

Article

# Nearly Exact Discrepancy Principle for Low-Count Poisson Image Restoration

Francesca Bevilacqua , Alessandro Lanza, Monica Pragliola \*  and Fiorella Sgallari

Department of Mathematics, University of Bologna, Piazza di Porta San Donato 5, 40126 Bologna, Italy; francesca.bevilacqua8@unibo.it (F.B.); alessandro.lanza2@unibo.it (A.L.); fiorella.sgallari@unibo.it (F.S.)

\* Correspondence: monica.pragliola2@unibo.it

**Abstract:** The effectiveness of variational methods for restoring images corrupted by Poisson noise strongly depends on the suitable selection of the regularization parameter balancing the effect of the regulation term(s) and the generalized Kullback–Liebler divergence data term. One of the approaches still commonly used today for choosing the parameter is the discrepancy principle proposed by Zanella et al. in a seminal work. It relies on imposing a value of the data term approximately equal to its expected value and works well for mid- and high-count Poisson noise corruptions. However, the series truncation approximation used in the theoretical derivation of the expected value leads to poor performance for low-count Poisson noise. In this paper, we highlight the theoretical limits of the approach and then propose a nearly exact version of it based on Monte Carlo simulation and weighted least-square fitting. Several numerical experiments are presented, proving beyond doubt that in the low-count Poisson regime, the proposed modified, nearly exact discrepancy principle performs far better than the original, approximated one by Zanella et al., whereas it works similarly or slightly better in the mid- and high-count regimes.

**Keywords:** image restoration; Poisson noise; discrepancy principle; alternating direction method of multipliers



**Citation:** Bevilacqua, F.; Lanza, A.; Pragliola, M.; Sgallari, F. Nearly Exact Discrepancy Principle for Low-Count Poisson Image Restoration. *J. Imaging* **2022**, *8*, 1. <https://doi.org/10.3390/jimaging8010001>

Academic Editor: Tanja Tarvainen

Received: 25 October 2021

Accepted: 16 December 2021

Published: 23 December 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The image restoration problem under Poisson noise corruption is a task that has been extensively addressed in the literature as it arises in many real-world applications, where the acquired image is obtained by counting the particles, e.g., photons, hitting the image domain [1]. The typical image formation model under blur and Poisson noise corruption takes the form

$$\mathbf{y} = \text{Poiss}(\boldsymbol{\lambda}), \quad \boldsymbol{\lambda} = \mathbf{H}\bar{\mathbf{x}} + \mathbf{b}, \quad (1)$$

where  $\mathbf{H} \in \mathbb{R}^{m \times n}$  models the blur operator, which we assume to be known;  $\mathbf{y} \in \mathbb{N}^m$  and  $\bar{\mathbf{x}} \in \mathbb{R}^n$  are the observed  $m_1 \times m_2$  and unknown  $n_1 \times n_2$  images in column-major vectorized form (with  $m = m_1 m_2$  and  $n = n_1 n_2$ ), respectively;  $\mathbf{b} \in \mathbb{R}^m$  is a non-negative background emission, and where  $\text{Poiss}(\boldsymbol{\lambda}) := (\text{Poiss}(\lambda_1), \dots, \text{Poiss}(\lambda_m))^T$ , with  $\text{Poiss}(\lambda_i)$  indicating the realization of a Poisson-distributed random variable of parameter (mean)  $\lambda_i$ .

When tackling the recovery of  $\bar{\mathbf{x}}$  starting from  $\mathbf{y}$ , one has also to consider the intrinsic constraint

$$\bar{\mathbf{x}} \in \Omega := \{\bar{\mathbf{x}} \in \mathbb{R}^n : \bar{\mathbf{x}} \geq \mathbf{0}\}, \quad (2)$$

which accounts for the pixel values being non-negative.

In a probabilistic perspective [2], problem (1) and (2) can be addressed by modeling the unknown  $\mathbf{x}$  as a random variable. In general, the information on the degradation process is encoded in the so-called *likelihood* probability density function (pdf)  $p(\mathbf{y} | \mathbf{x})$ , while the prior beliefs on the unknown  $\mathbf{x}$  are expressed by the *prior* pdf  $p(\mathbf{x})$ . In the Bayesian

framework, one aims to recover the *posterior* pdf, which is related to the likelihood and the prior term via the Bayes formula:

$$p(\mathbf{x} \mid \mathbf{y}) \propto P(\mathbf{y} \mid \mathbf{x})p(\mathbf{x}).$$

with  $P$  denoting the probability mass function (pmf) that replaces the continuous pdf  $p$  to account for the discrete nature of the data  $\mathbf{y}$ .

According to the Maximum A Posteriori (MAP) estimation approach, the mode of  $p(\mathbf{x} \mid \mathbf{y})$  can be selected as a single-point representative of the posterior distribution, so that the original problem (1) and (2) turns into:

$$\begin{aligned} \mathbf{x}^* &\in \arg \max_{\mathbf{x}} \{ p(\mathbf{y} \mid \mathbf{x}) p(\mathbf{x}) \} \\ &= \arg \min_{\mathbf{x}} \{ -\ln p(\mathbf{y} \mid \mathbf{x}) - \ln p(\mathbf{x}) \}. \end{aligned} \tag{3}$$

In light of the constraint expressed in (2), the general form of the prior pdf reads

$$p(\mathbf{x}) \propto p_c(\mathbf{x})p_0(\mathbf{x}),$$

with

$$p_c(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega \\ 0 & \text{otherwise} \end{cases},$$

and  $p_0(\mathbf{x})$  encoding other information possibly available on  $\mathbf{x}$ . A typical choice for  $p_0(\mathbf{x})$  is given by the Total Variation (TV) Gibbs prior—see [3]—which reads

$$p_0(\mathbf{x}) = \frac{1}{Z} \exp\left(-\alpha \sum_{i=1}^n \|(\mathbf{D}\mathbf{x})_i\|_2\right),$$

where  $Z > 0$  is a normalization constant,  $\alpha > 0$  is the prior parameter and  $\mathbf{D} := (\mathbf{D}_h^T, \mathbf{D}_v^T)^T \in \mathbb{R}^{2n \times n}$  denotes the discrete gradient operator with  $\mathbf{D}_h, \mathbf{D}_v \in \mathbb{R}^{n \times n}$ , two linear operators representing the finite difference discretizations of the first-order partial derivatives of the image  $\mathbf{x}$  in the horizontal and vertical direction. The negative logarithm of the prior pdf  $p(\mathbf{x})$  thus reads

$$-\ln p(\mathbf{x}) = \alpha \sum_{i=1}^n \|(\mathbf{D}\mathbf{x})_i\|_2 + \iota_{\Omega}(\mathbf{x}) + \ln Z, \tag{4}$$

with  $\iota_{\Omega}(\cdot)$  denoting the indicator function of set  $\Omega$ , which is equal to 0 if  $\mathbf{x} \in \Omega$ , or  $+\infty$  otherwise.

Concerning the likelihood pdf, first, we notice that the forward model (1) can be usefully rewritten in component-wise (pixel-wise) form as follows:

$$y_i = \text{Pois}(\lambda_i), \quad \lambda_i = \mathbf{H}_i \mathbf{x} + b_i, \quad i = 1, \dots, m,$$

with  $y_i \in \mathbb{N}$ ,  $\lambda_i, b_i \in \mathbb{R}_+$ , and where  $\mathbf{H}_i \in \mathbb{R}^{1 \times n}$  denotes the  $i$ -th row of matrix  $\mathbf{H}$ . Upon the assumption of independence of the Poisson noise realizations at different pixels, we have:

$$\begin{aligned} -\ln P(\mathbf{y} \mid \mathbf{x}) &= -\ln P(\mathbf{y} \mid \boldsymbol{\lambda}) = -\ln \prod_{i=1}^m P(y_i \mid \lambda_i) \\ &= -\sum_{i=1}^m \ln P(y_i \mid \lambda_i), \quad \boldsymbol{\lambda} = \mathbf{H}\mathbf{x} + \mathbf{b}, \end{aligned} \tag{5}$$

where  $P(y_i \mid \lambda_i)$ , which denotes the probability for  $y_i$  to be the realization of a Poisson-distributed random variable with mean  $\lambda_i$ , reads

$$P(y_i | \lambda_i) = \frac{\lambda_i^{y_i} e^{-\lambda_i}}{y_i!}, \quad y_i \in \mathbb{N}, \lambda_i \in \mathbb{R}_+,$$

with  $\mathbb{R}_+$  denoting the set of non-negative real numbers. Hence, the associated negative log-pmf reads

$$-\ln P(y_i | \lambda_i) = \lambda_i - y_i \ln \lambda_i + \ln(y_i!). \tag{6}$$

By plugging (6) into (5), the negative log-likelihood takes the following form:

$$-\ln P(\mathbf{y} | \mathbf{x}) = \sum_{i=1}^m (\lambda_i - y_i \ln \lambda_i + \ln(y_i!)). \tag{7}$$

Finally, plugging (4) and (7) into (3), dropping out the constant term  $\ln Z$  in (4), readjusting the constant terms in (7) (by adding  $y_i \ln y_i - y_i - \ln(y_i!)$  to each term in the sum), and then dividing the cost function by the positive scalar  $\alpha$ , we obtain the so-called TV-KL variational model:

$$\hat{\mathbf{x}}(\mu) = \arg \min_{\mathbf{x} \in \Omega} \{ J(\mathbf{x}; \mu) := \text{TV}(\mathbf{x}) + \mu \text{KL}(\boldsymbol{\lambda}; \mathbf{y}) \}, \quad \boldsymbol{\lambda} = \mathbf{H}\mathbf{x} + \mathbf{b}, \tag{8}$$

where  $\mu = 1/\alpha$ , the TV semi-norm term [4], is defined by

$$\text{TV}(\mathbf{x}) = \sum_{i=1}^n \|(\mathbf{D}\mathbf{u})_i\|_2,$$

and the term  $\text{KL}(\boldsymbol{\lambda}; \mathbf{y})$  indicates the generalized Kullback–Leibler (KL) divergence between  $\boldsymbol{\lambda} = \mathbf{H}\mathbf{x} + \mathbf{b}$  and the observation  $\mathbf{y}$ , which reads

$$\text{KL}(\boldsymbol{\lambda}; \mathbf{y}) = \sum_{i=1}^m F(\lambda_i; y_i), \quad \text{with } F(\lambda; y) := \lambda - y \ln \lambda + y \ln y - y. \tag{9}$$

Note that the adoption of the MAP strategy within a probabilistic framework yields a minimization problem, which is typically addressed in the context of variational methods for image restoration. The TV term and the KL divergence play the role of the *regularization* and *data fidelity term*, respectively. Moreover, the parameter  $\mu$ , which has been defined starting from the prior parameter  $\alpha$ , is the so-called *regularization parameter* balancing the action of regularization and data fidelity terms.

The selection of a suitable value for the regularization parameter  $\mu$  is of crucial importance for obtaining high-quality results. This relation is highlighted by the explicit dependence in (8) of the solution  $\hat{\mathbf{x}}$  on the parameter  $\mu$ . Very often,  $\mu$  is chosen empirically by brute-force optimization with respect to some visual quality metrics. However, for Poisson data, a large amount of literature has been devoted to the analysis of the Discrepancy Principle (DP), which can be formulated in general terms as follows [5]:

$$\text{Select } \mu = \mu^* \in \mathbb{R}_+ \text{ such that } \mathcal{D}(\mu^*; \mathbf{y}) = \Delta, \tag{10}$$

where the last equality and the scalar  $\Delta \in \mathbb{R}_+$  in (10) are commonly referred to as the *discrepancy equation* and the *discrepancy value*, respectively, while the *discrepancy function*  $\mathcal{D}(\cdot; \mathbf{y}) : \mathbb{R}_+ \rightarrow \mathbb{R}_+$  is defined by

$$\mathcal{D}(\mu; \mathbf{y}) := \text{KL}(\hat{\boldsymbol{\lambda}}(\mu); \mathbf{y}) = \sum_{i=1}^m \left( \mathcal{D}_i(\mu; y_i) := F(\hat{\lambda}_i(\mu); y_i) \right), \tag{11}$$

with function  $F$  defined in (9) and

$$\hat{\boldsymbol{\lambda}}(\mu) = \mathbf{H}\hat{\mathbf{x}}(\mu) + \mathbf{b}. \tag{12}$$

The DP in (10)–(12) formalizes a quite simple idea: choose the value  $\mu^*$  of the regularization parameter  $\mu$  in the TV-KL model (8) such that the value of the KL data fidelity term associated with the solution  $\hat{x}(\mu^*)$  is equal to a prescribed discrepancy value  $\Delta$ . However, applying the DP in an effective manner in practice is not straightforward as several issues concerning the computational efficiency and, more importantly, the quality of the output solutions arise. We examine both of them more closely.

- (I1) *Computational efficiency.* The solution function  $\hat{x}(\mu)$  of model (8) does not admit a closed-form expression and iterative solvers must be used to compute the restored image  $\hat{x}$  associated with any  $\mu$ . Hence, selecting  $\mu^*$  by solving the scalar discrepancy equation defined in (10)–(12) as an efficient preliminary step and then computing the sought restored image  $\hat{x}(\mu^*)$  by iteratively solving model (8) only once is not feasible.
- (I2) *Quality of solution(s).* Even if an efficient algorithm is used for the computation, the obtained restored image  $\hat{x}(\mu^*)$  may be of such low quality that it is of no practical use if the discrepancy value  $\Delta$  in (10) is not suitably chosen.

Issue (I1) concerning computational efficiency has been successfully addressed in [6], where the authors propose to automatically update  $\mu$  along the iterations of the minimization algorithm used for solving the TV-KL model so as to satisfy (at convergence) a specific version of the general DP defined in (10)–(12).

Concerning (I2), we highlight that, in the theoretical hypothesis that the target image  $\bar{x}$  is known, so that  $\bar{\lambda} = \mathbf{H}\bar{x} + \mathbf{b}$  is also known, one would select  $\mu^*$  such that the value of the KL fidelity term associated with the solution  $\hat{x}(\mu^*)$  is equal to the value of the KL fidelity term associated with  $\bar{x}$ . This clearly does not guarantee that the obtained solution  $\hat{x}(\mu^*)$  coincides with the target image  $\bar{x}$ . However, by constraining  $\hat{x}(\mu)$  to belong to the unique level set of the (convex) KL fidelity term containing  $\bar{x}$ , this abstract strategy, which we refer to as the Theoretical DP (TDP), represents an oracle for the general DP in (10)–(12). The TDP is thus formulated as follows:

$$\begin{aligned} &\text{Select } \mu = \mu^* \in \mathbb{R}_+ \text{ such that } \mathcal{D}(\mu^*; \mathbf{y}) = \Delta^{(T)}, \\ &\text{with } \Delta^{(T)} := \sum_{i=1}^m \left( \delta^{(T)}(\bar{\lambda}_i) := F(\bar{\lambda}_i; y_i) \right), \quad \bar{\lambda} = \mathbf{H}\bar{x} + \mathbf{b}, \end{aligned} \tag{13}$$

with function  $F$  defined in (9). Clearly, the value  $\Delta^{(T)}$  cannot be computed in practice as the original image  $\bar{x}$  is not available. As in the case of the Morozov discrepancy principle for Gaussian noise, one could replace the scalar  $\Delta^{(T)}$  with the expected value of the KL fidelity term in (9) regarded as a function of the  $m$ -variate random variable  $\mathbf{Y}$ . We will refer to this version of the DP as Exact (or Expected value) DP (EDP). In the following formula

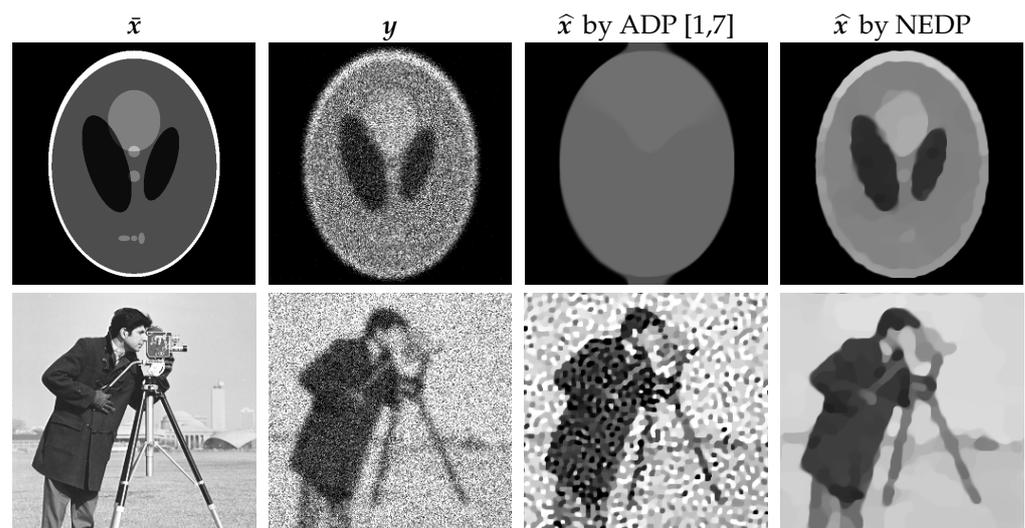
$$\begin{aligned} &\text{Select } \mu = \mu^* \in \mathbb{R}_+ \text{ such that } \mathcal{D}(\mu^*; \mathbf{y}) = \Delta^{(E)}(\mu^*), \\ &\text{with } \Delta^{(E)}(\mu) := \sum_{i=1}^m \left( \delta^{(E)}(\hat{\lambda}_i(\mu)) := E_{Y_i} [F(\hat{\lambda}_i(\mu); Y_i)] \right), \quad \hat{\lambda}(\mu) = \mathbf{H}\hat{x}(\mu) + \mathbf{b}, \end{aligned} \tag{14}$$

where  $E_{Y_i} [F(\hat{\lambda}_i(\mu); Y_i)]$  denotes the expected value of  $F(\hat{\lambda}_i(\mu); Y_i)$  regarded as a function of the Poisson-distributed random variable  $Y_i$ . Nonetheless, unlike the Gaussian noise case, the discrepancy value is not a constant but is a function  $\Delta^{(E)}(\mu)$  of the regularization parameter  $\mu$ , and deriving its analytic expression is a very challenging task. A popular and widespread strategy, originally proposed in [5] for denoising purposes and extended in [7] to the image restoration task, replaces the exact expected value function  $\Delta^{(E)}(\mu)$  with a constant approximation coming from truncating its Taylor series expansion. We will refer to this version of the DP as Approximate DP (ADP). It reads:

$$\begin{aligned} &\text{Select } \mu = \mu^* \in \mathbb{R}_+ \text{ such that } \mathcal{D}(\mu^*; \mathbf{y}) = \Delta^{(A)}, \\ &\text{with } \Delta^{(A)} := \sum_{i=1}^m \left( \delta^{(A)} := \frac{1}{2} \right) = \frac{m}{2}. \end{aligned} \tag{15}$$

Despite its extensive use due to the good performance achieved in the mid- and high-count regimes, the (15) is known to return poor-quality results in the low-count Poisson regime [8], i.e., when the number of photons hitting the image domain is small. In fact, in [7], where the ADP was first extended to the image deblurring task, the authors state (in Remark 3) that the choice of the constant value  $\delta^{(A)} = 1/2$  in (15) may not be “optimal” and suggest replacing it with  $1/2 + \epsilon$ , where  $\epsilon$  is a small positive or negative real number. As a preliminary, qualitative proof of the possible poor performance of (15) in the low-count regime, in the first column of Figure 1, we show the two test images *phantom* and *cameraman*, which have been corrupted by blur and heavy Poisson noise (second column). The TV-KL image restoration model in (8), with regularization parameter  $\mu$  selected according to the (15), has been performed. The output restorations are displayed in the third column of Figure 1. One can see that the rough approximation  $\delta^{(A)} = 1/2$  used in the (15) can either return oversmoothed results, as in the case of *phantom*, or undersmoothed restorations, as for *cameraman*.

