Journal of Mathematical Imaging and Vision
https://doi.org/10.1007/s10851-024-01204-y

An Edge-Preserving Regularization Model for the Demosaicing of Noisy Color Images

Antonio Boccuto[1] · Ivan Gerace[1] · Valentina Giorgetti[1] · Francesca Martinelli[1] · Anna Tonazzini[2]

Received: 5 January 2023 / Accepted: 17 June 2024

Abstract

This paper proposes an edge-preserving regularization technique to solve the color image demosaicing problem in the realistic case of noisy data. We enforce intra-channel local smoothness of the intensity (low-frequency components) and inter-channel local similarity of the depth of object borders and textures (high-frequency components). Discontinuities of both the low-frequency and high-frequency components are accounted for implicitly, i.e., through suitable functions of the proper derivatives. For the treatment of even the finest image details, derivatives of first, second, and third orders are considered. The solution to the demosaicing problem is defined as the minimizer of an energy function, accounting for all these constraints plus a data fidelity term. This non-convex energy is minimized via an iterative deterministic algorithm, applied to a family of approximating functions, each implicitly referring to geometrically consistent image edges. Our method is general because it does not refer to any specific color filter array. However, to allow quantitative comparisons with other published results, we tested it in the case of the Bayer CFA and on the Kodak 24-image dataset, the McMaster (IMAX) 18-image dataset, the Microsoft Demosaicing Canon 57-image dataset, and the Microsoft Demosaicing Panasonic 500-image dataset. The comparisons with some of the most recent demosaicing algorithms show the good performance of our method in both the noiseless and noisy cases.

Keywords Color image interpolation · Demosaicing · Color filter array · Edge-preserving regularization · Non-convex minimization · Color image denoising

1 Introduction

The demosaicing problem arises from acquiring RGB color images through CCD (charged coupled devices) digital cameras. In the RGB model, each pixel of a digital color image is associated with a triplet of numbers, representing the intensity of the red, green, and blue, respectively. However, most commercial cameras employ a single sensor associated with a color filter that only permits, at each pixel, the measurement of the reflectance of the scene at one of the three colors, according to a predefined scheme or pattern, called *color filter array* (CFA), as illustrated in Fig. 1. This situation implies that, for each pixel, the other two missing colors must be estimated. Although several diverse CFAs have been proposed for the acquisition (e.g., see [31] and [35] for the square pixel layout and [3] for the Penrose aperiodic pixel layout), the most largely diffuse is the Bayer pattern [4], shown in Fig. 2. Most of the literature on demosaicing is thus devoted to algorithms designed with explicit reference to the Bayer pattern. In [30], [46], and [56], comprehensive surveys of the state of the art can be found. In the Bayer scheme, green is sampled in a number of pixels double the number where red and blue are measured to exploit the human eye's higher sensibility to the green wavelength.

✉ Ivan Gerace
ivan.gerace@unipg.it

Antonio Boccuto
antonio.boccuto@unipg.it

Valentina Giorgetti
valentina90giorgetti@libero.it

Francesca Martinelli
framar80@gmail.com

Anna Tonazzini
anna.tonazzini@isti.cnr.it

[1] Dipartmento di Matematica e Informatica, Università degli Studi di Perugia, 1, Via L. Vanvitelli, 06123 Perugia, Italy

[2] Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche, 1, Via G. Moruzzi, 56124 Pisa, Italy

Published online: 26 July 2024

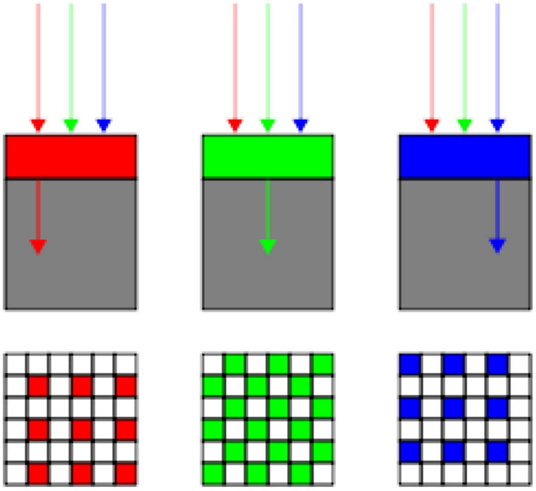

Fig. 1 Color image acquisition based on a CFA

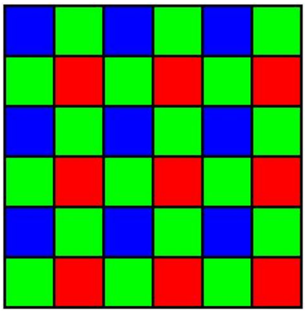

Fig. 2 The Bayer pattern

By decomposing the recorded digital image into the three color channels, one obtains three downsampled grayscale images so that demosaicing could be interpreted as the problem of interpolating grayscale images from sparse data.

In most recreational cameras, demosaicing is part of the processing pipeline required to convert these images into a viewable format. The camera's built-in firmware is substantially based on very fast, or even real-time, interpolation algorithms.

Nevertheless, interpolation is not sufficient to produce the high-quality images required, e.g., for professional purposes. Indeed, interpolation is often applied to each channel separately, disregarding the mutual solid correlation between the channels. Since interpolation algorithms are basically low-pass filters, they perform well in the low-frequency components but reduce the high-frequency ones, thus producing color artifacts in the output image. A remedy to this problem should consider the high correlation among the three color channels, which occurs mainly in correspondence with their high-frequency components. Indeed, the image edges, such as object borders and textures, are largely shared by the color channels. Unavoidably, exploiting the inter-channel information requires more complex algorithms, which are usually time-consuming and cannot run onboard.

Nowadays, many professional digital cameras can save images in a raw format, thus allowing users to demosaic them using offline software. When the raw image data is accessible, one can use various demosaicing algorithms instead of being limited to the one built into the camera. For example, the raw development program RawTherapee [64] gives the user an option to choose, among around ten different algorithms, which should be used. For the Bayer CFA, AMaZE (Aliasing Minimization and Zipper Elimination), developed by Emil J. Martinec in 2010, yields the best results in most cases. Nevertheless, since different demosaicing algorithms present differences in rendering the finest detail and grain texture, photographers often prefer a particular algorithm for aesthetic reasons related to this effect. On the other hand, when it comes to professional applications, computational time is less critical. Thus, there is an interest in developing dedicated and even very sophisticated algorithms that can provide demosaiced images where the high-frequency components are accurately reconstructed.

To this end, this paper proposes an algorithm for image demosaicing that falls within the framework of the edge-preserving regularization approaches and is suited, naturally, to deal with noisy data. More precisely, we propose an algorithm for joint demosaicing and denoising. Regularization requires the adoption of constraints for the solution. The constraints we consider here are intra-channel (spatial) and inter-channel (spectral) local correlation. Concerning the intra-channel correlation, we assume the intensity of each channel to be locally regular, i.e., piecewise smooth, so that noise can also be removed. We describe this constraint through stabilizers that discourage intensity discontinuities of first, second, and third order in a selective way, i.e., permitting to discontinuities associated with actual edges to emerge. This allows us to describe scenes that are even very complex. Indeed, first-order local smoothness characterizes images consisting of constant patches, second-order local smoothness describes patches whose pixels have values varying linearly, while third-order local smoothness is used to represent images made up of quadratic-valued patches. As per the inter-channel correlation, we enforce it in correspondence with the intensity discontinuities using constraints that promote their amplitude in the three channels to be equal almost everywhere.

Note that all these constraints are not biased in favor of one of the three channels, nor is the geometry of the sampling pattern in any way exploited. Thus, the method we propose is entirely independent of the CFA considered, although, in the experimental result section, we present its application to images mosaiced through the Bayer CFA.

All the above constraints, including the data fidelity term, are merged in a non-convex energy function, whose minimizer is our desired solution.

The non-convexity characteristics of the energy function preclude its minimization through conventional gradient descent optimization techniques. Stochastic and deterministic methodologies offer alternative avenues for minimizing non-convex energy functions. While stochastic techniques often yield exact outcomes, their computational demands are considerable (cf. [25]). Conversely, deterministic algorithms, while not guaranteeing convergence to the global optimum, facilitate the attainment of satisfactory reconstructions within reduced computational durations (cf. [6, 8]).

Here, we adopt the graduated non-convexity (GNC) algorithm [9], which is a prevalent one within the deterministic approaches. This technique approximates the energy function through progressively refined approximation functions converging toward the original function. Each approximation function is optimized using conventional optimization algorithms, leveraging the minima attained in the preceding approximation as initial values.

As mentioned, our edge-preserving regularization approach, combined with the GNC minimization strategy, can produce image solutions that exhibit reliable and geometrically consistent discontinuities of both the intensity and the gradients despite the necessary smoothness constraints.

In the first works proposing edge-preserving regularization, the image discontinuities were often represented by extra, explicit variables, the so-called line processes [25]. That way, it was relatively easy to formulate their required properties regarding constraints. Nevertheless, the use of explicit line variables entails significant computational costs. Thus, so-called duality theorems were derived (see, e.g., [26], [15], [14]) to demonstrate the edge-preserving properties of suitable stabilizers without introducing extra variables. In particular, we developed duality theorems to determine the properties required for stabilizers to manage lines with the desired regularity features implicitly. In this paper, we choose a suitable family of approximations for the GNC having the peculiarity that each function satisfies the conditions required for an implicit treatment of geometrically significant edges, as expressed in the duality theorems proposed in [14]. This allows for better adherence of the approximations to the ideal energy function, resulting in better coherence with the properties required for the desired solution.

GNC, initially introduced in [9], has seen significant advancements over the years. In [7], the GNC framework was extended to accommodate the restoration of noisy images, by incorporating considerations on the geometry of discontinuities. In [60], blur in noisy images has been considered as well. Since then, [15] proposed GNC for deblurring and denoising tasks, enforcing constraints such as line continuation or line non-parallelism, and a GNC variant was designed to deal with point spread functions with a vast domain [12].

The GNC methodology has found applications across diverse domains, which stand as a testament to its versatility and efficacy. Indeed, it has been successfully utilized in addressing the combinatorial data analysis challenge of seriation, as evidenced in [23]. Moreover, its utility extends to tackling stochastic problems, as discussed in [32], and resolving combinatorial optimization dilemmas defined on the domain of partial permutation matrices, as explored in [48]. Additionally, the GNC framework has been harnessed for addressing the maximum a posteriori inference problem, elucidated in [49], as well as for tasks such as pose estimation, documented in [67], and spatial perception, outlined in [71].

This paper extends the GNC methodology by explicitly focusing on the non-parallelism constraint for the discontinuities and proposing a first, initial approximation that is componentwise convex. We call this new version of GNC *graduated componentwise non-convexity* (GCNC).

The paper is organized as follows. In Sect. 2, state-of-the-art data in the color image demosaicing and denoising field are surveyed. Section 3 is devoted to formulating the problem and adopting the specific edge-preserving regularization strategy. In Sect. 4, the solution algorithm is described in detail. Section 5 is devoted to the quantitative comparison between the results obtained with our method and those of some of the most performing algorithms proposed in the recent literature, using the Kodak image dataset [43], the McMaster image dataset [75], the Microsoft Demosaicing Canon Dataset [38], and the Microsoft Demosaicing Panasonic Dataset [38] as benchmark sets, and with specific reference to the Bayer CFA. Conclusions and suggestions for prospects are given in Sect. 6, and finally, in the Appendices, some mathematical aspects are developed in detail.

2 Related Work

A significant problem of demosaicing is avoiding oversmoothing of the edges, so the fundamental feature of any method is its ability to perform interpolation along and not across the edges. Some methods perform directional interpolation by analyzing the variance of the color differences to exploit the high correlation between the color planes [18]. For example, the work in [2] proposes high-order interpolation and Sobel operators to compute the gradients, and in [24], a level set method is used to minimize an energy function that gives the direction of the edges. In [19], the interpolation direction is chosen by exploiting an edge-sensing parameter called integrated gradient, which simultaneously considers color intensity and color difference. In general, due to solid correlations existing among three color channels in nature, performing interpolation in the relatively smooth color difference field can be easier and more convenient.

In other methods, the best reconstruction of the missing data, first estimated by interpolating along horizontal and vertical directions, is chosen [33], [52], or the two reconstructions are fused [74], [75]. In particular, [33] proposes an algorithm based on the Laplacian filter by selecting the interpolation directions having the most minor level of color artifacts. Instead, the method in [17] infers the missing colors by considering the local image geometry through the image self-similarity.

As an alternative to color difference interpolation, the algorithms in [41] and [72], among others, are based on interpolation in a residual domain, where the residuals are differences between observed and tentatively estimated pixel values. This approach is justified because the residual field is smoother than the color difference data field. Hence, the methods based on residual interpolation may be advantageous regarding both peak signal-to-noise (PSNR) and subjective visual quality. In [59], the adaptive residual interpolation method is proposed to combine the residual interpolation [39] and the minimized-Laplacian residual interpolation [40]. The developed method is extensively tested on large datasets in [36].

In [29] and [45], the strong correlation between the high frequencies of the three color components is directly exploited. In particular, the algorithm presented in [29] forces similarity between the high frequencies of the red and the green and of the blue and the green.

The sparse nature of color images has also been exploited for demosaicing. A suitable dictionary is designed and applied with the iterative K-SVD algorithm in [50], whereas in [58], the dictionary is constructed based on a clear distinction between the inter-channel and intra-channel correlations of natural images and the sparse representation of the image is found through compressed sensing. In [1], a locally adaptive approach is used for demosaicing dual-tree complex wavelet coefficients.

Within regularization approaches, the total-variation principle is used in [66], while [54] proposes first a general quadratic smoothness regularizer and then an adaptive filter in order to improve the reconstruction near the edges of the first estimate.

The methods surveyed above have been mainly designed for noise-free data. An abundance of literature has also been devoted to the more realistic noisy data scenario. In this context, performing denoising as a pre-or post-processing has significant drawbacks. In the first case, denoising must be separately performed on the individual channel so that the entire image resolution cannot be exploited. In addition, denoising alters the color samples in the raw image. On the other hand, if demosaicing is performed first, the interpolation process changes the noise distribution. For instance, in [55] [54], the authors evaluate the statistical characteristics of the noise resulting from the demosaicing process performed through space-varying filters, and then design an ad hoc post-processing denoising strategy.

The method in [76] works in two steps. First, the full-resolution green component is recovered from the difference signals of the color channels by exploiting both spectral and spatial correlations to suppress sensor noise and interpolation error simultaneously. Second, the CFA channel-dependent noise is removed from the reconstructed green channel with a wavelet-based approach, and the red and blue channels are also estimated and denoised. The work in [34] presents an algorithm that uses a modified total least squared estimation technique to estimate an ideal demosaicing filter to deal with the noise affecting the data.

In [20], a noisy mosaiced image's luminance and chrominance channels are first reconstructed by exploiting a frequency analysis of the sampling pattern induced by the Bayer CFA. Wiener filters are then designed to denoise the chrominances, whereas the luminance is linearly filtered as a grayscale image. An extended variant of this approach is also proposed, in which the demosaiced image is mosaiced again and then demosaiced using the method in [74].

Regularization is an ideal setting for joint demosaicing and denoising, which can be performed simultaneously based on the same image priors. These can be manually designed, as in [21], where a total variation (TV) prior ensures the smooth property of the image or can be learned from the image dataset, as in [38], where regression tree fields are used to learn realistic image datasets. The method in [42] models the problem as a minimization problem and uses an image dataset to improve performance.

