–(j). It can be seen from Figure 1(f) that the extracted core structural patches μ_l finely reflect the knowledge of local edges and textures underlying the latent recovery image. This should be attributed to the utilization of more sufficient self-similar image patches under the proposed color and direction invariant representations as compared with the conventional NSS manners. Besides, from Figure 1(g)–(i), it is seen that the extracted \mathbf{c} , \mathbf{b} , and θ all comply with their expected physical meanings. Especially, by observing Figure 1(j), through elaborately adjusting the color and texture-direction of all image patches by our method, they can be finely categorized into rational clusters based on their intrinsic structures. In particular, the edges of all items in the image, albeit with different colors and different directions, can be well compensated between each other to facilitate a good boundary recovery and noise suppression effects. For more experimental results, please refer to Appendix C.

6.2 Experiments on real noisy color images

In this subsection, we evaluate the proposed method on real noisy color images from dataset [29] and dataset [30]. Since the noise level is unknown for real noisy images, we use the noise estimation method proposed in [43] to estimate the noise of each channel. Here, we only compare the visual quality of the denoised images since there is no “ground truth” for the real noisy images. For CBM3D and the proposed CDI-MoG, a single noise level should be the input, and we simply set the noise level as $\lambda = \frac{\lambda_r + \lambda_g + \lambda_b}{3}$.

Figures 6 and 7 show the denoised images by the competing methods on the two datasets, respectively. It can be seen that the images restored by CDI-MoG are capable of better removing the noise while finely preserving the structure. For more experimental results, please refer to Appendix D.

7 Conclusion

A new NSS prior, called CDI-NSS, has been designed in this study, by considering the similarity among patches with different colors and different texture-directions. Such prior can help extract more sufficient local image patches with self-similarity and thus expected to lead better performance of related image

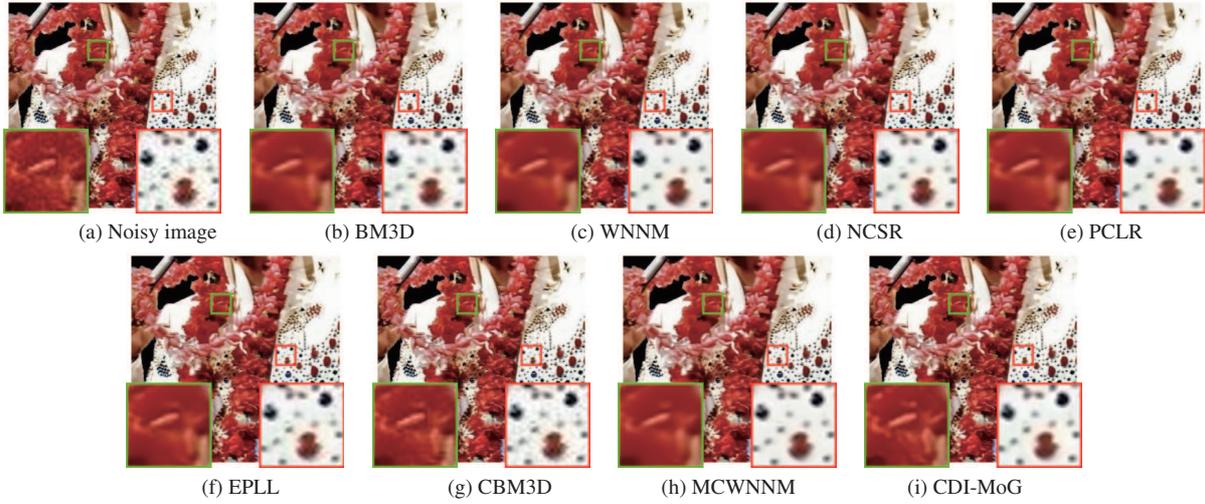


Figure 6 (a) A real noisy color image from dataset [29]. (b)–(i) The restored images obtained by 8 competing methods.