Since its proposal in [5], the ADP has been (and still is) widely used for variational image restoration (see, e.g., [9,10]) and it can be regarded as the standard extension of the Morozov DP for Gaussian noise to the Poisson noise case. Then, some literature exists working on the ADP, e.g., by proposing, analyzing, and testing its usage in KL-constrained variational models [11] or by analyzing it theoretically [12]. However, to the best of the authors’ knowledge, the only attempt to improve the ADP by giving a face to the  $\epsilon$  adjustment to the approximate, constant discrepancy value  $\delta^{(A)} = 1/2$  is the one in [8]. The authors in [8] correctly state that  $\epsilon$  must not be a constant, but a function  $\epsilon(\lambda)$  of the photon count level. However, they propose to take  $\epsilon(\lambda)$  as the sum of the second to tenth terms of the same Taylor expansion used in [5]. As we will highlight later in the paper, such expansion converges only for  $\lambda$  approaching  $+\infty$ ; hence, the choice in [8] cannot aspire to improve the performance of ADP in low-count regimes.



**Figure 1.** Original images (first column), observed images corrupted by blur and Poisson noise (second column), restored images obtained by the TV-KL model (8) and (9) with  $\mu$  selected according to the approximate DP [5,7] (third column), and the proposed nearly exact DP (last column).

*Contribution*

The goal of this paper is to provide novel insights about the EDP and the ADP in order to design a novel discrepancy principle capable of outperforming the classical ADP

proposed in [5]. In more detail, we will provide a qualitative study proving that the recovery of a closed-form expression for function  $\Delta^{(E)}(\mu)$  in (14) through its Taylor series expansion used in [5,8] is not only difficult to achieve but also theoretically unfeasible for low-count Poisson regimes. Moreover, we will explore in detail the properties of the ADP motivating the dichotomic behavior (i.e., oversmoothing/undersmoothing) arising upon its adoption in the low-count regime. Finally, based on a simple Monte Carlo simulation followed by weighted least-square fitting, we will derive a novel version of the general DP in (10)–(12) based on a nearly exact (NE) approximation of function  $\Delta^{(E)}(\mu)$  in (14), concisely referred to as NEDP. Our approach will successfully address issues (I1)–(I2). In particular, it will be demonstrated experimentally that NEDP can return high-quality results both for low-count and mid/high-count acquisitions. The good performance of NEDP is anticipated in the last column of Figure 1, where we show the output restorations achieved by using the TV-KL model in (8) and (9) coupled with the novel  $\mu$ -selection strategy.

### 2. Limits of the Approximate DP

The discrepancy principle proposed by Zanella et al. in [5] for Poisson image denoising and then extended to image restoration by Bertero et al. in [7] relies on Lemma 1 in [5], whose proof has been completed in [5] (corrigendum), which we report below for completeness.

**Lemma 1.** *Let  $Y_\lambda$  be a Poisson random variable with expected value  $\lambda \in \mathbb{R}_{++}$  and consider the function of  $Y_\lambda$  defined by*

$$F(Y_\lambda) = \lambda - Y_\lambda \ln \lambda + Y_\lambda \ln Y_\lambda - Y_\lambda = Y_\lambda \ln \left( 1 + \frac{Y_\lambda - \lambda}{\lambda} \right) + \lambda - Y_\lambda. \tag{16}$$

*Then, the following estimate of the expected value of  $F(Y_\lambda)$  holds true for large  $\lambda$ :*

$$\delta^{(E)}(\lambda) = E\{F(Y_\lambda)\} = \delta^{(A)} + O\left(\frac{1}{\lambda}\right), \quad \delta^{(A)} = \frac{1}{2}. \tag{17}$$

Based on the estimate above, and implicitly assuming a sufficiently large  $\lambda$  (i.e., a sufficiently high-count Poisson regime) such that the  $O(1/\lambda)$  term can be neglected, the exact DP outlined in (14) is replaced in [5,7] by the approximation given in (15) and recalled below:

$$\Delta = \Delta^{(A)} = \sum_{i=1}^m \delta^{(A)} = \frac{m}{2}.$$

However, the ADP performs poorly for low-count Poisson images. Our goal here is to highlight that the reason for this lies precisely in the constant approximation  $\delta^{(E)}(\lambda) \approx \delta^{(A)}$  used in (15) and then propose a nearly exact DP based on a much less approximate estimate  $\delta^{(NE)}(\lambda)$  of the expected value function  $\delta^{(E)}(\lambda)$ .

For this purpose, first, we carry out a preliminary Monte Carlo simulation aimed at highlighting the error associated with the approximation in (15). In particular, we consider a discrete set of  $\lambda$  values  $\lambda_i \in [0, 8]$  and, for each  $\lambda_i$ , we generate pseudo-randomly a large number ( $10^6$ ) of realizations of the Poisson random variable  $Y_{\lambda_i}$ . Then, we compute the associated values of the function  $F(Y_{\lambda_i})$  defined in (16) and, finally, for each  $\lambda_i$ , we obtain an estimate  $\widehat{\delta}^{(E)}(\lambda_i)$  of  $\delta^{(E)}(\lambda_i)$  by calculating the sample mean of these function values. The results of this simulation are shown in Figure 2. In particular, in the left figure, we report the computed estimates  $\widehat{\delta}^{(E)}(\lambda_i)$ , whereas, in the right figure, we report the percentage errors (with respect to the estimates) associated with using the constant value  $\delta^{(A)} = 1/2$  as in the (15). The percentage error approaches  $+\infty$  for  $\lambda$  tending to zero and is in the order of  $-10\%$  for  $\lambda \in [1, 4]$ ; then, as expected, it decreases (quite slowly) to zero for  $\lambda$  tending to  $+\infty$ . The error is thus quite large for small  $\lambda$  and this can explain the poor performance of the (15) in the low-count Poisson regime.

In order to obtain a more accurate approximation or even an exact analytical expression for the expected value function  $\delta^{(E)}(\lambda)$ , we now retrace in detail the proof of Lemma 1 given in [5], and completed in [5] (corrigendum), and check if the rough truncation carried out in [5] can be avoided.

After noting that function  $\ln(1 + \varphi)$  is  $C^\infty$  on its domain  $(-1, +\infty)$  and considering its Taylor expansion around 0, the Taylor theorem with the remainder in integral form allows one to write:

$$\begin{aligned} \ln(1 + \varphi) &= \sum_{i=1}^N \frac{(-1)^{i+1}}{i} \varphi^i + r_N(\varphi) = \varphi - \frac{1}{2}\varphi^2 + \frac{1}{3}\varphi^3 - \dots + \frac{(-1)^{N+1}}{N} \varphi^N + r_N(\varphi), \\ r_N(\varphi) &= (-1)^N \int_0^\varphi \frac{(\varphi - t)^N}{(1 + t)^{N+1}} dt, \quad \forall \varphi \in (-1, +\infty). \end{aligned} \tag{18}$$

Replacing the expansion above with  $\varphi = (Y_\lambda - \lambda)/\lambda$  in the expression of function  $F$  defined in (16), we obtain

$$\begin{aligned} F(Y_\lambda) &= Y_\lambda \left( \sum_{i=1}^N \frac{(-1)^{i+1}}{i} \left( \frac{Y_\lambda - \lambda}{\lambda} \right)^i + r_N(\varphi) \right) + \lambda - Y_\lambda \\ &= Y_\lambda r_N(\varphi) + (Y_\lambda - \lambda + \lambda) \left( \sum_{i=1}^N \frac{(-1)^{i+1}}{i} \left( \frac{Y_\lambda - \lambda}{\lambda} \right)^i \right) + \lambda - Y_\lambda \\ &= Y_\lambda r_N(\varphi) + (Y_\lambda - \lambda) \left( \sum_{i=1}^N \frac{(-1)^{i+1}}{i} \left( \frac{Y_\lambda - \lambda}{\lambda} \right)^i \right) \\ &\quad + \lambda \left( \sum_{i=1}^N \frac{(-1)^{i+1}}{i} \left( \frac{Y_\lambda - \lambda}{\lambda} \right)^i \right) + \lambda - Y_\lambda \\ &= Y_\lambda r_N(\varphi) + \sum_{i=1}^N \frac{(-1)^{i+1}}{i} \frac{(Y_\lambda - \lambda)^{i+1}}{\lambda^i} + \sum_{i=2}^N \frac{(-1)^{i+1}}{i} \frac{(Y_\lambda - \lambda)^i}{\lambda^{i-1}} \\ &= Y_\lambda r_N(\varphi) + \sum_{i=1}^N \frac{(-1)^{i+1}}{i} \frac{(Y_\lambda - \lambda)^{i+1}}{\lambda^i} + \sum_{i=1}^{N-1} \frac{(-1)^i}{(i+1)} \frac{(Y_\lambda - \lambda)^{i+1}}{\lambda^i} \\ &= Y_\lambda r_N(\varphi) + \sum_{i=1}^{N-1} \left( \frac{(-1)^{i+1}}{i} + \frac{(-1)^i}{(i+1)} \right) \frac{(Y_\lambda - \lambda)^{i+1}}{\lambda^i} + \frac{(-1)^{N+1}}{N} \frac{(Y_\lambda - \lambda)^{N+1}}{\lambda^N} \\ &= Y_\lambda r_N(\varphi) + \sum_{i=1}^{N-1} \frac{(-1)^{i+1}}{i(i+1)} \frac{(Y_\lambda - \lambda)^{i+1}}{\lambda^i} + \frac{(-1)^{N+1}}{N} \frac{(Y_\lambda - \lambda)^{N+1}}{\lambda^N}. \end{aligned} \tag{19}$$

After noting that the only random quantity in (19) is  $Y_\lambda$ , the expected value reads

$$\delta^{(E)}(\lambda) = E[F(Y_\lambda)] = \sum_{i=0}^{N-1} \omega_i^{(N)} \frac{\eta_{i+2}[Y_\lambda]}{\lambda^{i+1}} + R_N(\lambda), \tag{20}$$

with coefficients  $\omega_i^{(N)} \in \mathbb{Q}$ ,  $i = 0, \dots, N - 1$ , and remainder function  $R_N : \mathbb{R}_{++} \rightarrow \mathbb{R}$  given by

$$\omega_i^{(N)} = \begin{cases} \frac{(-1)^i}{(i+1)(i+2)} & \text{for } i = 0, \dots, N - 2 \\ \frac{(-1)^i}{i+1} & \text{for } i = N - 1 \end{cases}, \quad R_N(\lambda) = E \left[ Y_\lambda r_N \left( \frac{Y_\lambda - \lambda}{\lambda} \right) \right], \tag{21}$$

and where

$$\eta_{i+2}[Y_\lambda] = E \left[ (Y_\lambda - \lambda)^{i+2} \right], \quad i = 0, 1, \dots$$

denote the central moments of order  $i + 2$  of the Poisson random variable  $Y_\lambda$ . It is well known (see [13], p. 162) that these moments can be obtained by the recursive formula

$$\eta_1[Y_\lambda] = 0, \quad \eta_2[Y_\lambda] = \lambda, \quad \eta_{i+2}[Y_\lambda] = \lambda \left( \frac{d\eta_{i+1}[Y_\lambda]}{d\lambda} + (i + 1) \eta_i[Y_\lambda] \right). \quad (22)$$

After noting that, in (20), only moments  $\eta_{i+2}[Y_\lambda]$  with  $i \geq 0$  are present and that they are all divided by  $\lambda$ , it is easy to verify that, by applying (22), one obtains the following general algebraic polynomial expression:

$$\mathcal{P}_i(\lambda) := \frac{\eta_{i+2}[Y_\lambda]}{\lambda} = \sum_{j=0}^{d_i} \vartheta_i^{(j)} \lambda^j, \quad i = 0, 1, \dots, \quad (23)$$

where  $\vartheta_i^{(j)}$  are all integer coefficients with  $\vartheta_i^{(0)} = 1$  for any  $i = 0, 1, \dots$ , and where the degrees  $d_i$  of polynomials  $\mathcal{P}_i(\lambda)$  are given by

$$d_i = \left\lfloor \frac{i}{2} \right\rfloor = 0, 0, 1, 1, 2, 2, \dots \quad \text{for } i = 0, 1, 2, 3, 4, 5, \dots, \quad (24)$$

where  $\lfloor \cdot \rfloor$  denotes the floor function. The first 8 polynomials,  $\mathcal{P}_i(\lambda), i = 0, \dots, 7$ , read

$$\begin{aligned} \mathcal{P}_0(\lambda) &= 1, & \mathcal{P}_1(\lambda) &= 1, \\ \mathcal{P}_2(\lambda) &= 1 + 3\lambda, & \mathcal{P}_3(\lambda) &= 1 + 10\lambda, \\ \mathcal{P}_4(\lambda) &= 1 + 25\lambda + 15\lambda^2, & \mathcal{P}_5(\lambda) &= 1 + 56\lambda + 105\lambda^2, \\ \mathcal{P}_6(\lambda) &= 1 + 119\lambda + 490\lambda^2 + 105\lambda^3, & \mathcal{P}_7(\lambda) &= 1 + 246\lambda + 1918\lambda^2 + 1260\lambda^3. \end{aligned}$$

By replacing the expressions of  $\mathcal{P}_i(\lambda)$  given in (23) into (20), one obtains the following general formula:

$$\delta^{(E)}(\lambda) = E[F(Y_\lambda)] = \sum_{i=0}^{N-1} \left( Q_i^{(N)}(\lambda) := \sum_{j=0}^{d_i} \frac{\psi_i^{(N,j)}}{\lambda^{i-j}} \right) + R_N(\lambda) \quad (25)$$

where the coefficients  $\psi_i^{(N,j)} \in \mathbb{Q}$  of the rational polynomials  $Q_i^{(N)}(\lambda)$  in (25) read

$$\psi_i^{(N,j)} = \omega_i^{(N)} \vartheta_i^{(j)}, \quad i = 0, 1, \dots, N - 1, \quad j = 0, 1, \dots, d_i,$$

with  $\omega_i^{(N)}$  given in (21) and  $\vartheta_i^{(j)}$  defined in (23).

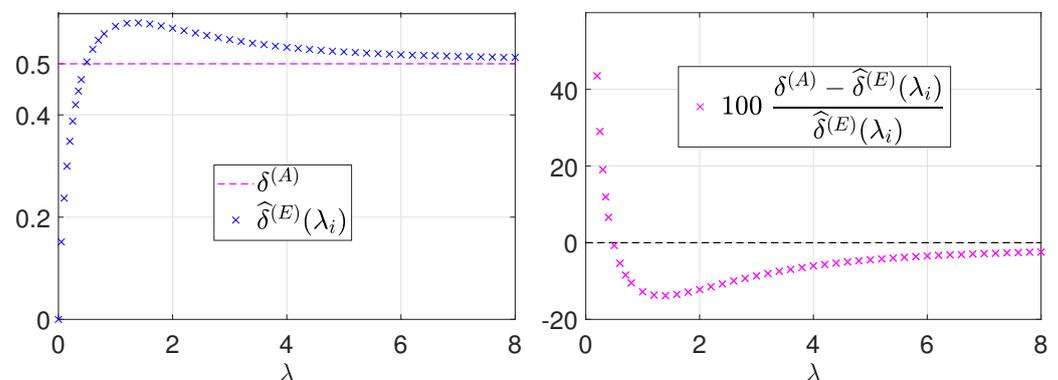
After noting that, from (24), it follows that  $d_i \leq i$  for any  $i = 0, 1, \dots$ , it is a matter of simple algebra to verify that (25) can be equivalently and more compactly rewritten as

$$\delta^{(E)}(\lambda) = E[F(Y_\lambda)] = \sum_{i=0}^{N-1} \frac{\gamma_i^{(N)}}{\lambda^i} + R_N(\lambda), \quad (26)$$

with  $\gamma_i^{(N)} \in \mathbb{Q}$  computable coefficients. In particular, for  $N = 1, \dots, 9$ , we have

$$\begin{aligned}
 \delta^{(E)}(\lambda) &= 1 + R_1(\lambda) \\
 &= \frac{1}{2} - \frac{1}{2\lambda} + R_2(\lambda) \\
 &= \frac{1}{2} + \frac{5}{6\lambda} + \frac{1}{3\lambda^2} + R_3(\lambda) \\
 &= \frac{1}{2} + \frac{1}{12\lambda} - \frac{29}{12\lambda^2} - \frac{1}{4\lambda^3} + R_4(\lambda) \\
 &= \frac{1}{2} + \frac{1}{12\lambda} + \frac{31}{12\lambda^2} + \frac{99}{20\lambda^3} + \frac{1}{5\lambda^4} + R_5(\lambda) \\
 &= \frac{1}{2} + \frac{1}{12\lambda} + \frac{3}{12\lambda^2} - \frac{1003}{60\lambda^3} - \frac{93}{10\lambda^4} - \frac{1}{6\lambda^5} + R_6(\lambda) \\
 &= \frac{1}{2} + \frac{1}{12\lambda} + \frac{1}{12\lambda^2} + \frac{797}{60\lambda^3} + \frac{687}{10\lambda^4} + \frac{713}{42\lambda^5} + \frac{1}{7\lambda^6} + R_7(\lambda) \\
 &= \frac{1}{2} + \frac{1}{12\lambda} + \frac{1}{12\lambda^2} + \frac{19}{120\lambda^3} - \frac{3001}{20\lambda^4} - \frac{39925}{168\lambda^5} - \frac{1721}{56\lambda^6} - \frac{1}{8\lambda^7} + R_8(\lambda) \\
 &= \frac{1}{2} + \frac{1}{12\lambda} + \frac{1}{12\lambda^2} + \frac{19}{120\lambda^3} + \frac{1899}{20\lambda^4} + \frac{516833}{504\lambda^5} + \frac{126829}{168\lambda^6} + \frac{4007}{72\lambda^7} \\
 &\quad + \frac{1}{9\lambda^8} + R_9(\lambda)
 \end{aligned} \tag{27}$$

from which we note how, as the truncation order  $N$  increases, the coefficients  $\gamma_i^{(N)}$  stabilize at some values, which we denote by  $\gamma_i^{(\infty)}$ . Unfortunately, we are not able to obtain an explicit analytical expression for the sequence of coefficients  $\gamma_i^{(\infty)}$  (as we are not able to obtain explicit analytic expressions for the coefficients  $\vartheta_i^{(j)}$  defining the central moments of a Poisson random variable). By means of the Matlab symbolic toolbox, we were able to compute the first 34 coefficients  $\gamma_i^{(\infty)}$ ,  $i = 0, \dots, 33$ , shown (in logarithmic scale) in Figure 3 (left). Determining the subsequent coefficients becomes unfeasible due the huge computation time required. Hence, the following short discussion must be regarded as conjectural as it relies on the assumption that the behavior of coefficients  $\gamma_i^{(\infty)}$ ,  $i = 34, 35, \dots$ , can be smoothly extrapolated from the first 34 coefficients shown in Figure 3 (left). These first 34 coefficients indicate that the coefficient sequence is positive and strictly increasing for  $i \geq 2$ . This implies that making the truncation order  $N$  tend to  $+\infty$ , the (infinite) weighted geometric series in (26) is divergent for  $\lambda \leq 1$ . Even without analyzing the case  $\lambda > 1$ , we can state that an analytical form for function  $\delta^{(E)}(\lambda)$  in the low-count Poisson regime is very unlikely to be obtainable as the sum of the series in (26). In fact, there will be, very likely, at least one pixel such that  $\lambda_i \leq 1$ .