In recent years, many demosaicing algorithms have been developed by exploiting the potential of convolutional neural networks (CNNs) to avoid dependence on empirically defined priors. For instance, in [27], a deep demosaicing and denoising network was trained on millions of artificially degraded images. A two-stage CNN and a three-stage CNN have been proposed in [69] and [22], respectively, and in [68], an image demosaiced using traditional interpolation schemes is refined using the residual network. The work in [44] a specific network architecture is designed, inspired by powerful classic image regularization and large-scale optimization. To improve the quality of the recovered image, in [47], the guidance of the density map and the exploitation of edge characteristics are proposed, and in [70], the structure of the CNN model and the loss functions are carefully studied. Finally, in [63], a CNN model is used as a prior within a regularization setup.

Although deep learning-based methods have demonstrated higher performance, they are highly data-dependent and require many training images, i.e., full-resolution RGB images. In many approaches, the authors have used as reference images already processed; that is, they apply a mosaic mask on images already demosaiced, thus obtaining unrealistic training pairs [68]. In fact, with this learning strategy, the main problem is that demosaicing the training data artifacts will hamper the reconstruction's performance and overall quality. Furthermore, methods based on learning require high memory and computation costs, which is undesirable for integrated sensor systems. For these reasons, interpolation-based methods are still widely applied in practical use due to their simplicity.

3 Formulation of the Demosaicing Problem and Its Regularization

3.1 The Data Generation Model

A *color image* of size $n \times m$ can be represented as a vector $\mathbf{x} \in \mathbb{R}^{3nm}$, $\mathbf{x} = \left((\mathbf{x}^{(r)})^T \, (\mathbf{x}^{(g)})^T \, (\mathbf{x}^{(b)})^T\right)^T$, where $\mathbf{x}^{(r)}, \mathbf{x}^{(g)}, \mathbf{x}^{(b)} \in \mathbb{R}^{nm}$ are the red, green and blue channels expressed in the lexicographic notation, respectively. The mosaicing problem is formulated as

$$\mathbf{y} = M\,(\mathbf{x} + \mathbf{n}), \tag{1}$$

where $\mathbf{x}, \mathbf{y}, \mathbf{n} \in [0, 255]^{3nm}$ denote the ideal color image, the mosaiced image, and the additive noise, respectively. We assume the noise to be independent, Gaussian, with null mean and variance σ^2. The matrix $M \in \{0, 1\}^{3nm \times 3nm}$ is a linear operator associated with the acquisition pattern, consisting of the following block diagonal matrix:

$$M = \begin{pmatrix} M^{(r)} & O & O \\ O & M^{(g)} & O \\ O & O & M^{(b)} \end{pmatrix}, \tag{2}$$

where $O \in \mathbb{R}^{nm \times nm}$ is the null matrix, and $M^{(r)}$, $M^{(g)}$, $M^{(b)}$ are diagonal matrices in $\mathbb{R}^{nm \times nm}$. For the Bayer pattern, the diagonal elements of these matrices are given by

$$\begin{aligned} m^{(r)}_{(i,j),(i,j)} &= \begin{cases} 1, & i \equiv_2 j \equiv_2 0, \\ 0, & \text{otherwise}, \end{cases} \\ m^{(g)}_{(i,j),(i,j)} &= \begin{cases} 1, & i \not\equiv_2 j, \\ 0, & \text{otherwise}, \end{cases} \\ m^{(b)}_{(i,j),(i,j)} &= \begin{cases} 1, & i \equiv_2 j \equiv_2 1, \\ 0, & \text{otherwise}, \end{cases} \end{aligned} \tag{3}$$

where (i, j) is the generic pixel index.

The demosaicing problem is the inverse problem associated with the direct problem formulated in (1) and consists of finding an estimate $\widetilde{\mathbf{x}}$ of the ideal image, given the mosaiced image $\mathbf{y}$ and the operator M. Since M is singular, the demosaicing problem is ill-posed in the Hadamard sense because, in general, it does not admit a unique solution. Given $\mathbf{y}$, there are infinitely many feasible solutions since, at each pixel, the values of the two unmeasured channels do not contribute to the data. Therefore, regularization techniques are necessary to reduce the number of solutions.

3.2 The Regularization Model

We define our regularized solution $\widetilde{\mathbf{x}}$ as an argument of the minimum of the following energy function:

$$\begin{aligned} E(\mathbf{x}) = \|M(\mathbf{x} - \mathbf{y})\|_2^2 &+ \sum_{k=1}^{3} \sum_{c \in C_k} \varphi\left(N_c^k \mathbf{x}, N_{p_k(c)}^k \mathbf{x}\right) \\ &+ \sum_{k=1}^{3} \sum_{c \in C_k} \varphi\left(V_c^k \mathbf{x}, V_{p_k(c)}^k \mathbf{x}\right), \end{aligned} \tag{4}$$

where $\|\cdot\|_2$ denotes the Euclidean norm, and the first term of the right hand of (4) expresses a data fidelity constraint, which is identically null in the noiseless case. The second term in the right hand of (4) regulates the intra-channel smoothness of the involved image. The third term imposes a correlation between the different channels, i.e., an inter-channel smoothness. Intra-channel and inter-channel smoothness are measured through the operators N_c^k and V_c^k, respectively, and φ is a stabilizer that weights the degree of smoothness required and relaxes it when a discontinuity is expected.

Let us start by analyzing the form of the operator N_c^k, given by

$$N_c^k \mathbf{x} = \left\| \left(D_c^k \mathbf{x}^{(r)}, D_c^k \mathbf{x}^{(g)}, D_c^k \mathbf{x}^{(b)}\right) \right\|_2, \tag{5}$$

where D_c^k is a finite difference operator of order k applied to a suitable set c of adjacent pixels, called *clique* of order k. Therefore, from (5), it appears that N_c^k is the norm of the vector of the finite differences of the intensities of the red, green, and blue channels computed on the clique c of order k. The set C_k collects all cliques of order k. Each of such cliques is uniquely associated with a discontinuity of order k, labeled by a hidden line element.

To reconstruct the finest details in the images, we consider finite differences and then discontinuities of the first, second, and third order, that is $k = 1, 2, 3$. The geometry of the associated cliques is described in Appendix A.

The edges of the first order separate homogeneous patches in the image, the edges of the second order mark the slope of linearly varying areas, and the edges of the third order are associated with the intensity discontinuities in regions where intensity varies quadratically.

As the inter-channel correlation aims to maintain the clue of the objects in the image, the finite difference operators should behave similarly in all three channels. So we define the operator V_c^k as follows:

$$V_c^k \mathbf{x} = \left\| \left(D_c^k \mathbf{x}^{(r)} - D_c^k \mathbf{x}^{(g)}, D_c^k \mathbf{x}^{(r)} - D_c^k \mathbf{x}^{(b)}, \right. \right. \\ \left. \left. D_c^k \mathbf{x}^{(g)} - D_c^k \mathbf{x}^{(b)} \right) \right\|_2 , \tag{6}$$

which is the norm of the vector of the inter-channel differences of the intra-channel k-order derivatives. Again, a hidden line variable is implicitly associated with the clique c for each order $k = 1, 2, 3$. These further sets of hidden line variables mark the discontinuities between areas having homogeneous clues.

In (4), N_c^k and V_c^k are weighted by suitable stabilizers. These stabilizers should regulate the degree of smoothness required in the two cases and relax it when discontinuities are expected and dependent on their amplitude. In (4), we adopted the same parametric stabilizer φ for both the operators N_c^k and V_c^k and let its parameters possibly vary in the two terms (see also [14]).

For a more accurate reconstruction, edges must not be thick, or the object contours must not be blurred equivalently. To this aim, it is advisable to inhibit the creation of discontinuities at two adjacent cliques. Specifically, to prevent double edges of order k, simultaneous discontinuities at the cliques c and the previous one $p_k(c)$ should be inhibited (see Appendix A for the definition of adjacent cliques).

When $p_k(c)$ is not defined for the mixed cliques and the cliques on the border of the image, we automatically assume that the adjacent discontinuity is null.

Having in mind the above-described properties to be featured by the stabilizer, we adopt herein a bivariate function $\varphi : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ (see also [15]), defined by

$$\varphi(t_1, t_2) = \begin{cases} g_1(t_1), & \text{if } |t_2| \leq s, \\ \left(1 - \frac{2(|t_2|-s)^2}{(\zeta-s)^2}\right) g_1(t_1) + \frac{2(|t_2|-s)^2}{(\zeta-s)^2} g_2(t_1), & \text{if } s < |t_2| \leq \frac{\zeta+s}{2}, \\ \frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2} g_1(t_1) + \left(1 - \frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\right) g_2(t_1), & \text{if } \frac{\zeta+s}{2} < |t_2| < \zeta, \\ g_2(t_1), & \text{if } |t_2| \geq \zeta, \end{cases} \tag{7}$$

where

$$s = \frac{\sqrt{\alpha}}{\lambda}, \tag{8}$$

and ζ is chosen in such a way that $\zeta - s$ is a positive and sufficiently small quantity, and for $i = 1, 2$ it is

$$g_i(t_1) = \begin{cases} \lambda^2 t_1^2, & \text{if } |t_1| < q_i, \\ \alpha_i - \frac{\tau}{2}(|t_1| - r_i)^2, & \text{if } q_i \leq |t_1| \leq r_i, \\ \alpha_i, & \text{if } |t_1| > r_i, \end{cases} \tag{9}$$

$$\alpha_i = \begin{cases} \alpha, & \text{if } i = 1, \\ \alpha + \varepsilon, & \text{if } i = 2, \end{cases} \tag{10}$$

$$q_i = \frac{\sqrt{\alpha_i}}{\lambda^2} \left(\frac{2}{\tau} + \frac{1}{\lambda^2} \right)^{-1/2}, \tag{11}$$

τ is a large enough real constant, and

$$r_i = \frac{\alpha_i}{\lambda^2 q_i}, \quad i = 1, 2. \tag{12}$$

The graph of the function in (7) for specific values of the parameters is shown in Fig. 3 (a). Note that when $\zeta - s = 0$ and τ tends to $+\infty$, we have an ideal model with Boolean line variables (see Fig. 3 (b)). We have set $\zeta = s + 10^{-4}$ and $\tau = 10^4$ in our experimental results.

In (7)–(12) λ^2 is a regularization parameter, which has the role of determining the smoothness of the solution, α is a cost, which we have to pay whenever we introduce the image discontinuities. When there are some parallel discontinuities, we add the quantity ε to this cost. Thus, the value s in (8) is just the threshold necessary to introduce a discontinuity in the image when there are no parallel discontinuities.

In general, the stabilizer's analytical form determines the amplitude of the discontinuities in the reconstructed image by promoting on-off discontinuities of large amplitude above a given threshold or more slowly varying discontinuities of graded amplitudes. In the primal-dual formalism, the first type's stabilizers implicitly address "hard", Boolean line elements, while the second type addresses hidden "soft" line elements, ideally valued in [0, 1].

We recall that the functions g_i, $i = 1, 2$, defined in (9), are approximations of class C^1 of the classical truncated parabola defined in [9] (see also [10], [14], [15]) that, when used as a stabilizer, implicitly addresses a Boolean line process. In the bivariate case, the function φ defined in (7) possesses the same characteristic when τ tends to $+\infty$ and ζ is very close to s. The actual form we propose is an approximation of such a function, with the property of class C^1, which is essential for converging the minimization algorithm (see Subsection 4.1).

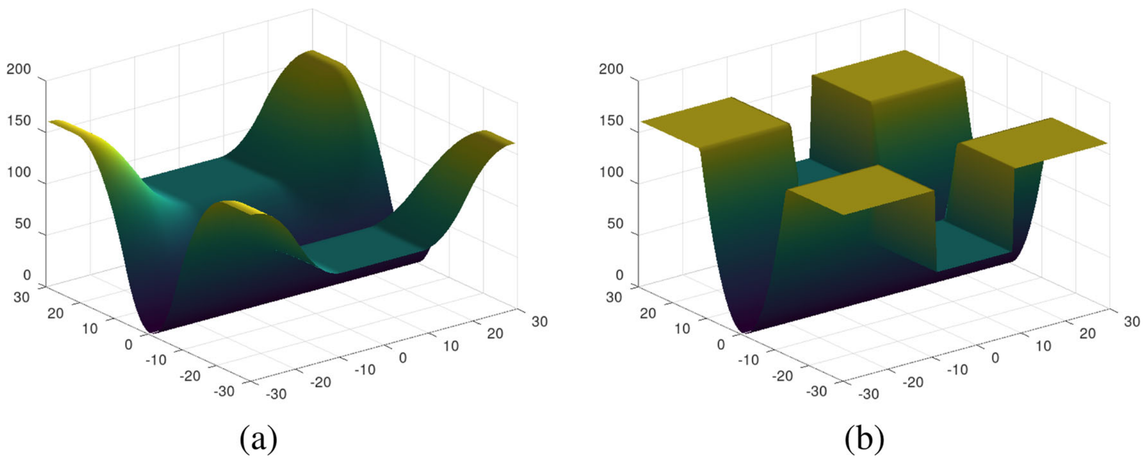

Fig. 3 (**a**) φ with $\lambda = 1$, $\alpha = 80$, and $\varepsilon = 80$; (**b**) φ in the case of $\zeta = s$ and $\tau = +\infty$

4 Graduated Componentwise Non-Convexity Algorithm

We observe that the function φ defined in (7), as a function of two variables, is not convex, and hence neither is the energy function $E(\mathbf{x})$, defined in (4) as a function of $3\,n\,m$ variables. Thus, to minimize E, we determine a finite family of approximating functions $\{E^{(p)}\}_p$, where $E^{(0)}$ is componentwise convex, and $E^{(2)}$ is the original energy function E. The initial point to minimize the componentwise convex approximation is found using the *Local Edge Preserving* (LEP) algorithm used in [13] for demosaicing. The LEP is a high-speed algorithm consisting of two phases. In the first phase, the missing components are determined by a weighted mean, which guarantees the preservation of the edges. In the second phase, the differences between the colors of the channels are imposed to be constant within homogeneous areas. Our algorithm in its whole is called *graduated componentwise non-convexity* (GCNC) and is presented in Algorithm 1.

Algorithm 1 GCNC algorithm

```
Input: A mosaiced image $\mathbf{y}$;
Output: An estimation of the demosaiced image $\mathbf{x}$;
initialize $\mathbf{x}$ by the LEP algorithm in [13];
$p = 0$;
while $p \neq 2$ do
    find the minimum of the function $E^{(p)}$ starting from the initial point $\mathbf{x}$;
    set $\mathbf{x}$ to the reached minimizer;
    update the parameter $p$;
end while
```

Note that our algorithm can be seen as a variant of the *graduated non-convexity* (GNC) algorithm (see also [7], [15], [57], [60], [61], [62], [65]).

To construct the first componentwise convex approximation $E^{(0)}$, we find a componentwise convex approximation for the stabilizers in (7), since the data term in (4) is globally convex. Such componentwise convex approximations can be constructed based on a componentwise convex approximation of the bivariate function φ, as shown in Appendix D. To do this, we proceed as follows. First of all, we approximate the functions $g_i(t_1)$ with the following convex approximations given by

$$\overline{g_i}(t_1) = \begin{cases} \lambda^2\, t_1^2, & \text{if } |t_1| \le q_i, \\ \lambda^2\,(2\,q_i|t_1| - q_i^2), & \text{if } |t_1| \ge q_i, \end{cases} \qquad i = 1, 2 \tag{13}$$

(see also [60]). Moreover, we find an approximation, convex concerning the variable t_2, of the function φ defined in (7), in the following way:

$$\overline{\varphi}(t_1, t_2) = \frac{\overline{t}^2 - t_2^2}{\overline{t}^2}\,\overline{g_1}(t_1) + \frac{t_2^2}{\overline{t}^2}\,\overline{g_2}(t_1) = \overline{g_1}(t_1) + \frac{t_2^2}{\overline{t}^2}\,(\overline{g_2}(t_1) - \overline{g_1}(t_1)), \tag{14}$$

where $\overline{t}$ is the maximum value that a finite difference operator can assume ($\overline{t} = 2^k \cdot \sqrt{2} \cdot 255$, $k = 1, 2, 3$, for the light intensity of the images in the range [0,255]). It is easy to check that $g_2 - g_1$ is convex since $0 < q_1 < q_2$.