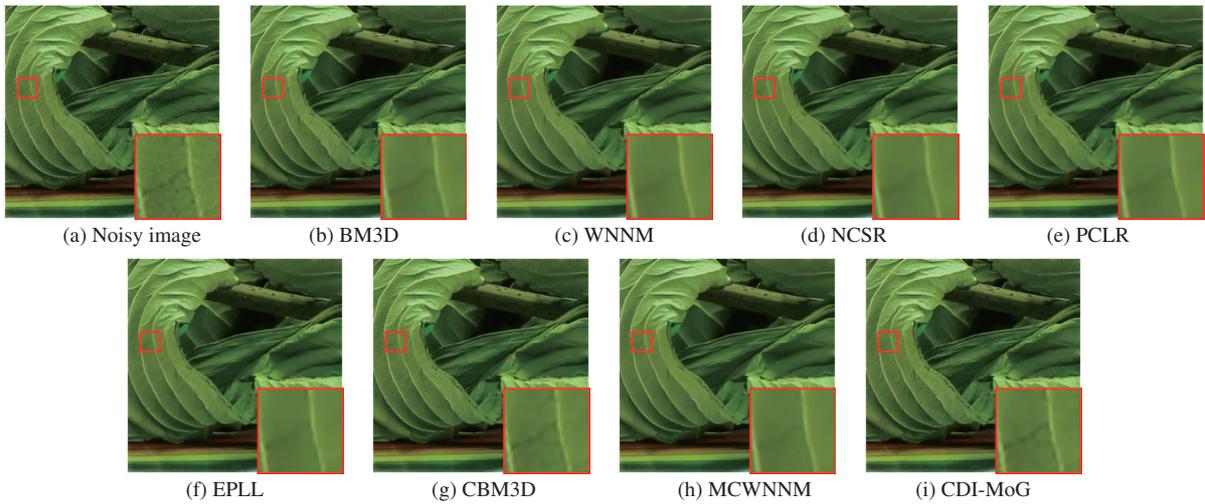


Figure 7 (a) A real noisy color image from dataset [30]. (b)–(i) The restored images obtained by 8 competing methods.

processing tasks. We have specifically testify the capability of this prior in color image denoising problem, and designed MAP model as well as its EM solving algorithm for handling this task. Experiments implemented on simulated and real color images have substantiated the superiority of this method, verifying the rationality of this newly constructed prior forms.

The image NSS study remains an open issue. Indeed there is still large room for further improvements of CDI-NSS. Firstly, the computational speed of the proposed method still has large room to be further improved. This is due to the fact that we need to apply an ADMM algorithm to solve the subproblem in each iteration. Actually, we might employ some inexact approximate method to solve them for speeding-up the algorithm. Secondly, when the image is badly damaged, the method inclines to suffer from the inaccurate estimation of the parameter θ and poor clustering results of MoG. Thus, a robust amelioration of the proposed method is required to be proposed specifically for complicated non-Gaussian noise cases. Thirdly, this work proposes a concise model with sound mathematical forms and evidently different from previous strategies specifically designed for this task. This, however, limits the flexible design of the proposed algorithm, especially make the algorithm not inherit advantageous aspects of other ones. It is thus meaningful to consider how to embed more beneficial points of current algorithms into the previous ones, under the guarantee of its concise mathematical expression.

Supporting information Appendixes A–D. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