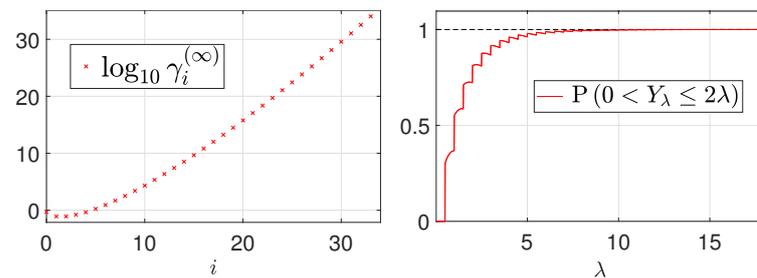
**Figure 2.** Comparison between the approximation  $\delta^{(A)} = 1/2$  of  $\delta^{(E)}(\lambda) = E\{F(Y_\lambda)\}$  used in the (15) proposed in [5,7] and the Monte Carlo estimates  $\widehat{\delta}^{(E)}(\lambda_i)$  for some  $\lambda_i \in [0, 8]$ .

We believe that it is worth concluding this section by pointing out the theoretical reason for the non-convergence of the series in (26). Function  $\ln(1 + \varphi)$  is analytical at

$\varphi = 0$ , but its Maclaurin series converges (pointwise to the function) only for  $\varphi \in (-1, 1]$ . Hence, as  $N$  tends to  $+\infty$ , the Taylor series expansion in (19) converges to the function  $F(Y_\lambda)$  only for  $\varphi = (Y_\lambda - \lambda)/\lambda \in (-1, 1] \iff Y_\lambda \in (0, 2\lambda]$ . However,  $Y_\lambda$  in (20) represents a Poisson random variable with parameter  $\lambda$ . Hence, for  $N$  tending to  $+\infty$ , the series in (20) converges to the function  $\delta^{(E)}(\lambda) = E[F(Y_\lambda)]$  only if the random variable  $Y_\lambda$  satisfies

$$P(0 < Y_\lambda \leq 2\lambda) = 1 \iff \sum_{i=1}^{\lfloor 2\lambda \rfloor} P_{Y_\lambda}(i) = 1. \tag{28}$$

From Figure 3 (right), where we plot the probability in (28) as a function of  $\lambda$ , one can notice that condition (28) for convergence of the series in (20) is fulfilled asymptotically for  $\lambda$  approaching  $+\infty$  but it is not satisfied at all for small  $\lambda$  values.



**Figure 3.** Visual representation of the first 34 terms of the sequence of coefficients  $\gamma_i^{(\infty)}, i = 0, 1, \dots$ , in (26) (left) and the behavior of the probability measure defined in (28) as a function of  $\lambda$  (right).

### 3. A Nearly Exact DP Based on Monte Carlo Simulation

Since it is not possible to derive analytically the expression of function  $\delta^{(E)}(\lambda)$  in (17), the goal in this section is to compute a nearly exact estimate  $\delta^{(NE)}(\lambda)$  of function  $\delta^{(E)}(\lambda)$  based on a simple Monte Carlo simulation approach analogous to that used at the beginning of Section 2. Based on the expected shape of function  $\delta^{(E)}(\lambda)$ —see Figure 2(left)—here, we consider a set of 1385 unevenly distributed  $\lambda$  values  $\lambda_i \in [0, 250]$ , namely

$$\lambda_i \in \{0, 0.01, 0.02, \dots, 5.99, 6, 6.1, 6.2, \dots, 65.9, 66, 67, 68, \dots, 249, 250\}.$$

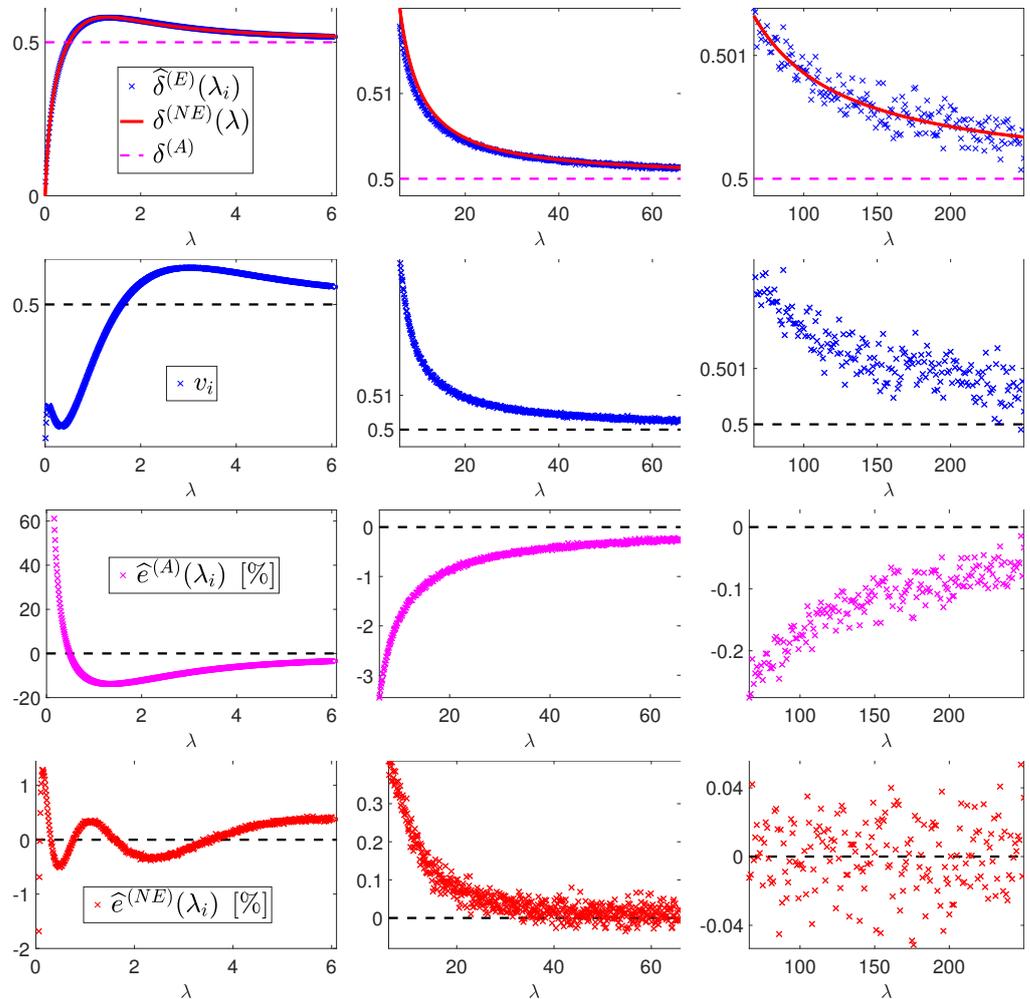
This set comes from the union of three subsets of equally spaced  $\lambda$  values, namely from 0 to 6 with step 0.01, from 6 to 66 with step 0.1, and from 66 to 250 with step 1. For each  $\lambda_i$ , we generate pseudo-randomly a very large number  $S = 5 \times 10^7$  of samples  $y_i^{(j)}, j = 1, \dots, S$ , of the Poisson random variable  $Y_{\lambda_i}$ ; then, we compute the associated values  $f_i^{(j)}, j = 1, \dots, S$ , of the function  $F(Y_{\lambda_i})$  defined in (16) and, finally, we calculate the sample mean  $\widehat{\delta}^{(E)}(\lambda_i)$  and variance  $v_i$  of these function values. In formula,

$$\begin{aligned} y_i^{(j)} = \text{Poiss}(Y_{\lambda_i}), \quad j = 1, \dots, S &\implies f_i^{(j)} = F(y_i^{(j)}), \quad j = 1, \dots, S \\ \implies \widehat{\delta}^{(E)}(\lambda_i) = \frac{1}{S} \sum_{j=1}^S f_i^{(j)}, \quad v_i = \frac{1}{S-1} \sum_{j=1}^S (f_i^{(j)} - \widehat{\delta}^{(E)}(\lambda_i))^2. \end{aligned} \tag{29}$$

Notation for the sample means come from them representing estimates of the sought theoretical means  $\delta^{(E)}(\lambda_i) = E[F(Y_{\lambda_i})], i = 1, \dots, 1385$ . The obtained values  $(\lambda_i, \widehat{\delta}^{(E)}(\lambda_i))$  and  $(\lambda_i, v_i)$  are shown (blue crosses) in the first and second row of Figure 4, respectively. It is well known that  $\widehat{\delta}^{(E)}(\lambda_i)$  and  $v_i$  represent unbiased estimators of the mean and standard deviation of the random variable  $F(Y_{\lambda_i})$  and that, according to the central limit theorem, for a very large number  $S$  of samples (which is definitely our case), the sample mean  $\widehat{\delta}^{(E)}(\lambda_i)$  can be regarded as a realization of a Gaussian random variable with the theoretical

mean  $\delta^{(E)}(\lambda_i)$  of the random variable  $F(Y_{\lambda_i})$  and the sample variance  $v_i$  divided by the number of samples  $S$ . In formulas,

$$\hat{\delta}^{(E)}(\lambda_i) = \text{Gauss}\left(\delta^{(E)}(\lambda_i), \frac{v_i}{S}\right). \tag{30}$$



**Figure 4.** Results of Monte Carlo simulation and weighted least-squares fitting for  $\lambda \in [0, 6]$  (first column),  $\lambda \in [6, 66]$  (second column), and  $\lambda \in [66, 250]$  (third column).

We now want to fit a parametric model  $f(\lambda; \mathbf{c})$ , with  $\mathbf{c}$  the parameter vector, to the obtained Monte-Carlo-simulated data points  $(\lambda_i, \hat{\delta}^{(E)}(\lambda_i))$ ,  $i = 1, \dots, 1385$ . First, in accordance with the trend of these data—see the blue crosses in the first row of Figure 4—and recalling the expected asymptotic behavior of function  $\delta^{(E)}(\lambda)$  for  $\lambda$  approaching  $+\infty$ —see the discussion in Section 2, particularly the first two terms of the expansion in (27)—we choose a model of the form

$$f(\lambda; \mathbf{c}) = \frac{1}{2} + \epsilon(\lambda; \mathbf{c}), \tag{31}$$

with function  $h$  exhibiting the following properties:

$$\epsilon \in C^0(\mathbb{R}_+), \quad \epsilon(0; \mathbf{c}) = -\frac{1}{2}, \quad \epsilon(\lambda; \mathbf{c}) \sim \frac{1}{12\lambda} \text{ for } \lambda \rightarrow +\infty.$$

Then, with the aim of achieving a good trade-off between the model’s ability to accurately fit data and the computational efficiency of its evaluation, we choose the following rational form for function  $\epsilon$ :

$$\epsilon(\lambda; \mathbf{c}) = \frac{\lambda^2 + c_1\lambda + c_2}{12\lambda^3 + c_3\lambda^2 + c_4\lambda - 2c_2}. \tag{32}$$

Thanks to (30), fitting model  $f$  in (31) with  $\epsilon$  as in (32) can be obtained via a Maximum Likelihood (ML) estimation of the parameter vector  $\mathbf{c} = (c_1, c_2, c_3, c_4) \in \mathbb{R}^4$ . In fact, according to (30), the likelihood reads

$$\begin{aligned} \mathcal{L}(\mathbf{c}) &= \prod_{i=1}^S p(\widehat{\delta}^{(E)}(\lambda_i) | \mathbf{c}) = \prod_{i=1}^S \frac{1}{\sqrt{2\pi v_i/S}} \exp\left(-\frac{1}{2} \frac{(\widehat{\delta}^{(E)}(\lambda_i) - f(\lambda_i; \mathbf{c}))^2}{v_i/S}\right) \\ &= \frac{1}{(2\pi/S)^{\frac{S}{2}} \prod_{i=1}^S \sqrt{v_i}} \exp\left(-\frac{S}{2} \sum_{i=1}^S \frac{(\widehat{\delta}^{(E)}(\lambda_i) - f(\lambda_i; \mathbf{c}))^2}{v_i}\right), \end{aligned} \tag{33}$$

and the ML estimate  $\mathbf{c}^{(ML)}$  of  $\mathbf{c}$  can be computed as follows:

$$\mathbf{c}^{(ML)} \in \arg \max_{\mathbf{c} \in \mathbb{R}^4} \mathcal{L}(\mathbf{c}) = \arg \min_{\mathbf{c} \in \mathbb{R}^4} \{-\ln \mathcal{L}(\mathbf{c})\} = \arg \min_{\mathbf{c} \in \mathbb{R}^4} \sum_{i=1}^S w_i (d_i - h(\lambda_i; \mathbf{c}))^2, \tag{34}$$

where we dropped constants (with respect to the optimization variable  $\mathbf{c}$ ) and defined

$$w_i := \frac{1}{v_i}, \quad d_i := \widehat{\delta}^{(E)}(\lambda_i) - \frac{1}{2}, \quad i = 1, \dots, S.$$

Problem (34) is a nonlinear (in particular, rational) weighted least-squares problem. The cost function is non-convex and local minimizers exist. We compute an estimate  $\widehat{\mathbf{c}}$  of  $\mathbf{c}^{(ML)}$  by solving (34) via the iterative trust-region algorithm 1000 times starting from 1000 different initial guesses  $\mathbf{c}^{(0)}$  randomly sampled from a uniform distribution with support  $[-20, 20]^4$  and then picking up the solution  $\widehat{\mathbf{c}}$  yielding the minimum cost function value. The obtained parameter estimate is as follows:

$$\widehat{\mathbf{c}} = (\widehat{c}_1, \widehat{c}_2, \widehat{c}_3, \widehat{c}_4) = (+2.5792, -1.5205, -5.6244, +17.9347). \tag{35}$$

We thus define the nearly exact estimate  $\delta^{(NE)}(\lambda)$  of the theoretical expected value function  $\delta^{(E)}(\lambda) = E[F(Y_\lambda)]$  as the parametric function  $f$  defined in (31), (32) with parameter vector  $\mathbf{c}$  equal to  $\widehat{\mathbf{c}}$  given in (35). In formula,

$$\delta^{(NE)}(\lambda) := f(\lambda; \widehat{\mathbf{c}}) = \frac{1}{2} + \epsilon(\lambda; \widehat{\mathbf{c}}) = \frac{1}{2} + \frac{\lambda^2 + 2.5792\lambda - 1.5205}{12\lambda^3 - 5.6244\lambda^2 + 17.9347\lambda + 3.0410}. \tag{36}$$

In the first row of Figure 4, we plot the constant approximate function  $\delta^{(A)}$  and the obtained nearly exact function  $\delta^{(NE)}(\lambda)$ , whereas, in the third and fourth row of Figure 4, we report the errors  $\widehat{e}^{(A)}(\lambda_i)$  and  $\widehat{e}^{(NE)}(\lambda_i)$ , respectively. They are defined by

$$\widehat{e}^{(X)}(\lambda_i) = 100 \times \frac{\delta^{(X)}(\lambda_i) - \widehat{\delta}^{(E)}(\lambda_i)}{\widehat{\delta}^{(E)}(\lambda_i)} \quad i = 1, 2, \dots, 1385, \quad X \in \{A, NE\},$$

and represent the percentage errors associated with using the approximations  $\delta^{(A)}$  and  $\delta^{(NE)}(\lambda)$  with respect to the very accurate Monte Carlo estimates  $\widehat{\delta}^{(E)}(\lambda_i)$  of the true underlying expected values  $\delta^{(E)}(\lambda_i) = E[F(Y_{\lambda_i})]$ . One can notice that  $|\widehat{e}^{(NE)}(\lambda_i)|$  is around 20 times smaller than  $|\widehat{e}^{(A)}(\lambda_i)|$  for  $\lambda \in [0, 6]$  (first column of Figure 4) and around 10 times less for  $\lambda \in [6, 250]$  (second and third column Figure 4). In particular, in the low-count Poisson regime (which we can roughly associate with  $\lambda \in [0, 6]$ ), the proposed nearly exact estimate of the theoretical expected value function  $\delta^{(E)}(\lambda)$  yields a percentage error

in the order of 0.5%, whereas the constant approximation used in [5,7] leads to a percentage error in the order of 10%. Such a large error is the reason for the poor performance of the (15) in the low-count regime.

We thus propose the following nearly exact DP (NEDP):

$$\begin{aligned} &\text{Select } \mu = \mu^* \in \mathbb{R}_+ \text{ such that } \mathcal{D}(\mu^*; \mathbf{y}) = \Delta^{(NE)}(\mu^*), \\ &\text{with } \Delta^{(NE)}(\mu) = \sum_{i=1}^m \left( \delta^{(NE)}(\hat{\lambda}_i(\mu)) \right) = \frac{m}{2} + \sum_{i=1}^m \epsilon(\hat{\lambda}_i(\mu); \hat{\mathbf{c}}), \\ &\hat{\lambda}(\mu) = \mathbf{H}\hat{\mathbf{x}}(\mu) + \mathbf{b}, \end{aligned} \tag{37}$$

with function  $\epsilon$  and parameter vector  $\hat{\mathbf{c}}$  given in (32) and (35), respectively.

#### 4. Numerical Solution via ADMM

In the following, we detail how to tackle the minimization problem in (8) and (9) when the regularization parameter  $\mu$  is automatically selected according to one of the considered versions of the DP, namely the TDP in (13), the ADP in (15), and the NEDP in (37).

In principle, one could set a fine grid of  $\mu$ -values and compute the solution  $\hat{\mathbf{x}}(\mu)$  corresponding to each  $\mu$ . Then, among the recorded solutions, one could select the one such that the TDP, the ADP, or the NEDP is satisfied. However, this algorithmic scheme, to which we refer as the *a posteriori* optimization procedure, turns out to be particularly costly.

In [6,14], the authors propose to update the regularization parameter according to the ADP along the iterations of the popular Alternating Direction Method of Multipliers (ADMM) [15,16]. Here, we detail the steps of the proposed algorithmic procedure, which can be employed for applying not only the ADP but also the TDP and the NEDP. Finally, we remark that the case of the TDP is only addressed for explanatory purposes and it cannot be performed in practice as  $\bar{\mathbf{x}}$  is not available.

After introducing the auxiliary variables  $\lambda \in \mathbb{R}^m$ ,  $\mathbf{g} \in \mathbb{R}^{2n}$ ,  $\mathbf{z} \in \mathbb{R}^n$ , problem (8) and (9) can be equivalently written as follows:

$$\begin{aligned} \{\mathbf{x}^*, \lambda^*, \mathbf{g}^*, \mathbf{z}^*\} \in \arg \min_{\mathbf{x}, \lambda, \mathbf{g}, \mathbf{z}} \left\{ \sum_{i=1}^n \|\mathbf{g}_i\|_2 + \mu \sum_{i=1}^m (\lambda_i - y_i \ln(\lambda_i)) + \iota_{\mathbb{R}_+^n}(\mathbf{z}) \right\} \\ \text{subject to: } \lambda = \mathbf{H}\mathbf{x} + \mathbf{b}, \quad \mathbf{g} = \mathbf{D}\mathbf{x}, \quad \mathbf{z} = \mathbf{x}, \end{aligned} \tag{38}$$

where, with a little abuse of notation, we define  $\mathbf{g}_i := ((\mathbf{D}_h \mathbf{x})_i, (\mathbf{D}_v \mathbf{x})_i) \in \mathbb{R}^2$ , for every  $i = 1, \dots, n$ .