We recall that $t_1 = N_c^k\mathbf{x}$, $t_2 = N_{p_k(c)}^k\mathbf{x}$ in the second term of the right hand of (4), and $t_1 = V_c^k\mathbf{x}$, $t_2 = V_{p_k(c)}^k\mathbf{x}$ in the third term of the right hand of (4). Let us fix $k \in \{1, 2, 3\}$ and $c \in C_k$, and choose $\Xi_c^k \in \{N_c^k, V_c^k\}$. So, t_1 is a function of $\mathbf{x}$, and $t_1(\mathbf{x}) = \Xi_c^k(\mathbf{x})$. In particular, t_1 depends only on the variables involved in the clique c. Analogously, $t_2(\mathbf{x}) = \Xi_{p_k(c)}^k(\mathbf{x})$ depends only on the variables involved in the clique $p_k(c)$. Note that the function $\overline{\varphi}$ defined in (14) is componentwise convex. However, the function $\overline{\Phi}(\mathbf{x}) = \overline{\varphi}(t_1(\mathbf{x}), t_2(\mathbf{x}))$ is not componentwise convex concerning the components of the vector $\mathbf{x} \in \mathbb{R}^{3nm}$. This is because $c \cap$

$p_k(c) \neq \emptyset$. However, if we choose $t_2(\mathbf{x}) = \Xi^k_{\pi_k(c)}(\mathbf{x})$ instead of $\Xi^k_{p_k(c)}(\mathbf{x})$, then the function t_2 is componentwise convex with respect to $\mathbf{x}$, as shown in Appendix D, since $c \cap \pi_k(c) = \emptyset$. Thus, to define the family of the approximations for the algorithm GCNC, we proceed as follows.
If $p \in [0, 1]$, put

$$E^{(p)}(\mathbf{x}) = \|M(\mathbf{x}-\mathbf{y})\|_2^2 + \sum_{k=1}^{3}\sum_{c\in C_k} \overline{\varphi}\left(N^k_c\mathbf{x},\, p\, N^k_{p_k(c)}\mathbf{x} + (1-p)\, N^k_{\pi_k(c)}\mathbf{x}\right) + + \sum_{k=1}^{3}\sum_{c\in C_k} \overline{\varphi} \times \left(V^k_c\mathbf{x},\, p\, V^k_{p_k(c)}\mathbf{x} + (1-p)\, V^k_{\pi_k(c)}\mathbf{x}\right). \qquad (15)$$

When $p \in [1, 2]$, set

$$E^{(p)}(\mathbf{x}) = \|M(\mathbf{x}-\mathbf{y})\|_2^2 + \sum_{k=1}^{3}\sum_{c\in C_k} \varphi^{(p)}\left(N^k_c\mathbf{x},\, N^k_{p_k(c)}\mathbf{x}\right) + + \sum_{k=1}^{3}\sum_{c\in C_k} \varphi^{(p)}\left(V^k_c\mathbf{x},\, V^k_{p_k(c)}\mathbf{x}\right), \qquad (16)$$

where

$$\varphi^{(p)}(t_1, t_2) = (2-p)\,\overline{\varphi}(t_1, t_2) + (p-1)\,\varphi(t_1, t_2). \qquad (17)$$

In Appendix B, we will prove that for each $p \in [1, 2]$, $\varphi^{(p)}$ satisfies the duality conditions that guarantee the edge-preserving properties and the inhibition of double edges [14]. Note that, for $p \in [0, 1]$, the stabilizer $\varphi(t_1, t_2)$ is equal to $\overline{\varphi}(t_1, t_2)$, and hence fulfils the same properties. Furthermore, in [11], it is proved that the associated line process is non-Boolean. The hidden line elements become Boolean as far as p tends to 2. However, we experimentally observed that, in natural images, graded discontinuities can help prevent the aliasing effect. Thus, in the experiments, we stop the minimization algorithm at a suitable value of p different from 2, as explained in Sect. 5.

4.1 The NL-SOR Algorithm

To minimize each approximation $E^{(p)}$, we use a *nonlinear successive over relaxation* (NL-SOR) algorithm, which is widely used in the literature (see also [7, 9, 16]). The NL-SOR is defined as in Algorithm 2, where $\epsilon > 0$ is a fixed threshold, $\omega > 0$ is the accelerator parameter,

$$T > \max_{\substack{i=1,2,\ldots,nm\\ e=r,g,b}} \max_{\mathbf{x}} \left\{ \frac{\partial^2_+ E^{(p)}(\mathbf{x})}{(\partial x_i^{(e)})^2}, \frac{\partial^2_- E^{(p)}(\mathbf{x})}{(\partial x_i^{(e)})^2} \right\},$$

and the symbols ∂^2_+ and ∂^2_- denote the right and left second partial derivatives, respectively.

Algorithm 2 NL-SOR algorithm

```
Input: An energy function $E^{(p)}$;
Output: An estimation of the minimizer $\mathbf{x}$;
given the initial vector $\mathbf{x}^{(0)}$
$l = 1$;
while $\|\nabla E^{(p)}(\mathbf{x})\| > \epsilon$ do
    for $i = 1, 2, \ldots, nm$ do
        for $e = r, g, b$ do
            $(x_i^{(e)})^{(l+1)} = (x_i^{(e)})^{(l)} - \frac{\omega}{T}\frac{\partial E^{(p)}(\mathbf{x}^{(l)})}{\partial x_i^{(e)}}$;
        end for
    end for
    $l = l + 1$;
end while
```

In our experimental results, we have set $\omega = 0.2$ and $\epsilon = 3\sqrt{nm}$, where $n \times m$ is the dimension of the involved image. We chose these values after a long experimental phase, where we monitored the trend of the energy function in each step of the algorithm to obtain high-performance results without excessive computational time.

In [16, Theorem 2], the convergence of the algorithm is proved when $E^{(p)}$ is strictly convex and of class C^2. However, such a theorem cannot be applied to our setting since our first approximation is componentwise convex and C^1 but neither strictly convex nor C^2. Thus, in Appendix C, we propose an extension of [16, Theorem 2] to prove that in our case, when $p = 0$, the algorithm stops in correspondence with a stationary point. Note that for $p \neq 0$, we have no guarantee of converging to a stationary point of $E^{(p)}$ using the NL-SOR algorithm. However, the experimental results confirm that excellent solutions can still be achieved (see Sect. 5).

5 Experimental Results

In assessing the efficacy of the proposed demosaicing algorithm, we analyze diverse color image datasets, encompassing the Kodak image dataset [43], the McMaster image dataset [75], the Microsoft Demosaicing Canon Dataset [38], and the Microsoft Demosaicing Panasonic Dataset [38]. The Kodak dataset comprises 24 comprehensive full-color images sized at 512 × 768 pixels each. Conversely, the McMaster image dataset features 18 images with dimensions of 500 × 500 pixels, derived from original high-resolution images sized at 2310 × 1814 pixels. Renowned for their widespread adoption as benchmark datasets within demosaicing and various other color image processing domains, the Kodak and McMaster datasets serve as pivotal standards.

Table 1 Parameters used in the noiseless case for weighting the intra-channel smoothness measure (5)

	Derivative order		
	$k = 1$	$k = 2$	$k = 3$
λ	0.1	0.2	0.2
α	9	3	2
ε	4.5	1.5	1

Table 2 Parameters used in the noiseless case for weighting the inter-channel correlation (6)

	Derivative order		
	$k = 1$	$k = 2$	$k = 3$
λ	0.5	0.28	0.28
α	3	3	3
ε	1.5	1.5	1.5

Moreover, the Microsoft Demosaicing Dataset Canon encompasses 57 images at 210 × 318 pixels, while the Microsoft Demosaicing Dataset Panasonic comprises 500 images of identical dimensions. These raw images originate from distinct camera sources, namely a Canon EOS 550D and a Panasonic Lumix DMC-LX3, respectively. Such datasets collectively facilitate a comprehensive evaluation of the proposed demosaicing algorithm across varied imaging scenarios and devices.

We implement the proposed algorithm in C language and run it in a Linux Ubuntu environment on a computer with an i5-9400F processor at 2.90 GHz. However, minimizing a non-convex function is a complex task; therefore, high computational costs are still necessary. For example, in the case of the Kodak set, the proposed algorithm has an average computation time of 12.493 min for each image.

The free parameters λ, α, and ε appearing in our energy function have been calibrated on the images of the Kodak dataset and then used for the other datasets. We employed a trial-and-error strategy to look for the parameters that give the best average color peak signal-to-noise ratio (CPSNR) on all the 24 images. The CPSNR quality index, a key metric in our calibration process, is defined as:

$$CPSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right), \tag{18}$$

Table 3 CPSNRs for noiseless Kodak images

Image	[29]	[33]	[50]	[54]	[24]	[19]	[1]	[2]	[58]	[41]	[59]	Proposed
1	37.70	35.17	39.37	38.22	35.64	39.96	37.31	39.86	37.81	36.28	38.84	**40.71**
2	39.57	39.34	40.71	38.18	36.46	**40.99**	38.90	**40.99**	38.61	40.25	39.70	40.90
3	41.45	41.52	43.19	42.04	37.25	43.26	41.76	42.86	37.28	42.66	42.94	**43.32**
4	40.03	38.87	41.29	40.04	36.74	40.56	40.40	41.25	41.05	41.20	40.84	**41.84**
5	37.46	35.70	38.70	38.04	35.45	38.31	37.44	38.41	37.91	37.36	38.29	**38.95**
6	38.50	37.55	40.05	39.70	36.39	**41.00**	39.59	40.31	39.34	38.70	40.67	40.22
7	41.77	40.87	42.83	42.10	37.07	42.64	41.85	42.94	41.59	42.55	42.85	**43.62**
8	35.08	33.80	36.42	36.08	34.59	**37.35**	34.58	37.05	35.49	34.55	35.30	37.25
9	41.72	41.10	43.28	42.15	37.46	43.42	41.77	**43.44**	42.40	42.06	41.59	43.31
10	42.02	40.77	42.70	42.15	37.26	42.83	41.80	**43.12**	42.27	42.06	41.98	42.70
11	39.14	37.48	40.22	39.78	36.41	40.66	39.09	**40.92**	39.22	38.96	39.82	40.51
12	42.51	41.81	43.53	42.94	37.56	44.13	43.01	44.01	43.49	42.86	43.56	**44.41**
13	34.30	31.41	35.29	34.94	33.68	36.03	34.97	35.94	34.19	32.61	35.40	**36.27**
14	35.60	35.50	37.95	36.34	35.07	37.10	35.79	36.99	36.27	37.59	37.50	**38.26**
15	39.35	38.02	40.21	39.15	36.22	39.84	39.39	40.03	39.30	38.90	38.85	**40.37**
16	41.76	41.37	43.62	43.27	37.53	**44.47**	43.62	43.74	42.65	42.58	43.56	43.77
17	41.11	39.25	42.01	41.83	41.09	41.77	41.17	**42.24**	41.15	40.88	41.31	41.54
18	37.45	35.20	37.47	37.13	35.98	**37.96**	37.12	37.89	37.05	35.88	37.09	37.45
19	39.46	38.44	41.27	40.15	40.20	**41.79**	39.78	41.46	40.15	39.56	40.54	41.11
20	40.66	39.23	41.00	40.39	32.49	41.71	40.46	**41.85**	40.72	40.28	41.23	41.59
21	38.66	36.56	39.74	39.27	36.47	39.99	38.57	**40.37**	38.48	37.82	39.44	40.20
22	37.55	36.46	**38.87**	38.25	37.32	38.48	37.33	38.69	38.40	38.39	38.26	38.51
23	41.88	41.88	42.41	40.40	39.45	43.20	42.00	43.04	38.75	43.28	43.30	**43.91**
24	34.78	33.42	35.63	35.37	34.32	35.39	34.52	35.21	35.37	34.33	**35.64**	34.80
mean	39.15	37.95	40.32	39.50	36.59	40.54	39.26	40.53	39.12	39.23	39.94	**40.65**

Bold indicates the best result

Table 4 CPSNRs for McMaster noiseless images

Image	[29]	[33]	[17]	[75]	[41]	[59]	Proposed
1	25.59	26.63	27.69	29.56	29.41	29.74	**30.02**
2	32.46	33.64	34.47	**35.67**	35.35	35.30	35.51
3	31.63	31.42	32.93	33.29	34.05	**34.92**	34.17
4	33.23	33.63	36.28	36.63	38.00	38.07	**38.48**
5	29.98	31.01	32.00	34.79	34.43	**35.52**	**35.52**
6	31.98	33.87	35.55	39.26	38.83	39.54	**39.81**
7	37.82	35.99	36.87	36.00	37.04	**39.87**	39.81
8	36.62	36.46	37.47	37.76	37.30	**39.53**	38.93
9	33.28	34.51	36.21	37.84	36.84	37.93	**38.18**
10	34.97	36.01	37.56	39.24	39.12	39.34	**39.57**
11	35.97	36.73	38.39	40.02	40.21	**40.38**	39.81
12	35.78	36.64	37.39	39.15	39.84	**40.26**	39.27
13	37.47	38.76	40.34	41.60	40.66	41.02	**41.63**
14	36.25	37.43	38.53	**39.45**	39.11	39.14	39.26
15	36.35	37.33	38.29	**39.54**	39.25	39.48	39.44
16	29.02	30.05	31.17	34.03	35.42	**35.68**	34.36
17	27.99	28.63	30.41	33.56	33.19	34.41	**35.20**
18	32.49	33.30	34.20	35.38	36.41	**36.56**	35.10
mean	33.27	34.00	35.32	36.82	36.92	**37.59**	37.45

Bold indicates the best result

Table 5 CPSNRs for Microsoft images—noiseless

Dataset	[59]	Prop
Canon	42.27	**43.54**
Panasonic	39.55	**41.24**

Bold indicates the best result

Table 6 Regularization parameters depending of the noise standard deviation σ

	Derivative order		
	$k = 1$	$k = 2$	$k = 3$
λ	0.156σ	0.094σ	0.094σ
α	$0.391\sigma^2$	$0.078\sigma^2$	$0.078\sigma^2$
ε	$0.195\sigma^2$	$0.039\sigma^2$	$0.039\sigma^2$

where MSE is the mean square error between the original image and the demosaiced one that, for color images, is defined as the arithmetic average of the mean square errors on the three channels.

We chose to increase p with a step of 0.01. Note that, for $p = 2$, the discontinuities present in the reconstructed image are Boolean, whereas, for $p \neq 2$, we have reconstructions with non-Boolean line variables, as stated in the duality theorem given in [15].

Reconstructions obtained with Boolean lines are often disturbed by the aliasing effect, so that it may be preferable to stop the algorithm at a value $\bar{p}$ smaller than 2. The optimal value of $\bar{p}$ has been set as the one for which the sum of the MSEs of all the 24 Kodak images reconstructed with that p is minimum. Calling $\eta_j(p)$ the MSE between the ideal image and the minimizer of the approximated energy function for the $j - th$ image at a given p, it is:

$$\bar{p} = \arg\min_{p} \left\{ \sum_{j=1}^{24} \eta_j(p) \right\}. \tag{19}$$

The proposed algorithm has been applied to both noiseless and noisy images, and the performance of the method has been compared with some of the most popular and best-performing methods in the literature.