- 1 Buades A, Coll B, Morel J M. A non-local algorithm for image denoising. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005
- 2 Dabov K, Foi A, Katkovnik V, et al. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans Image Process*, 2007, 16: 2080–2095
- 3 Gu S H, Xie Q, Meng D Y, et al. Weighted nuclear norm minimization and its applications to low level vision. *Int J Comput Vis*, 2017, 121: 183–208
- 4 Ji H, Liu C C, Shen Z W, et al. Robust video denoising using low rank matrix completion. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010
- 5 Yu G S, Sapiro G, Mallat S. Solving inverse problems with piecewise linear estimators: from Gaussian mixture models to structured sparsity. *IEEE Trans Image Process*, 2012, 21: 2481–2499
- 6 Mairal J, Bach F, Ponce J, et al. Non-local sparse models for image restoration. In: Proceedings of the 12th International Conference on Computer Vision, 2009. 2272–2279
- 7 Dong W S, Zhang L, Shi G M, et al. Nonlocally centralized sparse representation for image restoration. *IEEE Trans Image Process*, 2013, 22: 1620–1630
- 8 Dabov K, Foi A, Katkovnik V, et al. Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. In: Proceedings of IEEE International Conference on Image Processing, 2007
- 9 Xu J, Zhang L, Zhang D, et al. Multi-channel weighted nuclear norm minimization for real color image denoising. In: Proceedings of IEEE International Conference on Computer Vision, 2017. 1096–1104
- 10 McLachlan G J, Basford K E. *Mixture Models: Inference and Applications to Clustering*. New York: Marcel Dekker, 1988
- 11 Rudin L I, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. *Phys D-Nonlinear Phenom*, 1992, 60: 259–268
- 12 Zuo W M, Zhang L, Song C W, et al. Texture enhanced image denoising via gradient histogram preservation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2013. 1203–1210
- 13 Do M N, Vetterli M. Wavelet-based texture retrieval using generalized Gaussian density and Kullback-Leibler distance. *IEEE Trans Image Process*, 2002, 11: 146–158
- 14 Portilla J, Strela V, Wainwright M J, et al. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans Image Process*, 2003, 12: 1338–1351
- 15 Peyré G, Bougleux S, Cohen L. Non-local regularization of inverse problems. In: Proceedings of European Conference on Computer Vision, 2008. 57–68
- 16 Zoran D, Weiss Y. From learning models of natural image patches to whole image restoration. In: Proceedings of International Conference on Computer Vision, 2011. 479–486
- 17 Dong W S, Shi G M, Li X. Nonlocal image restoration with bilateral variance estimation: a low-rank approach. *IEEE Trans Image Process*, 2013, 22: 700–711
- 18 Rajwade A, Rangarajan A, Banerjee A. Image denoising using the higher order singular value decomposition. *IEEE Trans Pattern Anal Mach Intell*, 2013, 35: 849–862
- 19 Zhang K, Zuo W M, Chen Y J, et al. Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising. *IEEE Trans Image Process*, 2017, 26: 3142–3155
- 20 Chen Y J, Yu W, Pock T. On learning optimized reaction diffusion processes for effective image restoration. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2015. 5261–5269
- 21 Chen F, Zhang L, Yu H M. External patch prior guided internal clustering for image denoising. In: Proceedings of IEEE International Conference on Computer Vision, 2015. 603–611
- 22 Elad M, Aharon M. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans Image Process*, 2006, 15: 3736–3745
- 23 Pang Y W, Xie J, Li X L. Visual haze removal by a unified generative adversarial network. *IEEE Trans Circ Syst Video Technol*, 2019, 29: 3211–3221
- 24 Pang Y W, Li Y Z, Shen J B, et al. Towards bridging semantic gap to improve semantic segmentation. In: Proceedings of IEEE International Conference on Computer Vision, 2019. 4230–4239
- 25 Zhou Z H. Abductive learning: towards bridging machine learning and logical reasoning. *Sci China Inf Sci*, 2019, 62: 076101
- 26 Kervrann C, Boulanger J, Coupé P. Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal. In: Proceedings of International Conference on Scale Space and Variational Methods in Computer Vision, 2007. 520–532
- 27 Zhu F Y, Chen G Y, Heng P A. From noise modeling to blind image denoising. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016. 420–429
- 28 Lebrun M, Colom M, Morel J M. Multiscale image blind denoising. *IEEE Trans Image Process*, 2015, 24: 3149–3161
- 29 Lebrun M, Colom M, Morel J M. The noise clinic: a blind image denoising algorithm. *Image Process Line*, 2015, 5:

1–54

- 30 Nam S, Hwang Y, Matsushita Y, et al. A holistic approach to cross-channel image noise modeling and its application to image denoising. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016. 1683–1691
- 31 Elmoataz A, Lezoray O, Bogleux S. Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing. *IEEE Trans Image Process*, 2008, 17: 1047–1060
- 32 Gilboa G, Osher S. Nonlocal operators with applications to image processing. *Multiscale Model Simul*, 2009, 7: 1005–1028
- 33 International Telecommunication Union. Parameter values for the HDTV standards for production and international programme exchange. BT.709-5. <https://www.itu.int/rec/R-REC-BT.709/en>
- 34 Roth S, Black M J. Fields of experts. *Int J Comput Vis*, 2009, 82: 205–229
- 35 Cao X Y, Chen Y, Zhao Q, et al. Low-rank matrix factorization under general mixture noise distributions. In: Proceedings of IEEE International Conference on Computer Vision, 2015. 1493–1501
- 36 Yong H W, Meng D Y, Zuo W M, et al. Robust online matrix factorization for dynamic background subtraction. In: Proceedings of IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018
- 37 Dempster A P, Laird N M, Rubin D B. Maximum likelihood from incomplete data via the em algorithm. *J R Stat Soc-Ser B*, 1977, 39: 1–22
- 38 Boyd S, Parikh N, Chu E, et al. *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Boston: Now Foundations and Trends, 2011
- 39 Krishnan D, Fergus R. Fast image deconvolution using hyper-Laplacian priors. In: Proceedings of International Conference on Neural Information Processing Systems, 2009. 1033–1041
- 40 Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error visibility to structural similarity. *IEEE Trans Image Process*, 2004, 13: 600–612
- 41 Zhang L, Zhang L, Mou X Q, et al. FSIM: a feature similarity index for image quality assessment. *IEEE Trans Image Process*, 2011, 20: 2378–2386
- 42 Wang Z, Simoncelli E P, Bovik A C. Multiscale structural similarity for image quality assessment. In: Proceedings of the 37th Asilomar Conference on Signals, Systems & Computers, 2003. 1398–1402
- 43 Chen G Y, Zhu F Y, Heng P A. An efficient statistical method for image noise level estimation. In: Proceedings of IEEE International Conference on Computer Vision, 2015. 477–485