To solve problem (38), we introduce the augmented Lagrangian function,

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \lambda, \mathbf{g}, \rho_\lambda, \rho_g) &= \sum_{i=1}^n \|\mathbf{g}_i\|_2 + \mu \sum_{i=1}^m (\lambda_i - y_i \ln(\lambda_i)) \\ &\quad - \langle \rho_\lambda, \lambda - (\mathbf{H}\mathbf{x} + \mathbf{b}) \rangle + \frac{\beta_\lambda}{2} \|\lambda - (\mathbf{H}\mathbf{x} + \mathbf{b})\|_2^2 \\ &\quad - \langle \rho_g, \mathbf{g} - \mathbf{D}\mathbf{x} \rangle + \frac{\beta_g}{2} \|\mathbf{g} - \mathbf{D}\mathbf{x}\|_2^2 \\ &\quad - \langle \rho_z, \mathbf{z} - \mathbf{x} \rangle + \frac{\beta_z}{2} \|\mathbf{z} - \mathbf{x}\|_2^2 + \iota_{\mathbb{R}_+^n}(\mathbf{z}) \end{aligned} \tag{39}$$

where  $\rho_\lambda \in \mathbb{R}^m$ ,  $\rho_g \in \mathbb{R}^{2n}$ ,  $\rho_z \in \mathbb{R}^n$  are the vectors of Lagrange multipliers associated with the linear constraints in (38), while  $\beta_\lambda, \beta_g, \beta_z \in \mathbb{R}_{++}$  are the ADMM penalty parameters.

By setting  $\mathbf{u} := (\mathbf{x}; \lambda; \mathbf{g}; \mathbf{z})$ ,  $\boldsymbol{\rho} = (\rho_\lambda; \rho_g; \rho_z)$ ,  $\mathbf{U} := \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^{2n} \times \mathbb{R}^n$ , and  $\mathbf{R} := \mathbb{R}^m \times \mathbb{R}^{2n} \times \mathbb{R}^n$ , we observe that solving (38) amounts to seeking solutions of the saddle-point problem:

$$\begin{aligned} &\text{Find } (\mathbf{u}^*, \boldsymbol{\rho}^*) \in \mathbf{U} \times \mathbf{R} \text{ such that} \\ &\mathcal{L}(\mathbf{u}^*, \boldsymbol{\rho}) \leq \mathcal{L}(\mathbf{u}^*, \boldsymbol{\rho}^*) \leq \mathcal{L}(\mathbf{u}, \boldsymbol{\rho}^*) \quad \forall (\mathbf{u}, \boldsymbol{\rho}) \in \mathbf{U} \times \mathbf{R}. \end{aligned} \tag{40}$$

Upon suitable initialization, and for any  $k \geq 0$ , the  $k$ -th iteration of the ADMM algorithm applied to the solution of (40) with the augmented Lagrangian function  $\mathcal{L}$  defined in (39) reads

$$\boldsymbol{\lambda}^{(k+1)} \in \arg \min_{\boldsymbol{\lambda} \in \mathbb{R}^m} \mathcal{L}(\mathbf{x}^{(k)}, \boldsymbol{\lambda}, \mathbf{g}^{(k)}, \mathbf{z}^{(k)}, \boldsymbol{\rho}_\lambda^{(k)}, \boldsymbol{\rho}_g^{(k)}, \boldsymbol{\rho}_z^{(k)}), \tag{41}$$

$$\mathbf{g}^{(k+1)} \in \arg \min_{\mathbf{g} \in \mathbb{R}^{2n}} \mathcal{L}(\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k+1)}, \mathbf{g}, \mathbf{z}^{(k)}, \boldsymbol{\rho}_\lambda^{(k)}, \boldsymbol{\rho}_g^{(k)}, \boldsymbol{\rho}_z^{(k)}), \tag{42}$$

$$\mathbf{z}^{(k+1)} \in \arg \min_{\mathbf{z} \in \mathbb{R}^n} \mathcal{L}(\mathbf{x}^{(k)}, \boldsymbol{\lambda}^{(k+1)}, \mathbf{g}^{(k+1)}, \mathbf{z}, \boldsymbol{\rho}_\lambda^{(k)}, \boldsymbol{\rho}_g^{(k)}, \boldsymbol{\rho}_z^{(k)}), \tag{43}$$

$$\mathbf{x}^{(k+1)} \in \arg \min_{\mathbf{x} \in \mathbb{R}^n} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}^{(k+1)}, \mathbf{g}^{(k+1)}, \mathbf{z}^{(k+1)}, \boldsymbol{\rho}_\lambda^{(k)}, \boldsymbol{\rho}_g^{(k)}, \boldsymbol{\rho}_z^{(k)}), \tag{44}$$

$$\boldsymbol{\rho}_\lambda^{(k+1)} = \boldsymbol{\rho}_\lambda^{(k)} - \beta_\lambda (\boldsymbol{\lambda}^{(k+1)} - (\mathbf{H}\mathbf{x}^{(k+1)} + \mathbf{b})), \tag{45}$$

$$\boldsymbol{\rho}_g^{(k+1)} = \boldsymbol{\rho}_g^{(k)} - \beta_g (\mathbf{g}^{(k+1)} - \mathbf{D}\mathbf{x}^{(k+1)}), \tag{46}$$

$$\boldsymbol{\rho}_z^{(k+1)} = \boldsymbol{\rho}_z^{(k)} - \beta_z (\mathbf{z}^{(k+1)} - \mathbf{x}^{(k+1)}). \tag{47}$$

In the following subsections, we discuss the solution of subproblems (41)–(44) for the four primal variables  $\boldsymbol{\lambda}$ ,  $\mathbf{g}$ ,  $\mathbf{z}$ ,  $\mathbf{x}$ . Since the solution of the subproblem for variable  $\boldsymbol{\lambda}$  is the most complicated and requires the application of the DP, it is presented last.

#### 4.1. Solving Subproblem for $\mathbf{g}$

The subproblem for  $\mathbf{g}$  in (42) reads

$$\begin{aligned} \mathbf{g}^{(k+1)} &\in \arg \min_{\mathbf{g} \in \mathbb{R}^{2n}} \left\{ \sum_{i=1}^n \|\mathbf{g}_i\|_2 - \langle \boldsymbol{\rho}_g^{(k)}, \mathbf{g} - \mathbf{D}\mathbf{x}^{(k)} \rangle + \frac{\beta_g}{2} \|\mathbf{g} - \mathbf{D}\mathbf{x}^{(k)}\|_2^2 \right\} \\ &= \arg \min_{\mathbf{g} \in \mathbb{R}^{2n}} \left\{ \sum_{i=1}^n \|\mathbf{g}_i\|_2 + \frac{\beta_g}{2} \|\mathbf{g} - \mathbf{w}^{(k)}\|_2^2 \right\}, \quad \mathbf{w}^{(k)} = \mathbf{D}\mathbf{x}^{(k)} + \frac{1}{\beta_g} \boldsymbol{\rho}_g^{(k)}. \end{aligned} \tag{48}$$

Solving (48) is equivalent to solving the  $n$  independent two-dimensional minimization problems

$$\mathbf{g}_i^{(k+1)} = \arg \min_{\mathbf{g}_i \in \mathbb{R}^2} \left\{ \|\mathbf{g}_i\|_2 + \frac{\beta_g}{2} \|\mathbf{g}_i - \mathbf{w}_i^{(k)}\|_2^2 \right\}, \quad i = 1, \dots, n,$$

which yields the unique solution

$$\mathbf{g}_i^{(k+1)} = \mathbf{w}_i^{(k)} \max \left( 1 - \frac{1}{\beta_g \|\mathbf{w}_i^{(k)}\|_2}, 0 \right), \quad i = 1, \dots, n. \tag{49}$$

#### 4.2. Solving Subproblem for $\mathbf{z}$

The subproblem for  $\mathbf{z}$  in (43) reads

$$\begin{aligned} \mathbf{z}^{(k+1)} &\in \arg \min_{\mathbf{z} \in \mathbb{R}^n} \left\{ - \langle \boldsymbol{\rho}_z^{(k)}, \mathbf{z} - \mathbf{x}^{(k)} \rangle + \frac{\beta_z}{2} \|\mathbf{z} - \mathbf{x}^{(k)}\|_2^2 + \iota_{\mathbb{R}_+^n}(\mathbf{z}) \right\} \\ &= \arg \min_{\mathbf{z} \in \mathbb{R}_+^n} \|\mathbf{z} - \mathbf{q}^{(k)}\|_2, \quad \mathbf{q}^{(k)} = \mathbf{x}^{(k)} + \frac{1}{\beta_z} \boldsymbol{\rho}_z^{(k)}. \end{aligned} \tag{50}$$

Hence, the solution  $\mathbf{z}^{(k+1)}$  is given by the unique Euclidean projection of vector  $\mathbf{q}^{(k)}$  defined in (50) onto the (convex) nonnegative orthant and admits the simple closed-form expression

$$z_i^{(k)} = \max(q_i^{(k)}, 0), \quad i = 1, \dots, n. \tag{51}$$

4.3. Solving Subproblem for  $\mathbf{x}$

After dropping the constant terms, the  $\mathbf{x}$ -subproblem in (44) reads:

$$\begin{aligned} \mathbf{x}^{(k+1)} \in \arg \min_{\mathbf{x} \in \mathbb{R}^n} & \left\{ -\langle \boldsymbol{\rho}_\lambda^{(k)}, \boldsymbol{\lambda}^{(k+1)} - (\mathbf{H}\mathbf{x} + \mathbf{b}) \rangle + \frac{\beta_\lambda}{2} \|\boldsymbol{\lambda}^{(k+1)} - (\mathbf{H}\mathbf{x} + \mathbf{b})\|_2^2 \right. \\ & - \langle \boldsymbol{\rho}_g^{(k)}, \mathbf{g}^{(k+1)} - \mathbf{D}\mathbf{x} \rangle + \frac{\beta_g}{2} \|\mathbf{g}^{(k+1)} - \mathbf{D}\mathbf{x}\|_2^2 \\ & \left. - \langle \boldsymbol{\rho}_z^{(k)}, \mathbf{z}^{(k+1)} - \mathbf{x} \rangle + \frac{\beta_z}{2} \|\mathbf{z}^{(k+1)} - \mathbf{x}\|_2^2 \right\}. \end{aligned} \tag{52}$$

By imposing a first-order optimality condition on the quadratic cost function in (52), after simple algebraic manipulations, we obtain the following linear system of equations:

$$\begin{aligned} \left( \mathbf{D}^T \mathbf{D} + \frac{\beta_\lambda}{\beta_g} \mathbf{H}^T \mathbf{H} + \frac{\beta_z}{\beta_g} \mathbf{I}_n \right) \mathbf{x} &= \mathbf{D}^T \left( \mathbf{g}^{(k+1)} - \frac{1}{\beta_g} \boldsymbol{\rho}_g^{(k)} \right) \\ &+ \frac{\beta_\lambda}{\beta_g} \mathbf{H}^T \left( \boldsymbol{\lambda}^{(k+1)} - \mathbf{b} - \frac{1}{\beta_\lambda} \boldsymbol{\rho}_\lambda^{(k)} \right) \\ &+ \frac{\beta_z}{\beta_g} \left( \mathbf{z}^{(k+1)} - \frac{1}{\beta_z} \boldsymbol{\rho}_z^{(k)} \right), \end{aligned} \tag{53}$$

which is solvable since the coefficient matrix is symmetric positive definite and hence nonsingular. When assuming periodic boundary conditions for  $\mathbf{x}$ , the blur matrix  $\mathbf{H}$  is square, i.e.,  $m = n$ , and, more importantly,  $\mathbf{D}^T \mathbf{D}$ ,  $\mathbf{H}^T \mathbf{H}$  and—trivially— $\mathbf{I}$  are block circulant matrices with circulant blocks. Hence, the linear system (53) can be solved efficiently by one application of the direct 2D Fast Fourier Transform (FFT) and one application of the inverse 2D FFT, each at a cost of  $O(n \log n)$ .

4.4. Solving Subproblem for  $\boldsymbol{\lambda}$  and Applying the DP

The subproblem for  $\boldsymbol{\lambda}$  in (41) reads

$$\begin{aligned} \boldsymbol{\lambda}^{(k+1)} &= \arg \min_{\boldsymbol{\lambda} \in \mathbb{R}^m} \left\{ \mu \sum_{i=1}^m (\lambda_i - y_i \ln \lambda_i) - \langle \boldsymbol{\rho}_\lambda^{(k)}, \boldsymbol{\lambda} - (\mathbf{H}\mathbf{x}^{(k)} + \mathbf{b}) \rangle \right. \\ & \left. + \frac{\beta_\lambda}{2} \|\boldsymbol{\lambda} - (\mathbf{H}\mathbf{x}^{(k)} + \mathbf{b})\|_2^2 \right\}. \end{aligned} \tag{54}$$

After manipulating algebraically the last two terms of the cost function in (54), dropping constant terms, and then dividing by the positive scalar  $\beta_\lambda$ , problem (54) can be equivalently rewritten as follows:

$$\begin{aligned} \boldsymbol{\lambda}^{(k+1)}(\gamma) &= \arg \min_{\boldsymbol{\lambda} \in \mathbb{R}^m} \left\{ \gamma \sum_{i=1}^m (\lambda_i - y_i \ln \lambda_i) + \frac{1}{2} \|\boldsymbol{\lambda} - \mathbf{v}^{(k)}\|_2^2 \right\}, \\ \text{with } \gamma &= \frac{\mu}{\beta_\lambda}, \quad \mathbf{v}^{(k)} = \mathbf{H}\mathbf{x}^{(k)} + \mathbf{b} + \frac{1}{\beta_\lambda} \boldsymbol{\rho}_\lambda. \end{aligned} \tag{55}$$

In (55), we introduced the explicit dependence of the solution  $\boldsymbol{\lambda}^{(k+1)}$  on the parameter  $\gamma$ , which is the basis of the application of the DP. Notice that the ADMM penalty parameter  $\beta_\lambda$  is fixed; hence,  $\gamma$  plays the role of the regularization parameter  $\mu$  in the DP applied to this

subproblem. Problem (55) can be further simplified after noting that it can be equivalently rewritten in component-wise (pixel-wise) form as follows:

$$\lambda_i^{(k+1)}(\gamma) \in \arg \min_{\lambda_i \in \mathbb{R}_+} \left\{ \gamma(\lambda_i - y_i \ln \lambda_i) + \frac{1}{2}(\lambda_i - v_i^{(k)})^2 \right\}, \quad i = 1, \dots, m. \quad (56)$$

It is easy to prove that, for any  $\gamma \in \mathbb{R}_+$  and independently of the constants  $y_i \in \mathbb{N}$  and  $v_i^{(k)} \in \mathbb{R}$ , all the minimization problems in (56) admit a unique solution given by

$$\lambda_i^{(k+1)}(\gamma) = \frac{1}{2} \left( (v_i^{(k)} - \gamma) + \sqrt{(v_i^{(k)} - \gamma)^2 + 4y_i\gamma} \right). \quad (57)$$

We now want to apply one among the DP versions—namely, (13), (15), and the proposed (37)—outlined in Sections 1 and 3 for selecting a value of the free parameter  $\gamma$  in (57). In particular, we select  $\gamma = \gamma^{(k+1)}$  such that  $\gamma^{(k+1)}$  satisfies the discrepancy equation, which, in accordance with the general definition given in (10)–(12), takes here the form

$$G(\gamma; \mathbf{y}) := \mathcal{D}(\gamma; \mathbf{y}) - \Delta = 0 \quad (58)$$

where the discrepancy function reads

$$\mathcal{D}(\gamma; \mathbf{y}) = \sum_{i=1}^m \mathcal{D}_i(\gamma; y_i) = \sum_{i=1}^m F(\lambda_i^{(k+1)}(\gamma); y_i), \quad (59)$$

with function  $F$  defined in (9), and where the discrepancy value  $\Delta$ , according to the definitions given in (13), (15) and (37), takes one of the following values/forms:

$$\Delta = \begin{cases} \Delta^{(T)} & = \sum_{i=1}^m F((\mathbf{H}\bar{\mathbf{x}} + \mathbf{b})_i; y_i) & \text{for (13),} \\ \Delta^{(A)} & = \frac{m}{2} & \text{for (15),} \\ \Delta^{(NE)}(\gamma) & = \frac{m}{2} + \sum_{i=1}^m \epsilon(\lambda_i^{(k+1)}(\gamma); \hat{\mathbf{c}}) & \text{for (37),} \end{cases} \quad (60)$$

with rational polynomial function  $\epsilon$  defined in (32) and parameter vector  $\hat{\mathbf{c}}$  given in (35). We notice that  $\Delta^{(T)}$  and  $\Delta^{(A)}$  are two positive scalars that can be computed once for all and do not change their values during the ADMM iterations, whereas  $\Delta^{(NE)}(\gamma)$  is a function of  $\gamma$ , which almost certainly changes its shape along the ADMM iterations (due to function  $\lambda_i^{(k+1)}(\gamma)$  in (57) changing its shape when vector  $\mathbf{v}^{(k)}$  in (55) changes).

Summing up, the complete procedure for the DP-based update of the parameter  $\gamma$  and, then, of the variable  $\lambda$  reads as follows:

$$\mathbf{v}^{(k)} = \mathbf{H}\mathbf{x}^{(k)} + \mathbf{b} + \frac{1}{\beta_\lambda} \rho_\lambda, \quad (61)$$

$$\gamma^{(k+1)} = \text{root of the discrepancy equation in (58), (59), (60),} \quad (62)$$

$$\lambda_i^{(k+1)}(\gamma^{(k+1)}) \quad \text{computed by (57), for } i = 1, \dots, m. \quad (63)$$

Concerning the ADP, in [6], the authors have proven that along the ADMM iterations, the function  $\mathcal{D}(\gamma; \mathbf{y})$  is convex and decreasing so that the existence and the uniqueness of the solution of the discrepancy equation in (58) with  $\Delta = \Delta^{(A)}$  is guaranteed. The same result can be immediately extended to the case of the TDP. When considering the NEDP, the functional form of  $\Delta^{(NE)}(\gamma)$  is such that the above result cannot be straightforwardly applied. However, the following proposition on the existence of a solution for the discrepancy equation (58) with  $\Delta = \Delta^{(NE)}$  holds true.

**Proposition 1.** Consider the discrepancy equation in (58)–(60) with  $\Delta = \Delta^{(NE)}(\gamma)$  and with vector  $\mathbf{v}^{(k)}$  and function  $\lambda_i^{(k+1)}(\gamma)$  defined as in (55) and (57), respectively, and let

$$\mathbf{t}^{(k)} := \max\{\mathbf{v}^{(k)}, 0\}. \tag{64}$$

Then, the discrepancy equation admits a solution if the following condition is fulfilled:

$$\exists i: y_i \neq 0 \quad \wedge \quad \mathcal{T}(\mathbf{t}^{(k)}, \mathbf{y}) := \sum_{i=1}^m T(t_i^{(k)}, y_i) \geq \frac{m}{2}, \tag{65}$$

where function  $T: \mathbb{R}_+ \times \mathbb{N} \rightarrow \mathbb{R}$  is defined by

$$T(t, y) = F(t; y) - \epsilon(t; \hat{\mathbf{c}}), \tag{66}$$

with function  $F$ , function  $\epsilon$ , and parameter vector  $\hat{\mathbf{c}}$  given in (9), (32), and (35), respectively.