5.1 Noiseless Images

In the energy E (4), stabilizer φ depends on the free parameters λ, α and ε. In the second addend of the right hand of (4), we used φ with the free parameters reported in Table 1, which we found to be the best for noiseless mosaiced images.

In the third term of the right hand of (4), the best free parameters for stabilizer φ to be used in the noiseless case are indicated in Table 2.

Given all these free parameters, we found $\bar{p} = 1.42$.

For the noiseless Kodak dataset, we compared the proposed method with the algorithms in [29], [33], [50], [54], [19], [1], [2] [58], [41], and [59]. In particular, we used the source code available on the web to obtain the results of the algorithm in [59]. The results obtained on the Kodak dataset are reported in Table 3. As usual with the Kodak images, we removed the 3 pixels wide external frame to compute the errors. Our method exhibits the highest CPSNR in more images than the other methods (the highest CPSNR is highlighted in boldface for each image). Furthermore, the proposed method achieves the best result on average.

For the noiseless McMaster dataset, a comparison has been made with the algorithms in [29], [33], [17], [75], [41], and [59], respectively. In particular, to collect the results of the algorithm in [29], we used the original MATLAB code provided by the authors, and for the results of the algorithms in [41] and [59], we used the source code available on the web. The results obtained are reported in Table 4. The proposed method achieved the best result in more than 44% of cases. However, this time, the method presented in [59] is slightly better on average. We thus used the Microsoft Demosaicing Canon Dataset [38] and the Microsoft Demosaicing Panasonic Dataset [38] to further compare our method with the one proposed in [59]. Table 5 presents the obtained average results on both datasets; the proposed algorithm always obtains the best results.

Table 7 CPSNRs for noisy Kodak images, $\sigma = 16$

Image	Bilinear	[29]	[34]	[76]	[55]	[20]	[20] variant	proposed
1	23.38	24.24	22.35	27.63	27.71	**28.18**	28.14	28.11
2	25.86	24.50	23.55	28.75	30.86	31.01	28.98	**31.47**
3	25.98	24.47	23.84	31.51	31.81	32.58	32.67	**32.71**
4	25.84	24.38	23.48	30.10	30.82	31.34	30.69	**31.36**
5	23.68	24.42	22.85	27.70	28.02	28.51	**28.60**	28.27
6	24.09	24.40	23.09	28.84	28.87	**29.51**	29.40	29.01
7	25.78	24.43	23.47	30.67	31.24	**32.29**	31.96	32.03
8	21.89	24.18	22.18	27.19	27.37	**28.40**	28.32	27.66
9	25.57	24.36	23.66	31.42	31.51	32.61	**32.83**	32.53
10	25.58	24.38	22.85	31.11	31.38	32.58	**32.60**	32.21
11	24.78	24.45	23.28	29.36	29.62	**30.20**	30.17	30.02
12	25.69	24.44	23.72	31.13	31.44	32.27	32.19	**32.34**
13	22.03	24.12	21.99	26.51	26.43	**26.78**	26.63	26.67
14	24.76	24.25	22.96	28.35	28.44	27.99	28.65	**29.01**
15	25.64	24.79	23.94	30.14	30.85	31.21	30.76	**31.30**
16	25.31	24.36	23.44	30.52	30.50	31.33	**31.37**	30.85
17	25.70	24.68	23.79	30.90	31.02	31.97	**32.05**	31.81
18	24.29	24.39	23.01	28.01	28.56	**28.99**	28.35	28.83
19	24.30	24.35	22.93	29.59	29.78	**30.74**	30.56	30.21
20	26.00	25.44	24.80	29.95	30.45	30.82	30.77	**30.93**
21	24.43	24.32	23.21	29.06	29.40	**30.07**	29.76	30.00
22	25.13	24.28	23.16	29.22	29.55	29.87	29.69	**29.91**
23	26.07	24.47	23.96	31.02	32.48	33.10	31.65	**33.31**
24	26.98	25.26	25.50	27.98	28.20	**28.81**	28.64	28.48
mean	24.95	24.47	23.38	29.44	29.85	**30.47**	30.23	30.38

Bold indicates the best result

5.2 Noisy Images

In a second set of experiments, we considered noisy images corrupted by independent Gaussian noise, with zero mean and different standard deviation values σ. This time, the best free parameters for stabilizer φ to be used in the second and third addend of the right hand of (4) are shown in Table 6 .

As done for the noiseless images, for each value of the noise variance, the suitable value $\bar{p}$ for stopping the algorithm has been determined according to the criterion established in (19). The following empirical law that relates $\bar{p}$ to σ has also been found:

$$\bar{p}(\sigma) = \frac{3}{40}\sigma + \frac{4}{5}. \tag{20}$$

In the noisy case, we compared our method with the algorithms in [29] and [34] by using the original MATLAB code provided by the authors and with the algorithms in [20], [76], and [55], by using the source codes available in the authors' web pages. The CPSNR values computed for the case $\sigma = 16$ on the Kodak dataset are shown in Table 7.

Although the performance of our method is still satisfactory, this time, the method in [20] is slightly superior.

We then computed another quality index, sometimes used in the demosaicing problem, i.e., the S-CIELAB metric. This metric indicates the percentage of color distortion between two images and accounts for the spatial-color sensitivity of the human eye [73] [37]. Since it returns a pixel-by-pixel matrix of errors, we assumed the mean of the S-CIELAB matrix coefficients as the representative error index for the entire image. The results obtained for the case $\sigma = 16$ on the Kodak dataset, along with the results of the most performing among the methods used for comparison, are shown in Table 8.

The results in this case are excellent. The situation is even better when the noisy mosaiced McMaster images are processed. The CPSNR results obtained for the same amount of noise ($\sigma = 16$), along with the results of the most performing methods used for comparison on the Kodak dataset, are shown in Table 9.

It is apparent that this time, our method outperforms the other, with much higher values of CPSNR, which are only slightly lower than those we obtained in the noiseless case for

Table 8 S-CIELAB errors for noisy Kodak images, $\sigma = 16$

Image	[76]	[55]	[20]	[20] variant	Proposed
1	4.11	4.22	3.82	3.82	**3.41**
2	3.34	2.81	2.74	3.19	**2.64**
3	3.28	3.24	2.97	2.89	**2.52**
4	3.44	3.32	3.16	3.23	**2.89**
5	4.20	4.07	4.02	3.92	**3.81**
6	3.86	3.83	3.47	3.48	**3.23**
7	3.54	3.43	3.14	3.21	**2.82**
8	4.48	4.57	4.08	4.02	**3.96**
9	3.10	3.19	2.85	2.72	**2.38**
10	3.16	3.21	2.81	2.73	**2.54**
11	3.26	3.31	3.02	2.98	**2.79**
12	3.10	3.00	2.75	2.74	**2.30**
13	4.65	4.78	4.42	4.47	**4.30**
14	3.87	3.97	3.93	3.67	**3.39**
15	3.09	2.82	2.82	2.79	**2.56**
16	3.23	3.36	2.97	2.89	**2.68**
17	2.67	2.81	2.46	2.32	**2.24**
18	4.28	3.90	3.80	4.14	**3.62**
19	3.69	3.66	3.32	3.37	**3.02**
20	3.85	3.36	3.26	3.30	**3.09**
21	3.67	3.63	3.26	3.30	**2.90**
22	4.14	3.92	3.77	3.92	**3.59**
23	3.22	2.94	2.76	2.99	**2.58**
24	4.28	4.10	3.72	3.81	**3.58**
mean	3.65	3.56	3.31	3.33	**3.04**

Bold indicates the best result

Table 9 CPSNRs for noisy McMaster images, $\sigma = 16$

Image	[76]	[55]	[20]	[20] variant	Proposed
1	24.01	24.59	22.62	24.12	**29.36**
2	27.86	28.99	28.74	28.07	**34.85**
3	26.94	27.34	27.51	27.63	**33.53**
4	28.49	29.23	28.76	29.74	**37.44**
5	27.18	27.96	26.79	27.46	**34.55**
6	28.16	29.25	27.77	28.38	**39.12**
7	29.12	29.03	29.61	29.66	**36.04**
8	30.01	30.27	30.65	30.88	**38.47**
9	28.24	29.54	29.00	28.43	**37.65**
10	28.37	30.38	29.90	28.55	**39.09**
11	29.01	30.97	30.59	29.15	**39.79**
12	28.41	30.91	31.39	29.10	**38.98**
13	29.96	32.44	33.31	30.82	**40.94**
14	28.86	31.56	32.04	29.18	**38.52**
15	29.51	31.62	31.86	29.94	**39.08**
16	25.24	26.77	25.71	25.17	**34.24**
17	25.94	26.86	24.26	25.42	**34.69**
18	27.43	28.52	28.46	27.71	**35.00**
mean	27.93	29.24	28.83	28.30	**36.74**

Bold indicates the best result

the same dataset. This excellent performance can be ascribed once again to our very fine modeling of natural images in terms of local variations inside and between the color channels.

Furthermore, in Table 10, it is possible to observe the excellent behavior of the proposed method on the two Microsoft datasets, compared with the method in [59] and for different values of the standard deviation σ of the noise corrupting the data.

6 Conclusion

We approached the joint demosaicing and denoising of color images within a regularization framework, irrespective of the CFA employed to generate the data. A central feature of our method is adopting local image smoothness models that implicitly account for “soft” edges at low- and high-frequency levels. Soft edges are essential for preventing the aliasing effect and preserving the fine details in the image, especially when noise is present. In our method, this is achieved by using stabilizers possessing two main features: on the one hand, we include derivatives up to the third degree; on the other hand, these derivatives are weighted through non-quadratic functions. A duality theorem ensures that the implicitly addressed edge fields are equivalent to geometrically consistent image discontinuities at each cycle of the iterative algorithm employed to find the regularized solution.

The experimentation over the Kodak 24-image dataset, the McMaster (IMAX) 18-image dataset, the Microsoft Demosaicing Canon 57-image dataset, and the Microsoft Demosaicing Panasonic 500-image dataset under the Bayer CFA, with a high level of noise, demonstrates the good performance of our method against some of the best-performing demosaicing algorithms proposed so far.

Future developments involve experimenting with the algorithm on data from other CFAs and including the filter blurs that are known in this kind of application. For this latter issue, the extension of the algorithm is straightforward, since the data term is still convex, so the same approximations for the energy function can also be exploited.

Table 10 CPSNRs for Microsoft images—noisy

Dataset	[59] $\sigma = 3$	[59] $\sigma = 10$	prop. $\sigma = 3$	prop. $\sigma = 10$
Canon	37.14	29.81	**40.23**	**36.89**
Panasonic	35.90	29.01	**38.12**	**34.64**

Bold indicates the best result

Appendix A: Geometry of the Cliques and Expression of the Associated Finite Differences

The stabilizers used in this paper are functions of the finite differences D_c^k of order k applied to sets c consisting of adjacent pixels. We call *clique of order* k the set of pixels on which the finite difference of order k is well defined. We take $k = 1, 2, 3$ in order to reconstruct the finest details in images. Figures 4, 5 and 6 show the geometry of the sets c for the three orders of finite differences, respectively. As we can see, the cliques can be classified as *vertical* (Figs. 4 (a), 5 (a), 6 (a)), *horizontal* (Figs. 4 (b), 5 (c), 6 (d)), and *mixed* (Figs. 5 (b), 6 (b) and (c)).
The vertical cliques consist of the following pixels:

$$c = \{(i, j), (i + 1, j), \ldots, (i + k, j)\},\\ i = 1, \ldots, n - k,\ j = 1, \ldots, m,\ k = 1, 2, 3, \tag{A1}$$

while the horizontal cliques have the form

$$c = \{(i, j), (i, j + 1), \ldots, (i, j + k)\},\\ i = 1, \ldots, n,\ j = 1, \ldots, m - k,\ k = 1, 2, 3. \tag{A2}$$

Let us now describe how finite differences are computed at a generic clique c for a generic color channel $\mathbf{x}^{(e)}$, $e \in \{r, g, b\}$. When $k = 1$, as it is seen in Fig. 4, we have two different kinds of finite difference operators, associated with a horizontal and a vertical finite difference, given by

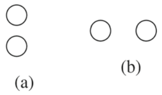

Fig. 4 Geometry of the sets c for $k = 1$

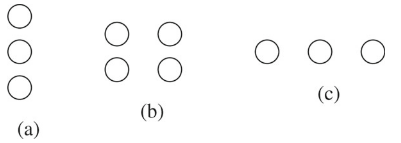

Fig. 5 Geometry of the sets c for $k = 2$

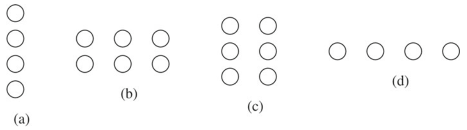

Fig. 6 Geometry of the sets c for $k = 3$

$$D_c^1\mathbf{x}^{(e)} = \begin{cases} x_{(i,j)}^{(e)} - x_{(i+1,j)}^{(e)} \text{ in } Fig.\ 4(a); \\ x_{(i,j)}^{(e)} - x_{(i,j+1)}^{(e)} \text{ in } Fig.\ 4(b), \end{cases} \tag{A3}$$

respectively. When $k = 2$, we have three different kinds of finite difference operators, expressed by

$$D_c^2\mathbf{x}^{(e)} = \begin{cases} x_{(i,j)}^{(e)} - 2x_{(i+1,j)}^{(e)} + x_{(i+2,j)}^{(e)} & \text{in } Fig.\ 5(a); \\ x_{(i,j)}^{(e)} - 2x_{(i,j+1)}^{(e)} + x_{(i,j+2)}^{(e)} & \text{in } Fig.\ 5(b); \\ x_{(i,j)}^{(e)} - x_{(i+1,j)}^{(e)} - x_{(i,j+1)}^{(e)} + x_{(i+1,j+1)}^{(e)} & \\ \qquad \text{in } Fig.\ 5(c). & \end{cases} \tag{A4}$$

When $k = 3$, we get four different kinds of finite difference operators, given by

$$D_c^3\mathbf{x}^{(e)} = \begin{cases} x_{(i,j)}^{(e)} - 3x_{(i+1,j)}^{(e)} + 3x_{(i+2,j)}^{(e)} - x_{(i+3,j)}^{(e)} \\ \qquad \text{in } Fig.\ 6(a); \\ x_{(i,j)}^{(e)} - 3x_{(i,j+1)}^{(e)} + 3x_{(i,j+2)}^{(e)} - x_{(i,j+3)}^{(e)} \\ \qquad \text{in } Fig.\ 6(b); \\ x_{(i,j)}^{(e)} - 2x_{(i+1,j)}^{(e)} + x_{(i+2,j)}^{(e)} - x_{(i,j+1)}^{(e)} \\ \quad +2x_{(i+1,j+1)}^{(e)} - x_{(i+2,j+1)}^{(e)} \quad \text{in } Fig.\ 6(c); \\ x_{(i,j)}^{(e)} - 2x_{(i,j+1)}^{(e)} + x_{(i,j+2)}^{(e)} - x_{(i+1,j)}^{(e)} \\ \quad +2x_{(i+1,j+1)}^{(e)} - x_{(i+1,j+2)}^{(e)} \quad \text{in } Fig.\ 6(d). \end{cases} \tag{A5}$$

Let us introduce the concept of *adjacent clique of order* k, which defines the non-parallelism constraint. Given a vertical clique
$c = \{(i, j), (i+1, j), \ldots, (i+k, j)\},\ i = k+1, \ldots, n-k,\ \ j = 1, \ldots, m,\ \ k = 1, 2, 3,$
we define its preceding clique $p_k(c)$ as follows:

$$p_k(c) = \{(i - k, j), (i - k + 1, j), \ldots, (i, j)\}.$$

When $i = k + 2, \ldots, n - k,\ \ j = 1, \ldots, m,\ \ k = 1, 2, 3$, a good approximation of $p_k(c)$ used to construct our approximating functions is given by

$$\pi_k(c) = \{(i - k - 1, j), (i - k, j), \ldots, (i - 1, j)\}.$$

We define $\pi_k(c)$ in such a way that $c \cap \pi_k(c) = \emptyset$. This will be useful to find the family of approximations of the energy function in Subsection 3.2.

If c is a horizontal clique, $c = \{(i, j), (i, j+1), \dots, (i, j + k)\}$, $i = 1, \dots, n$, $j = k+1, \dots, m-k$, $k = 1, 2, 3$, then its preceding clique $p_k(c)$ is defined by

$$p_k(c) = \{(i, j-k), (i, j-k+1), \dots, (i, j)\}.$$

When $i = 1, \dots, n$, $j = k+2, \dots, m-k$, $k = 1, 2, 3$, a good approximation of $p_k(c)$ is

$$\pi_k(c) = \{(i, j-k-1), (i, j-k), \dots, (i, j-1)\}.$$

For mixed cliques and cliques on the board of the image, $p_k(c)$ and $\pi_k(c)$ are considered not to be defined.

Appendix B: Duality Conditions on the Stabilizer

In order for a stabilizer φ to be edge-preserving and that the non-parallelism constraint on the implicit line process is satisfied, we require that the hypotheses of the following theorem are satisfied (see [14]):

Theorem 1 *For every $p \in [1, 2]$, let*

$$\varphi^{(p)}(t_1, t_2) = (2-p)\,\overline{\varphi}(t_1, t_2) + (p-1)\,\varphi(t_1, t_2), \quad t_1 \in \mathbb{R}, t_2 \in [-\bar{t}, \bar{t}], \tag{B6}$$

where $\bar{t} = 2^k \cdot \sqrt{2} \cdot 255$, $k = 1, 2, 3$, for light intensity of the images in the range $[0, 255]$, is the maximum value which the variable t_2 can assume, $\overline{\varphi}$ and φ are as in (14) and (7), respectively. Then, $\varphi^{(p)}$ satisfies the following conditions:

H1) *for every $t_2 \in [-\bar{t}, \bar{t}]$, the function $\varphi_{t_2} : \mathbb{R} \to \mathbb{R} \cup \{-\infty\}$ defined by $\varphi_{t_2}(t_1) = \varphi(t_1, t_2)$ is upper semicontinuous and even on $\mathbb{R}$, and $\varphi_{t_2}(0) \in \mathbb{R}$;*

H2) *for each $t_2 \in [-\bar{t}, \bar{t}]$, the function $\psi_{t_2} : \mathbb{R} \to \mathbb{R} \cup \{-\infty\}$ defined by*

$$\psi_{t_2}(t_1) = \begin{cases} \varphi(\sqrt{t_1}, t_2), & \text{if } t_1 \ge 0, \\ -\infty, & \text{if } t_1 < 0, \end{cases}$$

is concave on $\mathbb{R}_0^+$;

H3) *φ_{t_2} is non-decreasing on $\mathbb{R}_0^+$ for every $t_2 \in [-\bar{t}, \bar{t}]$;*

H4) $\displaystyle\lim_{t_1 \to +\infty} \frac{\psi_{t_2}(t_1)}{t_1} = 0$ *for each $t_2 \in [-\bar{t}, \bar{t}]$,*

H5) *there exists at least a real number t_1 such that the function $\varphi_{t_1}(t_2) = \varphi(t_1, t_2)$ is not constant on $[-\bar{t}, \bar{t}]$, and φ_{t_1} is even on $[-\bar{t}, \bar{t}]$ and non-decreasing on $[0, \bar{t}]$ for every $t_1 \in \mathbb{R}_0^+$.*

Proof We begin by proving that the function $\overline{\varphi}$ defined in (14) satisfies Hj), $j = 1, \dots, 4$.

It is readily seen that $\overline{\varphi}$ fulfills H1).

Now we prove H2).

For $i = 1, 2$ and $t_1 \in \mathbb{R}_0^+$, set

$$\overline{f_i}(t_1) = \overline{g_i}(\sqrt{t_1}) = \begin{cases} \lambda^2 t_1, & \text{if } 0 \le t_1 \le q_i^2, \\ \lambda^2 (2 q_i \sqrt{t_1} - q_i^2), & \text{if } t_1 \ge q_i^2. \end{cases} \tag{B7}$$

We have

$$\overline{\varphi}_{t_2}(\sqrt{t_1}) = \overline{\psi}_{t_2}(t_1) = \frac{\bar{t}^2 - t_2^2}{\bar{t}^2}\,\overline{f_1}(t_1) + \frac{t_2^2}{\bar{t}^2}\,\overline{f_2}(t_1),$$

and hence

$$\overline{f_i}'(t_1) = \begin{cases} \lambda^2, & \text{if } 0 \le t_1 \le q_i^2, \\ \lambda^2 q_i\, t_1^{-1/2}, & \text{if } t_1 \ge q_i^2; \end{cases}$$

$$\overline{f_i}''(t_1) = \begin{cases} 0, & \text{if } 0 \le t_1 < q_i^2, \\ -\dfrac{1}{2}\lambda^2 q_i\, t_1^{-3/2}, & \text{if } t_1 > q_i^2. \end{cases}$$

Let $\gamma_{t_2} = \dfrac{\bar{t}^2 - t_2^2}{\bar{t}^2}$. Then, $1 - \gamma_{t_2} = \dfrac{t_2^2}{\bar{t}^2}$.

It is not difficult to check that $0 \le \gamma_{t_2} \le 1$, since $|t_2| \le \bar{t}$. Thus, for every $t_1 \in \mathbb{R}_0^+$ and $t_1 \ne q_1^2$, $t_1 \ne q_2^2$, $t_2 \in [-\bar{t}, \bar{t}]$, we have

$$\overline{\psi}_{t_2}'(t_1) = \gamma_{t_2}\,\overline{f_1}'(t_1) + (1 - \gamma_{t_2})\,\overline{f_2}'(t_1) \ge 0. \tag{B8}$$

$$\overline{\psi}_{t_2}''(t_1) = \gamma_{t_2}\,\overline{f_1}''(t_1) + (1 - \gamma_{t_2})\,\overline{f_2}''(t_1) \le 0.$$

Observe that the inequality in (B8) will be useful to prove H3). Since $\overline{\psi}_{t_2}$ is of class C^1 on its domain (indeed, it is a composition of C^1 functions), then it is concave on $\mathbb{R}_0^+$ for all $t_2 \in [-\bar{t}, \bar{t}]$. So, $\overline{\varphi}$ satisfies condition H2).

Now we prove H3). From (B8), it follows that $\overline{\psi}_{t_2}$ is non-decreasing on $\mathbb{R}_0^+$, and hence so is $\overline{\varphi}_{t_2}$.

Thus we get

$$\overline{\varphi}_{t_2}'(t_1) = \gamma_{t_2}\,\overline{g_1}'(t_1) + (1 - \gamma_{t_2})\,\overline{g_2}'(t_1) \ge 0$$

for each $t_1 \in \mathbb{R}_0^+$. Thus, H3) holds.

Now we show that $\overline{\varphi}$ fulfils H4). Indeed, we have

$$\lim_{t_1 \to +\infty} \frac{\overline{f_i}(t_1)}{t_1} = \lim_{t_1 \to +\infty} \frac{\lambda^2 (2 q_i \sqrt{t_1} - q_i^2)}{t_1} = 0 \quad (i = 1, 2),$$

and hence

$$\lim_{t_1 \to +\infty} \frac{\overline{\psi}_{t_2}(t_1)}{t_1} = 0 \quad \text{for every } t_2 \in [-\bar{t}, \bar{t}].$$

Finally, we prove that $\overline{\varphi}$ satisfies H5).

Take $t_1 = q_2$. For each $t_2 \in [-\overline{t}, \overline{t}]$, it is

$$\overline{\varphi}(q_2, t_2) = \frac{\overline{t}^2 - t_2^2}{\overline{t}^2}\, \lambda^2\, (2\, q_1\, q_2 - q_1^2) + \frac{t_2^2}{\overline{t}^2}\, \lambda^2\, q_2^2,$$

and hence $\overline{\varphi}(q_2, 0) = \lambda^2\, (2\, q_1\, q_2 - q_1^2)$, $\overline{\varphi}(q_2, \overline{t}) = \lambda^2\, q_2^2$. We claim that $\overline{\varphi}(q_2, 0) \neq \overline{\varphi}(q_2, \overline{t})$. If not, then we would have $2\, q_1\, q_2 - q_1^2 = q_2^2$, and hence $0 = q_2^2 - 2\, q_1\, q_2 + q_1^2 = (q_1 - q_2)^2$, that is $q_1 = q_2$, which is absurd since we know that $0 < q_1 < q_2$. Therefore, the function $t_2 \mapsto \overline{\varphi}(q_2, t_2)$ is not constant, and hence, the first property of H5) is satisfied.

Moreover, it is easy to see that $\overline{\varphi}_{t_1}$ is even on $\mathbb{R}$ for each $t_1 \in \mathbb{R}$.

From (13), it is not difficult to deduce that $\overline{g_2}(t_1) - \overline{g_1}(t_1) \geq 0$ for all $t_1 \geq 0$.

We get

$$\begin{aligned}\frac{d\overline{\varphi}_{t_1}}{dt_2}(t_2) &= -\frac{2\, t_2}{\overline{t}^2}\, \overline{g_1}(t_1) + \frac{2\, t_2}{\overline{t}^2}\, \overline{g_2}(t_1) \\ &= \frac{2\, t_2}{\overline{t}^2}\, (\overline{g_2}(t_1) - \overline{g_1}(t_1)) \geq 0 \qquad \text{(B9)}\end{aligned}$$

for any $t_1 \in \mathbb{R}_0^+$. Hence, the function $\overline{\varphi}_{t_1}$ is non-decreasing on $\mathbb{R}_0^+$ for every $t_1 \in \mathbb{R}_0^+$. Thus, H5) is proved.

Now we prove that for $i = 1, 2$, the function φ defined in (7) satisfies conditions Hj), $j = 1, \ldots, 4$.

It is easy to see that, by construction, H1) holds.

We now prove H2). We begin with the case when $|t_2| \leq s$ or $t_2 \geq \zeta$ in (7). For $i = 1, 2$, set

$$f_i(t_1) = g_i(\sqrt{t_1}) = \begin{cases} \lambda^2 t_1, & \text{if } 0 \leq t_1 \leq q_i^2, \\ \alpha_i - \dfrac{\tau}{2}(\sqrt{t_1} - r_i)^2, & \text{if } q_i^2 \leq t_1 \leq r_i^2, \\ \alpha_i, & \text{if } t_1 \geq r_i^2. \end{cases} \qquad \text{(B10)}$$

We have

$$\varphi(\sqrt{t_1}, t_2) = \begin{cases} f_1(t_1), & \text{if } |t_2| \leq s, \\ \left(1 - \dfrac{2(|t_2| - s)^2}{(\zeta - s)^2}\right) f_1(t_1) + \frac{2(|t_2|-s)^2}{(\zeta-s)^2}\, f_2(t_1), & \text{if } s < |t_2| \leq \frac{\zeta+s}{2}, \\ \frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\, f_1(t_1) + \left(1 - \frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\right) f_2(t_1), & \text{if } \frac{\zeta+s}{2} < |t_2| < \zeta, \\ f_2(t_1), & \text{if } |t_2| \geq \zeta. \end{cases} \qquad \text{(B11)}$$

We claim that f_i is non-decreasing and concave on $\mathbb{R}_0^+$. We get

$$f_i{}'(t_1) = \begin{cases} \lambda^2, & \text{if } 0 \leq t_1 \leq q_i^2, \\ \alpha_i - \dfrac{\tau}{2}\left(1 - \dfrac{r_i}{\sqrt{t_1}}\right), & \text{if } q_i^2 \leq t_1 \leq r_i^2, \\ 0, & \text{if } t_1 \geq r_i^2. \end{cases} \qquad \text{(B12)}$$

Note that f_i is C^1, since it is a composition of functions of class C^1. Moreover, we have

$$f_i{}''(t_1) = \begin{cases} 0, & \text{if } 0 \leq t_1 < q_i^2, \\ -\dfrac{\tau\, r_i}{4\sqrt{t_1^3}}, & \text{if } q_i^2 < t_1 < r_i^2, \\ 0, & \text{if } t_1 > r_i^2. \end{cases} \qquad \text{(B13)}$$

From this, we deduce that φ fulfils H2), at least when $|t_2| \leq s$ or $|t_2| \geq \zeta$.

Now, we examine the case

$$s < |t_2| \leq \frac{\zeta + s}{2}. \qquad \text{(B14)}$$

We have

$$\psi_{t_2}'(t_1) = \left(1 - \frac{2(|t_2| - s)^2}{(\zeta - s)^2}\right) f_1'(t_1) + \frac{2(|t_2| - s)^2}{(\zeta - s)^2}\, f_2'(t_1), \qquad \text{(B15)}$$

$$\psi_{t_2}''(t_1) = \left(1 - \frac{2(|t_2| - s)^2}{(\zeta - s)^2}\right) f_1''(t_1) + \frac{2(|t_2| - s)^2}{(\zeta - s)^2}\, f_2''(t_1). \qquad \text{(B16)}$$

Observe that $\dfrac{2(|t_2| - s)^2}{(\zeta - s)^2} \geq 0$. Now, we claim that $\dfrac{2(|t_2| - s)^2}{(\zeta - s)^2} \leq 1$. Indeed, since $s < |t_2| \leq \dfrac{\zeta + s}{2}$, then $0 < |t_2| - s \leq \dfrac{\zeta - s}{2} < \dfrac{\zeta - s}{\sqrt{2}}$, and hence $(|t_2| - s)^2 \leq \dfrac{(\zeta - s)^2}{2}$, getting the claim. Therefore, $1 - \dfrac{2(|t_2| - s)^2}{(\zeta - s)^2} \geq 0$. From this, since $f_i'(t_1) \geq 0$ for every $t_1 \geq 0$ and $f_i''(t_1) \leq 0$ for each $t_1 \mathbb{R}_0^+$, $t_1 \neq q_1$, $t_1 \neq q_2$, in the case (B14) we obtain

$$\psi_{t_2}'(t_1) \geq 0 \text{ for every } t_1 \in \mathbb{R}_0^+, \qquad \text{(B17)}$$

$$\psi_{t_2}''(t_1) \leq 0 \text{ for any } t_1 \in \mathbb{R}_0^+,\ t_1 \neq q_1,\ t_1 \neq q_2. \qquad \text{(B18)}$$

Note that the inequality in (B17) will be useful to prove H3).

From (B18), taking into account the continuity of ψ_{t_2}, we deduce that φ satisfies H2) also in the case (B14).