**Proof.** Since functions  $F$  in (9),  $\epsilon$  in (32), and  $\lambda_i^{(k+1)}$  in (57) are all continuous, then the function  $G$  defined in (58)–(60) with  $\Delta = \Delta^{(NE)}(\gamma)$  is continuous in the variable  $\gamma$  on its entire domain  $\gamma \in \mathbb{R}_+$ , for any  $\mathbf{y} \in \mathbb{N}^m$  and any  $\mathbf{v}^{(k)} \in \mathbb{R}^m$ .

Then, it is easy to prove that function  $\lambda_i^{(k+1)}(\gamma)$  in (57) satisfies

$$\lambda_i^{(k+1)}(0) = \max\{v_i^{(k)}, 0\} = t_i^{(k)}, \quad \lim_{\gamma \rightarrow +\infty} \lambda_i^{(k+1)}(\gamma) = y_i, \tag{67}$$

with vector  $\mathbf{t}^{(k)}$  defined in (64).

It thus follows from (67) and from the definition of functions  $\mathcal{D}$  in (59) and  $\Delta^{(NE)}$  in (60) that

$$\begin{aligned} G(0; \mathbf{y}) &= \mathcal{D}(0; \mathbf{y}) - \Delta^{(NE)}(0) = \sum_{i=1}^m F(\lambda_i^{(k+1)}(0); y_i) - \frac{m}{2} - \sum_{i=1}^m \epsilon(\lambda_i^{(k+1)}(0); \hat{\mathbf{c}}) \\ &= \sum_{i=1}^m (F(t_i^{(k)}; y_i) - \epsilon(t_i^{(k)}; \hat{\mathbf{c}})) - \frac{m}{2} = \mathcal{T}(\mathbf{t}^{(k)}, \mathbf{y}) - \frac{m}{2}, \end{aligned} \tag{68}$$

and that

$$\begin{aligned} \lim_{\gamma \rightarrow +\infty} G(\gamma; \mathbf{y}) &= \lim_{\gamma \rightarrow +\infty} (\mathcal{D}(\gamma; \mathbf{y}) - \Delta^{(NE)}(\gamma)) \\ &= \lim_{\gamma \rightarrow +\infty} \left( \sum_{i=1}^m F(\lambda_i^{(k+1)}(\gamma); y_i) - \frac{m}{2} - \sum_{i=1}^m \epsilon(\lambda_i^{(k+1)}(\gamma); \hat{\mathbf{c}}) \right) \\ &= \sum_{i=1}^m F(y_i; y_i) - \sum_{i=1}^m \left( \frac{1}{2} + \epsilon(y_i; \hat{\mathbf{c}}) \right) \end{aligned} \tag{69}$$

$$= - \sum_{i=1}^m f(y_i; \hat{\mathbf{c}}) < 0 \text{ if } \exists i: y_i \neq 0, \tag{70}$$

where function  $\mathcal{T}$  in (68) is defined in (65), cancelling the first summation in (69) comes from  $F(y; y) = 0$  for any  $y \in \mathbb{R}_+$  (see the definition of function  $F$  in (9), where  $y \ln y = 0$  for  $y = 0$ ) and (70) comes from the definition of function  $f$  in (36).

From (70) and the continuity of function  $G(\gamma; \mathbf{y})$ , we can conclude that, for any  $\mathbf{y} \neq 0$ , the discrepancy equation  $G(\gamma; \mathbf{y}) = 0$  admits a solution if  $G(0; \mathbf{y}) \geq 0$ . It thus follows from (68) that the sufficient condition in (64)–(66) holds true.  $\square$

In Algorithm 1, we outline the general ADMM-based scheme used for solving image restoration variational models of the TV-KL form in (8) and (9) with automatic update/selection of the regularization parameter  $\mu$  according to one of the considered versions

of the DP. We refer to the general scheme as DP-ADMM, whereas the specific schemes using one among the DP versions TDP, ADP, and NEDP will be named TDP-ADMM, ADP-ADMM, and NEDP-ADMM, respectively. Notice that the  $\gamma$ -update at step 3 can be performed by means of a derivative-free approach, such as bisection or the secant method.

---

**Algorithm 1:** General DP-ADMM approach for image restoration variational models of the TV-KL form in (8) and (9) and automatic selection of  $\mu$  via DP.

---

**inputs:** observed degraded image  $\mathbf{y} \in \mathbb{N}^m$ , emission background  $\mathbf{b} \in \mathbb{R}_+^m$ , blur and regularization operators  $\mathbf{H} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{D} \in \mathbb{R}^{2n \times n}$

**output:** estimated restored image  $\hat{\mathbf{x}} \in \mathbb{R}^n$

1. **initialise:** set  $\mathbf{x}^{(0)} = \mathbf{y}$
  2. **for**  $k = 0, 1, 2, \dots$  *until convergence* **do:**
  3.     · compute  $\gamma^{(k+1)} = \mu^{(k+1)} / \beta_\lambda$  by (61) and (62)
  4.     · compute  $\lambda^{(k+1)}$  by (63)
  5.     · compute  $\mathbf{g}^{(k+1)}$  by (49)
  6.     · compute  $\mathbf{z}^{(k+1)}$  by (51)
  7.     · compute  $\mathbf{x}^{(k+1)}$  by (53)
  8.     · compute  $\rho_\lambda^{(k+1)}, \rho_g^{(k+1)}, \rho_z^{(k+1)}$  by (45)–(47)
  9. **end for**
  10.  $\hat{\mathbf{x}} = \mathbf{x}^{(k+1)}$
- 

### 5. Numerical Results

In this section, we evaluate the performance of the proposed NEDP in (37) for the automatic selection of the regularization parameter  $\mu$  in image restoration variational models of the TV-KL form in (8) and (9).

Our approach is compared with the TDP and the ADP in (13) and (15), respectively. For each criterion, we perform the ADMM-based scheme outlined in Algorithm 1. As recalled in Section 4, the  $\mu$ -selection problem along the ADMM iterations always admits a unique solution under the adoption of the ADP and TDP. Concerning the NEDP-ADMM, at the generic iteration  $k$  of Algorithm 1, the regularization parameter  $\mu$  is updated provided that the condition stated in Proposition 1 is satisfied. If this is not the case, the parameter update is not performed and  $\mu^{(k)} = \mu^{(k-1)}$ .

The numerical tests have been designed with the following twofold aim:

- (i) to prove that the proposed NEDP criterion is capable of selecting optimal  $\mu$  values returning high-quality restoration results and, in particular, that it outperforms the classical ADP criterion in the low-count Poisson regime;
- (ii) to prove that the proposed NEDP-ADMM scheme outlined in Algorithm 1 is capable of automatically selecting such optimal  $\mu$  values in a robust (and efficient) manner.

More specifically, the latter point will be proven by showing that the iterated and the *a posteriori* version of our approach behave similarly in terms of  $\mu$ -selection.

The  $\mu$ -values selected by the TDP, the ADP, and the NEDP applied *a posteriori* will be denoted by  $\mu^{(T)}, \mu^{(A)}, \mu^{(NE)}$ , respectively, while the output  $\mu$ -value of the ADP-ADMM and of the NEDP-ADMM scheme will be denoted by  $\hat{\mu}^{(A)}$  and  $\hat{\mu}^{(NE)}$ , respectively.

The quality of the restorations  $\hat{\mathbf{x}}$  with respect to the original uncorrupted image  $\bar{\mathbf{x}}$  will be assessed by means of two scalar measures, namely the Structural Similarity Index (SSIM) [17] and the Improved-Signal-to-Noise Ratio (ISNR), defined by

$$\text{ISNR}(\hat{\mathbf{x}}, \bar{\mathbf{x}}) = 10 \log_{10} \frac{\|\bar{\mathbf{x}} - \mathbf{b}\|_2^2}{\|\bar{\mathbf{x}} - \hat{\mathbf{x}}\|_2^2}.$$

For all tests, the iterations of the ADMM-based scheme in Algorithm 1 are stopped as soon as

$$\epsilon_x^{(k)} = \frac{\|x^{(k)} - x^{(k-1)}\|_2}{\|x^{(k-1)}\|_2} < 10^{-5}, \quad k \in \mathbb{N} \setminus \{0\},$$

and the ADMM penalty parameters  $\beta_\lambda, \beta_g, \beta_z$  are set manually so as to achieve fast convergence. More precisely, in each test, the same triplet  $(\beta_\lambda, \beta_g, \beta_z)$  of ADMM penalty parameters is used for the three compared discrepancy principles TDP, ADP, and NEDP, with  $\beta_\lambda, \beta_g, \beta_z \in [0.5, 2]$ .

We consider the four test images, each with pixel values between 0 and 1, shown in Figure 5. The acquisition process (1) has been simulated as follows. First, the original image is multiplied by a factor  $\kappa \in \mathbb{N} \setminus \{0\}$  representing the maximum emitted photon count, i.e., the maximum expected value of the number of photons emitted by the scene and hitting the image domain. Clearly, the lower  $\kappa$ , the lower the SNR of the observed noisy image and the more difficult the image restoration problem. For each image, several values  $\kappa$  ranging from 3 to 1000 have been considered. Then, the resulting images have been corrupted by space-invariant Gaussian blur, with a blur kernel generated by the Matlab routine `fspecial`, which is characterized by two parameters, namely the band parameter, representing the side length (in pixels) of the square support of the kernel, and `sigma`, which is the standard deviation (in pixels) of the isotropic bivariate Gaussian distribution defining the kernel in the continuous setting. We consider two different blur levels characterized by the parameters `band = 5, sigma = 1` and `band = 13, sigma = 3`. The blurred noiseless image  $\lambda = Hx + b$  is then generated by adding to the blurred image a constant emission background  $b$  of value  $2 \times 10^{-3}$ . The observed image  $y = \text{Poiss}(\lambda)$  is finally obtained by pseudo-randomly generating an  $m$ -variate independent Poisson realization with mean vector  $\lambda$ .

The black solid curves plotted in Figure 6a,c represent the function  $\mathcal{D}(\mu; y)$  as defined in (11) and (12) for the first test image `cameraman` and  $\kappa = 5$ , for the less severe (first row) and more severe (second row) blur level. They have been computed by solving the TV-KL model in (8) for a fine grid of different  $\mu$ -values, and then calculating  $\mathcal{D}(\mu; y)$  for each  $\mu$ . The horizontal dashed cyan and green lines represent the constant discrepancy values  $\Delta^{(T)}$  and  $\Delta^{(A)}$  used in (13) and (15), respectively, while the dashed magenta curve represents the discrepancy value function  $\Delta^{(NE)}(\mu)$  defined in (37). We remark that  $\Delta^{(NE)}(\mu)$  has been obtained in the same way as  $\mathcal{D}(\mu; y)$ , i.e., by computing the expression in (37) for each  $\mu$  of the selected fine grid. One can clearly observe that the intersection points between the curve  $\Delta^{(NE)}(\mu)$  and the function  $\mathcal{D}(\mu; y)$  and between the line representing  $\Delta^{(T)}$  and  $\mathcal{D}(\mu; y)$  are very close, and both at a significant distance from the intersection point detected by  $\Delta^{(A)}$ .

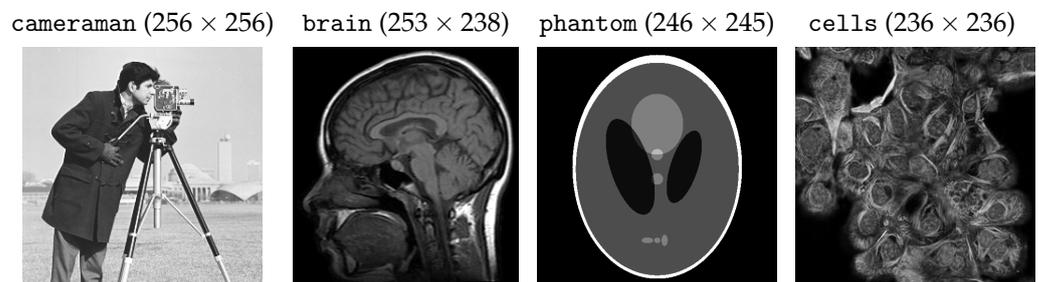


Figure 5. Grayscale test images considered for the numerical experiments.

In Figure 6b,d, we show the INSR and SSIM values achieved for different  $\mu$ -values with  $\kappa = 5$ . The vertical cyan, green, and magenta lines correspond to the  $\mu$ -values detected by the intersection of  $\mathcal{D}(\mu; y)$  and  $\Delta^{(T)}$ ,  $\Delta^{(A)}$ ,  $\Delta^{(NE)}(\mu)$ , respectively. As a reflection of the behavior of the discrepancy function and of the three curves, the INSR/SSIM corresponding to  $\mu^{(T)}$  and  $\mu^{(NE)}$  are very close to each other and almost reach the maximum of the two curves. We also highlight that, when considering the more severe blur case, the ADP selects

a very large  $\mu$ -value, which returns very low ISNR and SSIM values—see the thumbnail image in the right-hand corner of Figure 6d.

We are also interested in verifying that the proposed NEDP-ADMM scheme outlined in Algorithm 1 succeeds in automatically selecting such optimal  $\mu$  in a robust and efficient way: the blue and red markers in Figure 6b,d represent the final ISNR and SSIM values, respectively, of the image restored via NEDP-ADMM. Clearly, the markers are plotted in correspondence of  $\hat{\mu}^{(NE)}$ , which is, as we recall, the output  $\mu$ -value of the iterative scheme NEDP-ADMM; one can clearly observe that  $\hat{\mu}^{(NE)}$  is very close to the optimal  $\mu^{(NE)}$  detected *a posteriori* by the NEDP.

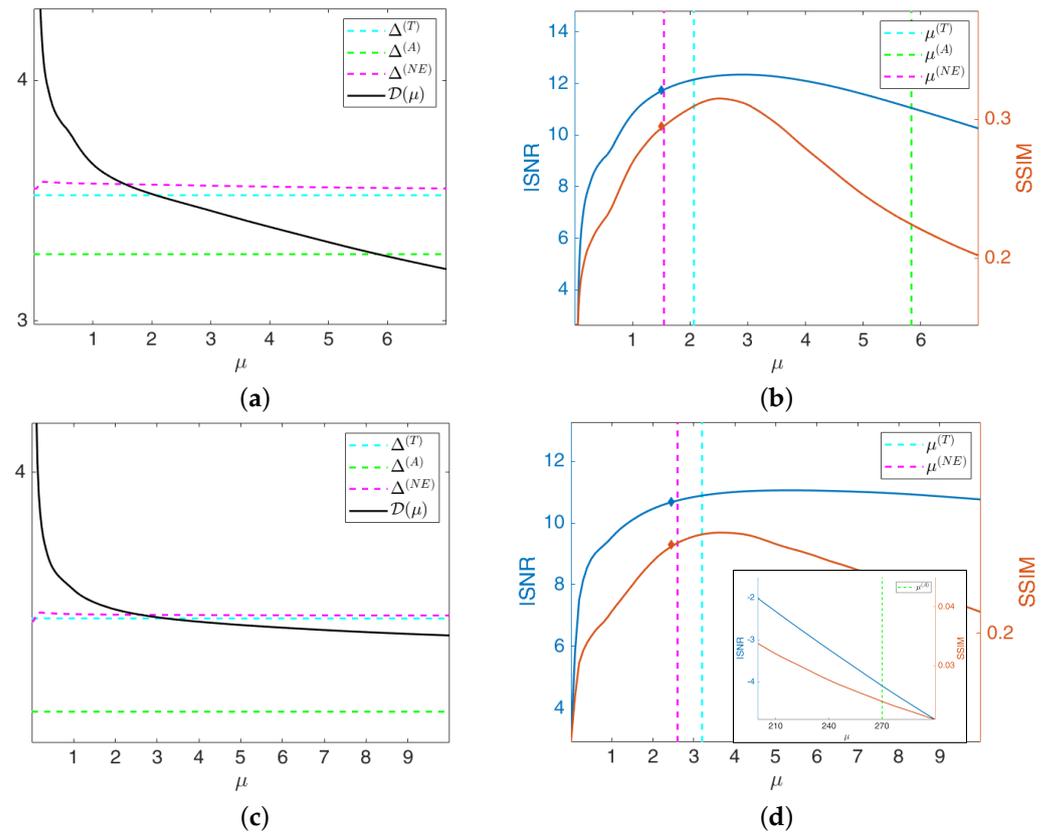
As a further analysis, at the bottom of Figure 6, we report the output  $\mu$ -values, the ISNR and the SSIM values for the two blur levels, and the 11  $\kappa$ -values considered to be obtained by the ADP-ADMM (first column) and the NEDP-ADMM (second column). To facilitate the comparison, we also report in blue/red the increments/decrements of the ISNR and SSIM achieved by our method with respect to the approximate criterion. Notice that the NEDP outperforms the ADP both in terms of ISNR and SSIM for the low-count acquisitions. However, when the  $\kappa$  increases, the two methods behave very similarly, with the ISNR and SSIM values obtained by the ADP-ADMM being slightly larger than those obtained by the NEDP-ADMM. In accordance with this analysis, the output  $\hat{\mu}^{(A)}$  and  $\hat{\mu}^{(NE)}$  are significantly different in low-count regimes, similar in mid-count regimes, and particularly close in high-count regimes. Notice that this behavior can be easily explained in light of the analysis carried out in Section 2, where we have shown that the approximation provided by  $\Delta^{(A)}$  becomes more and more accurate as the number of pixels with large values increases.

For a visual comparison, in Figures 7 and 8, we show the observed images (left column), the restorations via ADP-ADMM (middle column) and via NEDP-ADMM (right column) for the less and more severe blur level, respectively, and when different photon count regimes, ranging from very low to very high, are considered. As already observed from the ISNR and SSIM values reported at the bottom of Figure 6, we notice that for low-count acquisitions, the  $\mu$ -value selected by the ADP does not allow for a proper regularization, so that NEDP-ADMM clearly outperforms the competitor. However, starting from  $\kappa = 15$ —for the first blur level—and from  $\kappa = 40$ —for the second blur level—the two approaches return similar output images.

For the second test image, brain, we report in Figure 9 the behavior of the discrepancy function  $\mathcal{D}(\mu; \mathbf{y})$  and of the ISNR/SSIM curves obtained by applying the TDP, the ADP, and the NEDP, for  $\kappa = 5$  and for the two considered blur levels. In addition, in this case, the NEDP and the TDP behave similarly and they almost achieve the maximum of the ISNR and of the SSIM curves. In contrast,  $\mu^{(A)}$  appears to be largely underestimated with respect to the optimal  $\mu$ —which can be intended as the one maximizing either the ISNR or the SSIM. As for the first test image, the blue and red markers, indicating the output ISNR and SSIM, respectively, of the iterated version of our approach, are very close to the ones achieved by applying the NEDP *a posteriori*, suggesting that also  $\hat{\mu}^{(NE)}$  and  $\mu^{(NE)}$  are very close.