Now, we consider the case

$$\frac{\zeta+s}{2}<|t_2|<\zeta. \tag{B19}$$

We get

$$\psi_{t_2}'(t_1)=\frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\,f_1'(t_1)+\left(1-\frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\right)f_2'(t_1), \tag{B20}$$

$$\psi_{t_2}''(t_1)=\frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\,f_1''(t_1)+\left(1-\frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\right)f_2''(t_1). \tag{B21}$$

Note that $\frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\geq 0$. Now we claim that $\frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\leq 1$. Indeed, as $\frac{\zeta+s}{2}<|t_2|<\zeta$, then $0<\zeta-|t_2|\leq\frac{\zeta-s}{2}<\frac{\zeta-s}{\sqrt{2}}$, and so $(|t_2|-\zeta)^2\leq\frac{(\zeta-s)^2}{2}$, getting the claim. Thus, $1-\frac{2(|t_2|-\zeta)^2}{(\zeta-s)^2}\geq 0$. From this, in the case (B19), analogously as in the case (B14), we obtain

$$\psi_{t_2}'(t_1)\geq 0 \text{ for every } t_1\in\mathbb{R}_0^+, \tag{B22}$$

$$\psi_{t_2}''(t_1)\leq 0 \text{ for any } t_1\in\mathbb{R}_0^+,\ t_1\neq q_1,\ t_1\neq q_2. \tag{B23}$$

From (B23) and thanks to the continuity of ψ_{t_2}, it follows that φ satisfies H2) also in the case (B19).

Now we prove that φ satisfies H3).

First, when $|t_2|\leq s$ or $|t_2|\geq\zeta$, observe that it is readily seen that g_i is non-decreasing on $\mathbb{R}_0^+$ for $i=1,2$. Hence, ψ_{t_2} is non-decreasing on $\mathbb{R}_0^+$, and thus H3) holds.

Moreover, when t_2 satisfies the case (B14) or the case (B19), from (B17) and (B22) it follows that ψ_{t_2} is non-decreasing on $\mathbb{R}_0^+$, and hence φ_{t_2} is too. Thus, φ fulfils H3).

Now we prove H4). That φ satisfies H4) is a consequence of the fact that $\lim_{t_1\to+\infty}\frac{f_i(t_1)}{t_1}=0$.

Now we prove H5). Let

$$a(t_1)=2\,\frac{g_2(t_1)-g_1(t_1)}{(\zeta-s)^2},\quad t_1\in\mathbb{R}.$$

First of all, observe that $\varphi(r_2,s)=g_1(r_2)=\alpha$, $\varphi(r_2,\zeta)=g_2(r_2)=\alpha+\varepsilon\neq\varphi(r_2,s)$. Thus, the function $t_2\mapsto\varphi(r_2,t_2)$ is not constant. Moreover, it is easy to see that the function $\varphi_{t_1}(t_2)$ is even on $[-\bar{t},\bar{t}]$ for every $t_1\in\mathbb{R}_0^+$, since it depends on $|t_2|$. Furthermore, it is not difficult to check that

$$g_2(t_1)\geq g_1(t_1)\quad\text{for every } t_1\in\mathbb{R}_0^+. \tag{B24}$$

Let $t_2\in[0,\bar{t}]$. We get

$${\varphi_{t_1}}'(t_2)=-\frac{2(t_2-s)^2}{(\zeta-s)^2}\,g_1(t_1)+\frac{2(t_2-s)^2}{(\zeta-s)^2}\,g_2(t_1) \tag{B25}$$

in the case (B14), and

$${\varphi_{t_1}}'(t_2)=-\frac{2(t_2-\zeta)^2}{(\zeta-s)^2}\,g_1(t_1)-\frac{2(t_2-\zeta)^2}{(\zeta-s)^2}\,g_2(t_1) \tag{B26}$$

in the case (B19). From (B24), (B25) and (B26) it follows that $\varphi_{t_1}'(t_2)\geq 0$ for each $t_2\in[0,\bar{t}]$. By arbitrariness of $t_1\in\mathbb{R}_0^+$, we deduce that φ satisfies H5).

Now, we observe that the functions $\varphi^{(p)}$, $p\in[0,2]$, satisfy conditions Hj), $j=1,\ldots,4$, since they are non-negative linear combinations of functions satisfying Hj), $j=1,\ldots,4$. Since $\overline{\varphi}_{t_1}$ and φ_{t_1} are non-decreasing for each $t_1\in\mathbb{R}_0^+$, $\overline{\varphi}_{t_2}$ and φ_{t_2} are non-decreasing for every $t_2\in[-\bar{t},\bar{t}]$, and the functions $t_2\mapsto\overline{\varphi}(q_2,t_2)$, $t_2\mapsto\varphi(r_2,t_2)$ are not constant, it follows that for every $p\in[0,2]$ there exists at least a $t_1\in\mathbb{R}_0^+$ such that the function $t_2\mapsto\varphi^{(p)}(t_1,t_2)$ is not constant. The other properties of H5) hold because the $\varphi^{(p)}$'s are non-negative linear combinations of functions satisfying H5). □

Appendix C: Convergence of the NL-SOR Algorithm

To minimize each approximation $E^{(p)}$, $p\in[0,2]$, we use the NL-SOR algorithm, described in Subsection 4.1.

We will prove the existence of suitable limit points, which are stationary points of $E^{(0)}$ (in general, they are not minimum points of $E^{(0)}$). In [16, Theorem 2] the convergence of the algorithm is proved when $E^{(0)}$ is strictly convex and of class C^2. Such assumptions are too strong for the componentwise convex approximation of the regularization term in our setting because we deal with functions of class C^1. So, we give an extension of the theorem under these weaker hypotheses.

First, we state the following technical lemma, whose proof is given in [28, Lemma 2.7.1].

Lemma 2 *Let $\phi:[x_0,\overline{x}]\to\mathbb{R}$ be convex, of class C^1, having both left and right second derivative on $[x_0,\overline{x}]$. Suppose that ϕ is second differentiable on $[x_0,\overline{x}]\setminus P$, where $P=\{x_j : j=1,\ldots,N\}$, with $x_0<x_1<\ldots<x_N<\overline{x}$.*

Then, for every $x\in[x_0,\overline{x}]$ there exist $\xi\in]x_0,x[$ and $\mu\geq 0$, such that

$$\mu\in I_\xi=[\min\{\phi_-''(\xi),\phi_+''(\xi)\},\max\{\phi_-''(\xi),\phi_+''(\xi)\}] \tag{C27}$$

and

$$\phi(x) = \phi(x_0) + (x - x_0)\phi'(x_0) + \frac{(x - x_0)^2}{2}\,\mu. \tag{C28}$$

Observe that when $p = 0$, the NL-SOR can be formulated as in Algorithm 3. At the iterate $l \in \mathbb{N}$, fixed $i \in 1, 2, \ldots, 3\,m$ and $e \in \{r, g, b\}$, the vector $\mathbf{x}^{(l,i,e)}$ is defined by

$$(x^{(l,i,e)})^{(q)}_j = \begin{cases} (x^{\text{prec}(l,i,e)})^{(q)}_j & \text{if } i \neq j \text{ or } q \neq e, \\ (x^{\text{prec}(l,i,e)})^{(q)}_j - \dfrac{\omega}{T}\,\dfrac{\partial E^{(0)}\left(\mathbf{x}^{\text{prec}(l,i,e)}\right)}{\partial x^{(e)}_i} & \text{if } i = j \text{ and } q = e, \end{cases} \tag{C29}$$

where

$$\mathbf{x}^{\text{prec}(l,i,e)} = \begin{cases} \mathbf{x}^{(l,i,e-1)} & \text{if } e \neq r; \\ \mathbf{x}^{(l,i-1,b)} & \text{if } i \neq 1, e = r; \\ \mathbf{x}^{(l-1,nm,b)} & \text{if } i = 1, e = r. \end{cases} \tag{C30}$$

Algorithm 3 NL-SOR algorithm

```
Input: An energy function $E^{(0)}$;
Output: An estimation of the minimizer $\mathbf{x}^{(l,i,e)}$;
given the initial vector $\mathbf{x}^{(0,mn,b)}$
for $l = 1, 2, \ldots$ do
    for $i = 1, 2, \ldots, nm$ do
        for $e = r, g, b$ do
            set the vector $\mathbf{x}^{(l,i,e)} \in \mathbb{R}^{3mn}$ as in Eq. (C29);
        end for
    end for
end for
```

Algorithm 3 allows to denote the image vectors actually defined at each iterate l, at every pixel i and at each color e. We observe that the algorithm here proposed is a particular case of that given in Subsection 4.1, and has been suitably modified in order to give a rigorous definition of the image vector $\mathbf{x}$ at every iterate l, at every pixel i and at each color e.

Moreover, fixed the step (l, i, e), let $\mathbf{x}^{(l,i,e)} \backslash (x^{(l,i,e)})^{(e)}_i \in \mathbb{R}^{3nm-1}$ be the vector whose elements are those of $\mathbf{x}^{(l,i,e)}$ except $(x^{(l,i,e)})^{(e)}_i$. The value of this pixel $(x^{(l,i,e)})^{(e)}_i$ is an unknown variable, which we call z. For each fixed value of $\mathbf{x}^{(l,i,e)} \backslash (x^{(l,i,e)})^{(e)}_i$, let us define the following energy function $\overline{E}^{(l,i,e)} : \mathbb{R} \to \mathbb{R}$ by

$$\overline{E}^{(l,i,e)}(z) = E^{(0)}(\mathbf{x}^{(l,i,e)} \setminus (x^{(l,i,e)})^{(e)}_i, z). \tag{C31}$$

Theorem 3 *Let $E^{(0)} : \mathbb{R}^{3nm} \to \mathbb{R}$ be a function of class C^1 and coercive, that is*

$$\lim_{\|x\| \to +\infty} E^{(0)}(x) = +\infty\,; \tag{C32}$$

fix $\mathbf{x}^{(0,nm,b)} \in \mathbb{R}^{3nm}$, and let $\{\mathbf{x}^{(l,i,e)}\}$, $l \in \mathbb{N}, i = 1, 2, \ldots, 3\,m, e \in \{r, g, b\}$, be the sequence defined iteratively in (C29), where $0 < \omega < 2$ and

$$T > \max_{i=1,2,\ldots,nm, e=r,g,b} \; \max_{\mathbf{x}} \left\{ \frac{\partial^2_+ E^{(0)}(\mathbf{x})}{(\partial x^{(e)}_i)^2}, \frac{\partial^2_- E^{(0)}(\mathbf{x})}{(\partial x^{(e)}_i)^2} \right\}. \tag{C33}$$

Let $\overline{E}^{(l,i,e)} : \mathbb{R} \to \mathbb{R}$ be the function defined in (C31). Assume that $\overline{E}^{(l,i,e)}$ is convex, admits both left and right derivative on $\mathbb{R}$ and is not secondly differentiable (at most) at a finite number of points.

Then, $\lim\limits_{(l,i,e)} \nabla E^{(0)}(x^{(l,i,e)}) = 0.$

Proof We begin by proving that, during the updating of $(x^{(l,i,e)})^{(e)}_i$, the function $E^{(0)}$ is non-increasing.

If

$$\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})^{(e)}_i) = 0, \tag{C34}$$

then, since in (C29) it is $(x^{\text{prec}(l,i,e)})^{(e)}_i = \dfrac{\partial E^{(0)}\left(x^{\text{prec}(l,i,e)}\right)}{\partial x^{(e)}_i}$, we get

$$(\mathbf{x}^{(l,i,e)})^{(e)}_i = (x^{\text{prec}(l,i,e)})^{(e)}_i \tag{C35}$$

and hence $\mathbf{x}^{(l,i,e)} = \mathbf{x}^{\text{prec}(l,i,e)}$, Moreover, the value of the energy function does not change.

Now, we treat the case when

$$(x^{\text{prec}(l,i,e)})^{(e)}_i \neq 0. \tag{C36}$$

Note that, by (C32), we get

$$\lim_{z \to +\infty} \overline{E}^{(l,i,e)}(z) = \lim_{z \to -\infty} \overline{E}^{(l,i,e)}(z) = +\infty, \tag{C37}$$

that is the function $\overline{E}^{(l,i,e)}$ is coercive on $\mathbb{R}$. Since $\overline{E}^{(l,i,e)}$ is also continuous, then, by [5, Theorem 2.32], $\overline{E}^{(l,i,e)}$ assumes the minimum value, say $(t_*)^{(l,i,e)}$.

We get that, for any $t > (t_*)^{(l,i,e)}$, the level set $L_t = \{z \in \mathbb{R} : \overline{E}^{(l,i,e)}(z) = t\}$ has exactly two points, and $\overline{E}^{(l,i,e)}(z) < t$ whenever z is in the interior of the interval whose endpoints are the elements of L_t.

Now we claim that, for every $t > (t_*)^{(l,i,e)}$, the level set $L_t = \{z \in \mathbb{R} : \overline{E}^{(l,i,e)}(z) = t^{(l,i,e)}\}$ has exactly two points.

Since $\overline{E}^{(l,i,e)}$ is convex and differentiable, we get that $\overline{E}^{(l,i,e)}(z) = (t_*)^{(l,i,e)}$ if and only if $\overline{E}^{(l,i,e)\prime}(z) = 0$. From the continuity of $\overline{E}^{(l,i,e)}$ and (C32) it follows that $\overline{E}^{(l,i,e)}$ assumes all values $t \in [(t_*)^{(l,i,e)}, +\infty[$. Since $\overline{E}^{(l,i,e)\prime}$ is non-decreasing, then $\overline{E}^{(l,i,e)\prime}$ is positive (resp. negative), and hence $\overline{E}^{(l,i,e)}$ is strictly increasing (resp. decreasing) at all greater (resp. smaller) points than the minimum points of $\overline{E}^{(l,i,e)}$. Thus, $\overline{E}^{(l,i,e)}$ assumes each value $t > (t_*)^{(l,i,e)}$ exactly two times, getting the claim.

Set $t^{(l,i,e)} = \overline{E}^{(l,i,e)}((x^{\text{prec}(l,i,e)})_i^{(e)})$, and $L_{t^{(l,i,e)}} = \{(x^{\text{prec}(l,i,e)})_i^{(e)}, \overline{z}^{(l,i,e)}\}$, where $\overline{E}^{(l,i,e)}(\overline{z}^{(l,i,e)}) = t^{(l,i,e)}$.

Without loss of generality, let us consider the case $\overline{z}^{(l,i,e)} < (x^{\text{prec}(l,i,e)})_i^{(e)}$. Note that, in this case, $\overline{E}^{(l,i,e)\prime}(\overline{z}^{(l,i,e)}) < 0$, while $\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}) > 0$.