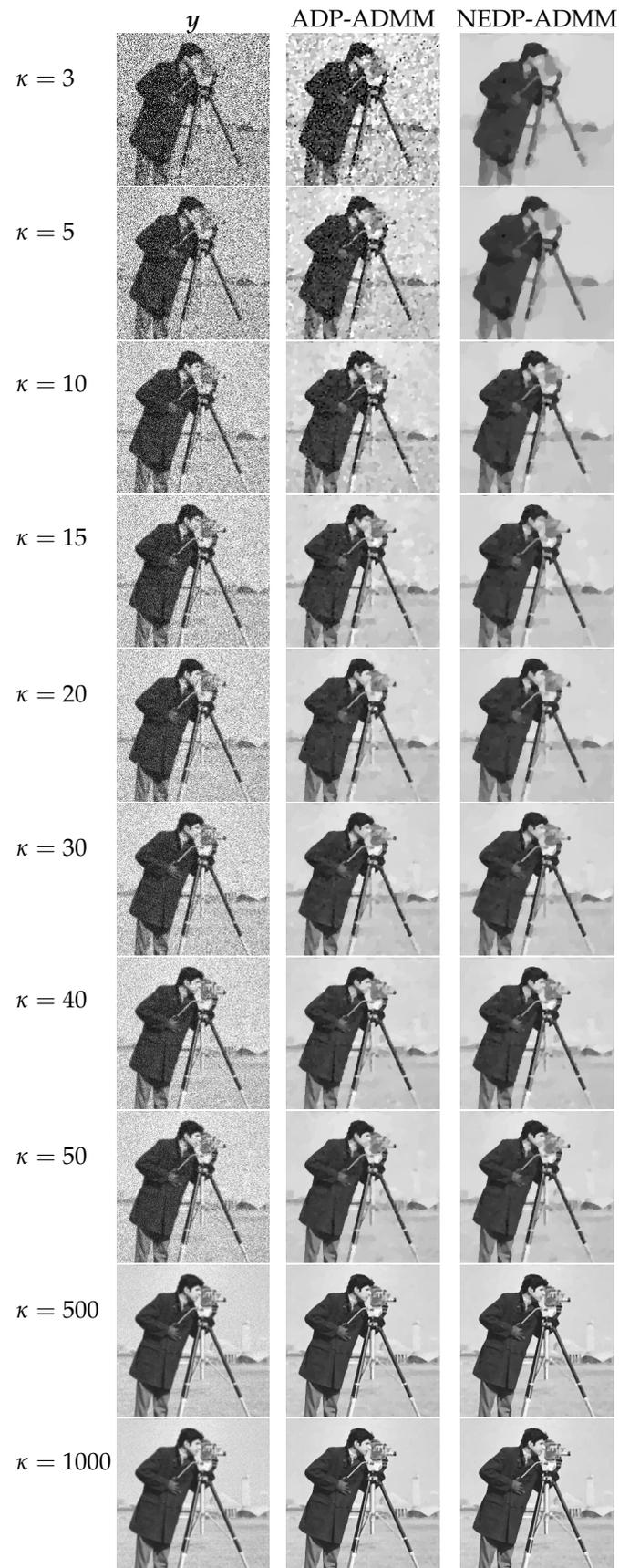
From the table reported at the bottom of Figure 9, we observe that the proposed  $\mu$ -selection criterion, for every  $\kappa$ , returns restored images outperforming the ones achieved via the ADP both in terms of ISNR and SSIM. The poor behavior of the ADP can be related to the nature of the processed image, which, either for the low-count or high-count acquisitions, presents few pixels with large values so that the approximation in (15) is particularly inaccurate. As a signal of this, note that the output  $\hat{\mu}^{(A)}$  is always smaller—or significantly smaller—than  $\hat{\mu}^{(NE)}$ .

The restored images in Figures 10 and 11 reflect the values recorded in the table as, for the two considered blur levels, the output of the NEDP-ADMM appears to be remarkably sharper than the final restoration by ADP-ADMM, especially in low- and mid-count regimes.

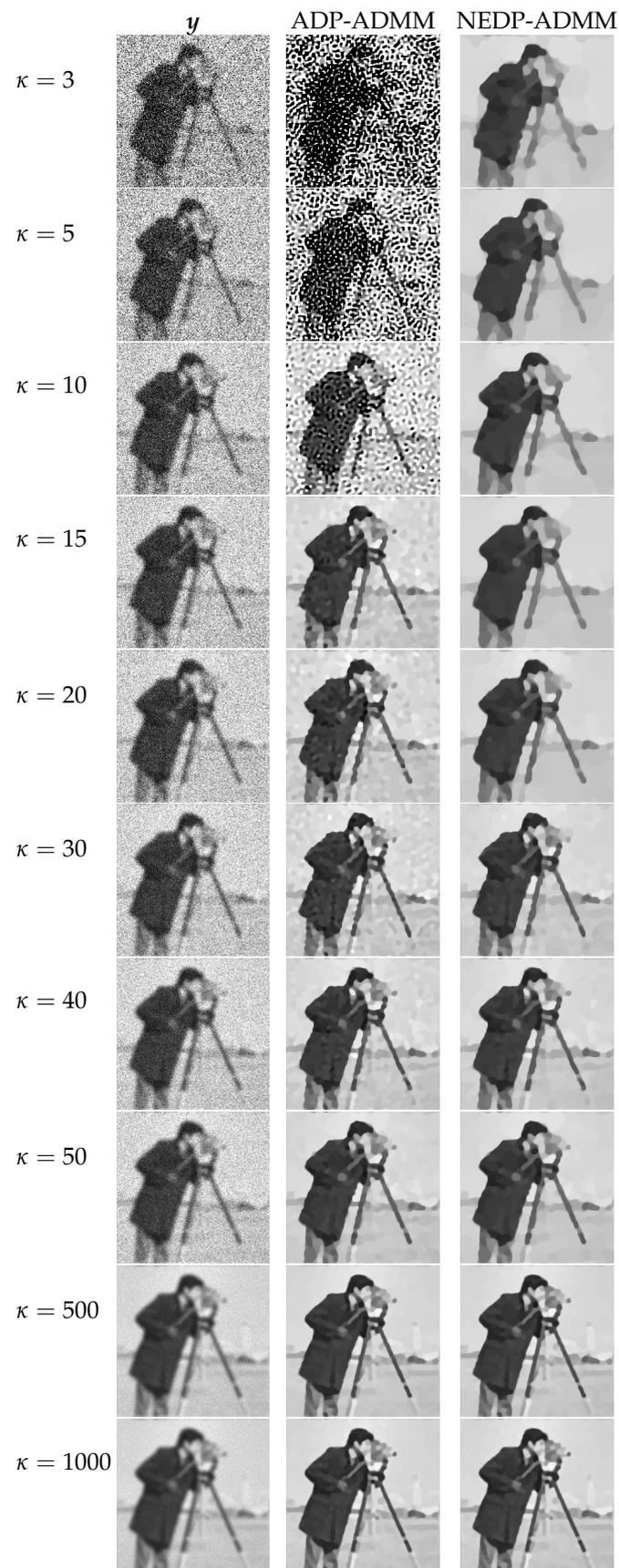


		ADP-ADMM		NEDP-ADMM			Difference		
	$\kappa$	$\hat{\mu}^{(A)}$	ISNR	SSIM	$\hat{\mu}^{(NE)}$	ISNR	SSIM	ISNR	SSIM
								ISNR	SSIM
band = 5, sigma = 1	3	6.1613	10.1928	0.1654	1.3228	13.4871	0.3720	+3.2942	+0.2065
	5	5.8076	11.0567	0.2247	1.5100	11.7258	0.2949	+0.6690	+0.0701
	10	6.5414	9.9561	0.2504	3.3778	10.0664	0.2540	+0.1103	+0.0035
	15	6.4521	8.9297	0.2624	4.2140	8.7557	0.2491	-0.1740	-0.0133
	20	6.7853	8.1429	0.2634	5.0019	7.9470	0.2482	-0.1959	-0.0152
	30	7.7725	6.9186	0.2780	6.6048	6.8030	0.2677	-0.1138	-0.0102
	40	9.5005	6.1588	0.2892	8.5250	6.0814	0.2832	-0.0774	-0.0060
	50	10.2516	5.5940	0.2982	9.5042	5.5270	0.2932	-0.0670	-0.0050
	100	15.3829	4.0231	0.3335	14.9409	3.9963	0.3316	-0.0267	-0.0019
	500	43.4069	2.5746	0.4148	43.2463	2.5705	0.4146	-0.0040	-0.0002
1000	70.4637	2.6296	0.4509	70.3529	2.6280	0.4508	-0.0015	-0.0001	
band = 13, sigma = 3	3	291.2252	-5.3677	0.0156	2.2867	12.5234	0.3214	+17.8914	+0.3058
	5	270.6214	-4.0339	0.0241	2.4535	10.6776	0.2393	+14.7115	+0.2152
	10	117.0970	2.8109	0.0888	4.2731	8.5199	0.1657	+5.7090	+0.0769
	15	20.0084	7.2967	0.1606	3.6564	6.9299	0.1322	-0.3667	-0.0283
	20	25.0534	6.4344	0.1554	6.7164	6.4093	0.1444	-0.0250	-0.0109
	30	35.2842	5.4458	0.1720	15.5595	5.5369	0.1659	+0.0911	-0.0061
	40	30.6417	4.8883	0.1785	16.5360	4.8324	0.1688	-0.0559	-0.0097
	50	21.0590	4.2611	0.1686	14.4928	4.1632	0.1607	-0.0978	-0.0078
	100	29.4505	3.0947	0.1868	25.1986	3.0469	0.1835	-0.0477	-0.0033
	500	75.9439	1.9758	0.2222	74.3578	1.9680	0.2216	-0.0078	-0.0006
1000	136.0108	1.9261	0.2378	134.6466	1.9233	0.2375	-0.0028	-0.0002	

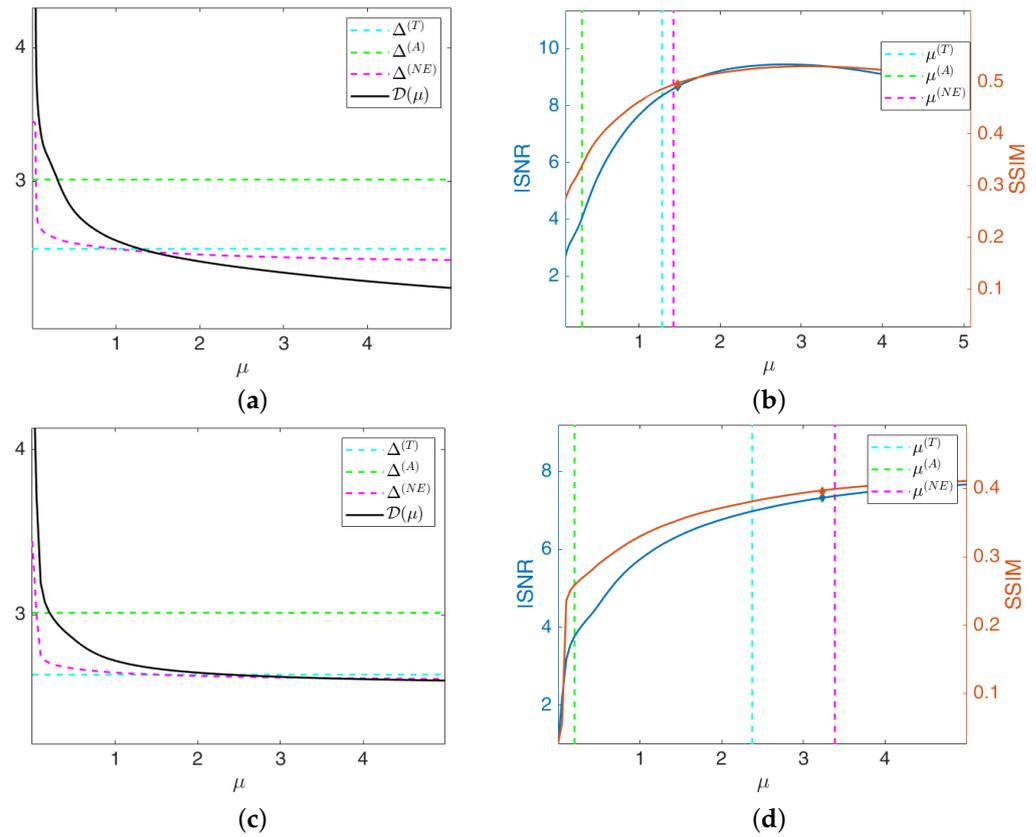
**Figure 6.** Test image cameraman. **Top:** discrepancy curve divided by  $10^4$  (a,c) and ISNR/SSIM values (b,d) achieved for different  $\mu$ -values with  $\kappa = 5$  and Gaussian blur with parameters band = 5, sigma = 1 (first row) and band = 13, sigma = 3 (second row). **Bottom:** output  $\mu$ -values and ISNR/SSIM values obtained by the ADP-ADMM and the NEDP-ADMM for the two blur levels considered and different photon counts  $\kappa$ .



**Figure 7.** Test image cameraman. **Left column:** observed data  $y$  corrupted by Gaussian blur with parameters  $\text{band} = 5$ ,  $\text{sigma} = 1$  and Poisson noise with different  $\kappa$ -values ranging from 3 to 1000. **Middle column:** restorations by ADP-ADMM. **Right column:** restorations by NEDP-ADMM.

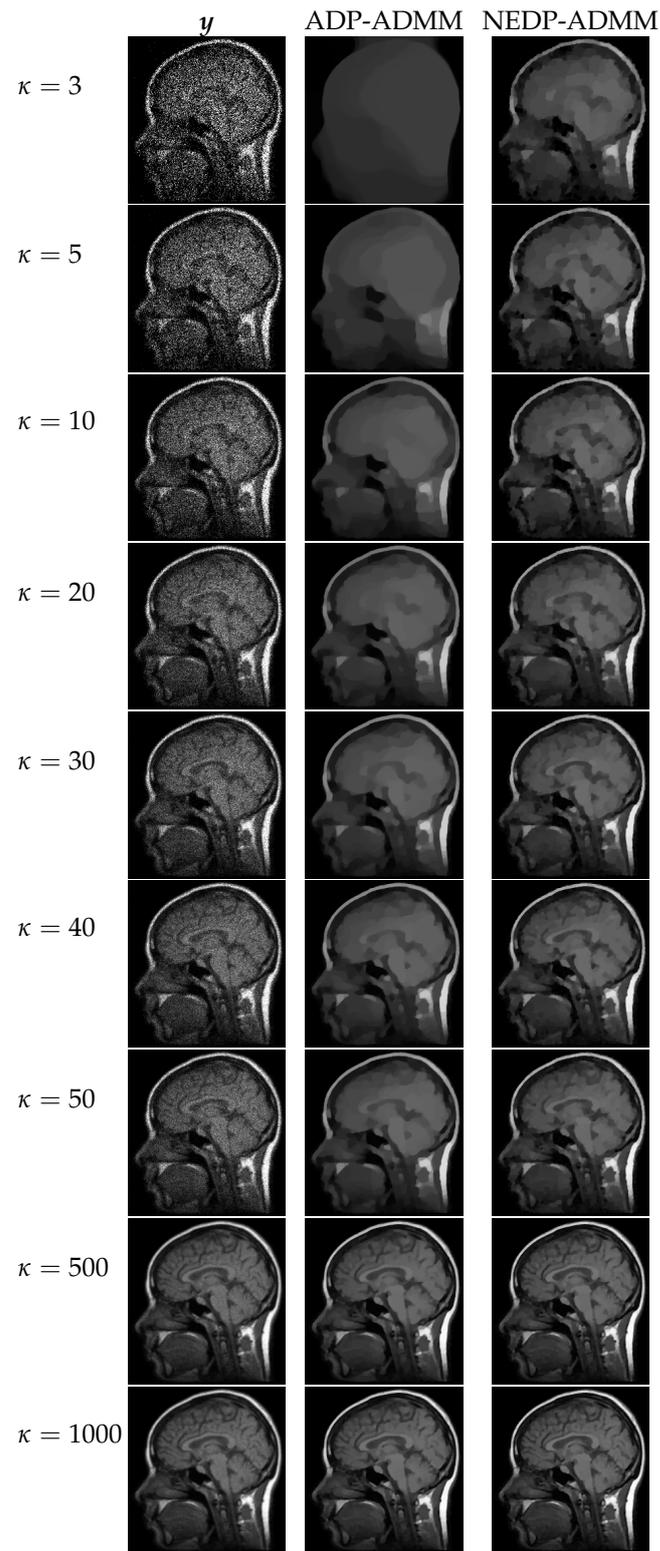


**Figure 8.** Test image cameraman. **Left** column: observed data  $y$  corrupted by Gaussian blur with parameters  $\text{band} = 13$ ,  $\text{sigma} = 3$  and Poisson noise with different  $\kappa$ -values ranging from 3 to 1000. **Middle** column: restorations by ADP-ADMM. **Right** column: restorations by NEDP-ADMM.