By hypothesis, taking into account that $\overline{E}^{(l,i,e)}$ is of class C^1, from Lemma 2 applied to the interval $[\overline{z}^{(l,i,e)}, (x^{\text{prec}(l,i,e)})_i^{(e)}]$ we find $\xi \in]\overline{z}^{(l,i,e)}, (x^{\text{prec}(l,i,e)})_i^{(e)}[$ and $\mu \geq 0$, such that

$$\begin{aligned} &\min\{\overline{E}^{(l,i,e)\prime\prime}{}_-(\xi), \overline{E}^{(l,i,e)\prime\prime}{}_+(\xi)\} \\ &\quad \leq \mu \leq \max\{\overline{E}^{(l,i,e)\prime\prime}{}_-(\xi), \overline{E}^{(l,i,e)\prime\prime}{}_+(\xi)\} \end{aligned} \tag{C38}$$

and

$$\begin{aligned} \overline{E}^{(l,i,e)}(\overline{z}^{(l,i,e)}) = \overline{E}^{(l,i,e)}&((x^{\text{prec}(l,i,e)})_i^{(e)}) + \\ &+\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}) \\ &\times(\overline{z}^{(l,i,e)} - (x^{\text{prec}(l,i,e)})_i^{(e)}) \\ &+\frac{1}{2}\,\mu\,(\overline{z}^{(l,i,e)} - (x^{\text{prec}(l,i,e)})_i^{(e)})^2. \end{aligned}$$

Since $\overline{E}^{(l,i,e)}((x^{\text{prec}(l,i,e)})_i^{(e)}) = \overline{E}^{(l,i,e)}(\overline{z}^{(l,i,e)})$, then we have

$$\begin{aligned} &\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)})\,(\overline{z}^{(l,i,e)} - (x^{\text{prec}(l,i,e)})_i^{(e)}) + \\ &+\frac{1}{2}\,\mu\,(\overline{z}^{(l,i,e)} - (x^{\text{prec}(l,i,e)})_i^{(e)})^2 = 0. \end{aligned} \tag{C39}$$

Note that from (C38) and (C33) we get

$$\mu \leq T. \tag{C40}$$

Now we claim that $\mu > 0$. Indeed, if $\mu = 0$, then from (C39) we get

$$\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)})\,(\overline{z}^{(l,i,e)} - (x^{\text{prec}(l,i,e)})_i^{(e)}) = 0,$$

and hence $\overline{z}^{(l,i,e)} = (x^{\text{prec}(l,i,e)})_i^{(e)}$, because $\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}) > 0$. This is absurd because $\overline{z}^{(l,i,e)} < (x^{\text{prec}(l,i,e)})_i^{(e)}$. Therefore, we get the claim. From (C39), we obtain

$$(x^{\text{prec}(l,i,e)})_i^{(e)} - \overline{z}^{(l,i,e)} = \frac{2}{\mu}\,\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}). \tag{C41}$$

We recall that, by (C29), it is

$$(x^{(l,i,e)})_i^{(e)} = (x^{\text{prec}(l,i,e)})_i^{(e)} - \frac{\omega}{T}\,\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}). \tag{C42}$$

Since $0 < \omega < 2$, from (C40), (C41) and (C42) we have

$$\begin{aligned} &0 < (x^{(l,i,e)})_i^{(e)} - x^{\text{prec}(l,i,e)})_i^{(e)} = (x^{(l,i,e)})_i^{(e)} = \\ &\quad = \frac{\omega}{T}\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}) < \frac{2}{T}\overline{E}^{(l,i,e)\prime}(x^{\text{prec}(l,i,e))}{}_i^{(e)}) \leq \\ &\quad \leq \frac{2}{\mu}\overline{E}^{(l,i,e)\prime}(x^{\text{prec}(l,i,e)})_i^{(e)} = (x^{\text{prec}(l,i,e)})_i^{(e)} - \overline{z}^{(l,i,e)}. \end{aligned} \tag{C43}$$

From (C43) it follows that

$$\begin{aligned} &\overline{z}^{(l,i,e)} < (x^{(l,i,e)})_i^{(e)} < (x^{\text{prec}(l,i,e)})_i^{(e)} \quad \text{when } \overline{z}^{(l,i,e)} \\ &\quad < (x^{\text{prec}(l,i,e)})_i^{(e)}. \end{aligned} \tag{C44}$$

Analogously, it is possible to prove that

$$\begin{aligned} &\overline{z}^{(l,i,e)} > (x^{(l,i,e)})_i^{(e)} > (x^{\text{prec}(l,i,e)})_i^{(e)} \quad \text{when } \overline{z}^{(l,i,e)} \\ &\quad > (x^{\text{prec}(l,i,e)})_i^{(e)}. \end{aligned} \tag{C45}$$

Thus, in the case (C36), we get $\overline{E}^{(l,i,e)}((x^{(l,i,e)})_i^{(e)}) < \overline{E}^{(l,i,e)}((x^{\text{prec}(l,i,e)})_i^{(e)})$. Therefore, in both cases (C34) and (C36), the sequence $\{\overline{E}^{(l,i,e)}((x^{(l,i,e)})_i^{(e)})\}$, $l \in \mathbb{N}$, $i = 1, 2, \ldots, nm$, $e \in \{r, g, b\}$, is non-increasing. Since $E^{(0)}$ is bounded from below, then the sequence $\{\overline{E}^{(l,i,e)}((x^{(l,i,e)})_i^{(e)})\}$, $l \in \mathbb{N}$, $i = 1, 2, \ldots, nm$, $e \in \{r, g, b\}$, is non-increasing, and hence it is convergent.

Now, we claim that the sequence $\{(x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)}\}$, $l \in \mathbb{N}$, $i = 1, 2, \ldots, nm$, $e \in \{r, g, b\}$, converges to 0.

Fix $l \in \mathbb{N}$, $i = 1, 2, \ldots, nm$, $e \in \{r, g, b\}$. If $\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}) = 0$, then, as seen in (C35), we get $(x^{(l,i,e)})_i^{(e)} = (x^{\text{prec}(l,i,e)})_i^{(e)}$.

Now, we consider the case when $\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}) \neq 0$. By an argument similar to that used in the proof of [16, Theorem 2], from Lemma 2 applied to the interval whose endpoints are $(x^{(l,i,e)})_i^{(e)}$ and $(x^{\text{prec}(l,i,e)})_i^{(e)}$, we find a non-negative real number μ with

$$\begin{aligned}&\overline{E}^{(l,i,e)}((x^{\text{prec}(l,i,e)})_i^{(e)}) - \overline{E}^{(l,i,e)}((x^{(l,i,e)})_i^{(e)})\\ &\quad = \overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)})((x^{\text{prec}(l,i,e)})_i^{(e)} - (x^{(l,i,e)})_i^{(e)})\\ &\quad - \frac{\mu}{2}((x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)})^2.\end{aligned} \tag{C46}$$

From (C42) we get

$$\begin{aligned}&(x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)} =\\ &\quad = ((x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)})^2\\ &\quad \times \frac{T}{\omega\, \overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)})}.\end{aligned} \tag{C47}$$

From (C46) and (C47) we obtain

$$\begin{aligned}&0 < \overline{E}^{(l,i,e)}((x^{\text{prec}(l,i,e)})_i^{(e)}) - \overline{E}^{(l,i,e)}((x^{(l,i,e)})_i^{(e)}) =\\ &\quad = \frac{T}{\omega}((x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)})^2\\ &\quad -\frac{\mu}{2}((x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)})^2 =\\ &\quad = \left(\frac{T}{\omega} - \frac{\mu}{2}\right)((x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)})^2.\end{aligned} \tag{C48}$$

As $0 < \omega < 2$ and $0 \le \mu < T$, we get

$$\frac{T}{\omega} - \frac{\mu}{2} \ge T\left(\frac{1}{\omega} - \frac{1}{2}\right) > 0. \tag{C49}$$

From (C48) and (C49) we obtain

$$\begin{aligned}&0 \le ((x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)})^2 \le\\ &\quad \le \frac{2\,\omega}{T\,(2-\omega)}(\overline{E}^{(l,i,e)}((x^{\text{prec}(l,i,e)})_i^{(e)})\\ &\quad -\overline{E}^{(l,i,e)}((x^{(l,i,e)})_i^{(e)})).\end{aligned} \tag{C50}$$

Note that (C50) holds also when $\overline{E}^{(l,i,e)\prime}((x^{\text{prec}(l,i,e)})_i^{(e)}) = 0$. Thus, in both cases (C34) and (C36), from (C50) and the convergence of the sequence $\{\overline{E}^{(l,i,e)}((x^{(l,i,e)})_i^{(e)})\}$, $l \in \mathbb{N}$, $i = 1, 2, \ldots, nm$, $e \in \{r, g, b\}$, it follows that the sequence $\{(\overline{E}^{(l,i,e)}((x^{\text{prec}(l,i,e)})_i^{(e)}) - \overline{E}^{(l,i,e)}((x^{(l,i,e)})_i^{(e)}))\}$, $l \in \mathbb{N}$, $i = 1, 2, \ldots, nm$, $e \in \{r, g, b\}$, converges to 0. From this, it follows that

$$\lim_{(l,i,e)} ((x^{(l,i,e)})_i^{(e)} - (x^{\text{prec}(l,i,e)})_i^{(e)}) = 0, \tag{C51}$$

getting the claim.

From (C29) we deduce

$$(x^{(l,i,e)})_j^{(q)} - (x^{\text{prec}(l,i,e)})_j^{(q)} = -\frac{\omega}{T}\,\frac{\partial E^{(0)}\left(\mathbf{x}^{\text{prec}(l,i,e)}\right)}{\partial x_i^{(e)}}. \tag{C52}$$

By arbitrariness of $i \in \{1, 2, \ldots, nm\}$ and $e \in \{r, g, b\}$, from (C51) and (C52) we deduce that $\lim_{(l,i,e)} \nabla E^{(0)}(x^{(l,i,e)}) = 0$, that is the assertion. □

Appendix D: Componentwise Convexity of the First Approximation

Now, we prove that the first approximation is componentwise convex.

Theorem 4 *When $p = 0$, the function $E^{(p)}$ in (15) is componentwise convex.*

Proof We recall that $\overline{\varphi}$ is componentwise convex on $\mathbb{R}^2$ with respect to t_1 and t_2. Fix $k \in \{1, 2, 3\}$ and $c \in C_k$, and choose $\Xi_c^k \in \{N_c^k, V_c^k\}$.

Now we claim that the function $\mathbf{x} \mapsto \overline{\varphi}(\Xi_c^k\mathbf{x}, \Xi_{\pi_k(c)}\mathbf{x})$ is componentwise convex concerning the components of $\mathbf{x} \in \mathbb{R}^{3nm}$. Indeed, fix $x_{i,j}^{(e)}$ with $i \in \{1, 2, \ldots, n\}$, $j \in \{1, 2, \ldots, m\}$ and $e \in \{r, g, b\}$.

Now we prove the convexity of $\overline{\varphi}$ with respect to the variable $x_{i,j}^{(e)}$, in the following three cases:

I) $(i, j) \notin c \cup \pi_k(c)$;
II) $(i, j) \in c$;
III) $(i, j) \in \pi_k(c)$.

We observe that it is impossible that $(i, j) \in c \cap \pi_k(c)$, thanks to our definition of $\pi_k(c)$.

Fix arbitrarily $u, w \in \mathbb{R}$ and $t \in [0, 1]$.

In case I), note that $\Xi_c^k\mathbf{x}$ and $\Xi_{\pi_k(c)}^k\mathbf{x}$ are independent of the value of the variable $x_{i,j}^{(e)}$. So, we get the function $\overline{\varphi}$ evaluated in Ξ_c^k, where all pixels are fixed except $x_{i,j}^{(e)}$.

Fixed an image $\mathbf{x}$, let $\mathbf{x}\backslash x_{i,j}^{(e)} \in \mathbb{R}^{3nm-1}$ be the vector whose elements are those of $\mathbf{x}$ with the exception of $x_{i,j}^{(e)}$. Observe that the value of this pixel $x_{i,j}^{(e)}$ is an unknown variable. We have:

$$\begin{aligned}&\overline{\varphi}(\Xi_c^k(\mathbf{x} \setminus x_{i,j}^{(e)}, x_{i,j}^{(e)}\\ &\quad = tu + (1-t)w),\ \Xi_{\pi_k(c)}^k(\mathbf{x} \setminus x_{i,j}^{(e)}, x_{i,j}^{(e)} = tu + (1-t)w)) =\\ &\quad = t\,\overline{\varphi}(\Xi_c^k(\mathbf{x} \setminus x_{i,j}^{(e)}, x_{i,j}^{(e)} = u),\ \Xi_{\pi_k(c)}^k(\mathbf{x} \setminus x_{i,j}^{(e)}, x_{i,j}^{(e)} = u))\\ &\qquad +(1-t)\overline{\varphi}(\Xi_c^k(\mathbf{x} \setminus x_{i,j}^{(e)}, x_{i,j}^{(e)} = w),\\ &\qquad \times \Xi_{\pi_k(c)}^k(\mathbf{x} \setminus x_{i,j}^{(e)}, x_{i,j}^{(e)} = w)),\end{aligned}$$

since

$$\Xi^k_c(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = a) = \Xi^k_c(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = b) \quad \text{and} \tag{D53}$$

$$\Xi^k_{\pi_k(c)}(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = a) = \Xi^k_{\pi_k(c)}(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = b) \tag{D54}$$

for each $a, b \in \mathbb{R}$.

Now we deal with the case II).

It is not difficult to see that since the finite difference operators D^k_c are linear and the norm $\| \cdot \|_2$ is a convex function, the operators Ξ^k_c and $\Xi_{\pi_k(c)}$ are convex on their domain. We get

$$\begin{aligned}
&\overline{\varphi}(\Xi^k_c(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} \\
&\quad = tu + (1-t)w), \Xi^k_{\pi_k(c)}(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = tu + (1-t)w)) \le \\
&\quad \le \overline{\varphi}(t\, \Xi^k_c(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = u) + (1-t)\Xi^k_c(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = w), \\
&\qquad \times \Xi^k_{\pi_k(c)}(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = \\
&\quad = tu + (1-t)w)) \le \\
&\quad \le t\, \overline{\varphi}(\Xi^k_c(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = u), \Xi^k_{\pi_k(c)}(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = u)) \\
&\qquad + (1-t)\overline{\varphi}(\Xi^k_c(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = w), \\
&\qquad \times \Xi^k_{\pi_k(c)}(\mathbf{x} \setminus x^{(e)}_{i,j}, x^{(e)}_{i,j} = w)).
\end{aligned} \tag{D55}$$

The first inequality in (D55) holds, since Ξ^k_c is (globally) convex. Note that $\overline{\varphi}$ is increasing in the first component.

Furthermore, the third inequality in (D55) follows from (D54), since the function $(t_1, t_2) \mapsto \overline{\varphi}(t_1, t_2)$ is componentwise convex with respect to the variables t_1 and t_2, and since $(i, j) \notin \pi_k(c)$.

Case III) is analogous to the case II). Thus, the assertion follows.

□

Funding Open access funding provided by Universitá degli Studi di Perugia within the CRUI-CARE Agreement.

References

1. Aelterman, J., Goossens, B., De Vylder, J., Pizurica, A., Philips, W.: Computationally Efficient Locally Adaptive Demosaicing of Color Filter Array Images Using the Dual-Tree Complex Wavelet Packet Transform. PLoS ONE **8**(5), 1–18 (2013)
2. Baek, M., Jeong, J.: *Demosaicing algorithm using high-order interpolation with sobel operators*. In: Proceedings of the World Congress on Engineering, WCE 2014, Lecture Notes in Engineering and Computer Science 1, pp. 521-524 (2014)
3. Bai, C., Li, J., Lin, Z., Yu, J., Chen, Y.-W.: Penrose demosaicking. IEEE Trans. Image Process. **24**, 1672–1684 (2015)
4. Bayer, B. E.: Color imaging array. U.S. Patent. 3 971 065 (1976)
5. Beck, A.: Introduction to nonlinear optimization - Theory, Algorithms, and Applications with MATLAB. SIAM Mathematical Optimization, Philadelphia, PA, USA (2014)
6. Bedini, L., Gerace, I., Pepe, M., Salerno, E., Tonazzini, A.: Stochastic and deterministic algorithms for image reconstruction with implicitly referred discontinuities. Internal report n. r/2/85, pages 37, Istituto di Elaborazione della Informazione, C.N.R., Pisa (Italy), (1992)
7. Bedini, L., Gerace, I., Tonazzini, A.: *A GNC algorithm for constrained image reconstruction with continuous-valued line processes.* Pattern Recognit. Lett. 15(9), 907–918, (1994)
8. Blake, A.: Comparison of the efficiency of deterministic and stochastic algorithms for visual reconstruction. IEEE Trans. Pattern Anal. Mach. Intell. **11**(1), 2–12 (1989)
9. Blake, A., Zissermann, A.: Visual Reconstruction. MIT Press, Cambridge, MA (1987)
10. Boccuto, A., Gerace, I.: *Image reconstruction with a non-parallelism constraint.* In: Proceedings of the International Workshop on Computational Intelligence for Multimedia Understanding, Reggio Calabria, Italy, 27-28 October 2016, IEEE Conference Publications, pp. 1–5 (2016)
11. Boccuto, A., Gerace, I., Giorgetti, V.: A Blind Source Separation Technique for Document Restoration. SIAM J. Imaging Sci. **12**(2), 1135–1162 (2019)
12. Boccuto, A., Gerace, I., Giorgetti, V.: *A Graduated Non–Convexity Technique for Dealing Large Point Spread Funtion* Appl. Sci. 2023, 13(10), pages 28 (2023)
13. Boccuto, A., Gerace, I., Giorgetti, V., Rinaldi, M.: A fast algorithm for the demosaicing problem concerning the bayer pattern. The Open Signal Proc. J. **6**, 1–14 (2019)
14. Boccuto, A., Gerace, I., Martinelli, F.: Half-Quadratic Image Restoration with a Non-Parallelism Constraint. J. Math. Imaging Vis. **59**(2), 270–295 (2017)
15. Boccuto, A., Gerace, I., Pucci, P.: Convex approximation technique for interacting line elements deblurring: a new approach. J. Math. Imaging Vis. **44**(2), 168–184 (2012)
16. Brewster, M.E., Kannan, R.: Nonlinear successive over-relaxation. Numer. Math. **44**(2), 3015–3019 (1984)
17. Buades, A., Coll, B., Morel, J.M., Sbert, C.: Self-similarity driven color demosaicking. IEEE Trans. Image Process. **18**(6), 1192–1202 (2009)
18. Chung, K.-H., Chan, Y.-H.: Color demosaicing using variance of color differences. IEEE Trans. Image Process. **15**(10), 2944–2955 (2006)
19. Chung, K.-H., Chan, Y.-H.: Low-complexity color demosaicing algorithm based on integrated gradients. J. Electron. Imaging **19**(2), 1–15 (2010)
20. Condat, L.: A simple, fast and efficient approach to demoisaicking: Joint demosaicking and denoising. Proc. IEEE Int. Conf. Image Proc. **50**, 905–908 (2010)
21. Condat, L., Mosaddegh, S.: *Joint demosaicking and denoising by total variation minimization*. In: Proceedings of the Image Process-

ing (ICIP), 2012 19th IEEE International Conference on. IEEE; September 2012; Orlando, FL, USA, pp. 2781–2784

22. Cui, K., Jin, Z., Steinbach, E.: *Color image demosaicking using a 3-Stage convolutional neural network structure*, Proc. 25th IEEE Int. Conf. Image Process. (ICIP), pp. 2177-2181, Oct. (2018)
23. Evangelopoulos, X., Brockmeier, A. J., Mu, T., Goulermas, J. Y.: *A Graduated Non-Convexity Relaxation for Large-Scale Seriation.* In: Proceedings of the 2017 SIAM International Conference on Data Mining, Houston, Texas (USA), pp. 462–470 (2017)
24. Ferradans, S., Bertalmío, M., Caselles, V.: Geometry-based demosaicking. IEEE Trans. Image Process. **18**(3), 665–670 (2009)
25. Geman, S., Geman, D.: Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Trans. Pattern Anal. Machine Intell. **6**, 721–740 (1984)
26. Geman, D., Reynolds, G.: Constrained restoration and the recovery of discontinuities. IEEE Trans. Pattern Anal. Mach. Intell. **14**(3), 367–383 (1992)
27. Gharbi, M., Chaurasia, G., Paris, S., Durand, F.: Deep joint demosaicking and denoising. ACM Trans. Graph. **35**(6), 1–12 (2016)
28. Giorgetti, V.: Ill-posed Problems in Computer Vision. Ph. D. Thesis, (2022)
29. Gunturk, B.K., Altunbasak, Y., Mersereau, R.: Colorplane interpolation using alternating projections. IEEE Trans. Image Process. **11**, 997–1013 (2002)
30. Gunturk, B.K., Glotzbach, J., Altunbasak, Y., Schafer, R.W., Mersereau, R.M.: Demosaicking: Color Filter Array Interpolation. IEEE Signal Process. Magazine **22**(1), 44–54 (2005)
31. Hamilton, J. F., Compton, J. T: *Processing Color and Panchromatic Pixels.* U.S. Patent 2007 0024879, Feb. (2007)
32. Hazan, E., Levy, K. Y., Shalev-Shwartz, S.: *On Graduated Optimization for Stochastic Non-Convex Problems.* In: Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W& CP 48, pp. 1–9 (2016)
33. Hirakawa, K., Parks, T.W.: Adaptive homogeneity-directed demosaicing algorithm. IEEE Trans. Image Process. **14**(3), 360–369 (2005)
34. Hirakawa, K., Parks, T.W.: Joint Demosaicing and Denoising. IEEE Trans. Image Process. **15**(8), 2146–2157 (2006)
35. Hirakawa, K., Wolfe, P.: *Spatio-spectral color filter array design for enhanced image fidelity.* In: 2007 IEEE International Conference on Image Processing (ICIP), pp. 81–84 (2007)
36. Jin, Q., Guo, Y., Morel, J.-M., Facciolo, G.: A Mathematical Analysis and Implementation of Residual Interpolation Demosaicking Algorithms. Image Processing On Line **11**, 234–283 (2021). https://doi.org/10.5201/ipol.2021.358
37. Johnson, G.M., Fairchild, M.: A top down description of S-CIELAB and CIEDE2000. Color. Res. Appl. **28**, 425–435 (2003)
38. Khashabi, D., Nowozin, S., Jancsary, J., Fitzgibbon, A.W.: Joint demosaicing and denoising via learned nonparametric random fields. IEEE Trans. Image Process. **23**(12), 4968–4981 (2014)
39. Kiku, D., Monno, Y., Tanaka, M., Okutomi, M.: *Residual interpolation for color image demosaicking.* In: Proceedings of the 2013 20th IEEE International Conference on Image Processing (ICIP), Melbourne, VIC, Australia, pp. 2304?2308 (2013)
40. Kiku, D., Monno, Y., Tanaka, M., Okutomi, M.: *Minimized-Laplacian residual interpolation for color image demosaicking.* In: Proceedings of the SPIE, San Francisco, CA, USA, p. 90230L (2014)
41. Kiku, D., Monno, Y., Tanaka, M., Okutomi, M.: Beyond color difference: Residual interpolation for color image demosaicking. IEEE Trans. Image Process. **25**, 1288–1300 (2016)
42. Klatzer, T., Hammernik, K., Knobelreiter, P., Pock, T.: *Learning joint demosaicing and denoising based on sequential energy minimization.* In: Proceedings of the Computational Photography (ICCP), 2016 IEEE International Conference on. IEEE; May; Evanston, IL, USA. pp. 1–11 (2016)
43. Kodak Eastman Company, "PhotoCD PCD0992", (http://r0k.us/graphics/kodak/)
44. Kokkinos, F., Lefkimmiatis, S.: Iterative joint image demosaicking and denoising using a residual denoising network. IEEE Trans. Image Process. **28**(8), 4177–4188 (2019)
45. Li, X.: Demosaicing by successive approximation. IEEE Trans. Image Process. **14**(3), 370–379 (2005)
46. Li, X., Gunturk, B. K., Zhang, L.: *Image demosaicing: A systematic survey.* In: Proc. SPIE-IS&T Electronic Imaging, Visual Communications and Image Processing, Jan, vol. 6822, no. 1, pp. 1–15 (2008)
47. Liu, L., Jia, X., Liu, J., Tian, Q.: Joint demosaicing and denoising with self guidance. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; August; Seattle, WA, USA. pp. 2240-2249 (2020)
48. Liu, Z.-Y., Qiao, H.: GNCCP-Graduated NonConvexity and Concavity Procedure. IEEE Trans. Pattern Anal. Mach. Intell. **36**(6), 1258–1267 (2014)
49. Liu, Z.-Y., Qiao, H., Su, J.–H: *MAP Inference with MRF by Graduated Non-Convexity and Concavity Procedure.* In: Loo, C. K.; Yap, K. S.; Wong, K. W.; Teoh, A.; Huang, K. (Eds): Neural Information Processing. ICONIP 2014. Lecture Notes in Computer Science, 8835, 404–412 (2014)
50. Mairal, J., Elad, M., Sapiro, G.: Sparse Representation for Color Image Restoration. IEEE Trans. Image Process. **17**(1), 53–69 (2008)
51. Martinec, E. J.: https://github.com/ethiccinema/mlrawviewer
52. Menon, D., Andriani, S., Calvagno, G.: Demosaicing with directional filtering and a posteriori decision. IEEE Trans. Image Process. **16**(1), 132–141 (2007)
53. Menon, D., Calvagno, G.: Demosaicing based on wavelet analysis of the luminance component. Proc. IEEE Int. Conf. Image Processing **2**, 181–184 (2007)
54. Menon, D., Calvagno, G.: Regularization Approaches to Demosaicking. IEEE Trans. Image Process. **18**, 2209–2220 (2009)
55. Menon, D., Calvagno, G.: *Joint demosaicking and denoising with space-varying filters*. In: Proceedings of the 2009 16th IEEE International Conference on Image Processing (ICIP), pp. 477–480 (2009)
56. Menon, D., Calvagno, G.: Color image demosaicking - An overview. Signal Processing: Image Communication **26**, 518–533 (2011)
57. Mobahi, H., Fisher, J.W.: A Theoretical Analysis of Optimization by Gaussian Continuation. In: Wong, W.-K., Lowd, D. (eds.) Twenty-Ninth Conference on Artificial Intelligence of the Association for the Advancement of Artificial Intelligence (AAAI), Proceedings, pp. 1205–1211. Austin, Texas, USA, January 25–30(2015) (2015)
58. Moghadam, A., Aghagolzadeh, M., Kumar, M., Radha, R.: Compressive framework for demosaicing of natural images. IEEE Trans. Image Process. **22**, 2356–2371 (2013)
59. Monno, Y., Kiku, D., Tanaka, M., Okutomi, M.: Adaptive residual interpolation for color and multispectral image demosaicking. Sensors **17**(12), 2787 (2017)
60. Nikolova, M.: Markovian reconstruction using a GNC approach. IEEE Trans. Image Process. **8**(9), 1204–1220 (1999)
61. Nikolova, M., Ng, M.K., Tam, C.-P.: On ℓ_1 Data Fitting and Concave Regularization for Image Recovery. SIAM J. Sci. Comput. **35**(1), A397–A430 (2013)
62. Nikolova, M., Ng, M.K., Zhang, S., Ching, W.-K.: Efficient Reconstruction of Piecewise Constant Images Using Nonsmooth Nonconvex Minimization. SIAM J. Imaging Sci. **1**(1), 2–25 (2008)
63. Park, Y., Lee, S., Jeong, B., Yoon, J.: Joint demosaicing and denoising based on a variational deep image prior neural network. Sensors **20**(10), 2970 (2020)
64. RawTherapee Program, https://rawpedia.rawtherapee.com

65. Robini, M.C., Magnin, I.E.: Optimization by Stochastic Continuation. SIAM J. Imaging Sci. **3**(4), 1096–1121 (2010)
66. Saito, T., Komatsu, T.: *Demosaicing approach based on extended color total-variation regularization.* In: Proc. IEEE Int. Conf. Image Processing, Sep, vol. 1, pp. 885–888 (2008)
67. Smith, T., Egeland, O.: Dynamical Pose Estimation with Graduated Non-Convexity for Outlier Robustness. MIC—Model. Identif. Control **43**(2), 79–89 (2022)
68. Tan, D.S., Chen, W.-Y., Hua, K.-L.: DeepDemosaicking: Adaptive image demosaicking via multiple deep fully convolutional networks. IEEE Trans. Image Process. **27**(5), 2408–2419 (2018)
69. Tan, R., Zhang, K., Zuo, W., Zhang, L.: *Color image demosaicking via deep residual learning*. In: Proceedings of the IEEE Int. Conf. Multimedia Expo (ICME), pp. 793–798 (2017)
70. Xing, W., Egiazarian, K.: End-to-end learning for joint image demosaicing, denoising and super-resolution. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; June; Nashville, TN, USA. pp. 3507–3516 (2021)
71. Yang, H., Antonante, P., Tzoumas, V., Carlone, L.: Graduated non-convexity for robust spatial perception: from non-minimal solvers to global outlier rejection. IEEE Robot. Auto. Lett. **5**(2), 1127–1134 (2020)
72. Ye, W., Ma, K.-K.: Color image demosaicing using iterative residual interpolation. IEEE Trans. Image Process. **24**(12), 5879–5891 (2015)
73. Zhang, X., Wandell, B.: A spatial extension of CIELAB for digital color-image reproduction. J. Soc. Inf. Disp. **5**(1), 61–63 (1997)
74. Zhang, L., Wu, X.: Color demosaicking via directional linear minimum mean square-error estimation. IEEE Trans. Image Process. **14**(12), 2167–2177 (2005)
75. Zhang, L., Wu, X., Buades, A., Li, X.: *Color Demosaicking by Local Directional Interpolation and Non-local Adaptive Thresholding.* J. of Electronic Imaging 20 (2), 023016, pp. 1–17 (2011)
76. Zhang, L., Wu, X., Zhang, D.: Color reproduction from noisy CFA data of single sensor digital cameras. IEEE Trans. Image Process. **16**(9), 2184–2197 (2007)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Antonio Boccuto was born in Catanzaro, Italy, in 1964. He received the degree in Mathematics from the University of Perugia, Italy, in 1987, and the Ph.D. degree in Mathematical Analysis from the Mathematical Institute of the Slovak Academy of Sciences in Bratislava, Slovakia, in 2000. He received the habilitation in Mathematics to Associate Professor from the Comenius University in Bratislava in 2008 and the Italian habilitation in Mathematical Analysis in 2019. He has been a researcher and an assistant professor in Mathematical Analysis at University of Perugia since the Academic Year 1991/1992. His research interests include Measure Theory and Integration, Real Analysis, Approximation Theory, Convex analysis, Applications to Numerical Analysis. He has published more than 150 papers on Mathematics in journals and conference proceedings and is an author and coauthor of books and chapters of books on Measure and Integration Theory and Applications.

Ivan Gerace was born in Cosenza, Italy, in 1967. He received a degree in Computer Science from the University of Pisa, Italy, in 1992 and a Ph.D. in Computational Mathematics and Operational Research from the University of Milan, Italy, in 1999. He has been a researcher in Numerical Analysis since May 2000 at the University of Perugia, Italy. His research interests include Image Processing, Numerical Linear Algebra, and Computational Complexity.

Valentina Giorgetti was born in Narni, Italy, in 1990. In 2014, she received a bachelor's degree in Mathematics and, in 2018, a laurea degree in Mathematics (cum laude) from the University of Perugia, Italy. In 2022, she received a Ph.D. degree in Mathematics from the University of Florence, Italy. Her research interests are Blind Image Deconvolution, Document Analysis, Image Demosaicing, and Blind Source Separation.

Francesca Martinelli was born in Umbertide, Italy, in 1980. She received the degree in mathematics from the University of Perugia, Italy, in 2005, and the Ph.D. degree in mathematics and computer science for information and knowledge management from the University of Perugia in 2009. From 2009 to 2012, she was a Postdoctoral Fellow at the Istituto di Scienza e Tecnologie dell'Informazione, Italian National Research Council (CNR), Pisa, Italy. Currently, she is a math high school teacher.

Anna Tonazzini is a senior researcher at the Signals and Images Lab of the Institute of Information Science and Technologies, CNR, in Pisa. She has coordinated national and international projects on neural networks and learning, computational biology, and document image processing, and has been supervisor of various postdoctoral fellowships. Her current research interests concern image analysis for cultural heritage, and computational methods for structural genomics.