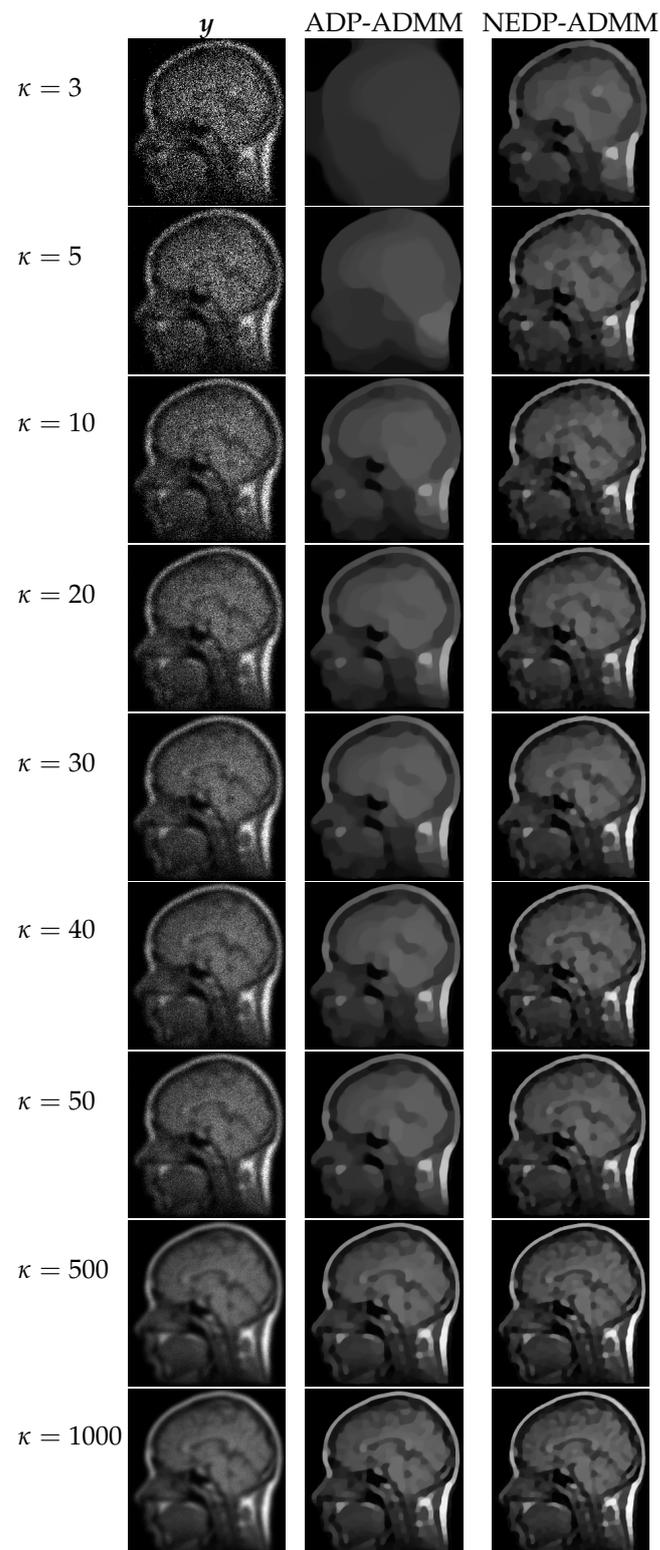


		ADP-ADMM			NEDP-ADMM			Difference	
	$\kappa$	$\hat{\mu}^{(A)}$	ISNR	SSIM	$\hat{\mu}^{(NE)}$	ISNR	SSIM	ISNR	SSIM
	band = 5, sigma = 1	3	0.0580	4.1577	0.2297	1.0580	9.7776	0.4793	+5.6199
5		0.3028	4.0752	0.3382	1.4745	8.6706	0.4955	+4.5953	+0.1573
10		0.6460	3.3797	0.4017	2.3046	7.0980	0.5409	+3.7182	+0.1392
20		1.0616	2.1217	0.4586	3.1936	5.3490	0.5795	+3.2272	+0.1209
30		1.4671	1.5826	0.4991	4.1296	4.4531	0.6125	+2.8705	+0.1133
40		1.8886	1.3450	0.5308	5.2176	4.0403	0.6381	+2.6953	+0.1073
50		2.2322	1.0878	0.5508	5.8860	3.5566	0.6532	+2.4688	+0.1024
100		3.8521	0.4355	0.6109	8.7490	2.3912	0.6980	+1.9557	+0.0871
500		12.9452	0.3040	0.7657	24.1260	1.5161	0.8142	+1.2121	+0.0485
1000		20.7261	0.6053	0.8150	36.1662	1.5187	0.8459	+0.9134	+0.0309
band = 13, sigma = 3	3	0.0560	4.2295	0.1567	1.3193	7.9382	0.3751	+3.7086	+0.2184
	5	0.2075	3.7819	0.2590	3.2337	7.3272	0.3957	+3.5452	+0.1367
	10	0.6767	2.6315	0.2816	5.9275	5.7485	0.4230	+3.1169	+0.1413
	20	1.0271	1.2943	0.2991	6.3652	3.9673	0.4288	+2.6730	+0.1296
	30	1.3701	0.8786	0.3147	7.1293	3.1839	0.4316	+2.3053	+0.1168
	40	1.8293	0.8087	0.3387	12.7947	3.3898	0.4676	+2.5811	+0.1289
	50	2.0648	0.6481	0.3482	11.1175	2.9012	0.4628	+2.2531	+0.1146
	100	3.7821	0.7344	0.3928	24.5532	2.9241	0.5036	+2.1897	+0.1108
	500	13.5166	1.6380	0.4853	56.1196	3.0920	0.5560	+1.4539	+0.0706
	1000	22.7867	2.0293	0.4857	82.6358	3.2317	0.5351	+1.2024	+0.0493

**Figure 9.** Test image brain. **Top:** discrepancy curve divided by  $10^4$  (a,c) and ISNR/SSIM values (b,d) achieved for different  $\mu$ -values with  $\kappa = 5$  and Gaussian blur with parameters band = 5, sigma = 1 (first row) and band = 13, sigma = 3 (second row). **Bottom:** output  $\mu$ -values and ISNR/SSIM values obtained by the ADP-ADMM and the NEDP-ADMM for the two blur levels considered and different photon counts  $\kappa$ .



**Figure 10.** Test image brain. **Left column:** observed data  $y$  corrupted by Gaussian blur with parameters band = 5, sigma = 1 and Poisson noise with different  $\kappa$ -values ranging from 3 to 1000. **Middle column:** restorations by ADP-ADMM. **Right column:** restorations by NEDP-ADMM.



**Figure 11.** Test image brain. **Left column:** observed data  $y$  corrupted by Gaussian blur with parameters band = 13, sigma = 3 and Poisson noise with different  $\kappa$ -values ranging from 3 to 1000. **Middle column:** restorations by ADP-ADMM. **Right column:** restorations by NEDP-ADMM.

In Figure 12, for the test image phantom, we report the curve of the discrepancy function  $\mathcal{D}(\mu; y)$  obtained *a posteriori*, as well as the curves of the ISNR and of the SSIM for  $\kappa = 3$  and the two blur levels considered. As for the test image brain, in this case, the ADP also selects a  $\mu$ -value that is far from the optimal one, either if measured in terms of

ISNR or SSIM. On the other hand,  $\mu^{(T)}$  and  $\mu^{(NE)}$  are very close and, in particular, one can notice that the output of the NEDP-ADMM represents the optimal compromise in terms of ISNR and SSIM. Notice also that, when considering the larger blur level, the output value  $\hat{\mu}^{(NE)}$  of the NEDP-ADMM, detected by the red and blue markers, is larger than the one selected by the *a posteriori* version of the NEDP. However, the difference in terms of ISNR and SSIM is not particularly significant. This behavior is due to the use of different penalty parameters  $\beta_\lambda, \beta_g, \beta_z$  in the ADMM for the *a posteriori* and the iterated version of our approach. In fact, when considering a large blur level, the convergence of the ADMM is particularly slow and can be achieved upon suitable selection of the penalty parameters, whose values may not coincide in the two scenarios addressed.

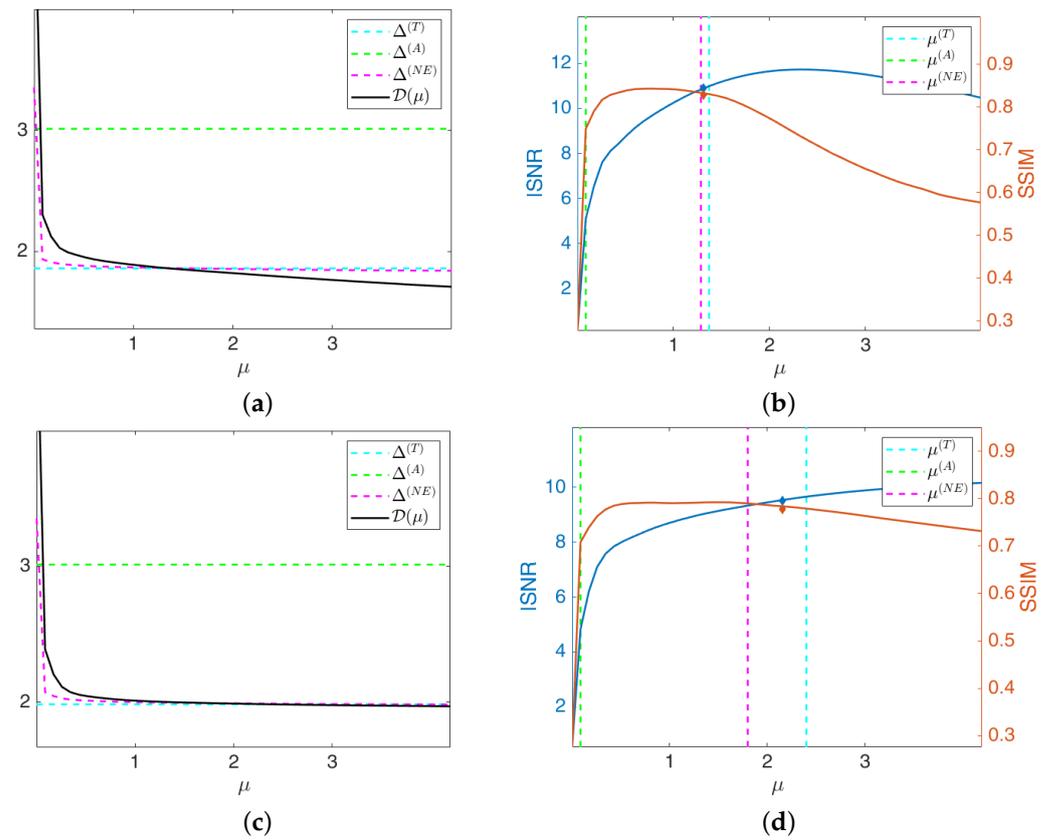
The mismatch observed in Figure 12 between the curves of the ISNR and of the SSIM is reflected in the values reported in the bottom part of the figure, whence we can conclude that the NEDP-ADMM outperforms the ADP-ADMM in terms of ISNR for each  $\kappa$ -value, while the ADP-ADMM returns slightly better results in terms of SSIM for high-count acquisitions. As for the test image *brain*, in this case, the output  $\hat{\mu}^{(A)}$  also appears to be significantly small. Once again, this behavior can be related to the considered image, which is mostly characterized by pixels with very small values.

From the restorations shown in Figures 13 and 14, one can also notice that the slight improvement in terms of SSIM does not correspond to any significant visual improvements. In fact, along the whole range of considered photon counts  $\kappa$ , the NEDP-ADMM is capable of returning sharper restorations. This reflects the tendency of ADP-ADMM applied on the current test image in selecting not sufficiently large  $\mu$ -values, so that, in the TV-KL, the regularization term takes over. We also remark that for the current test image, the SSIM value does not seem to be particularly meaningful.

For the fourth and final test image, *cells*, we show in Figure 15 the behavior of the discrepancy function  $\mathcal{D}(\mu; \mathbf{y})$ , as well as of the ISNR and SSIM values in the *a posteriori* framework for the two blur levels and  $\kappa = 3$ . Note that, in the *a posteriori* setting, for both blur levels, the NEDP achieves higher ISNR and SSIM values when compared to the ADP. However, we observe that when considering the larger blur level, the output  $\hat{\mu}^{(NE)}$  of the NEDP-ADMM is smaller than  $\mu^{(NE)}$ ; this behavior can be ascribed, once again, to the ADMM convergence issues and the different values selected for the penalty parameters.

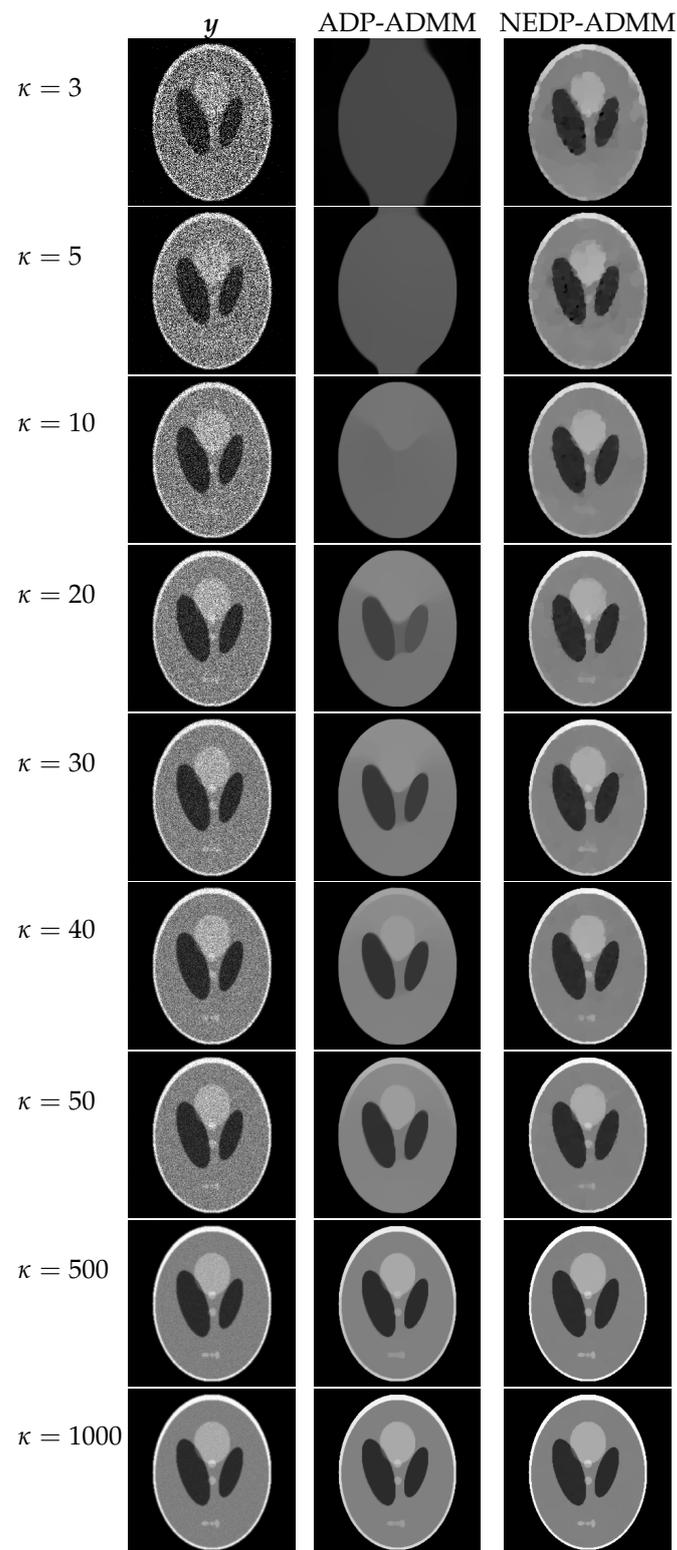
From the values reported in the bottom part of Figure 15, we notice that the NEDP-ADMM outperforms the ADP-ADMM in every photon count regime. Clearly, the closer  $\hat{\mu}^{(A)}$  and  $\hat{\mu}^{(NE)}$ , the smaller the difference in terms of ISNR and SSIM.

The restorations computed by the ADP-ADMM and the NEDP-ADMM are shown in Figure 16 for the smaller blur level and in Figure 17 for the larger one. The obtained results confirm the values reported in the bottom of Figure 15. Moreover, also from a visual viewpoint, the difference between the two performances increases when going from high- to low-count regimes.

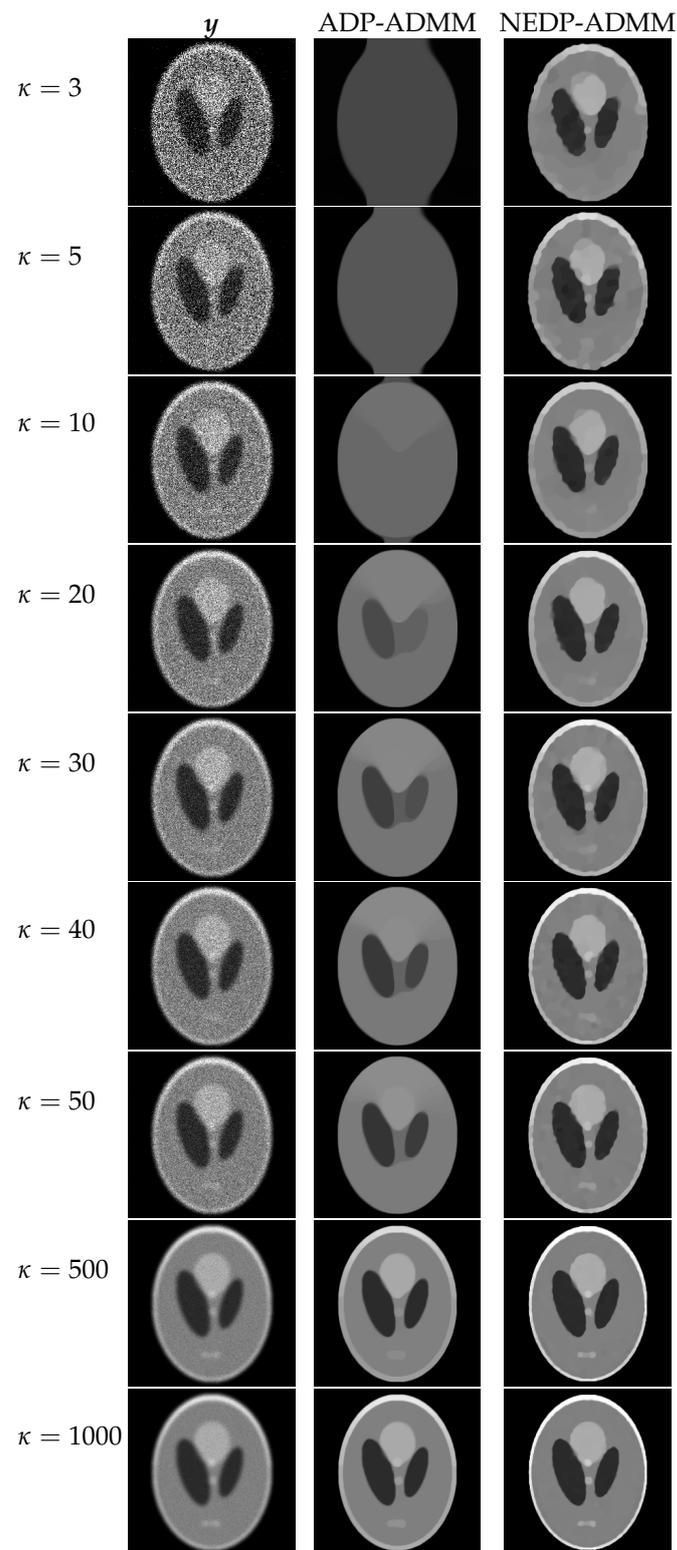


		ADP-ADMM		NEDP-ADMM		Difference			
	$\kappa$	$\hat{\mu}^{(A)}$	ISNR	SSIM	$\hat{\mu}^{(NE)}$	ISNR	SSIM	ISNR	SSIM
band = 5, sigma = 1	3	0.0251	2.4956	0.4731	1.3116	10.8958	0.8290	+8.4002	+0.3559
	5	0.0499	1.8412	0.6714	2.0924	10.0898	0.7928	+8.2486	+0.1213
	10	0.1226	0.6924	0.7374	2.5823	8.1711	0.8169	+7.4787	+0.0795
	20	0.2321	-0.0124	0.7611	4.1877	7.2274	0.7739	+7.2398	+0.0127
	30	0.3824	-0.5303	0.7660	5.2651	6.7886	0.7554	+7.3189	-0.0106
	40	0.5727	-0.7640	0.7775	6.4738	6.6109	0.7426	+7.3749	-0.0349
	50	0.7566	-0.7936	0.7845	6.9334	6.4298	0.7631	+7.2235	-0.0214
	100	1.4043	-0.6231	0.7864	10.8129	6.2304	0.7138	+6.8536	-0.0725
	500	4.4672	1.3594	0.8054	27.1367	7.2687	0.6776	+5.9093	-0.1277
	1000	7.0698	2.8491	0.8413	41.5306	8.1662	0.6787	+5.3171	-0.1625
band = 13, sigma = 3	3	0.0246	2.6822	0.3757	2.1562	9.5223	0.7783	+6.8400	+0.4025
	5	0.0449	1.9322	0.6773	3.8968	8.3637	0.7373	+6.4314	+0.0600
	10	0.1201	1.0697	0.6922	3.3357	5.9521	0.7377	+4.8824	+0.0455
	20	0.2047	0.4592	0.6964	5.8572	4.9162	0.7304	+4.4570	+0.0339
	30	0.2833	0.2225	0.7021	11.4953	4.7800	0.6728	+4.5574	-0.0292
	40	0.3716	0.0590	0.7066	18.2478	5.2786	0.6487	+5.2195	-0.0579
	50	0.4720	0.0193	0.7029	16.9469	4.7202	0.6811	+4.7008	-0.0217
	100	1.0801	0.0283	0.7097	29.7678	4.9542	0.6405	+4.9259	-0.0692
	500	4.753	1.3551	0.7235	75.4425	6.3189	0.6341	+4.9638	-0.0893
	1000	8.9802	2.3915	0.7536	124.9501	7.1415	0.6256	+4.7500	-0.1280

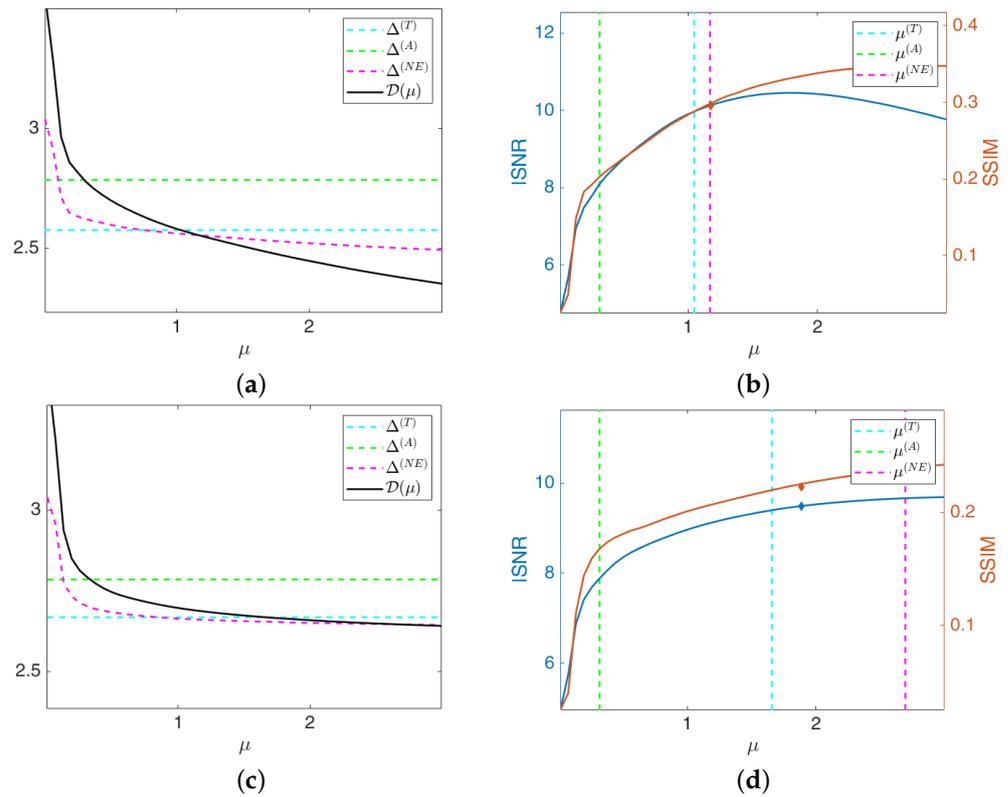
**Figure 12.** Test image phantom. **Top:** discrepancy curve divided by  $10^4$  (a,c) and ISNR/SSIM values (b,d) achieved for different  $\mu$ -values with  $\kappa = 3$  and Gaussian blur with parameters band = 5, sigma = 1 (first row) and band = 13, sigma = 3 (second row). **Bottom:** output  $\mu$ -values and ISNR/SSIM values obtained by the ADP-ADMM and the NEDP-ADMM for the two blur levels considered and different photon counts  $\kappa$ .



**Figure 13.** Test image phantom. **Left** column: observed data  $y$  corrupted by Gaussian blur with parameters band = 5, sigma = 1 and Poisson noise with different  $\kappa$ -values ranging from 3 to 1000. **Middle** column: restorations by ADP-ADMM. **Right** column: restorations by NEDP-ADMM.

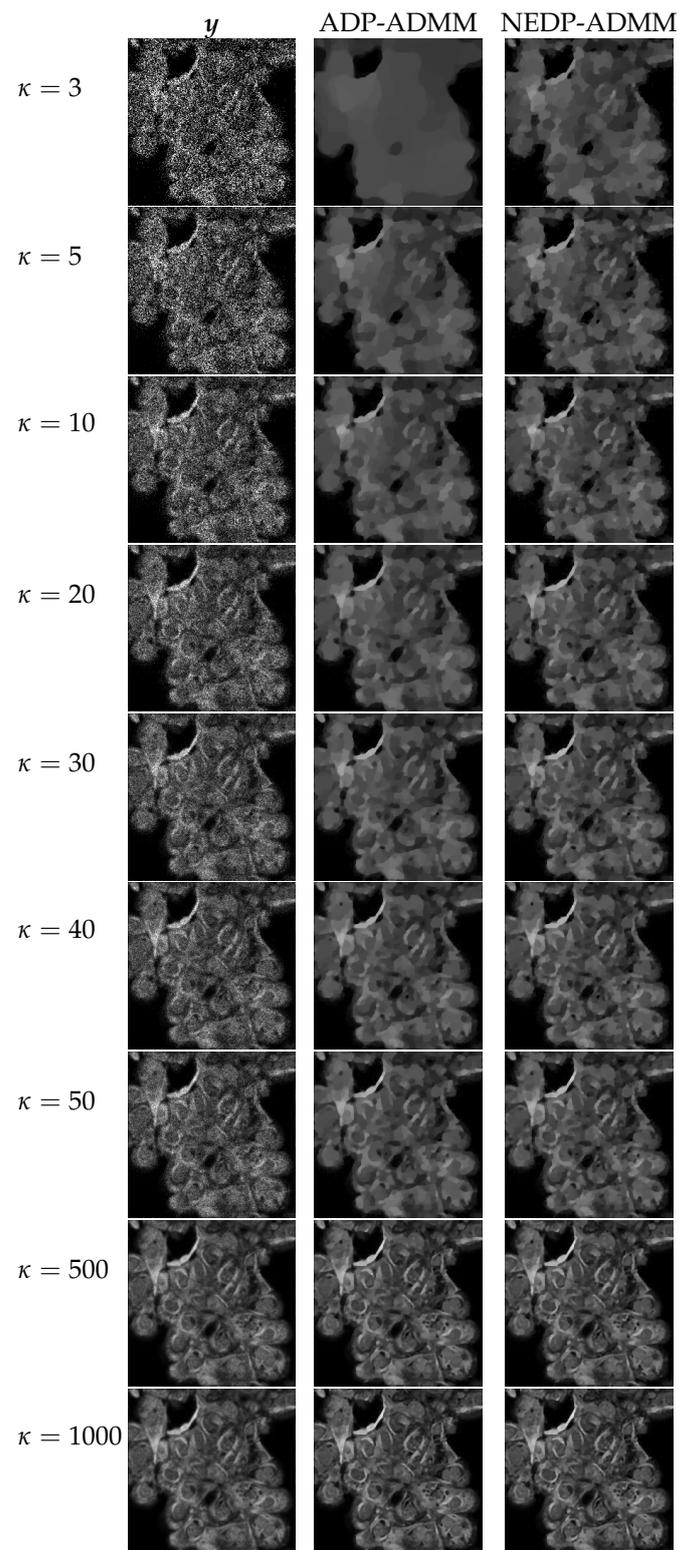


**Figure 14.** Test image phantom. **Left** column: observed data  $y$  corrupted by Gaussian blur with parameters  $\text{band} = 13$ ,  $\text{sigma} = 3$  and Poisson noise with different  $\kappa$ -values ranging from 3 to 1000. **Middle** column: restorations by ADP-ADMM. **Right** column: restorations by NEDP-ADMM.

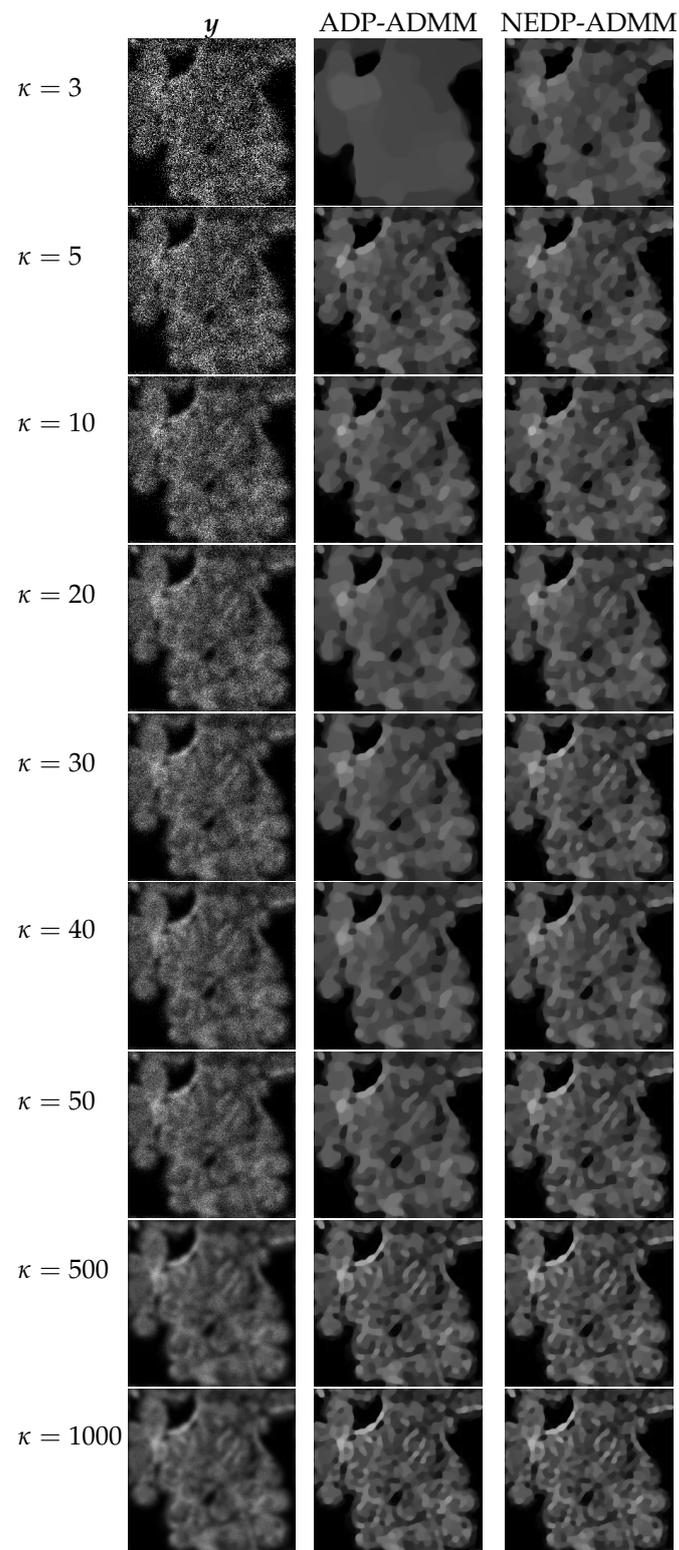


	ADP-ADMM				NEDP-ADMM			Difference	
	$\kappa$	$\hat{\mu}^{(A)}$	ISNR	SSIM	$\hat{\mu}^{(NE)}$	ISNR	SSIM	ISNR	SSIM
band = 5, sigma = 1	3	0.3042	8.0428	0.2002	1.1719	10.1281	0.2952	+2.0852	+0.0949
	5	1.1283	8.0419	0.2799	1.8656	8.7165	0.3295	+0.6745	+0.0496
	10	1.7170	5.9522	0.3167	2.6968	6.5660	0.3643	+0.6137	+0.0475
	20	2.4343	3.8227	0.3698	3.6823	4.4218	0.4155	+0.5991	+0.0457
	30	3.2547	2.9204	0.3999	4.9249	3.4823	0.4449	+0.5619	+0.0449
	40	4.1041	2.4395	0.4382	5.8986	2.9103	0.4759	+0.4707	+0.0376
	50	5.0053	2.0792	0.4655	7.0349	2.4818	0.4991	+0.4026	+0.0335
	100	8.4224	1.0624	0.5383	10.9451	1.3581	0.5629	+0.2957	+0.0246
	500	27.1550	0.3788	0.6718	32.0158	0.5370	0.6827	+0.1581	+0.0108
	1000	45.5340	0.6410	0.7221	52.7334	0.7729	0.7310	+0.1319	+0.0089
band = 13, sigma = 3	3	0.3251	7.9171	0.16940	1.8841	9.4912	0.2228	+1.5740	+0.0534
	5	3.6943	8.1023	0.2420	3.9060	8.1195	0.2436	+0.0171	+0.0015
	10	3.5550	5.7064	0.2282	5.2098	5.8821	0.2425	+0.1756	+0.0143
	20	2.9412	3.5252	0.2285	5.6999	3.9632	0.2586	+0.4380	+0.0301
	30	4.1855	2.7059	0.2396	10.8952	3.2246	0.2798	+0.5186	+0.0402
	40	4.3731	2.0708	0.2450	8.6807	2.4853	0.2744	+0.4145	+0.0293
	50	6.1442	1.9499	0.2654	14.4495	2.3613	0.2980	+0.4114	+0.0325
	100	10.0409	1.2177	0.2862	18.6804	1.52056	0.3094	+0.3028	+0.0231
	500	37.5732	0.9416	0.3546	58.8299	1.1146	0.3698	+0.1730	+0.0152
	1000	64.4743	1.0410	0.3871	98.0256	1.1868	0.4019	+0.1458	+0.0148

**Figure 15.** Test image cells. **Top:** discrepancy curve divided by  $10^4$  (a,c) and ISNR/SSIM values (b,d) achieved for different  $\mu$ -values with  $\kappa = 3$  and Gaussian blur with parameters band = 5, sigma = 1 (**first row**) and band = 13, sigma = 3 (**second row**). **Bottom:** output  $\mu$ -values and ISNR/SSIM values obtained by the ADP-ADMM and the NEDP-ADMM for the two blur levels considered and different photon counts  $\kappa$ .



**Figure 16.** Test image cells. **Left column:** observed data  $y$  corrupted by Gaussian blur with parameters  $\text{band} = 5$ ,  $\text{sigma} = 1$  and Poisson noise with different  $\kappa$ -values ranging from 3 to 1000. **Middle column:** restorations by ADP-ADMM. **Right column:** restorations by NEDP-ADMM.



**Figure 17.** Test image cells. **Left column:** observed data  $y$  corrupted by Gaussian blur with parameters  $\text{band} = 13$ ,  $\text{sigma} = 3$  and Poisson noise with different  $\kappa$ -values ranging from 3 to 1000. **Middle column:** restorations by ADP-ADMM. **Right column:** restorations by NEDP-ADMM.

## 6. Conclusions and Future Work

We propose an automatic selection strategy for the regularization parameter of variational image restoration models under Poisson noise corruption based on a nearly exact version of the approximate discrepancy principle originally proposed in [5]. Our approach

relies on Monte Carlo simulations, which have been designed with the purpose of providing meaningful insights into the limitations of the original approximate strategy, especially in the low-count Poisson regime. The proposed version of the discrepancy principle has then been derived by means of a weighted least-square fitting and embedded along the iterations of an efficient ADMM-based optimization scheme. Our approach has been extensively tested on different images and for different photon count values, ranging from very low to high values. When compared to the original approximate selection criterion, the proposed strategy has been shown to drastically improve the quality of the output restorations in low-count regimes and in mid-count/high-count regimes on images characterized by few large pixel values.

From an analytical point of view, investigating the uniqueness of the regularization parameter value satisfying the proposed discrepancy principle will certainly constitute future work. From a modeling and applicative perspective, the effectiveness of the proposed approach when applied to variational models containing regularizers other than TV or aimed at solving inverse problems other than image restoration will be the subject of future analysis. Finally, from an algorithmic viewpoint, a matter that deserves further investigation is the (possibly automatic) selection of the three ADMM penalty parameters, which can significantly affect the speed of convergence of the numerical solution scheme.

**Author Contributions:** F.B., A.L., M.P., F.S. contributed in equal parts to the conceptualisation, methodology, writing and editing of the manuscript. F.B. was the main contributor for numerical implementation, data curation, validation, and visualisation. All authors contributed to investigation and resources. All authors have read and agreed to the published version of the manuscript.

**Funding:** Research was supported by the ‘National Group for Scientific Computation (GNCS-INDAM)’ and by ex60 project by the University of Bologna ‘Funds for selected research topics’. The research of MP has been also funded by the GNCS-INDAM ‘Fund for young researchers’.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Bertero, M.; Boccacci, P.; Ruggiero, V. *Inverse Imaging with Poisson Data*; IOP Publishing: Bristol, UK, 2018.
- Calvetti, D.; Somersalo, E. *Introduction to Bayesian Scientific Computing: Ten Lectures on Subjective Computing (Surveys and Tutorials in the Applied Mathematical Sciences)*; Springer: Berlin/Heidelberg, Germany, 2007.
- Geman, S.; Geman, D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* **1984**, *6*, 721–741. [[CrossRef](#)] [[PubMed](#)]
- Rudin, L.I.; Osher, S.; Fatemi, E. Nonlinear total variation based noise removal algorithms. *Physica D* **1992**, *60*, 259–268. [[CrossRef](#)]
- Zanella, R.; Boccacci, P.; Zanni, L.; Bertero, M. Efficient gradient projection methods for edge-preserving removal of Poisson noise. *Inverse Probl.* **2009**, *25*, 0450100; Erratum in *Inverse Probl.* **2013**, *29*, 119501. [[CrossRef](#)]
- Carlavan, M.; Blanc-Feraud, L. Sparse Poisson Noisy Image Deblurring. *IEEE Trans. Image Process.* **2012**, *21*, 1834–1846. [[CrossRef](#)] [[PubMed](#)]
- Bertero, M.; Boccacci, P.; Talenti, G.; Zanella, R.; Zanni, L. A discrepancy principle for Poisson data. *Inverse Probl.* **2010**, *26*, 105004. [[CrossRef](#)]
- Bonettini, S.; Prato, M. Accelerated gradient methods for the X-ray imaging of solar flares. *Inverse Probl.* **2014**, *30*, 055004. [[CrossRef](#)]
- Benvenuto, F. A study on regularization for discrete inverse problems with model-dependent noise. *SIAM J. Numer. Anal.* **2017**, *55*, 2187–2203. [[CrossRef](#)]
- Guastavino, S.; Benvenuto, F. A mathematical model for image saturation with an application to the restoration of solar images via adaptive sparse deconvolution. *Inverse Probl.* **2021**, *37*, 0150104. [[CrossRef](#)]
- Zanni, L.; Benfenati, A.; Bertero, M.; Ruggiero, V. Numerical methods for parameter estimation in Poisson data inversion. *J. Math. Imaging Vis.* **2015**, *52*, 397–413. [[CrossRef](#)]
- Sixou, B.; Hohweiller, T.; Ducros, N. Morozov principle for Kullback-Leibler residual term and Poisson noise. *Inverse Probl. Imaging* **2018**, *12*, 607–634. [[CrossRef](#)]
- Johnson, N.L.; Kempas, A.W.; Kotz, S. *Univariate Discrete Distributions*; Wiley: New York, NY, USA, 2005.

14. Teuber, T.; Steidl, G.; Chan, R.H. Minimization and parameter estimation for seminorm regularization models with  $I$ -divergence constraints. *Inverse Probl.* **2013**, *29*, 035007. [[CrossRef](#)]
15. Boyd, S.P.; Parikh, N.; Chu, E.K.; Peleato, B.; Eckstein, J. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* **2011**, *3*, 1–122. [[CrossRef](#)]
16. Glowinski, R.; Marroco, A. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires. *Math. Model. Numer. Anal.* **1975**, *9*, 41–76. [[CrossRef](#)]
17. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [[CrossRef](#)] [[PubMed](#)]